Infosys Springboard

Telecom Churn Prediction

Infosys Springboard Batch - 2

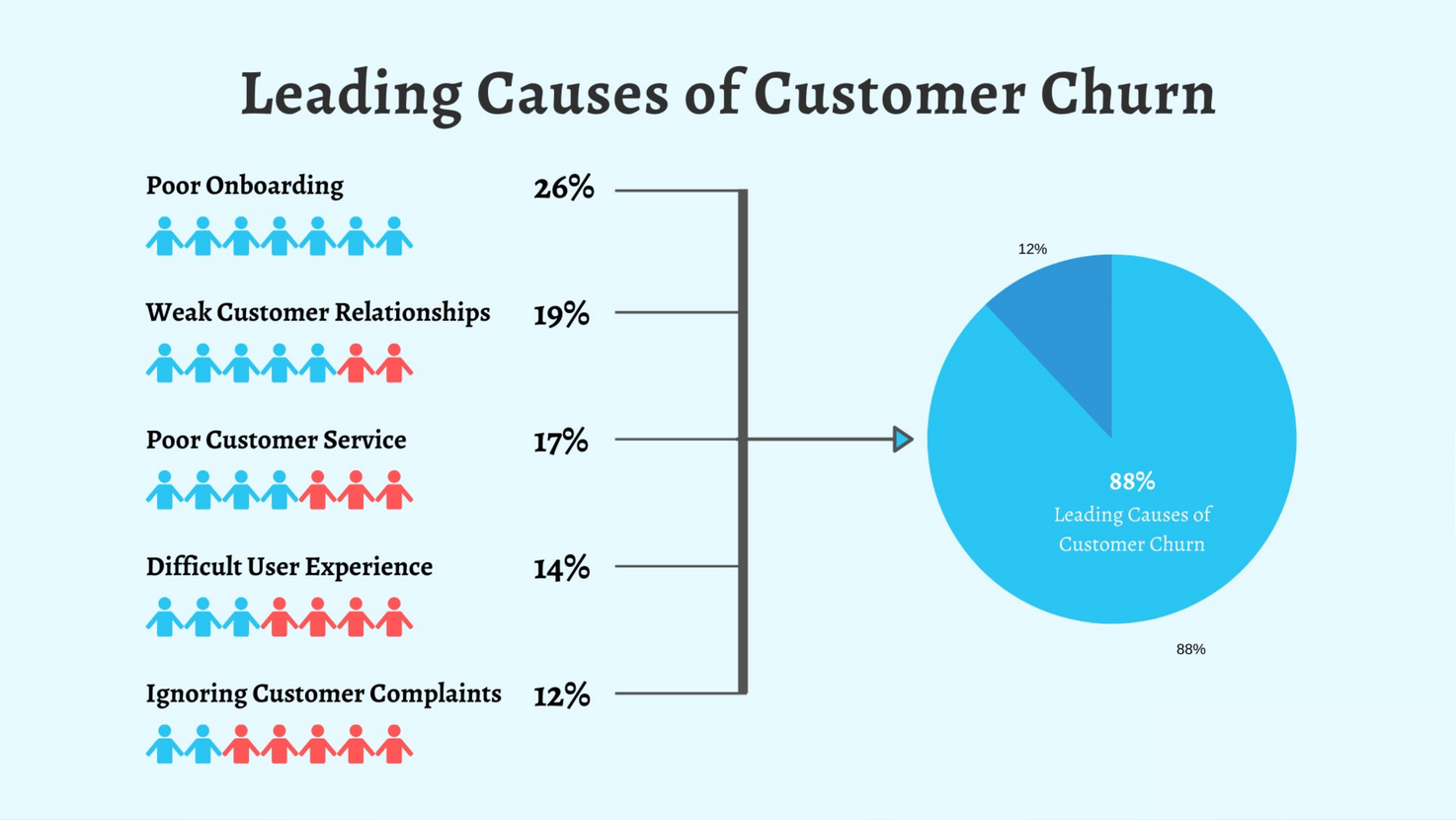
Abhishek Gadge
Mentor : Mr. Bhaskar Naidu
Github : https://github.com/springboardmentor113/Batch2_Churn_Modeling_On_Telecom_Data/tree/main/Abhishek%20Gadge

Contents

1. What is Telecom Churn Prediction ?

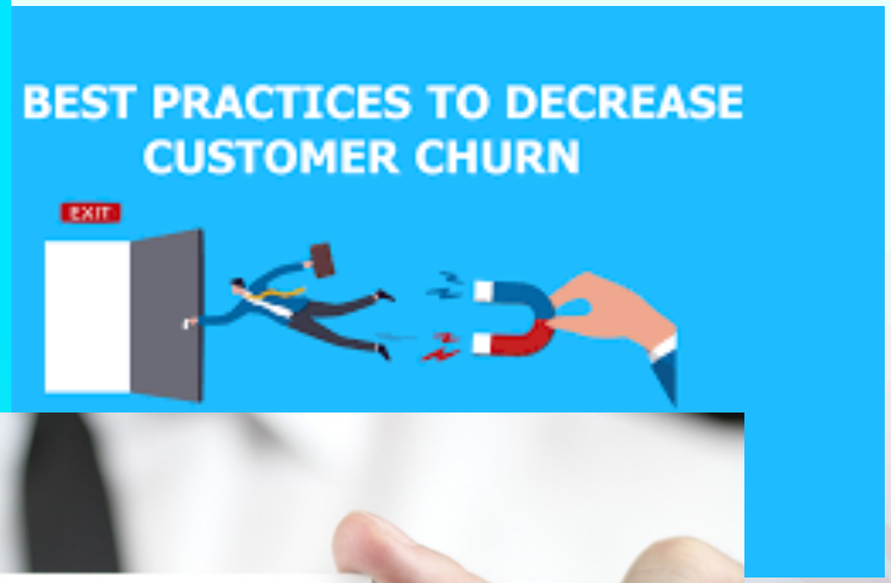
Telecom churn prediction is the process of using data analysis and machine learning techniques to identify customers who are likely to leave a telecommunications service provider.

Why do Customers Churn ?



Leading causes of customer churn include poor onboarding (26%), weak customer relationships (19%), poor customer service (17%), difficult user experience (14%), and ignoring customer complaints (12%); these are just a few examples, as the real problem is yet to be discovered through further analysis.

Our goals



Our goal is to develop a predictive model to accurately identify potential customer churn and implement strategies to retain them.

Methodology

Data Collection,
Data Understanding

Week 1

We follow a structured approach involving data collection, preprocessing, model building, evaluation, and interpretation to predict churn.

