CUSTOMER CHURN PREDICTION

PROJECT FINAL REPORT

BACHELOR OF TECHNOLOGY- V Sem CSE

Department of Computer Science & Engineering

SUBMITTED BY

Batch No:- 12

S NO	NAME	SRN
1	CHANDAN KUMAR.S	PES2UG20CS804
2	VIJAY.J	PES2UG20CS815
3	YUVARAJ.S	PES2UG20CS819

PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)

100 Feet Ring Road, BSK III Stage, Bengaluru-560085

ABSTRACT

Problem Statement

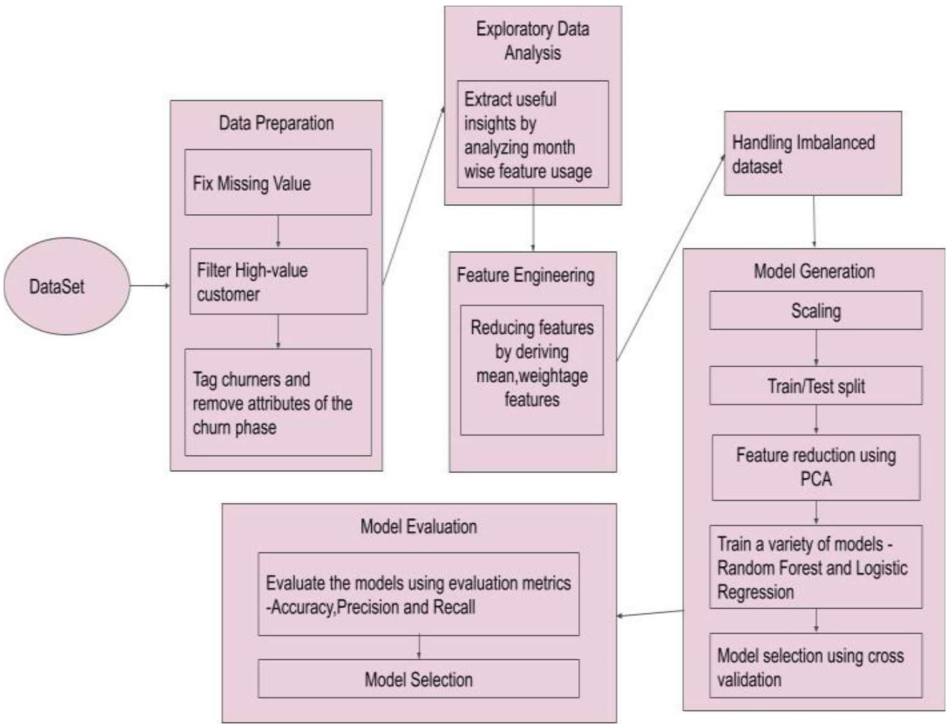
Churn prediction means detecting which customers are likely to leave a service or to cancel a subscription to a service. It is a critical prediction for many businesses because acquiring new clients often costs more than retaining existing ones. Once we can identify those customers that are at risk of cancelling, we should know exactly what marketing action to take for each individual customer to maximize the chances that the customer will remain

Different customers exhibit different behaviors and preferences, so they cancel their subscriptions for various reasons. It is critical, therefore, to proactively communicate with each of them in order to retain them in your customer list. You need to know which marketing action will be the most effective for each and every customer. The Technique proposed in this paper will overcome discussed issues and it will be applied on those customers who want to leave in near future and predict them based on some parameters

FEASIBILITY STUDY

Customer churn is a common problem across businesses in many sectors. If you want to grow as a company, you have to invest in acquiring new clients. Every time a client leaves, it represents a significant investment lost. Both time and effort need to be channeled into replacing them. Being able to predict when a client is likely to leave, and offer them incentives to stay, can offer huge savings to a business. As a result, understanding what keeps customers engaged is extremely valuable knowledge, as it can help you to develop your retention strategies, and to roll out operational practices aimed at keeping customers from walking out the door.

