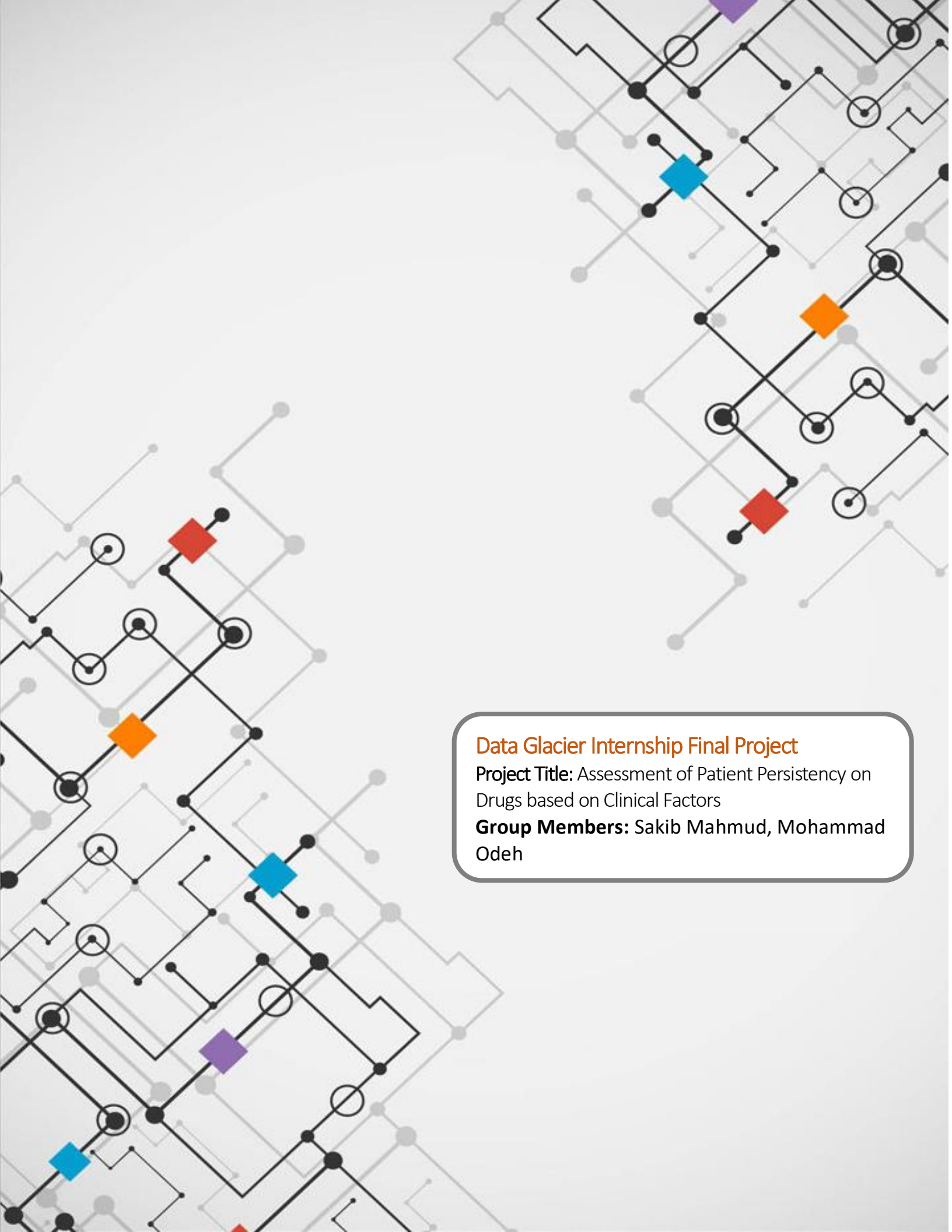

Data Glacier Internship Final Project

Project Title: Assessment of Patient Persistency on Drugs based on Clinical Factors

Group Members: Sakib Mahmud, Mohammad Odeh

Data Science Healthcare Project: Assessment of Patient Persistency on Drugs based on Clinical Factors

Table of Contents

Details of Team Members

Group Name: Health+	
Member 1: Sakib Mahmud	**Student Email:** sm1512633@qu.edu.qa **Personal Email:** sakib1263@hotmail.com **Official Email:** sakib.mahmud@qu.edu.qa
	Country: Qatar
	College + Company: Qatar University
	Specialization: Data Science
Member 2: Mohammad Odeh	**Personal Email:** odeh4893@gmail.com
	Country: United Arab Emirates (UAE)
	Specialization: Data Science

Project Lifecycle

The entire project along with all requirements is due for submission by the 15th of May 2021, along with some weekly submissions or updates. The project has been broken into several sub-tasks to smooth and timely progression, which are as follows:

- Problem Understanding.
- Data Understanding.
- Data Cleaning and Feature engineering.
- Model(s) Development.
- Model Selection.
- Model Evaluation.
- Report evaluation metrics such as Confusion Matrix and its derivatives (Accuracy, Precision, Recall, and f1-scores).
- Report ROC-AUC.
- Deploy the model.
- Explain the challenges and further improvements.

Data Intake Report

Name: Persistency of a Drug
Report Date: 06 May 2021
Internship Batch: LISP01
Version: 2.0
Data Intake: Mohammad Odeh and Sakib Mahmud
Data Intake Reviewer:
Data Storage Location: Local Storage (PC)

Tabular Data Details:	
Total number of Observations	3424
Total number of Features (Independent Variables or Predictors)	68
Total Number of Data Points	3424*68 = 232832
Total number of File(s)	1
Base format of the File	.xlsx
Size of the dataset	898 KB

Proposed Approach:

- Perform Exploratory Data Analysis (EDA) on the data set and visualize various characteristics of the data looking at it from different angles.
- Data pre-processing and cleansing such as getting rid of null values, removing unnecessary columns, check for outliers and unrepresentative data, so on and so forth.
- Transform data such as creating Dummy Variables from categorical data to perform Machine Learning (ML) and other regression-based statistical analysis.
- Build, Test, and Evaluate different ML models based on the features that are correlated.
- Apply dimensional reduction techniques such as Autoencoders to create compact models to represent the whole dataset while preserving the performance.

Assumptions:

- The data follows Normal Distribution.
- Patients' history data were recorded accurately without any errors in testing or examination.

Problem Description and Business Understanding

Among the most critical challenges that pharmaceutical companies face in general, the most common one is the task of understanding the "Persistency of a Drug" as per the physician's prescription. To solve this problem "ABC Pharma" company approached an analytics company to automate this process of identification. ABC Pharma provided the company with their recorded data in an Excel file for analysis. The dataset contains four types of predictor parameters (or independent variables) for unique patients (each patient had a unique identifier or ID) such as Patient Demographics, Provider (Doctor/Nurse/Other Medical Staff) attributes, Clinical Factors, and Disease or Treatment Factors. On the other hand, the target or dependent variable for the dataset is the "Persistency Flag" for the patient i.e., whether the patients were persistent on their prescribed medicine(s) or not.

Among the patient demographics parameters, there were patients' age, race, region, etc. Attributes of the physician who prepared the prescription or performed the task of observing the patient might be an important predictor, so it was included. The primary disease for which patients were treated in this case is Nontuberculous Mycobacterial (NTM). Various tests such as DEXA Scans are performed for NTM which produces metrics like T-Score. Clinical Factors like the outcomes of these tests during the Rx and the performance shift during the last 1 to 2 years were also accounted for, along with the Risk Segment of the patient and the possibility of prevalence of multi-risk among the patients. Other treatment factors such as comorbidity of patients for other diseases alongside NTM, Injectable Experience, and concomitancy of various drugs applied on the patient for NTM were also accounted for. All these parameters will be used to produce models using Machine Learning to correctly classify patients based on their "Persistency Flag". Efforts will also need to be given to determine the most influential parameters (or class of parameter) for determining peoples' choice on continuing the medicine.

More than the healthcare perspective of this problem, it has an even more important business perspective. As discussed earlier, one of the challenges for all pharmaceutical companies is to understand the persistency of drugs as per the physician's prescription. The general trend of persistency of pharmaceutical products among a group of patients is downward, as depicted by the "Persistency Curve" in Figure 1. The pharmaceutical companies aim to determine the factors which affect the decline most so that they can address these issues properly and slow down the process (i.e., the smaller slope of the model). Pharmaceutical companies, healthcare organizations, and hospitals in the USA lose billions

of dollars per year due to patients not being persistent in their prescribed medicines and/or treatments [1]. Based on the collected data by any pharmaceutical company or an Integrated Delivery Network (IDN), the most important factors behind the lower level of persistence among patients can be determined. The company or organization can address those issues properly to mitigate the process and over time they can resurvey to check for improvements.