Model Evaluation
Model Interpretation
Documentation

Week 4

Week 2

Data Cleaning and
Preprocessing
Exploratory Data Analysis
(EDA)

Week 3

Machine Learning Model Implementation
Hyperparameter Tuning

Data Understanding

Data understanding involves analyzing the dataset to gain insights into its structure and the relationships between variables.

Shape of the Data:

Indicates the dataset contains 25,000 records and 111 features, giving an idea of the data's size and complexity.

Checking the Datatype:

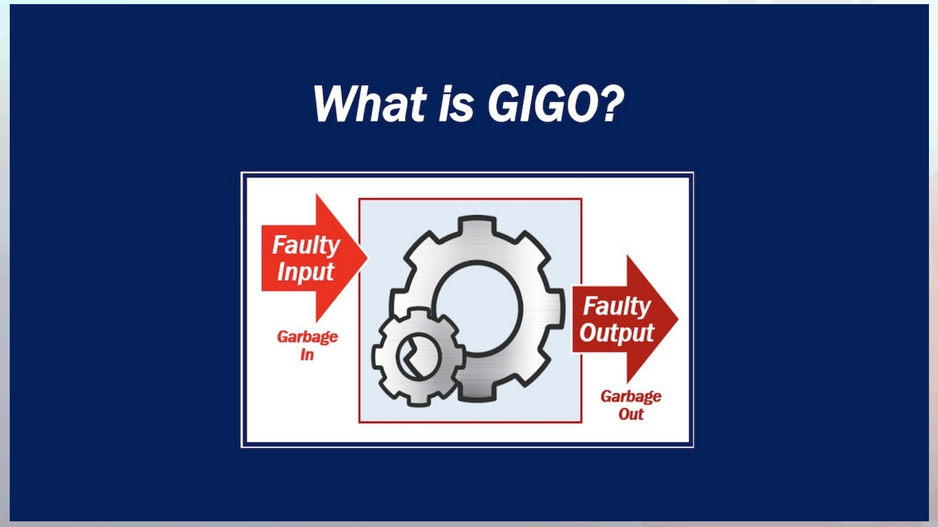
Shows that the dataset comprises 80 float and 31 integer features, informing the types of preprocessing steps required.

Checking the Statistical Summary:

Provides a summary of the dataset's central tendency, dispersion, and shape of the distribution, aiding in understanding feature behavior.

Data Cleaning and Preprocessing

Data cleaning and preprocessing involve handling missing values, outliers, and transforming data to make it suitable for analysis.



Data cleaning and preprocessing involve handling missing values, outliers, and transforming data to make it suitable for analysis.

Remove Duplicate Records:

Confirms there were no duplicate records in the dataset, ensuring data integrity remains intact.

Remove Columns with a Single Unique Value:

No columns with a single unique value were found, so the dataset's feature count remains unchanged.

Remove Zero Variance Variables:

No zero variance variables were found, confirming all numerical features provide some level of information.

Handling Missing Values:

Indicates there are no missing values in the dataset, simplifying further analysis and model building.

Exploratory Data Analysis

Using Boxplot to Visualize Outliers:

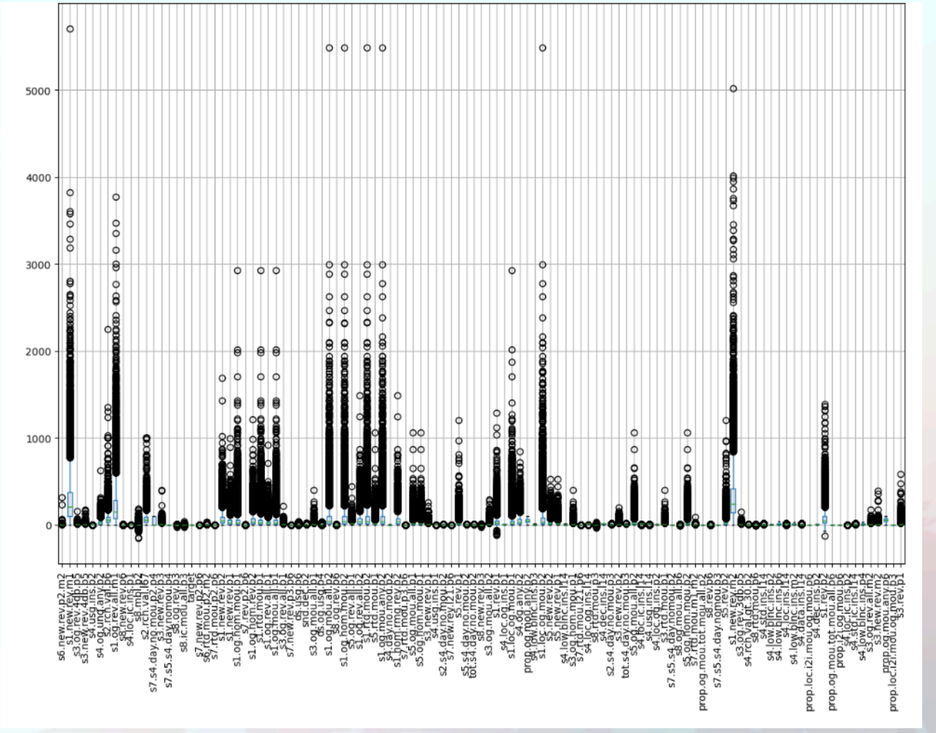
Helps identify the presence of outliers across different features for subsequent treatment.

Capping and Flooring the Outliers:

Reduces the impact of extreme values while retaining the majority of data distribution.

3-Sigma Approach for Outlier Treatment:

Another method to treat outliers, ensuring data falls within a more reasonable range.



Exploratory data analysis (EDA) is the process of visually and statistically exploring data to uncover patterns and relationships.

Distribution of Numerical Features:

Provides insights into the distribution and range of numerical features post-scaling.

