**Churn Project Report**

**Outline:**

1. **Data**
   1. Evaluation method
2. **Data Manipulation**
3. **Exploratory data analysis**
   1. \*\*Customer attrition in data
   2. \*\*Variables distribution in customer attrition
   3. ## Monthly Minutes, Monthly Revenue and Age
   4. \*\*Variable summary
   5. \*\*Correlation
4. **Data Preprocessing**
   1. ## Handling Missing Values
5. **Modeling**
   1. Baseline model with logistic regression
   2. Synthetic Minority Oversampling Technique (SMOTE)
   3. Recursive Feature Elimination
   4. Univariate Selection
   5. Decision Tree Algorithm
   6. Visualizing a decision tree from random forest
   7. Random Forest Algorithm
   8. Support Vector Machine
   9. Tuning parameters and Hyperparameters for Support Vector Machine
6. **Model Performance**
   1. Model performance metrics
   2. Comparing model metrics
   3. Confusion matrix comparison
   4. ROC-Curves for models
   5. Precision recall curves
7. **Data**

The Data used in this work is the [Telco Customer Churn](https://www.kaggle.com/blastchar/telco-customer-churn) dataset on Kaggle, the dataset was contributed for the purpose of improving customer retention. This dataset was picked because it contains unique features and data that organizations should look into collecting in order to learn well about their customer attrition and improve customer retention.

This data and feature overview is contained in the [churn.html](file:///C:\Users\ajayi\Documents\Nocean\Churn\churn.html) file. We finally read the data in into a pandas dataframe. The pandas dataframe helps us to perform various data manipulation.

1. **Data Manipulation**

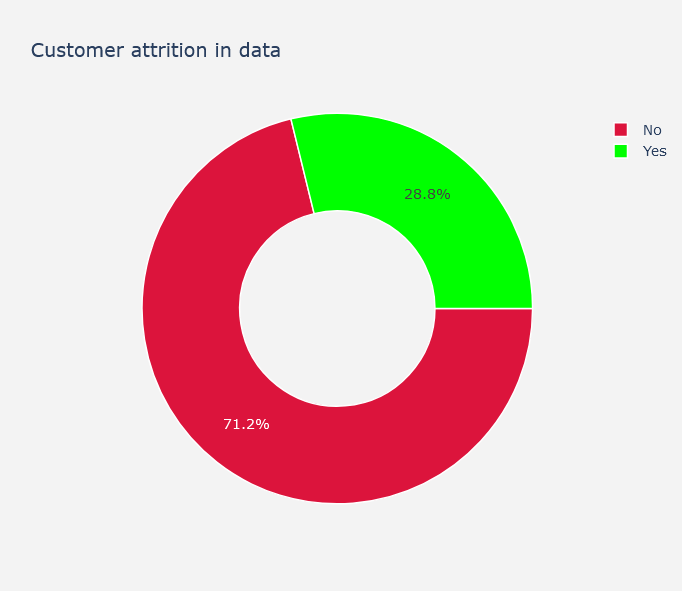
Several operations are performed on the dataset in this stage. Operations like,

* Replacing spaces in the feature names and correctly formatting into what can be consumed in development.
* Dropping null values (values not available). Even though our dataset contains 0% missing values, some null values were created during the replacement step mentioned above.
* All numerical features preprocessed above are converted into a float data type.
* There are certain features that contains “No internet service” as a unique value. Columns like 'OnlineSecurity', 'OnlineBackup', 'DeviceProtection', 'TechSupport','StreamingTV', 'StreamingMovies', we replace this unique value with “No”, a shorter alternative.
* In the churn.html file, we will observe that most features contain Boolean in the form of [0 and 1] as unique values, we replace this to [No and Yes].
* The “tenure” feature is a numerical feature, it shows the number of months the customer has stayed with the company. We convert this feature to a categorical feature by binning the values in groups.
* We separate customers that churn and customers that did not churn into different data frames.
* Finally, we separate the categorical and numerical features. Categorical features are features that are discrete; features having limited unique values, while numerical features are the continuous ones; having values between 0 and infinity.

1. **Exploratory Data Analysis**
   1. **Customer attrition in data**

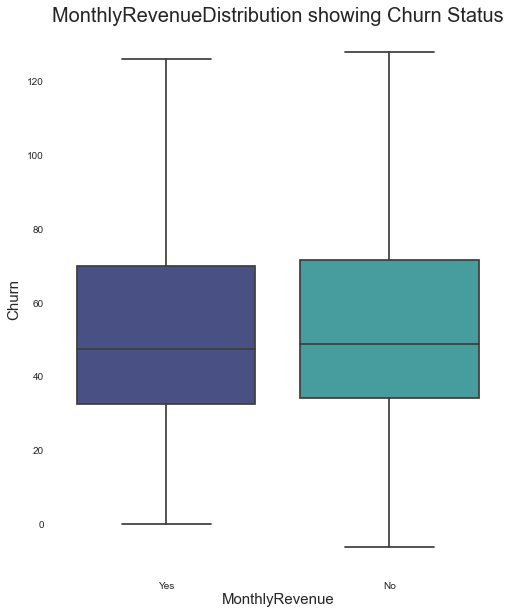
The customer churn prediction problem is one with an imbalanced dataset. An imbalanced dataset Is one that has a higher percentage of one of the class labels we are trying to predict than the other class label.

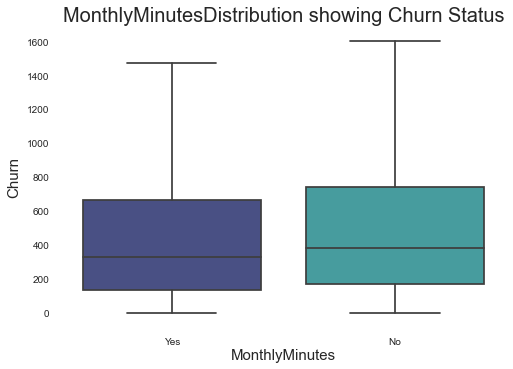
The image below shows the imbalance in our dataset class labels.