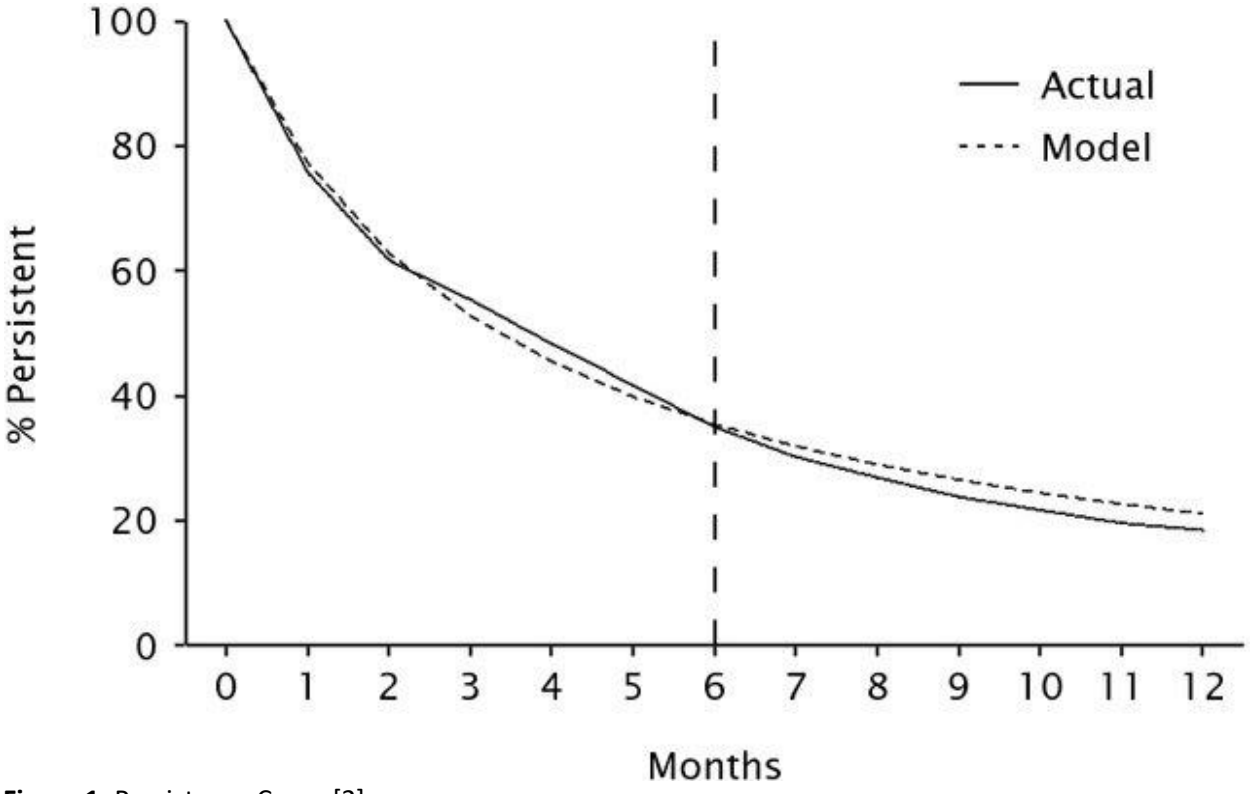

Figure 1: Persistency Curve [2]

From the Persistency Curve in Figure 1, it can be understood that the general trend of the Persistency Curve is downward i.e., patients generally do not adhere to the medicine or the set of medicines prescribed by their doctors. They either stop taking the drug for many reasons or swap to another. Since the overall trend is always downward, the important thing to focus on is how to slow down the rate. The main key behind slowing down this rate is to determine the most influential factor(s) behind so that they can be assessed in time and important business decisions can be taken which could save millions, sometimes even millions for the company or the government. To make people adhere to a certain prescription or guidelines for a long time is not a trivial task and IDNs across the USA were formed primarily due to this.

Dataset Understanding: Explorative Data Analysis (EDA) on the Dataset

The Dataset can be briefly described based on Table 3 below as it was included in the dataset. From this table, it can be understood that there mainly four types of predictor variables for this dataset, the "Target Variable" being the "Persistency Flag". These types are "Demographics", "Provided Attributes", "Clinical Factors" and "Disease/Treatment Factor". All these factors will be analyzed based on inter and intraclass mean and variances.

Bucket	Variable	Variable Description
Unique Row Id	Patient ID	Unique ID of each patient
Target Variable	Persistency Flag	Flag indicating if a patient was persistent or not
Demographics	Age	Age of the patient during their therapy
	Race	Race of the patient from the patient table
	Region	Region of the patient from the patient table
	Ethnicity	Ethnicity of the patient from the patient table
	Gender	Gender of the patient from the patient table
	IDN Indicator	Flag indicating patients mapped to IDN
Provider Attributes	NTM - Physician Specialty	The specialty of the HCP that prescribed the NTM Rx
Clinical Factors	NTM - T-Score	T Score of the patient at the time of the NTM Rx (within 2 years prior from rxdate)
	Change in T Score	Change in T-score before starting with any therapy and after receiving therapy (Worsened, Remained Same, Improved, Unknown)
	NTM - Risk Segment	Risk Segment of the patient at the time of the NTM Rx (within 2 years days prior from rxdate)
	Change in Risk Segment	Change in Risk Segment before starting with any therapy and after receiving therapy (Worsened, Remained Same, Improved, Unknown)
	NTM - Multiple Risk Factors	Flag indicating if a patient falls under multiple risk category (having more than 1 risk) at the time of the NTM Rx (within 365 days prior from rxdate)
	NTM - Dexa Scan Frequency	Number of DEXA scans taken before the first NTM Rx date (within 365 days prior from rxdate)
	NTM - Dexa Scan Recency	Flag indicating the presence of Dexa Scan before the NTM Rx (within 2 years prior from rxdate or between their first Rx and Switched Rx; whichever is smaller and applicable)
	Dexa During Therapy	Flag indicating if the patient had a Dexa Scan during their first continuous therapy
	NTM - Fragility Fracture Recency	Flag indicating if the patient had a recent fragility fracture (within 365 days prior from rxdate)
	Fragility Fracture During Therapy	Flag indicating if the patient had fragility fracture during their first continuous therapy
	NTM - Glucocorticoid Recency	Flag indicating usage of Glucocorticoids (>=7.5mg strength) in the one-year look-back from the first NTM Rx
	Glucocorticoid Usage During Therapy	Flag indicating if the patient had a Glucocorticoid usage during the first continuous therapy
Disease/Treatment Factors	NTM - Injectable Experience	Flag indicating any injectable drug usage in the recent 12 months before the NTM OP Rx
	NTM - Risk Factors	Risk Factors that the patient is falling into. For chronic Risk Factors complete lookback to be applied and for non-chronic Risk Factors, one-year lookback from the date of the first OP Rx
	NTM - Comorbidity	Comorbidities are divided into two main categories - Acute and chronic, based on the ICD codes. For chronic disease, we are taking a complete look back from the first Rx date of NTM therapy and for acute diseases, a period before the NTM OP Rx with one-year lookback has been applied
	NTM - Concomitancy	Concomitant drugs recorded before starting with a therapy(within 365 days before the first rxdate)
	Adherence	Adherence to the therapies

Demographics – Imbalance or Bias in the Dataset

Collected Subject data based on Gender, Race, Age, Region, Ethnicity, and IDN Indicator were assessed in MS Excel using Pivot Table and visualized using Pivot Charts as shown in Figure 2. The extract from the descriptive analysis is that the dataset has some imbalance in terms of demography, from some aspects. For example, the dataset has around 94% female patients compared to only 6% male which shows that

the dataset is biased towards females. It is a matter of research that whether the dataset is focused on a certain disease that occurs most to females or females of certain age-class are the ones to reach for medications in large numbers. Nevertheless, due to this class imbalance, any outcome from the dataset will be more representative for female patients.

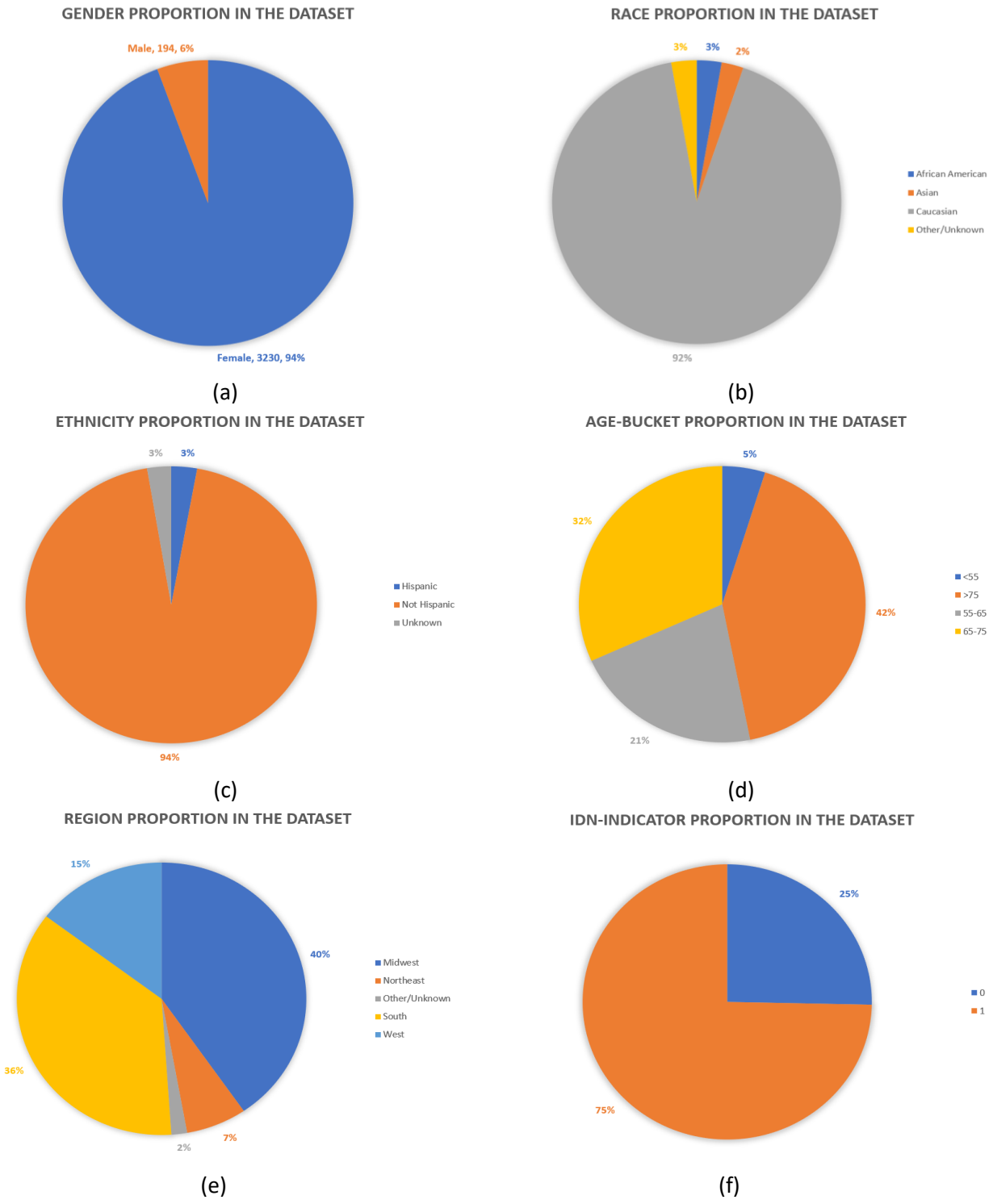

Figure 2: Visualization of the Descriptive Analysis on Demographics of the Subjects

On the other hand, the dataset was dominated by non-Hispanic, Caucasian patients in large proportions. Even though the Region and Age classes were more proportionate, it is clear that most of the patients belonged to higher age classes (only a few lower than 55) and the greatest number of patients came from the Midwest region. One of the most clinically important factors in this analysis was the high proportion of the subjects belonging to the IDN. Around 75% of the subjects belonged to a certain IDN implying the success of forming clusters of health service providers for better patient experience.

