1 Machine Learning to Predict Successful Discharge of Children from Behavioral Health Treatment

1.1 Reasons This Topic and Data were chosen:

- It has become increasingly important to achieve an accurate assessment of likelihood of successful discharge from treatment. There are multiple contingencies to consider: contractual and client care outcome goals, improved care, increased revenue, and business stability during times of crisis, such as COVID-19. Efficient and effective utilization of services enables client flow through our system and provides resources for new clients. Moreover, the field is shifting from fee for service to value-based cost reimbursement models that structure payment based on an expected time to discharge and an expected treatment outcome.
- Data driven leadership recognizes the importance of ascertaining successful outcomes in naturalistic settings (Webb et al., 2020) when the use of Randomized Clinical Trials is not feasible or practical. Machine learning allows for the use of naturally occurring data to train new models and utilize them with future data (Triantafyllidis & Tsanas, 2019; Wolff et al., 2020).
- The use of statistical prediction models is highly recommended for their versatility at many levels of mental health care delivery: diagnosis, prediction, and treatment planning (Garb et al., 2019)
- It is important to accurately identify high-risk clients at onset of treatment in order to intervene early and at the right level of intensity to ensure successful outcomes and to sustain well-being over time (Xu et al., 2019; Nie et al., 2017).
- The use of machine learning models to predict client outcomes is a nascent field of study in behavioral health; however, there is growing body of literature that supports machine learning to augment traditional clinical judgment and to strengthen the prediction of outcomes (Webb et al., 2020).
- Previous studies have shown the utility of machine learning to predict depression, treatment of obsessive-compulsive disorder in children, pediatric bipolar disorder, foster care permanency and likelihood of remaining cancer-free over five years (Elgin, 2018a; Gao et al., 2018; Lenhard et al., 2018; Wolff et al., 2020; Xu et al., 2019; Youngstrom et al., 2018)

1.2 Data Source Description, Extraction (ETL) and Exploration

Data for this project was extracted from our agency's EMR system. There were challenges with the extraction of this data that necessitated additional steps:

- Our system is a legacy system using an Intersystems Cache database, with the requirement that data be extracted using 32- ODBC drivers, with Crystal Reports and 32-bit WinSQL to generate SQL statements to extract data from underlying tables.
- There are hundreds of tables within the system that need to be considered and selection of tables to join is complex. The vendors provide an older version of Crystal Reports to be used in crafting reports and data extractions. I first applied Crystal Reports to test my connections between tables to get the views I needed, then ported the Crystal Reports structure into SQL, which then was used with our agency SQL platform, 32-bit WinSQL. The SQL setups used to extract the data are included as .txt files in the GitHub repository.
- Resultant data tables were exported as .csv files and stored in Postgres on my project computer. Schema to create tables in Postgres is included in the SQL subfolder.
- All data was collected on previously discharged clients in order to have completed outcome data on discharge reason. Only children and youth between the ages of 6 and 25 were

included, as I am focusing on programs serving children and young adults. At the time of discharge, however, there was one client currently 27 years old.

- Data was loaded into Python using Pandas and Psycopg2 library to directly connect to the Postgres database. An additional revised table with alternative CANS items was directly imported into Pandas after extraction from our company EMR (transfer and load process). This revised CANS measure table replaced the earlier table to test an alternative set of features. Copies of all original tables were kept in Power BI for later use. Encoded tables generated from the pre-processing and machine learning code were imported into another Power BI report (.pbix file) to create dashboards. A video link to the presentation of the interactive features is included: https://drive.google.com/file/d/1jeb9RN3ye3utKl6nt8YXqNbpsoc_I0Sc/view?usp=sharing

- Tables were:
    - **Demographics**—de-identified client information: patient ID, birth date, gender, ethnicity, education level, employment status, country of birth and primary language. These features/variables were used in previous studies (cited above) as predictors of treatment outcome. Patient ID ("patid") is the Primary Key.
    - **Diagnosis**—Patient ID, episode of treatment, primary client diagnosis at admission, trauma history and preexisting medical conditions.
    - **Program**—the type of program is a proxy feature representing level of care.
    - **Level of Care**—client programs were categorized along a continuum of care from prevention/early intervention to intense crisis-based care. This feature was constructed from the program indicator using customized encoding.
    - **Discharge data**—patient ID, date of discharge, episode (selected to most recent treatment episode), length of stay, reason (successful vs. not successful), program from which discharged. This data will be thoroughly examined for outliers, especially in length of stay when preprocessing data prior to machine learning.
    - **Assessment data**—use of an assessment tool that has been adopted agency-wide for children and youth called the Children and Youth Needs and Strengths Assessment (CANS). Included are initial scores on the following dimensions: depression, anxiety, conduct/oppositional disorders, school performance, and on the strengths dimension, included resiliency and family supports. The CANS tool is much more extensive in its inclusion of needs and strengths, but the first machine learning models sought to use fewer features to minimize overfitting and to simplify interpretation. CANS data was joined to the other tables using Patient ID as the foreign key.
    - **Outcome**—from the discharge data, did client successfully achieve treatment goals or did client discharge unsuccessfully? Binary coding (1=successful; 0=not successful) was used to create the y-variable for all train-test splitting and subsequent analysis.

