Abstract

The customer churn prediction (CCP) is one of the challenging problems in business and industry .With the advancement in the field of machine learning and artificial intelligence, the possibilities to predict customer churn has increased significantly. Our proposed methodology, consists of six phases. In the first two phases, data pre-processing and feature analysis is performed. In the third phase, feature selection is taken into consideration using Random Forest .Next, the data has been split into two parts train and test set in the ratio of 80% and 20% respectively. In the prediction process, most popular predictive models have been applied, namely, logistic regression, naive bayes, K-nearest neighbors (KNN), random forest and AdaBoost. In addition, K-fold cross validation has been used over train set for hyperparameter tuning and to prevent overfitting of models. Finally, the obtained results on test set have been evaluated using confusion matrix and AUC curve. It was found that Adaboost and random forest Classifier gives the highest accuracy of 92.91 % and 93.29 % respectively.

Keywords Customer Churn Prediction . Machine Learning. Predictive Modeling. Confusion Matrix. AUC Curve. Logistic regression. naive bayes. K nearest neighbors (KNN). random forest. AdaBoost.

1. Introduction

Customer churn is simply the rate at which customers leave doing business with an entity. Simply put, churn prediction involves determining the possibility of customers stopping doing business with an entity. It is a critical prediction for many businesses because acquiring new clients often costs more than retaining existing ones. Customer churn measures how and why are customers leaving the business. Churn rate is a marketing metric that describes the number of customers who leave a business over a specific time period. Every user is assigned a prediction value that estimates their state of churn at any given time. This value is based on: User demographic information, Browsing behavior, Historical purchase data among other information. It factors in our unique and proprietary predictions of how long a user will remain a customer. This score is updated every day for all users who have a minimum of one conversion. The values assigned are between 1 and 5. One of the main aim of Customer Churn prediction is to help in establishing strategies for customer retention. Along with growing competition in markets for providing services, the risk of customer churn also increases exponentially. Therefore, establishing strategies to keep track of loyal customers (non-churners) has become a necessity. The customer churn models aim to identify early [1] churn signals and try to predict the customers that leave voluntarily. Thus many companies have realized that their existing database is one of their most valuable asset [2,5] and according to,[3,4] churn prediction is a useful tool to predict customers at risk.

1.1 Problem description

In order to capture the aforementioned problem, company should predict the customer's behavior correctly. Customer churn management can be done in two ways: (1) Reactive & (2) Proactive. In the reactive approach, company waits for the cancellation request received from the customer, afterwards, company offers the attractive plans to the customer for the retention. In the proactive approach, the possibility of churn is predicted, accordingly the plans are offered to the customers. Its a binary classification problem where churners are separated from the non churners. In order to tackle this problem, machine learning has proved itself as a highly efficient technique, for forecasting information on the basis of previously captured data [10,4,5]. In machine learning models, after pre-processing feature selection plays a significant role to improve the classification accuracy. A plenty of approaches were developed by researchers for feature selection that are useful to reduce the dimension, computation complexity & overfitting. In churn prediction, those feature are extracted from the given input vector which are useful for the prediction of churn. In this work, to tackle this problem we have used the following Machine Learning techniques:(1) Logistic Regression,(2) Naïve Bayes (3) K-nearest neighbors (KNN) (4) Random Forest Classifier (5)Ada Boost. Furthermore, for better understanding of the data, the data have been pre-processed and important feature vectors have been extracted using RandomForest.

1.2 Advantage of proposed technique over the existing

The merits of the proposed algorithm has listed as follows:

