Journal of Advanced College of Engineering and Management, Vol. 5, 2019

CUSTOMER CHURN PREDICTION FOR IMBALANCED CLASS DISTRIBUTION OF DATA IN BUSINESS SECTOR

Aayush Bhattarai[1], Elisha Shrestha[2], Ram Prasad Sapkota[3]

Advanced College of Engineering and Management, Kupondole, Lalitpur, Nepal

[1]Email Address: me.aayus@gmail.com [2]Email Address : elisha.shrestha32@gmail.com

[3]Email Address: ram.sapkota@acem.edu.np

Abstract

Churners are those people who are about to transfer their business to a competitor or simply who cancel a subscription to a service. This paper is based on a specific business sector, which is telecommunication sector. With a churn rate of 30%, the telecommunication sector takes the first place on the list. In this paper, we present some advanced data mining methodologies which predicts customer churn in the pre-paid mobile telecommunications industry using a call detail records dataset. To implement the predictive models, we initially propose and then apply four machine learning algorithms: Random Forest, Naïve Bayes, Logistic Regression, and XGBoost. To evaluate the models, we use various evaluation metrics and find the best model which will be suitable for any class imbalance ddata and also our business case. This paper can also be viewed as a comparative study on the most popular machine learning methods applied to the challenging problem of customer churn prediction.

Keywords: *Churn Prediction; Machine Learning; Business Intelligence; CRM; Telecommunication; Class Imbalance Problem; Naïve Bayes; Logistic Regression; XGBoost; Random Forest; Model Selection*

1. Introduction

Customer churn refers to when a customer switches from one service provider to another. Since the cost of winning a new customer is far greater than the cost of retaining an existing one, churn prediction has become an important and crucial topic for any business sectors. Churn is a problem for any provider of a subscription service or a business company, e.g. telecommunications, banking and insurance, retail market, etc.

The focus of this paper is mainly on telecom industry because of its tremendous growth in the recent years. With easy communication and a number of service providers almost everyone today has a telecom subscription. Moreover, changing of mobile numbers is not an obstacle which makes customers of telecom companies easier to churn. Churn Prediction model can help analyze the historical data available with the business to find the list of customers which are at high risk to churn. This will help the telecom industry to focus on a specific group rather than using retention strategies on every customer. Customer retention for individual customer is difficult because for large customer base companies cannot afford to spend much time and money for it. However, customer retention can be easier if we could predict in advance which customers are at risk of leaving, by directing them solely towards such customers.

This is where the churn prediction model can help the business to identify such high-risk customers and thereby helps in maintaining the existing customer base and increase in revenues. Churn prediction is also important because of the fact that acquiring new customers is much costly than retaining the existing one. As the telecom users are billions in number even a small fraction of churn leads to high loss of revenue. Retention has become crucial especially in the present situation because of the increasing number of service providers and the competition between them, where everyone is trying to attract new customers and lure them to switch to their service.

Churn prediction, in our case, is a two-class classification problem i.e. the churn result is either yes or no. The major problem in this case is that the number of customers who do not churn are much greater than the customers who churn. This is known as class imbalance problem. We will use four predictive models that will help us find the pattern among already churned customers based on the historical data. The best model after the evaluation will not only be suitable for churn prediction in telecommunication, but any churn problem in business sector because the historical data that is used as a training set will always have a class imbalance problem. The results obtained will provide useful insight which can then be used strategically to retain customers.

2. Literature Review

A lot of research has been done in the field of CRM (Customer Relationship Management) in various industries for retention of customers and develop strategies to build an efficient model so that specific group of customers can be targeted for retention. Various data mining and statistical techniques have been used for churn prediction of which some famous techniques include Decision trees, Regression Models, Neural Networks, Clustering, Bayesian Models, Support Vector Machine, etc.

Cimpoeru and Anca et al.,[1] discovered a way to classify clusters and find out predictions through commercial ways in a relational database management system. The researchers have identified two models of predicting customers who have a high possibility to leave. They are 'decision tree' and 'naïve Bayes classification' models.

Sharma et al.,[2] proposed a Neural Network (NN) - based approach to predict customer churn in subscription of cellular wireless services. Furthermore, it was found that when different neural network's topologies were experimented medium- sized NNs performance is essential for the customer churn prediction.