1.3 Questions we hoped to answer

- What factors lead to successful completion of treatment for youth and children who receive mental health services at our agency?
- Conversely, what types of client characteristics are indicative of high risk and subsequent failure to complete treatment and achieve treatment goals? I aim to construct a profile of high-risk youth who may need early and vigorous intervention to succeed.
- Can a simpler, more parsimonious model suffice to fit the data and thus have a replicable model for future data sets? Thus, we start simply and then add or remove features and vary the learning algorithm if our accuracy, precision, and sensitivity are lower than optimal. The objective is to avoid overfitting that would decrease the effectiveness of the model to generalize to other datasets.

2 Methodology Used in Machine Learning

The overriding purpose of this study require the following elements:

- Prediction of outcome, using several supervised machine learning algorithms (logistic regression, random forest, ensemble, ADABoost and Gradient Boosted Tree methods)
- I also aimed to classify children into groups based on their initial characteristics, thus I also implemented an unsupervised machine learning mode using PCA. This classification model enabled visualization of groupings of clients that have similar characteristics and thus similar risk profiles. Two visualizations that I employed are: a boxplot to examine how each client cluster is distributed and a plot using Hvplot method. The interactive Hvplot was kept in the jupyter notebooks and interactive boxplots were imported into the dashboards using Python scripting embedded in Power BI.

2.1 Pre-processing and data exploration

The following steps were taken to explore the data and prepare it for the machine learning phase. All code is included in the ML subfolder in the form of jupyter notebooks. Key findings are in the Power BI dashboard and the Google slide presentation.

- Data was extracted from the agency EMR using Crystal Reports and SQL (please refer back to section 1 for more detail).
- Resultant .csv files were then imported into a new database created in Postgres.
- Import from Postgres and direct import of revised CANS dataset (.csv format) using Pandas
- Merging the individual datasets into one DataFrame for analysis using Pandas merge function
- Listing of categorical variables and using value count to determine need for binning. Variables were binned to reduce categories.
- Descriptive statistics on key predictors individually to examine means and to group by discharge outcome to explore possible relationships
- Grouping by the key categorical features to examine how they related to successful discharge proportions. Please refer to the jupyter notebook entitled **"Final Preprocessing Code"** for the complete code used to pre-process the data.
- Graphical representation of the data for each key feature and for the outcome of successful discharge. Examples of these are included in the dashboard and Google slides
- Removal of categorical features with limited variability, i.e., majority of clients in one category, such as country of origin

- Creation of dummy variables for categorical predictors using the Pandas get_dummies function, which created reference coding to establish one level of a factor as a reference level to which other categories are compared. One-Hot-Encoding was not used as it would have retained all levels for the factors, creating problems with multi-collinearity (refer to p.145-6 & p.187) of (Bruce & Bruce, 2017) for a discussion of factor variables in regression models.

2.1.1 Original Features

The original model included many features that eventually did not contribute significantly to prediction of successful discharge but were in the data frame created using Pandas. The reader is referred to the finalized pre-processing code in Jupyter Notebook entitled "**Final Preprocessing Code.ipynb** " to see all of the features entered into the preliminary Machine Learning engineering process.

2.2 Analysis—Machine Learning Model Selection

The decision to employ Logistic Regression was based on a search of the extant peer-reviewed literature that is directed at treatment outcomes in the fields of medicine and behavioral health. The preliminary feature selection followed researching the machine learning peer-reviewed literature that delineated factors such as diagnosis, age, ethnicity, gender, length of stay, level of care and family factors are features that have shown to contribute to models with high precision and accuracy (Elgin, 2018b; Webb et al., 2020) in predicting outcomes. **Figure 1** is the schematic of how Machine Learning was conducted.