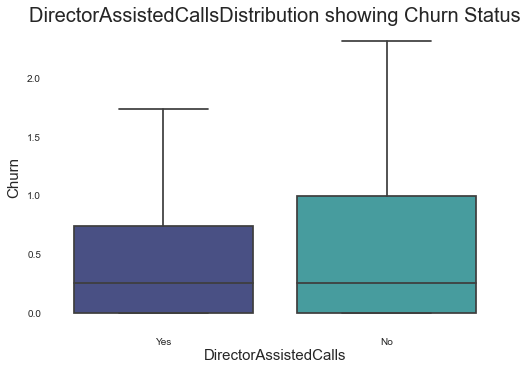


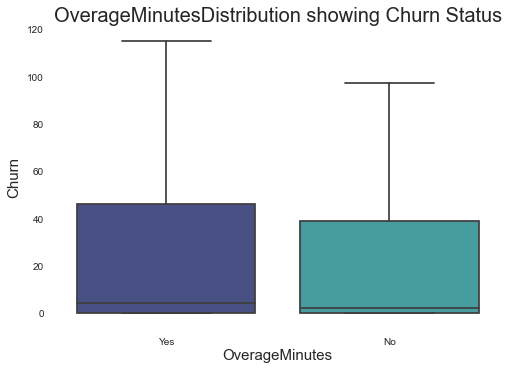
* 1. **Variable Distribution**

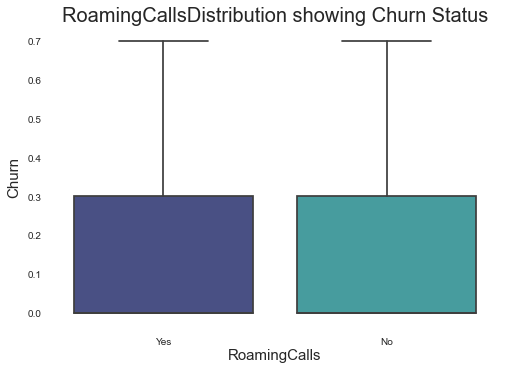
Here we also show the distribution in our dataset, visualizing all features one at a time with an interactive plotter. Few images are shown here, while the full visualizations are in our codes

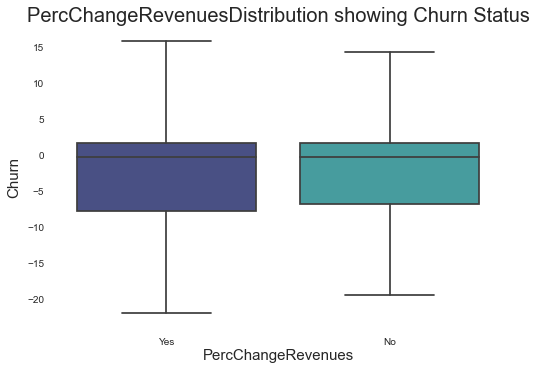


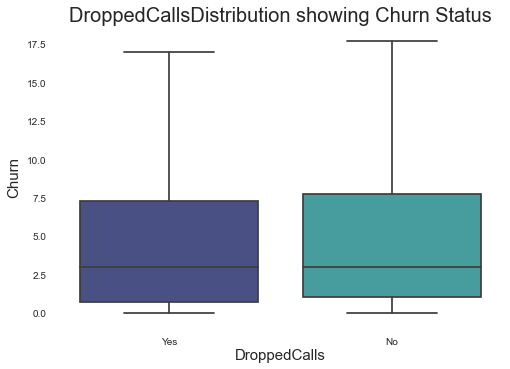


