STAT 441/841 Final Project Report

Fall 2018

Group #8

Presented by:

Jiayue Zhang
Lingyun Yi
Rongrong Su
Siao Chen

Introduction

Traditional telecommunication industry is made up of telecommunication companies and Internet service providers, which play important role in daily life. According to Insight Research, the market revenue is expected to grow to $2.4 trillion in 2019. [1] With technology evolution, information technology giants such as Google and Facebook are entering the telecommunication market, and offering different ways of information exchange. The traditional telecommunication companies are facing huge pressure. It is crucial for the telecommunication companies to analyze and maintain their relationship with existing customers, as well as winning new customers with marketing strategies. In fact, it costs 5 times as much to attract a new customer than to keep an existing one. [2] Therefore, to retain existing customers and build a loyal relationship is the key concerns for traditional telecommunication companies to compete with information technology giants.

Project Motivation

As mentioned in Social Network Analysis in Telecommunications by Carlos Andre Reis Pinheiro, actions such as providing innovative bundles and a range of pricing discounts are based on customer behavior and individual values to the corporation. [3] This project, is aim to provide insights for the company in predicting the chance of a customer leaving the company.

Data Preprocessing

1. Data transformation

The original dataset is obtained from a Chinese Machine Learning competition held by a local telecommunication company. It has about 43 GB, including about 9 million users with 2 billion call records in a 3-month duration.
Since the original dataset is about call records, it is hard to see a specific user behaviour. Preprocessing was done to combine all the call records by caller ID. Data transformation was done to reduce the dimensions of the original dataset. To be specific, for each caller ID, the number of callers contacted (callersum), total number of calling records (recordsum), and calling duration (timesum) was calculated by month.
The correlations between any two of the variables can be found in Appendix (1).

2. Data Cleansing

(1) Maintain the Validity of the Dataset
It was found that for caller ID ‘8503722’, ‘12243495’ and ‘12881153’, conflicting information was recorded. So, the observations were removed from the analysis.

(2) Maintain the Completeness of the Dataset
Since missing values are incompatible with most of the pre-implemented classification models, and that the original dataset is large enough to provide necessary information, the data points with missing feature values were not included in running the classification model. However, handling missing values will be discussed in the later section of the project.
A statistical summary for the after-transformed dataset can be found in Appendix (2).

(3) Outlier Detection
From the statistical summary, it was found that there exists data point with call duration longer than the natural time in a month (>30*24*60 mins). These observations were tagged as outliers and were removed from the analysis.

3. Features

Features included in the classification models are 'day', 'brand', 'city_flag', along with 'recordsum', 'timesum', 'callersum' for 3 consecutive months. There are in total 12 features.
Based on the classifier value (flag 1 means left the company), it could be found that most of the customers who left the company tend to use the services less. Certainly, there is patterns behind users' actions, and classification was used to help figuring out the patterns. Detailed feature value distribution charts can be found in Appendix (3).

Models Building

1. Original Models

Four different models were used: Random Forest, Logistic Regression, GBDT (Gradient Boosting Decision Tree) and XGBoost.

(1) Models without weight adjustment
Results for the four different models mentioned above are shown below:

	TN	FN	FP	TP	Test Accuracy
Random Forest	850263	22250	104106	41433	0.873884
Logistic Regression	915407	7430	128533	17121	0.846720
GBDT	850919	22296	102498	42339	0.855418
XGBoost	851827	21281	104225	40724	0.876622

(2) Models with weight adjustment
From the result shown above, Random Forest and XGBoost provided better test accuracy than the other two algorithms. Therefore, Random Forest and XGBoost algorithm was selected for further analysis.
It is observed that the dataset is imbalanced as only around 16% of the data points belong to class 1, whereas about 84% of the data points belong to class 0. This means that if a model simply predicts everything to be class 0, it would still have a relatively high accuracy (~84%). To avoid this, one simple technique is to assign weights for different class.
Evaluation of algorithm performance is done through Confusion Matrix. The goal is to choose a set of weights that would prevent False Positive and False Negative values tremendously exceed True Positive and True Negative values, but still maintaining relatively good test accuracy and ROC score.
Based on the test results, there is a trade-off between test accuracy and ROC score.
(Class 0 weight = 0.2, Class 1 weight = 0.8) was chosen because it turned out to be balanced on both the test accuracy and ROC score.
Detailed results of algorithm evaluations with weight adjustment for the algorithms can be found in Appendix (4) and (5).

2. Feature Selection

Since in real-life situation, the dataset used by telecommunication companies is much larger than the train set used in this project, algorithm runtime is a real concern.
To reduce runtime, feature selection was done to see whether certain features can be removed from the model. 'XGBRegressor' and 'RandomForestClassifier' from Scikit-Learn were used.
With a threshold = 0.2, meaning that features with importance <= 0.02 would not be considered.