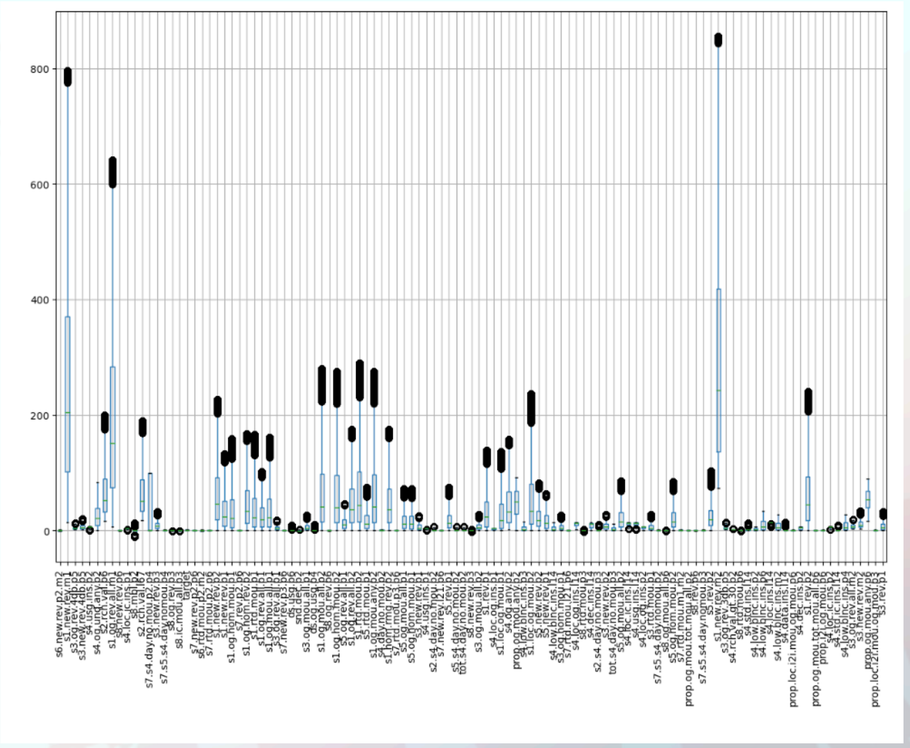
Creating a Correlation Matrix:

Identifies the strength of relationships between features, aiding in feature selection

Exploratory Data Analysis

Using Boxplot to Visualize Outliers:

Helps identify the presence of outliers across different features for subsequent treatment.

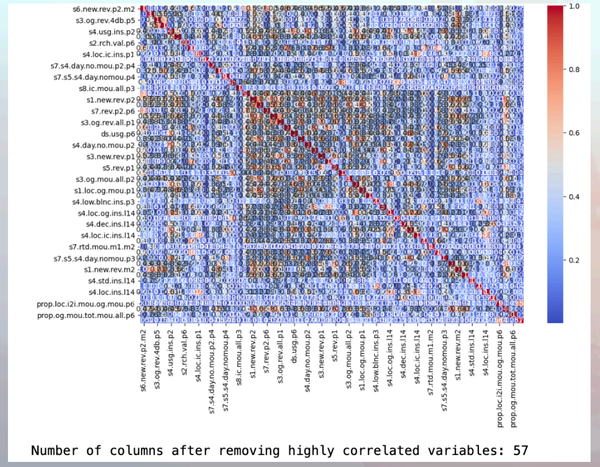


Capping and Flooring the Outliers:

Reduces the impact of extreme values while retaining the majority of data distribution.

3-Sigma Approach for Outlier Treatment:

Another method to treat outliers, ensuring data falls within a more reasonable range.



Distribution of Numerical Features:

Provides insights into the distribution and range of numerical features post-scaling.

Creating a Correlation Matrix:

Identifies the strength of relationships between features, aiding in feature selection

Exploratory Data Analysis

Using Boxplot to Visualize Outliers:

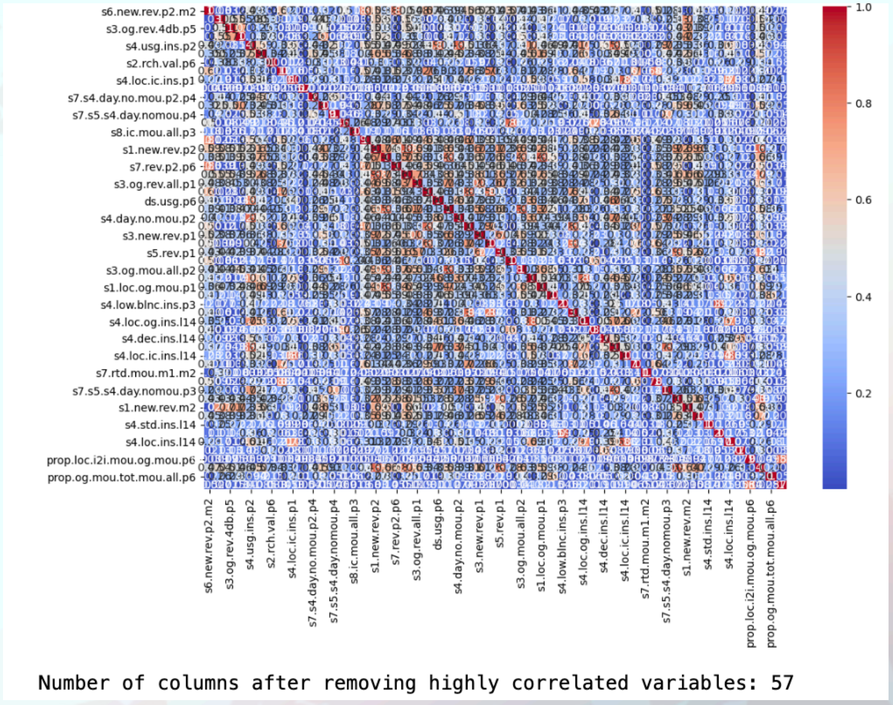
Helps identify the presence of outliers across different features for subsequent treatment.

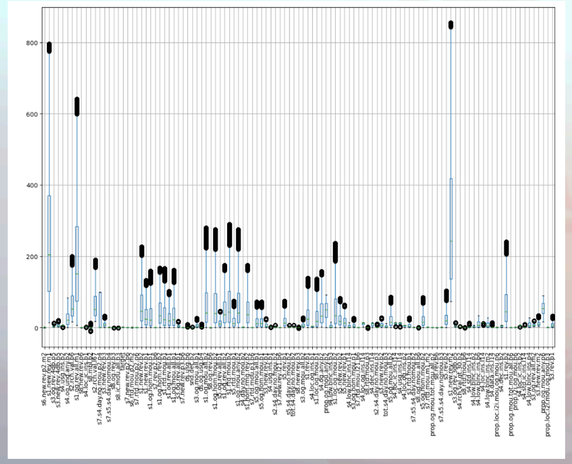
Capping and Flooring the Outliers:

Reduces the impact of extreme values while retaining the majority of data distribution.

3-Sigma Approach for Outlier Treatment:

Another method to treat outliers, ensuring data falls within a more reasonable range.





Distribution of Numerical Features:

Provides insights into the distribution and range of numerical features post-scaling.