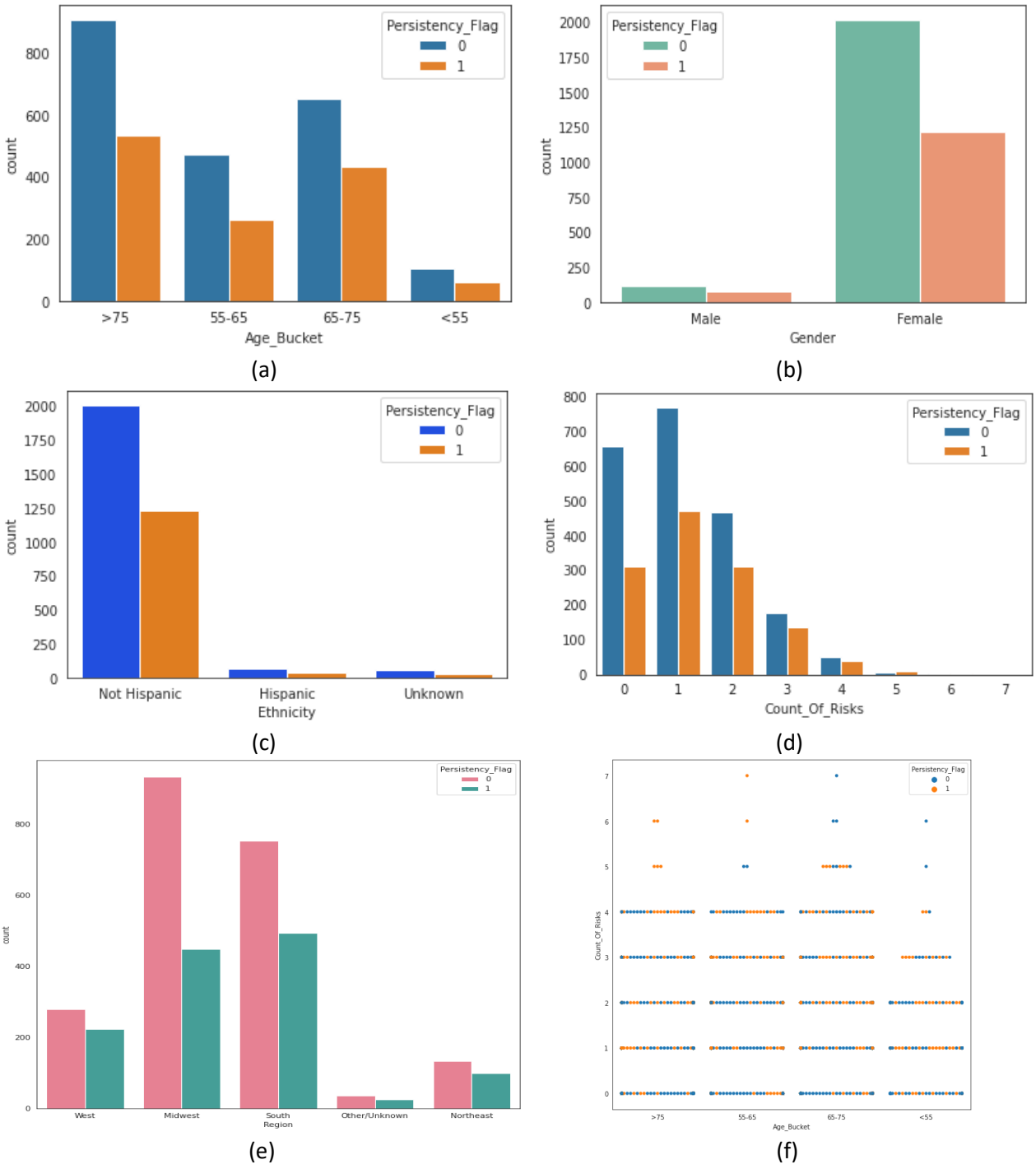

Figure 3: Visualization of Persistency Flags Related with Respect to Other Parameters

In Figure 3, the target variable “Persistency Flag” with respect to some independent variables or predictors have been plotted. It can be seen than most people on the dataset belonged to the aged class, but the non-persistency level (or ratio) is more among the older patients. As discussed earlier, this study has been imbalanced towards female subjects but among females, the non-persistency level is higher than the males. But no concrete conclusion can be drawn due to the data imbalance. Also, low-risk patients were found to be less persistent than the high-risk ones, as shown in Figure 3(d) and 3(f).

Physician Specialty Type and Specialist Flag for the Observing Physician

In this section, the focus was given to the practitioners involved in the cases based on their specialty type and specialist flag (expert on their respective departments).

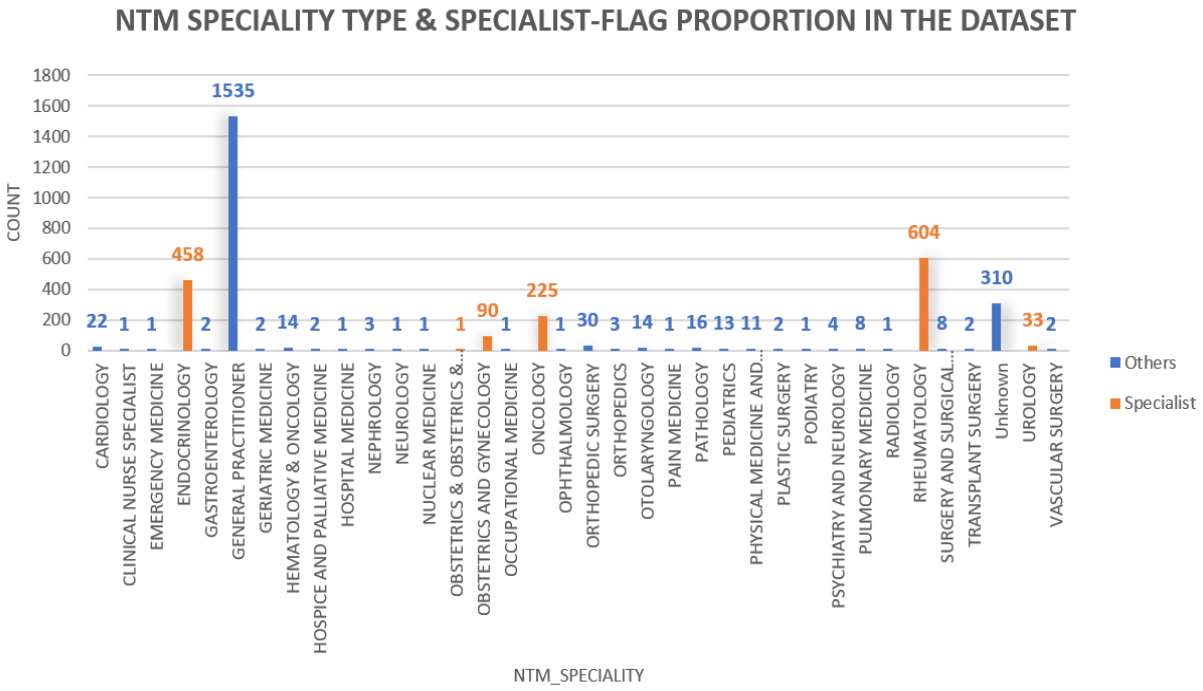

Figure 4: Specialty and Specialist-Flag of the Observing Medical Staff or Physician for the Patient

It can be observed that a large number of physicians who handled the NTM cases were general practitioners, around 45% of the total. But among the other groups, all were not specialists. As the specialist flags indicate, only the physicians belonging to “Endocrinology”, “Obstetrics and Gynecology”, “Rheumatology” and “Urology” were specialists in those respective fields. It might indicate an important assumption that NTM is more critical for these categories of patients, so specialists had to be involved.

Clinical Factors

The clinical factor can be divided into few major classes such as Comorbidity of Diseases, Concomitancy of Drugs, and Risk Factors among the subjects. Percentage-based column charts are plotted for each sub-category of them and shown in Figures 5-7 respectively. The concomitancy of some drugs is around 35% while other drugs can be as low as 10% of subjects. In this case, the Cholesterol mitigating drug was most commonly used by the subjects alongside the main drug. On the other hand, the comorbidity parameter varies a lot as well-meaning that some diseases are comorbid with NTM than other ones. The comorbidity of Lipoprotein Disorder is around 51%, which is the highest, meaning that these diseases occur commonly together with the main disease.

