Service Express Sales Outcome Prediction

Peiyan Wang
School of Information
University of Michigan
peiyanw@umich.edu

Zhengyang Zhao
School of Information
University of Michigan
zzyang@umich.edu

1. Introduction

Applied Machine Learning is meant to apply machine learning models to real-world problems. As a result, we successfully earned the opportunity to deal with real-world issues by using techniques and machine learning skills learned from the course. This is an external project cooperating with Service Express, a company that provides data center maintenance services to its customers. When there is a potential customer, the company would try to sell the services to the customer and track their interactions by creating an "opportunity". The company wins an opportunity if a service is successfully provided to a customer. The company would like to build a model to predict whether the company wins the opportunities from its potential customers, which can support sales teams and help them to make decisions. With a model of 85-90% accuracy, the company can point their sales leadership at larger opportunities where the model is predicting a loss and they can try to intervene earlier and provide assistance to help win the opportunity or if it's looking like the company will win the opportunity or will lose the opportunity and close the opportunity sooner to avoid unnecessary loss.

2. Methods

2.1 Description of the dataset

All the data are stored in an excel provided by the Service Express company. We stored the data in a csv file named *Umich Dataset.csv*. There are 18 variables and 57170 samples in this dataset. The samples whose 'isclosed'=1 is the train set while those 'isclosed'=0 is the test set. The train set has 45507 samples and the test set has 11663 samples. The 'iswon' variable is the sales outcome we need to predict. The variables and definitions are listed in table 1.

2.2 Description of the code

There are several scripts submitted. Scripts, named as *feature_selections_*.ipynb,* contain different methods of feature selection, including using r squared and correlation coefficient to select highly related features used in later model training. In addition, the script named as *exploratory and ML models (containing the final model).ipynb* contains data exploration, data cleaning and preprocessing, all the machine learning models we tried, parameter tuning, and predicting on the test dataset. It is also the script producing the best and final model that could be used in practice.

2.3 Data Exploration

We drew bar plots and histograms to explore if the distribution of the features are related to the sales outcome (‘iswon’ variable). Chi-square tests are applied to explore if the categorical variables are significantly associated with the sales outcome.

2.4 Preprocessing the data

2.4.1 Drop the missing values

Because there are over 50,000 samples while less than 1,500 samples have missing values, We drop the missing values directly instead of imputation.

2.4.2 Encode the categorical variables

We use OrdinalEncoder() to encode ordinal variables, which are ‘daysopen’ ('0-30 Days', '31-60 Days', '61-90 Days', '91+ Days') and ‘amount_bucket’ ('0-99', '100-499', '500', '501-999', '1,000-4,999', '5,000-9,999', '10,000-'). We use LabelEncoder() to encode nominal variables, which are ‘industry', 'title', 'salestype', 'sales_Department', 'PipelineQualityModel'. We did not use OneHotEncoder() because there are too many unique values in some of the nominal variables, which will lead to high dimensional features.

2.4.3 Remove the outliers

During exploratory analysis, We found that the amount variable is extremely right-skewed. 93% amount values are less than $2,000. Therefore, I drop out the samples with amounts larger than $2,000 for the training dataset.

2.4.4 Add more features

We also add features that are important for outcome prediction. ‘amount250’ equals 1 when ‘amount’ is smaller than $250 else 0. ‘monthopen1’ is 1 when ‘Months_Open’ is less than or equal to 1 else 0. ‘Industry_selected’ is 1 when ‘industry’ is in a list of ['Manufacturing', 'Professional, Scientific and Technical Services','Health Care and Social Assistance'] else 0 because ‘iswon’ is more likely equals 1 when the potential customers are in these industries. ‘title_selected’ is 1 when ‘title’ is 'Customer Success Specialist' else 0 because the salespersons with this title are more likely to sell out the service. ‘sales_Department_selected’ is 1 when ‘sales_Department’ is 'Internal Accounts’ else 0 because this department is more likely to win the opportunity. ‘PipelineQualityModel_selected’ is 0 when ‘PipelineQualityModel’ is ‘Data Quality’ else 0 because this PipelineQualityModel is more likely to lose the sales opportunity.

2.4.5 Split and normalize the data

We split the whole dataset into train_val and test dataset based on the ‘isclosed’ variable. Then the train_val dataset is split into train and validation dataset using train_test_split() module with default parameters. MinMaxScaler() and StandardScaler() are used to fit and transform the training dataset and transform the validation dataset.