DESIGN APPROACH/ METHODOLOGY/



Architectural Design for Customer Churn Prediction

Feature selection is a crucial step for selecting the required elements from the data set based on the knowledge. The dataset used here consists of many features out of which we chose the needed features, which enable us to improve performance measurement and are useful for decision-making purposes while remaining will have less importance. The performance of classification increases if the dataset is having only valuable variables and which are highly predictable. Thus having only significant features and reducing the number of irrelevant attributes increases the performance of classification. Many techniques have been proposed for customer churn prediction in the telecommunication industry. Here by using logistic regression, Random Forest and KNN we can predict the probability of a churn i.e., the likelihood of a customer to cancel the subscription and we can evaluate the models using performance metrics like accuracy

Predictive modeling is mainly concerned with predicting how the customer will behave in the future by analyzing their past behavior. Predicting customers who are likely to churn is one example of the predictive modeling. Predictive modeling is used in analyzing Customer Relationship Management

(CRM) data and DM to produce customer-level models that describe the likelihood that a customer will take a particular action. The actions are usually sales, marketing and customer retention related. There are many models that can used to define distinguish between churners and non-churners in an organization. These models can be classified into traditional models or techniques (RA and DT) and soft computing techniques

- **Analyze the problem and find a goal**

The kind of insights you want to glean from your analysis would decide what kind of problem you are going to solve. You have to drill down the problem area of the customer churn into the right questions so that the response to them would give right predictions. To predict customer churn machine learning problems can be either of two types: Classification or Regression.

- **Classification**

This type would need the data scientists to determine to which class or category the customer belongs. It is referred to as a data point.To train the algorithm, they make use of the historical data of the customers and use the predefined target variables.

- **Regression**

Regression analysis is one of the widely used methods in customer churn prediction software. It is a value that is used in statistical analysis to define the relationship between the customer churn and the data values that influence it. This analysis helps in finding exact values for the business prediction. E.g. it can give you the exact time within which a customer is predicted to churn

- **Import Libraries required to create the Customer Churn Model**

So, we import pandas for data analysis, NumPy for calculating N-dimensional array, seaborn, and matplotlib to visualize the data, these all are the basic libraries required for the preprocessing of the data.

Attributes of Dataset:-

In our Dataset where we have of 10000 observations and 12 variables. Independent variables contain information about customers. Dependent variable refers to customer abandonment status.

Variables:

RowNumber — corresponds to the record (row) number and has no effect on the output. This column will be removed.

CustomerId — contains random values and has no effect on customer leaving the bank. This column will be removed.

Surname — the surname of a customer has no impact on their decision to leave the bank. This column will be removed.

CreditScore — can have an effect on customer churn, since a customer with a higher credit score is less likely to leave the bank.

Geography — a customer's location can affect their decision to leave the bank. We'll keep this column.

Gender — it's interesting to explore whether gender plays a role in a customer leaving the bank. We'll include this column, too.

Age — this is certainly relevant, since older customers are less likely to leave their bank than younger ones.

Tenure — refers to the number of years that the customer has been a client of the bank. Normally, older clients are more loyal and less likely to leave a bank.

Balance — also a very good indicator of customer churn, as people with a higher balance in their accounts are less likely to leave the bank compared to those with lower balances.

NumOfProducts — refers to the number of products that a customer has purchased through the bank.

HasCrCard — denotes whether or not a customer has a credit card. This column is also relevant, since people with a credit card are less likely to leave the bank. (0=No,1=Yes)

IsActiveMember — active customers are less likely to leave the bank, so we'll keep this. (0=No,1=Yes)

EstimatedSalary — as with balance, people with lower salaries are more likely to leave the bank compared to those with higher salaries.

Exited — whether or not the customer left the bank. This is what we have to predict. (0=No,1=Yes)

```
[ ] # The first 5 observation units of the data set were accessed.
    df.head()
```

RowNumber	customerid	surname	creditscore	geography	gender	age	tenure	balance	numofproducts	hascrcard	isactivemember	estimatedsalary	exited
1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	101348.88	1
2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0
3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0
5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0

The above picture which shows the top or the first five observation of the dataset which is used

- **Load Churn Prediction Dataset**

```
In [2]: # Reading the dataset
        df = pd.read_csv("../input/predicting-churn-for-bank-customers/Churn_Modelling.csv", index_col=0)
        df.columns = map(str.lower, df.columns)
```