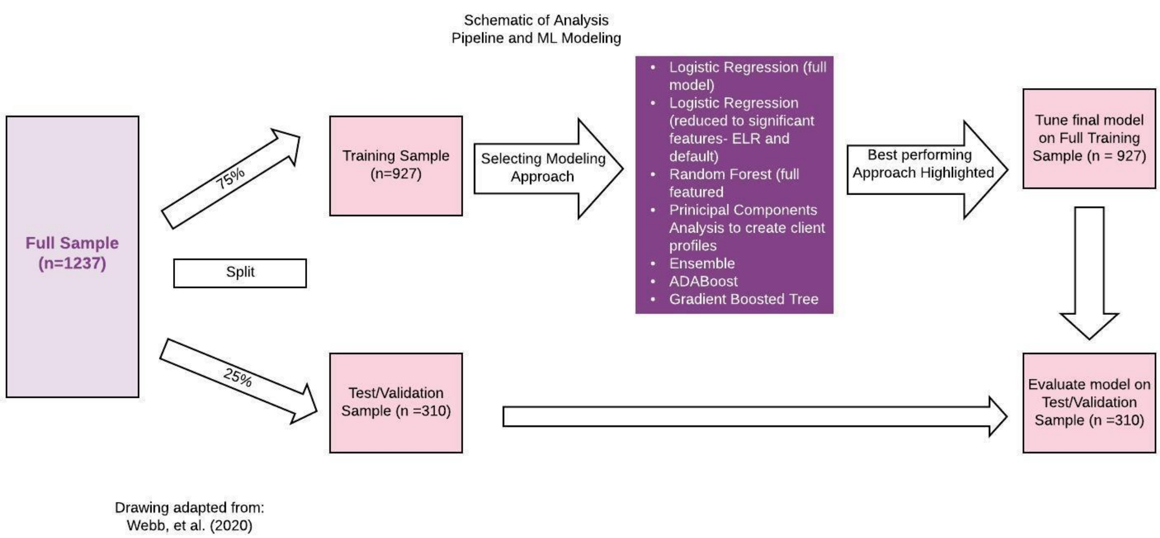

Figure 1: Schematic of Machine Learning Analysis

Logistic regression was chosen first as ease of interpretation of results provides a level of transparency and understanding that contributes to application in direct service settings. Providing internal and external stakeholders with detailed understanding of the factors that weigh into prediction of successful discharge will boost the subsequent use of the models in data driven leadership to improve policy and practice. Other methods, such as Random Forest, are less prone to interpretability, but nevertheless, are useful. Therefore, for the purposes of this project and to provide a comparative analysis that might boost precision, I added Random Forest modeling, Ensemble, ADABoost and Gradient Boosted Tree algorithms. The final accepted models employed Logistic Regression, Random Forest and Gradient Boosted Tree methods. **Table 1** lists the names of the Jupyter Notebooks and the Models they test; all are included in the Final Project Github repository in the ML subfolder; resources/data are in the Resources subfolder.

MODEL/METHOD	JUPYTER NOTEBOOK NAME
DATA CLEANING AND PRE-PROCESSING (ETL)	Final Preprocessing Code.ipynb
LOGISTIC REGRESSION	final logistic regression model.ipynb
RANDOM FOREST	Final Random Forest.ipynb
ADABOOST	Final ADABoost.ipynb
GRADIENT BOOSTED TREE	Final Gradient Boosted Tree.ipynb
PCA ANALYSIS	Final PCA Analysis.ipynb

Table 1: Code Used for Preprocessing and to Complete Machine Learning Analysis

Finally, I employed Principal Component Analysis (PCA) in unsupervised machine learning to classify clients along several dimensions, such as length of stay, level of care, age, social functioning, family strength, and oppositional behavior to create a preliminary classification of risk. These factors were the factors that were significant predictors of outcome in the final logistic regression model. I plan to conduct further research in later projects to improve the classification scheme.

These are the steps I performed[1]:

- Logistic regression using Statsmodels to show output of beta weights used to calculate odds and probability of successful discharge for selected key features as they individually and jointly contribute to the outcome. The first logistic regression trial involved the entire dataset, but the final model fitting involved splitting the data into training and testing samples before proceeding.
- Data was split into training and test sets using Train-Test-Split function, using 75% training and 25% percentage testing. This was modified from the first trials (default 70-30) to provide a larger training sample, as the total number of clients was only 1,237. I wanted a larger training sample to possibly improve classification yet have enough cases for the validation/test data set. Using 80-20 would have created a very small test data set, thus, I rested on 75%-25%.
- Logistic model fitting using Scikit after running the logistic regression model using Statsmodels.
- Metrics to evaluate model:
    - Classification report: 1) Precision, 2) Sensitivity/recall, 3) F1 Score, 4) Accuracy
    - Confusion matrix
    - ROC curve to show plot of false positive rate vs. true positive rate at varying threshold levels (different points in 0,1) using the AUC-ROC library in Scikit.
- Random Forest with classification report and rankings of feature importance using Scikit.
- Ensemble and ADABoost algorithms were employed, but not used in the final analysis.
- Gradient Boosted Tree with confusion matrix, classification report, and a test of various learning rates to select the best one.

[1] Note: earlier versions from previous commits follow this naming convention: all code is in the ML subfolder in Jupyter Notebooks. Note books that reference “ML Model” are the preprocessing notebooks. Those referencing “analysis” are the analysis portion.

- PCA in unsupervised learning to derive groups of youth based on age, level of care, length of stay, scores on key CANS measures and diagnosis at entry to begin exploration of a grouping method to classify youth at greater risk for unsuccessful discharge.
- Data from PCA was exported with the classifier added as an additional variable and joined to the original xvalue subset used for PCA. This was exported to .csv files to be used in Power BI for the dashboards, joining on the Pandas DataFrame index.
- All code is in the ML subfolder in Jupyter Notebooks.
- For Random Forest, the same preprocessed data employed in the logistic regression was used. All visualizations produced during pre-processing and analysis are included as .png files. The ROC curve produced during Logistic Regression analysis is called “LogisticReg_ROC.png”.

3 Results

3.1 Logistic Regression Modeling

At this point, the accuracy of the logistic regression model using Elastic Net Regularization is 0.68. Given that this is not optimal, I used the aforementioned methods to train the model, with the goal of improvement. The precision is 0.67, and the recall/sensitivity for successful discharge is 0.88. Analysis using Gradient boosted tree found that a learning rate of 0.75 yielded the best model in terms of accuracy (**68%**), with recall/sensitivity for successful discharge at 0.80. **Table 2** shows the results of the Logistic Model analysis conducted using the Statsmodels library in Python.