Yaya Xie et al, [3] proposed Improved Balanced Random Forests (IBRF) based churn prediction. This approach integrates sampling techniques and cost-sensitive learning with random forests to predict churn. However, the time varying variables are not employed in prediction causing limitations in performance.

3. Data Analysis

The dataset used in this paper consists of call detail record sand is obtained from theUCI repository of Machine Learning [4]. It contains information about the usage of a mobile telecommunication system and has a total number of 3333 customers with 15 continuous and 5 discrete variables each, and the Churn dependent variable with two classes *Yes/No*. Three discrete variables (*State*, *Areacode*, and *Phone*) are omitted due to inconsistent information contained [5]. For each of these customers we can find information about their corresponding inbound/outbound calls count, inbound/outbound SMS count, and voice mail.

The visualization is done using various libraries of python like *Pandas*a nd *Matplotlib*. Sometimes data does not make sense until we look at it in a visual form, such as with charts and plots. This can be useful when exploring data, identifying patterns, detecting outliers and much more. The distribution of the features in our data is shown in figure 1.

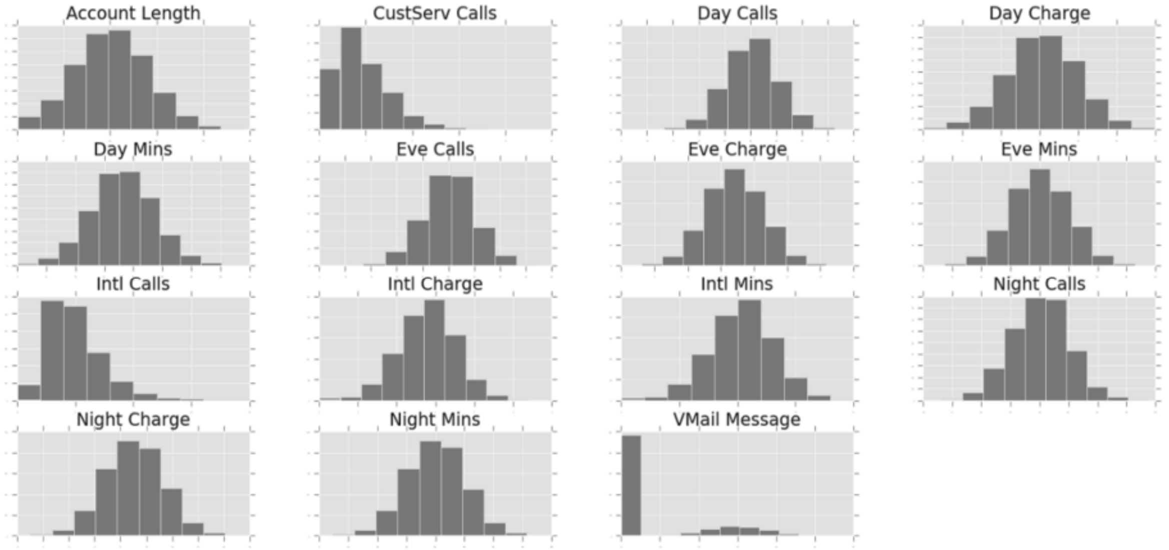


Fig1 Data distribution

We can see that there are a lot of Gaussian-like distribution and perhaps some exponential distribution for other attributes. For example, if we take into account, customer service calls (*CustServCalls*), we can clearly see that the frequency of few calls to the customer service center is greater i.e. even if there was some problem, it was sort out in a fewer number of calls for maximum people. But some of the customers are making more number of calls to sort out their problem. So, we can initially assume that these people, who are calling frequently, might have a higher probability to churn. But the decision cannot be based on just one attribute because there are many other factors like day calls, night calls, call durations and other features that should be taken into account. This is where the machine learning algorithms play their role to find out the patterns of churn by going through each attribute in the dataset to provide the accurate result.