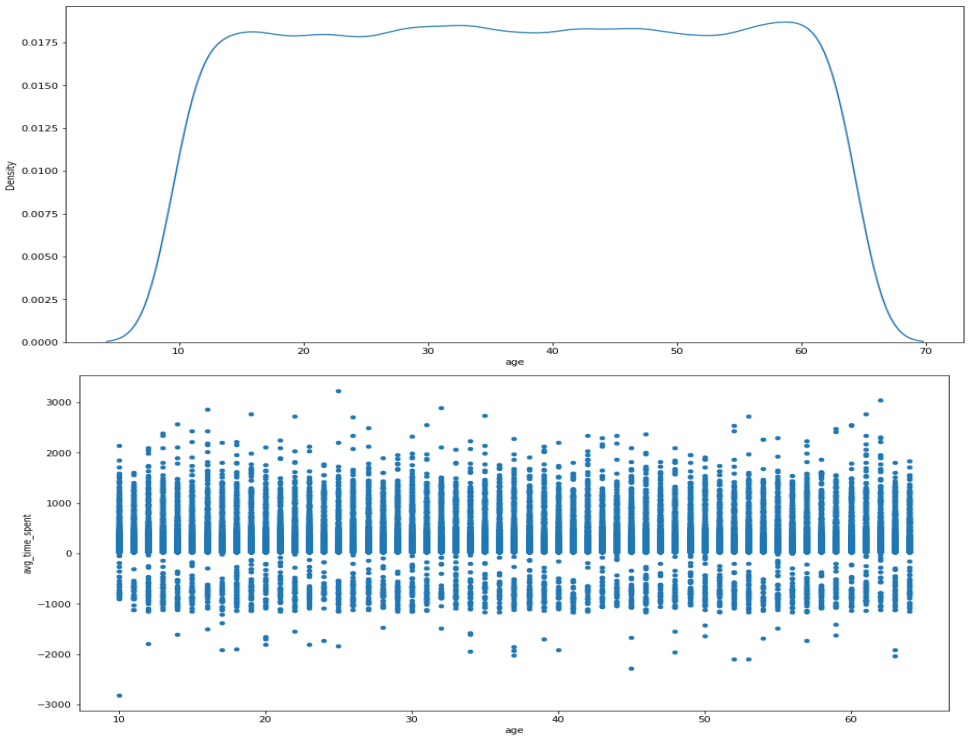
- We have applied Random Forest Classifier algorithm to perform feature selection and to reduce the dimensions of the data-set.
- After, pre-processing of data, we have applied some of the famous machine learning techniques which are used for predictions and k-fold cross validation has been performed to prevent overfitting.
- Then we have evaluated the algorithms on test set using confusion matrix and AUC curve, which have been mentioned in the form of graphs and tables in order to compare which algorithm performs best for this particular data-set.

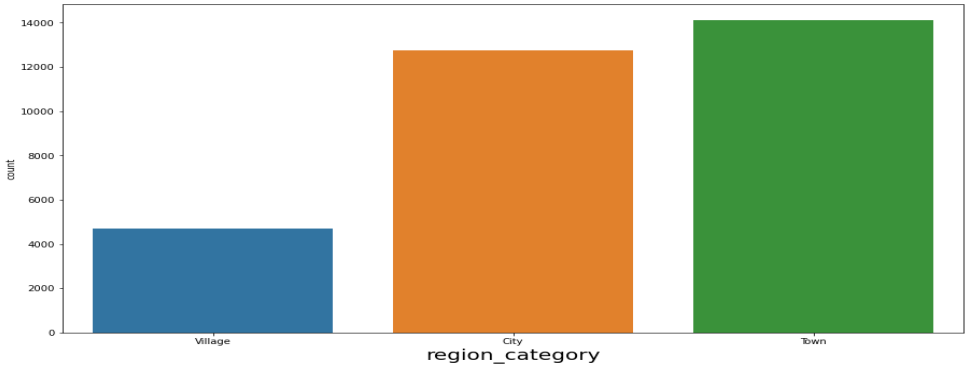
2. Preliminaries

In the current, we have tried to describe techniques we have used for data cleaning and pre-processing in order to make the predictions more robust and machine learning models applied for the classification.

2.1 Exploratory data analysis (EDA)

It is a way of exploring the hidden features that are present in the rows and columns of data by visualizing, summarizing and interpreting of data as seen in Figure(1).





Figure(1) Exploratory Data Analysis: age feature vs Probability density function(top)age vs averge time spent (middle) region category number(bottom).

In our data-set we divided the data-set into two parts that is 1st Categorical features and 2nd Numerical features. From 21 features, 17 features were categorical and 6 were numerical as shown in Figure(2).

	gender	security_no	region_category	membership_category	joining_date	joined_through_referral	referral_id	preferred_offer_types	medium_of_operation	internet_option	last_visit_time	avg_frequency_login_days	used_special_discount	offer_application_preference	past_complaint	complaint_stat
0	F	XW0DQ7H	Village	Platinum Membership	2017-08-17	No	xxxxxxxx	Gift Vouchers/Coupons	?	Wi-Fi	16:08:02	17.0	Yes	Yes	No	Not Applicab
1	F	5K0N3X1	City	Premium Membership	2017-08-28	?	CID21329	Gift Vouchers/Coupons	Desktop	Mobile_Data	12:38:13	10.0	Yes	No	Yes	Solv
2	F	1F2TCL3	Town	No Membership	2016-11-11	Yes	CID12313	Gift Vouchers/Coupons	Desktop	Wi-Fi	22:53:21	22.0	No	Yes	Yes	Solved in Follo
3	M	WGJ33N	City	No Membership	2016-10-29	Yes	CID3793	Gift Vouchers/Coupons	Desktop	Mobile_Data	15:57:50	6.0	No	Yes	Yes	Unsolv
4	F	SVZXCWB	City	No Membership	2017-09-12	No	xxxxxxxx	Credit/Debit Card Offers	Smartphone	Mobile_Data	15:46:44	16.0	No	Yes	Yes	Solv