Social functioning did not significantly contribute to prediction of successful discharge in the Logistic Regression Model. Because logistic regression works with a binary outcome, special transformations must be made to interpret the findings. We are seeking to measure the probability that the outcome will be a “1” or success, which is mapped on a 0-1 scale. We use the logistic response function (Bruce & Bruce, 2017) as shown on p. 185[2] of this book, $p = \frac{1}{1+e^{-(\beta 0+\beta 1x1 \ldots.+\beta qxq)}}$. This transform ensures that p stays between 0 and 1. While not going into great detail, the final linear function is the log-odds using the logistic response function of the ***e*** shown in the equation, which gives the probability that Y=1. The log-odds converts the exponent consisting of the β coefficients into a linear equation.

Feature (n=927-training data used)	β-Coefficient	SE	Z	p	LCL	UCL
_Constant (y-intercept)	-0.1072	0.3300	-0.3250	0.7450	-0.7540	0.5400
Length of Stay in Days	0.0044	0.0010	7.7630	0.0000	0.0030	0.0060
Recreational functioning (CANS score)	-0.2222	0.0920	-2.4270	0.0150	-0.4020	-0.0430
Social functioning (CANS)	-0.1095	0.0900	-1.2180	0.2230	-0.2860	0.0670
Defiant oppositional behavior (CANS)	-0.2544	0.0880	-2.8840	0.0040	-0.4270	-0.0820
Family strength (CANS)	-0.2955	0.0860	-3.4440	0.0010	-0.4640	-0.1270
Age at Discharge	-0.0368	0.0190	-1.9090	0.0560	-0.0750	0.0010
Level of Care	0.2949	0.1020	2.8960	0.0040	0.0950	0.4940
DSM5 ADHD	-0.7246	0.3040	-2.3870	0.0170	-1.3200	-0.1300
DSM 5 Oppositional defiant disorder	-0.6632	0.2840	-2.3340	0.0200	-1.2200	-0.1060
p < 0.05 is statistically significant	Pseudo-R-squared =.105					

Table 2: Logistic Regression Results

[2] Bruce and Bruce (2017) provide one of the most understandable and cogent explanations of logistic regression across many textbooks I have studied.

Final logistic regression model fitting employed the Elastic Net Regularization method, which has performed well in previous research (Webb et al., 2020). **Table 3** shows the confusion matrix for testing/validation data and **Table 4** shows the Classification Report for the ENR model. The recall/sensitivity is quite high for the model, indicating that **88%** of the successful discharges are correctly classified. Additionally, **67%** of the predicted successes were actually successes (precision); however, only **68%** overall of the cases were correctly classified, as indicated by the accuracy score. **Figure 2** shows the ROC Curve for the ENR model.

Category	Predicted Not Successful	Predicted Successful	Totals
Actual Not Discharged Successfully	50	78	128
Actual Discharged Successfully	21	161	182
Totals	71	239	310

Table 3: Confusion Matrix for ENR

Element	precision	recall	f1-score	support
Not Successful	0.70	0.39	0.50	128
Successful	0.67	0.88	0.76	182
Accuracy			0.68	310
Macro avg	0.69	0.64	0.63	310
Weighted avg	0.69	0.68	0.66	310

Table 4: Classification Report for Logistic Model using Elastic Net Regularization

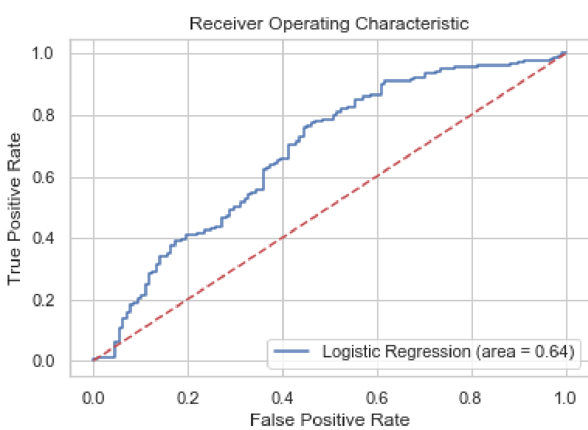

Figure 2: ROC for False vs. True Positive Rate

3.2 Random Forest

The second model employed was Random Forest, evaluated against the testing/validation sample. The accuracy of this model was **69%**. **Table 5** shows the Confusion Matrix for the Random Forest Model. **Table 5** presents the classification report.

Confusion Matrix

Outcome	Predicted Not Successful	Predicted Successful	Totals/Support
Actual Not Discharged Successfully	64	72	136
Actual Discharged Successfully	25	149	174
Totals	89	221	310

Table 5: Confusion Matrix for Random Forest Model

Classification Report

Outcome	Precision	Recall	F1-score	support
Not Successful	0.72	0.47	0.57	136
Successful	0.67	0.86	0.75	174
accuracy			0.69	310
Macro Avg	0.70	0.66	0.67	310
Weighted Avg	0.69	0.69	0.68	310

Table 6: Classification Report for Random Forest

Table 7 shows the top five features contributing to prediction of successful discharge for the Random Forest model.