3.1 Correlation

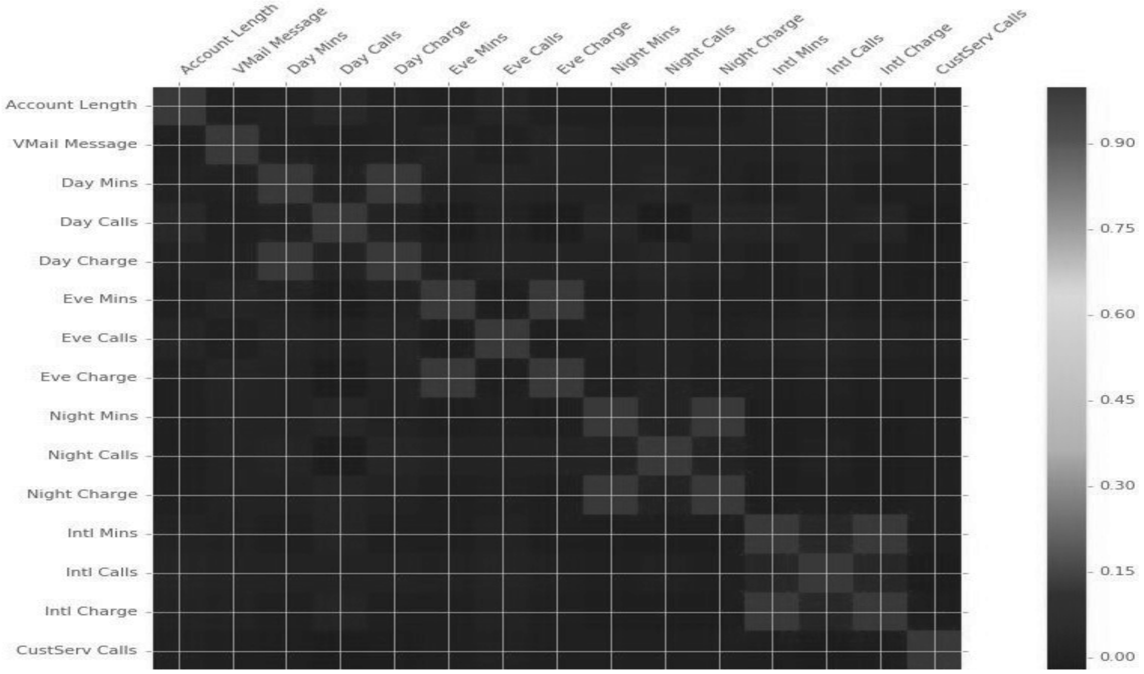


Fig 2 Correlation matrix among data attributes

The figure above is generated using *matplotlib* in python 3.6and we can visualize the correlation among various features of the data with this correlation matrix. Correlation gives an indication of how related the changes are between two variables. We can calculate the correlation between each pair of attributes. This is useful because some machine learning algorithms like logistic regression can have poor performance if they are highly correlated to each other. In our case, we can see that features like *DayCharge* and *Day Mins* are positively correlated which is obvious because the charge increases as calls increase. On the other hand, attributes like *Day Calls and Eve Calls, Day Calls* and *Night Calls*are negatively correlated.

4. Methodology

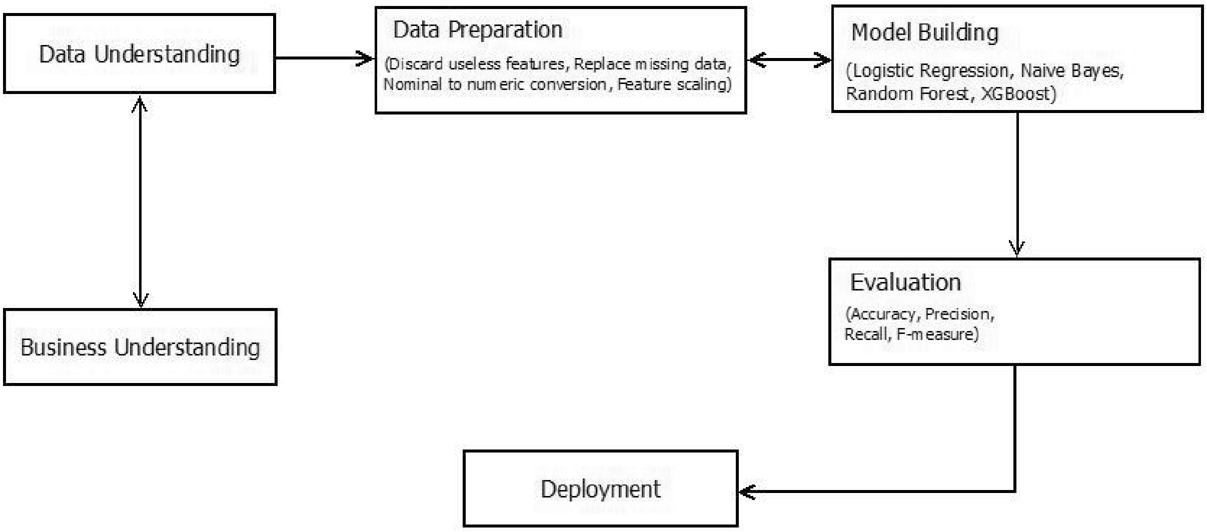

Data Understanding
Data Preparation
(Discard useless features, Replace missing data, Nominal to numeric conversion, Feature scaling)
Model Building
(Logistic Regression, Naive Bayes, Random Forest, XGBoost)
Business Understanding
Evaluation
(Accuracy, Precision, Recall, F-measure)
Deployment