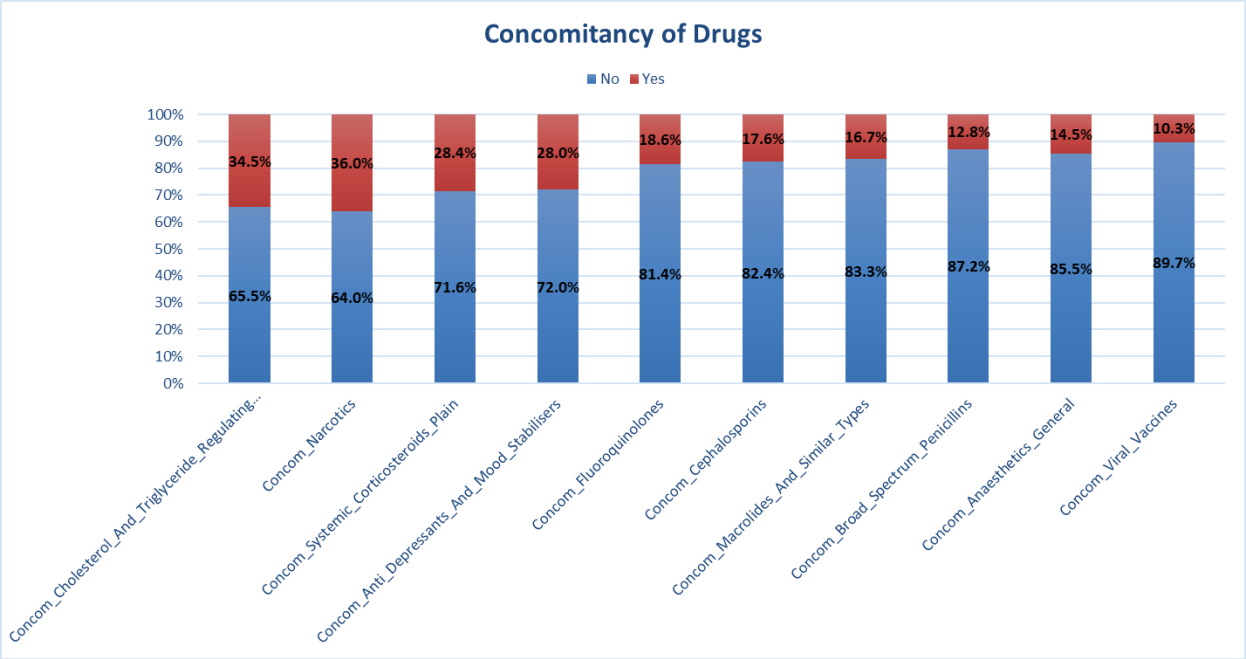

Figure 5: Concomitancy of Various Drugs among Subjects

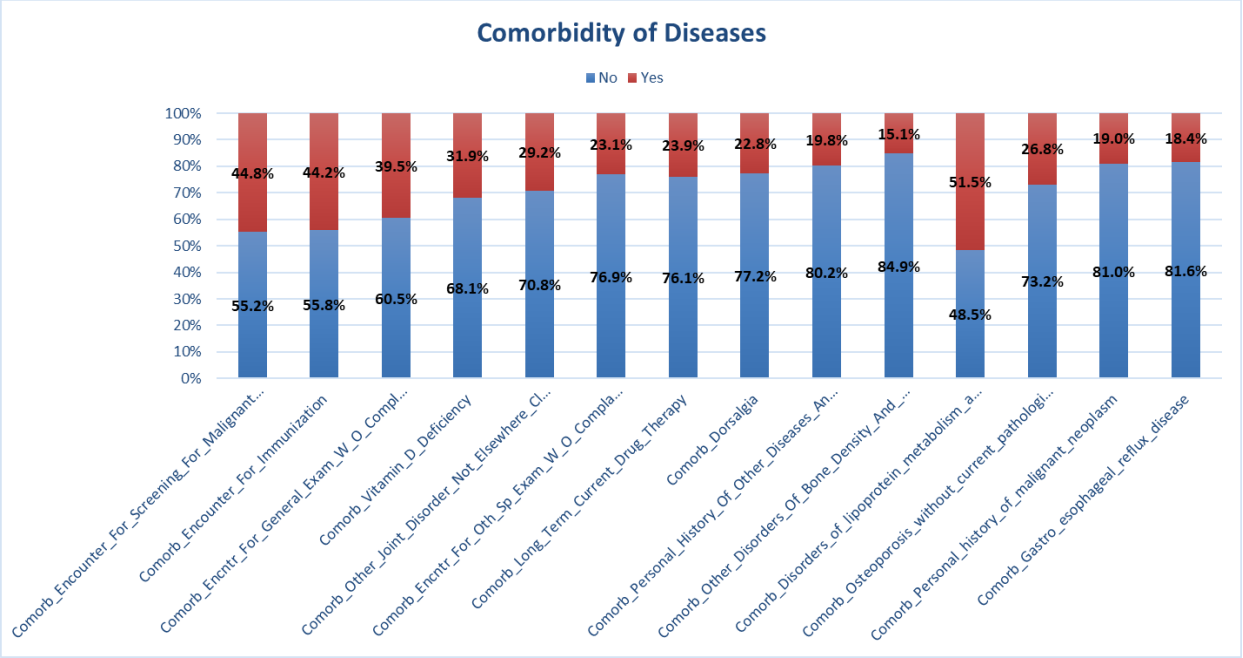

Figure 6: Comorbidity of Various Diseases among Subjects

Various risk factors were observed for the subjects under the study. Among the risk factors, the risk of Vitamin D insufficient was the most acute while the risk of smoking tobacco and the risk of chronic malnutrition were also other influential factors. Nevertheless, the strength of the relations between various clinical factors can be understood from the Machine Learning and Dominance Analysis discussed in the next sections.

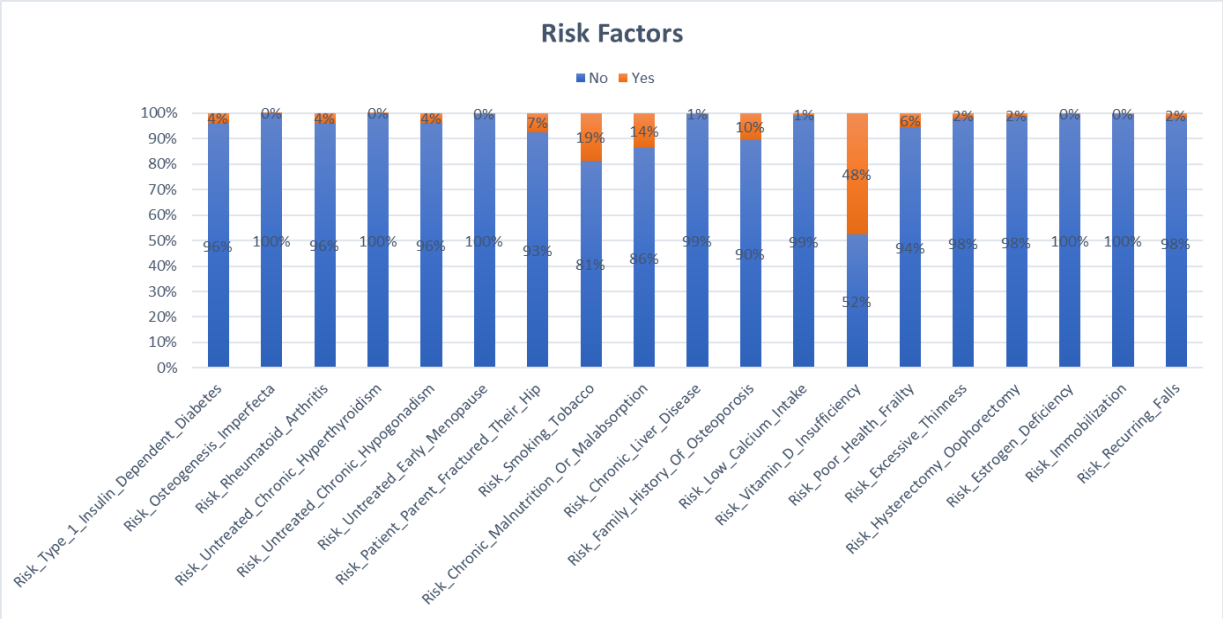

Figure 7: Risk Factors among Subjects

Recommendations

From the Exploratory Data Analysis (EDA) done on the dataset, following recommendations are given to the ABC company's technical team:

- Demographic Factors provided in the dataset is not strongly related to the "Persistency Level" of the patients.
- NTM Specialist type or Specialist Flag did not show any correlation to the target variable.
- Clinical Factors such as "Concomitancy of Drugs", "Comorbidity of Various Diseases" and "Risk Factors" do show some correlations with the target variable "Persistency Level" of the patients which needs to be investigated further through a Quantitative Analysis such as Machine Learning.

Data Cleansing Procedures: Analysis of Problems in the Dataset

This section briefly discusses the existence of any problem in the dataset such as Null values, Outliers, and Skewness, and the tests performed on the dataset to detect these issues.

Presence of Null Values and Outliers in the Dataset

Using Python PANDAS library's Null value detection commands, the number of null values present in each column and the entire dataset was tested. No Null value could be detected in the dataset proving it to be clean in this aspect. The python code output is provided in Figure 8. Null values are important to be detected and removed or replaced from the dataset since they provide errors while performing Machine Learning and other tasks. Even if they do not show any error during any analysis, output from them is not important or relevant to any analysis. Almost all the variables, independent or dependent, in the dataset were categorical and for this reason, there was no presence of any outlier or abnormal data in the dataset. On the other hand, the dataset did have some imbalance and skewness for some predictors which is discussed in detail in the next section.