Out[6]:

	count	mean	std	min	25%	50%	75%	max
customerid	10000.0	1.569094e+07	71936.186123	15565701.00	15628528.25	1.569074e+07	1.575323e+07	15815690.00
creditscore	10000.0	6.505288e+02	96.653299	350.00	584.00	6.520000e+02	7.180000e+02	850.00
age	10000.0	3.892180e+01	10.487806	18.00	32.00	3.700000e+01	4.400000e+01	92.00
tenure	10000.0	5.012800e+00	2.892174	0.00	3.00	5.000000e+00	7.000000e+00	10.00
balance	10000.0	7.648589e+04	62397.405202	0.00	0.00	9.719854e+04	1.276442e+05	250898.09
numofproducts	10000.0	1.530200e+00	0.581654	1.00	1.00	1.000000e+00	2.000000e+00	4.00
hascrcard	10000.0	7.055000e-01	0.455840	0.00	0.00	1.000000e+00	1.000000e+00	1.00
isactivemember	10000.0	5.151000e-01	0.499797	0.00	0.00	1.000000e+00	1.000000e+00	1.00
estimatedsalary	10000.0	1.000902e+05	57510.492818	11.58	51002.11	1.001939e+05	1.493882e+05	199992.48
exited	10000.0	2.037000e-01	0.402769	0.00	0.00	0.000000e+00	0.000000e+00	1.00

In this dataset there are 10000 rows and 13 columns are present. There are some categorical and some numerical columns present.

- **Preprocess Dataset**

Now it's time to pre-process the data, firstly we will observe the dataset, this means we have to see the data types of the columns, other functionalities, and parameters of each column. First, we check the dataset information using the info() method

```
# Feature information
df.info()
```

```
Int64Index: 10000 entries, 1 to 10000
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   customerid       10000 non-null  int64
 1   surname          10000 non-null  object
 2   creditscore      10000 non-null  int64
 3   geography        10000 non-null  object
 4   gender           10000 non-null  object
 5   age              10000 non-null  int64
 6   tenure           10000 non-null  int64
 7   balance          10000 non-null  float64
 8   numofproducts    10000 non-null  int64
 9   hascrcard        10000 non-null  int64
 10  isactivemember   10000 non-null  int64
 11  estimatedsalary  10000 non-null  float64
 12  exited           10000 non-null  int64
dtypes: float64(2), int64(8), object(3)
memory usage: 1.1+ MB
```

Checking for the Missing values in dataset

In our dataset where we are going to check whether there is any missing values or empty observation before implementing the model

```
[ ] # Missing Observation Analysis
    df.isnull().sum()
```

```
customerid         0
surname            0
creditscore        0
geography          0
gender             0
age                0
tenure             0
balance            0
numofproducts      0
hascrcard          0
isactivemember     0
estimatedsalary    0
exited             0
NewAge             0
dtype: int64
```

Splitting data

This is the important part is we have to split our data into training and testing parts by which we do further processes.

Now we have to split our dataset into train and test sets, where the training set is used to train the model, and the testing set is used for testing the values of targeted columns

In [57]:
```
# Train-Test Separation
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.20,
                                                    random_state=12345)
```

In [58]:
```
# Because it's an unstable data set, we're going to increase the number of samples.
# References: https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.combine.SMOTETomek.html
from imblearn.combine import SMOTETomek

smk = SMOTETomek()
# Oversample training  data
X_train, y_train = smk.fit_sample(X_train, y_train)

# Oversample validation data
X_test, y_test = smk.fit_sample(X_test, y_test)
```

In [59]:
```
print(X_train.shape, X_test.shape, y_train.shape, y_test.shape)
```

```
(12692, 17) (3136, 17) (12692,) (3136,)
```

Feature Engineering

```
[ ] df["NewAGT"] = df["age"] - df["tenure"]
    df["CreditsScore"] = pd.qcut(df['creditscore'], 10, labels = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
    df["AgeScore"] = pd.qcut(df['age'], 8, labels = [1, 2, 3, 4, 5, 6, 7, 8])
    df["BalanceScore"] = pd.qcut(df['balance'].rank(method="first"), 10, labels = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
    df["EstSalaryScore"] = pd.qcut(df['estimatedsalary'], 10, labels = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
    df["NewEstimatedSalary"] = df["estimatedsalary"] / 12
```

```
df.head()
```

›fproducts	hascrcard	isactivemember	estimatedsalary	exited	NewAge	NewAGT	CreditsScore	AgeScore	BalanceScore	EstSalaryScore	NewEstimatedSalary
1	1	1	101348.88	1	(40.0, 46.0]	40	4	6	1	6	8445.740000
1	0	1	112542.58	0	(40.0, 46.0]	40	4	6	5	6	9378.548333
3	1	0	113931.57	1	(40.0, 46.0]	34	1	6	10	6	9494.297500
2	0	0	93826.63	0	(35.0, 40.0]	38	7	5	1	5	7818.885833
1	1	1	79084.10	0	(40.0, 46.0]	41	10	6	8	4	6590.341667

We have used the featured engineering which is used to transform raw data.

