PSTAT 234 Project Report

Alibaba Advertisement CTR Algorithm Optimization

Deng, Haohua
Li, Wenjing (Sarah)
Wang, Yiru
Wu, Yuyang (Alex)

Supervisors: Prof. Oh Sang-Yun

June 12, 2021

Contents

Abstract

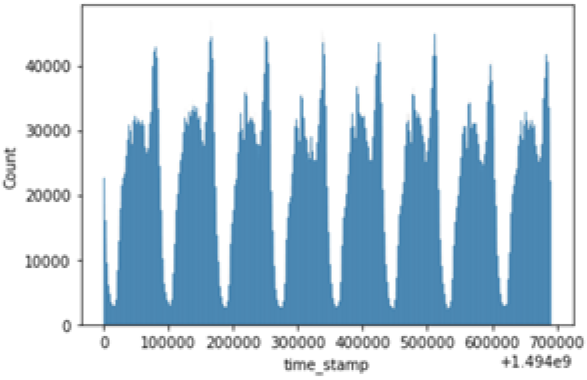
Coming to the second decade of the 21st century, data has become the key to business. The price of data is numerous. Companies depend on data to maximize their revenue. In the advertisement industry, how to put one advertisement on the market is critical. The data itself creates profit along. Our project will be using a million-level data from Alibaba advertisements to optimize the algorithm for putting advertisements on the market. We will build our model based on neural networks and PCA. We have considered many other models like–; however, depends on the data structure and function, we found that neural network and SVM would generate the best results.

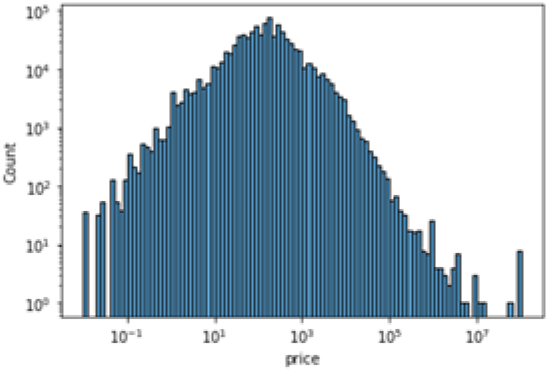
1 Data Exploratory Analysis

Originally, we have four very large data sets each of which contains the information of user, the advertisement and the underlying product. By cleaning and merging the data, we found that we got several millions rows of data with 17 explanatory variables and 1 response variable. After finishing the exploratory data analysis, we decided to keep 11 explanatory variables and 60,000 rows of data.(Due to the computation power of our devices.) Here is the list of the original variables:

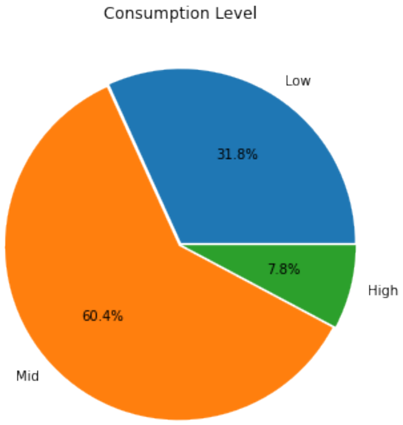
1. user_idmasked user id
2. time_stamptime_stamp
3. adgroup_idmasked advertisement id
4. pidmasked position of the advertisement
5. clk0 means non-click the ad, 1means click the ad;
6. cms_segidusers' ID in their group
7. cms_group_idusers' group ID;
8. final_gender_code1 is male, 2 is female
9. age_level5 level of age range
10. shopping_levelshopping level, 1 is low, 2 is medium, 3 is high;
11. occupationwhether the user is college student, 1 is yes, 0 is no;
12. new_user_class_levelusers' city class level;
13. cate_idmasked product category ID
14. campaign_idmasked advertisement plan ID
15. customer_id:masked advertisement ID
16. brandmasked brand ID
17. price: price of the product;
18. btag: the behavior toward this product, 0 for browsing, 1 for adding to cart, 2 for adding to favorite, 3 for purchasing;