Skewness and Kurtosis in the Dataset

Using Excel's Data Analysis toolbox, various important statistical parameters for the dataset variables were calculated, as shown in Figures 9-11. From Figure 4, the dataset had similar errors for all the variables, no extremity could be noticed while from Figure 5, the demographic parameters were least skewed (closer to being Normal) even though some parameters like Gender were imbalanced. Most skewed were the Risk parameters. Similar observation for the Kurtosis (Figure 9) as it is directly related to Skewness.

```
Total NULL values in the Original DataFrame = 0
Female                          0
Male                            0
African American                0
Asian                           0
Caucasian                       0
                               ..
Risk_Estrogen_Deficiency        0
Risk_Immobilization             0
Risk_Recurring_Falls            0
Count_Of_Risks                  0
Persistency_Flag                0
Length: 119, dtype: int64
```

Figure 8: Testing Presence and Number of Null Values in the Dataset

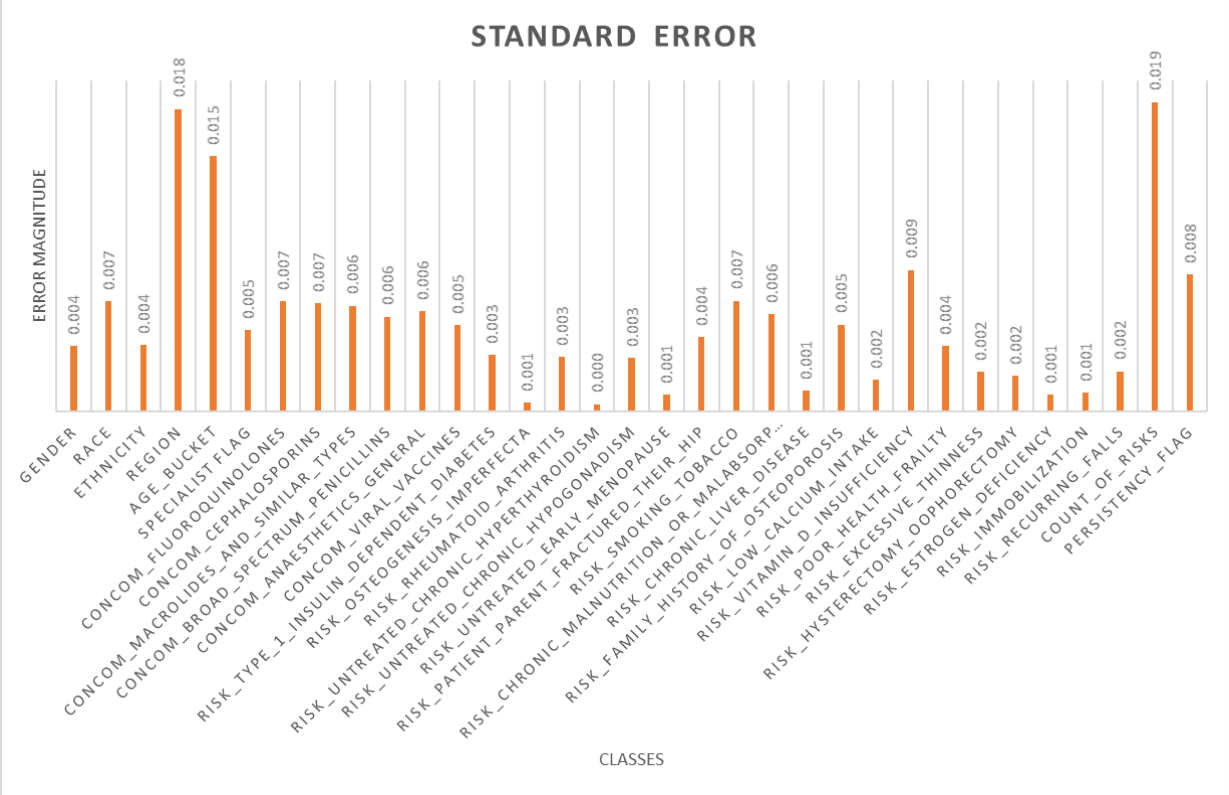

Figure 9: Standard Errors of the Dataset Variables

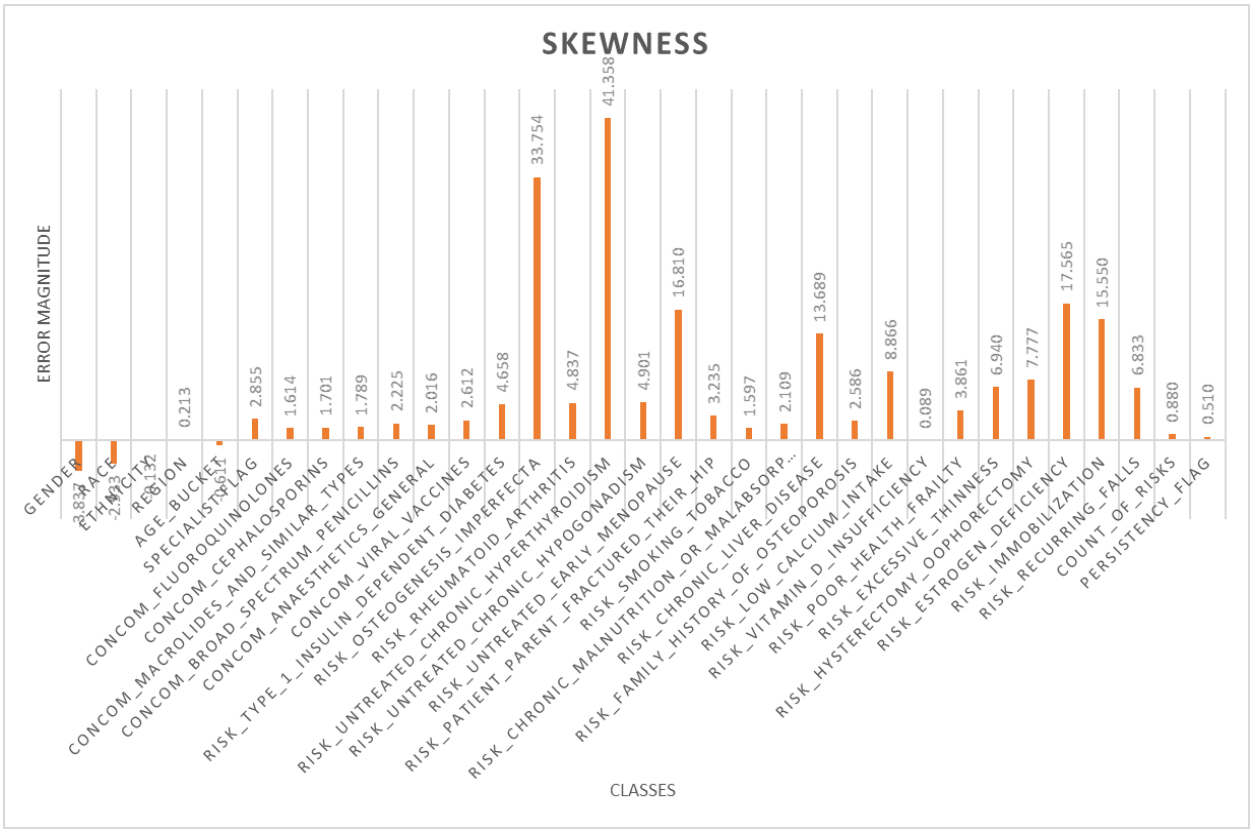

Figure 10: Skewness of the Dataset Variables

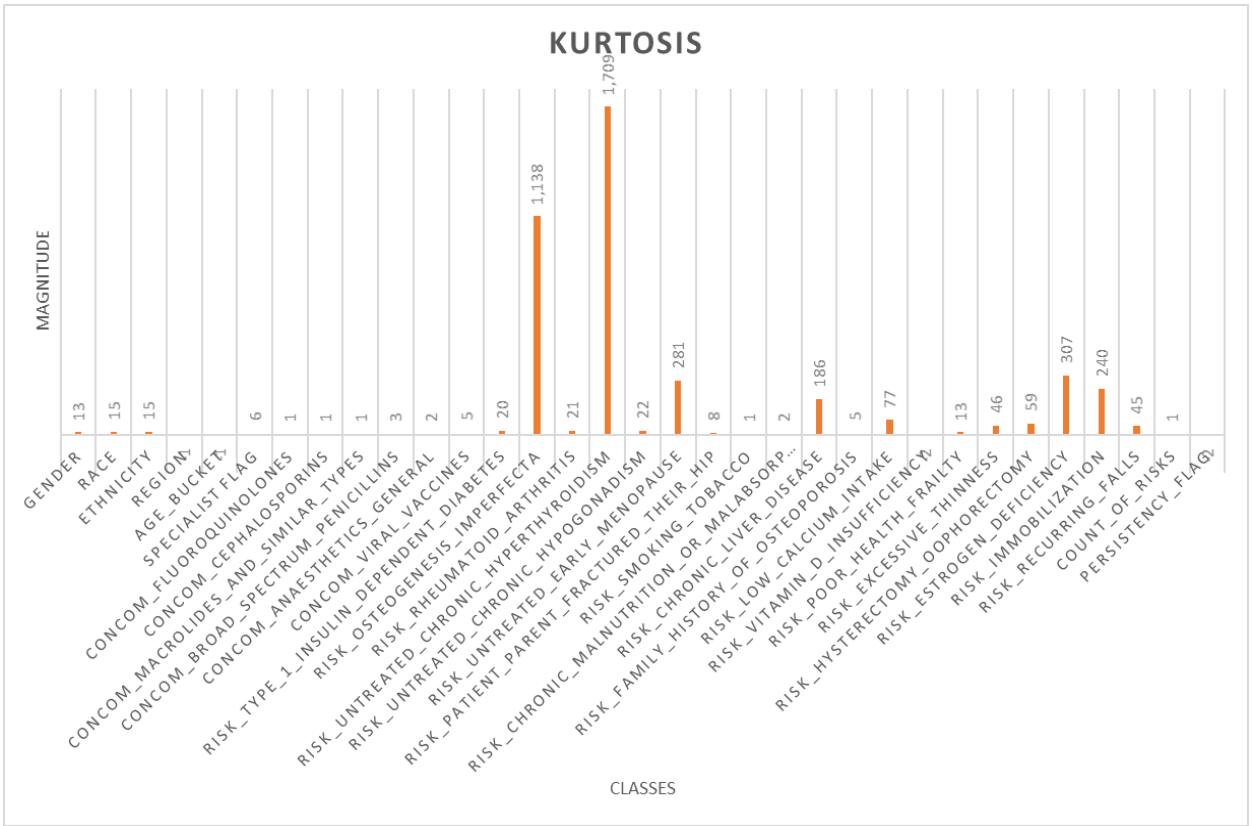

Figure 11: Kurtosis of the Dataset Variables

Data Wrangling and Transformation Approaches

The dataset contains a mixture of categorical (object type) and numerical variables which are problematic for data analysis. Moreover, the dataset might contain independent variables which are interrelated to each other. From a Linear Algebraic point of view, this type of redundancy might cause a singularity in the dataset and less variability in the dataset will downgrade the prediction performance. So, in this section, we discuss the few approaches we followed to solve this issue by creating dummy variables, taking only non-correlated predictors, finding out low-dimensional representational feature-map of the whole dataset, so on and so forth.