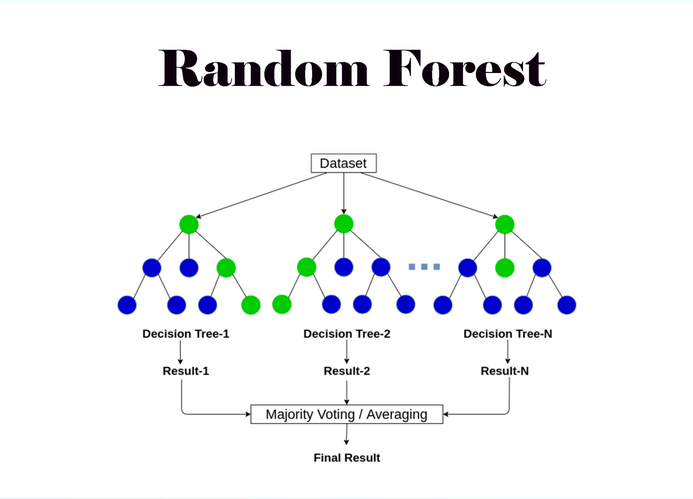
Creating a Correlation Matrix:

Identifies the strength of relationships between features, aiding in feature selection

Machine learning Model Building

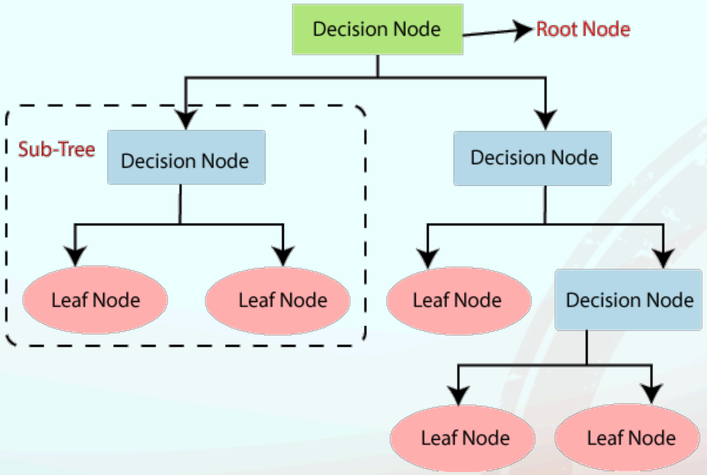
Machine learning model building involves training algorithms on processed data to predict customer churn.

Random Forest



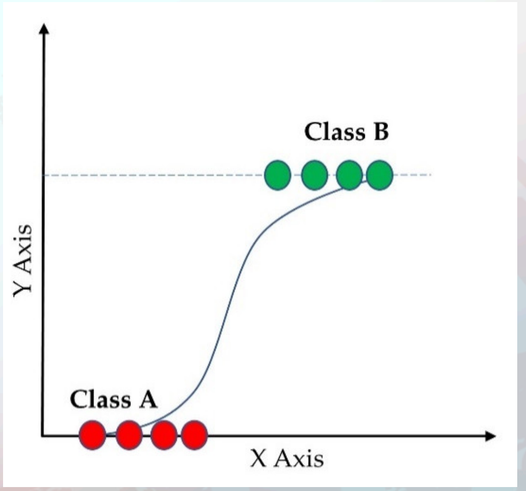
A random forest is an ensemble learning method that constructs multiple decision trees during training. It aggregates the predictions of each tree to improve accuracy and control overfitting, making it more robust and reliable compared to individual decision trees.

Decision Tree



A decision tree is a flowchart-like model used for classification and regression. It splits the data into subsets based on the value of input features, forming a tree structure where each node represents a decision rule, and each leaf node represents an outcome.

Logistic Regression



Logistic regression is a statistical model used primarily for binary classification. It predicts the probability of an outcome by fitting data to a logistic curve, and it is particularly useful for models where the dependent variable is dichotomous (e.g., yes/no, true/false).

Machine learning Model Building

Machine learning model building involves training algorithms on processed data to predict customer churn.

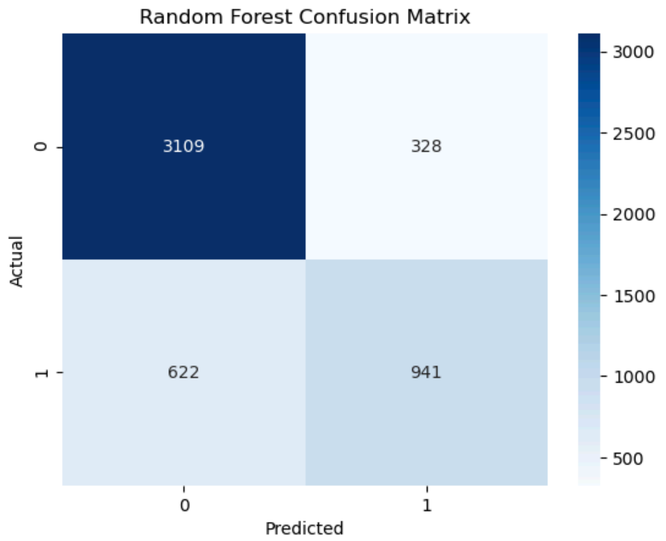
Random Forest

```
Random Forest Model Accuracy: 0.81
Random Forest Model Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.90      0.87      3437
           1       0.74      0.60      0.66      1563

    accuracy                           0.81      5000
   macro avg       0.79      0.75      0.77      5000
weighted avg       0.80      0.81      0.80      5000

Random Forest Model training and evaluation complete.
Random Forest Confusion Matrix:
```



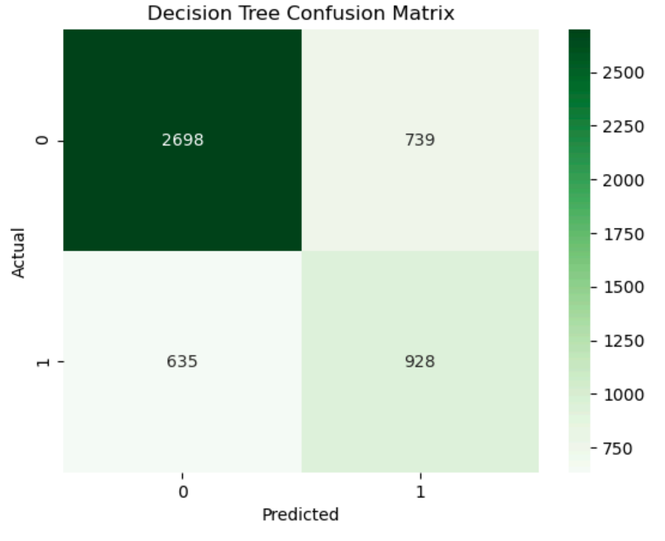
Decision Tree

```
Decision Tree Model Accuracy: 0.7252
Decision Tree Model Classification Report:
              precision    recall  f1-score   support

           0       0.81      0.78      0.80      3437
           1       0.56      0.59      0.57      1563

    accuracy                           0.73      5000
   macro avg       0.68      0.69      0.69      5000
weighted avg       0.73      0.73      0.73      5000

Decision Tree Model training and evaluation complete.
Decision Tree Confusion Matrix:
```