Top Five	Weight
Length of stay in Days	0.17
Child Age	0.08
Number of most recent episode	0.03
Level of Care (higher number=higher level of care)	0.03
Family Functioning	0.03

Table 7: Top Five Features by Importance for Random Forest Model

3.3 Gradient Boosted Tree

The last model employed was the Gradient Boosted Tree algorithm. **Table 8** shows the Confusion Matrix, followed by **Table 9** depicting the Classification Report. The final model used a 0.75 learning rate as this produced the highest validation accuracy. **Figure 3** graphically represents the relationship between learning rates and accuracy for both training and validation data sets.

Outcome	Predicted Not Successful	Predicted Successful	Totals/Support
Actual Not Successful	71	65	136
Actual Successful	34	140	174
Total	105	205	310

Table 8: Confusion Matrix for Gradient Boosted Tree Model

Outcome	Precision	Recall	F1-score	Support
Actual Not Successful	0.68	0.52	0.59	136
Actual Successful	0.68	0.8	0.74	174
Accuracy			0.68	310
Macro avg	0.68	0.66	0.66	310
Weighted avg	0.68	0.68	0.67	310

Table 9: Classification Report for Gradient Boosted Tree Model

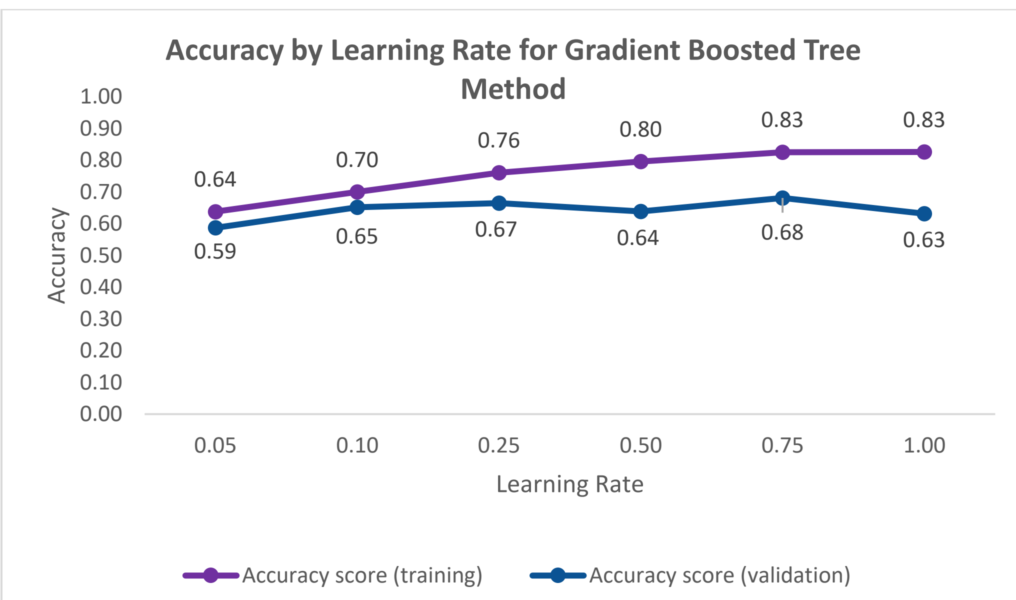

Figure 3: Accuracy by Learning Rates for Gradient Boosted Tree Model

3.4 Summary of Supervised Learning Models

As noted in **Figure 4**, the three selected models did not vary much in terms of accuracy, indicative of consistency in performance. However, the rather low accuracy points to the need to consider other features not included in this project.

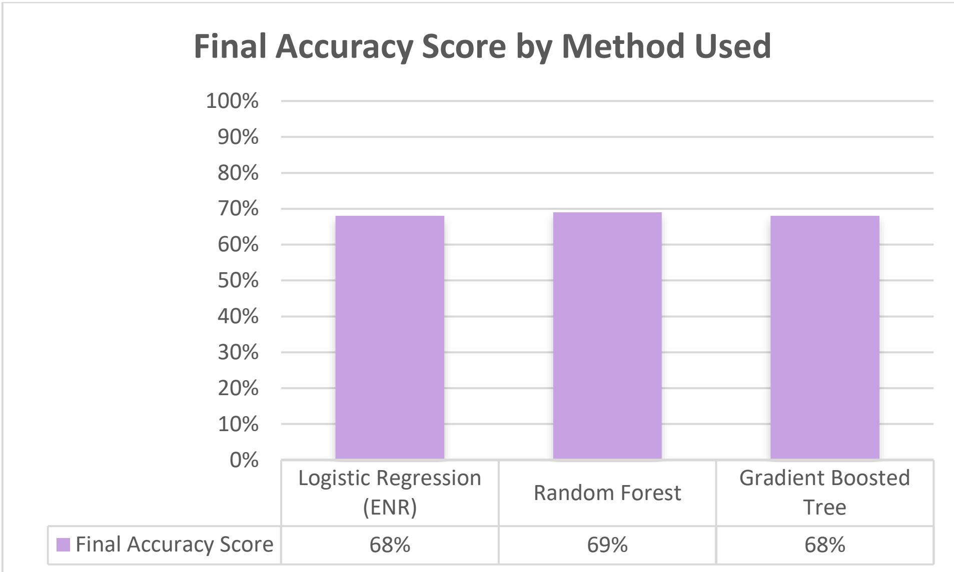

Figure 4: Comparison of Accuracy Scores for Three Models Selected

3.5 Principal Components Analysis—Unsupervised Learning

The final model that was used entailed Principal Components Analysis (PCA), an unsupervised learning model that aims to classify data by determining the most important traits that differentiate individuals from each other. It is also a method to simplify data and still derive robust information, to quote (Jaadi, 2020):

