Web Economic Coursework Individual Report

Ziyang Liu
University College London
ziyang.liu.17@ucl.ac.uk

```
In [10]: sum(train['bidprice'] >= train['payprice']) \
             / len(train['bidprice']) * 100

Out[10]: 100.0
```

Figure 1: The probability of bidprice >= payprice

1 INTRODUCTION

In this coursework, we are required to explore and utilize the training, validation and test data downloaded from here. We also need to explore and compare different bidding strategies in real time bidding so as to decide the optimal one.

2 DATA EXPLORATION

2.1 Basic Statistics

I explore some further statistics such as CTR based on the provided training dataset and the results are summarized in Table 1.

From Figure 1, we can clearly observe that all pay prices in the provided training data are smaller or equal to corresponding bid prices. This indicates the training data merely contain successful biddings.

It is also interesting to observe that advertiser 2997 has the highest CTR (0.435%) among all the advertiser whereas the others all have CTR less than 0.1%. The advertiser 2997 merely use Android operating system whereas others use a mixture of operating systems. This indicates Android operating system must have some unknown advantages over other operating systems. One suggestion is that clicks are more straightforward to produce on the Android operating system than other systems. In addition, the advertiser 2997 has the lowest Cost(3129), avg CPM (62.80) and eCPC(14.42). There is also no correlation between the total cost and the CTR.

2.2 User Feedbacks

I subsequently investigate the impact of some of the features such as weekday on CTR for some of the advertisers. Due to limited space, I merely investigate these for some of the advertisers in the report.

Figure 2, 3 and 4 illustrate CTR of the advertiser 1458, 2997 and 3427 on each weekday. From the Figure 2, we can clearly observe the advertiser 1458 obtain highest CTR on weekday = 1. On the other hand, the highest CTR occurs on weekday = 2 and weekday = 6 when the advertiser is 2997 and 3427 respectively. In addition, the CTR distributions are very different, which indicates the advertiser behaves very differently when the weekday varies.

Similarly, Figures 5, 6 and 7 illustrate CTR of the advertiser 1458, 2997 and 3427 in each region. These figures indicate the highest CTR is obtained by the advertisers in different regions. The CTR distribution also varies when the advertiser changes. These results consequently suggest we might need to train the prediction models independently for each advertiser so as to result in a highly accurate model.

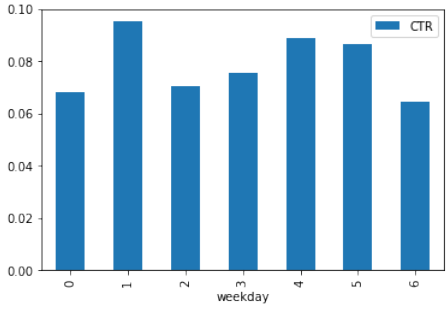

Figure 2: CTR of the advertiser 1458 on each week day

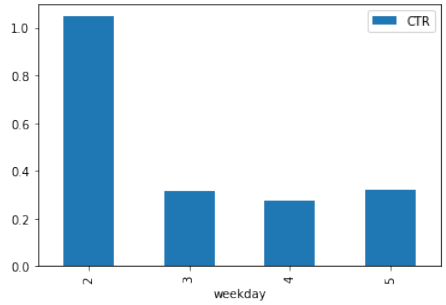

Figure 3: CTR of the advertiser 2997 on each week day

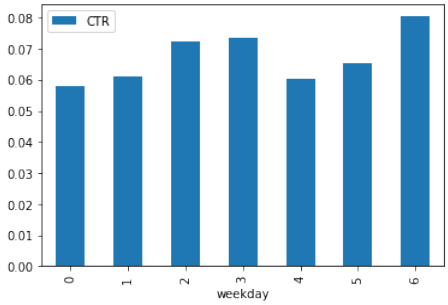

Figure 4: CTR of the advertiser 3427 on each week day

Table 1: Further statistics based on provided training dataset

Advertisor	Avg CPM	CTR	Clicks	Cost	Imps	eCPC
1458	68.99	0.078	385	33969	492353	88.23
2259	92.97	0.032	43	12428	133673	289.03
2261	89.66	0.033	36	9874	110122	274.27
2821	89.08	0.062	131	18828	211366	143.73
2997	62.80	0.435	217	3129	49829	14.42
3358	84.72	0.076	202	22447	264956	111.12
3386	76.77	0.070	320	34932	455041	109.16
3427	75.61	0.068	272	30459	402806	111.98
3476	76.95	0.060	187	23919	310835	127.91

