# ITNPBD6 Machine Learning

# Assignment Spring 2022

## Contents

Abstract -------------------------------------------------------------------------------------------------------------------- 2

Business Understanding ------------------------------------------------------------------------------------------------- 2

Data Summary------------------------------------------------------------------------------------------------------------- 3

Data Preparation --------------------------------------------------------------------------------------------------------- 7

Modelling ----------------------------------------------------------------------------------------------------------------- 16

Evaluation ---------------------------------------------------------------------------------------------------------------- 19

Conclusion --------------------------------------------------------------------------------------------------------------- 21

## Abstract

This report provides a machine learning solution to predict if a customer will subscribe to StirCom’s 5G network. The goal was to develop a classification model to predict which customer is likely to subscribe to the service based on parameters provided.

The dataset provided contained 50662 rows and 20 columns including the label column. The label showed class imbalance with ‘no’ classes controlling 80.46% of the total observations and ‘yes’ having 19.54%. The data was then profiled with pandas\_profiling to gain insights into each column.

Outliers were detected in the data and were treated using the .clip data frame property of keep all values between the 5th percentile and 95th percentile. The data was further split into training, testing and validation using the train\_test\_split property of the scikit-learn library. PCA was used to reduce the dimensions and 15 variables were chosen based on the output of the scree diagram, Extra Trees Classifier and Random Forest.

Extra Trees Classifier, Logistic Regression and Neural network with TensorFlow were experimented. In the end, Extra Tress Classifier emerged the algorithm with the best accuracy, precision and recall with an accuracy of 87.24% followed by neural network and Logistic Regression respectively.

## Business Understanding

StirCom is a telecommunication company with clients in the United Kingdom. Recently, the company enrolled a 5G network and wants customers to subscribe to the network. To get customers to sign-on to the service, the company embarked on a marketing campaign to encourage existing landline customers to take out mobile contracts. StirCom over the years has collected data of over fifty thousand customers and their responses to earlier marketing campaigns. The response is either a ‘yes’ or a ‘no’.

The goal at hand is to predict as accurately as possible whether a customer will respond positively or negatively to a marketing call which presents a classification problem in machine learning. Personal identifiable information (PII) is of major concern when sharing data, but the data provided does not breach sharing of any PIIs hence raises no ethical concerns.

The data occupies 7.7 MB memory space, has no duplicated rows, and no missing values. In addition, the data contained twenty columns and 50662 observations. Out of the twenty variables, seven of them were numeric data type, nine were categorical and four were Boolean and more importantly there is a class imbalance. The classes of ‘no’ is 40763 representing 80.46 percent and ‘yes’ class is 9899 representing 19.54 percent of total observations.

## Data Summary

The data summary process was conducted using python and python libraries and jupyter notebook. The dataset was uploaded into the notebook using the libraries. Pandas data profiling was used to profile the data. The table below details each column and its schema.

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Data type | Consideration in final model | Justification |
| ID | Numeric; nominal | No | At this stage, ID does not give any valuable insight in predicting the label and will not be of use in building the final model |
| town | categorical | Yes | Because town has high cardinality judging from the data profiling |
| country | categorical | No | the variable at this stage will not considered because 99.99 percent are UK as shown in fig 1. |
| age | Numeric; continuous | Yes | This variable is positively skewed (as shown in fig 2) with a mean age of approximately 41 years, a maximum of 95 and a minimum of 18 years. Age will be useful in building the model |
| job | Categorical; Nominal | Yes | A person’s job can greatly affect his or her decision to subscribe to the service. |
| married | Categorical; Nominal | Yes | With 3 distinct values, a customer’s marital status could influence his or her decision to subscribe. |
| education | Categorical; Nominal | Yes | A person’s educational level is likely to tell their reception to technology which is why at this age it is likely to be considered in building the final model |
| arrears | Categorical; Boolean / Nominal | Yes | At this stage it is skeptical because more than 98 percent of the records is ‘False’ hence the algorithm will work in favor of ‘False’ since it carries the majority. |
| current\_balance | Numeric; Continuous | Yes | Most balances are zero with few outliers. This could be useful in the final model |
| housing | Categorical; Boolean / Nominal | Yes | Based on the distribution (fig 3), housing could be a feature in the final model. |
| has\_tv\_package | Categorical; Boolean / Nominal | Yes | A person having an additional tv package could influence his or her decision to subscribe to the service |
| Last\_contact | Categorical; Nominal | Yes | Due to its high correlation, it could be a variable in building the model |
| conn\_tr | Numeric; nominal | No | This is related to a customer’s ID and could not be a feature in the final model. |
| last\_contact\_this\_campaign\_day | Numeric; continuous | No | Being a calendar day, it would give no details as to its importance and will not be a good feature. |
| last\_contact\_this\_campaign\_month | Categorical; Nominal | No | Being a calendar month, it would give no details as to its importance and will not be a good feature. |
| this\_campaign | Numeric; continuous | Yes | Number of times a customer has been aware of the service could possibly influence the decision to subscribe to the service. |
| days\_since\_last\_contact\_previous\_campaign | Numeric; nominal | No | At this stage, judging from the distribution (fig 4), the variable does not display any useful information. |
| contacted\_during\_previous\_campaign | Numeric; nominal | Yes | This could influence a prediction of the label depending on how recent the customer was reached. |
| outcome\_previous\_campaign | Categorical; Nominal | Yes | The success or failure of a previous campaign can be of great influence. |
| new\_contract\_this\_campaign | Categorical; Nominal | Yes | This is the label we want to predict |