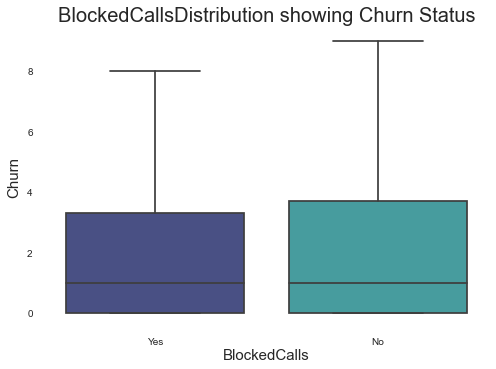


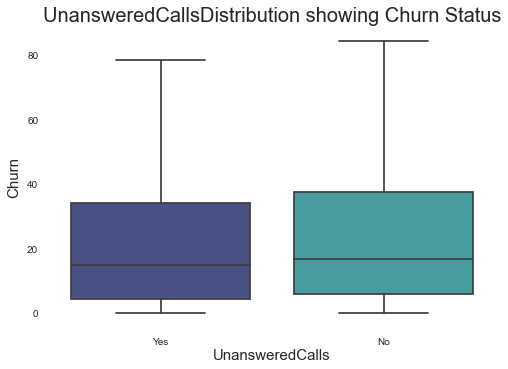
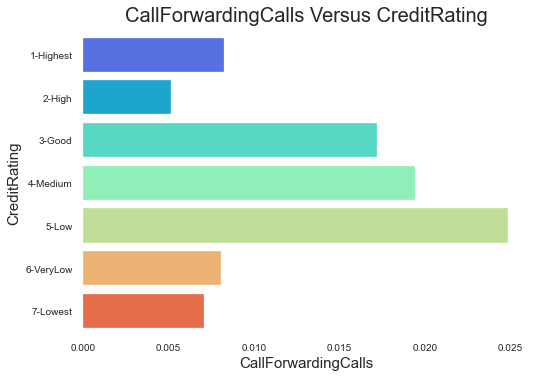
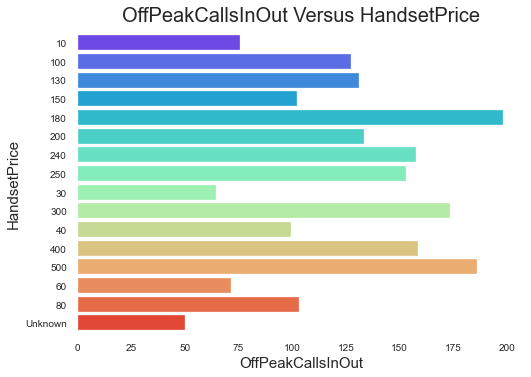
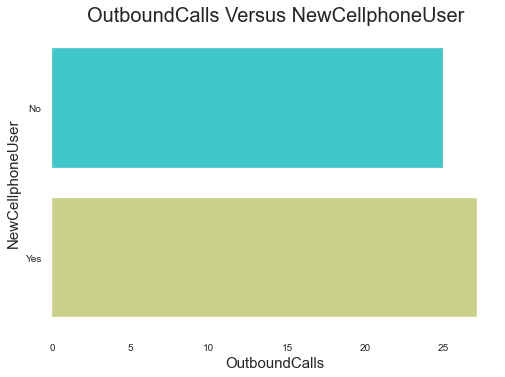
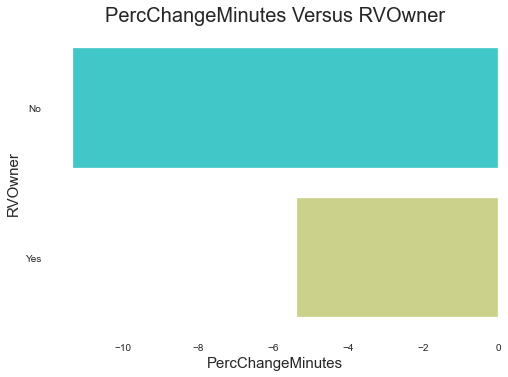
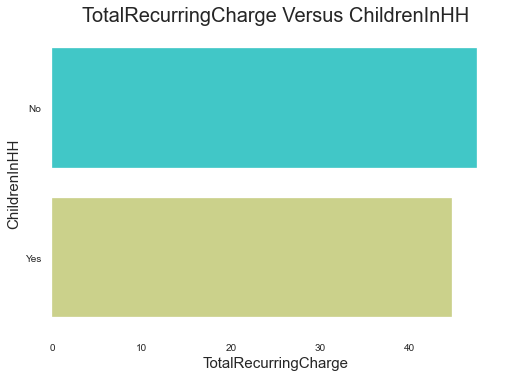
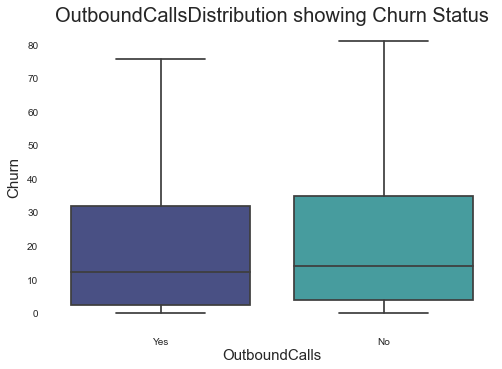
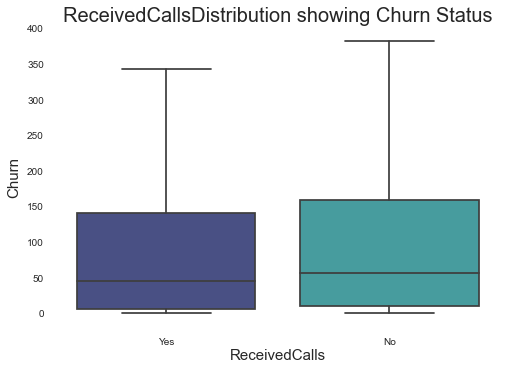
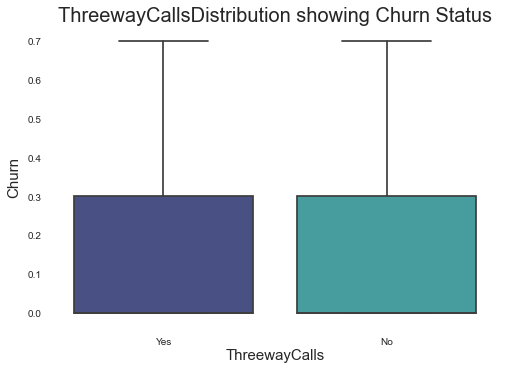
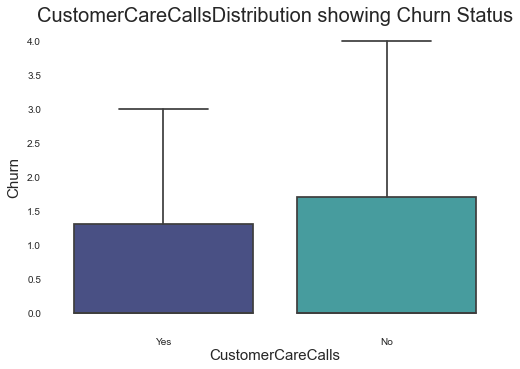