> "Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set.
>
> Reducing the number of variables of a data set naturally comes at the expense of accuracy, but the trick in dimensionality reduction is to trade a little accuracy for simplicity. Because smaller data sets are easier to explore and visualize and make analyzing data much easier and faster for machine learning algorithms without extraneous variables to process.
>
> So to sum up, the idea of PCA is simple — reduce the number of variables of a data set, while preserving as much information as possible"[3].

The analysis conducted involved construction of an elbow curve to determine the optimal number of clusters from the data. A two-component solution fit the data best. Common practice dictates that when the elbow reaches a low point after a steep decline and a direction shift occur is the junction where the number of components should be chosen. **Figure 5** points to 2 clusters as optimal. Refer also to our course material **Module 18.4.2** for an explanation of the elbow curve interpretation.

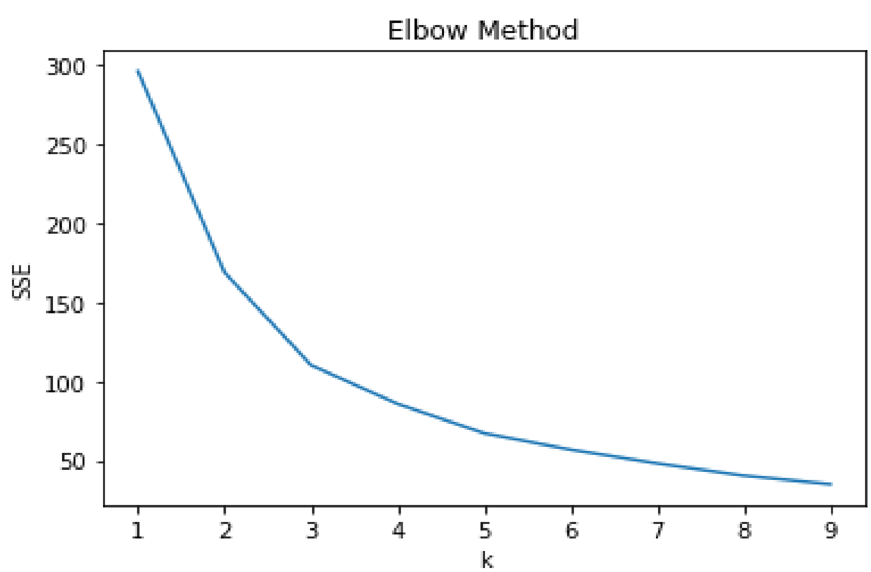

Figure 5: Elbow Curve to Determine Number of Principal Components

The next image in **Figure 6** shows the way in which the two principal components are distributed in the data. **Figure 7** shows differences by cluster in presenting features of the youth. It is apparent that Cluster 1 depicts clients with better functioning, as the CANS scores are interpreted as higher scores equate to lower functioning, or greater need. **Figure 7** is a static representation of an interactive report page in Power BI showing how the two clusters differ in key aspects of functioning along the dimension of clients' age. The

[3] Refer to: https://builtin.com/data-science/step-step-explanation-principal-component-analysis for a cogent explanation of how PCA is conducted.

two components together accounted for **36%** of the variance in the data, reported in the jupyter notebook output (**Component 1=0.21, Component 2= 0.15**).