fig 1. Count plot of country fig 2: Age distribution

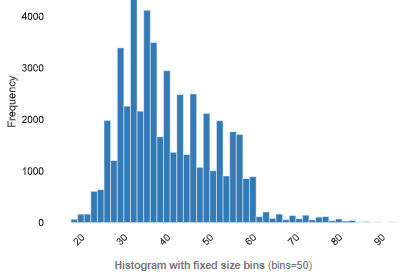
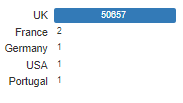
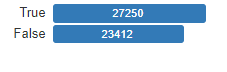
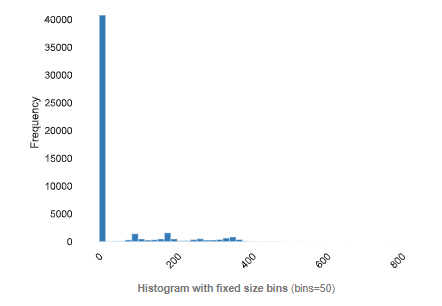


fig 3: housing fig 4: days\_since\_last\_contact\_previous\_campaign



## **Data Preparation**

This stage of the machine learning process was to visualize the data and model the data with libraries available in python. This stage involved two steps: the state of the data before pre-processing and the state of the data after pre-processing.

**Before Data processing**

**Summary Statistics**

* The data has 50662 rows and 20 columns
* The data has no duplicate rows
* There no null values present

## **Label**

The label in the data is the new\_contract\_this\_campaign column which indicates whether a client has taken out a new contract. A bar plot was used to visualize the column which indicated a class imbalance. In all, the class count for ‘yes’ was 9899 representing 19.54% and class count for ‘no’ was 40763 representing 80.46%.

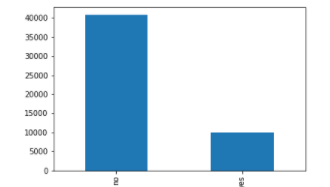
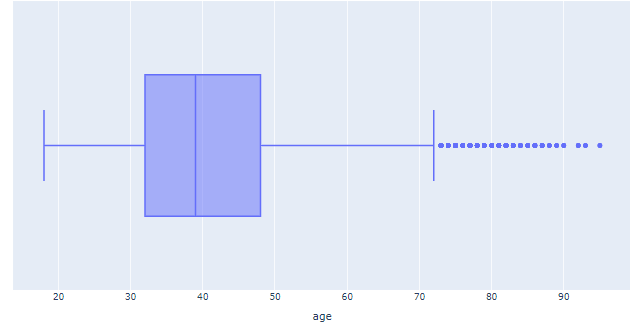


Fig 5: *A bar plot of the label classes. This represents a class imbalance.*

## **Outliers**

From the preliminary exploratory data analysis, five of the numerical variables had outliers. The outliers were detected by using the box plot from the Plotly library. Below are visualizations of columns with outliers. By default, the box and whisker plot, in this case the box plot considers values that fall between the 25th and the 75th percentile and any values outside these boundaries are considered outliers.