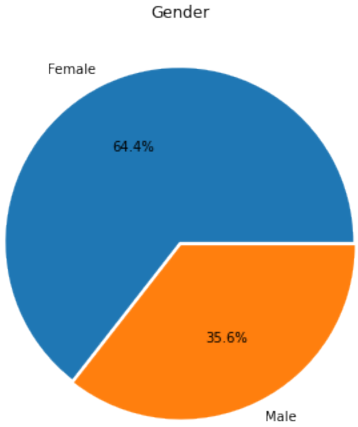
Since the masked ID is hard to explain and we noticed that there are many unique value of the masked ID variables, we dropped a lot of masked variables such as 'adgroup_id', 'cms_segid', 'campaign_id', 'customer_id', 'brand'. The masked variables we kept are 'pid', the position of the advertisement because there are only 2 different types of position, we made it to binary(0 or 1) type variable, 'cm_group_id', the masked users' group ID, because there are only 13 types of group which represented by 1 to 13, 'cate_id', the masked product category ID, since it denotes the product category which will be very helpful in prediction and there are only 292 types of it, so we made 'cate_id' as categorical variable.

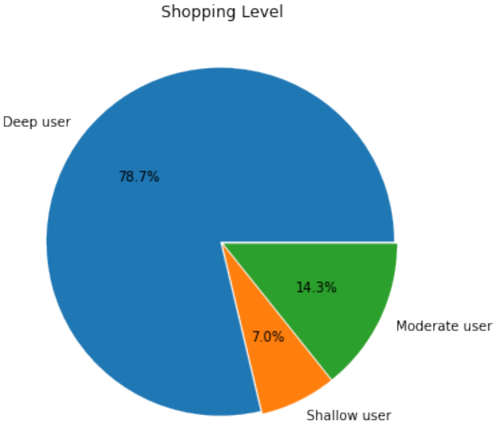




These two plots above shows that these datasets are randomly selected and large enough to represent the real-world situation. Below is a graph of the number of advertisement clicks over time, which shows a repeated changing model of 8 cycles. That fits the daily schedule of most people, working in the day and resting at night. Another graph of the number of goods over prices shows most goods have a price between 10 and 1000 RMB (about 1.5-154 dollars) and the number of goods has a decreasing trend in either direction of the goods.







These three pie charts help generate a better understanding of the data. Considering that these users and goods are randomly selected by the Alibaba company on an 8-day basis from Taobao, the figure we generated from these datasets could in fact represent the backgrounds of Taobao website's users. Based on the following analyzed results, it is easy to find out that most of Taobao's users are female instead of male and between 20 and 59 years old, which fits our stereotypes. Also, most of them are deep users, non-students and have a mid or low consumption level. It is also interesting to see that most of Taobao's users are from Second-tier or lower-tier City instead of First-tier City.

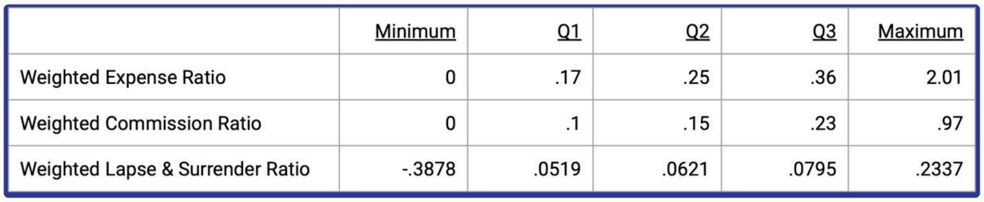
Since all the four datasets introduced above are linked with each other by the user ID or the adgroup ID, we decided to first manage the dataset into one single file for further analysis. After merging the dataset, we abandoned several features of the dataset for which these descriptions are not supporting our target of research, including the user ID, adgroup ID, time stamp, brand, and so on, and only left for 9 variables by the end. Besides, since the dataset is too large for the analysis, and our goal for the project is to predict the click through result for the display advertisement, we randomly select 60000 users and their corresponding information from the original dataset as our final version of data for analysis.

