**Bank Customer Churn Prediction**

Nguyễn Hoàng Long - 19521788, Phan Phạm Quỳnh Hoa – 19521520,

Huỳnh Minh Thư - 19522304

**1. Abstract**

Predicting customers leaving the service (Customer Churn) is an essential problem to help businesses make the right decisions, classifying likely customers to quit their services, by that, businesses limit the scope of customer services to retain them, optimize costs and increase profits.

We use a bank data set to analyze, evaluate and compare results across three commonly machine learning models: Logistic Regression [1], Decision Trees [2] and an algorithm that is based on gradient boosting (XGBoost [3]). We compare and assume the outcome of these models, then choose appropriate methods to optimize the model and improve accuracy.

**2. Introduction**

Society is growing, followed by the rise of human demands. To meet these requirements, more and more services and utilities are developing. More and more banks have appeared, and along with this came the policy of customer services, so retaining customers seems more difficult. “Acquiring a new customer is anywhere from five to 25 times more expensive than retaining an existing one”, says Frederick Reichheld (BAIN & COMPANY). It makes sense: you don’t have to spend time and resources going out and finding a new client — you just have to keep the one you have happy.

Therefore, retaining customer relationship is so valuable, predicting customers leaving service makes it possible for enterprises to classify customers who are likely to leave the service, optimize costs and increase profits by limiting the scope of customer services for starting the Customer Services on the specific user.

Which has 10,000 samples, include some information, behaviors of customer and the binary feature for predicting which one will leave the bank information, behaviors of customer and the binary feature for predicting which one will leave the bank.

Training and evaluation on two popular models are Logistic Regression and Decision Trees. In addition, XGBoost (Extreme Gradient Boosting) implements machine learning algorithms under the Gradient Boosting framework that gained high prizes in data analysis competitions, regardless of the type of prediction task at hand, regression or classification.

Moreover, (1) because the dataset has a problem with the balance of data, our team will use Oversampling technique to solve that problem. (2) Fine-tune model used GridSearchCV to find the best hyper-parameter that gave a high result.

Our team have public all source code of project at: <https://github.com/nghoanglong/bank-customer-churn>

**3. Models**

**3.1 Logistic Regression:** is a popular classification model that is a process of modeling the probability of a discrete outcome given an input variable. Logistic Regression is a transformation of a linear regression using the sigmoid function. We use the activation function (sigmoid) to convert the outcome into categorical value.

The logistic regression model passes the outcome of a linear function of features through a logistic function to calculate the probability of an occurrence. The model then maps the probability to binary outcomes.

The formula calculates the probability of output:

P( = = σ(

with **a = x,** and the task we need to minimize the following loss function:

J(w) =

The updated formula is similar to Linear Regression using Gradient Descent:

**3.2 Decision Trees:** are a Supervised learning technique that can be used for both classification and regression problems. A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees. The decisions are performed on the basis of features of the given dataset (attribute) as categorical, independent, and in no particular order.

Decision tree also works with specific vectors of dataset including *categorical* and numerical. Moreover, decision trees probably require little data standardization.

Advantages of the Decision Tree:

* Possible to validate a model using statistical tests.
* Can handle a large amount of data in the short term.
* Seems like human decision.
* Easy to explain. The result is a single model that aggregates the results of several models.
* Trees can be visualized.

Measures:

* Gini impurity – Gini index: CART
* Information gain: ID3, C4.5, C5.0
* Variance reduction: CART

**3.3 XGBoost(Extreme Gradient Boosting):** belongs to a family of boosting algorithms and uses the gradient boosting (GBM) framework at its core, Boosting – Ensemble Learning consist of: (1) combines a set of weak learners, learn to complement each other. (2) Instead of finding the global optimal, models of Boosting method try to find local optimal for each model with the desire to gradually come to a global solution. The problem as follows:

with

in which:

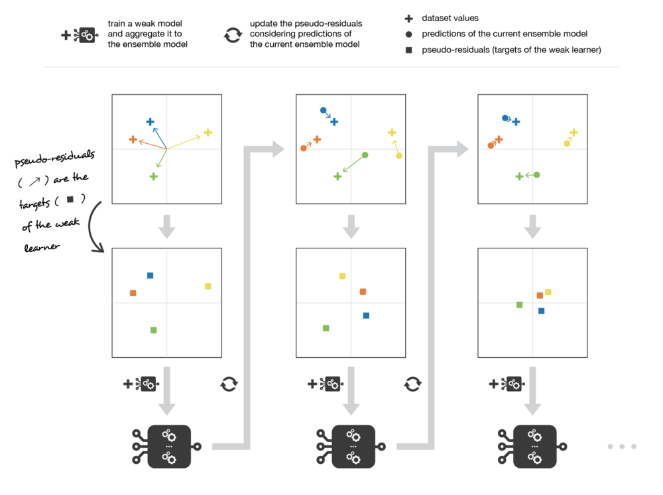
* L: Loss function
* y: label
* : confidence score of weak learner n (called weighting)
* : weak learner n