Modeling:-

4) Modelling

```
[ ] models = []
    models.append(('LR', LogisticRegression(random_state = 12345)))
    models.append(('KNN', KNeighborsClassifier()))
    models.append(('CART', DecisionTreeClassifier(random_state = 12345)))
    models.append(('RF', RandomForestClassifier(random_state = 12345)))
    models.append(('SVM', SVC(gamma='auto', random_state = 12345)))
    models.append(('XGB', GradientBoostingClassifier(random_state = 12345)))
    models.append(("LightGBM", LGBMClassifier(random_state = 12345)))
    models.append(("CatBoost", CatBoostClassifier(random_state = 12345, verbose = False)))

    # evaluate each model in turn
    results = []
    names = []
```

For our Dataset where we have used multiple machine learning models such as Logistic Regression , KNeighourClassifier, Decision Tree, Random Forest , Support Vector Machine (SVM), GradientBoosting(XGB), LightGBM and CatBoost by using this we are going to find the best prediction or the accuracy for the dataset and will be we compared between them.

Accuracy Score:-

```
[ ] for name, model in models:
            model.fit(X_train, y_train)
            y_pred = model.predict(X_test)
            accuracy = accuracy_score(y_test, y_pred)
            msg = "%s: (%f)" % (name, accuracy)
            print(msg)
```

```
LR: (0.749523)
KNN: (0.739352)
CART: (0.812778)
RF: (0.849332)
SVM: (0.795296)
XGB: (0.889065)
LightGBM: (0.904641)
CatBoost: (0.906866)
```

This are the accuracy score which have given by the machine learning models and from this accuracy score where we will be considering accuracy value which as more than accuracy>(0.80) and where accuracy<(0.80) will be discarded

Hyperparameter Model Tuning

```
[ ] # Hyperparameters have previously been obtained with the help of GridSearchCV.
    models = []
    models.append(('XGB', GradientBoostingClassifier(random_state = 12345,learning_rate = 0.05, max_depth = 5, min_samples_split = 2, n_estimators = 50
    models.append(("LightGBM", LGBMClassifier(random_state = 12345,  learning_rate = 0.05, max_depth = 3, n_estimators = 1000)))
    models.append(("CatBoost", CatBoostClassifier(random_state = 12345, verbose = False, depth = 10, iterations = 1000, l2_leaf_reg = 5, learning_rate

    # evaluate each model in turn
    results = []
    names = []

    for name, model in models:
            model.fit(X_train, y_train)
            y_pred = model.predict(X_test)
            accuracy = accuracy_score(y_test, y_pred)
            msg = "%s: (%f)" % (name, accuracy)
            print(msg)
```

```
XGB: (0.906866)
LightGBM: (0.912905)
CatBoost: (0.898284)
```

By using Model tuning where we have found the optimized value hyperparameter were we have found by using the gridsearch and LightGBM gives more accuracy which have (0.912985) then the gradient boosting and cat boost

So we will say that LightGBM model will give optimized accuracy result from the remaining machine Learning model

LITERATURE REVIEW

The hybrid approach contains three phases: In the first phase, SVM-RFE (SVM-recursive feature elimination) is employed to reduce the feature set. In the second phase, dataset with reduced features is then used to obtain SVM model and support vectors are extracted.

A method to predicts the customer churn in a Bank, using machine learning techniques, which is a branch of artificial intelligence is proposed. The research promotes the exploration of the likelihood of churn by analyzing customer behavior. The KNN, SVM, Decision Tree, and Random Forest classifiers are used in this study. As a result, the use of the Random Forest model after oversampling is better compared to other models in terms of accuracy.

Approach: Random Forest model after oversampling is better compared to other models in terms of accuracy.

