SunnyBridge Data Science Case Study

The final project report of
Machine Learing

by
Yedi Zhang (14751580)
Tongliang Deng (67062986)
Pengfei Gao (25103453)

School of Information Science and Technology
ShanghaiTech University

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The application of machine learning appears in all aspects of our lives, including the company's forecast of the market. The traditional marketing approach is straightforward, with little preprocessing. This approach consumes a lot of manpower and financial resources and has a very low rate of return. If companies can make full use of historical data and predict customer's behavior before marketing, companies will reduce expenses and increase profitability. Machine learning is a good way to do these things.

1.2 Objective

Our objective is to determine which set of customers the marketing firm should contact to maximize profit. For each customer that can provide a response, what is the profit he may give to the conpany.

1.3 Organization of the Report

The rest of the report is organized as follows: In Chapter 2, we introduce the process of data preprocessing. In Chapter 3, we propose our two plans to solve the problem. In Chapter 4, we analyze the experimental results. In Chapter 5, we conclude the report.

Chapter 2

Data Preprocessing

2.1 Missing Data

Figure 2.1 shows the number of NaN(unknown) in each attribute in "Data-Training.csv". The existence of missing data may blur the real pattern hidden in the data. So the first step we need to do is "imputation". We use chi-square test to analysis the independence between existenting data and missing data in these attributes. Then we use random forest to predict the unknown variables. More details will be shown in Chapter 3.

2.2 Imbalanced Data

There are 7310 data points which are been labeled as "no", while only 827 data points are labeled as "yes"(almost 9:1). Obviously, the dataset is imbalanced. If we use this dataset straightforward, any classifier will achieve a high accuracy simply by reporting "no" to every new customer. As a result, we need take measures to address class imbalance in dataset.

There are some ways which have been proposed to solve this problem.

- Over-sampling, is an approach that add copies of under-represented class to the dataset.
- Under-sampling, is an approach that delete instances from over-represented class.
- Mix-sampling, is a mixture of over-sampling and under-sampling.

In our implementation, we use `resample` module from `sklearn.utils` to handle imbalanced data problem.

Figure 2.1: The number of unknown

```
Out[8]: schooling          2637
        custAge            1992
        default            1606
        day_of_week         782
        housing             182
        loan                182
        profession           71
        marital              10
        responded             0
        contact               0
        month                 0
        profit                0
        previous              0
        poutcome              0
        emp.var.rate          0
        cons.price.idx        0
        cons.conf.idx         0
        euribor3m             0
        nr.employed           0
        pmonths               0
        pastEmail             0
        campaign              0
```

It is worth noting that we use ROC(Receiver Operating Characteristic) curve and AUC(Area Under Curve) as the performance measurement. The traditional measurement of performance(i.e. test accuracy rate) is not suitable here as it may tent to fit the majority class.

Chapter 3

Approach

3.1 Data Preprocessing – recover the missing data

In our recovering process, we notice that the first six features: `custAge`, `profession`, `marital`, `schooling`, `housing`, `loan` are all the information about customers. While the rest information are all about society, time, company. So we recover the customer's personal information with the first 6 features one another, and then recover the missing data in `day_of_week` from the rest features. What's more, we find out the two features `pmonths`, `pdays` show exactly the same thing, so we omit the latter, remain the feature `pmonths` only. Here we mainly use random forest to recover the missing data.

3.2 Plan A

3.2.1 Overall Approach

The output of the problem is whether or not to carry out marketing for each candidate. Therefore, we regard this project as a classification problem. In the existing dataset, we label each data point as showing of Table 3.1.

Table 3.1: Label Data Points

target		response	
		0	1
profit	$\geq$30	-	1
	<30	-	0
	NA	0	-

Figure 3.1: Chi-square test result

```
Out[5]: [['custAge', 4.3313350431095419],
         ['profession', 0.47919301334167835],
         ['marital', 0.013970747141167994],
         ['schooling', 1.8540152446885148],
         ['default', 0.098122720101061975],
         ['housing', 0.80782457579870537],
         ['loan', 0.81684168242222432],
         ['day_of_week', 0.8845089761991175]]
```

- For data with a response of 0, label it as 0. These people do not need market.
- For data with a response of 1 and a profit of less than 30, label it as 0. Although these people responded but did not have a gain for the profile, they do not need market.
- For data with a response of 1 and a profit of more than 30, we will label it as 1. These people can bring us benefits.

According to these labels, we learn a hypothesis $h_0(x)$ whose input is features of a user and output is whether need to market to the user. We use four machine learning algorithms which are Random Forest, Decision Tree, Logistic Regression and Gaussian Naive Bayes. We evaluate the performance of them and select the best one.