Fig 3 Block diagram of the system

The first phase after the selection of data is data preparation. The data had various missing parameters which were replaced by taking the mean of entire particular column. Similarly, some features that did not contribute to churn were also excluded. The data had numeric as well as categorical attributes. Since all of the tasks were done in python, the compiler would not understand the categorical features. Hence it was converted to numeric form and finally all the data were normalized to avoid problems like multico-linearity and false predictions.

After the data is prepared, it is then split into two sets; training set and test set. The training set is used to build the model and the performance of the model is tested on the test set.

Four machine learning models namely, Logistic Regression, Naïve Bayes, Random Forest and Extreme Gradient Boosting (XGBoost) will be built from the training set. The models will have different prediction results and the performance of each model will be evaluated using various performance metrics. We will compare the models on the basis of different measures to select the best one for our business case which will eventually be suitable for most class imbalance data.

5. Machine Learning Approaches for Churn Prediction

In our approach we will use four machine learning algorithms that are described as follows:

5.1 Logistic Regression

Logistic Regression is a very popular statistical algorithm, widely used when the dependent variable is dichotomous. In the telecom data set, the variables are dichotomous since they represent the status of a given customer. More specifically, they highlight the probability of a subscriber to churn in the future. LR can estimate the probability of an event to take place. Below is the mathematical formula for this model:

$$P(Y/X) = \frac{1}{1+e^{Y}}$$

Given the variable X, the conditional probability of the event Y is $P(Y/X)$ where Y is a linear function of the independent input variables:

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \cdots$$

The values of (b_0, b_1, b_2, b_3) are obtained using the maximum likelihood method. Logistic Regression can be easily applied after data preprocessing and cleaning leading to a quite good performance [6], also this method is highly effective for producing a binary classification (in our case for predicting if the customer will churn or not).

5.2 Naïve Bayes

Naïve Bayes is a supervised learning algorithm that applies the theorem of Bayes' assuming "naively" that every pair of features is statistically independent.

Given a set of features X = ($X_{1,}$ $X_{2,}$.......,X_n), the conditional probability for each Class C_j is written as follows:

$$P(C_j|X) = P(X|C_j)\,P(C_j)/P(X)$$

The NB classifier can improve significantly the prediction rates compared to the decision tree [7] and it is widely used in the telecom industry.

5.3 Ensemble Methods

Ensemble methods are learning algorithms that construct a set of classifiers and then classify new data points by taking a (weighted)vote of their predictions [8]. They are the combination of multiple and diverse models. Each model in the ensemble makes the prediction and the final prediction is determined by the majority vote among the models. We have used two ensemble algorithms in this paper: Random Forest and XGBoost.

5.3.1 Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees [14]. The optimal decision tree is selected amongst the many and hence it can be a better approach than decision tree alone.