Random Forest are the best algorithms to predict Bank Customer Churn since they have the highest accuracy (86,85% and 86.45%)

The fast expansion of the market in every sector is leading to superior subscriber base for service providers. Added competitors, novel and innovative business models and enhanced services are increasing the cost of customer acquisition. In such a fast set up, service providers have realized the importance of retaining the on-hand customers. Itis therefore essential for the service providers to prevent churn- a phenomenon which states that customer wishes to quit the service of the company. This paper reviews the most popular machine learning algorithms used by researchers for churn predicting, not only in banking sector but also other sectors which highly depends on customer participation. This paper reviews the most popular machine learning algorithms used by researchers for churn predicting, not only in banking sector but also other sectors which highly depends on customer participation.Customer Relationship Management (CRM) can improve productivity, recommend relevant promotions to the group of likely churn customers based on similar behavior patterns, and excessively improve marketing used a Decision Tree, Random Forest, and XGBoost to predict the customers who are likely to cancel the subscription which can offer them

better services and reduce the churn rate. By pre-processing and feature selection, the data set for training and testing. For the above-mentioned algorithm, it is necessary to do some feature engineering to have more efficient and accurate results. We implemented an EDA using Visualization, statistical tests for feature selection and Data mining methods for predicting the likely churners by utilizing a Logistic Regression Model. Here dataset has been analysed by using the data visualization techniques before entering into the modelling process

As churn prediction models should be both accurate and comprehensible, we will focus on the use of rule-based classification techniques. More specifically, we will induce rule-sets from a churn dataset using Ant Miner+ and ALBA, as well as with more tradition alrule induction techniques C4.5 and RIPPER. The workings of Ant Miner+ and ALBA are explained briefly in the next two sections

This paper proposed two main contributions; the first one is a model for customer Churn prediction by analysing user-generated content, and the second model is identifying main attributes that help the retention department to keep their customers and prevent them from the churn.

Customer churn prediction model using UGC proposed in Fig. 1, the proposed model consists of multiple processes, as shown in Fig. 1; these steps are:

Step 1: User creates his user-generated content; this content could be post, opinion, or comments.

Step 2: English treebank applies text preprocessing, stemming, and lemmatization on English text to extract essential words in their basic form.

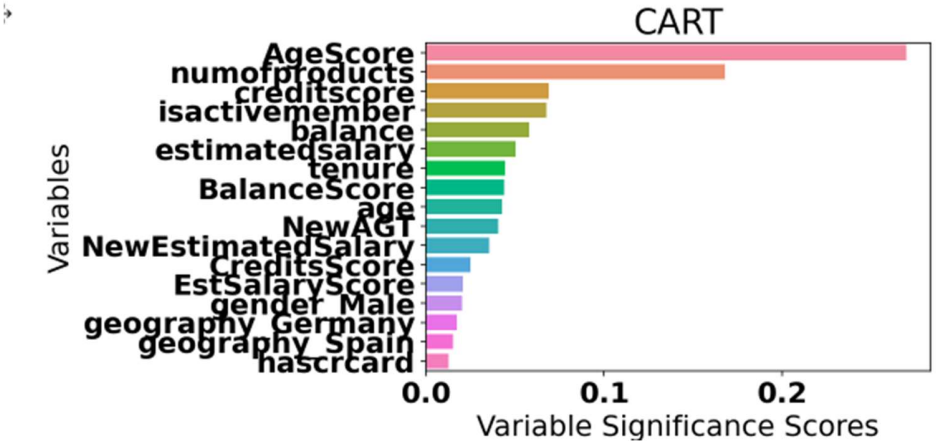
Research on churn algorithm and model: the existing research mainly focuses on regression, neural network, decision tree, and other algorithms. Predicted customer churn with decision tree and artificial neural network algorithm. Sato et al. [19] compared the effects of principal component analysis and decision tree algorithm on customer churn prediction and the laws

The first hybrid model is based on cascading two ANN models,in which the first one performs the data reduction task and the secondone for churn prediction. That is, the original training set isused to ‘test’ the fist created ANN model, which is based on the‘best’ baseline model identified above.