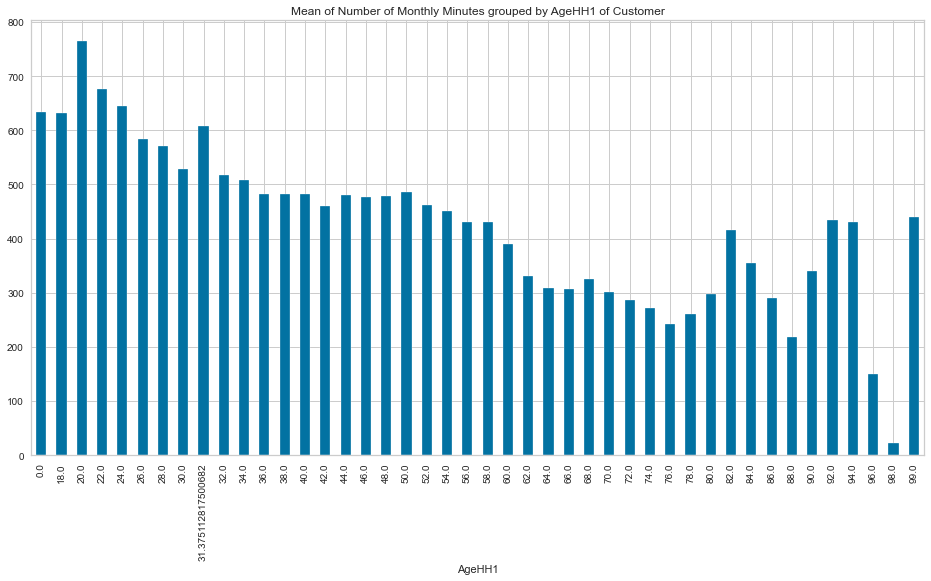


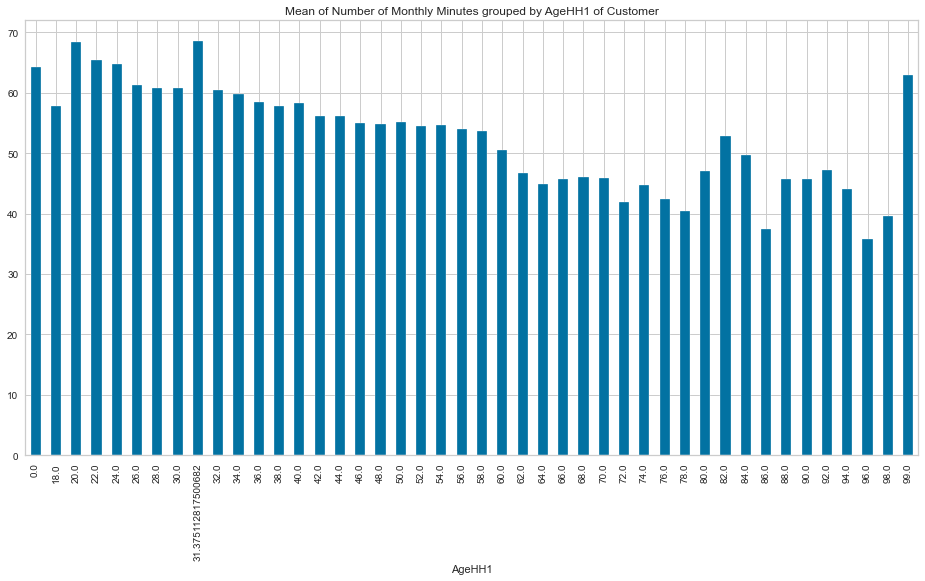




* 1. **Monthly Minutes, Monthly Revenue and Age**

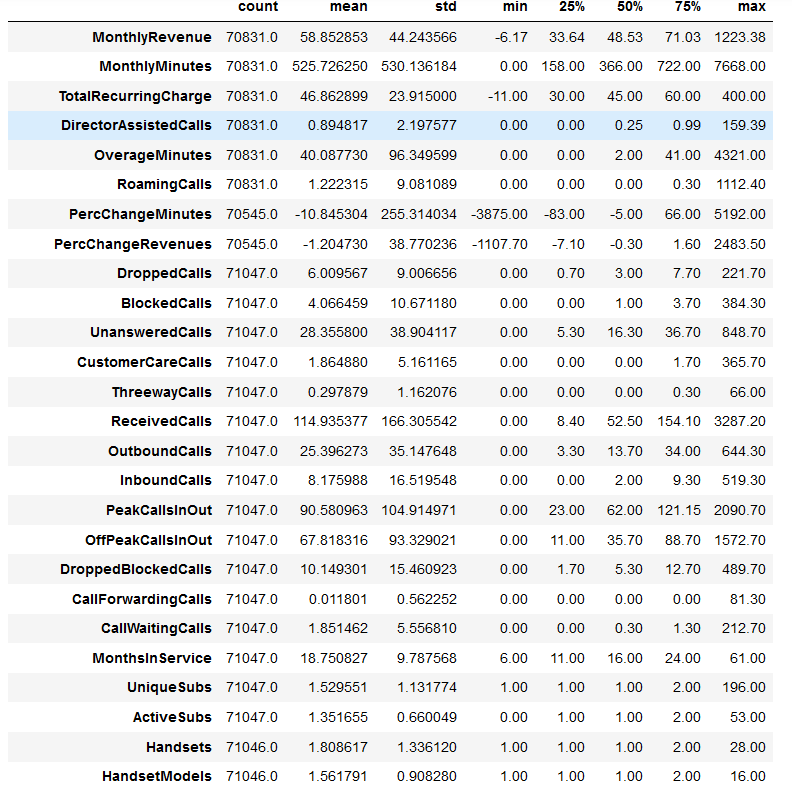
Here also, we compare the monthly Minutes and Age.

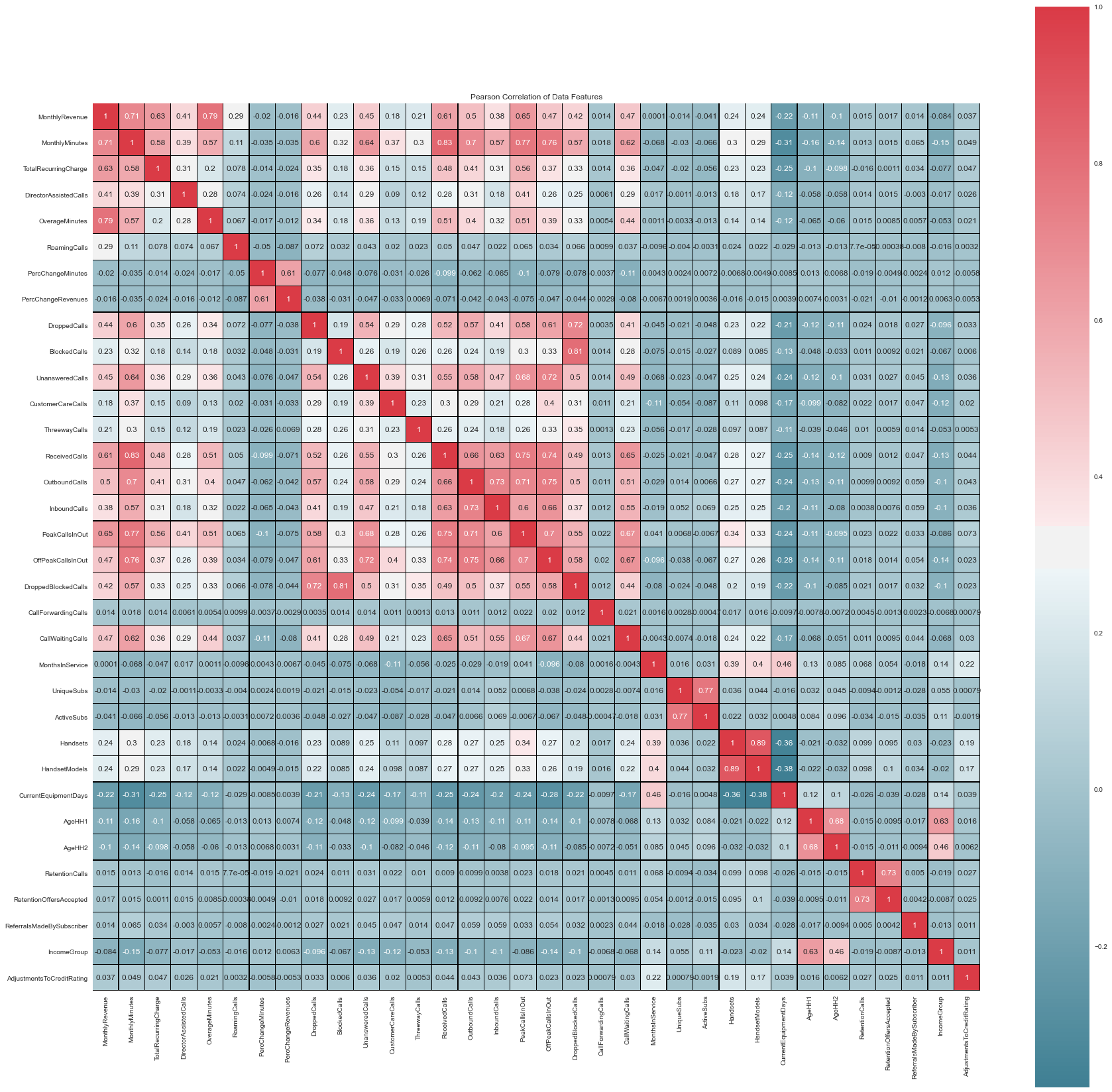




* 1. **Variable Summary and correlation matrix**

Here, we give an overview and a descriptive statistic of our dataset, and we examine the correlation between each feature. Correlation values are within 0 and 1, 0 for perfectly not correlated, and 1 for perfectly correlated