The number of important features was reduced from 12 to 10. The detailed feature importance rate can be found in Appendix (6).

(1) Test Accuracy Difference
Based on the test accuracy, it was found that both XGBoost and Random Forest algorithm perform better with features selected by 'XGBRegressor'.

(2) Runtime
Use XGBoost algorithm as an example, with ***(Class 0 weight = 0.2, Class 1 weight = 0.8)***, runtime is about 452.905s with test accuracy = 0.8766. By using the 10 features selected by 'XGBRegressor', the runtime decreased to 359.208s whereas the test accuracy remains almost the same. Thus, feature selection is useful and necessary when dealing with large datasets.

3. Missing Data Handling

One of the amazing characteristics of XGBoost is that it is able to automatically handle missing value in the dataset. First of all, the result of running XGBoost with data that contains neither missing values nor outliers yield the following result with appropriate weight for each class ***(Class 0 weight = 0.2, Class 1 weight = 0.8)***:

Runtime (in second)	Accuracy score / ROC score	Confusion matrix
452.905	0. 8766/ 0.743	[[746397, 126434], [53690, 91531]]

After feature selection (down to 10 features from 12 features), the following was obtained:

Runtime (in second)	Accuracy score / ROC score	Confusion matrix
394.649	0.8732/ 0.742	[[746344, 126487], [53728, 91493]]

It is shown that the runtime has reduced for a fair amount, while the accuracy and the confusion matrix stay very similar to the previous result.

By keeping the NA values in the dataset, XGBoost algorithm yields the following results (with 12 features):

Runtime (in second)	Accuracy score / ROC score	Confusion matrix
431.509	0.8673/ 0.758	[[746698, 151493], [53673, 116627]]

The accuracy score is slightly lower than the result previously observed, it is mainly due to the impact of the missing value handling. However, compared to the result without missing values, the runtime is very similar, meaning that the presence of missing value does not slow down the algorithm. Hence, the fact that XGBoost can handle missing value while maintaining a high score is definitely a real advantage of this model.

In contrast, we input the data with mean of the corresponding feature, and we feed the imputed data to the Random Forest model. Below are the results with the adjusted weight:

Runtime (in second)	Accuracy score/ROC score	Confusion matrix
367.095	0.8661/ 0.643	[[871776, 27095], [115998, 53627]]

It can be observed that even though the accuracy score is high, its ROC score is not high, for which the main reason could be that despite of the weight adjustment, the confusion matrix is not ideal. For example, the true positive rate is much less than the false positive rate. Thus, one of the drawbacks of the Random Forest model is that it is indifferent to the weight adjustment, which indicates that it would not be a good model for classification of imbalanced data.

Conclusion

In brief, we applied various popular classification models, such as Random Forest and Logistic Regression, to a large dataset of client activities from several telecom companies. By applying different settings to the models and comparing the result, we showed that feature selection is indeed very useful in real life as it can reduce the runtime while maintaining a high accuracy score. Also, we learned that it is very important to realize that data being imbalanced can significantly affect the end result if it was not handled properly. Furthermore, we experimented the performance of XGBoost under several conditions, and the condition is as what we expected. In fact, we proved that XGBoost can indeed handle missing value very well by comparing its accuracy and runtime with the dataset without the missing value, and it can obtain a high ROC score by adjusting the weight for different classes.

Moreover, further analysis can be done on the possible existence of clustering phenomenon for customers leaving the service. In other words, given more time on studying this topic, it would be interesting to study the relation between each customer in a graph with weighted edge, which could represent the closeness of customer with another customer. Thus, further studies on this subject could generate more constructive advice to the telecom companies in order to improve their strategies to build a stronger relation with their customers.

Reference

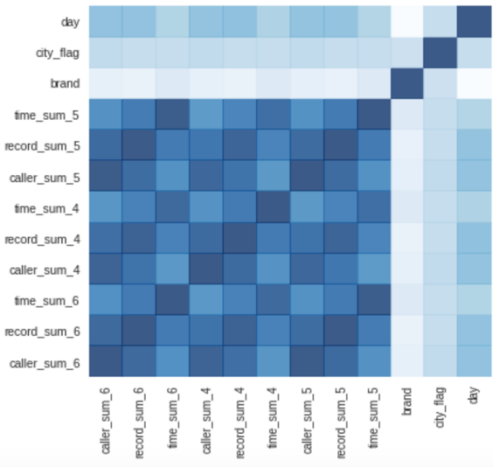
1. Research, Insight. “Worldwide Telecommunications Industry Revenue to Hit $2.4 Trillion in 2020, Says Insight Research.” *PR Newswire: News Distribution, Targeting and Monitoring*, 10 Mar. 2015, www.prnewswire.com/news-releases/worldwide-telecommunications-industry-revenue-to-hit-24-trillion-in-2020-says-insight-research-300047963.html.