2.5 Prediction Models

We use a dummy classifier with ‘most_frequent’ strategy as a baseline. We use other 9 machine learning models to predict the outcome, which are Naive Bayes, MLP, LogisticRegressor, SVM, KNN, XGBoost, AdaBoost, GBDT, and RandomForest classifiers. Besides, we also built a manual prediction model. ‘iswon’ is predicted as 1 when ‘amount’ is less than $250 (‘amount250’=1) and open duration is less than one month (‘monthopen1’=1) else ‘iswon’ is predicted as 0. Accuracy is used as a matrix to measure the models’ performance.

3. Evaluation and Analysis

3.1 Feature selection by r-squared and correlation has low accuracy

We first did data exploration without date, and used r-squared to compare each pair of features and outcome. We noticed that for each pair, r-squared values are lower than what we thought could be (lowest as 0.001). There were no r-squared higher than 0.4. Only 5 features have r-squared over 0.2, and 9 over 0.1, out of 15 features. We also did exploration that includes date value by extracting *year*, *month*, *days*, *week of year*. We used correlation coefficient to apply feature selection (lowest as 0.0002). Only 9 out of 19 features are selected with absolute correlation coefficient over 0.1. For both feature selection methods, we used each set of them to train our models. However, for either models with all features or selected features, the accuracies were having average value around 0.65 (highest as 0.74), which are much lower than ideal accuracy as 0.85.

3.2 Amount and open duration are important features for outcome prediction

As shown in figure 1, we found that the distribution of amount and days open duration in the two outcomes (‘iswon’ =0 or 1). After the data was processed, we measured the correlation between the features and ‘iswon’ outcome. As shown is figure 2, there are 2 features related to amount (‘amount250’ and ‘amount_bucket’), and 3 features related to open duration (‘daysopen’, ‘monthopen1’, and ‘Months_Open’) among the top 10 features with the most absolute correlation with the service outcome.

3.3 Training one machine learning model on the whole dataset has low accuracy

We tried to train one machine learning model with default parameters on the whole dataset. However, as shown in figure 3A, the performance is poor. The best model is the Naive Bayesian classifier (accuracy = 0.72), whose accuracy is even lower than that of manual prediction (accuracy = 0.75).

We are curious why the performance is so poor. Therefore, we calculated the accuracy separately for the two outcomes. We found that the accuracy for successful outcomes (‘iswon’ =1) is much higher than unsuccessful outcomes (‘iswon’=0). Take the KNN classifier as an example, the accuracy of ‘iswon = 1’ is 0.96 while the accuracy of ‘iswon=0’ is 0.14, which means the model predicts many outcomes as 1 while the true outcomes are 0.

3.4 Adjusting decision thresholds could improve the accuracy of some classifiers

Since many unsuccessful outcomes are predicted as successful, we are thinking if the decision thresholds are too low for the models. Because the default decision threshold is 0.5 for the classifiers, we compute the change of accuracy with different decision thresholds. Figure 3B indicates that some models, such as the Logistic Regression model and Naive Bayes classifier, will have better accuracy with adjusted decision thresholds while some models do not, such as Adaboost and GBDT classifiers. However, the best accuracy achieved in this way is still less than 0.8, which is much lower than the company's goal (accuracy larger than 0.85). Besides, we also found that the decision thresholds with the best accuracy are not stable when classifiers are training on different feature combinations.

3.5 Training two models on split datasets have much better performance

Because the classifier training on the whole dataset has poor performance, we tried training two classifiers on the split datasets. We trained classifier 1 on the samples whose amount is less than $250 and trained classifier 2 on the remaining samples. In the predicting process, we used classifier 1 to predict the validation data whose amount is less than $250 and used classifier 2 to predict the validation data whose amount is larger than $250. Then, we combined the predicted results and calculated the accuracy score. As shown in Figure 3C, the best classifier with default parameters has an accuracy score as high as 0.84.

XGBoost and GBDT classifiers are the top two classifiers with the highest accuracy. We tuned the parameters of these two models. The best accuracy could be as high as 0.85. We used an XGBoost classifier with the tuned parameters to predict the test data and saved the results in the file named *output.csv*.