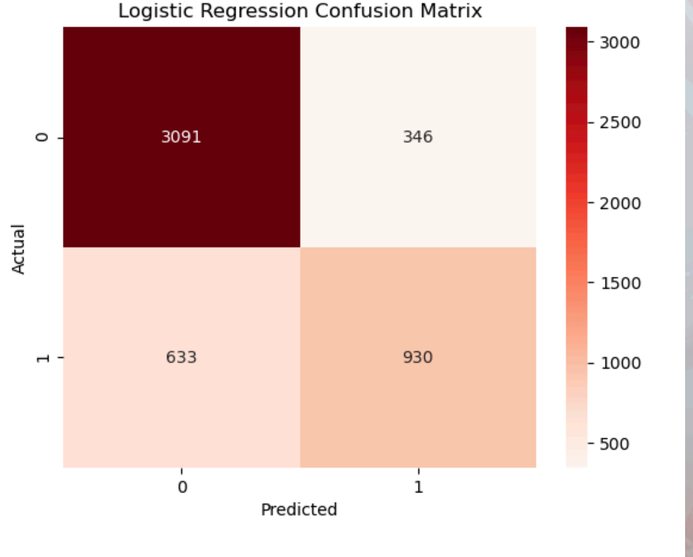
Logistic Regression

```
Logistic Regression Model Accuracy: 0.8042
Logistic Regression Model Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      3437
           1       0.73      0.60      0.66      1563

    accuracy                           0.80      5000
   macro avg       0.78      0.75      0.76      5000
weighted avg       0.80      0.80      0.80      5000

Logistic Regression Model training and evaluation complete.
Logistic Regression Confusion Matrix:
```



Hyperparamater tuning

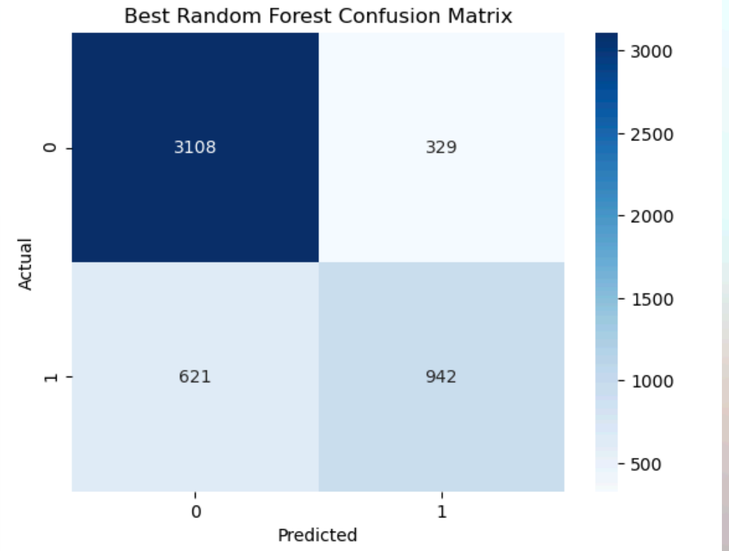
Hyperparameter tuning optimizes the performance of machine learning models by adjusting their parameters.

Random Forest

```
Hyperparameter tuning complete.
Best Model Accuracy: 0.81
Best Model Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.90      0.87      3437
           1       0.74      0.60      0.66      1563

    accuracy                           0.81      5000
   macro avg       0.79      0.75      0.77      5000
weighted avg       0.80      0.81      0.80      5000
```



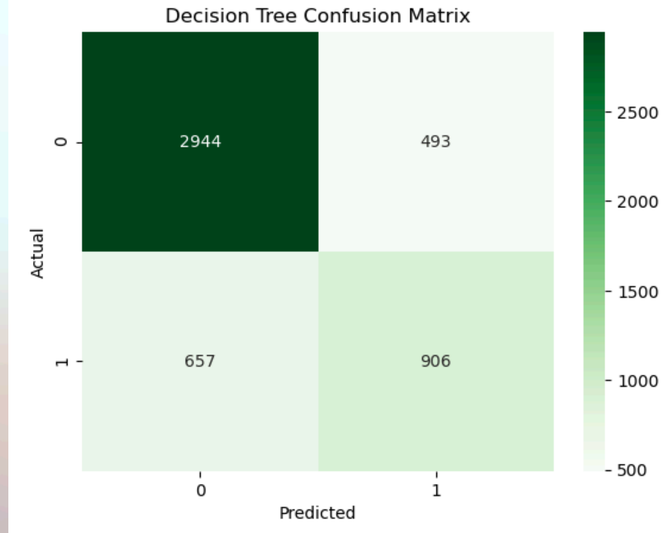
Decision Tree

```
Best Decision Tree Model Accuracy: 0.77
Best Decision Tree Model Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.86      0.84      3437
           1       0.65      0.58      0.61      1563

    accuracy                           0.77      5000
   macro avg       0.73      0.72      0.72      5000
weighted avg       0.76      0.77      0.77      5000

Decision Tree Model training and evaluation complete.
Decision Tree Confusion Matrix:
```



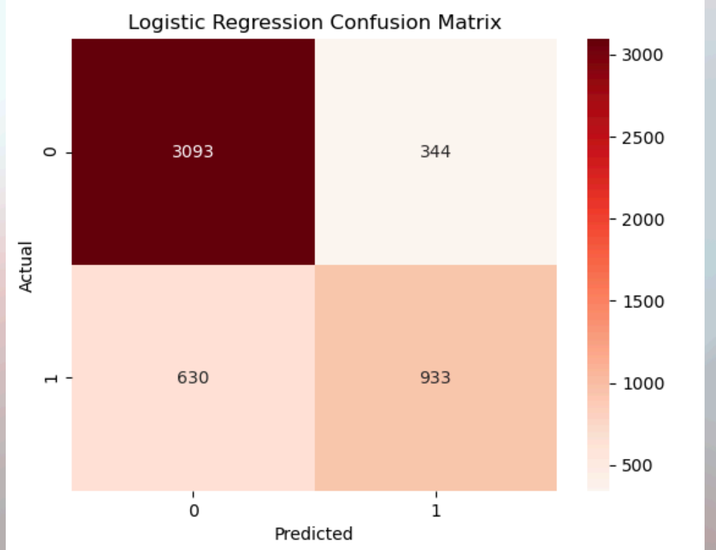
Logistic Regression

```
Best Logistic Regression Model Accuracy: 0.8052
Best Logistic Regression Model Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      3437
           1       0.73      0.60      0.66      1563

    accuracy                           0.81      5000
   macro avg       0.78      0.75      0.76      5000
weighted avg       0.80      0.81      0.80      5000

Logistic Regression Model training and evaluation complete.
Logistic Regression Confusion Matrix:
```



Overall Result

	Model	Normal Model Accuracy	Hypertuned Model Accuracy
0	Random Forest	0.81	0.81
1	Logistic Regression	0.8042	0.8052
2	Decision Tree	0.7252	0.77

Random Forest: Accuracy remains the same at 0.81 before and after hyper-tuning.
Logistic Regression: Accuracy slightly improves from 0.8042 to 0.8052 after hyper-tuning.
Decision Tree: Accuracy significantly improves from 0.7252 to 0.77 after hyper-tuning.

