Analysis on PANSS Dataset

STATS 202: Data Mining and Analysis, Final Project

Tianyu Du

August 7, 2019

Notes

- I am using my real name as user name on Kaggle;
- This report is 10-page long without tables and figures;
- All source codes are on Github.

Contents

1 Introduction

The Positive and Negative Syndrome Scale (PANSS) score is widely used as a measure for schizophrenia and other disorders in clinical trials. PANSS scores are collected by trained raters and reported by patients or their relatives. One assessment of PANSS scores consists 30 sub-scores from 3 sub-categories: 7 positive scores, 7 negative scores, and 16 general scores. Every score ranges from 1 to 7 denoting increasing levels of psychopathology. The aggregation of all 30 scores provides a detailed assessment of patient's current psychological status.

The entire dataset consists of five different studies ranging from study A to study E. In this practice, the first four datasets were used as a training to fit, select, and evaluate models. Then, these selected models were used to recover missing data in study E.

2 Treatment Effects

This section is devoted to analyze whether the treatment assigned led to significant improvements on patients' psychological status. Because the 18-th week assessments were missing in the dataset of study E, in this section, only data from study A to D were considered. There were 20947 observations (assessments) from above-mentioned dataset, in which 10524 observations came from patients assigned to the control group, and the remaining 10423 observations were from participants belonged to the treatment group. The evenly-split dataset allowed us to deploy various models to analyze whether there exists significant treatment effects.

Multiple evidences were found to support that there were indeed no treatment effect in this study. Firstly, the effects on four aggregate scores were analyzed, namely sum of all PANSS scores, and sums of scores from positive (`P_Total`), negative (`N_Total`), and general (`G_Total`) sub-categories respectively.

One challenge associated with treatment effect analysis was the initial status of patients in different groups. In some cases, the *prior* psychological status for a randomly selected patient from the treatment group is expected to be different from that of a random patient in the control group. In these cases, even if significant differences in PANSS scores *posterior* to the treatment were implied, one would not be able to distinguish whether the "effect" came from the discrepancy in the prior distributions or the treatment, and the treatment effect was not well-identifies. Figure 1 below presents the distributions of the four aggregate scores at day 0 visit. Both groups shared

similar histograms in terms of all four metrics. Moreover, the estimated kernel densities collided, which provided further evidence that there were no significant prior discrepancies between patients from these two groups. Therefore, one can conclude that most posterior differences in PANSS metrics were resulted from the treatment assigned instead of the prior discrepancy.

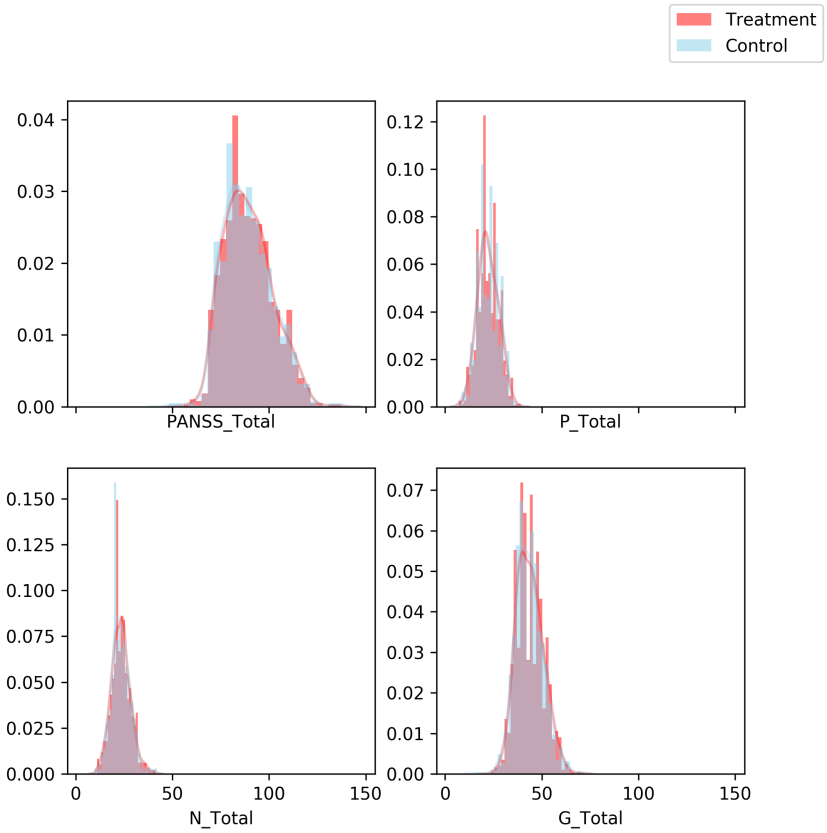

Figure 1: Distributions of Aggregate Scores at Day 0

The evolving path of these four above mentioned metrics over time could provide reasonable proxies to the treatment effect. As mentioned before, we believed the data failed to provide sufficient evidence to support the existence of treatment effect. Let $t \in \mathbb{N}$ denote the `VisitDay` variable in the dataset, t ranges from 0 to 480 in the complete dataset. Figure 2 presents the distribution of t, the 95-percent quantile was located at $t_{95\%} = 297$. Therefore, the top 5% of observations occupied more than on third of the total range of t, these observations were potentially troublesome as outliers. To deal with this issue, the top 5% observations were excluded in following analysis.

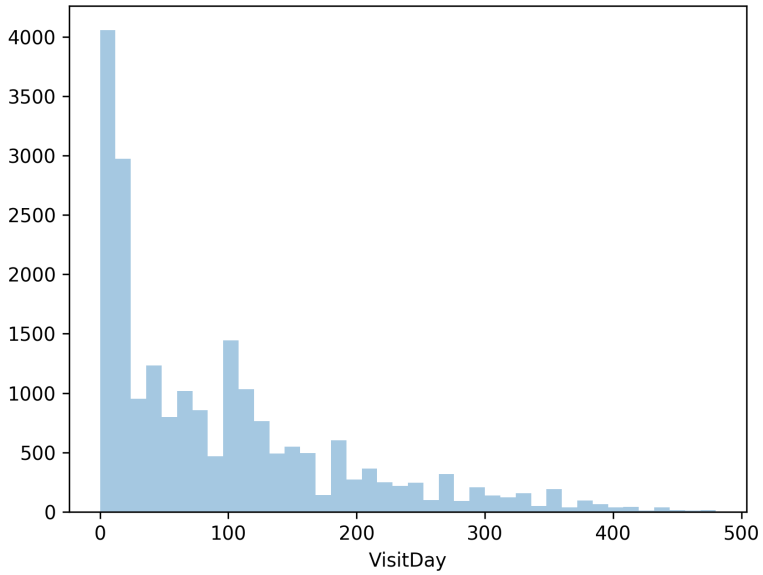

Figure 2: Distribution of `VisitDay`

The general strategy to identify treatment effect utilized linear models. Let Y denote the target metric, which takes value from the four aggregate PANSS metrics: $\mathcal{M} := \{\Sigma_{PANSS}, \Sigma_P, \Sigma_N, \Sigma_G\}$. Let X denote the set other characteristics of the patient. And then a hybrid linear model was fit:

$$Y = f_0(X, t) + \mathbb{1}\{\texttt{Treatment}\} f_1(X, t) \tag{1}$$

where f_0 and f_1 are two additive linear models, so that f_0 captures the evolving path of Y over time for the population of the control group, and the additive term f_1 measures the discrepancy of the treatment group. If the treatment dummy variable is statistically significant, then one can conclude the existence of treatment effect.

2.1 Patterns of PANSS over Time

Figure 3 below shows the scatter plot of Σ_{PANSS} scores against time for both groups with corresponding locally weighted linear regression estimations (LOWESS). The LOWESS for the treatment and control groups collided almost perfectly, which means the trends for both group estimated from non-parametric model were essentially identical. This provides preliminary evidence supporting our claim that there was no significant treatment effect.

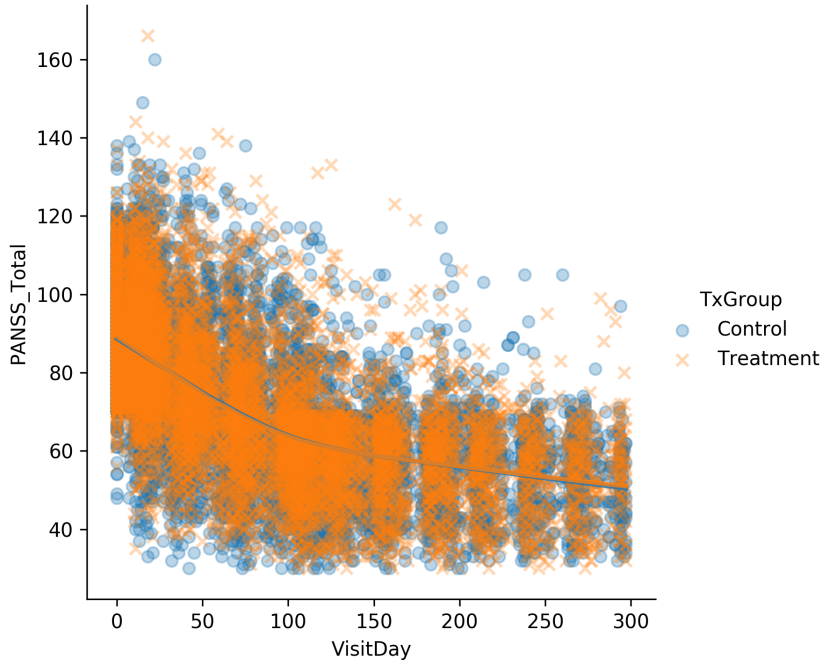

Figure 3: Changes of `PANSS_Total` Over Time and LOWESS

2.2 Identifying Treatment Effect with Hypothesis Testing

As the plot in figure 3 suggests, the relationship between PANSS scores and t is more or less quadratic. Moreover, similar quadratic characteristics were also found on the other three aggregate metrics (see appendix). Therefore, f_0 and f_1 are identified as quadratic functions of t, specifically,

$$
\begin{aligned}
Y &= f_0(X,t) + \mathbb{1}\{\texttt{Treatment}\} f_1(X,t) && (2)\\
&= \left(\beta_{0,0} + \beta_{0,1}t + \beta_{0,2}t^2 + \vec{\gamma}_0\texttt{Country} + \vec{\delta}_0\texttt{Study}\right) && (3)\\
&+ \mathbb{1}\{\texttt{Treatment}\}\left(\beta_{1,0} + \beta_{1,1}t + \beta_{1,2}t^2 + \vec{\gamma}_1\texttt{Country} + \vec{\delta}_1\texttt{Study}\right) + \varepsilon && (4)\\
&= \theta_0 + \theta_1 t + \theta_2 t^2 + \theta_3 \mathbb{1}\{\texttt{Treatment}\} + \theta_4 \mathbb{1}\{\texttt{Treatment}\}t + \theta_5 \mathbb{1}\{\texttt{Treatment}\}t^2 + && (5)\\
&+ \vec{\gamma}_0\texttt{Country} + \vec{\delta}_0\texttt{Study} + Z + \varepsilon && (6)
\end{aligned}
$$

where `Study` and `Country` denote the collection of dummy variables, and Z denotes the set of interaction terms between variables in X and $\mathbb{1}\{\texttt{Treatment}\}$. It could be controversial whether to include interacting terms between $\mathbb{1}\{\texttt{Treatment}\}$ and variables other than t. As a result, regression results under all combinations of optional covariates are reported in table 1[1], in which `S`, `C`, and `T` stand for `Study`, `Country`, and `Treatment` respectively. The negative coefficients of $\texttt{T} \times t$ in all settings suggest the treatment actually helped reduce the severity of schizophrenia. However, due to the large standard errors of estimators, almost all of these coefficients were found insignificant. Even though in some rare cases, the treatment indicator on intercept terms were found significant.

[1]Significance codes: $0 * * * 0.001 * *0.01 * 0.05 \dagger 0.1 \quad 1$

Consequently, we conclude that the treatment did no affect how PANSS metrics changed over time, so there were no treatment effect.