3.2.2 Data Preprocessing

Figure 3.1 shows the chi-square test result between features and the label shown above. The chi-square value shows the importance of unknown pattern for our model. For example, the `schooling` attribute, there are 2637 missing lines marked as "NA" or "unknown". When applying chi-square test the result is 1.85, which shown we cannot ignore the missing values. Therefore, we use the rest known data to impute the missing terms. Whats more, the other attributes are all in this case except for `default` and `marital`. Since there are only 10 marital-unknown samples, so we can just delete these 10 samples. However for the `default`, even though its chi-square value is quite small, there are too many samples, about 1600. If we delete all the default-unknown samples, since the total amount of "yes" response is very small, only have one client. Then all the rest samples' default value is "no". So we can not just drop the unknown samples in this field(variable). So instead, we consider the "unknown" as an another feature type of `default`.

Then we recover all the other missing data with the method introduced in 3.1.

Figure 3.2: Chi-square test result

```
K_value
```

```
[['custAge', 3.3791997129207223],
 ['profession', 0.49524349766965337],
 ['marital', 1.0610242981939131],
 ['schooling', 2.9712540615799292],
 ['default', 0.11314660013693342],
 ['housing', 0.962579685431598],
 ['loan', 0.52609284124150368],
 ['day_of_week', 2.0569434236488511]]
```

3.3 Plan B

3.3.1 Overall Approach

Although $h_0(x)$ can tell us whether need to market to a user, it can not predict the specific profit value. Therefore, we propose another approach: the combination of classification and regression.

1. We classify the data according to the value of the response, from which we learn a binary classification function $h_1(x)$. The input of $h_1(x)$ is the feature of a user, and the output is whether the user will respond.

2. We use the data in the dataset whose value of response is 1 as the input and learn a hypothesis $h_2(x)$ as a regression problem. The input of $h_2(x)$ is a user's feature, and the output is the profit that the user can generate.

3. Use h_1 to predict whether a user will response.

4. Use h_2 to predict the profit for all the users.

We use four machine learning algorithms which are Random Forest, Decision Tree, Logistic Regression and Gaussian Naive Bayes. We evaluate the performance of them and select the best one: Random Forest.

3.3.2 Data Preprocessing

Figure 3.2 shows the chi-square test result between responded and other user features. In this case, the smallest chi-square value is `default`: 0.11. It shows that the missing data in all the other features should be manually

recoverd. However, out of the same reason as above, we cannot also just ignore the unknown samples in `default`. So in this case, we keep all the unknown samples, and use the same method introduced in 3.1 to recover the miss data by Random Forest.

Chapter 4

Experimental Result

4.1 Plan A

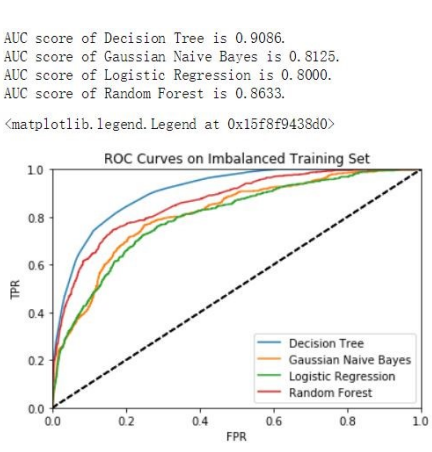
We divided the training set into two parts, one for training and one for testing. Figure 4.1 shows the ROC Curves on imbalanced Training Set and ROC Curves on Test Set. Although Decision Tree has a good performance in imbalanced training set, it does not perform well in test set. Random Forest has been shown as the best model. Figure 4.2 shows Lift Chart on imbalanced training set and the Lift Chart on test set. The result also shows that **random forest** is the best model.

4.2 Plan B

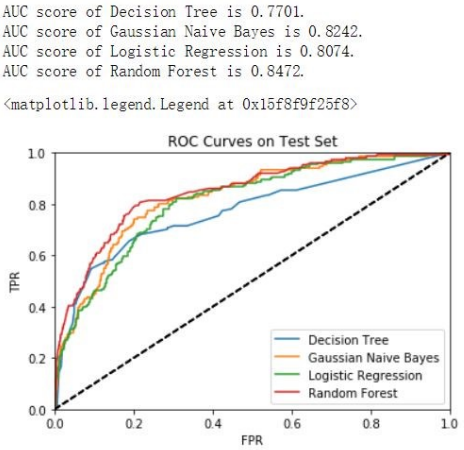
We divided the training set into two parts, one for training and one for testing. Figure 4.3 shows the ROC Curves on imbalanced Training Set and ROC Curves on Test Set. Although Decision Tree has a good performance in imbalanced training set, it does not perform well in test set. Random Forest has been shown as the best model. Figure 4.4 shows Lift Chart on imbalanced training set and the Lift Chart on test set. The result also shows that random forest is the best model in classifying whether users will respond.