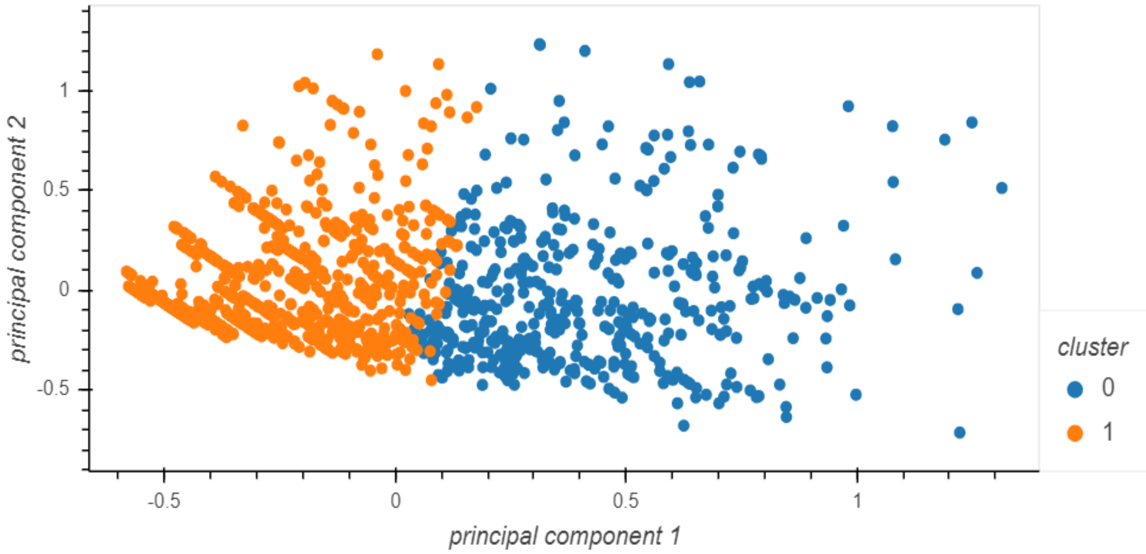

Figure 6: Principal Components Clustering

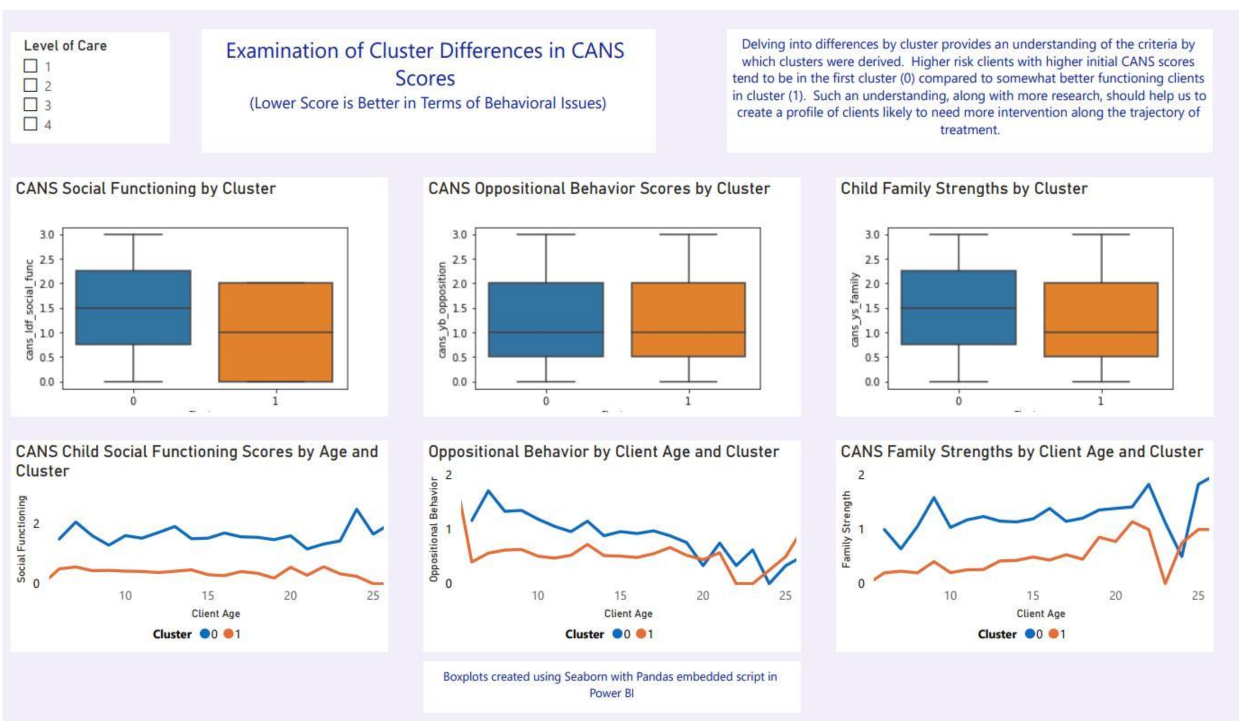

Figure 7: Analysis of differences by PCA Cluster

3.6 Conclusion

In this section, I address my conclusions, point out limitations, and refer to implications for future research. I provide a link to view my video on the interactive functioning of the Power BI dashboard that highlights the pre-processing and machine learning analysis.

- Competing models showed consistent accuracy but varied in terms of precision and sensitivity. The accuracy was only fair, as it is common practice in machine learning to have accuracy achieve 75% or better.
- The classification metrics indicated that the models may have underfit the data, thus more features and larger sample size should be considered. The logistic regression showed that the variables entered into the model only accounted for about **11%** of the variance (Pseudo R squared was .105). This is a limitation of the current study.
- Along with the low R-squared value, very few of the originally included features remained, as they did not contribute significantly to prediction of outcome. This indicates that other salient features not considered in this project were likely to boost the classification and should be included in future research.
- While the results addressed factors that contribute to successful outcomes, much work remains to refine the model. For example, the diagnosis features were possibly too granular, and should have been categorized into broad diagnoses, such as depression, anxiety, conduct disorders, etc. instead of retaining the detailed sub-categories of disorders as they are defined in the DSM-5 manual of disorders.
- Future work will entail inclusion of additional features, such as the dosage of treatment received, the types of interventions used, and refined categorization of diagnosis, as well as inclusion of substance use diagnoses, even when they are secondary. It will be recalled that I restricted the features to primary intake diagnosis for the most recent treatment episode.
- The Principal Components model is a beginning attempt to classify clients according to salient factors that lead to an understanding of risk. I will continue to work on refining this model to elicit a profile of clients who are at risk for treatment failure, as the current model only accounted for **36%** of the total variance in the features. By knowing what factors play into risk, it will enable practitioners to tailor interventions specific to the needs and strengths of their clients and thus increase the likelihood of successful completion of treatment goals.