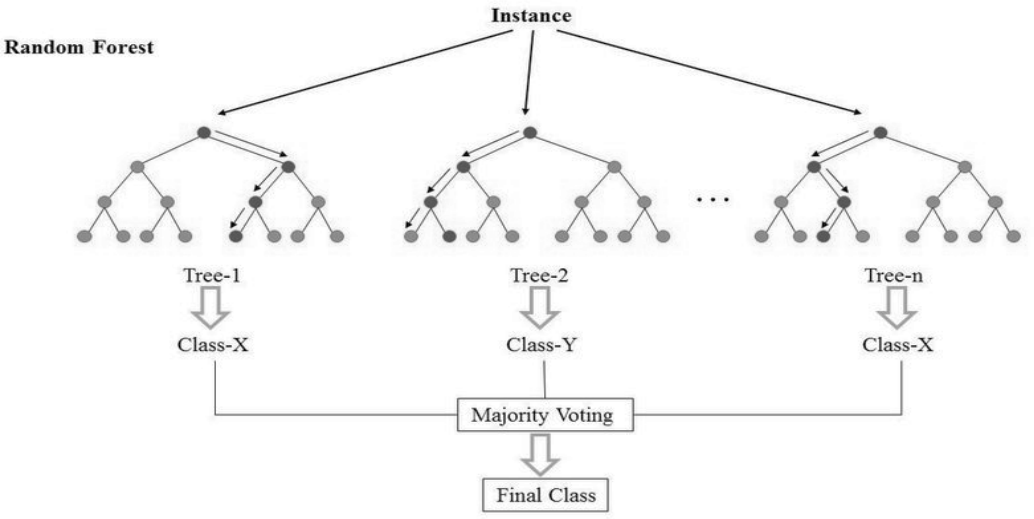


Fig 4 Random forest

Unlike other algorithms Random Forest approach solves the problem of over fitting and can be a very good competitor among other approaches for the challenging problem of churn prediction.

5.3.2 XGBoost

XGBoost is a scalable machine learning system for tree boosting that proposed by Chen in 2016 [9]. Among many winning solutions in the machine learning competitions in Kaggle, majority of the algorithm used is always XGBoost since 2016.Gradient boosting is the original model of XGBoost, combining weak base learning models into a stronger learner in an iterative fashion [11]. At each iteration of gradient boosting, the residual will be used to correct the previous predictor that the specified loss function can be optimized. Asan improvement, regularization is added to the loss function to establish the objective function in XGBoost measuring themodel performance, which is given by

$$J(\theta) = L(\theta) + \Omega(\theta)$$

The parameters trained from given data are denoted asθ.L isthe training loss function, such as square loss or logistic loss,which measures how well the model fits on training data. Ω is the regularization term, such as L1 norm or L2 norm, which measures the complexity of the model. Simpler models tend to have better performance against over fitting. Since the base model is decision tree, the output of model is voted or averaged by a collection of k trees [10].

6. Evaluation

In this paper, we consider accuracy, precision, recall, and F-measure as the methods of evaluation to examine the performance of different prediction models. Table 1 shows the confusion matrix in order to calculate these evaluation measures.

Accuracy

It is the proportion of the total number of predictions that were correct and is calculated from the equation

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision

Table 1 Confusion Matrix

		Predicted Class	
		Churner	Non-churner
Actual class	Churner	TP	FN
	Non-churner	FP	TN

It is the proportion of the predicted positive cases that were correct and is calculated from the equation,

$$Precision = \frac{TP}{TP\ +\ FP}$$

Recall (or Sensitivity)

It is the proportion of positive cases that were correctly identified and is calculated from the equation,

$$Recall = \frac{TP}{TP\ +\ FN}$$

F-measure

Precision or recall alone cannot describe the efficiency of a classifier since good performance in one of those indices does not necessarily imply good performance on the other. For this reason, F-measure, a popular combination is commonly used as a single metric for evaluating classifier performance. F-measure is defined as the harmonic mean of precision and recall. A value closer to one implies that a better combined precision and recall is achieved by the classifier [13].

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Here TP, TN, FP and FN are the True Positive, True Negative, False Positive and False Negative respectively.

7. Result and Discussion

In this work, the machine learning models are trained on python 3.6.3 with several scientific computing libraries, such as NumPy 1.13.3 and pandas 0.20.3, which provides efficient data structures and preprocessing methods. Besides, Scikitlearn0.19.1 and XGBoost 0.71 are imported to support all the learning models [12].