	age	days_since_last_login	avg_time_spent	avg_transaction_value	points_in_wallet	churn_risk_score
0	18	17	300.63	53005.25	781.75	0
1	32	16	306.34	12838.38	NaN	0
2	44	14	516.16	21027.00	500.69	1
3	37	11	53.27	25239.56	567.66	1
4	31	20	113.13	24483.66	663.06	1

Figure(2) Exploratory Data Analysis: categorical variables (top) numerical variables (bottom).

2.2 Data Preprocessing

Data preprocessing in Machine Learning is a crucial step that helps enhance the quality of data to promote the extraction of meaningful insights from the data. Data preprocessing in Machine Learning refers to the technique of preparing (cleaning and organizing) the raw data to make it suitable for a building and training Machine Learning models. The concepts that will cover for this dataset are:

1. Handling Standard/ Non-Standard Missing Values
2. Feature Engineering Handling Categorical Variables
3. Outliers
4. Categorical Data Encoding
5. Feature Scaling

2.2.1 Handling Standard/ Non-Standard Missing Values

Standard/ Non-Standard missing values badly affect our model performance because, are irreverent in nature they are misplaced in the dataset so we have to remove them and replace them with other values.

Table(1) show Standard missing values which are NaN values but there are some non-standard missing values which needs to be treated before further processing

	missing_values	percentage
region_category	5428	14.673443
points_in_wallet	3443	9.307418
preferred_offer_types	288	0.778547
age	0	0.000000
avg_time_spent	0	0.000000
feedback	0	0.000000
complaint_status	0	0.000000
past_complaint	0	0.000000
offer_application_preference	0	0.000000

Table(1) standard missing values and their percentage.

There are 10 features that has non-standard missing values. referral_id has the most missing values around 48% showing in table (2).

	missing_values	percentage
referral_id	17846	48.242863
joined_through_referral	5438	14.700476
region_category	5428	14.673443
medium_of_operation	5393	14.578828
avg_frequency_login_days	4205	11.367323
points_in_wallet	3579	9.675065
days_since_last_login	1999	5.403871
avg_time_spent	1719	4.646951
preferred_offer_types	288	0.778547
gender	59	0.159494
age	0	0.000000
feedback	0	0.000000
complaint_status	0	0.000000

Table(2) non-standard missing values and their percentage.

The non-standard missing values are replaced with NaN:

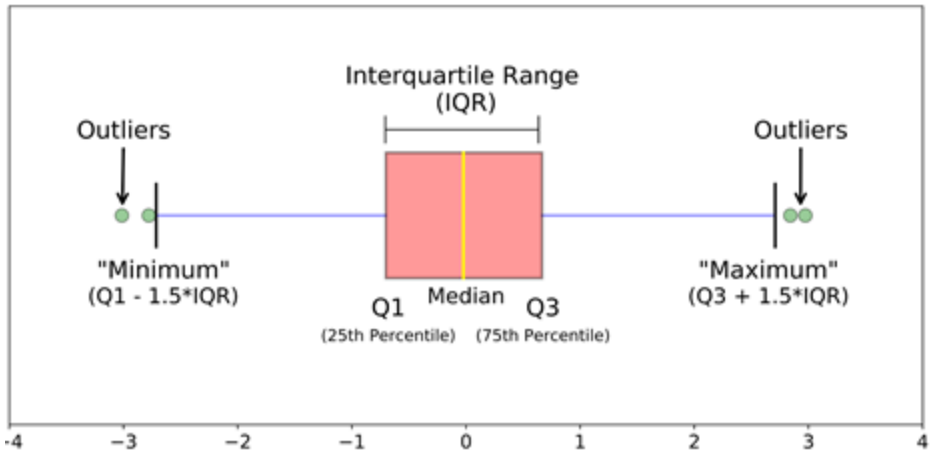
- The feature joined_through_referral has unindentified '?' values which are replaced with Nan
- The feature gender has unindentified 'Unknown' values which are replaced with Nan.
- The feature referral_id has unindentified 'xxxxxxxx' values which are replaced with Nan.
- The feature medium_of_operation has unindentified '?' values which are replaced with Nan.
- The feature days_since_last_login has unindentified '-999' values which are replaced with Nan.
- The feature avg_time_spent has negative values which are replaced with Nan
- The feature points_in_wallet has negative values which are replaced with Nan
- The feature avg_frequency_login_days has negative values as well as unidentified 'Error' values which are replaced with Nan