Covariates from Z	$\varnothing$	C*T	S*T	C*T + S*T
PANSS_Total $\sim$ T	0.5167(0.1873)**	-2.113(1.577)	-0.4924(0.5505)	-3.717(1.798)*
PANSS_Total $\sim$ T $\times t$	-35.95(26.40)	-31.33(26.91)	-36.58(27.18)	-2.811(2.736)
PANSS_Total $\sim$ T $\times t^2$	24.99 (26.36)	23.72(26.36)	32.84(26.40)	2.819(2.641)
P_Total $\sim$ T	0.06806(0.06473)	-1.296(0.5447)*	-0.6036(0.1902)	-2.294(0.6210)***
P_Total $\sim$ T $\times t$	-7.643(9.121)	-8.135(9.297)	-12.59(9.391)**	-10.58(9.453)
P_Total $\sim$ T $\times t^2$	14.65(9.107)	1.449(9.108)	17.65(9.122)†	14.89(9.126)
N_Total $\sim$ T	0.1132(0.06793)†	-0.1981(0.5700)	-0.002971(0.1996)	-0.3724(0.6498)
N_Total $\sim$ T $\times t$	-10.22(9.571)	-6.610(9.728)	-6.791(9.856)	-2.909(9.890)
N_Total $\sim$ T $\times t^2$	0.4369(9.557)	1.472(9.530)	3.389(9.573)	4.399(9.548)
G_Total $\sim$ T	0.3353(0.09998)***	-0.6194(0.8418)	0.1142(0.2939)	-1.051(0.9601)
G_Total $\sim$ T $\times t$	-18.09(14.09)	-16.49(14.37)	-17.20(14.51)	-14.62(14.61)
G_Total $\sim$ T $\times t^2$	9.901 (14.07)	7.751(14.08)	11.80(14.10)	8.900(14.12)

Table 1: Regression Results with Different Combination of Variables (Standard Error in Parenthesis)

3 Patient Segmentation

In this section, segmentation methods were deployed to cluster 2438 patients based on their PANSS metrics at day zero. Two decisions must be made during this process: (i) how many clusters to use(ii); and what type of cluster to use.

3.1 Preprocessing

Even though the prior ranges of all PANSS sub-scores were the same, they had different sample means and variances. Most clustering algorithms are sensitive to the means and variances of features. To avoid potential issues, all 30 sub-scores were standardized before applying clustering on them, so that they all shared zero mean and unitary variance.

3.2 Identify Optimal Number of Clusters

Elbow method can provide some rough ideas on the optimal number of clusters to use. By analyzing the improvement in total (within group) sum of squares from adding more clusters, one can observe that adding extra cluster after $k = 7$ did not offer significant improvement. We can conclude that the optimal k would lie between 2 and 7.

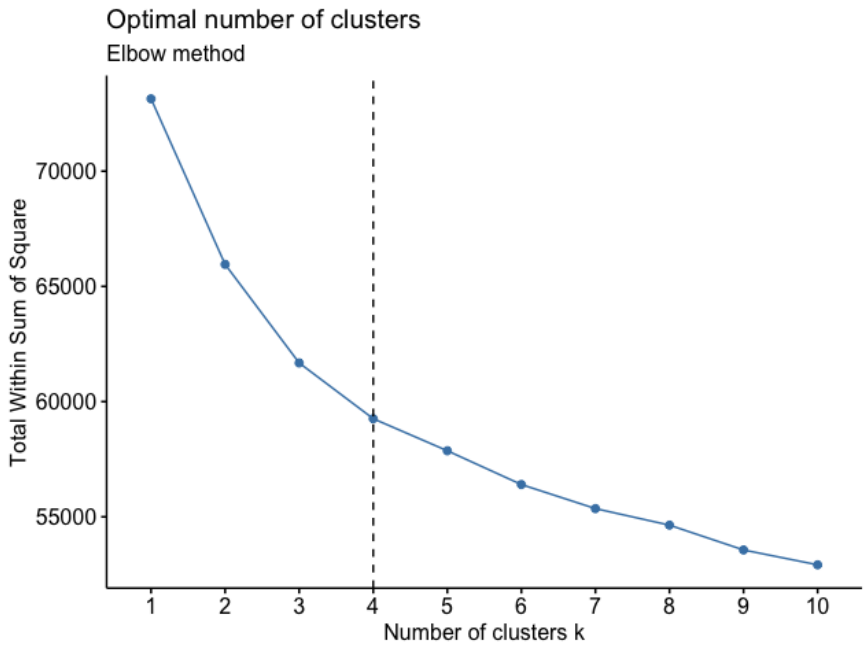

Figure 4: Elbow Method to Identify Number of Clusters

To find the optimal value of k, we used `NbClust` package in `R`, which implemented 30 criterions to identify the optimal k. Each criterion proposed one optimal k, and the final number of clusters was set to be the one proposed by most criterions. According to the result, 3 was the number of clusters supported by most criterions (14 criterions).

3.3 Choosing Clustering Algorithms

Agglomerative algorithms with four different types of linkages were deployed together with k-means. As mentioned before, the optimal number of clusters was identified to be 3. The experiment implied k-means delivered a more balanced segmentation in terms of group sizes when $k = 3$. As a result, k-mean with three clusters was used in following sections.

Model	2 Clusters	3 Clusters	4 Clusters
Agglomerative			
Ward	(1714, 724)	(1714, 452, 272)	(860, 854, 452, 272)
Complete	(2178, 260)	(2177, 260, 1)	(1580, 597, 260, 1)
Average	(2437, 1)	(2436, 1, 1)	(2435, 1, 1, 1)
Single	(2437, 1)	(2436, 1, 1)	(2435, 1, 1, 1)
K-Means	(1411, 1027)	(882, 828, 728)	(787, 642, 543, 466)

Table 2: Group Sizes from Different Algorithms

3.4 Visualization and Evaluation

Ideally, one can create a scatter plot of observations in their feature space, and then mark instances belonging to different clusters using different colours, to tell whether the clustering algorithm provides reasonable segmentation. However, it is impossible for us to visualize feature space beyond three dimensions.[2] Instead, clustering results were plotted using reduced feature spaces. PANSS sub-scores were inherently defined using three sub-groups, it is natural to use the totals of sub-scores belongs positive, negative, and general scores as three axes of the reduced feature space.