Figure 4.5 shows top ten features that influence profit and responded most in h_0 and h_1 respectively.

After that, we use the data in the dataset whose value of response is 1 as the input and learn a hypothesis $h_2(x)$ as a regression problem with `sklearn` package. We divide it(827 samples) into training and test set. And the Linear Regression model Figure 4.6 shows 83.3% accuracy rate in this test set. Then we use this LR model to predict all the users' potential profit.

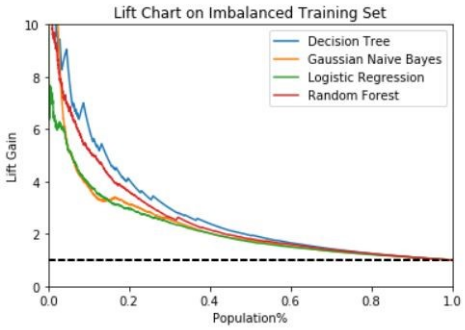


(a) Imbalanced Training Set

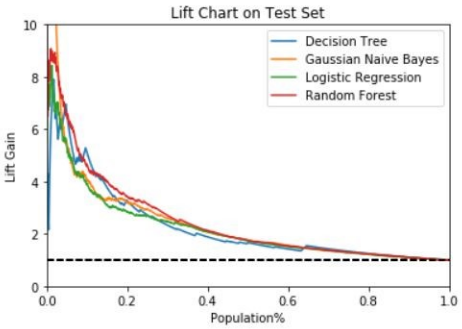


(b) Test Set

Figure 4.1: ROC Curves



(a) Imbalanced Training Set

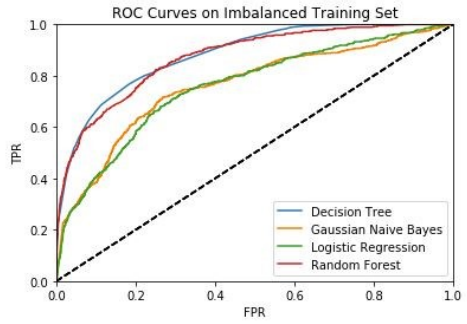


(b) Test Set

Figure 4.2: Lift Chart

AUC score of Decision Tree is 0.8820.
AUC score of Gaussian Naive Bayes is 0.7598.
AUC score of Logistic Regression is 0.7577.
AUC score of Random Forest is 0.8724.

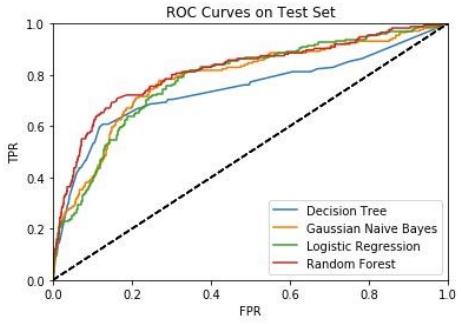
<matplotlib.legend.Legend at 0x2404f24dda0>



(a) Imbalanced Training Set

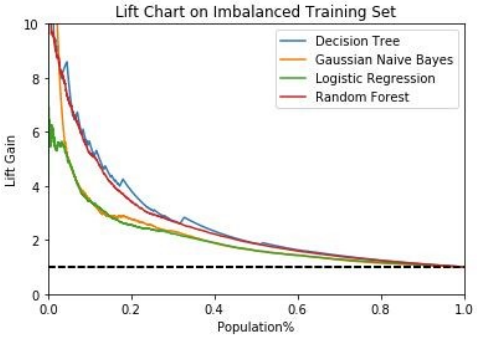
AUC score of Decision Tree is 0.7430.
AUC score of Gaussian Naive Bayes is 0.7831.
AUC score of Logistic Regression is 0.7832.
AUC score of Random Forest is 0.8154.