2 Logistic Regression

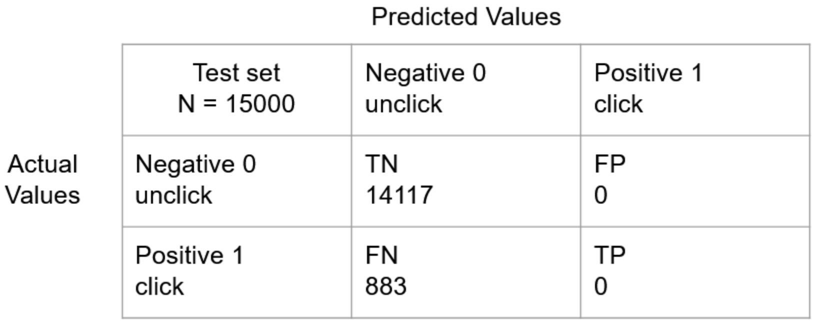
We first started with a straightforward model, logistic regression, and tried to see how much signal to pull out using basic methods. We simply split the data into training and testing set with a test size equal to 0.25. The result of the logistic regression is surprisingly not bad. It has an accuracy of 0.9411 which means we successfully predict whether the user will click the advertisement. However, the logistic regression classified everything as 0 which is nonclick. Therefore, the logistic regression didn't have predicting power at all, since it can not capture the click features. So we need to move on to a more advanced method to see if we can improve our performance.

3 Neural Networks

We employed the basic form of neural network which is a 3 layer fully-connected neural network, with ReLU activation function between each layer to our problem. After testing different parameters such as the number of neurons in each layer, the number of the layer and the learning rate of the optimizer, we found that the basic form of neural networks had very similar performance and can not improve much by fine-tuning the parameters. The network reached to its best performance and kept staying at that level after the first few epoch. The accuracy in the training set is around 0.946 while the accuracy in the validation set is 0.9412 which is slightly higher than logistic regression method.



	Minimum	Q1	Q2	Q3	Maximum
Weighted Expense Ratio	0	.17	.25	.36	2.01
Weighted Commission Ratio	0	.1	.15	.23	.97
Weighted Lapse & Surrender Ratio	-.3878	.0519	.0621	.0795	.2337



		Predicted Values	
	Test set N = 15000	Negative 0 unclick	Positive 1 click
Actual Values	Negative 0 unclick	TN 14117	FP 0
	Positive 1 click	FN 883	TP 0

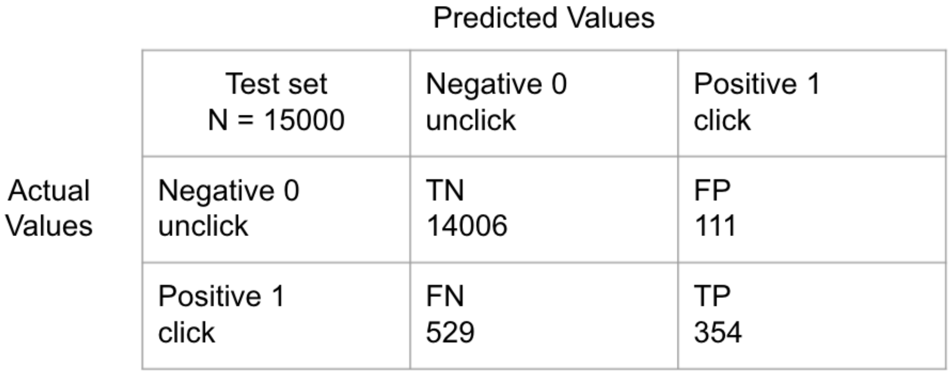
Our speculation is that the general rule of whether the user will click the ad is easy to capture by the models, so even the simplest logistic regression method can have relatively high accuracy. Since we dramatically decrease the data size, our model can only capture the most obvious and general rule and is not sensitive enough to capture some abnormal users' behavior. The ways to improve may include:

- Adding the data size, since originally, there are several millions of data.
- Using more advanced neural network such as Deep Interest Network (DIN) which introduces a local activation unit to adaptively learn the representation of user interests from historical behaviors w.r.t. given ads. DIN can improve the expressive ability of model greatly and better capture the diversity characteristic of user interests.
- Including the masked ID variables since they may contain essential information that determine whether the user click or not while all other variables remain the same.