In additional, the reduced feature space of the first three principle components was used as well. Since the clustering algorithm only had access to standardized scores but not their principle components, a clustering algorithm is dong a good job if the clustering result also segment the principle component space reasonably as well. Figures below present the clustering result on both reduced feature spaces mentioned above. From the plot on PANSS scores, one can observed that the green group was featured by high scores in all three aggregate metrics, which is likely corresponding to the high-severity patients. While the orange group was characterized by low positive score with moderate negative and general scores. This group could be low-severity patients. Members in the blue group in general had above average positive scores, average general score, and low negative scores.[3]

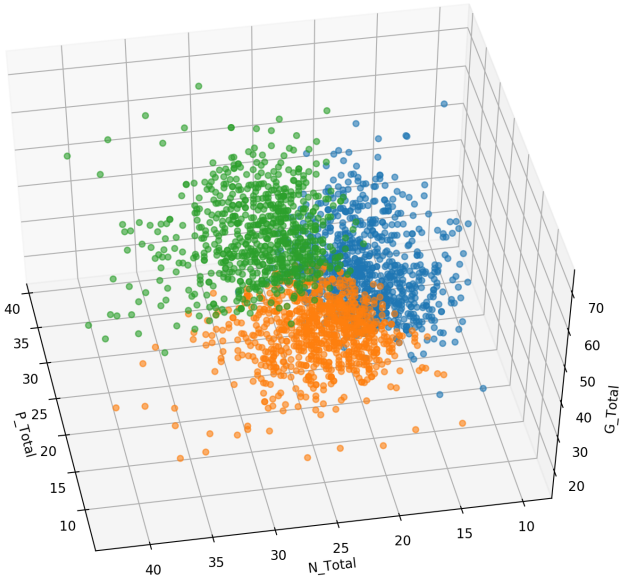

Figure 5: K-Mean Result On Aggregate Metrics of PANSS

Moreover, k-mean segmentation seemed to be reasonable on the principle component space as well, which further fortify the algorithm chosen.

[2]Theoretically, one can draw the fourth dimension by colouring observations differently. But in our case, colours are used to identify clusters.

[3]Statements here might be hard to observe using the static 3D plot, one can either run the script to generate an interactive 3D plot or check corresponding 2D plots in appendix.

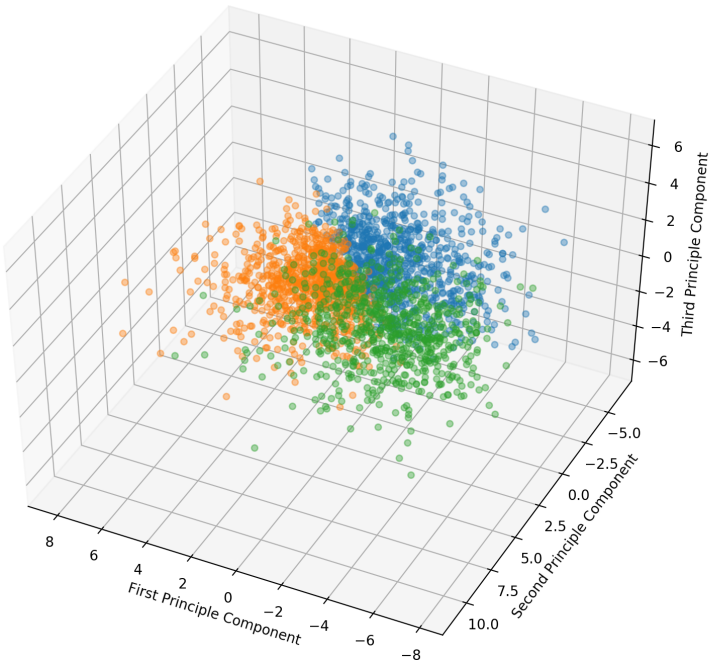

Figure 6: K-Mean Result on Principle Components

4 Patient 18-th Week PANSS Forecasting

4.1 General Strategy to Identify the Best Learner

In the following two sections, various models are proposed for forecasting and classification. In general, the hyper-parameters of each model must be tuned over a large hyper-parameter space $\mathcal{M}$. However, it is infeasible to search over the entire hyper-parameter space as there are uncountably infinite combinations of hyper-parameters for each model. Instead, a grid search algorithm with cross-validation was implemented. Firstly, a finite subset of $\mathcal{M}$, $\mathcal{G}$, was constructed manually, in which each element $g \in \mathcal{G}$ characterized a valid model. Then for each $g \in \mathcal{G}$, the grid search algorithm fit the model with hyper-parameter set g five times on different training sets given by a 5-fold cross validation. [4] And the generalization error for such model was estimated using the collection of different CV test sets. After this, for each category of models (random forests, support vector machines, etc), the hyper-parameter set achieved the best CV performance measured by mean squared error or cross-entropy loss was selected to represent the best performance for this model category.

After the best hyper-parameter set for each type of learner was identified, these models were evaluated again using 5-fold cross validation techniques to make comparisons across different types of models.

[4] Due to the time constraint, 5-fold CV was used instead of the more conventional 10-fold CV. Typically, hundreds of possible configurations were searched over for each type of model, using k-fold CV requires $k \times |\mathcal{G}|$ model fitting. While $k = 5$, the grid search for each type of model took around 4 hours on a 64-core server, it seemed to be infeasible to run grid search with $k = 10$.

4.2 Feature Space

This section is devoted to forecasting the total PANSS scores in the last visit of each participating patient. Several variables including indicator for treatment and country were invariant throughout the experiment period. The binary `TxGroup` variable was simply reduced to one binary variable `Treatment` $:= \mathbb{1}\{$`TxGroup` $=$ Treatment$\}$.

There were 284 unique site IDs and 639 unique rater IDs. Because IDs were numerical but not ordinal, adding these two features would require an addition of more than 900 one-hot-encoded variables, which were significantly more than the number of raw PANSS scores. Including them could be helpful, but at a risk of potential overfitting and cruise of dimensionality. Therefore, these IDs were excluded.

Assessments in the dataset came from 27 different countries (figure below). Note that there were 18 assessments (belonged to 3 patients) did not possess valid values for country. To reduce the dimension of feature space, and handle issue when there are incoming patients from countries not in the training set (there were patients from UK in the test set, but not in the training set), only information on the top five countries was preserved, and all other countries were reduced to one single "other" category. As a result, the `Country` feature in the raw dataset was transformed into six one-hot-encoded dummies: `Country_USA`, `Country_Ukraine`, `Country_Japan`, `Country_Russia`, `Country_China`, and `Country_Other`.