2.2.2 Feature Engineering

Feature engineering is the process of selecting, manipulating, and transforming raw data into features that can be used in supervised learning. In order to make machine learning work well on new tasks. We created new feature year from joining_date and dropped features such as security_no, referral_id because either they aren't valuable for predicting churn.

2.2.3 Outliers

One of the most important steps as part of data preprocessing is detecting and treating the outliers as they can negatively affect the statistical analysis and the training process of a machine learning algorithm resulting in lower accuracy. So we Detected outliers in the dataset using the Inter Quantile Range(IQR) as shown in Figure (3).



Figure(3) Detecting outliers using the Inter Quantile Range(IQR).

The outliers for each feature in the dataset are :

- Total number of Outliers in age are 0
- Total number of Outliers in avg_transaction_value are 1131
- Total number of Outliers in churn_risk_score are 0
- Total number of Outliers in avg_frequency_login_days are 4399
- Total number of Outliers in points_in_wallet are 2911
- Total number of Outliers in days_since_last_login are 0
- Total number of Outliers in avg_time_spent are 417
- Total number of Outliers in year are 0

So we made an algorithm for treating and removing the oultliers explained in the steps below:

- **Sort** the dataset in ascending order
- **calculate** the 1st and 3rd quartiles(Q1, Q3)
- **compute** IQR=Q3-Q1
- **compute** lower bound = (Q1–1.5*IQR), upper bound = (Q3+1.5*IQR)
- **loop** through the values of the dataset and check for those who fall below the lower bound and above the upper bound and mark them as outliers and remove them.

2.2.4 Categorical Data Encoding

The performance of a machine learning model not only depends on the model and the hyperparameters but also on how we process and feed different types of variables to the model. Since most machine learning models only accept numerical variables, preprocessing the categorical variables becomes a necessary step. We need to convert these categorical variables to numbers such that the model is able to understand and extract valuable information. For features such as membership_category and complaint_status we have used OrdinalEncoder because there is kind hierarchy in which we can order classes of these features .We have used dummy encoded features such as age, region, medium_of_operation, and internet_option etc

2.2.5 Feature Scaling

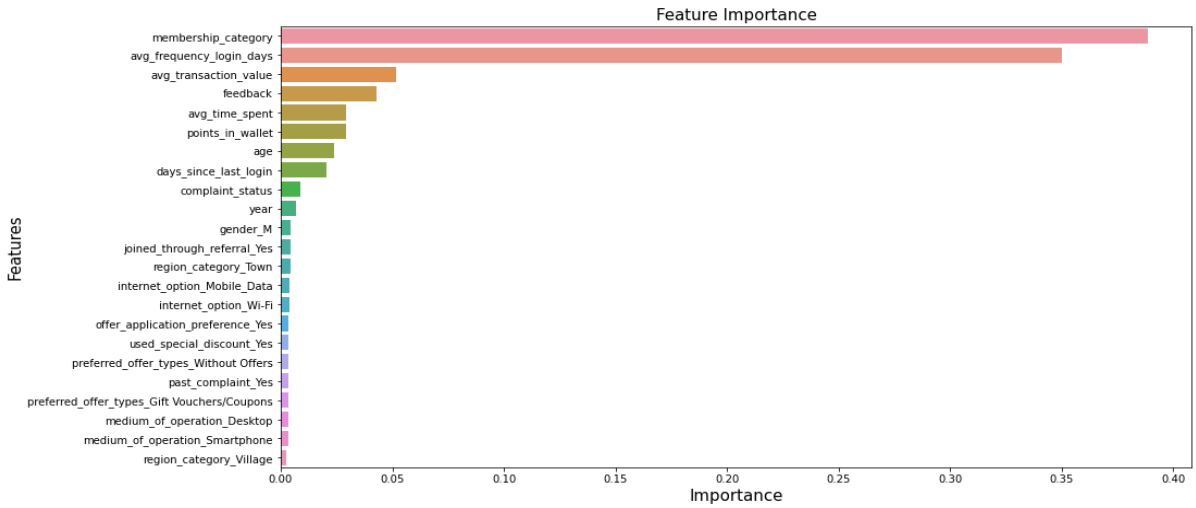
Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing to handle highly varying magnitudes or values or units. We applied **StandardScaler** function from **scikit-learn** library to Standardize only the numerical features by removing the mean and scaling to unit varianceand this done by Centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the training set. The result are showing in table (3).