Random Forest should be mainly considered because:
It maintains the highest accuracy of 0.81 in both normal and hyper-tuned scenarios.
It provides robust and reliable predictions, leveraging the ensemble method to reduce overfitting and improve generalization.

RoC - AuC Curve

The ROC AUC curve is a graphical representation that illustrates the diagnostic ability of a binary classifier system by plotting the true positive rate against the false positive rate at various threshold settings.

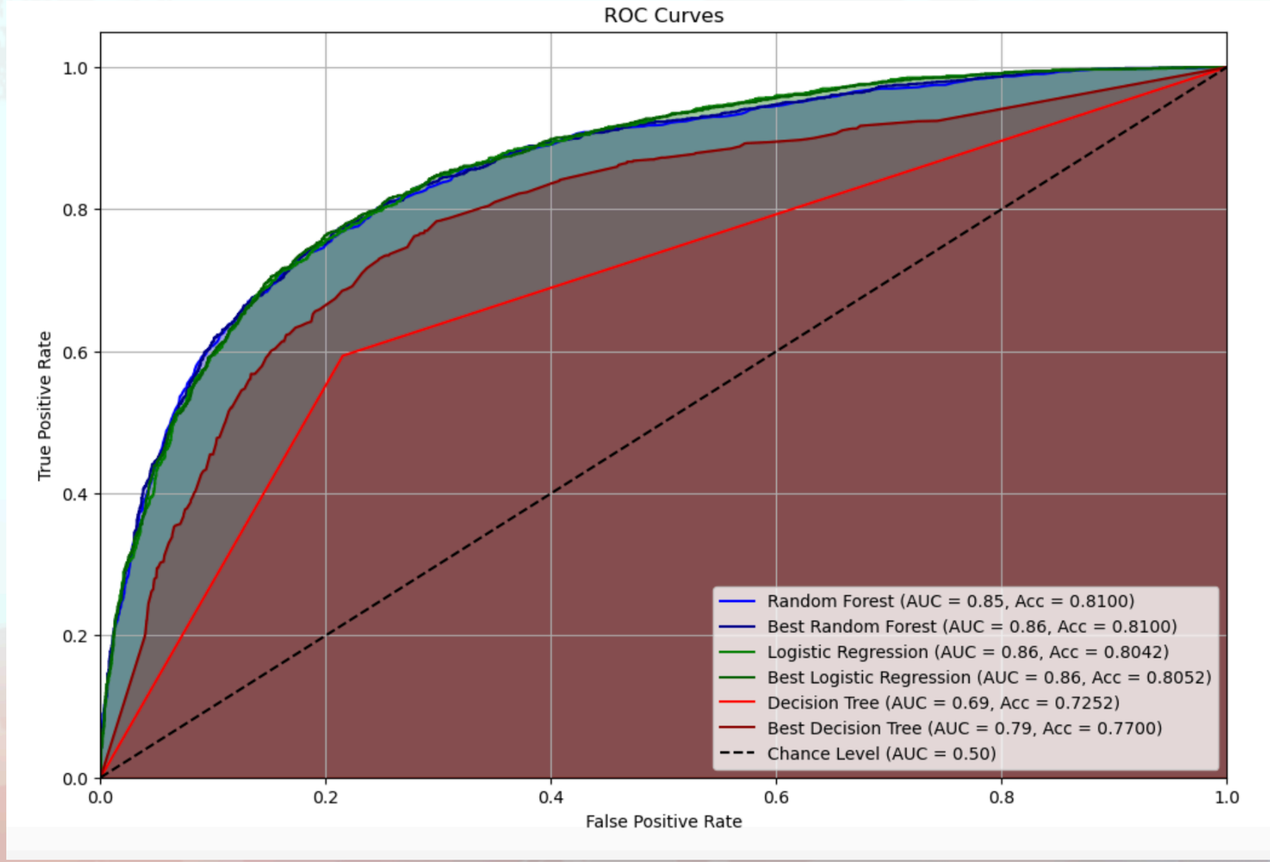
Interpretation of the Result:

The Random Forest model (both normal and best) shows a high AUC of 0.85 and 0.86 respectively, with an accuracy of 0.81, indicating strong performance.

Logistic Regression also performs well, with an AUC of 0.86 in both normal and best configurations, and a slight accuracy improvement from 0.8042 to 0.8052 after hyper-tuning.

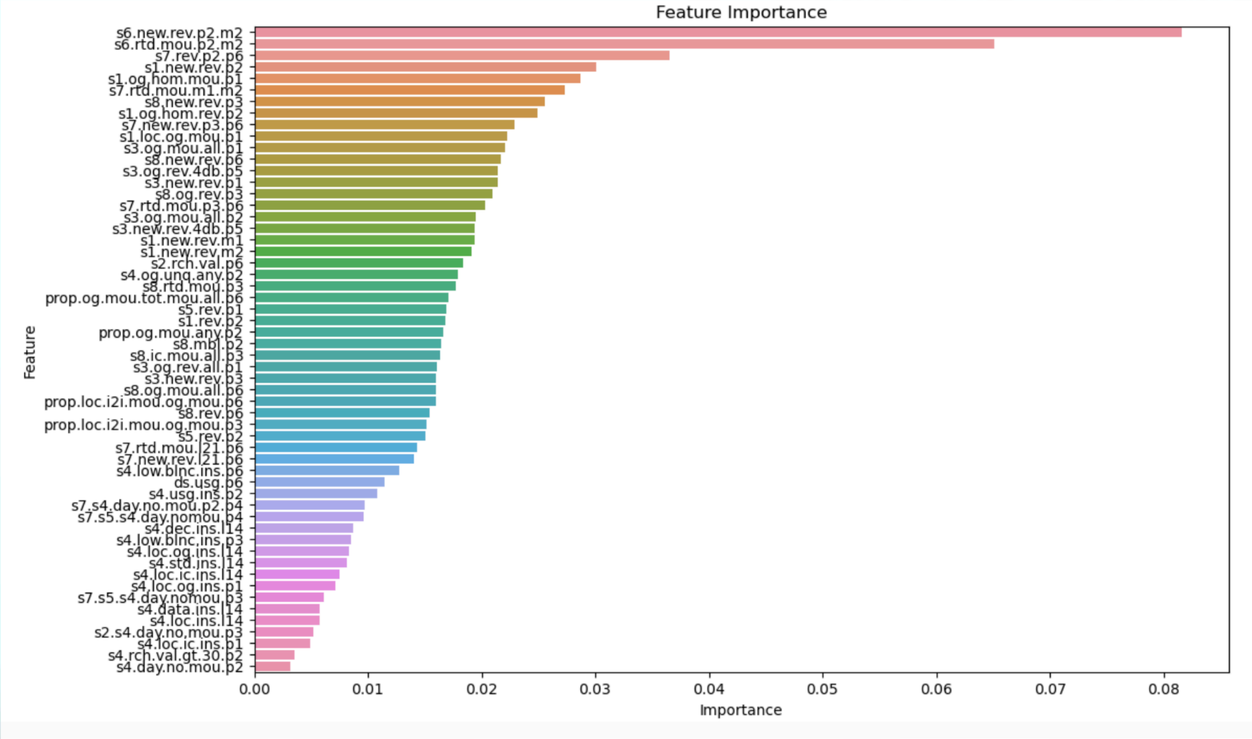
The Decision Tree model shows the most significant improvement after hyper-tuning, increasing from an AUC of 0.69 to 0.79 and accuracy from 0.7252 to 0.77.