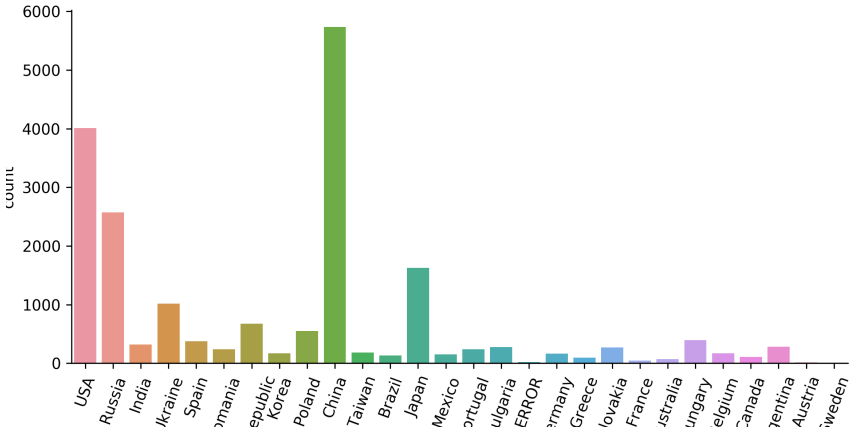

Figure 7: Distribution of `Country`

Each of the 1826 patients in the training set had different numbers of visits on record, most statistical learning models admit dataset in which all training instances have the same number of features. To accommodate this, summary statistics of PANSS scores were used together with time-invariant features like country and treatment. Specifically, the mean, maximum, minimum, and standard deviations of each PANSS sub-score were included. In additional, scores at day 0 and the last visit before the 18-th week visit were used as well to help the model capture patient's psychological status better.

4.3 Ensemble Method: Random Forest

Even though not all PANSS records were fed to the model, there were still hundreds of features actually used. To prevent potential overfitting, a random forest model was proposed. The grid searching suggested that an ensemble of 500 deep trees with depth of 4096 preformed best in terms of MSE estimated using 5-fold cross validation.

H-Param	Range	Best	Total
Max Depth (δ)	$\{\infty, 32, 64, 128, \cdots, 8192\}$	4096	10
Num. Trees (τ)	$\{100, 300, 500, \cdots, 3900\}$	500	20
Max. Features (ϕ)	$\{p, \sqrt{p}, \log_2(p)\}$	p	3
All Combinations			600

Table 3: Hyper-parameter Scope and Result for Random Forest

4.4 Support Vector Regression with Polynomial and RBF Kernel

Support vector regressions use different kernels to engineer input features implicitly. Two commonly chosen kernels are polynomial and RBF kernels. SVRs in general take longer to fit than other methods, but deliver superior performances. The scopes of hyper-parameter searching for different kernels are presented in tables below.

H-Param	Range	Best	Total
Polynomial Kernel			
Kernel Size (γ)	$\{\frac{1}{p}, 0.1, 0.01, 0.001, 0.0001\}$	0.0001	5
Err. Penalty (C)	$\{2, 4, 8, 16\}$	4	4
Polynomial Deg. (δ)	$\{3, 4, 5, 6\}$	3	4
All Combinations			80
RBF Kernel			
Kernel Size (γ)	$\{\frac{1}{p}, 10^{-10}, 10^{-9}, \cdots 0.1\}$	10^{-5}	10
Err. Penalty (C)	$\{2, 4, 8, \cdots, 516\}$	128	9
All Combinations			90

Table 4: Hyper-parameter Scope and Result for Support Vector Regression

4.5 Ensemble Method: Gradient Boosting

Gradient boosting machines refine predictions by iteratively fitting additional models on the residuals from previous models. The table below summaries the scope of hyper-parameters searched.

H-Param	Range	Best	Total
Max Depth (δ)	$\{3, 6, 9, 12\}$	3	4
Learning Rate (α)	$\{0.001, 0.003, 0.01, 0.03, 0.1, 0.3\}$	0.003	6
Num. Estimators (τ)	$\{100, 500, 900, 1300, 1700, 2100, 2500, 2900, 3300, 3700\}$	1700	10
Max. Features (ϕ)	$\{p, \sqrt{p}, \log_2(p)\}$	p	3
All Combinations			720

4.6 Summary on Model Performance

After selecting the best hyper-parameter profile for each type of model, the relative performances of the four candidate models were estimated using similar 5-fold cross validations on the entire dataset. From the result, one can see that support vector regression with RBF kernel and large random forest delivered superior performances on this forecasting task.

Model	Mean	Max	Min
Random Forest	78.59	61.04	96.03
SVM Poly. Kernel	95.24	85.31	107.8
SVM RBF Kernel	75.93	65.91	87.36
Gradient Boosting	130.17	110.27	147.24

Table 5: Relative Performance (MSE) of Selected Models on 5-Fold Cross Validation

5 Assessment Validity Classification

5.1 Feature Space

In this section, each training instance became one assessment instead of one patient. The feature space consisted of 30 PANSS scores and several identifiers such as country and rater's ID. In this study, all 30 PANSS scores and `PANSS_Total` were taken to be preliminary features. As for treatment and country identifiers, the same procedure mentioned in the previous section was followed.

In the training set, the standard deviations of 30 PANSS sub-scores ranged from 0.9374 to 1.562, and their averages ranged from 1.572 to 3.258. To eliminate this discrepancy, these 30 metrics were standardized so that all of them shared mean of zero and standard deviation of one.

Moreover, the figure below plots out the fraction for an assessment taken on particular visiting day to be flagged or assigned to CS (i.e. the empirical probability of positive class). The plot suggests that there exists nontrivial relation between the empirical probability of anomaly assessment and the day of visit. Therefore, `VisitDay` was also included as a feature.

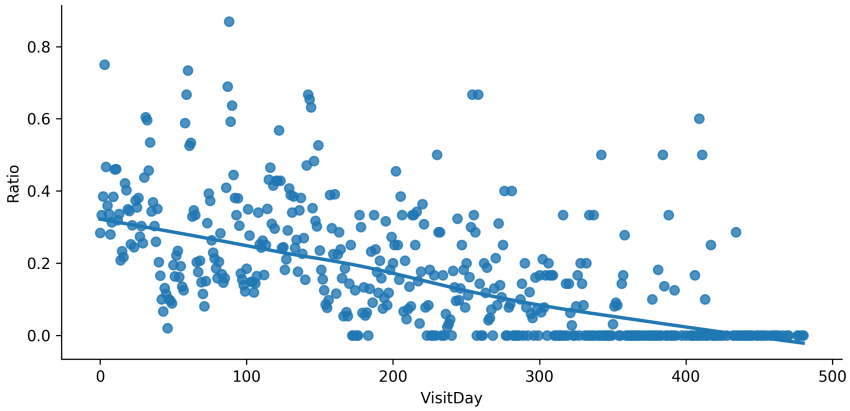

Figure 8: Fraction of Assessments Flagged on Each Day

5.2 Baseline Model: Logistic Regression

A logistic regression model was used as a performance baseline. There were only few customizations can be made on logistic regression, the grid search looked over the inverse of regularization strength C and the form of regularization. An elastic net regularization was applied to the baseline logistic regression and the regularization takes form $\alpha||\theta||_1 + (1-\alpha)||\theta||_2$, where $\alpha \in [0, 1]$ controlled the exact form of regularization term. The grid search results suggested a logistic regression with almost fully L-1 regularization performed the best.

