Revenue CatVisual

Group members: Qifu Yin, Yue Zheng, Fengfan Yang, Xuchen Qiu

Nowadays, it is popular to buy things on GStore (Google Store) because it offers a variety of products and discounts. This trend motivates us to investigate how GStore's customers behave and predict its transaction revenue based on customer behaviors. In general, many features may affect the revenue of a GStore, such as the number of visits and pageviews, and traffic sources. Thus we are going to analyze how each factor relates to the revenue and how to predict the revenue based on these factors.

In this problem, we use the dataset provided by Kaggle. It includes customer behavior information such as transaction information, geographic information (where the customers are from), device information (which device customers use to access GStore), etc.

Given the dataset, we approach the problem through the following steps:
First, we did data cleaning and preprocessing for further analysis.
Second, we did data visualization to extract business insights from data.
Third, we used machine learning tools to make the prediction based on historical customer data.
We write our visualization and machine learning parts into two different notebooks since our methods of data cleaning are different.

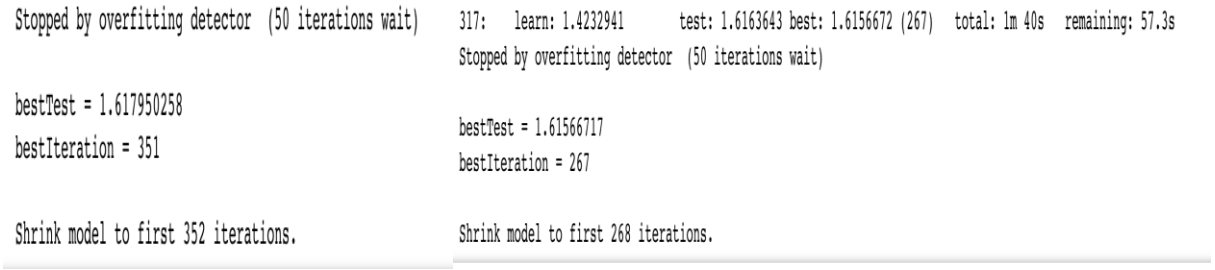
The data visualization part gives us a chance to explore the features in more details. The Geo Network graph vividly presents the footprints of our customers, i.e., which our customers are from. The graph is imaginative in the way that we can scroll our mouse to see different parts of the world map and the corresponding customers buying behavior.

The machine learning part is the highlight of our project. There are two parts in the catboost machine learning process. First, data prepossessing: 1. We fill NaN values with 0s, or empty strings depend on the datatypes of the columns. 2. We calculate and add some other useful indicators, such as average hits per month, which the original dataset doesn't contain.
3. We drop useless columns. We drop the columns with only one constant value and columns with all unique IDs which are only used for numbering and contain no consumers behavior information.

Next, we introduce our CatBoost library to do our regression training. Catboost is a newly invented library published in 2017. The CatBoost library includes various advantages. First, it is robust to reduce the need for extensive hyper-parameter tuning. Second, it is easy-to-use by offering Python interfaces integrated with scikit, R and command-line interfaces. Third, it can directly deal with categorical features (string, int, float). One characteristic of our dataset is that

the majority of columns are categorical. Thus with this advantage, we can skip the feature encoding step while other machine learning models usually require.

We both tried feature encoding by ourselves and let the model do the encoding work. The results are the almost the same in terms of RMSE on our test dataset, but the time complexity is much better (less than two minutes) if we take feature encoding manually in data preprocessing stage; otherwise the model will take 30 minutes if it directly deals with original dataset but can save our efforts in preparing the data.



The left-hand picture is the model without preprocessed feature encoding; while the right-hand is the picture with feature encoding. (bestTest is measured by RMSE). Since the model runs faster with preprocessed feature encoding, we choose to manually do the feature encoding.(However, we also reserve the code which enables the model to directly deal with categorical features in our notebook).

Lastly, it is much faster to use CatBoost comparing to other gradient boosting on decision trees algorithms, like Light GBM and Xboost. It only takes us less than 2 minutes to run. Moreover, there are more than 30 features in this dataset. Thus we use the function within CatBoost library to plot the features in a descending order in terms of their importance. The result shows that the most important features are total pageview times, the total hits, and visit numbers of users.

The predicted results are like the following picture: we make the prediction of log(user revenue) through our catboost regression model, and match the predicted revenue to the unique user ID. We finally export the final dataframe to a CSV file called 'test_submit.csv'.

	fullVisitorId	Predicted_log_Revenue
0	6167871330617112363	0.011854
1	0643697640977915618	0.019300
2	6059383810968229466	0.006478
3	2376720078563423631	0.000000
4	2314544520795440038	0.007392