	age	days_since_last_login	avg_time_spent	avg_transaction_value	avg_frequency_login_days	points_in_wallet
0	-1.209342	0.756268	0.098431	1.566300	1.032977	0.435508
1	-0.326297	0.573495	-0.849870	-0.883072	-1.090570	0.465956
2	0.430599	0.207949	0.775788	-0.383731	-1.992103	1.584823
3	-0.010923	-0.340370	-1.391756	-0.126849	-1.271298	-0.883541
4	-0.389371	1.304586	-0.037041	-0.172944	-0.244497	-0.564337

Table(3)Results for Standardizing the numerical features in the dataset.

2.2.6 Feature Selection using Random Forest

Feature selection is the process of reducing the number of input variables when developing a predictive model. It is desirable to reduce the number of input variables to both reduce the computational cost of modeling and, in some cases, to improve the performance of the model. Random Forests are often used for feature selection in a data science workflow. The reason is because the tree-based strategies used by random forests naturally ranks by how well they improve the purity of the node. Nodes with the greatest decrease in impurity happen at the start of the trees, while notes with the least decrease in impurity occur at the end of trees. Thus, by pruning trees below a particular node, we can create a subset of the most important features. We identified important features using Random Forest and dropping inconsistent features. Figure(4) showing the results for applying Random Forest and the important for each feature.



Figure(4) importance for each feature in the dataset

3. Machine learning models

In the following, five well casted and popular techniques used for churn prediction has been presented succinctly, under the canopy of facts considered such as reliability, efficiency, and popularity in the research community most popular predictive models have been applied, namely, logistic regression, naive bayes, K-nearest neighbors (KNN), random forest and AdaBoost.

3.1 Logistic Regression

Regression analysis is a statistical process for estimating the relationships among variables. It includes many techniques for modeling and analyzing several variables. In terms of customer churning, regression analysis is not widely used, and that is because linear regression models are useful for the prediction of continuous values. On the other hand, Logistic Regression is a type of probabilistic statistical classification model. It is also used to produce a binary prediction of a categorical variable (e.g customer churn) which depends on one or more predictor variables (customers' features) In the churn prediction problem, and is usually used after proper data transformation is applied on initial data, with quite good performance [9,5].

3.2 Naive Bayes

Naive Bayes classifier is a probabilistic approach in which each vector feature is considered as independent of each other. Naive Bayesian classifiers assume that the value of each feature has an independent influence on a given class, and this assumption is called class conditional independence that is used to simplify the computation. In simple terms that this classifier assumes that the presence of feature vector (customer churn) is independent from the other feature vectors that are present in the class.

3.3 K-nearest neighbors (KNN)

The k-nearest neighbors algorithm, also known as KNN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

3.4 Random Forest

Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. It works on the divide and conquer approach. It is based on the random subspace method [12]. In this method a number of trees are formed and each decision tree is trained by selecting any random sample of attributes from the predictor attributes set.

Each tree matures up to maximum extent based on the attributes or parameters present. The final decision tree is formed for the prediction mainly based on weighted averages. It has the ability to handle thousands of input parameters without deletion. It can also handle the missing values inside the data-set for training the predictive model.

3.5 AdaBoost

Ada – boost like Random Forest Classifier is another ensemble classifier. (Ensemble classifier are made up of multiple classifier algorithms and whose output is combined result of output of those classifier algorithms). A single algorithm may perform poorly in classification of the objects. But when combined with boosting ensemble algorithms like Ada-boost and selection of training set at every iteration and assigning right amount of weight in final voting, we can obtain good accuracy score for overall classifier.

