**Prediction of Bank Marketing Based on Machine Learning**

Li, Xin, [li.xin1@husky.neu.edu](mailto:li.xin1@husky.neu.edu)

Guan, Weijie, [guan.we@husky.neu](mailto:guan.we@husky.neu).edu

**Abstract**

Machine Learning is becoming an efficient way in many areas to do prediction and we have discussed many theories especially in finance. Trying to put theory into practice, we propose an approach to make a prediction of bank marketing. In this report, we analysis the structure of dataset and then do data processing to ensure our data is ready to input. Then we compare four different algorithms, Random Forest(RF), Logistic Regression(LR), XGBoost and Neural Network(NN), to explore the efficiency. Further, we blend all these four models to make a new prediction. Two evaluation methods are applied in this report: the class report and ROC. The results show that XGBoost(precision = 0.6, f1-score = 0.92, area of ROC = 0.945) perform better than other four models. Notwithstanding that XGBoost is good at find the number of people who are willing to subscribe term deposit as many as possible, LR approach is the most efficient way to find true potential client.

**1. Introduction**

**1.1 Overview**

In this report, we first explore the structure of our dataset by drawing histogram and violin plot. Then we transfer features of string types into numeric as input of our approach. Third, we calculate the correlation between outcomes and features to explore the significance of our features. Also, smote approach is applied to balance our training dataset. Since our dataset is ready, we do tests to compare four algorithms, Random Forest(RF), Logistic Regression(LR), XGBoost, Neural Network(NN) and blending models. The results reveals that the performance of XGBoost(precision = 0.6, f1-score = 0.92, area of ROC = 0.945) is the best. Notwithstanding that XGBoost is good at find the number of people who are willing to subscribe term deposit as many as possible, LR approach is the most efficient way to find true potential client.

**1.2 Motivation**

Often we contact with clients by phone call to know if they will subscribe a term deposit. Notwithstanding phone call is a direct way to get in touch with clients, it’s inefficient and a waste of human resource. For the foregoing reason, we have developed a system to distinguish the person with high probability to subscribe a term deposit.

**1.3 Approach**

The first algorithm we choose is LR. LR is a easy understanding and straight algorithm to use in machine learning. The results have good interpretability and are beneficial to decision analysis.

RF is also one of the most popular algorithm. It can process data with high dimensions without making feature selection and the ability of model generalization is strong.

Neural Network is a relatively new but mature algorithm. The advantage is that it can automatically learn what features to extract. Besides, the capability to calculate a large number of data enable NN to perform better than other algorithms.

The XGBoost includes regularization to prevent overfitting. And it uses the second derivative for more precise losses. In addition, XGBoost support for column sampling not only reduces overfitting, but also reduces computation.

At last, we used blending which combine all the models mentioned above together to generate new features and train a new model base on these features.

**1.4 Dataset**

The dataset we use is from the research which Moro et al. (2014)[1] have done. We download it from UCI Machine Learning Repository.

Here’s the link: <http://archive.ics.uci.edu/ml/datasets/Bank+Marketing#>

**2. Background**

Moro et al. (2011)[2] described an implementation of a DM project based on three machine learning methods, Naive Bayes(NB), Decision Tree(DT) and Support Vector Machine(SVM). They presented that the best model was materialized by SVM, which achieved high predictive performance.

Furthermore, Moro et al. (2014)[1]  did another deeper research. They compared four DM models: Logistic Regression, DT, NN and SVM and evaluated these four models by AUC and ALIFT. Then they concluded NN performed better than other algorithms(AUC=0.8, ALIFT=0.7). Also, two knowledge extraction techniques were applied in the research: a sensitivity analysis and a decision tree. They pointed out that the three month Euribor rate was the most significant feature by both two techniques, followed by the direction call, the bank agent experience and so on. However, the limitation is that they included the data within the range 2008-2012 rather than split into two sub-periods of time, which made them fail to analyze impact between years.

**3. Approach**

**3.1 Logistic Regression**

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable. It is a very popular choice that operates a smooth nonlinear logistic transformation over a multiple regression model and allows the estimation of class probability  where p is the probability that y = 1, {x1 , …… , xn} are the n features from the dataset with the related weights {β0,……,βn} (x0 =1). Due to the additive linear combination of its independent variables (x), the model is easy to interpret. Yet, the model is quite rigid and cannot model adequately complex nonlinear relationships.

**3.2 Random Forest**[4]

Random forest is just an improvement over the top of the decision tree algorithm. The core idea behind Random Forest is to generate multiple small decision trees from random subsets of the data (hence the name “Random Forest”).

Each of the decision tree gives a biased classifier (as it only considers a subset of the data). They each capture different trends in the data. This ensemble of trees is like a team of experts each with a little knowledge over the overall subject but thorough in their area of expertise.

Now, in case of classification the majority vote is considered to classify a class. In analogy with experts, it is like asking the same multiple choices question to each expert and taking the answer as the one that most no. of experts vote as correct. In case of Regression, we can use the avg. of all trees as our prediction. In addition to this, we can also weight some more decisive trees high relative to others by testing on the validation data.