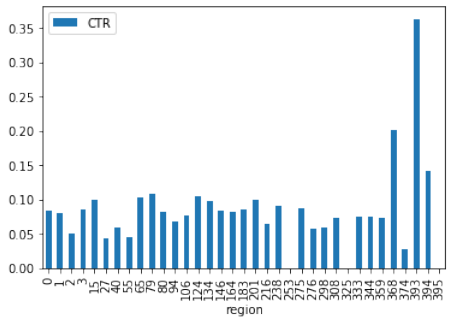

Figure 5: CTR of the advertiser 1458 in each region

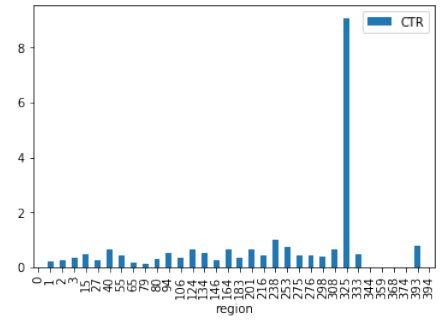

Figure 6: CTR of the advertiser 2997 in each region

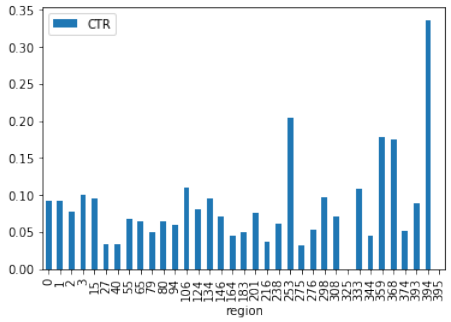

Figure 7: CTR of the advertiser 3427 in each region

3 MY BEST BIDDING STRATEGY

3.1 Bidding below max eCPC

For each advertiser, the max eCPC is obtained by using its cost divided by the observed number of clicks in the training set. The bid prices are subsequently calculated by multiplying the pCTR and the max eCPC. This strategy essentially tries to reduce the resulted eCPC by providing the maximum upper bound of eCPC. In addition, this strategy is non parametric, which means it is much easier to implement[2].

The pCTR is predicted by using logistic regression and I obtain the evaluation metrics results on the validation set (clicks:54, ctr:0.0663, cost:6250.043, avgCPM:76.74, eCPC:115.74). This indicates this strategy is actually worse performed than basic bidding strategies such as the constant bidding.

3.2 Predicting pCTR using xgboost

Logistic regression is a linear model and it does not allow us to learn the nonlinear features. This model also relies hugely on the feature engineering. I thus used an extreme gradient boosting regression tree model called "xgboost". The new model is able to automatically select features and learn the nonlinear features during training.

I also implemented a k fold cross-validation xgboost model and averaged the predictions in five folds. This consequently improved the accuracy of the model slightly. In the preprocessing step, I also had to sort the feature columns so as to solve the feature mismatch issues in xgboost.

After tuning the parameters including the base bid, I found the optimal base bid is 74 with the evaluation metrics results on the validation set (clicks:165, ctr:0.138, cost:6123.062, avgCPM:51.24, eCPC:37.11). I actually used this model and obtained the metrics results on the unseen test data(clicks:178, ctr:0.149, cost:6099.896, eCPC:34.27). I have included the corresponding Figure 8 and 9 in the report and "linear_bid_xgboost.csv" on the GitHub.

Furthermore, I set the bid prices of the advertisers 2259 and 2261 to a very low price such as 5. This consequently improved the clicks by a very small amount.