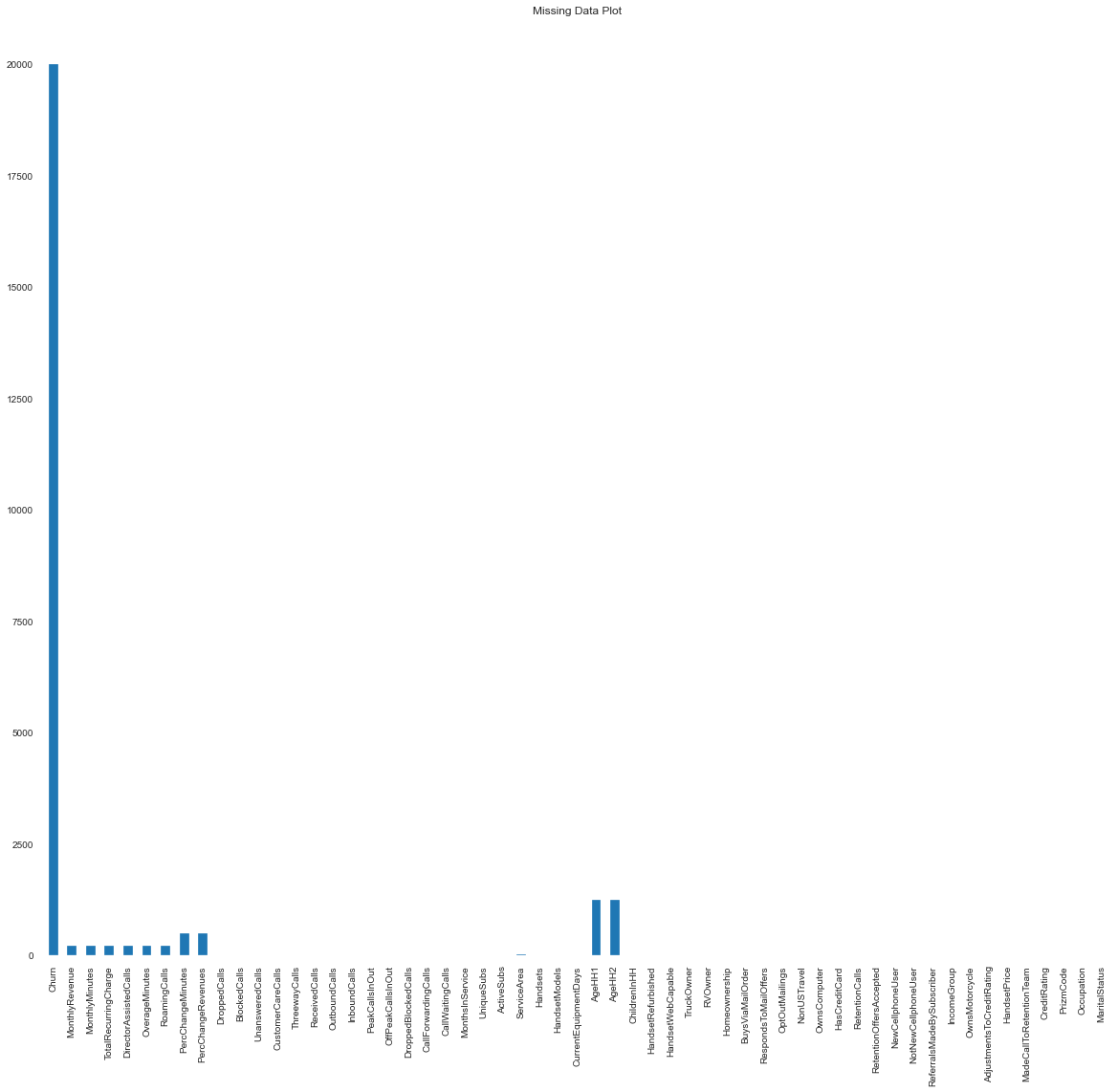
****

****

1. **Data Preprocessing**
   1. **Handling Missing Values**

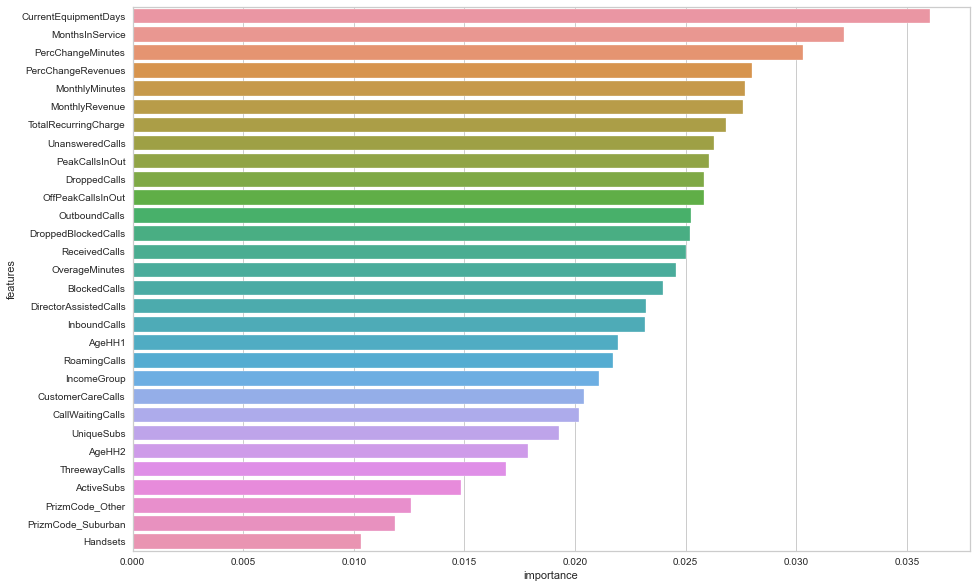
The Plot below shows the missing values in our dataset. The target class, Churn has the most missing values, this is because the training data were merged this the test data, to solve this, we have isolated the train data with labels not missing, and continue with our preprocessing on the training data. In this dataset, numerical columns in the plot contains the missing the values except for the ServiceArea feature, which we dropped because of its non-uniqueness.

We treat all the missing vales the same way by filling them with the mean of their respective columns. This decision was made after carefully looking at their descriptive statistics.



* 1. **Feature Import with Extratreeclassifier(XTree)**

ExtraTreeClassifier is an ensemble model just like RandomForest, because our data has 57 features, there is need to reduce the number of features so that we can avoid the curse of dimensionality. Here, we fit our preprocessed data to the XTree classifier and we extract the best performing models from the algorithm’s inbuilt feature importance. Below is the plot of import features ranked according to how important they are



1. **Modeling**

At this stage, we already have our data preprocessed and ready to be feed to the machine learning algorithms. We compare 4 machine learning algorithms in this work, namely: Logistic regression, Support vector machine, decision tree and Random forest algorithms.

**metrics**

It is important that we understand the kind of metrics used in judging the performance of algorithm on imbalance data.

Let me explain, most machine learning task are judge based on accuracy. Accuracy only measures how well an algorithm predicts, irrespective of the class label it is predicting. Say we have a dataset of 100 rows, with class labels in the ratio of 90(Yes):10(No). If we train with this dataset, our algorithm will achieve an accuracy of 90%. This is so because the algorithm as only learned to classify only one class. So, when we predict with this algorithm, it will only predict Yes, even for examples that are No. This is because the algorithm sees the No class as noise in the dataset and found a way to neglect them.