**3.3 XGBoost**[5]

XGBoost is a popular and efficient open-source implementation of the gradient boosted trees algorithm. Gradient boosting is a supervised learning algorithm, which attempts to accurately predict a target variable by combining the estimates of a set of simpler, weaker models.

When using gradient boosting for regression, the weak learners are regression trees, and each regression tree maps an input data point to one of its leafs that contains a continuous score. XGBoost minimizes a regularized (L1 and L2) objective function that combines a convex loss function (based on the difference between the predicted and target outputs) and a penalty term for model complexity (in other words, the regression tree functions). The training proceeds iteratively, adding new trees that predict the residuals or errors of prior trees that are then combined with previous trees to make the final prediction. It's called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.

**3.4 Neural Network**[6]

An Artificial Neuron Network (ANN), popularly known as Neural Network is a computational model based on the structure and functions of biological neural networks. It is like an artificial human nervous system for receiving, processing, and transmitting information in terms of Computer Science.

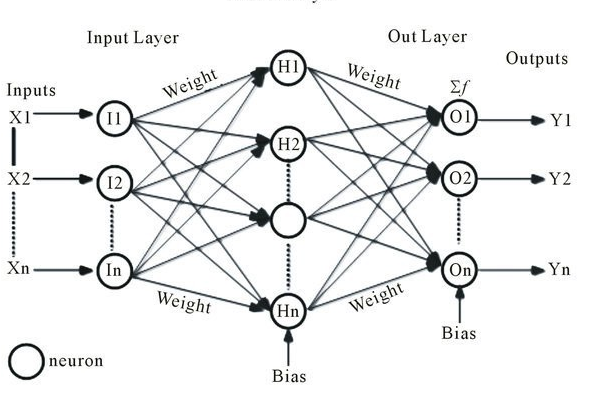
Basically, there are 3 different layers in a neural network :-Input Layer (All the inputs are fed in the model through this layer), Hidden Layers (There can be more than one hidden layers which are used for processing the inputs received from the input layers), Output Layer (The data after processing is made available at the output layer).

Figure 1. The Graph of MLR NN Model

In our experiment, we construct a MLR NN model with 3 hidden layer and use ReLU(between the input layer and hidden layer) and Sigmoid (the rest layers)as the activate function. The structure is showed as follow

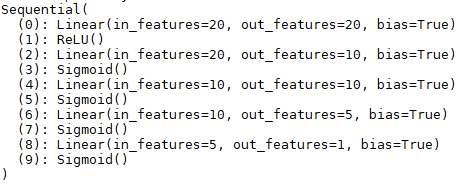


Figure 2. The structure of NN Model

**3.5 Blending**

Blending is a model ensembling technology with basic idea that use a pool of base predictors, and then use another predictor to combine the base predictions. The process of blending is showed as follow:

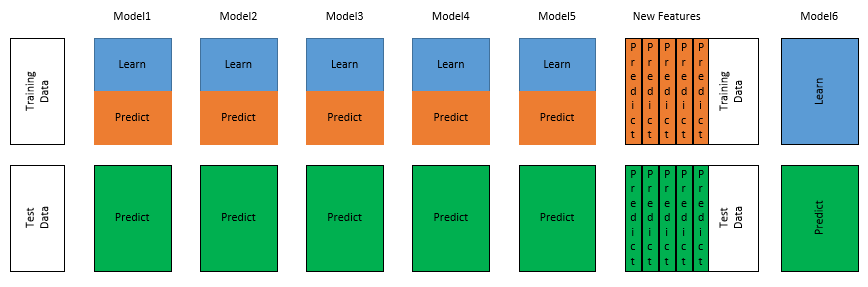


Figure 3. The process of Blending

In our experiment, we combine the 4 models we mentioned above to generate the new features. Firstly, we trained the 4 models with training set and then make prediction on the validation set. After that, we combined the 4 predictions together to generate new features. Each prediction represent a feature in the new features. Then we used these new features to train a random forest model(just like the model6 showed in the graph) and use the new model to make prediction on the test set.

1. **Results**

**4.1 Dataset**

The dataset we use in our research is from UCI Machine Learning Repository. The data includes 20 features and 1 output. The feature are age, job, marital, education, default(has credit in default?), housing(has a housing loan), loan(has personal loan?), contact(contact communication type), month(last contact month of year), day\_of\_week(last contact day of the week), duration(last contact duration, in seconds), campaign(number of contacts performed during this campaign and for this client), pdays(number of days that passed by after the client was last contacted from a previous campaign), previous(number of contacts performed before this campaign and for this client), poutcome(outcome of the previous marketing campaign), emp.var.rate(employment variation rate), cons.price.idx(consumer price index), cons.conf.idx(consumer confidence index), euribor3m(euribor 3 month rate), nr.employed(number of employees). And our output is y: If the client subscribed a term deposit.