Create Dummy Dataset

As discussed earlier, among 68 parameters, only 2 are numerical, the rest are categorical in the dataset. To convert the data to numerical, we have to apply encoding techniques to it, so it can be translated into numerical labels instead namely Dummy Variables (Example in Figure 12). The Dummy Dataset had dummy 119 variables produced from 68 original parameters. So, the dummy dataset has 119*3424 = 407456 total data points, compared to 68*3424 = 219136 data points previously. So, the dummy dataset has around twice the data points as the original dataset. To create the dummy dataset, pandas.get_dummies(X) function was applied to each column containing categorical variables (denoted by 'X') [3]. The Dummy dataset, which represents a numerical version of the whole original dataset, was used in the regression analysis using Machine Learning techniques. 118 of the independent variables were used to predict the dependent variable "Persistency Flag".

Turn your Categorical Column (Ex: "Name")...

Index	Name	8/6/2020
0	Liho Liho	$234.54
1	Chambers	$45.74
2	The Square	$56.22
3	Liho Liho	$32.31

...Into Dummy Indicator Columns

Index	Liho Liho	Chambers	The Square	8/6/2020
0	1	0	0	$234.54
1	0	1	0	$45.74
2	0	0	1	$56.22
3	1	0	0	$32.31

Figure 12: Encoding Techniques to be Applied on the Dataset

Feature Extraction using Autoencoder

Autoencoders can be used to extract features from a large dataset. The feature map can act as a low-dimensional representation of the dataset. Depending on the sparsity and variability level of the dataset, the optimum number of features can be determined. For example, if the data is containing too much redundancy, only a few important features can represent the whole dataset and perform almost the same during any ML or other statistical analysis. Features can be extracted manually, and they can be ranked based on their relevance to the target variable(s). But autoencoders try to do this task

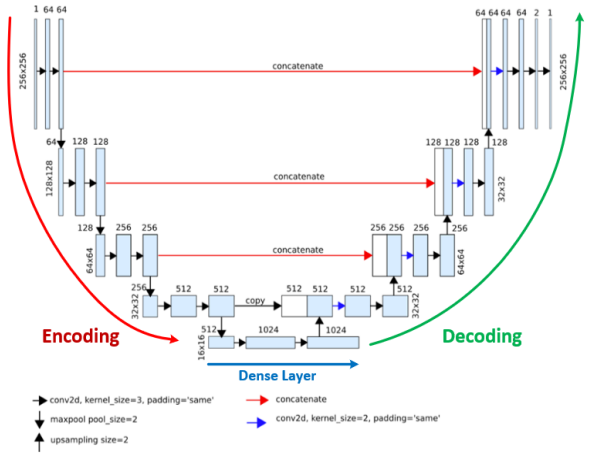

Figure 13: Structure for the U-Net based Autoencoder

automatically, hence the name autoencoder came forth. Autoencoders can be constructed in a few ways, the most common type is Vanilla Autoencoders [4]. But for this project, our team aims to use the encoder portion of the U-Net Deep Segmentation network (Figure 13) [5] to encode the dataset into a compact, representative set of features and extract it from the deepest dense layer of the network, as shown in Figure 13. The U-Net, which was originally developed with an aim of image segmentation, has also been extensively used in the 1D domain such as PPG to ABP signal reconstruction [6]. The U-Net network has been annotated in Figure 12 where it can be seen that U-Net mainly consists of an encoder part that encoded the dataset into a compact set of features and a decoder part which decodes this set of features into a target entity (signal or image). There is an optional dense layer that has been added to extract features from the U-Net-based autoencoder. As discussed earlier, the optimum number of features required for the maximum performance is a matter of trial and error as it depends on the variability of the dataset which is unique for each one. It is to be remembered that the autoencoder aims to extract a low-dimensional feature map from the dataset. If the number of required features is more than the independent variables, there is no use of the autoencoder. For example, if the optimum of feature is 128 for this dataset while there are 118 independent variables in the original dataset, the used autoencoder is not useful except there has been a considerable jump in performance. While maintaining the performance, if a compact feature set can represent the whole dataset, the autoencoder experiment is a success. The complete pipeline for this experiment is shown in Figure 14.

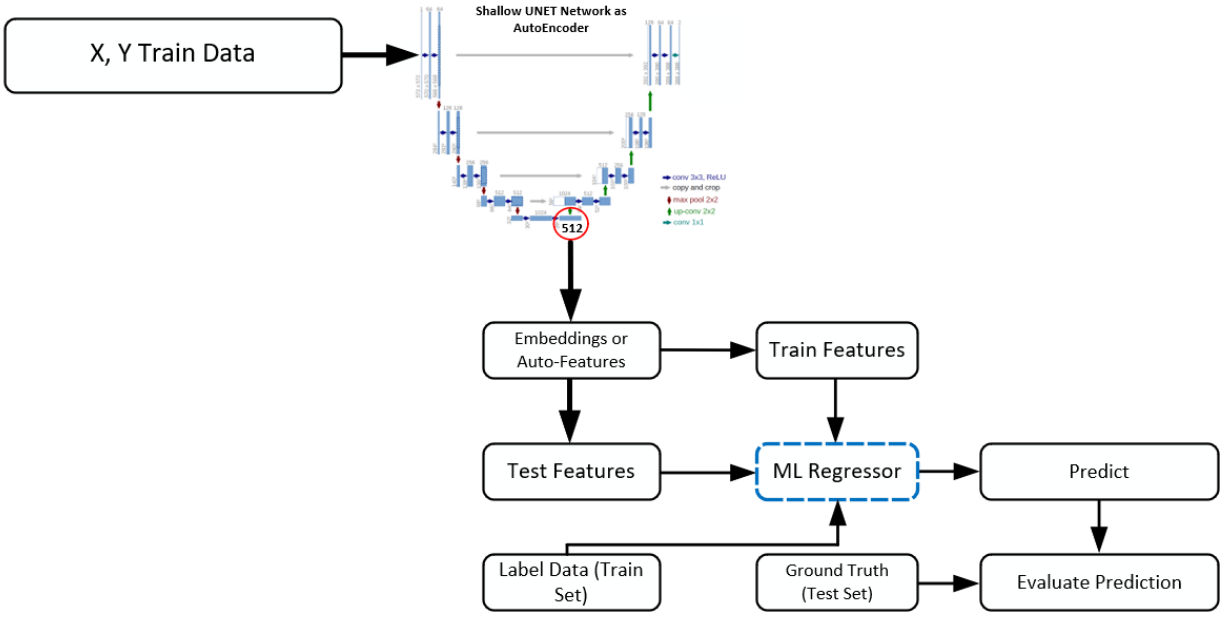

Figure 14: Autoencoder Pipeline

Autoencoder is used to extract features from both train and test sets. Then they are trained using Machine Learning regression algorithms for making predictions.

Dominance Analysis for Data Transformation

As described by Azen and Budescu [7], Dominance Analysis meets three important criteria for measuring relative importance. First, the technique should be robust i.e., should be able to reduce error in predicting the target variable. Second, it facilitates a direct comparison of parameters contributing to the model's performance. Finally, the technique should be able to measure the attributes' direct effect

(self-contribution), total effect (influence when considered with other attributes), and partial effect (influence when considered with various combinations of other predictors). Based on these points, the Dominance Statistics can be divided into four different types of measures such as,

- **Individual Dominance:** Individual dominance is the R^2 of the model between a certain predictor (or independent variable) and the dependent variable. So, the individual dominance, which can be formulated as $R^2_{Y.X_1}$ for a predictor X_1, represents the quantum of impact by the predictor in absence of others.
- **Average Partial Dominance:** The average partial dominance measures the average impact of a predictor when it is tested against all possible combinations formed by other predictors except when all other predictors are available. Statistically speaking, this is just the average of average incremental R^2 contribution of the target-independent variable to all subset models except the final model and bi-variate.
- **Interactional Dominance:** Interactional dominance can be expressed as the impact or variability described by a predictor in presence of all other predictors. In other words, the interactional dominance of a certain independent 'X_1' will be the difference between the R^2 of the overall model and the R^2 of the model with all other predictors except 'X_1'. For example, if X_1, X_2, X_3, and X_4 are four independent variables, the interactional dominance can be expressed as, $R^2_{Y.(X_1,X_2,X_3,X_4)} - R^2_{Y.(X_1,X_2,X_3)}$
- **Total Dominance:** It is the average of conditional values from all types of dominance factors for a certain predictor thus summarizing the contributions of each predictor to all subset models.