In short, the models created and tested provided a good beginning to a research trajectory whose purpose is to determine what contributes to successful completion of treatment and thus client well-being and better psychological health. While the models are not as robust as I would have desired, I will improve them in the future with more features, consideration of diagnosis more comprehensively, and use of other models, such as XGBoost, revisiting logistic regression and deep learning approaches. I also aim to re-test the principal components model with more features.

Note: link to the video presentation of the interactive dashboard is here:
https://drive.google.com/file/d/1jeb9RN3ye3utKl6nt8YXqNbpsoc_I0Sc/view?usp=sharing

Link to Google Slides Presentation:
https://docs.google.com/presentation/d/1nKInttgj_mGxHx019cQ8JfChaD_7QNjgaEdZLVqrs6s/edit?usp=sharing

I will post this in the README.MD file as well.

References Cited:

Bruce, P., & Bruce, A. (2017). *Practical Statistics for Data Scientists*. O'Reilly.

Elgin, D. J. (2018a). Utilizing predictive modeling to enhance policy and practice through improved identification of at-risk clients: Predicting permanency for foster children. *Children & Youth Services Review*, *91*, 156–167. http://10.0.3.248/j.childyouth.2018.05.030

Elgin, D. J. (2018b). Utilizing predictive modeling to enhance policy and practice through improved identification of at-risk clients: Predicting permanency for foster children. *Children and Youth Services Review*, *91*, 156–167. https://doi.org/10.1016/j.childyouth.2018.05.030

Gao, S., Calhoun, V. D., & Sui, J. (2018). Machine learning in major depression: From classification to treatment outcome prediction. *CNS Neuroscience and Therapeutics*, *24*(11), 1037–1052. https://doi.org/10.1111/cns.13048

Garb, H. N., Wood, J. M., Garb, H. N., Base, J., Antonio, S., Antonio, S., & Wood, J. M. (2019). Methodological Advances in Statistical Prediction. *Psychological Assessment*, *March 11*, *2019*, 1–12.

Jaadi, Z. (2020). *A Step by Step Explanation of Principal Component Analysis* (Issue May 19). https://builtin.com/data-science/step-step-explanation-principal-component-analysis

Lenhard, F., Sauer, S., Andersson, E., Månsson, K. N. T., Mataix-Cols, D., Rück, C., & Serlachius, E. (2018). Prediction of outcome in internet-delivered cognitive behaviour therapy for paediatric obsessive-compulsive disorder: A machine learning approach. *International Journal of Methods in Psychiatric Research*, *27*(1), 1–12. https://doi.org/10.1002/mpr.1576

Nie, Z., Xue, G., & Li, J. (2017). *Machine Learning Methods for Diagnosis, Prognosis and Prediction of Long-term Treatment Outcome of Major Depression*.

Peters, R. (n.d.). *Classification is Easy with SciKit ' s Logistic Regression*.

Triantafyllidis, A. K., & Tsanas, A. (2019). Applications of machine learning in real-life digital health interventions: Review of the literature. In *Journal of Medical Internet Research* (Vol. 21, Issue 4). Journal of Medical Internet Research. https://doi.org/10.2196/12286

Webb, C. A., Cohen, Z. D., Beard, C., Forgeard, M., Peckham, A. D., & Björgvinsson, T. (2020). Personalized prognostic prediction of treatment outcome for depressed patients in a naturalistic psychiatric hospital setting: A comparison of machine learning approaches. *Journal of Consulting and Clinical Psychology*, *88*(1), 25–38. https://doi.org/10.1037/ccp0000451

Wolff, J., Gary, A., Jung, D., Normann, C., Kaier, K., Binder, H., Domschke, K., Klimke, A., & Franz, M. (2020). Predicting patient outcomes in psychiatric hospitals with routine data: A machine learning approach. *BMC Medical Informatics and Decision Making*, *20*(1), 1–10. https://doi.org/10.1186/s12911-020-1042-2

Xu, Y., Ju, L., Tong, J., Zhou, C., & Yang, J. (2019). Supervised machine learning predictive analytics for triple-negative breast cancer death outcomes. *OncoTargets and Therapy*, *12*, 9059–9067. https://doi.org/10.2147/OTT.S223603

Youngstrom, E. A., Halverson, T. F., Youngstrom, J. K., Lindhiem, O., & Findling, R. L. (2018). Evidence-Based Assessment From Simple Clinical Judgments to Statistical Learning: Evaluating a Range of Options Using Pediatric Bipolar Disorder as a Diagnostic Challenge. *Clinical Psychological Science*, *6*(2), 243–265. https://doi.org/10.1177/2167702617741845