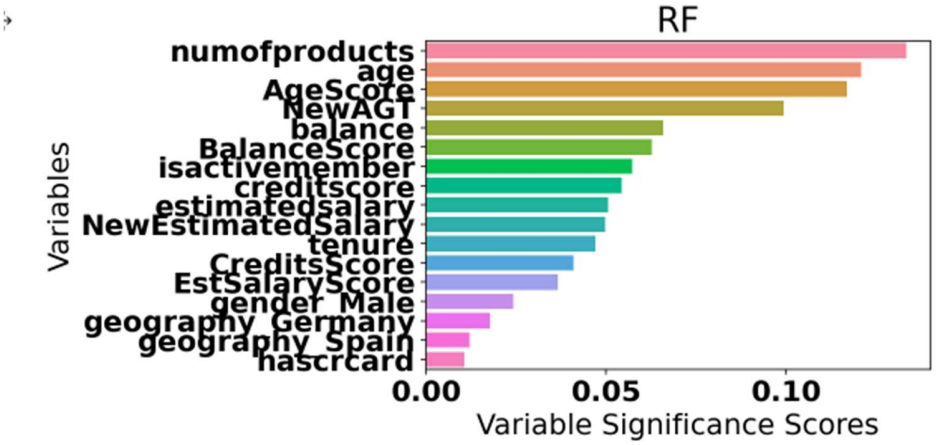
As there is no 100% accuracy,there are a numberofcorrectly and incorrectly predicted datafrom the training set by the first ANN model. Consequently, theincorrectly predicted data can be regarded as outliers since theANN model cannot predict them accurately. Then, the correctlypredicted data by the firstANN model are used to train the secondANN model as the prediction model for later prediction.

Implementation

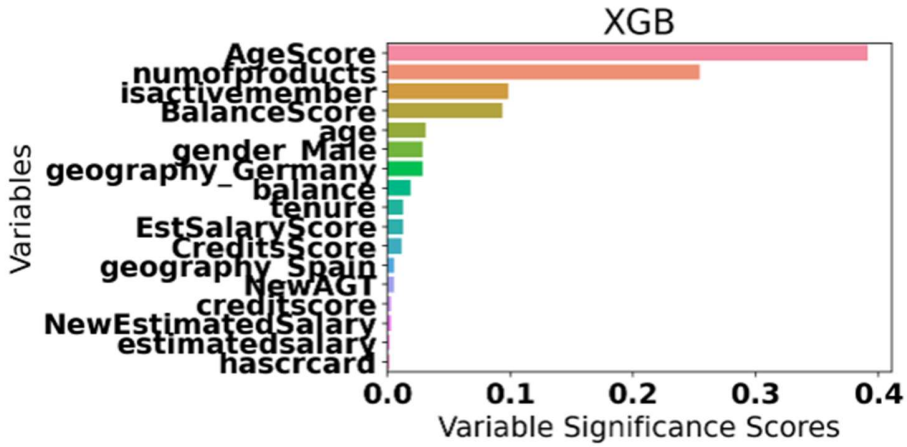
Hyperparameter:
Decision tree:



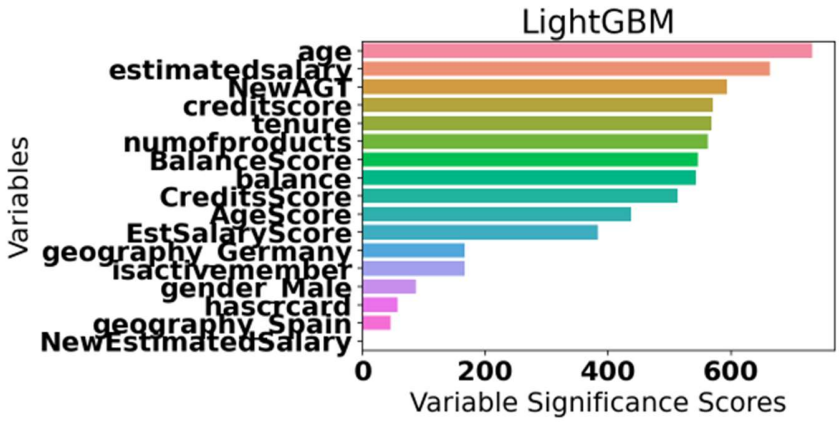
Random forest:



Gradient:



LightGbm



The parameter have be optimized are

Randon_state- Random number seed. If int, this number is used to seed the C++ code.