In the case of Dominance Analysis, if there are 'p' predictors, there will be $2^p - 1$ model i.e., all possible subset models, and the incremental contribution of each predictor is evaluated for the model(s) created by the combinations of all other predictors. So, if there are 4 independent variables, there will 4_{C_1} = 4 models with only 1 predictor, 4_{C_2} = 6 models with two predictors, 4_{C_3} = 4 models with three predictors and 4_{C_4} = 1 model with all predictors i.e., total $2^4 - 1$ = 15 models. So, the complexity of the procedure increases geometrically as the number of predictors increase. Since we have 118 independent variables, we will have $(2^{118} - 1)$ subset models to evaluate. The GitHub Library for implementing Dominance Analysis [8] in Python has been used for implementation where the number of top predictors to be accounted for dominance analysis can be varied (the default 15 was used for this experiment).

Model Building and Best Model Selection Process

In the 2nd part of data analysis, various statistical regression techniques were used to classify Drug Persistency of subjects based on the predictor variables. As mentioned earlier, there are 118 predictor variables in the dummy dataset (representing the whole original dataset) and one target variable.

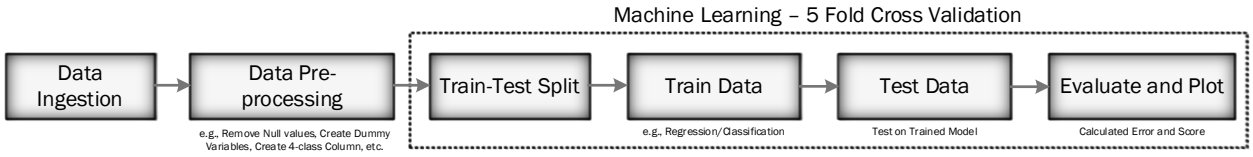

Figure 15. Data Pre-processing and Machine Learning (ML) Pipeline (in Python)

A Data pre-processing and ML pipeline was developed using Python Jupyter Notebooks in the Google COLAB platform. The overall structure of the pipeline is shown in Figure 15. The CSV file containing the dataset was uploaded into Google Drive which is connected to COLAB. Using Python's PANDAS Library, a DataFrame was created from the CSV dataset and used for further analysis. At first, unnecessary data

such as the “Patient ID” column was removed from the dataset and NULL values were filled up (not dropped) by ‘N/A’ or Not Answered. For descriptive analysis, categorical data could directly be used using PANDAS DataFrame while regression techniques can only work on numerical data. So, dummy variables were created from categorical data for regression analysis. All versions of the processed datasets were saved in Google Drive. The dummy DataFrame was converted into a NumPy array (which can only take numerical values) to perform ML on it. The dataset of 3424 subjects was split into train and test set following a ratio of 80:20 using Python’s Scikit-learn library. So, 2739 subjects were randomly taken into the train set and 685 for the test. While normally the test set is smaller than the train set, this standard ratio can be altered to check the performance of the ML model (e.g., 70:30 or 85:15). Since Scikit-learn creates the train-test split randomly, the test data is not completely independent and during each run, the samples inside train and test datasets will be different. It might affect the performance since the randomly chosen test data might represent a certain type of people more, or it can be of completely different characteristics than the training dataset. In both cases, the outcome will be biased and might change during each run. To get rid of this bias, 5-Fold Cross-Validation (CV) was performed while training and testing the data i.e., the entire process of splitting dataset and performing ML was run 5 times and the final result is the average of the results from these 5 runs.

Eleven Regression techniques were used as Classified with Supervised Machine Learning. Among them, there are commonly used techniques like Logistic Regression, K-Nearest Neighbor (KNN), Support Vector Machines (SVM), Stochastic Gradient Descent (SGD), Decision Trees, Ensemble techniques such as Gradient “Tree” Boosting, Random Forest, Extra Trees, AdaBoost, XgBoost, and Artificial Neural Network (ANN) technique like Multi-layer Perceptron (MLP). Moreover, a Deep Neural Network sequential model was developed using KERAS. The Regression Algorithms are below discussed in brief:

Logistic Regression: Logistic Regression is another regression method like Linear Regression but can accept categorical data in the dependent variable which linear regression is inefficient at. Logistic Regression can be used in classification tasks due to its logarithmic, non-linear kernel (Equation 1).

$$Y = \ln\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n - - - Equation\ 1$$

K-Nearest Neighbor (KNN): KNN classifies new data points or cases based on the classes of its neighbors. The number of neighboring points to be tested can be changed for optimum performance on a dataset. Condensed Nearest Neighbor (CNN, the Hart algorithm) can be used to find the border ratio (Equation 2) to efficiently perform KNN on multiple neighbor.

$$a(x) = \frac{\|x' - y\|}{\|x - y\|} - - - Equation\ 2$$

Here, $\|x - y\|$ is the distance to the closest example y having a different color than x, and $\|x' - y\|$ is the distance from y to its closest example x' with the same label as x.

Support Vector Machines (SVM): The primary aim of SVM is to create hyperplanes that can act as decision boundaries during the classification of multidimensional data so that any new data can fall into the correct category. SVM chooses the extreme points/vectors, known as the Support Vectors, which help in creating the hyperplane.

Stochastic Gradient Descent (SGD): SGD is an iterative method for optimizing an objective function by Gradient Descent on a randomly selected portion of the dataset. The cost function for SGD during training of Machine Learning can be performed as shown in Equation 3.

$$cost\left(\theta,\left(x^{(i)},y^{(i)}\right)\right)=\frac{1}{2}(h_\theta\left(x^{(i)}\right)-y^{(i)})^2 \quad --- Equation\ 3$$

Decision Trees: Decision Tree Learning in ML uses a Decision Tree for classification or regression. A Decision Tree is a flowchart-like model based on events and outcomes to reach a certain goal. A decision tree comprises nodes, branches, and leaves where each node is known as a test, each branch represents an outcome of a test, and each leaf represents a class label.

Ensemble Techniques: In Machine Learning, Ensemble techniques such as Gradient "Tree" Boosting, Random Forest, Extra Trees, AdaBoost, and XgBoost use a weaker base model, typically Decision Trees, to produce a more efficient predictive model by variously changing the base model through their algorithms.

Multi-Layer Perceptron (MLP): MLP is a class of (feedforward) Artificial Neural Network or ANN which uses Artificial Neuron as nodes in the network to perform Machine Learning tasks. MLP comprises at least three layers viz. Input, Hidden, and Output layer. MLP uses non-linear functions such as ReLU, sigmoid, etc. as activation functions for each node, which makes them different from a Perceptron, which uses a linear activation function instead. MLP uses backpropagation, a supervised technique during training. The learning process of MLP can be represented by Equation 4.

$$-\frac{\delta\varepsilon(n)}{\delta v_j(n)}=\phi'\left(v_j(n)\right)\sum_k -\frac{\delta\varepsilon(n)}{\delta v_k(n)}w_{kj}(n) \quad --- Equation\ 4$$

Evaluation of the Machine Learning (ML) Techniques: The evaluation of the ML models was performed by measuring the errors, creating the Confusion Matrix, and extracting other parameters such as Accuracy, Precision, Recall (or Sensitivity), and f1-Score from the Confusion Matrix. Their formulae are shown in Equations 5 to 8. Here, TP = True Positive (equivalent with Hit), TN = True Negative (equivalent with Correct Rejection), FP = False Positive (false alarm, type I error, or underestimation), and FN = False Negative (equivalent with miss, type II error, or overestimation) are the four main parameters of the Confusion Matrix.

$$\boldsymbol{Accuracy}=\frac{TP+TN}{TP+FN+TN+FP} \quad --- Equation\ 5$$

$$\boldsymbol{Precision}=\frac{TP}{TP+FP} \quad --- Equation\ 6$$

$$\boldsymbol{Recall}=\frac{TP}{TP+FN} \quad --- Equation\ 7$$

$$\boldsymbol{f1-Score}=\frac{TP+TN}{TP+FN+TN+FP} \quad --- Equation\ 8$$

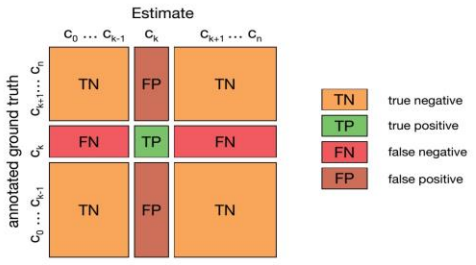

Figure 16. Confusion Matrix in General Format

The Receiver Operating Characteristic or ROC Curves were plotted on the same window for all classes of the dependent variable for the 4-Class problem. The ROC curve is created by plotting the True Positive Rate (TPR, also known as Sensitivity or Recall) against the False Positive Rate (FPR, also known as the Probability of False Alarm) at various thresholds settings to discover the optimal model. The formulae

for TPR and FPR are shown in Equations 9 and 10, respectively. Area Under the ROC Curve, commonly known as AUC, was calculated for each ROC curve. AUC can be found by integrating the ROC curve over the whole interval. AUC as a scale-invariant measure can measure how well the predictions are ranked among classes, regardless of their respective magnitudes. It can also measure the quality of the model's prediction irrespective of the classification threshold.

$$\boldsymbol{True\ Positive\ Rate\ (TPR)} = \frac{TP}{TP + FN} \equiv 1 - False\ Negative\ Rate(FNR) \quad --- Equation\ 9$$

$$\boldsymbol{False\ Positive\ Rate\ (FPR)} = \frac{FP}{FP + TN} \equiv 1 - True\ Negative\ Rate(TNR) \quad --- Equation\ 10$$

There are various types of error calculating metrics such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Median Absolute Deviation (MAD), MAE was chosen since it is one of the most common ways errors measurement among the ML researcher community. The formula for MAE is shown in Equation 11 where the predicted values are being compared to the true/actual labels from the test data.

$$\boldsymbol{MAE} = \frac{\sum_{i=1}^{n} |y_i - \lambda(x_i)|}{n} --- Equation\ 11$$

Testing Results and Plots:

The final test results parameters from three experiments viz. Whole Dataset (Dummy), Feature-Map created by the U-Net based Autoencoder and the subset dataset formed by Dominance Analysis are shown in terms of MAE, Accuracy, Precision, Recall, f1-Score and AUC. The results for all ML networks, including a custom ANN model developed by KERAS for evaluating the Autoencoder is also provided in Table 2.