 fig 6: Box plot of age column fig 7: Box plot of current\_balance column

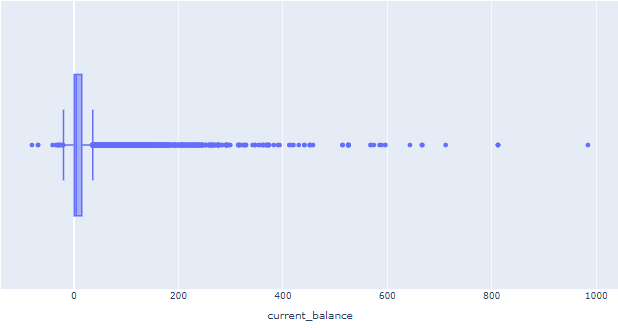


fig 8: Box plot of this\_campaign column fig 9: Box plot of days\_since\_last\_contact\_previous\_campaign column

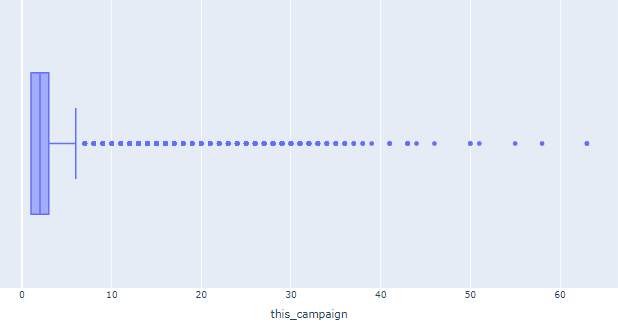
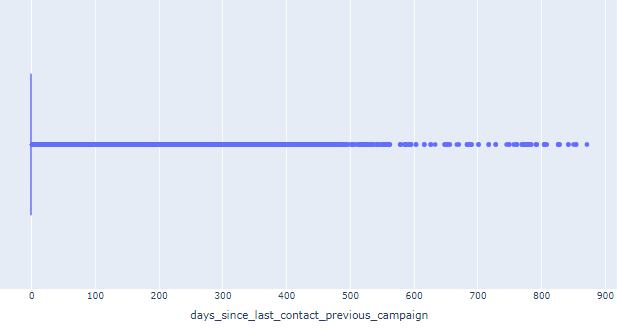
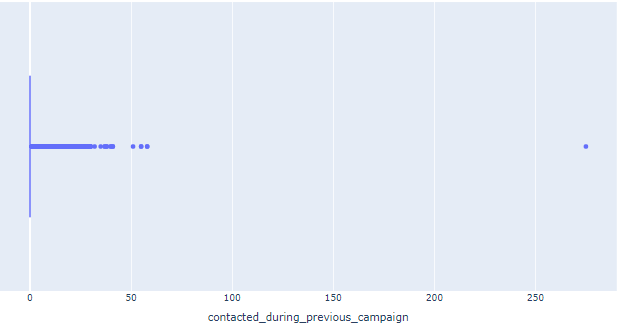
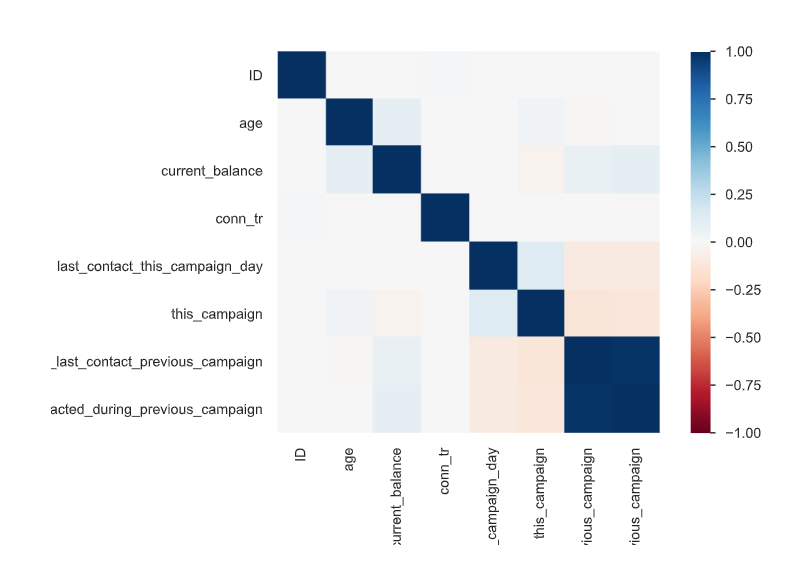


fig 10: Box plot of contacted\_during\_previous\_campaign campaign column



## **Correlation**

Correlated variables are also an indication of the bond that exists among variables. From the heatmap below using Spearman’s correlation, age has very good positive correlation with current\_balance, same with current balance and days\_since\_last\_contact\_previous\_campaign. A few weak correlation relations are also spotted.