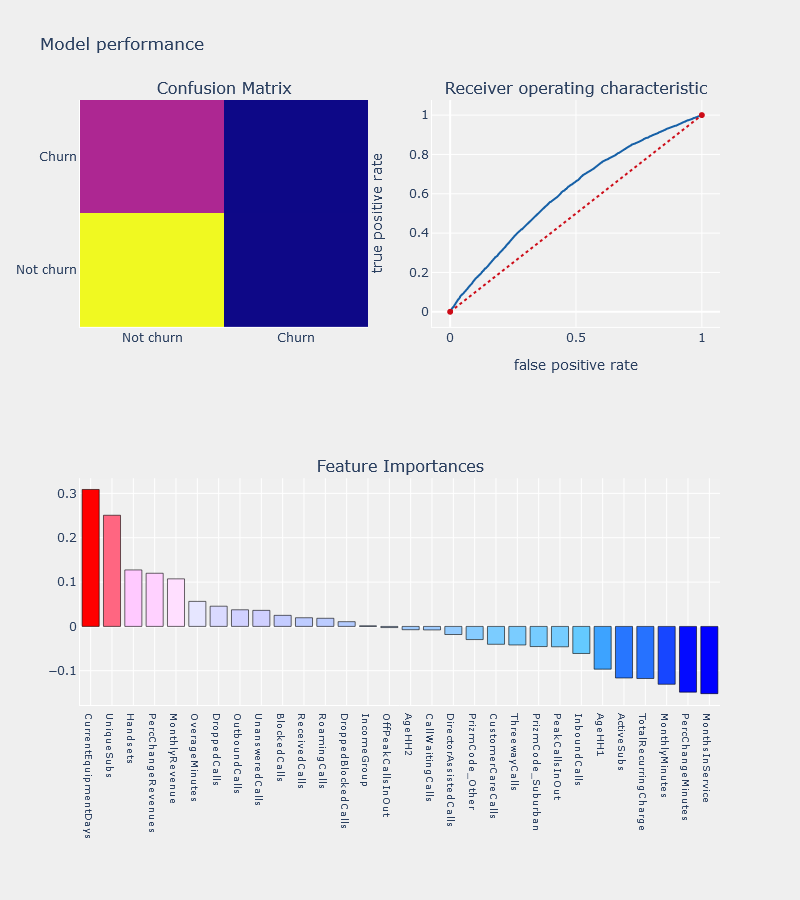
Hence, **accuracy** is not a good metric for judging imbalance problems.

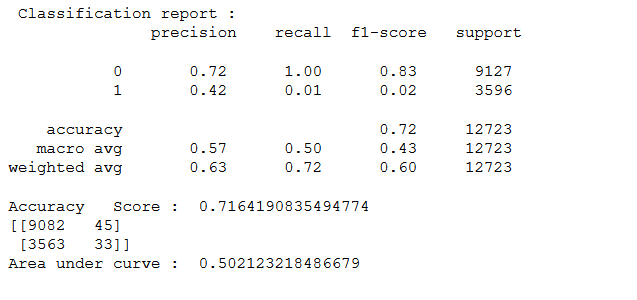
The metrics used in judging this problem are

* Confusion Matrix
* F1 score (we focus on this)
* Precision recall curve
* And ROC (receiver operating characteristic curve) curve (we also focus on this)
  1. **Baseline model with logistic regression and it feature importance**

Before applying sampling technique, we apply the logistic regression algorithm on the dataset as baseline for all our future modeling.’

Report from fitting the model is below. The feature importance is also attached, and it is calculated based on the weight of the feature

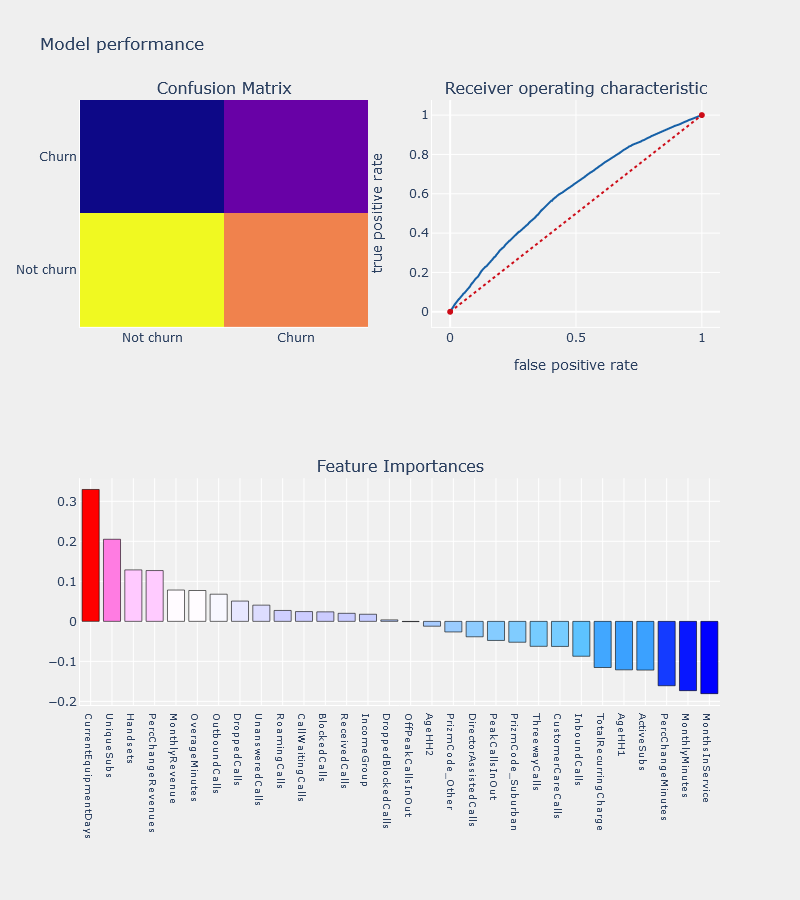


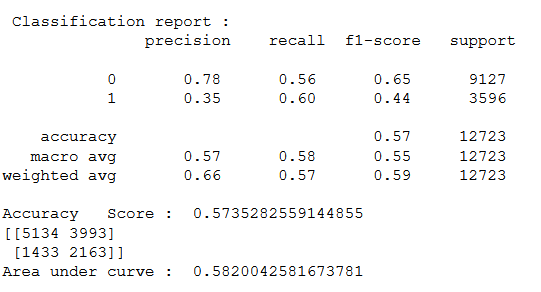
**Numerical report:**

* 1. **Synthetic Minority Oversampling technique (SMOTE)**

SMOTE is a technique for up sampling our dataset. It generates synthetic examples of the minority classes to balance up with the majority classes.

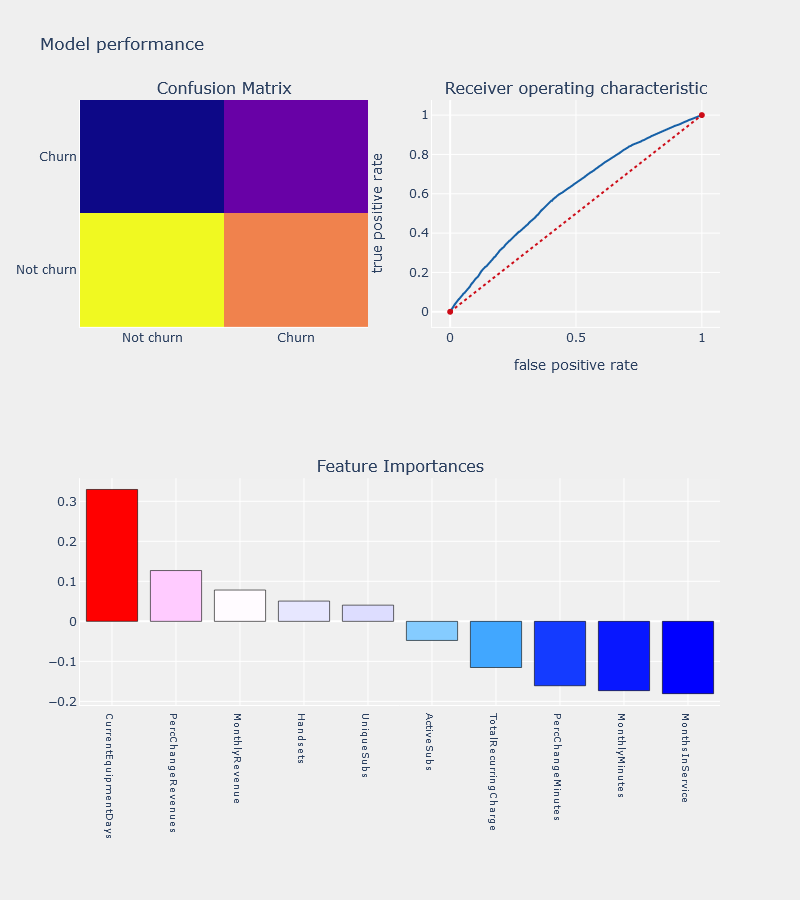
We balance our dataset with smote and apply our logistic regression on the balanced dataset this time. The results are below. We observed a better F1 score and ROC after balancing our dataset.