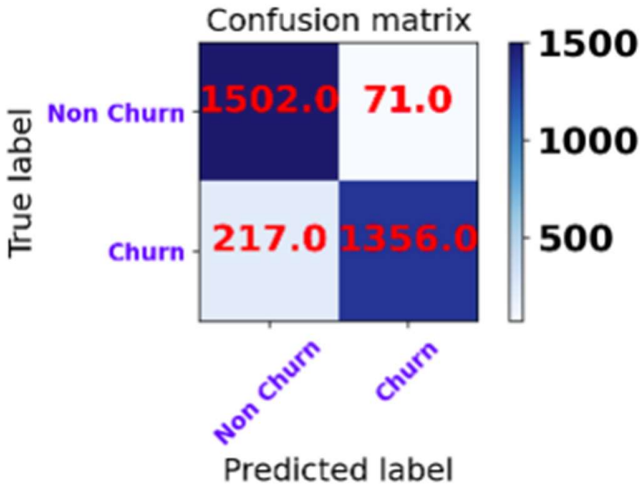
If RandomState object (numpy), a random integer is picked based on its state to seed the C++ code. If None, default seeds in C++ code are used

learning_rate- learning rate of the tree is 0.05

Max Depth- Maximum tree depth for base learners, <=0 means no limit

n_estimators :- Number of basics learners it's value is 1000

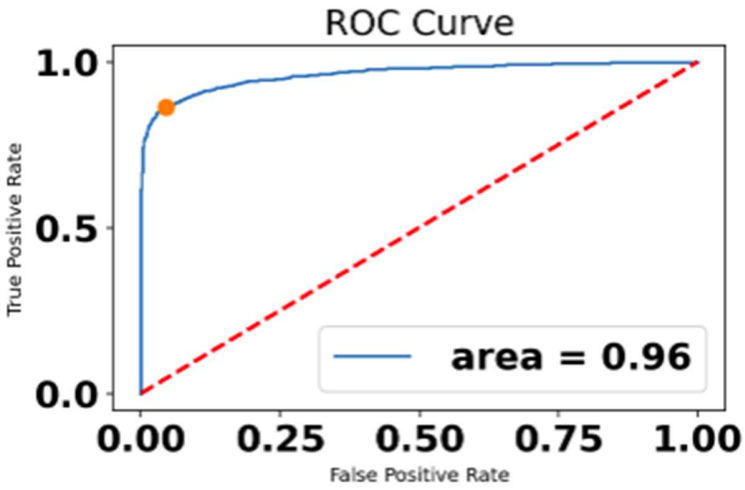
Confusion matrix for LightGM



Where in the confusion matrix for

- non - churn & non - churn value is 1502.0
- Non - churn & churn value is 71.0
- Churn & non - churn value is 217.0
- Churn value & churn value is 1356.0

ROC(rate of curve)



The rate of the curve for the lightgbm is 0.96

Conclusion

Churn prediction and management is very important for enterprises in the competitive market to predict possible churners and take proactive actions to retain valuable customers and profit. Therefore, to build an effective customer churn prediction model, which provides a certain level of accuracy, has become a research problem for both academics and practitioners in recent years. In this paper, we consider two different hybrid data mining techniques by neural networks to examine their performances

REFERENCES

Paper1: International Journal of Computer Applications (0975 8887)

Volume 154 No.10, November 2016. A Survey on Customer Churn Prediction using Machine Learning Techniques

Paper 2: **International Advanced Research Journal in Science, Engineering and Technology**

Vol. 8, Issue 6, June 2021

Paper 3: International Journal for Research in Applied Science & Engineering Technology (IJRASET)*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429Volume 8 Issue V May 2020- Available at www.ijraset.com*©IJRASET: All Rights are Reserved Telecom Customer Churn Prediction

Paper 4: Building comprehensible customer churn prediction models with advanced rule induction techniques

Paper 5: DOAA, M. E.; MOHAMED, H. A survey on sentiment analysis challenges. Journal of King Saud University-Engineering Sciences, 2016, 4.

Paper 6: GAUR, Abhishek; DUBEY, Ratnesh. Predicting Customer Churn Prediction In Telecom Sector Using Various Machine Learning Techniques. In: 2018 International Conference on Advanced Computation and Telecommunication (ICACAT). IEEE, 2018. p. 1-5.

Paper7: VAFEIADIS, Thanasis, et al. A comparison of machine learning techniques for customer churn prediction. Simulation Modelling Practice and Theory, 2015, 55: 1-9.

Paper8:ALHARBI, Ahmed Sulaiman M.; DE DONCKER, Elise. Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information. Cognitive Systems Research, 2019, 54: 50-61.