## **Pre-processing**

To prepare the data for the machine learning process, the following steps were taken:

## **Clipping outliers**

From the detected outliers, the .clip pandas DataFrame property was used to set all values below the 5th percentile to the 5th percentile and all values above the 95th to the 95th percentile. The following charts show the box and whisker plot after deal with outliers. From the plot, days\_since\_last\_contact\_previous\_campaign and contacted\_during\_previous\_campaign, did not change much due to the distribution of the data.

fig 11: Box plot of age column fig 12: Box plot of current\_balance column

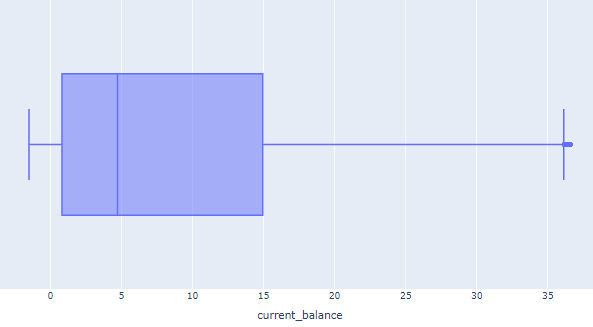
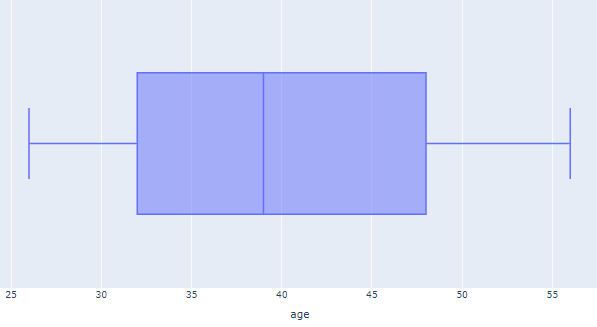
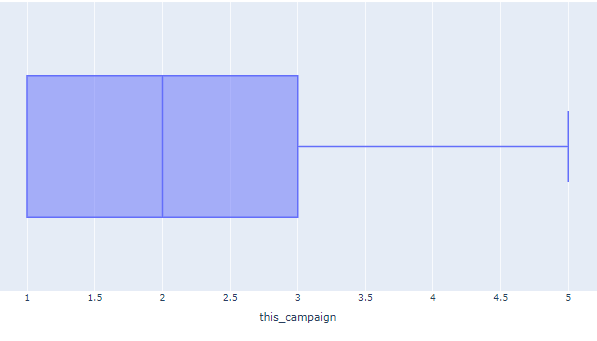
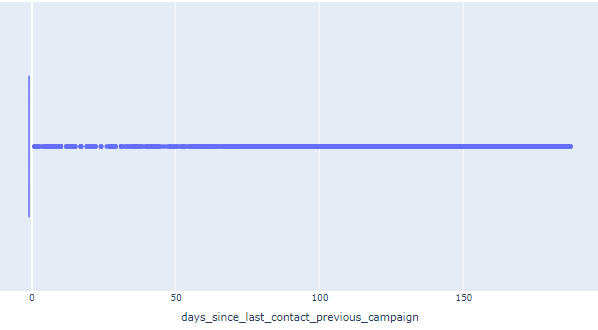


fig 13: Box plot of this\_campaign column fig 14: Box plot of days\_since\_last\_contact\_previous\_campaign column



## fig 14: Box plot of days\_since\_last\_contact\_previous\_campaign column

## 

## **Column selection**

After the preliminary exploratory data analysis, ID, country, and conn\_tr columns were dropped because they do not display any relevant information to be used in the machine learning process. This leaves us with sixteen features.

## **Scalar transformation**

The numerical variables in the data were scaled using the StandardScalar property of scikit-learn library. The property standardizes features by removing the mean and scaling it to a unit variance. The numerical columns scaled were age, current\_balance, last\_contact\_this\_campaign\_day,this\_campaign,days\_since\_last\_contact\_previous\_campaign,and contacted\_during\_previous\_campaign

## **Encoding of categorical variables**

The categorical features were also encoded using the LabelEncoder property of scikit-learn. The property encodes features by transforming them into integers, making it easier to work with. The categorical columns scaled were town, job, married, education, arrears, housing, has\_tv\_package, last\_contact, last\_contact\_this\_campaign\_month,

and outcome\_previous\_campaign.

