**Patience Persistence Classification Model**

**Our data has a total of 3424 rows and 69 columns. Our target variable is “Persistence\_Flag”.**

The processed data called “Final\_Data” is used for model building and evaluation purposes. The following changes have been made to the original data to create “Final\_Data

1. **Outlier Treatment**: Our numerical features Dexa\_Freq\_During\_Rx and Count\_Of\_Risks contain outliers. Hence, we use the Interquartile Range (IQR) method to help remove the outliers. We replace outliers with upper limit value if their value exceeds the upper limit and we replace outliers with lower limit value if their value exceeds the lower limit.
2. **Dropped Columns**: We drop the column, Ptid as it is not relevant to our analysis.
3. We check for the distribution of the categorical variables and drop those which are unevenly distributed. Retaining these variables will lead to more misclassification & induce error (The full list of dropped categorical variables is mentioned in the Python script).
4. **Variable Transformation**: The categorical variable, Ntm\_Speciality has too many sub-categories with low frequency of values. Hence, we club those sub-categories having < 1% frequency as "Other" into a single category.
5. **Removing Duplicates**: We check for duplicates in our data and remove them.
6. **Label Encoding**: We convert our categorical variables into numerical variables using label encoder.
7. **Standardizing**: We have also standardized the data.
8. **Feature Importance**: Using feature importance scores we keep only the 15 features with relatively high scores for use in our model. A complete list of those 15 features is mentioned in the Python script.

* **Why did we use Random Forest?**

In order to decide which classification model we would be using for our prediction, we check the accuracy of 7 different classification models such as, Logistic Regression, KNN Classifier, XGBoost Classifier, Random Forest Classifier, Support Vector Machines (SVM) Classifier, Decision Tree Classifier, and GB Classifier.

We realize that Random Forest classifier has the highest score among all the 7 classification models and hence, we train our model on it.

**Note:** Apart from the above-mentioned points, it would be useful to highlight that a **Grid Search** algorithm has been used to fine-tune the parameters of Random Forest. We use a “grid” of values for the different parameters and the model will choose the value using which it performs best.