4. Related work

Companies adopting data-driven decision making tend to be more productive and more profitable than competitors. We reviewed some articles related to sales forecasting and sales outcome prediction to help us understand how to do feature selections and model selections at the business level. Marko Bohanec et al provide comprehensive explanations for the random forest model in sales forecasting [1]. Michael Giering predicted retail sales using customer demographics at the store level [2]. Neda Khalil Zadeh et al forecasted sales of PDCs with ARIMA methodology, neural network, and an advanced hybrid neural network approach [3]. Yuta Kaneko et al used a deep learning model with L1 regularization that achieved high accuracy[4].

5. Discussion and Conclusion

During the exploration of sales outcome predicting, we found that *amount* and *open duration* are the most important features. If an *opportunity* (the company Service Express names it as *opportunity*) has an amount less than $250, it is quite possible to have a successful outcome in one month (the company calls this *win the opportunity*). While if the *opportunity* has an amount larger than $250 and does not succeed in one month, it is very likely to lose the opportunity, which means Service Express needs to point their sales leadership, try to intervene, and provide assistance to help win the opportunity.

Training separate classifiers on split datasets based on whether the amount is less than $250 would have accuracy as high as 0.85. However, there are limitations of this model:

(1) In the training process, we excluded the data whose amount is larger than $2,000. The classifier may have a higher bias when predicting on samples with amounts larger than $2,000.

(2) We trained the classifier on data whose opportunity has been closed. The open duration does not change. However, in practice, the opportunity is not closed and the open duration keeps changing, which may lead to bias in the predictions.

In this project, we realize that data exploration and feature engineering sometimes are more important than training classifiers. We noticed that most successful outcomes had amounts less than $250 and split the dataset based on it. We add additional features based on the observations during the exploration stage, which are important indicators for prediction. At first, we trained the classifier on the whole dataset but the accuracy is low. Training separate models on different subgroups has much better performance.

In the future, we would communicate with Service Express and try to add more features to the dataset if possible. We could try more splitting methods on the dataset such as splitting based on ‘quickadd’, which is also an important feature for prediction. We could also try splitting on different combined conditions such as splitting based on ‘amount’ and ‘open time’, or ‘amount’ and ‘quickadd’.

6. References

[1] Bohanec, M., Borštnar, M. K., & Robnik-Šikonja, M. (2017). Explaining machine learning models in sales predictions. *Expert Systems with Applications*, *71*, 416-428.

[2] Giering, M. (2008). Retail sales prediction and item recommendations using customer demographics at store level. *ACM SIGKDD Explorations Newsletter*, *10*(2), 84-89.

[3] Khalil Zadeh, N., Sepehri, M. M., Farvaresh, H. (2014). Intelligent Sales Prediction for Pharmaceutical Distribution Companies: A Data Mining Based Approach. *Hindawi Publishing Corporation*, 420310

[4] Y. Kaneko, K. Yada. (2016). A Deep Learning Approach for the Prediction of Retail Store Sales. *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), Barcelona, 2016, pp*, 531-537

7. Appendix

Variables	**Definitions**
id	Unique Opportunity Identifier
quickadd	Opportunity type indicating that the customer wanted a quick change to their contract
isclosed	Indicates which opportunities are closed and can be used for training or which opportunities are open and need to have a prediction
amount	$ value of the opportunity
industry	Industry associated with the customer
title	Title of the sales person
salestype	We have two sales types, "N" (New) which means it would be a new contract and "A" (Add) which means we're adding to an existing contract
daysopen	The number of days the opportunity was/is open since it was created
amount_bucket	Related to the Amount field, bucketing the $ amounts
meetingcount	The number of meetings held with the customer about this opportunity
sales_Department	The sales department that the title rolls up to
PipelineQualityModel	This is an indicator of the quality of the data related to the opportunity. High quality means it's actively being updated, Data Concern means most items are getting updated on a timely basis and data quality means the sales person isn't keeping their information up to date.
Won_First_Opp	This is an indicator of whether or not we won the first opportunity with the customer. Customers can have multiple opportunities and if we win the first one, we tend to have a higher win % on future opps.
Months_Open	Number of months the opportunity was/is open.
Partner_Flag	This indicates whether or not one of our partners brought this opportunity to us.

SDR_Opportunity	This indicates whether or not a sales development rep created this opportunity
Opp_Created_Date	The date the opportunity was created.
iswon	This is the target variable, iswon = 1 means we won the opportunity and iswon=0 means we lost it.

Table 1. The features and description of the dataset.