Hyperparameter Tuning with GridSearchCV

GridSearchCV validation has been used over the training set for hyperparameter tuning and to prevent overfitting of models**.** GridSearchCV is a function that comes in Scikit-learn library. This function helps to loop through predefined hyperparameters and fit our models on the training set. So, we can select the best parameters from the listed hyperparameters. We applied the GridSearchCV fuction to models so we can find the best parameters of each model on the training set and the results are showing below:

- Best parameters for **KNN**: 'number of neighbors=11'
- Best parameters for **Random Forest** :{'criterion': 'gini', 'max_depth': 3, 'max_features': 'sqrt', 'min_samples_split': 2, 'n_estimators': 300}
- Best parameters for **AdaBoost** :{'learning_rate': 0.01, 'n_estimators': 250}

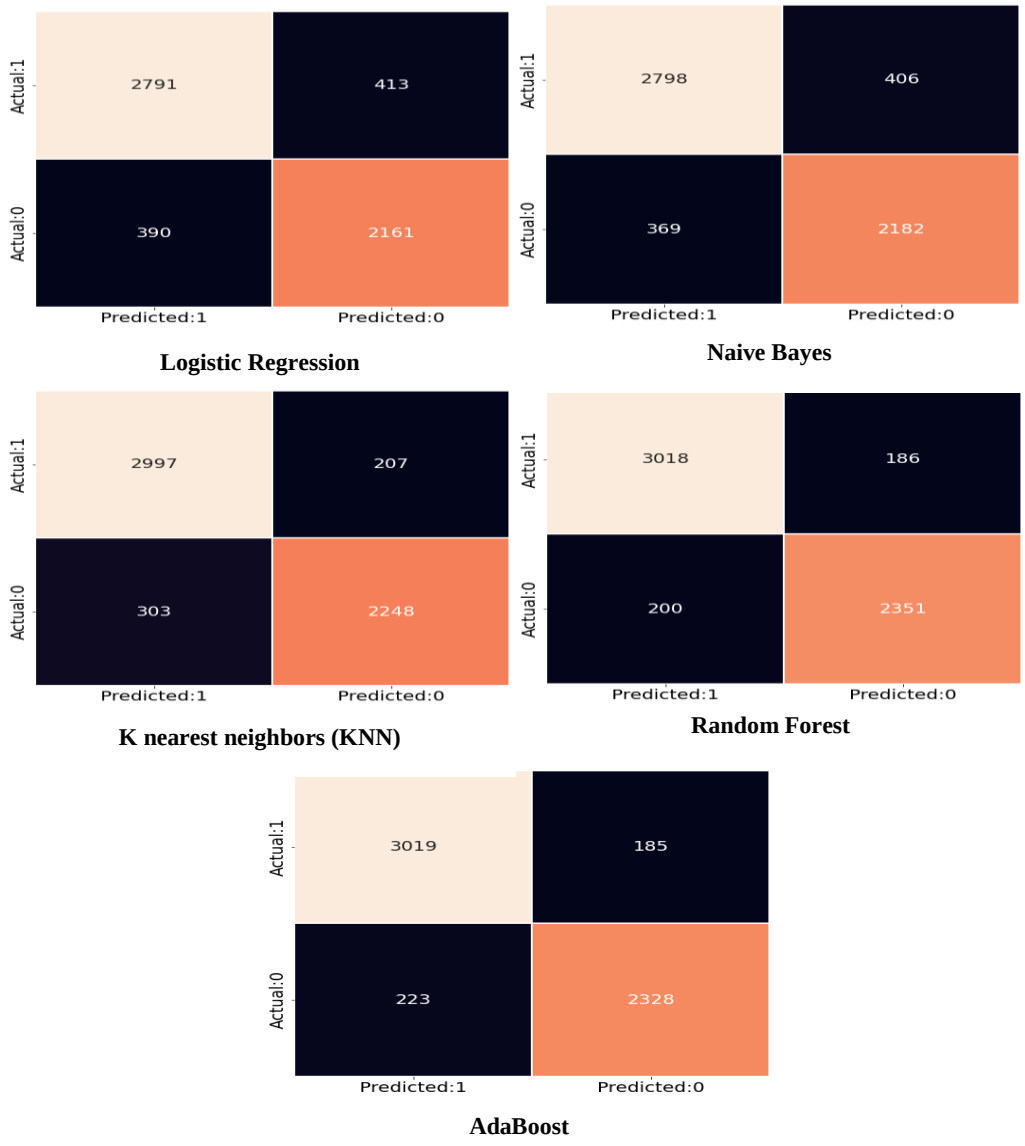
4. Performance analysis

4.1 Confusion matrix

To evaluate the performance of applied models or throughput of Customer Churn Prediction on the test set, different metrics have been used, namely, precision, recall, accuracy and F -measure [8]. It measures the ability of the predictive models for forecasting the churning customers correctly [7]. The aforementioned four measures are calculated from the information captured using confusion matrix. The representation of confusion matrix is shown in Table 4. True positive and false positive are denoted as Tp and Fp, whereas, false negative and true negative as Fn and Tn. The Confusion matrix of each model are shown in figure(5).