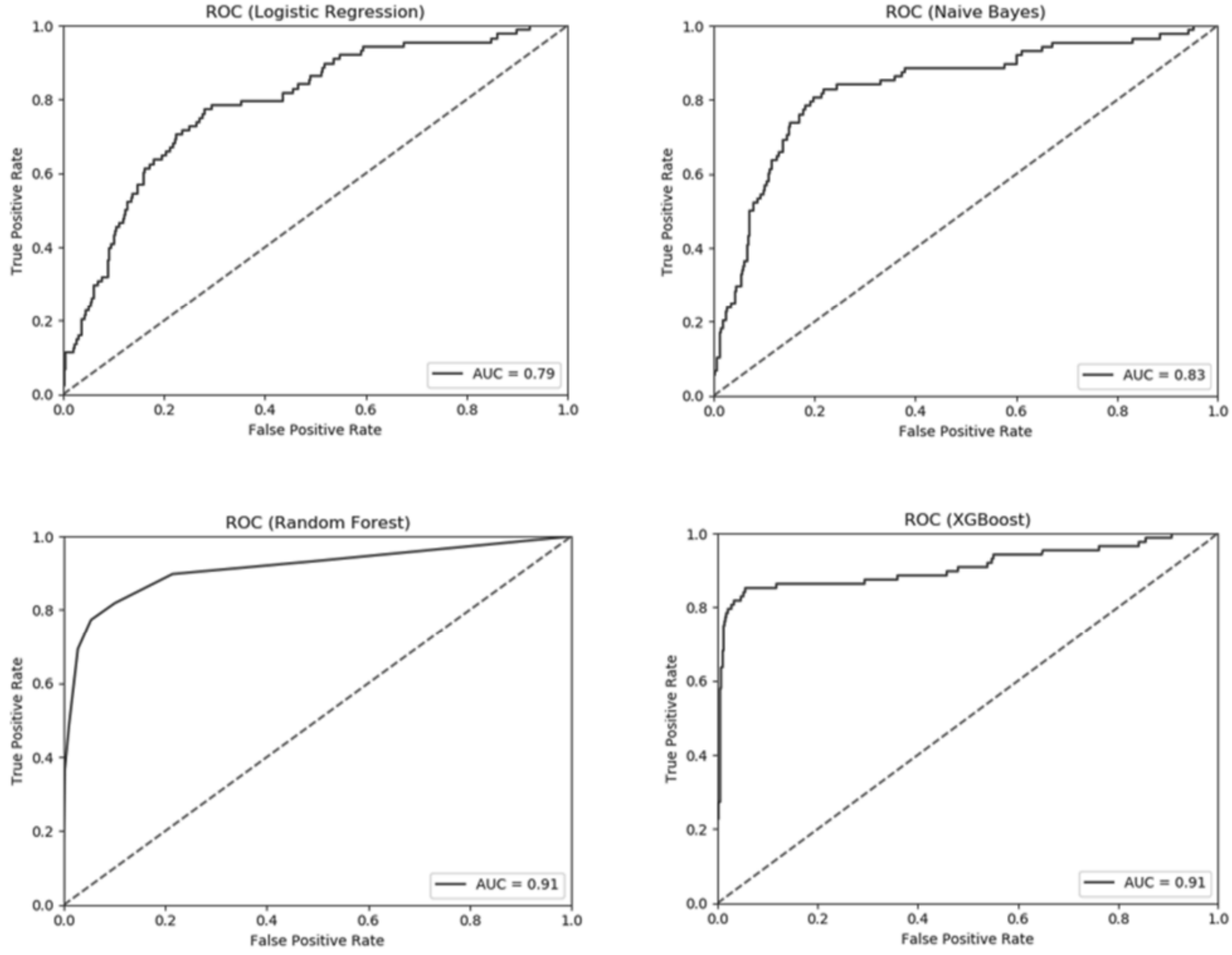
We have tested all the proposed machine learning algorithms on the original dataset by dividing it into training and test sets. The predicted test set results were compared with the original data and the results obtained are shown in Table 2.

Table 2 Performance of Different Classifiers

	Logistic Regression	Naïve Bayes	Random Forest	XGBoost
Accuracy	0.857	0.851	0.923	0.955
Precision	0.425	0.450	0.877	0.881
Recall	0.227	0.568	0.488	0.761
F-measure	0.296	0.502	0.627	0.817

The accuracy of all four classifiers is above 85%, which is a very good result. But if we look at some other metrics like precision and recall, there are some classifiers that have performed better. Precision is basically the accuracy of positive predictions (i.e. churn = yes in our case). So we must be more focused on precision because misprediction of churners is far more costly than misprediction of non-churners. But to get the best classifier, we must look at every performance metrics. Logistic Regression obtained the lowest accuracy and lowest overall performance, but 85.7% accuracy is still good. Naïve Bayes (NB) has better recall than Random Forest (RF), but other than that RF is superior than NB. XGBoost have obtained the highest accuracy of 95.5% and superior precision, recall and F-measure in comparison to all the others. Besides, in terms of time needed for training, performances of machine al learning algorithms are similar, with exception of the Logistic regression which is few times slower due to its iterative nature.

