Customer Churn prediction based on eXtreme Gradient Boosting classifier

Previsione della probabilità di non ritorno della clientela attraverso il metodo di classificazione eXtreme Gradient Boosting

Mohammed Hassan Elbedawi Omar and Matteo Borrotti

Abstract Nowadays, Machine Learning (ML) is a hot topic in many different fields. Marketing is one of the best sectors in which ML is giving more advantages. In this field, customer retention models (churn models) aim to identify early churn signals and recognize customers with an increased likelihood to leave voluntarily. Churn problems fit in the classification framework, and several ML approaches have been tested. In this work, we apply an innovative classification approach, eXtreme Gradient Boosting (XGBoost). XGBoost demostrated to be a powerful technique for churn modelling purpose applied to the retail sector.

Abstract *Al giorno d'oggi, il Machine Learning (ML) è un argomento estremamente importante in differenti settori. Ad esempio, il marketing rappresenta uno dei settori più vivi per l'applicazione di metodi di ML. In questo settore, i modelli di customer retention (modelli di churn) analizzano il comportamento dei clienti per individuare quali di questi non torneranno a effettuare acquisti. Questo problema può essere tradotto in un problema di classificazione e molti modelli di ML sono stati testati. In questo lavoro applicheremo un innovativo approccio di classificazione, eXtreme Gradient Boosting (XGBoost) al settore del commercio al dettaglio. Dai risultati ottenuti, si può notare che XGBoost può essere considerato come una tecnica molto efficace per i modelli di churn.*

Key words: churn, classification, XGBoost, boosting

Mohammed Hassan Elbedawi Omar
Energia Crescente S.r.l., Piazza Missori 2, Milano, e-mail: mohammedhassan.omar@en-cre.it

Matteo Borrotti
Energia Crescente S.r.l., Piazza Missori 2, Milano, e-mail: matteo.borrotti@en-cre.it
Institute of Applied Mathematics and Information Technology (IMATI-CNR), Via Alfonso Corti 12, Milano.

1 Introduction

Machine learning (ML) is gaining momentum due to a virtually unlimited number of possible uses and applications. ML is capable to produce models that can analyse bigger and more complex data and deliver accurate insights in order to identify profitable opportunities or to avoid unknown risks. As more data becomes available, more ambitious problems can be tackled. ML is widely used in many fields, such as: recommender systems, credit scoring, fraud detection, drug design, and many other applications.

An important sector where ML is widely applied is marketing. An important topic is related to customer segmentation [6]. In customer segmentation, clustering approaches are used to group customers based on their purchase behaviour. Another important ML application is customer retention. The cost of customer acquisition is much greater than the cost of customer retention and companies are interested in improving customer retention models (churn models). Churn models aim to identify early churn signals and recognize customers with an increased likelihood to leave voluntarily [8, 4].

In this work, we evaluate the performance of a novel approach, eXtreme Gradient Boosting (XGBoost) [2] classifier, on the churn problem. XGBoost is a scalable machine learning system for tree boosting. XGBoost differs from classical tree boosting algorithms for handling sparse data and a theoretically justified weighted quantile sketch procedure, which enables handling instance weights in approximate tree learning. XGBoost algorithm is compared with a classical Decision Tree classifier [7]. The two algorithms are applied to the problem of customers churning in the retail sector. XGBoost outperforms the Decision Tree classifier demonstrating to be a promising approach for customer retention models.

2 eXtreme Gradient Boosting (XGBoost)

The XGBoost algorithm [2] is based on the Gradient Boosted Decision Tree (GBDT) [3]. The following description is based on the work of [9]. XGBoost efficiently deals with sparse data and is suitable for large-scale dataset since implements distrubuted and parallel computing flexibility. XGBoost estimates the target feature by a series of decision tree and defining quantised weight for each leaf node. The prediction function is defined as follows:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i), \tag{1}$$

where $\hat{y}_i$ is the predicted class of the $i-th$ observation, $\mathbf{x}_i$ is the corresponding feature vector, K is the total number of decision trees. The function f_k is defined as

$$f_k(\mathbf{x}_i) = \omega_{q_k(\mathbf{x}_i)}, \tag{2}$$

where $q_k(\mathbf{x}_i)$ is the structure function of the $k-th$ decision tree that map $\mathbf{x}_i$ to the corresponding leaf node and ω is the vector of the quantised weight.