	Prediction category	
	Churners	con-churners
churn	Tp	Fn
Non-churn	Fp	Tn

Table (4) Confusion matrix forevaluation of classifier



Figure(5) Confusion matrix for the models.

4.1.1 Performance indicators

4.1.1.1 Accuracy

It is ration of number of all correct predictions, and is calculated under the following:

$$Accuracy = \frac{(T_{p+}T_n)}{(T_p + F_p + F_n + Tn)}$$

4.1.1.2 Recall

It is the ratio of real churners (i.e. True Positive), and is calculated under the following:

$$Recall = \frac{T_p}{(T_p + F_n)}$$

4.1.1.3 Precision

It is the ratio correct predicted churners, and is calculated under the following:

$$\text{Precision} = \frac{T_p}{(T_p + F_p)}$$

4.1.1.4 F – measure

It is the harmonic average of precision and recall, and it is calculated under the following:

$$F - measure = \frac{(2 * Precision * Recall)}{(Precision + Recall)}$$

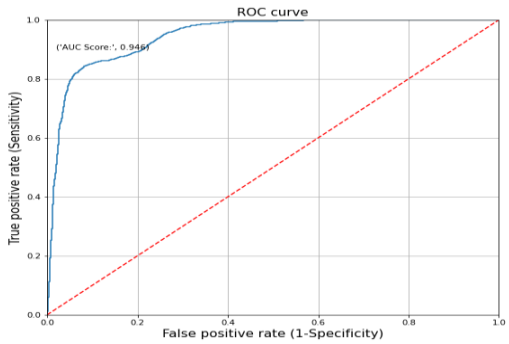
We made a Classification Report that includes the Performance indicators for each model as shown in table (5).

Model/Indicator		precision	recall	f1-score
Logistic Regression	0	0.84	0.85	0.84
	1	0.88	0.87	0.87
Naive Bayes	0	0.84	0.86	0.85
	1	0.88	0.87	0.88
(KNN)	0	0.92	0.88	0.90
	1	0.91	0.94	0.92
Random Forest	0	0.93	0.92	0.92
	1	0.94	0.94	0.94
AdaBoost	0	0.93	0.91	0.92
	1	0.93	0.94	0.94

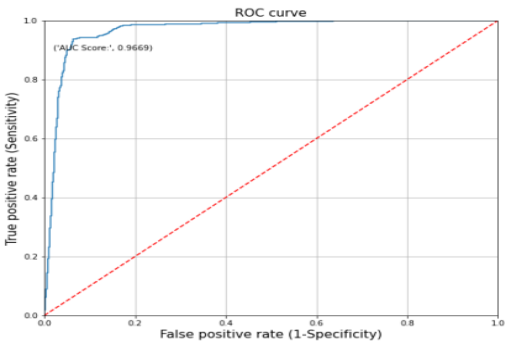
table (5) Classification Report for each model

4.2 AUC curve analysis

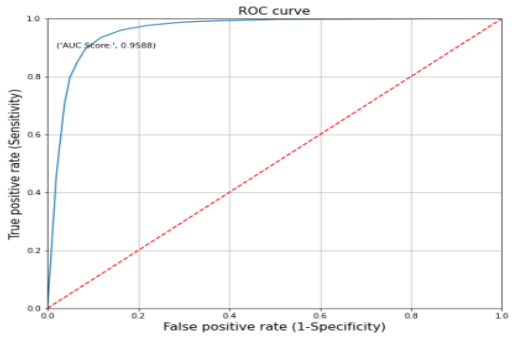
AUC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability . It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. Figure (6) graphically represents the obtained AUC scores of logistic regression, naive bayes, K-nearest neighbors (KNN), random forest and AdaBoost.