71	164	0.14149519	5869.953	50.6445192	35.7923963
72	165	0.14089197	5954.108	50.8415776	36.085503
73	165	0.13948416	6036.805	51.0326477	36.586697
74	165	0.13807878	6123.062	51.2402989	37.1094667
75	165	0.13671956	6207.95	51.4392841	37.6239394
76	163	0.13468737	6250.053	51.6443675	38.3438834

Figure 8: Part of linear bidding experiment csv results using xgboost

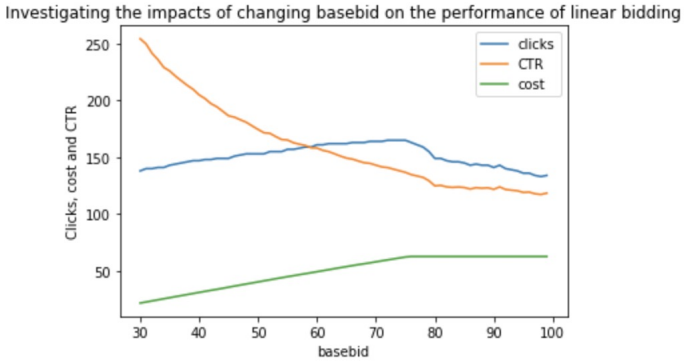

Figure 9: Linear bidding experiment plot results using xgboost

67	161	0.12756315	6068.681	48.083233	37.6936708
68	162	0.12687871	6172.881	48.3461204	38.1042037
69	163	0.12665309	6250.026	48.5635053	38.3437178
70	161	0.12567619	6250.047	48.7877087	38.8201677
71	156	0.12230306	6250.009	48.9996942	40.0641603

Figure 10: Part of linear bidding experiment csv results using average predictions of xgboost and logistic regression

3.3 Predict pCTR using averaged predictions

I tried to predict pCTR using weighted averaged predictions of logistic regression and xgboost models. The optimal clicks number on the validation set is actually slightly worse than that of xgboost linear bidding model.(clicks:163, ctr:0.127, cost:6250.026, eCPC:38.34) Similar to before, I have included the corresponding Figure 10 in the report and "linear_bid_xgboost_lr.csv" on the GitHub.

3.4 Bidding using ORTB strategy

The bidding price can be essentially calculated using the equation 1. The pCTR is predicted using xgboost instead of logistic regression since xgboost is a better model to predict pCTR. This strategy contains two parameters so it is very time-consuming to find the optimal parameter for this bidding function[1]. Due to limited time, I can only spend limited time tuning parameters. After tuning parameters, I found the optimal lambda and c to be $7 * 10^{-5}$ and 980 respectively. The performance on the validation set is clicks:166, ctr:0.130, cost:6181.05, eCPC:37.24. I obtained the metrics results on the unseen test data(clicks:178, ctr:0.139, cost:6191.14, eCPC:34.78). Similar to before, I have included the corresponding Figure 11 in the report and "ortb_xgboost.csv" on the GitHub.

7.00E-05	960	165	0.12938335	6174.471	48.4165909	37.4210364
7.00E-05	970	165	0.12934278	6178.505	48.4330318	37.4454848
7.00E-05	980	166	0.13009812	6181.054	48.4423806	37.2352651
7.00E-05	990	166	0.1300655	6183.832	48.4520011	37.252
7.00E-05	1000	166	0.13004003	6185.94	48.4590256	37.2646988
7.00E-05	1010	166	0.1299993	6189.67	48.4730565	37.2871687

Figure 11: Part of ORTB experiment csv results using xgboost

$$b_{ORTB1}(\theta) = \sqrt[2]{\frac{c}{\lambda}\theta + c^2} - c \quad (1)$$

where θ is the pCTR and λ and c are the parameters to tune.

4 CONCLUSIONS

According to the performance results, the best model is xgboost linear bidding model since it results in the highest clicks and ctr on the unseen test data. However, I still believe the ORTB strategy can potentially exceed the best performance I obtained. This is because if we fully tune the parameters, ORTB may result in a higher click than 178.

REFERENCES

[1] Jun Wang Weinan Zhang, Shuai Yuan. [n. d.]. Optimal Real-Time Bidding for Display Advertising. *Commun. ACM* ([n. d.]). http://wnzhang.net/papers/ortb-kdd.pdf

[2] Jun Wang Weinan Zhang, Shuai Yuan and Xuehua Shen. 2014. Real-time bidding benchmarking with ipinyou dataset. *arXiv preprint arXiv* (2014), 1407–7073. https://arxiv.org/pdf/1407.7073.pdf