**4.2 Experiments and performance evaluation**

After the data processing, we compared the performance of our four models: LR, RF, XGBoost and NN. We apply the classification report method from the package Sklearn as evaluation.

For LR model, Figure 4. shows the performance.

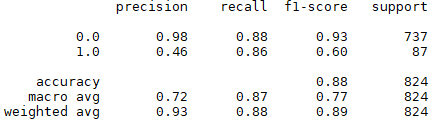


Figure 4. The performance of LR Model

From the report table, we can conclude that the performance is quite acceptable since our f1-score reaches to 0.88. It’s obvious that our model’s ability to recognize class 0 is quite outstanding. The precision of that is 0.98. Notwithstanding the excellent performance of recognizing class 0, we find that the ability to distinguish class 1 is less than satisfactory. In that case, we may learn from this model that we are able to eliminate the person who won’t subscribe our term deposit. But we still have difficulty finding the potential client.

As for RF, here’s the result.

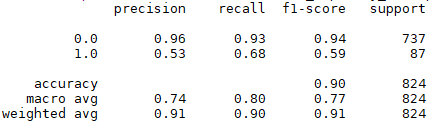


Figure 5. The performance of RF Model

The things we need to pay attention is that by using model RF, we do have an improvement of distinguishing class 1. Nevertheless, the precision of class 0 is not as excellent as LR model we have. Further, since the recall is 0.68, which means that nearly half of the people our model selected are not willing to subscribe a term deposit.

Here’s another model, XGBoost.

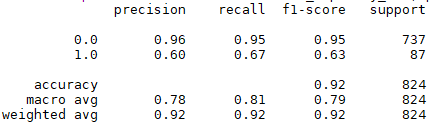


Figure 6. The performance of XGBoost Model

From the table, we find that the performances of XGBoost and RF are quite similar. Both of them do well in distinguishing class 1 but have limitation in true potential clients.

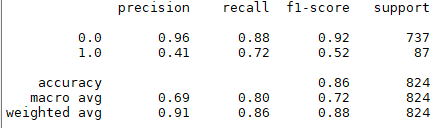


Figure 7. The performance of NN Model

The interesting thing we find is that even though the f1-score of NN model is not satisfied, the improvement of recall of class 1 is obvious.

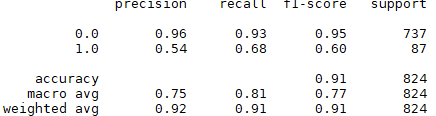


Figure 8. The performance of Blending Model

Further, we do a model emerge by blending all these four models to make a new prediction. Results has showed above.

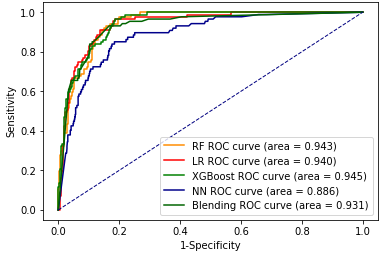


Figure 9. The ROC of four Models

Also, we use Receiver Operating Characteristic curve(ROC) to evaluate our model. From the figure above, we could conclude that the performances of Blending, XGBoost, LR and RF are quite approach, while XGBoost does a better prediction.

**4.3 Discussion**

Figures above reveal that XGBoost is the best approach on both two evaluation methods. However, a lot of work still need to be done. The dataset we has is still limited. If we could have more data on people who subscribe term deposit, the performance would be better. Further, all of our work are based on the features that we download from the website.

**5. Conclusion**

From the perspective of recall, we can conclude that LR(recall = 0.86) has better performance than other models since one of our goals are to find the person who are willing to subscribe term deposit efficiently. However, if we consider the total number of people who are willing to do that, there is no doubt that XGBoost(precision = 0.6, f1-score = 0.92, area of ROC = 0.945) does the best. If we consider the f1-score, the differences between XGBoost, RF and Blending model is negligible. On the other hand, ROC reveals that the performances of XGBoost, RF and LR are almost same.

For further study, If we can extract new information from original features to create a brand new variable to improve our model’s predictions would be another challenge. Besides, blending model should be the best model. But the fact is that XGBoost does well both in two evaluation methods. We still take a consideration how to balance different models to make a better prediction.

**References**

[1] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014

[2] S. Moro, R. Laureano and P. Cortez. Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology. In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimaraes, Portugal, October, 2011. EUROSIS. [bank.zip]

[3] Wikipedia, Logistic regression, <https://en.wikipedia.org/wiki/Logistic_regression>

[4] Raghav Aggiwal, Introduction to Random forest https://dimensionless.in/introduction-to-random-forest/, Feb 28, 2017

[5] How XGBoost Works, <https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost-HowItWorks.html>

[6] David Fumo, A Gentle Introduction To Neural Networks Series, <https://towardsdatascience.com/a-gentle-introduction-to-neural-networks-series-part-1-2b90b87795bc>, Aug 4, 2017