Overall, the Random Forest and Logistic Regression models perform similarly well, with the Random Forest having a slight edge in terms of stability and reliability, as indicated by the ROC AUC curves.



Model Interpretation (Feature Importance)

Feature importance measures the contribution of each feature to the predictive power of a machine learning model.

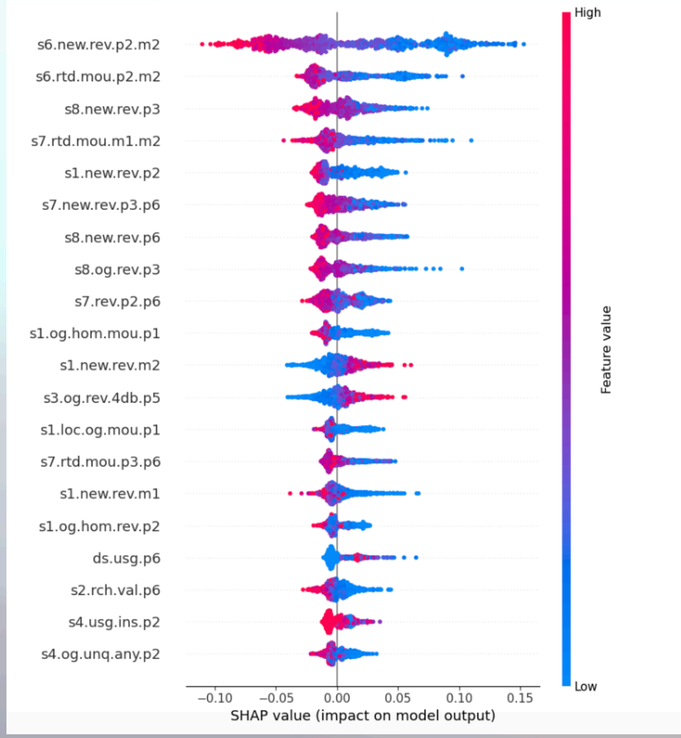


Top features such as s6.new.rev.p2.m2 and s6.rtd.mou.b2.m2 have the highest impact on the model's predictions, indicating they are critical for accurate forecasting.

Model Interpretation(SHAP & LIME)

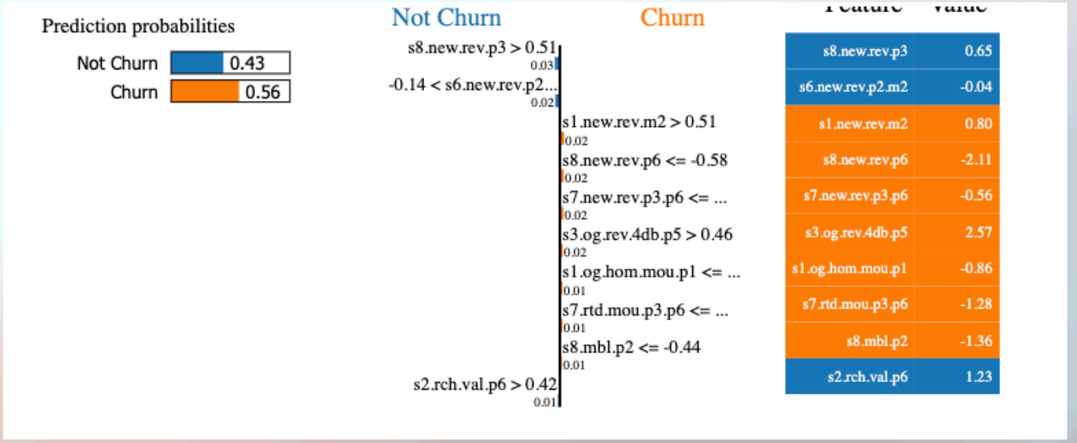
SHAP

SHAP (SHapley Additive exPlanations) quantifies the impact of each feature on model predictions, providing a clear, global and local interpretation based on cooperative game theory.