H-Param	Range	Best	Total
Inverse Reg. (C)	$\{2^{-10}, 2^{-9}, \cdots, 2^{9}\}$	2^{-8}	20
L1 Reg. Weight (α)	$\{0, 0.2, 0.4, \cdots, 0.98, 1\}$	0.98	51
All Combinations			1020

Table 6: Hyper-parameter Scope and Result for Logistic Regression

5.3 Ensemble Model: Random Forest

Random forests handles overfitting problems naturally as an ensemble method. Several hyper-parameters play crucial rules while fitting a random forest, including number of trees built, the criterion for choosing the best split, the number of features to consider while identifying the best split, as well as the maximum depth of each tree. The table below shows the scope of above mentioned hyper-parameters searched over during the grid search process.

H-Param	Range	Best	Total
Max Depth (δ)	$\{\infty, 2, 4, \cdots, 1024\}$	64	7
Num Trees (τ)	$\{100, 300, 500, \cdots, 1900\}$	1900	10
Criterion	Entropy, Gini Coef.	Gini Coef.	2
Max Features (ϕ)	$\log_2(p), \sqrt{p}$	$\log_2(p)$	2
All Combinations			280

Table 7: Hyper-parameter Scope and Result for Random Forest

As mentioned before, all 280 candidates were evaluated using 5-CV and cross entropy loss. The best combination of hyper-parameters found after 1400 model fittings was included in the table. Note that the optimal value of τ was found on the boundary of $\mathcal{G}$, it is reasonable to suspect that further increase of τ could lead to an even further improvement in model performance. To accommodate this issue, two additional models with $\tau = 2000, 2100$ were evaluated using 5-CV and compared with the best model identified using grid search. It turned out that the improvements in generalization error were both only around 0.4%, and the performance even dropped when τ was risen from 2000 to 2100. Therefore, the model identified from grid searching was chosen to represent random forest class.

τ	Avg. Test Err. ($\pm\frac{1}{2}$ Range)
1900	0.3709($\pm$0.008014)
2000	0.3693($\pm$0.005638)
2100	0.3694($\pm$0.01037)

Table 8: Further Increments of Number of Estimators

5.4 Ensemble Model: Gradient Boosting

Gradient boosting is another type of ensemble models, in contrast to the parallel ensemble strategy of random forests, GBs ensemble multiple models vertically. For GBs, Friedman mean squared error was used to measure the quality of splits in the boosting process. The table below presents the scope of grid searching and the best combination of hyper parameters identified among all 480 candidates. Note that the optimal values were all in the interior of our pre-defined scope $\mathcal{G}$.

H-Param	Range	Best	Total
Max Depth (θ)	$\{3, 6, 9, 12\}$	6	4
Learning Rate (α)	$\{0.001, 0.003, 0.01, 0.03, 0.1, 0.3\}$	0.003	6
Num Estimators (τ)	$\{100, 300, \cdots, 1900\}$	700	10
Max Features (ϕ)	$\log_2(p), \sqrt{p}$	$\sqrt{p}$	2
All Combinations			480

Table 9: Hyper-parameter Scope and Result for Gradient Boosting Machines

5.5 Support Vector Classifier

Gradient boosting machines used the entire raw feature space and recursively fit new model on the residual of previous model, then ensemble all hierarchical models together. Each tree in a random forest actively omit some features at random to prevent overfitting. In contrast to these models, support vector machines implicitly engineer input features (i.e. map features to a higher dimensional space using some feature mappings) with different kernels. By using a radial basis function (RBF) kernel, SVMs effectively map input features to an infinite-dimensional space before fitting the dataset. Because SVMs generally take longer to fit than other methods on large dataset, only two hyper-parameters were searched over: the penalty for error terms C, and the radius of RBF kernel γ.

H-Param	Range	Best	Total
Err. Penalty C	$\{2, 4, 8, \cdots, 512\}$	512	9
Radius of RBF γ	$\{\frac{1}{p}, 0.1, 0.01, \cdots, 10^{-9}\}$	0.0001	10
All Combinations			90

5.6 Calibration

Models in this section were built to predict a *probability* for one assessment to be flagged, calibration methods were used to refine model outputs. The non-parametric isotonic regression was deployed instead of Platt scaling since the dataset was sufficiently large. Figure below presents a comparison between the uncalibrated random forest and the one calibrated using isotonic regression with 10-fold CV, in which the curve for the calibrated curve adhered the 45-degree line (i.e. perfect prediction) better. In reality, calibration does not guarantee improvements in the entropy loss, the model with best CV performance would be selected selected even if it was uncalibrated.

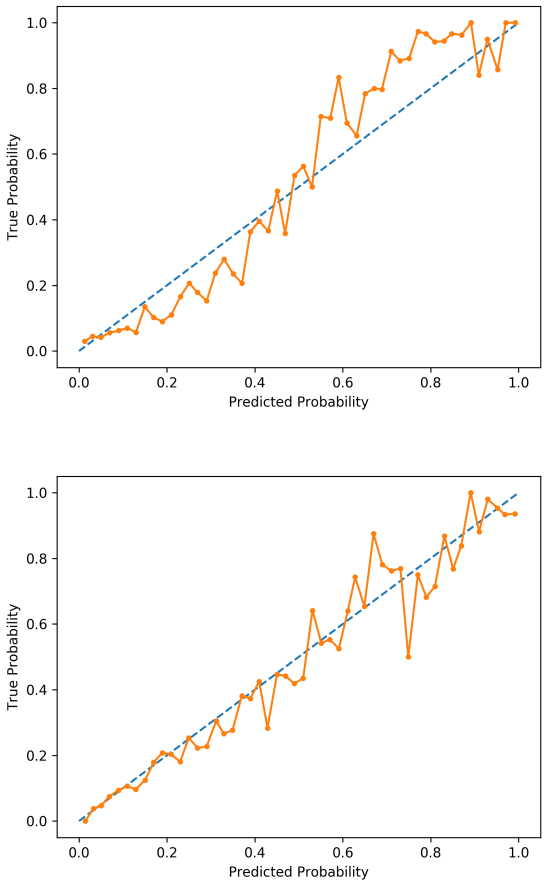

Figure 9: Calibration Curve of Raw Model and Isotonic Calibrated Model

5.7 Summary on Model Performances

Cross validations and the cross entropy were used to compare the relative performance across different types of models. It turned out that the uncalibrated random forest performed best.