* **Label encoding**

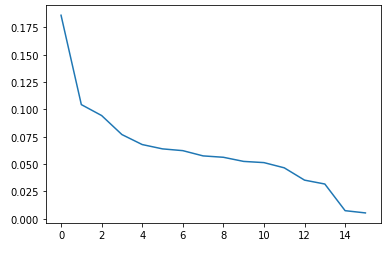
The label column was manually encoded to numeric, where ‘yes’ was replaced with ‘1’ and ‘no’ was replaced with ‘0’.

## **Feature selection methods**

The aim of performing feature selection was to limit model training to columns that are relevant and will help to better separate the classes. The methods used were:

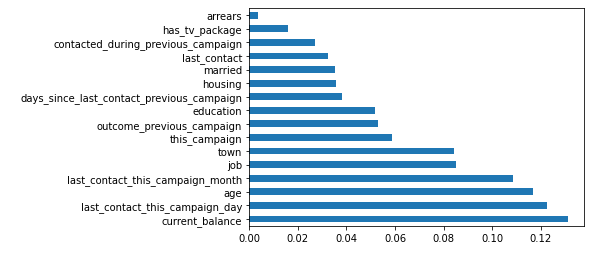
**Method 1: Scree plot**

According to Cattell (1966), the scree plot focuses on the point where there is a sharp decline in size of the eigenvalues. When the eigenvalues drop dramatically in size, an additional factor would add relatively little to the information already extracted. From the graph below, the graph sharply declines or elbows where x = 14 which informs the number of components for the Principal Component Analysis (PCA). Fig 15 shows the scree diagram.

 Fig 15: *A scree plot to determine the optimal number of components to pass as argument for the dimensionality reduction with the PCA. From the plot, there is a sharp reduction from where the x-axis is 14. This means 14 is the optimal number of features to use for the modeling*.

**Method 2: Extra Trees Classifier**

The scikit-learn library in python has extra trees classifier for feature selection. This algorithm uses statistical methods to arrive at the features with the specified number of expected features for the model training. From the algorithm, the following columns emerged most important features in descending order of importance as shown in fig 16.

Fig 16: *Results from the Extra Trees Classifier from the scikit-learn library. The classifier returns the most important features and their level of influence in order of importance starting from the bottom*. *From the plot, arrears is the least important feature.*

**Method 3: Random Forest Classifier**

Random Forest Classifier has a feature that can determine the maximum number of features for the machine learning model using the prepared data. From the Random Forest Classifier, the maximum features are 15. This result corroborates with the results of the Extra Trees Classifier and Scree plot above.

* **Dimensionality reduction with PCA**

Principal Component Analysis is a technique for reducing dimensionality and yield a better training result. PCA relies on the number of components of the training set to reduce dimensionality. The final model was constructed using all features but arrears. The number of components used for the PCA was 15.

## Modelling

The problem we’re faced with is a classification model. Three classification algorithms were sampled and compared to choose the one that gives the best result. The optimal choice of model will be based on the results of the confusion matrix, AUC curve and accuracy. The three algorithms were Extra Trees Classifier, Logistic Regression and Neural networks with TensorFlow.

The data was split using the train\_test\_split of sklearn preprocessing property. 70 percent of the dataset was used for training and 30 percent for validation and testing. The training dataset used for the training was the results from the PCA. Below details the algorithms and their hyperparameter tuning:

## Extra Trees

To handle class imbalance, the class\_weight argument was set to “balanced”. Instead of experimenting with different parameters for the maximum features and minimum samples leaf, a cross validation was used to determine the optimal choice of these two parameters. The scoring method used was roc\_auc. The grid search cross validation returned a maximum feature of 15 and minimum samples leaf of 3.

After running a prediction with the y\_test, the model returned an accuracy of 87.24 %

## Logistic Regression with cost-sensitive classification

Just as it was done for Extra Trees Classifier, a Grid Search Cross Validation was performed to determine the optimal regularization parameter for the model. To handle the class imbalance, class\_weight was set to “balanced”. The grid search cross validation returned 0.1 as the optimal regularization parameter.

The model returned an accuracy of 67.27%

## Neural Network with Keras

The neural network used was Keras, from the TensorFlow library. The first model was trained with two input layers and one output layer. The input layers used relu as the activation functions and sigmoid for the output layer. In addition, epochs were 50 and a batch size of 1. The model returned an accuracy of 82.65% and a loss of 42.27%

The model was retrained but this time with 70 epochs and a batch size of 5. This returned an accuracy of 82.65% and a loss of 41.92%

A third training included four input layers with the relu as activation function and one output layer with sigmoid as the activation function. The epochs were 50 and a batch size of 5. This returned an accuracy of 82.38% and a loss of 41.81%

In this situation, retraining the model with 70 epochs and a batch size of 5 a better combination of accuracy and loss.