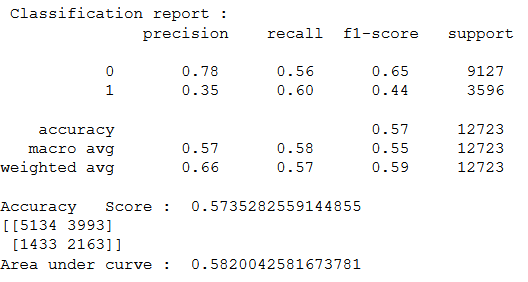




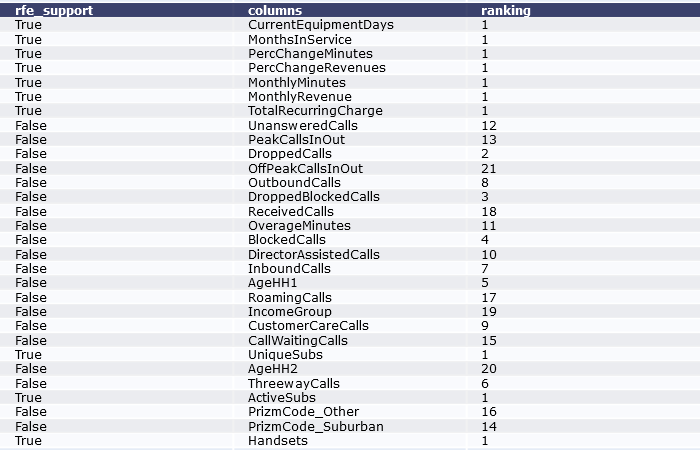
* 1. **Recursive Feature Elimination**

Recursive Feature Elimination (RFE) is based on the idea to repeatedly construct a model and choose either the best or worst performing feature, setting the feature aside and then repeating the process with the rest of the features. This process is applied until all features in the dataset are exhausted. The goal of RFE is to select features by recursively considering smaller and smaller sets of features.

We also test this on the logistic regression algorithm and the result is below.

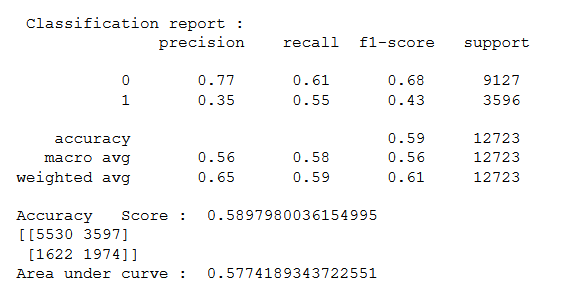


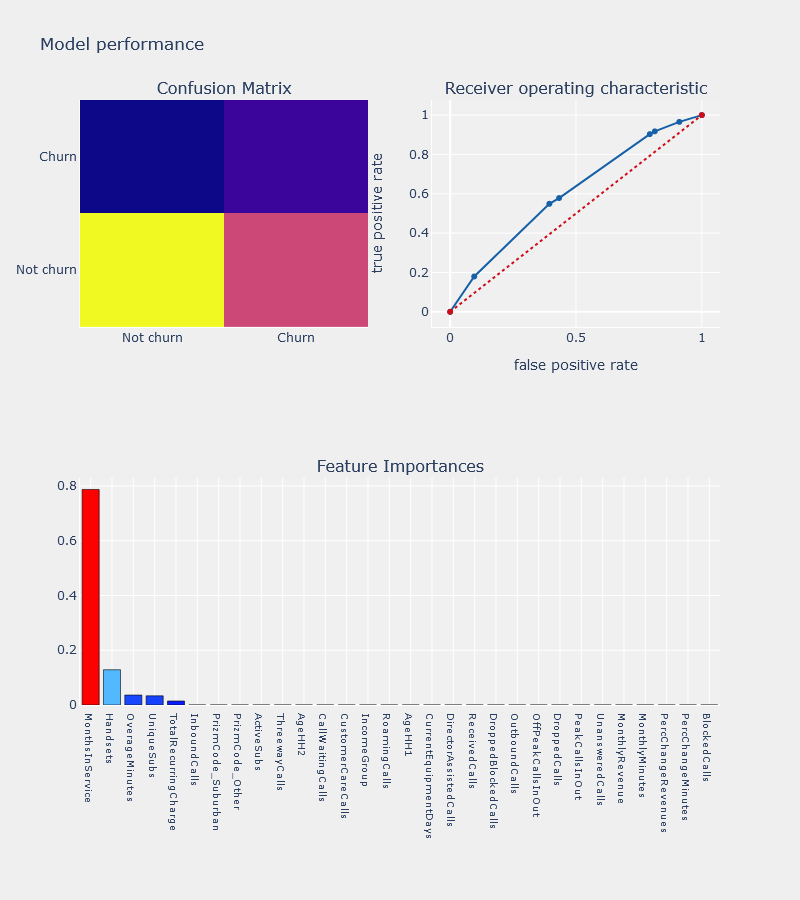
The result of the RFE is the same with that of vanilla smote. The RFE ranking is below.