4 Support Vector Machine

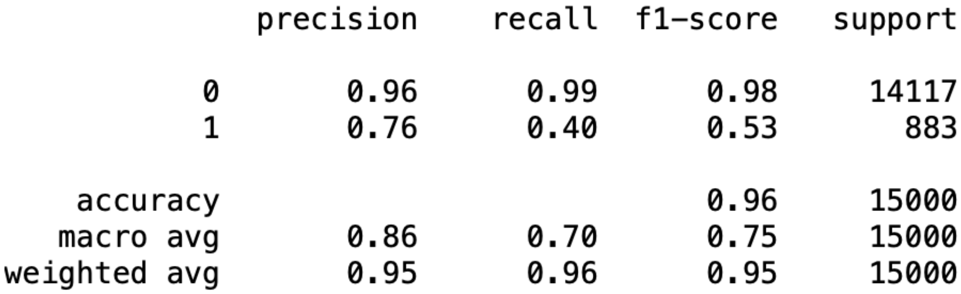
Support vector machine is one of the machine learning methods, also known as SVM, and it is a method used for two group classification problems. Usually there are two types of the SVM, first is the simple SVM, and it is generally used for linear classification problems, where the data points can be divided into two parts with a straight line. The other type is using kernel, and it is more broadly used, for it is more flexible and can deal with non-linear distributed data or hyperplane datasets.

In our research, we first set the response factor as the clicking data, and all other features as the variables for the prediction. Then we randomly split the data into two parts with 75% of the training data and 25% as the testing data for comparing the prediction result. Since we target to forecast the click outcomes using features in the given dataset, there will only be two results for the click, whether a click or unclick, so we apply the SVC algorithm under the SVM method, which is known as support vector classification approach. We then apply the kernel SVM to fit the features into the model, since we have a high dimensional dataset with 10 variables and the data points are randomly distributed based on the user's favorable, and kernel with gaussian radial basis function will help us have a better result. Under this step, in order to find the index value of parameters, we test for the possible values and find the highest true positive rate results for the model. After fitting the best resulting values into the SVM model, we get the prediction results of the confusion matrix and classification report as follows.



		Predicted Values	
	Test set N = 15000	Negative 0 unclick	Positive 1 click
Actual Values	Negative 0 unclick	TN 14006	FP 111
	Positive 1 click	FN 529	TP 354

The confusion matrix generated from the SVM model is shown above, which describes the performance of a classification model. The horizontal columns are the actual result from the test set, and the vertical rows are the predicted value from the model. In our case, the true positive case(TP) 354 happens when both actual case and predicted case get the result on click, which the predicted yes also corresponds to the actual yes; the true negative(TN) result 14006 means we predict the result as unclick while the actual result is also unclick; the false positive(FP) value of 111 means we predict the result as click while the actual result is unclick for the users; and the last value 529 is false negative value(FN) where we predict it as unclick but the real case showed it is click.

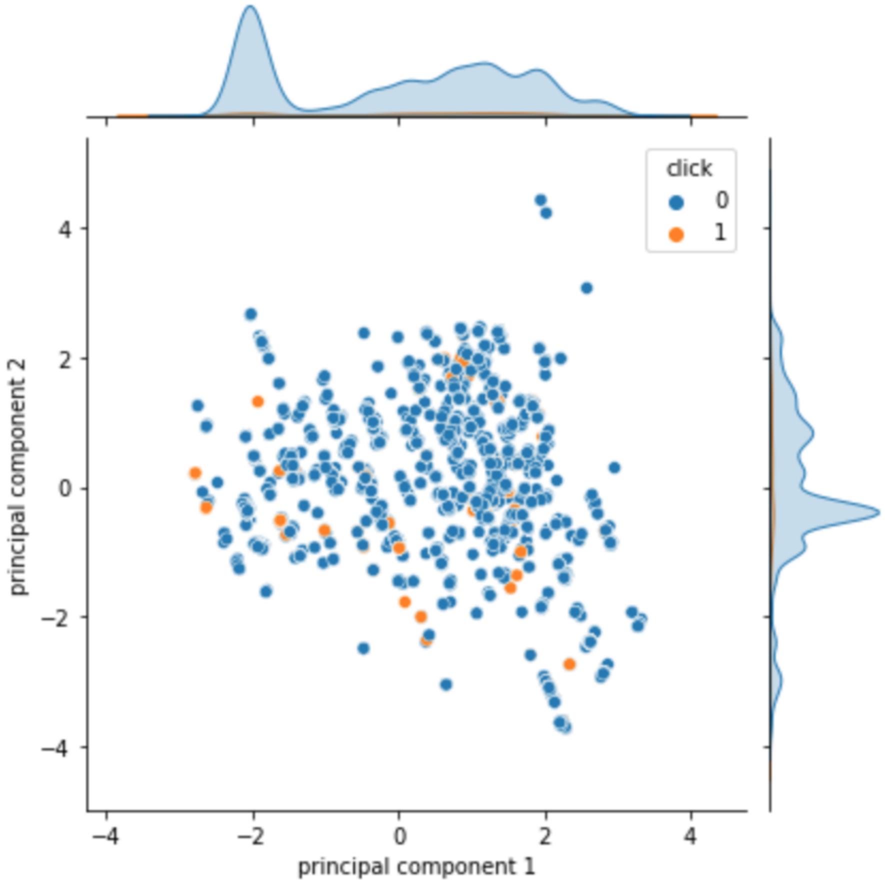


```
              precision    recall  f1-score   support

           0       0.96      0.99      0.98     14117
           1       0.76      0.40      0.53       883

    accuracy                           0.96     15000
   macro avg       0.86      0.70      0.75     15000
weighted avg       0.95      0.96      0.95     15000
```