The accuracy and complexity of the model are taken into account by a regularisation term added to the loss function. The learning process is based on the minimisation of the following loss function:

$$L_t = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \tag{3}$$

where l is the loss function. The loss function measures the differences between the observed and predicted classes. Ω denotes the complexity of the model.

3 Experimental settings

We consider a dataset composed of 7013 customers belonging to a pet shop. Sensible data were anonymised for privacy purpose. For each customer, the following features are considered: (1) Total net amount of money spent in the considered period, (2) Total amount of discount in the considered period, (3) Number of receipts in the considered period, (4) Number of days since last purchase, (5) Number of days between first and last purchases in the considered period, (6) Average number of days between purchases and (7) Label that is equal to 1 if the customer made at least a purchase in a precise period or 0 otherwise (target feature). It should be noticed that features number (1), (3) and (4) correspond to the Recency (R), Frequency (F) and Monetary (M) features of the well-known RFM model [5].

Features were computed considering the transactions from 29^{th} October 2017 to 21^{th} January 2018. The target feature was identified considering the transactions from 22^{th} January 2018 to 2^{nd} February 2018.

The dataset is divided in three parts: 65% training set, 20% validation set and 15% test set. The validation set is used for parameters optimisation purposes. The Decision Tree and XGBoost's parameters considered for optimisation are the maximum depth of a tree (*max_depth*: $\{2,6,10\}$) and the number of trees (*n_estimators*: $\{100,500,900\}$). For both algorithms, the maximum number of iterations (*nrounds*) is fixed to 100. The XGBoost's learning rate (*eta*) is fixed to 0.01 to avoid early convergence.

XGBoost and Decision Tree were implemented in Python 3.5.4 using XGBoost package version 0.7 and scikit-learn package version 0.19.1.

3.1 Performance metrics

The XGBoost and Decision Tree's performance were evaluated using two metrics: accuracy and logarithmic loss (log-loss) [10]. The confusion matrix (see Table 1) is also used to analysed the general behaviour of the approach.

Accuracy is defined as $Accuracy = \frac{T_p+T_n}{T_p+F_p+T_n+F_n}$.

Table 1 Confusion matrix.

	Observed class 1	Observed class 0
Predicted class 1	True positive (T_p)	False positive (F_p)
Predicted class 0	False negative (F_n)	True negative (T_n)

Log-loss is defined as $l_n = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c} y_{ij} log(\hat{y}_{ij})$, where n is the number of observations to be predicted, c is the number of classes (i.e. $c = 2$), y is the observed class and $\hat{y}$ is the predicted probability class. Log-loss should be minimized.

In simple words, accuracy is the count of predicted classes correctly classified among the observed ones. Log-loss takes into account the uncertainty of the predicted class based on how much it varies from the observed class.

These metrics were used to compare XGBoost and Decision Tree on the validation set. Furthermore, we considered the Receiver Operating Characteristic (ROC) curve [1] as a performance metric on the test set. Receiver Operating Characteristic (ROC) curve is based on true positive rate, $P(Tp) = \frac{T_p}{T_p+F_n}$, and false positive rate, $P(Fp) = \frac{F_p}{T_n+F_p}$. ROC curve is used to show in a graphical way the trade-off between true positive rate and false positive rate. The area under the ROC curve is a measure of accuracy.

4 Results

XGBoost and Decision Tree were compared on the validation set. Table 2 shows the performance metrics for all the possible parameters configurations considered in this study. Decision Tree has the best performance with number of trees (*n_estimators*) equal to 500 and maximum depth of a tree (*max_depth*) equal to 10. More precisely, an accuracy of 0.834 and log-loss 0.396. Decision Tree has a similar behaviour also increasing the number of trees. XGBoost reaches the best accuracy (0.895) and log-loss (0.317) with number of trees (*n_estimators*) equal to 900 and maximum depth of a tree (*max_depth*) equal to 10. XGBoost outperforms Decision Tree in all parameters configurations except when the number of trees (*n_estimators*) is set to 100.

Given the obtained results, XGBoost were deplyed on the test set with the best configuration previously found (*n_estimators* = 900 and *max_depth* = 10). Table 3

Table 2 Performance results of Decision Tree classifier on the Validation set.