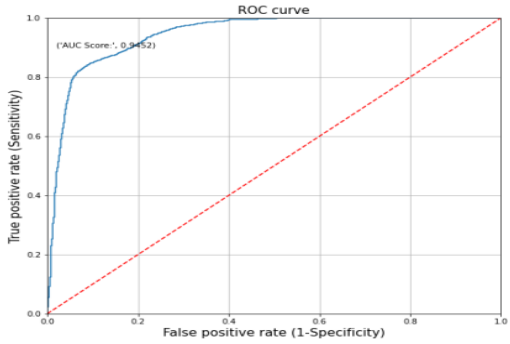
Logistic Regression



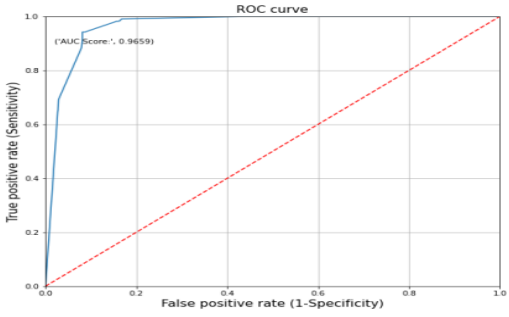
Naive Bayes



K nearest neighbors (KNN)



Random Forest



AdaBoost

Figure (6) graphically represents the obtained AUC scores of logistic regression, naive bayes, K-nearest neighbors (KNN), random forest and AdaBoost.

5. Model Performance

Table(6) shows the performance of each model on the training and the testing set. it is obvious that Random Forest and AdaBoost have the higher accuracy on the testing set among the others.

Model	Train Accuracy	Test Accuracy	Test f1-Score
Logistic Regression	0.857894	0.860469	0.858777
Naive Bayes	0.862977	0.865334	0.863774
KNN	0.926883	0.911381	0.909854
Random Forest	0.933313	0.932928	0.932015
AdaBoost	0.930272	0.929105	0.928068

table (6) Classification Report for each model

Conclusion

With advancement of technology, there comes an increase in services and it is hard for a company to predict the customers who are likely to leave their services. churn prediction is a problem which has gathered attraction by various researchers in the recent years. Through this technical report we provide a comparative study of Customer Churn prediction using famous machine learning techniques such as Logistic Regression, Naïve Bayes, Random Forest, KNN and AdaBoost Classifier. The experimental results show that two ensemble learning techniques that is Adaboost classifier and Random Forest classifier gives maximum accuracy with respect to other models. They outperformed other algorithms in terms of all the performance measures such as accuracy, precision, F-measure, recall and AUC score. Churn prediction for a company tends to be a very tedious task and as of many upcoming company's and startups there is a tough competition in the market to retain the customers by providing services that are beneficial to both sides. In future, with the upcoming concepts and frameworks in the field of reinforcement learning and deep learning sector, machine learning is proving to be one of the most efficient way to address problems like churn prediction with better accuracy and precision.

Reference

[1] Abbasimehr H, Setak M, Tarokh M (2011) A neuro-fuzzy classifier for customer churn prediction. International Journal of Computer Applications 19(8):35–41

[2] Adwan O, Faris H, Jaradat K, Harfoushi O, Ghatasheh N (2014) Predicting customer churn in telecom industry using multilayer preceptron neural networks: Modeling and analysis. Life Science Journal 11(3):75–81

[3] Ahmad AK, Jafar A, Aljoumaa K (2019) Customer churn prediction in telecom using machine learning in big data platform. Journal of Big Data 6(1):28

[4]Umayaparvathi V, Iyakutti K. A survey on customer churn prediction in telecom industry: datasets, methods and metric. Int Res J Eng Technol. 2016;3(4):1065–70.

[5] Yu W, Jutla DN, Sivakumar SC. A churn-strategy alignment model for managers in mobile telecom. In: Communication networks and services research conference, vol. 3. 2005. p. 48–53.

[6] Sß. Gürsoy, U. Tug˘ba, Customer churn analysis in telecommunication sector, J. School Bus. Admin. Istanbul Univ. 39 (1) (2010) 35–49.

[7] Coussement K, De Bock KW (2013) Customer churn prediction in the online gambling industry: The beneficial effect of ensemble learning. Journal of Business Research 66(9):1629–1636

[8] Coussement K, Van den Poel D (2008) Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. Expert systems with applications 34(1):313–327

[9] Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier,

[10] Sokolova, M., Japkowicz, N., Szpakowicz, S.: Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In: Australasian joint conference on artificial intelligence, pp. 1015–1021. Springer (2006)

[11] Wei CP, Chiu IT (2002) Turning telecommunications call details to churn prediction: a data mining approach. Expert systems with applications 23(2):103–112

[12] Yu, W., Jutla, D.N., Sivakumar, S.C.: A churn-strategy alignment model for managers in mobile telecom. In: 3rd Annual Communication Networks and Services Research Conference (CNSR'05), pp. 48–53. IEEE (2005)