We then compute the relevant rates for classification report based on the results of confusion matrix. The first term is precision, and it is a measurement of how many results are correctly classified among this class, and the results show that we are 96% correct in predicting the unclick situation, and are 76% correct in predicting the click situation for the displaying advertisement. The second term recall is also known as the true positive rate, which measures how often we predict the result as positive when the actual result is positive, and the rate for unclick is 99%, while for click, the rate is 40%. The next term is F1-score, and it is a harmonic mean between recall and precision, which is a measurement of the total accuracy of the classifier, and a larger value means a better prediction. For our result, the F1-score is lower for the click rate, and we have a better prediction towards the unclick advertisements. The last column in the table is support, which shows the number of actual total number in the data, and we can see the total number of unclick is 14117 and is way larger than the total number of click cases 883, and this can explain why we have a lower accuracy in predicting the click cases. On the vertical side, there are three terms calculated from the confusion matrix. The first term accuracy is an overall accuracy for the prediction combined with unclick and click cases, and we are 96% correct in the prediction result. The other term macro average is the overall average for precision and recall, and the last rate is the weighted average computed combined with the total number weight for the dataset.



After getting the numerical result from the SVM model, we then reduce the dimension for the x from total 10 variables to 2 dimensional using PCA method after scaling the dataset, and got the joinplot based on the test dataset as shown above. From the scatter plot, we can see that there are more data points for the unclick cases, and the points are randomly distributed as we stated before. This graph is a better explanation of the predicting results where the click cases are harder to forecast for the low weights, and the reason can be we have a relatively low relationship between variables and response, and the dataset is also limited for displaying advertisement prediction.

5 Issues and Future Study

The limitation in this project is the computational power of our devices. Since we are studying online, we have no access to university facilities. Without a device with large disk and graphics card, we are not able to process 26 million data points especially when half of our group members' laptops have no ability to download TensorFlow. Based on emputber's coma lrelatively small data set, 60 thousand data, SVM is the best model in this case. It has the best accuracy and fastest performance comparing to logistic regression and neural networks.

To continue our research, we will try to use a better device and build Deep Interest Evolution Network and complete a better Deep Cross Network.

6 References

1. *“Deep amp; Cross Network (DCN) nbsp;: nbsp; TensorFlow Recommenders.” TensorFlow, www.tensorflow.org/recommend*

2. *Gai K, Zhu X, Li H, et al. Learning Piece-wise Linear Models from Large Scale Data for Ad Click Prediction[J]. arXiv preprint arXiv:1704.05194, 2017.*

3. *Guorui Zhou, Chengru Song, Xiaoqiang Zhu, et al. Deep t ck-Through Rate Prediction.https://arxiv.org/abs/1706.06978.*

4. *Markham, Kevin. “Simple Guide to Confusion Matrix Terminology.” Data School, Data School, 3 Feb. 2020, www.dataschool.io/simple-guide-to-confusion-matrix-terminology/: :text=A%20confusion%20matrix%20is%20a,the%20*

5. *Wang, Ruoxi, et al. “Deep amp; Cross Network for Ad Click Predictions.” Proceedings of the AD-KDD’17, 2017, doi:10.1145/3124749.3124754.*

6. *Alibaba. Data. https://tianchi.aliyun.com/dataset/dataDetail?dataId=56.*