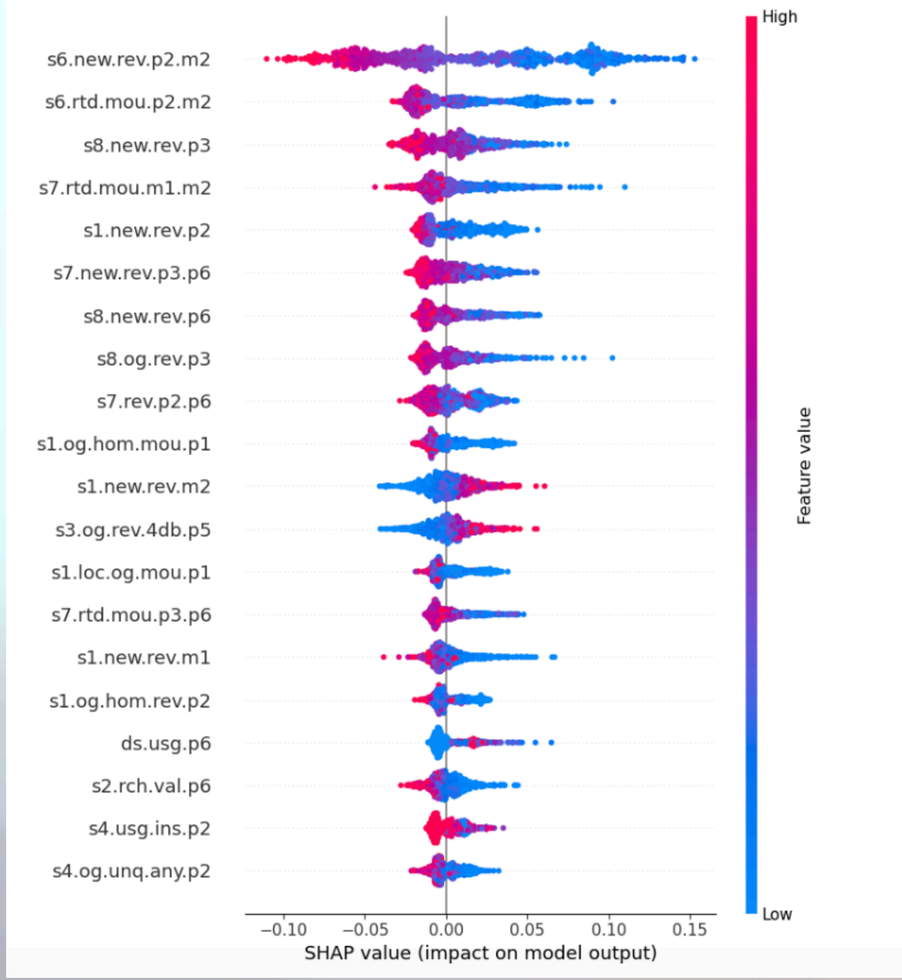
LIME

LIME (Local Interpretable Model-agnostic Explanations) explains individual model predictions by approximating the model locally with an interpretable surrogate model.



Model Interpretation(SHAP & LIME)

SHAP



This image is a SHAP (SHapley Additive exPlanations) summary plot, which visualizes the impact of various features on the output of a machine learning model.

Here is a brief interpretation:

Features Importance: The features are listed on the y-axis, with the most important features at the top. For instance, "s6.new.rev.p2.m2" and "s6.rtd.mou.p2.m2" are among the most impactful features.

SHAP Values: The x-axis represents the SHAP values, indicating the impact of each feature on the model's output. Positive SHAP values push the prediction higher, while negative values push it lower.

Feature Values: The color gradient (from blue to red) represents the feature values, where red indicates high feature values and blue indicates low feature values.

Distribution: The density of the points shows how the feature impacts the model output across different instances. For example, "s6.new.rev.p2.m2" shows a spread of impacts, with both positive and negative influences depending on the feature value.

Overall, this plot helps to understand which features are most influential and how they affect the predictions of the model.

Model Interpretation(SHAP & LIME)

LIME

This image is a partial dependency plot with prediction probabilities, and it shows the contribution of various features to the probability of "Churn" or "Not Churn" for a particular instance.

Here's a brief interpretation:

Prediction Probabilities: The model predicts a 56% probability for "Churn" and a 43% probability for "Not Churn".

Decision Path:

For "Not Churn" (blue path):

"s8.new.rev.p3 > 0.51" contributes 0.03 towards "Not Churn".

"-0.14 < s6.new.rev.p2.m2 <= 0.51" contributes 0.02 towards "Not Churn".

Other features with minor contributions.

For "Churn" (orange path):

"s1.new.rev.m2 > 0.51" contributes 0.02 towards "Churn".

"s8.new.rev.p6 <= -0.58" contributes 0.02 towards "Churn".

"s7.new.rev.p3.p6 <= -0.56" contributes 0.02 towards "Churn".

Other features with minor contributions.

Feature Values: The rightmost section lists the actual feature values for this instance:

"s8.new.rev.p3" = 0.65

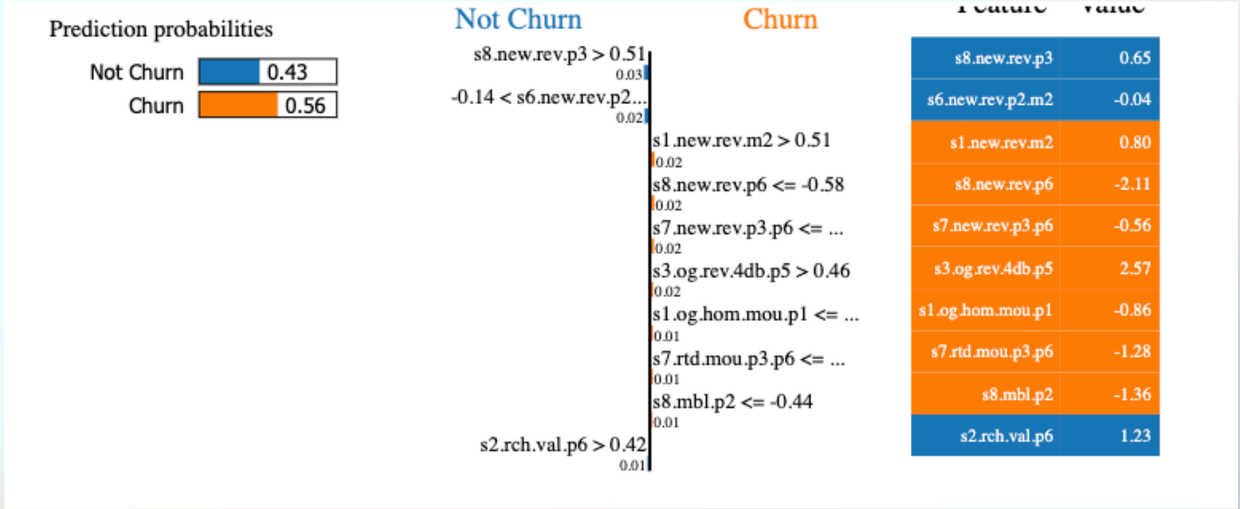
"s6.new.rev.p2.m2" = -0.04

"s1.new.rev.m2" = 0.80

"s8.new.rev.p6" = -2.11

And so on...

This plot helps understand which features are pushing the prediction towards "Churn" or "Not Churn" and by how much. In this case, the higher values of "s1.new.rev.m2" and the lower values of "s8.new.rev.p6" significantly push the prediction towards "Churn".



Through our project, we conducted a comprehensive analysis of customer churn data using a RandomForestClassifier model. By employing SHAP values for model interpretation, we identified that features such as "s1.new.rev.m2" and "s8.new.rev.p6" significantly impact the likelihood of customer churn. The RandomForest model, enhanced with feature scaling and categorical encoding, demonstrated effective performance in predicting churn. This interpretability allows us to devise targeted strategies to mitigate churn by focusing on the most influential features, ultimately enhancing customer retention efforts. Based on the results, the RandomForest model is recommended for its robustness and interpretability in this context.

Thank You