## **Model Comparison & Selection**

After the model training, it is crucial to compare the various performance of the models to select which model is best for the test dataset. The table below tabulates the results.

|  |  |  |
| --- | --- | --- |
| Algorithm | Accuracy | Loss function |
| Extra Trees Classifier | 87.24% | - |
| Logistic Regression | 67.27% | - |
| Neural Network | 82.65% | 41.92% |

Table 1: *A comparison of machine learning algorithms used to train the*

*dataset and the accuracy scores. All models were trained using*

*Principal Component Analysis and cost-sensitive was factored.*

From the results above, Extra trees gave a better accuracy followed by Neural Network and then Logistic Regression in that order. Extra Trees Classifier was chosen as the optimal model since it had the better results and generalized better.

## Evaluation

* **Accuracy**

Extra Trees, which is our model of choice for this data set produced an accuracy of 87.24%. This is an indication that 87.24 % of dataset was correctly classified out of the total observations.

* **Confusion Matrix**

To get a better understanding of the precision and recall, a confusion matrix was used to visualize as in fig 17. From the matrix, the model had a problem with the false negatives because it classified some respondents as no (in this case 0) but are yes (in this case 1) in reality. The model has a problem with type II error however, the model has a very good precision.

It is recommended that StirCom pays attention to the current balance and age of the target customers since it plays a very important role in determining if customer will subscribe to the 5G service or not.

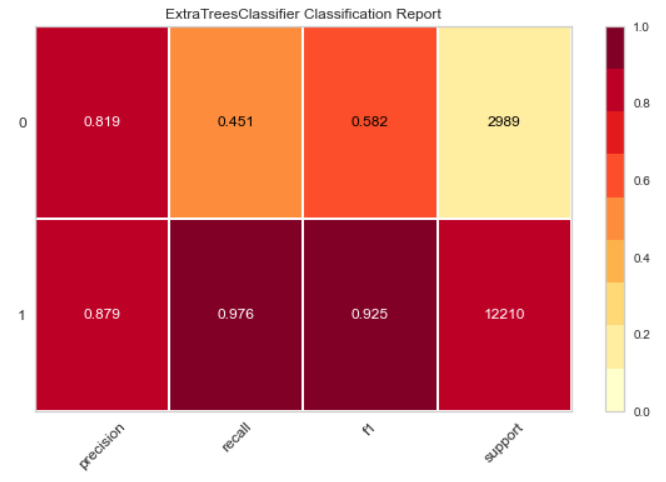


Fig 17: *Confusion matrix report for Extra Trees Classifier in percentage of correctly or wrongly classified labels.*

* **Receiver Operator Characteristic (ROC)**

The ROC is a probability curve that plots the True Positive Rate (TPR or sensitivity) against the False Positive Rate (FPR or 1-sensitivity) at various threshold values which is plotted with the help of an AUC (Area Under the Curve) curve. AUC is between 0 and 1. If AUC =1, then the classifier can perfectly distinguish between yes and no points correctly. If, the AUC is 0, then the classifier would be predicting all yes as no and no as yes. The higher the AUC, the better the performance of the model at distinguishing between the yes and no classes. From fig 18, we have an overall AUC of 0.89 which is a good result in general.

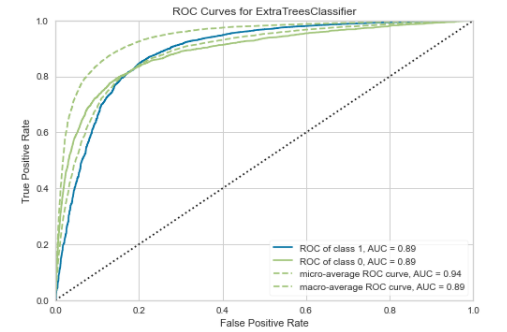


Fig 18: ROCAUC *for Extra Trees Classifier*

## Conclusion

The dataset we were presented with had 50662 rows and 20 columns. The columns were reduced to 15 after performing feature importance with the scree plot, Extra Trees Classifier and Random Forest. The dimensions were further reduced using Principal Component. Out of the three algorithms applied to the data, Extra Trees Classifier performed best with an accuracy of 87.24%.

It is recommended that StirCom pays attention to the current balance, age and profession of their customers as they compose some of the important features in predicting the label.