2. "Customer Acquisition Vs.Retention Costs [Infographic]". *Invespcro.Com*, 2018, https://www.invespcro.com/blog/customer-acquisition-retention/. Accessed 13 Dec 2018.

3. Reis Pinheiro, Carlos Andre. Social Network Analysis In Telecommunications. Wiley, 2011.

Appendix

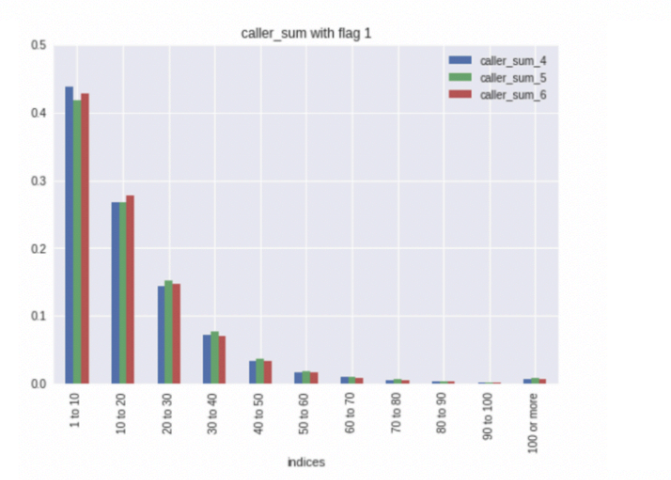
1. Feature Correlations

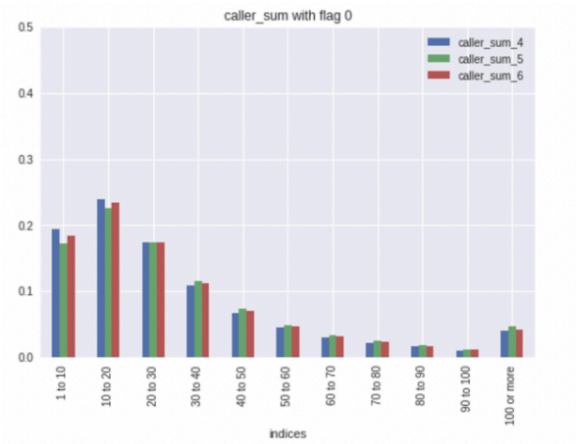


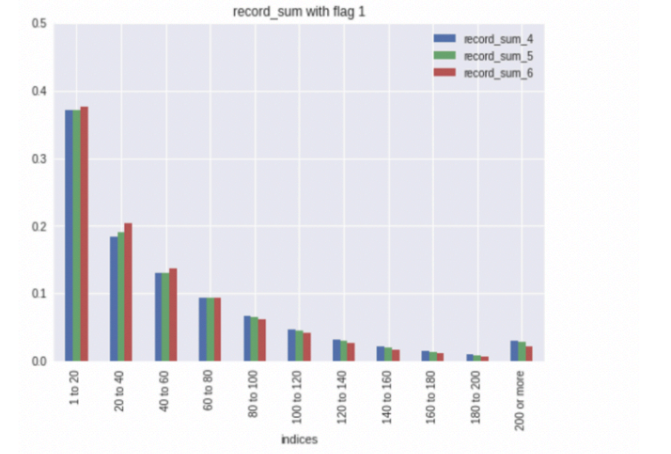
2. Statistical Summary for the After-transformed Dataset

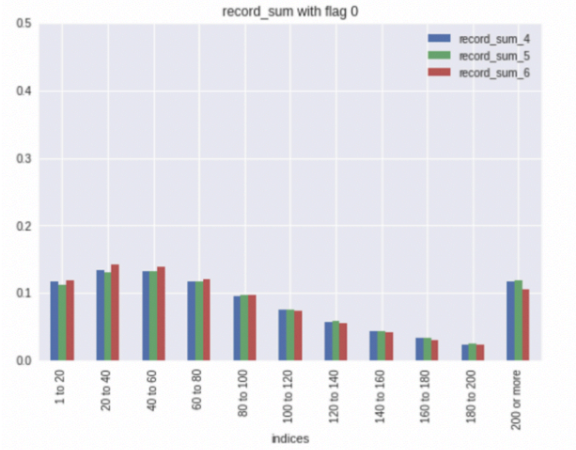
Features	Missing %	Missing Count	Min	Max	Mean	Mediate	Outlier Value	Outlier Count
ID	0.000	0						
day	0.000	0	61	7180	2034	2210		
callersum4	0.000	0	1	3164	33.45	23		
recordsum4	0.000	0	1	3861	107.3	77		
timesum4	0.000	0	0	17408	437.9	283		0
callersum5	0.026	27252	1	1906	36.56	26		
recordsum5	0.026	27252	1	2992	110.4	80		
timesum5	0.026	27252	0	19286	449.5	296		0
callersum6	0.016	17204	1	1684	34.57	24		
recordsum6	0.016	17204	1	2595	102.8	74		
timesum6	0.016	17204	0	1829633932	2189.586	273	1829633932	1