Model	Uncalibrated	Calibrated
Logistic Regression	0.5041 ± 0.007183	0.5301 ± 0.01452
Random Forest	0.3689 ± 0.008212	0.3919 ± 0.009972
Gradient Boosting	0.4354 ± 0.01348	0.4277 ± 0.002407
Support Vector Classifier	0.4523 ± 0.01080	0.4686 ± 0.007865

Table 10: Relative Performances (log-loss/entropy) of Selected Models

Random forest also provided insights on relative importances of each features. From the feature importance plot, one can see that whether an assessment was taken in china or not, the visit day when the assessment was made, and the total PANSS score were three major most correlated with the validity of assessment.

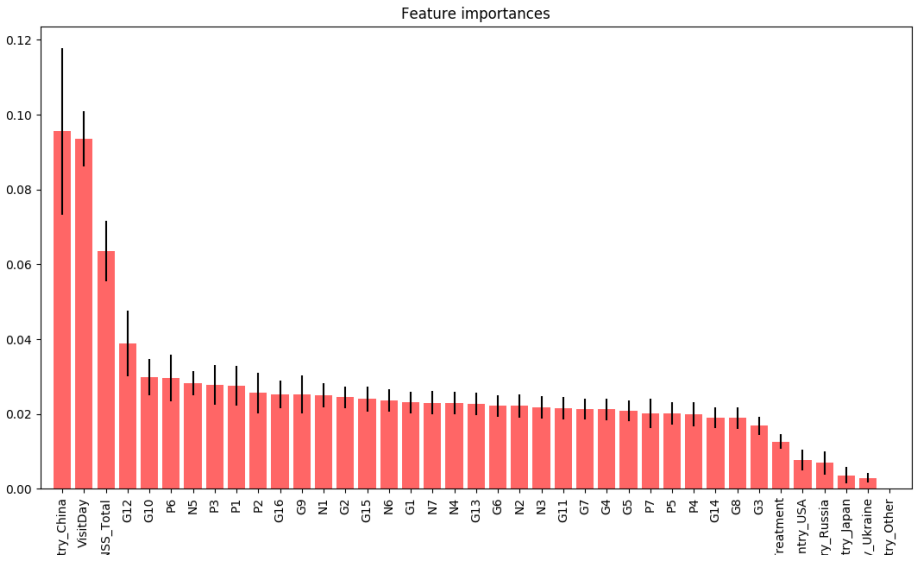

Figure 10: Feature Importance from Random Forest

6 Appendix

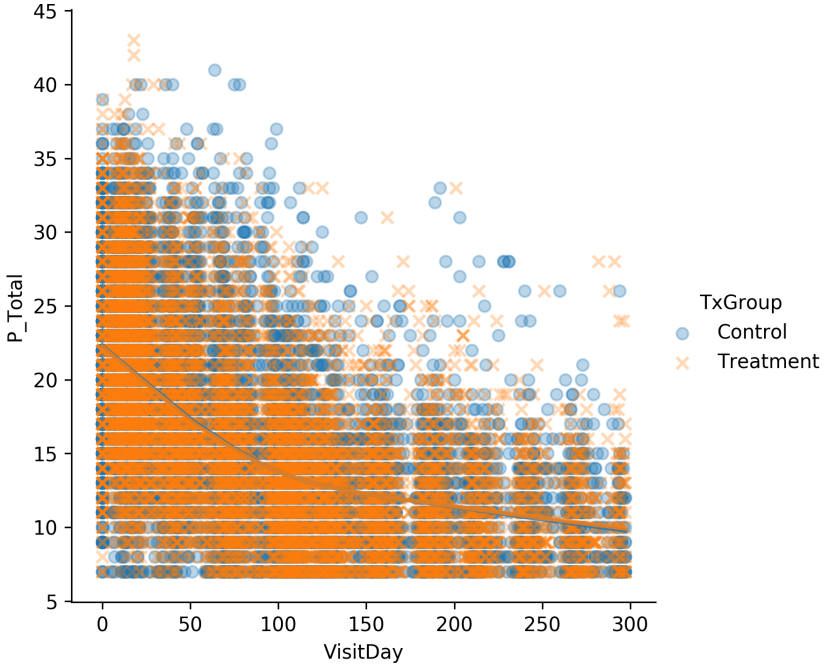

Figure 11: Positive Score against Time

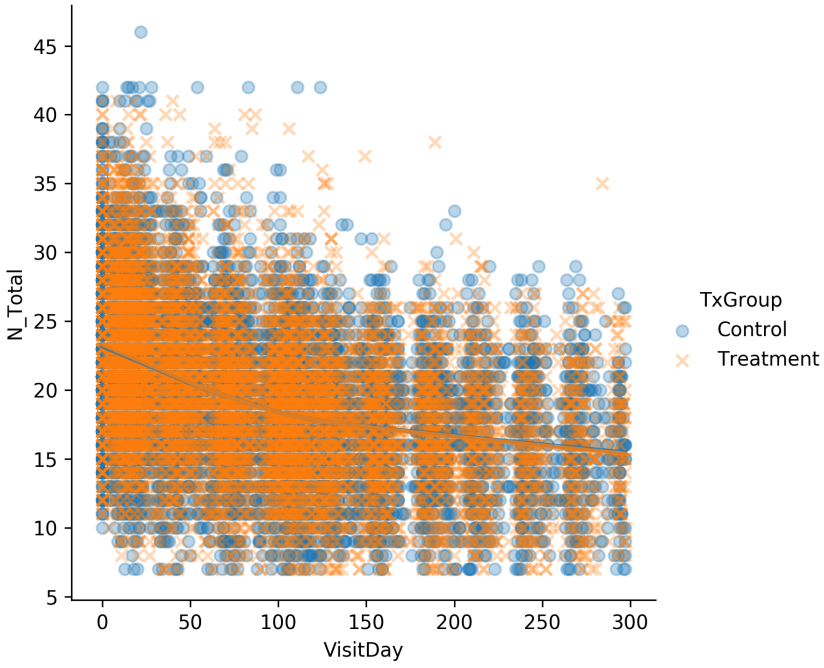

Figure 12: Negative Score against Time

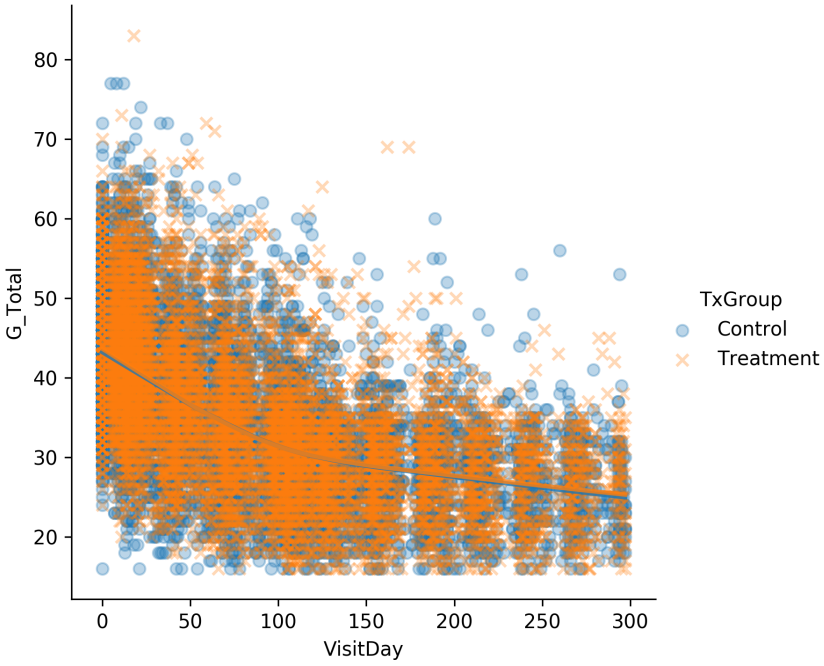

Figure 13: General Score against Time

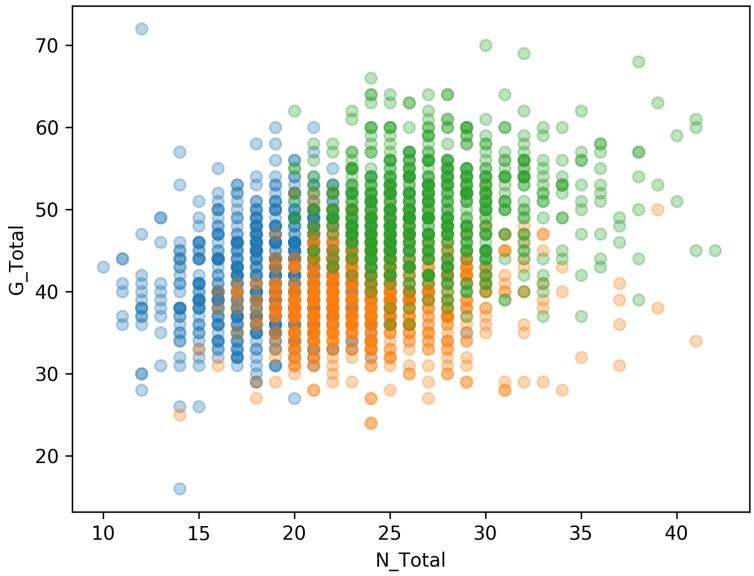

Figure 14: 2D Plots of K-Mean Result

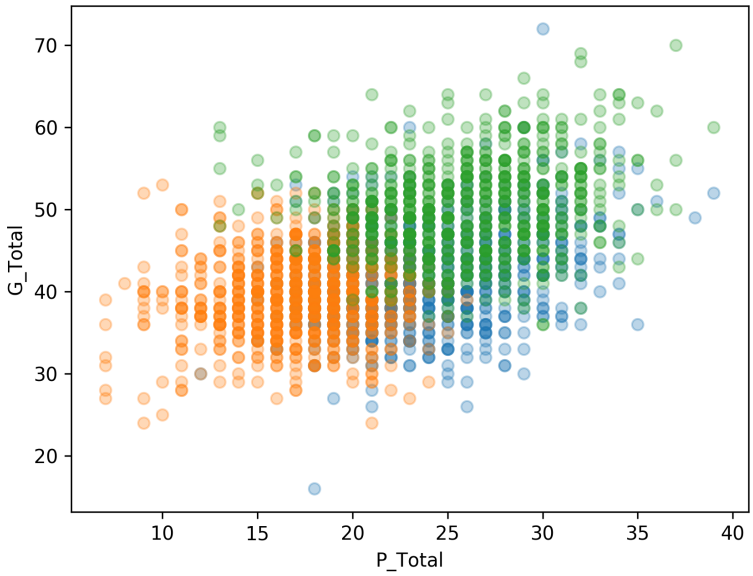

Figure 15: 2D Plots of K-Mean Result

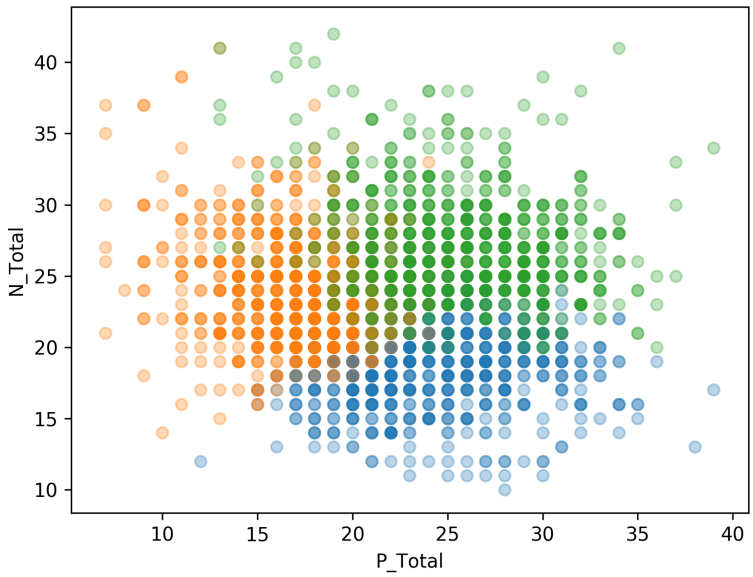

Figure 16: 2D Plots of K-Mean Result