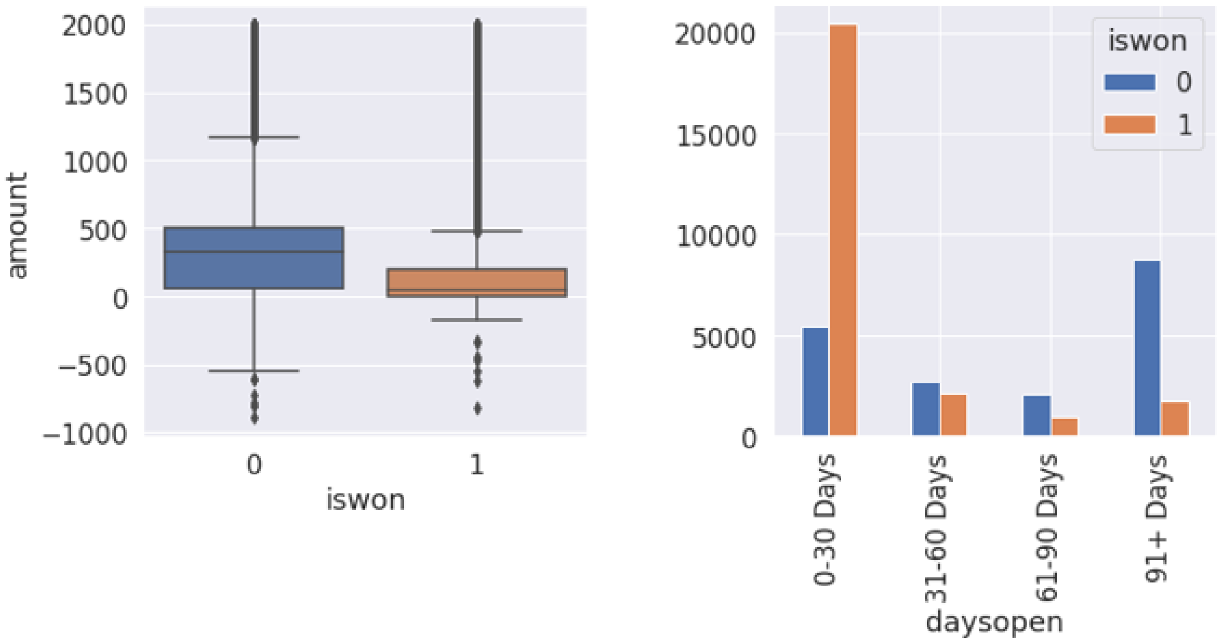

Figure 1. The distribution of amount (left) and open duration (right) are different between two sales outcomes. Most successful outcomes (‘iswon’ = 1) have amount less than $250 and open days less than 30 days.

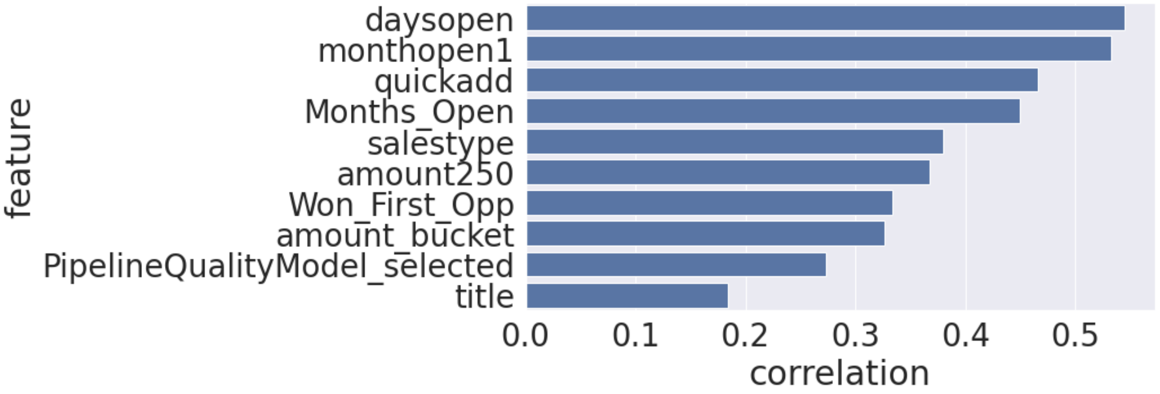

Figure 2. The top 10 features with most absolute correlation with outcome. 3 of them are related to open duration (‘daysopen’, ‘monthopen1’, ‘Months_open’). 2 of them are related to amount (‘amount250’, ‘amount_bucket’)