Table 2: Results

ML Algorithms	Original						Autoencoder						Dominance Analysis					
	MAE	Accuracy	Precision	Recall	f1-Score	AUC	MAE	Accuracy	Precision	Recall	f1-Score	AUC	MAE	Accuracy	Precision	Recall	f1-Score	AUC
Logistic Regression	0.19	0.81	0.81	0.79	0.81	0.88	0.20	0.80	0.79	0.78	0.79	0.87	0.27	0.73	0.73	0.68	0.72	0.76
K-Nearest Neighbour (KNN)	0.22	0.78	0.78	0.73	0.76	0.84	0.21	0.79	0.80	0.78	0.79	0.83	0.31	0.69	0.68	0.63	0.67	0.70
Support Vector Machine (SVM)	0.21	0.79	0.78	0.76	0.78	0.86	0.20	0.80	0.80	0.79	0.80	0.82	0.28	0.72	0.72	0.69	0.72	0.71
Stochastic Gradient Descent (SGD)	0.24	0.76	0.76	0.75	0.76	0.81	0.21	0.79	0.80	0.79	0.79	0.86	0.28	0.72	0.71	0.67	0.71	0.74
Decision Tree	0.27	0.73	0.73	0.71	0.73	0.71	0.25	0.75	0.75	0.73	0.75	0.73	0.35	0.65	0.63	0.59	0.63	0.62
Gradient Boosting	0.19	0.81	0.81	0.78	0.81	0.88	0.20	0.80	0.80	0.79	0.80	0.86	0.28	0.72	0.71	0.68	0.71	0.76
Random Forest	0.19	0.81	0.80	0.78	0.80	0.88	0.22	0.78	0.78	0.76	0.78	0.84	0.33	0.67	0.66	0.63	0.66	0.69
Extra Trees	0.21	0.79	0.79	0.77	0.79	0.87	0.23	0.77	0.77	0.75	0.77	0.84	0.34	0.66	0.64	0.60	0.64	0.67
AdaBoost	0.19	0.81	0.81	0.79	0.81	0.87	0.21	0.79	0.79	0.78	0.79	0.86	0.28	0.72	0.72	0.67	0.71	0.75
XgBoost	0.21	0.79	0.80	0.75	0.78	0.87	0.20	0.80	0.80	0.79	0.80	0.86	0.28	0.72	0.71	0.66	0.70	0.76
Multiple Layer Perceptron (MLP)	0.25	0.75	0.75	0.74	0.75	0.82	0.21	0.79	0.79	0.77	0.79	0.86	0.32	0.68	0.68	0.65	0.68	0.70
ANN Developed with KERAS								0.80	0.78	0.79	0.79							

It can be noticed that AdaBoost performed best on the whole dataset while GradBoost performed best for the Autoencoder feature-map. The subset dataset created out of Dominance Analysis has performed much lower for all networks so this can be discarded. The incredible thing about the autoencoder is that only 2 best features (after trial and error for feature number 2, 4, 8, 16, 32, 64 and 128) were selected for creating the feature map of dimension (3424*2) = 6848 which is 59 times smaller than the whole dataset which has about 404032 parameters! And it performed almost the same as the whole dataset. So, autoencoders in some cases can represent a large dataset in a very compact form while maintain the performance. It allows both producibility, deployment easiness in mobile devices with less computational ability and storages, and portability of models. So, we select the Autoencoder based approach for this project as the final propose model.

The ROC curves and the Confusion Matrix for the best performing model for autoencoder approach i.e., “GradBoost” is shown in Figures 17 and 18, respectively.

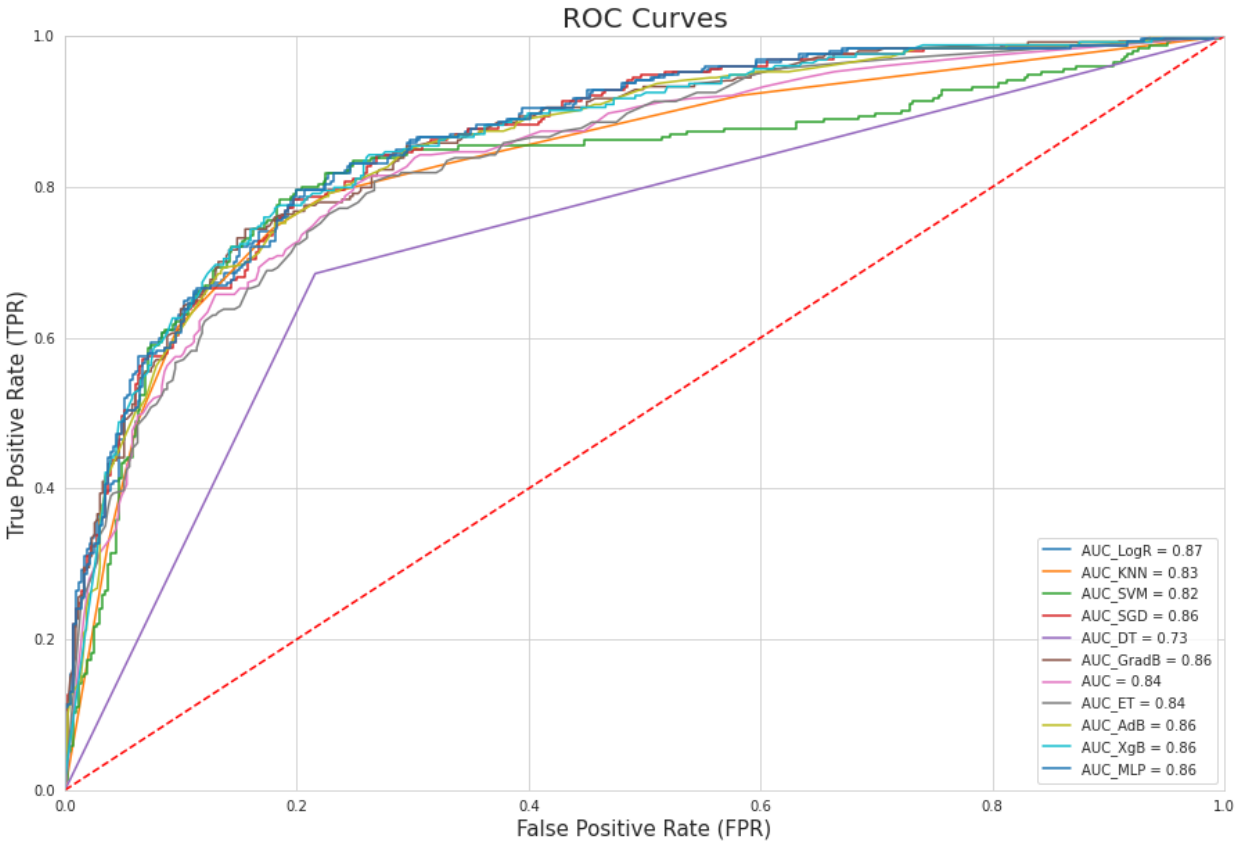

Figure 17. ROC Curves for all ML Models used to Evaluate AutoEncoder Extracted Features

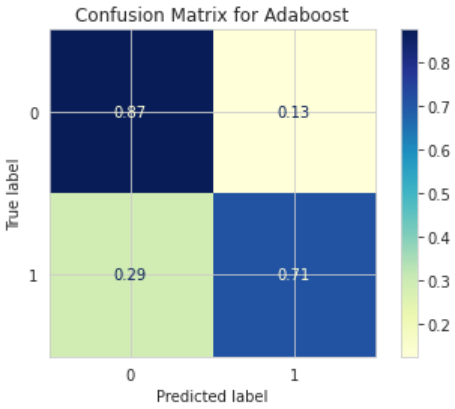

Figure 18. Confusion Matrix for GradBoost (Autoencoder pipeline)

Conclusion

In conclusion, the autoencoder based approach seemed to be the best pipeline for this project which can be further upgraded by tuning the model by changing the U-Net model parameters. The best model was able to detect Patient Persistency with an Accuracy of 81%. More data can be collected especially focusing into removing data imbalance discussed beforehand to improve the performance.

GitHub Repo Link

https://github.com/m-odeh/Persistency-of-a-drug

References

[1] https://decisionresourcesgroup.com/solutions/integrated-delivery-networks-and-their-growing-influence-on-regional-healthcare-in-the-us/

[2] https://www.researchgate.net/publication/5055576_How_to_Project_Patient_Persistency

[3] "pandas.get_dummies — pandas 1.2.4 documentation", Pandas.pydata.org, 2021. [Online]. Available: https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.get_dummies.html.

[4] "Deep inside: Autoencoders", Medium, 2021. [Online]. Available: https://towardsdatascience.com/deep-inside-autoencoders-7e41f319999f.

[5] [3]"Understanding Semantic Segmentation with UNET", Medium, 2021. [Online]. Available: https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47.

[6] "nibtehaz/PPG2ABP", GitHub, 2021. [Online]. Available: https://github.com/nibtehaz/PPG2ABP. [Accessed: 15- May- 2021].

[7] Azen R, Budescu D V. The Dominance Analysis Approach for Comparing Predictors in Multiple Regression. Psychol Methods 2003;8:129–48. https://doi.org/10.1037/1082-989X.8.2.129.

[8] "dominance-analysis/dominance-analysis", GitHub, 2021. [Online]. Available: https://github.com/dominance-analysis/dominance-analysis. [Accessed: 15- May- 2021].