Receiver Operating Characteristics (ROC) curve


ROC (Logistic Regression)

ROC (Naive Bayes)

ROC (Random Forest)

ROC (XGBoost)

Fig 5 ROC curves of four classifiers

Figure 5 illustrates the ROC curves corresponding to four predictive models. Once again, the Logistic Regression with the AUC (Area Under the Curve) of 0.79, have shown lower performance in comparison to others. Naïve Bayes has a modest AUC of 0.83. Following the curves, one can observe that the ensemble methods provide the best thresholds for separating samples into appropriate classes with the best AUC of 0.91 which is equal for both Random Forest and XGBoost. But still if we have to select just one model, it would be XGBoost because of its high execution speed, performance and its ability to maintain the speed for large data.

8. Conclusion

In this paper, we explained four methodologies for building the classifier models that will predict customer churn for our business case i.e. in telecommunication sector. The prediction results can be used to build strategies and policies for customer retention targeting the high-risk customers. Furthermore, we observed that for predicting both churners and non-churners, the models have an overall accuracy of 85.7% for LR, 85.1% for NB, 92.3% for Random Forest and 95.5% for XGBoost. In our case, XGBoost stood out to be the best one all in terms of accuracy, AUC and execution speed. There is no 100% accuracy because the accuracy is limited by the problem itself, in the sense that there is no 100% correlation between information of customer and their decision to churn. So, 95.5% is very close to the best accuracy for this business problem.

We also found that ensemble methods like Random Forest and XGBoosthave better performancefor imbalanced class distributions of data just like in our case, which is a challenging problem in machine learning. So these methods can be applied to any business problem for churn prediction because churn analysis in any sector will always have class imbalance problem.

9. Recommendations

1. Data can be explored in depth to know the underlying reasons behind customer churn
2. Social network analysis and text mining techniques can be applied in conjunction with these methodologies to reduce the churn rate even more.
3. Artificial Neural Networks can be tested for this problem because they have proved to be great in many predictive applications.

References

1. Cimpoeru, C. & Andreescu, A., *"Predicting Customers Churn in a Relational Database."* Informatica Economica,Vol. 18(3), 5-16, (2014)
2. Sharma, A.,"*A Neural Network based Approach for Predicting Customer Churn in Cellular Network Services.*", International Journal of Computer Applications, Vol. 27(11), 0975 – 8887, (2011)
3. Xie, Y., Li, X., E. W. T., Ngai, and Ying, W., "*Customer churn prediction using improved balanced random forests.*" Expert Systems with Applications, Vol.36(3), 5445-5449, (2009).
4. "Teleco churn dataset," [online] Available: https://archivRe.ics.uci.edu/ml/datasets.html. [Accessed 28 February 2018]
5. Brandusoiu, I. B. andToderean, G., "*Churn prediction in the telecommunications sector using support vector machines.*" Annals of theOradea University Fascicle of Management and TechnologicalEngineering, Vol. 22(1), 19-22,(2013)
6. Tugba,U. &Gursoy, S.,"*Customer churn analysis in telecommunication sector.*" Istanbul University Journal of the School of Business, Vol. 39(1), 35-49, (2010)
7. Kirui, C., Hong, L., Cheruiyot, W.&Kirui, H., "*Predicting customer churn in mobile telephony industry using probabilistic classifiers in data mining.*" IJCSI International Journal of Computer Science Issues, Vol. 10(2), 1694-0814, (2013)
8. Kittler, J., Roli, F., "*Ensemble methods in machine learning."* 1(3), pp 1-15 (2000)

9. Chen, Q. &Guestrin, C., "*XGBoost: a scalable tree boosting system*."Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785-794, (2016)

10. Zhang, D., Qian, L., Mao, B., Huang, C. &Si, Y., "*A Data-Driven Design for Fault Detection of Wind Turbines Using Random Forests and XGBoost*", IEEE Access, Vol.6, 21020-21031, (2018)

11. Friedman, J. H., "*Greedy function approximation: a gradient boosting machine.*" Annals of Statistics, pp. 1189-1232, (2001)

12. Pedregosa, F. &Varoquaux, G., "*Scikit-learn: machine learning in python.*" Journal of Machine Learning Research, vol. 12, pp. 2825-2830,(2011).

13. Han, J., &Kamber, M., "*Data mining: Concepts and techniques*." San Francisco: Morgan Kaufmann Publishers, (2001)

14. https://en.wikipedia.org/wiki/Random_forest, June 20, 2018