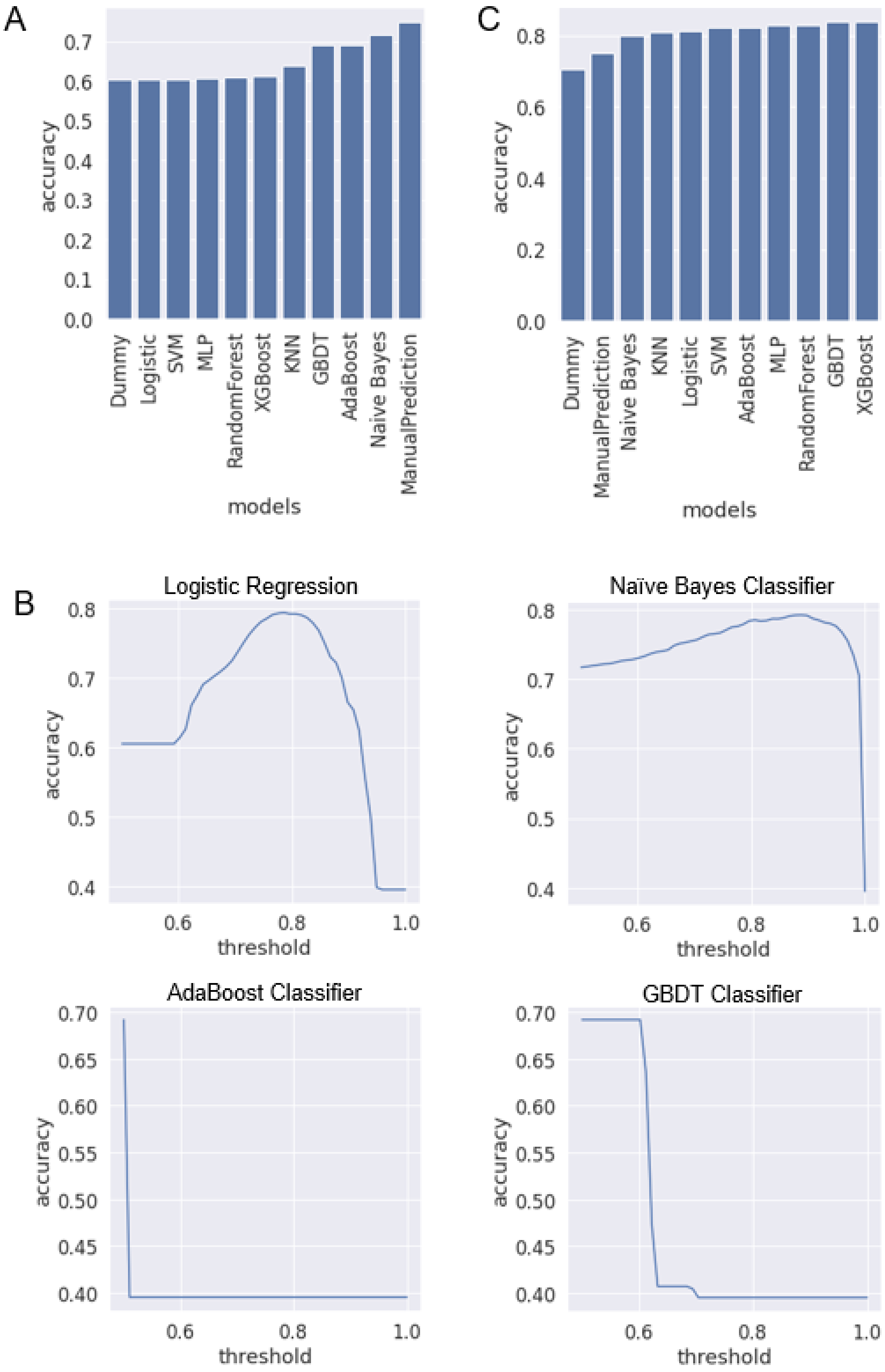

Figure 3. Training predicting classifiers. (A) The performance of machine learning models training on the whole dataset. The accuracy of all the models are less than Manual predictional models. (B) Some classifiers will have better accuracy with adjusted

decision threshold. Logistic Regression and Naive Bayesian model will have better accuracy when the decision threshold is larger than 0.5. In contrast, the AdaBoost classifier and GBDT classifier have the best accuracy with a decision threshold of 0.5. (C) Training two separate classifiers with default parameters on split datasets has better accuracy. The best accuracy score can be as high as 0.84.