Reference to the formula of update model parameters with Gradient Descent:

If consider that weak learners to be a *W function,* then each learner can be considered a *w function.* Minimize loss function **L(y, W)** by Gradient Descent

Related to:

with is the next learner. At that time, the next model has to learn to fit the value on the right, called **pseudo-residuals**

****

*Figure 1: Illustration of the Gradient Boosting learning process*

**4. Experiments**

**4.1 Experimental Setup**

Our team perform some preprocessing data follow this steps: fetch file dataset and save it to the relative path, apply Exploratory Data Analysis to get the information about the data, perform data cleaning (handle null data by fill them with median, eliminate outliers with IQR [4] technique) and feature engineering ( encode sentences with OneHotEncoding, scale numerical feature by Normalization and reduce the input features by calculate the correlation between them using Chi-square [5] and ANOVA [6] testing). After that, we vectorized all input features and put it to the models for training steps. Lastly, evaluate the outcome with classification metrics and get the results which show in Table 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Approach** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| **Logistic Regression** | 0.826 | 0.80 | 0.83 | 0.80 |
| **Decision Tree** | 0.806 | 0.80 | 0.81 | 0.80 |
| **XGBoost** | 0.863 | 0.85 | 0.86 | 0.85 |

Table 1: Performance of three models

Table 1 shows us that accuracy of Decision Tree model ≈ 80.6% and respectively increase with Logistic Regression model ≈ 82.6% (80.6% → 82.6% ≈ 2%), XGBoost model ≈ 86.3% (80.6% → 86.3% ≈ 5.7%). Another metrics such as Precision, Recall and F1-Score also increased. For general, we could see that XGBoost model gave us the best result in this dataset

**4.2 Optimize**

Because the imbalance problem of data, we used Oversampling [7] technique for handling it which shows in the Table 2

|  |  |  |
| --- | --- | --- |
|  | **Original** | **Oversampling** |
| **Class: 0** | 7677 | 7677 |
| **Class: 1** | 2323 | 7677 |

Table 2: Oversampling for handling imbalance data

Moreover, we apply GridSearchCV for hyper-parameter tuning. After that, repeatly evaluate the outcome of three models and get the results which show in Table 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Approach** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| **Logistic Regression** | 0.726 | 0.73 | 0.73 | 0.73 |
| **Decision Tree** | 0.78 | 0.78 | 0.78 | 0.78 |
| **XGBoost** | 0.89 | 0.89 | 0.89 | 0.89 |

Table 3: Performance of three models

We could see that the Decision Tree model (criterion = entropy, max\_depth = 6) has the performance decrease ~ 2.6% and the same thing for Logistic Regression model (C = 10.0, solver = Liblinear [8]) ~ 10%. XGBoost gave the better result which all metrics have increase ~ 2.7%. So generally, XGBoost still gave us the best performance at all

**5. Conclusion**

In this report, we have defined the problem, conduct experiments on three models and attempt to optimize the result.

During the implementation of this project, our team also encountered restrictions such as a lack of knowledge about machine learning models, the model performance issues. On the other hand, the implementation of the project is resource-intensive, seems quite simple but close to real life and is an essential issue in many major banks across the world. We hope that our project can contribute to extensive research and improve the performance of large datasets.

**6. References**

[1] Hosmer, David W., and Stanley Lemeshow. Applied Logistic Regression. 1989.

[2] Quinlan, J. R. “Induction of Decision Trees.” Machine Learning, vol. 1, no. 1, 1986, pp. 81–106.

[3] Chen, Tianqi, and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System.” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[4] (2008) Interquartile Range. In: The Concise Encyclopedia of Statistics. Springer, New York, NY.

[5] Wuensch K.L. (2011) Chi-Square Tests. In: Lovric M. (eds) International Encyclopedia of Statistical Science. Springer, Berlin, Heidelberg

[6] Girden, E. R. (1992). ANOVA: Repeated measures. Sage.

[7] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. J. Artif. Int. Res. 16, 1 (January 2002), 321–357.

[8] Fan, Rong-En, et al. “LIBLINEAR: A Library for Large Linear Classification.” Journal of Machine Learning Research, vol. 9, no. 61, 2008, pp. 1871–1874.