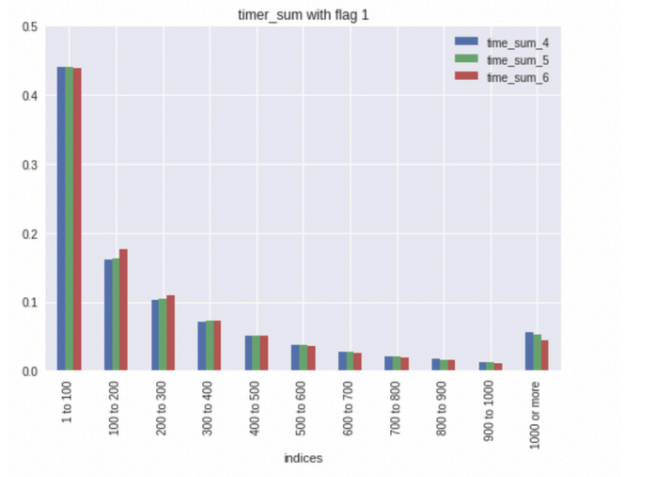
3. Feature Exploration

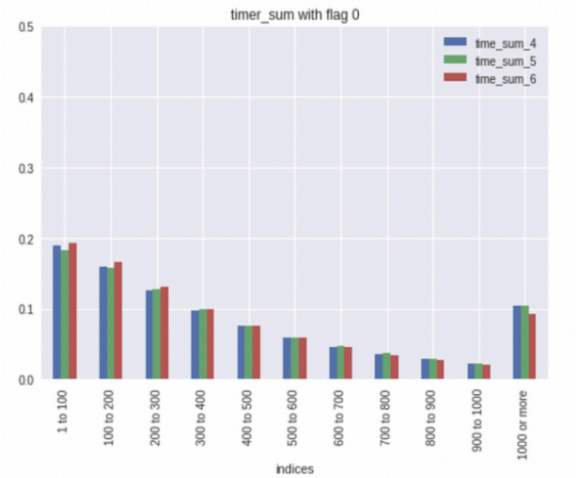












4. XGBoost Evaluation with Weight Adjustment

True Negative/ False Negative	False Positive/ True Positive	Class 0 Weight/ Class 1 Weight	Test Accuracy/ ROC Score
566217, 306891	23889, 121060	0.1, 0.9	0.67508695, 0.74184885
747602, 125506	53931, 91018	0.2, 0.8	0.87662263, 0.74209248
801410, 71698	71532, 73417	0.3, 0.7	0.85931043, 0.71219208
832975, 40133	89092, 55857	0.4, 0.6	0.87306703, 0.66969528
851827, 21281	104225, 40724	0.5, 0.5	0.87672007, 0.62829007

5. Random Forest Evaluation with Weight Adjustment

True Negative/ False Negative	False Positive/ True Positive	Class 0 Weight/ Class 1 Weight	Test Accuracy/ ROC Score
855013, 17680	107908, 37451	0.1, 0.9	0.87663891, 0.61869286
853726, 18967	106812, 38547	0.2, 0.8	0.87645130, 0.62172547
852433, 20260	105693,39666	0.3, 0.7	0.87628038, 0.62483375
851140, 21553	104786, 40573	0.4, 0.6	0.87590123, 0.62721280
849802, 22891	103858, 41501	0.5, 0.5	0.87549850, 0.62963831

6. Feature Importance

Feature	Importance
day	0.2014
record_sum_5	0.1557
caller_sum_5	0.0986
record_sum_6	0.0914
record_sum_4	0.0900
brand	0.0843
time_sum_5	0.0757
time_sum_4	0.0700
caller_sum_4	0.0643
city flag	0.0357
time_sum_6	0.0200
caller_sum_4	0.0129

Feature Importance for XGBoost

Feature	Importance
day	0.2062
record_sum_5	0.1017
record_sum_4	0.1041
time_sum_4	0.0966
time_sum_5	0.0958
time_sum_6	0.0908
record_sum_6	0.0854
caller_sum_4	0.0646
caller_sum_5	0.0597
caller_sum_6	0.0569
brand	0.0279
city flag	0.0104

Feature Importance for Fandom Forest