		Decision Tree		**XGBoost**	
n_estimators	*max_depth*	*Accuracy*	*Log-loss*	*Accuracy*	*Log-loss*
100	2	0.732	0.549	0.727	0.575
100	6	0.756	0.508	0.761	0.543
100	10	0.832	0.396	0.832	0.483
500	2	0.735	0.549	0.743	0.536
500	6	0.757	0.507	0.780	0.462
500	10	0.834	0.396	0.870	0.346
900	2	0.733	0.549	0.743	0.533
900	6	0.756	0.507	0.792	0.437
900	10	0.833	0.396	**0.895**	**0.317**

shows the obtained confusion matrix. The confusion matrix highlights an issue on the prediction of class 0 (not churned people). This issue arises when the two classes are unbalanced. An accuracy of 0.734 and log-loss of 0.567 were obtained. The two metrics exhibit a deterioration from validation set to test set. This deterioration shows a low generalization power of the XGBoost with the parameters configuration selected.

Table 3 Confusion matrix of the XGBoost Classifier on the Test set.

	Observed class 1	Observed class 0
Predicted class 1	557	97
Predicted class 0	169	178

The ROC curve shows a good performance in terms of area under the curve, which is equal to 0.744.

5 Conclusions and future work

In this work, XGBoost is applied to the customer retention (churn) modelling problems. Churn models are appealing tools for the marketing sector. The novel classifier is compared with the Decision Tree. The two approaches were tested on a dataset related to retail sector. From the empirical study, XGBoost outperforms Decision Tree classifier in all the parameters configurations tested except when the number of trees (*n_estimators*) is set to 100.

The best XGBoost's configuration (*n_estimators* = 900 and *max_depth* = 10) was evaluated on the Test set. XGBoost confirms the good results obtained in the validation set but some issues arise. First of all, the target feature was identified considering the transactions from 22^{th} January 2018 to 2^{nd} February 2018. A sensitive

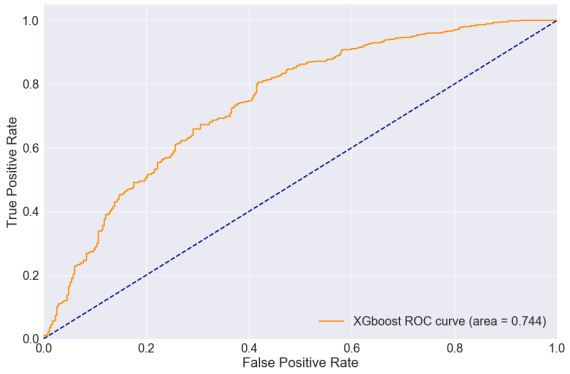

Fig. 1 ROC curve of the XGBoost classifier on the Test set.

analysis on the time window used for target feature definition should be done in oder to improve prediction capability. Additionally, XGboost should be compared with more classification approaches in order to understand the main advantages and disadvantages. A wider parameter analysis needs to be carried out to improve the generalization power. Techniques for unbalanced data should be considered to avoid misclassification problems.

Given that, XGBoost is a promising approach for customer retention modelling problems.

References

1. Brandley, A.P.: The use of the area under the ROC curve in the evaluation of Machine Learning algorithms. Pattern Recognition, **30**, 1145–1159 (1997)
2. Chen, T., Guestrin, C.: XGBoost: a Scalable Tree Boosting system. ArXiv, 1–10 (2016)
3. Friedman, J. Greedy function approximation: A Gradient Boosting machine. Annals of Statistics, **30**, 11891232 (2001)
4. García, D.L., Nebor, Á., Vellido, A.: Intelligent data analysis approaches to churn as a business problem: a survey. Knowledge and Information Systems. **51**, 719–774 (2017)
5. Hughes, A.M.: Strategic database marketing. Probus Publishing Company, Chicago (1994)
6. Ngai, E.W.T., Xiu, L., Chau, D.C.K.: Application of data mining techniques in customer relationship management: A literature review and classification, Expert Systems with Applications, **30**, 1145–1159 (1997)
7. Safavian, R.S., Landgrebe, D.: A survey of Decision Tree classifier methodology. IEEE Transactions on Systems, Man, and Cybernetics, **21**, 660–674 (2016)
8. Vafeiadis, T., Diamantaras, K.I., Sarigiannidis, G., Chatzisavvas, K.Ch.: A comparison of machine learning techniques for customer churn prediction. Simulation Modelling Practice and Theory. **55**, 1–9 (2015)
9. Wang, S., Dong, P., Tian, Y.: A novel method of statistical line loss estimation for distribution feeders based on feeder cluster and modified XGBoost. Energies, **10**, 1–17 (2017)
10. Whitehill, J.: Climbing the Kaggle leaderboard by exploiting the log-loss Oracle. ArXiv, 1–9 (2017)