<matplotlib.legend.Legend at 0x2404f301a58>

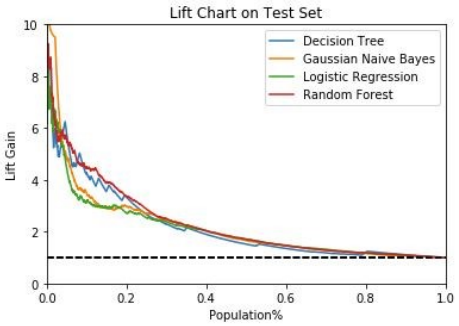


(b) Test Set

Figure 4.3: ROC Curves

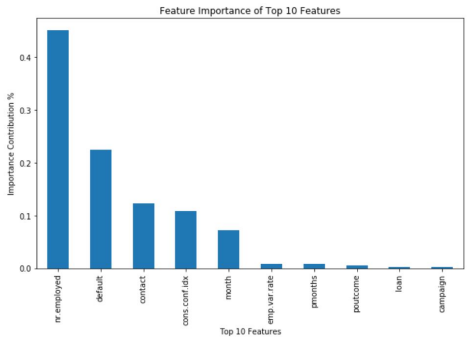


(a) Imbalanced Training Set

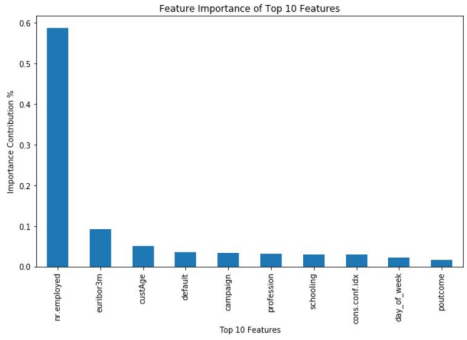


(b) Test Set

Figure 4.4: Lift Chart



(a) h_0



(b) h_1

Figure 4.5: top ten features

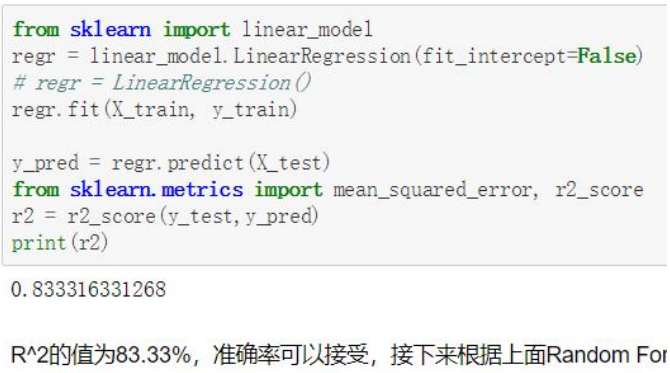


Figure 4.6: Linear Regression of profit function

Table 4.1: Comparison between PlanA and PlanB

DataSet	PlanA	PlanB	True Profit
Training data	247240.77	380796.45	127879.0
Testing data	31923.08	44526.26	–

4.3 Comparison between Plan A and Plan B

Table 4.1 shows that the comparison of the benefits of the learned model from Plan A and Plan B. And we found the following two things:

1. if use our model before the market, the final profit will be higher than the current profit in this DataTraining data. Since the original benefit of training data is 127879.0. If we use Plan A, profit will reach 247240.77 and if we use Plan B, profit will reach 380796.45.

2. Apparently, based on the same profit approximate value, Plan A and Plan B have a different profit outcome. Using Plan B, it can reach 44526.26 profit on testing data. However, using Plan A, it can only reach 31923.0 profit on testing data.

Chapter 5

Conclusion

In this project, we first tried to classify directly according to whether market to a customer or not(i.e. Plan A). Even though this method has already give an output result, some useful information in this training data set is not used. In other words, we ignored the relationship between users' information, company, society information and specific profit value. So we consider Plan B to utilize these information: We tried to classify according to responsed first, and then predict the profit of the users, and finally decide whether to perform the market action(i.e. Plan B). In the training process, we first recoverd the missing data manually, and tried various methods to train such as Random Forest, Decision Tree, etc., and use ROC and Lift Chart to evaluate the performance. The experimental results show that Random Forest can achieve good results in classification and prediction. Finally, the experimental results show that Plan B can bring the greatest benefits for the company. The maximum revenue in the DataPredict.csv file we forecast is 44526.26. For each customer, whether to market it or not, you can refer to the results in "market.csv".

Note that, the code of Plan A, B can be found in *insurance_PlanA.ipynb* and *insurance_PlanB.ipynb* respectively. The comparison result is also in the *insurance_PlanB.ipynb*. What's more, we find out the largest profit will have a little difference every time when we run the code. So we list another running result in *insurance_PlanB_0.ipynb*. Even though the different result, the largest profit always comes from Plan B.

So here we submit the best result we learn, i.e., the largest profit is: 44526.26. And the corresponding market result is saved in "market.csv".