* 1. **Decision Tree algorithm**

Our Decision tree algorithm is applied on the smote dataset





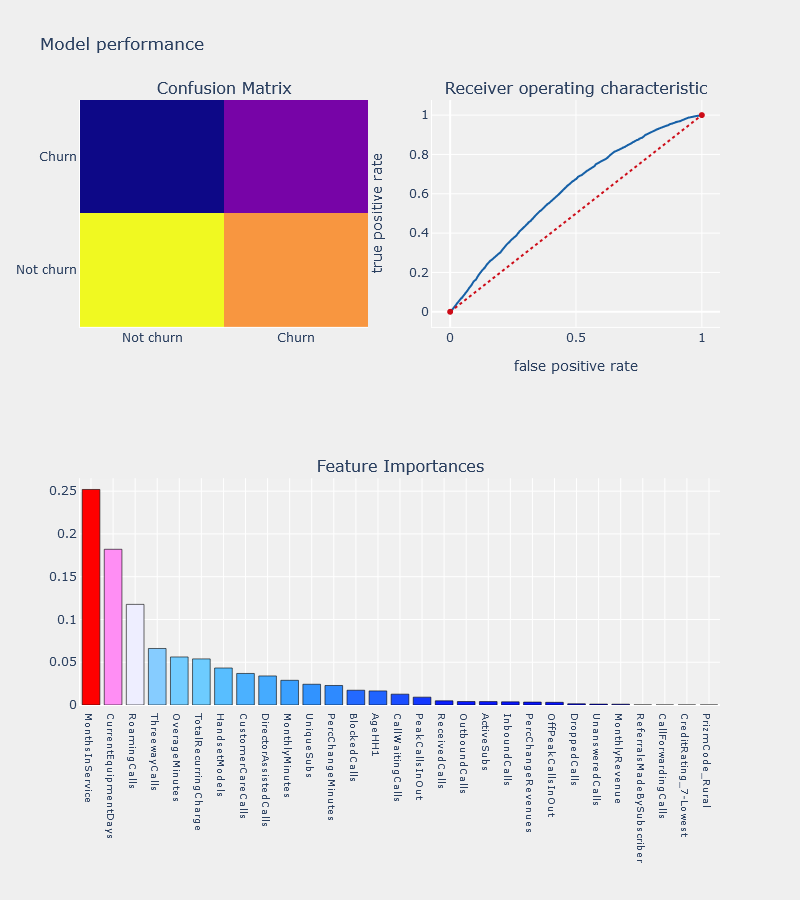
**Changes has not been made here.**

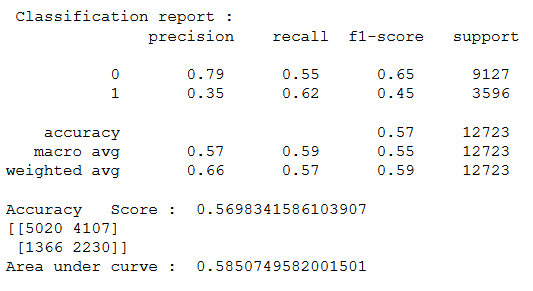
* 1. **Random Forest**

A random forest is a meta estimator that fits several decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size, but the samples are drawn with replacement.

Result is below.

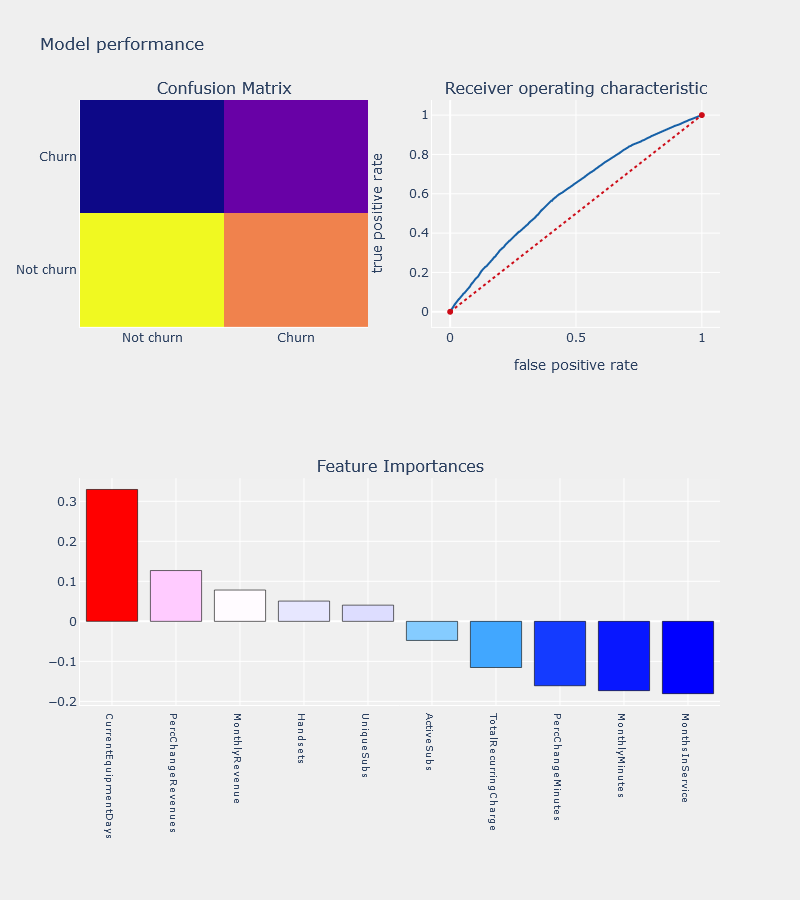
The fitted tree visualizations are displayed in the project notebook.

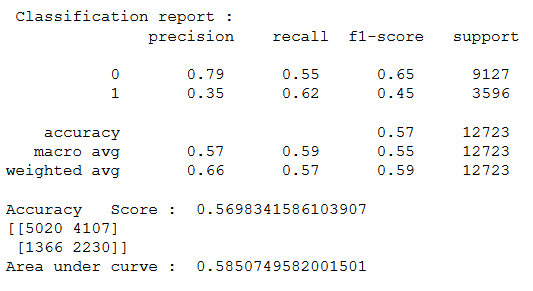




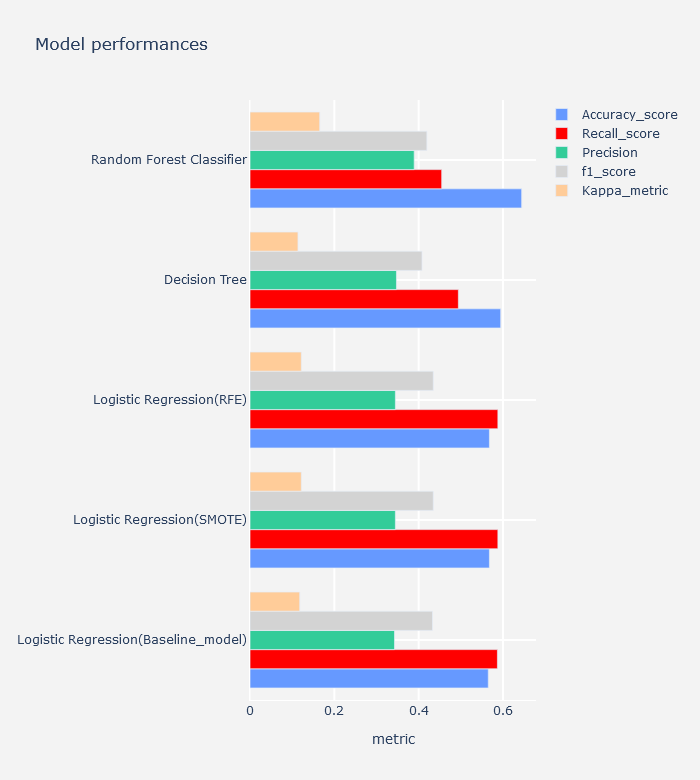
* 1. **Support Vector Machine**

Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space. Where n is number of features you have) with the value of each feature being the value of a coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes.





1. **Model Performance**

We compare model performance using visualizations

