Google Smart Bidding Challenge

Action Learning Project
BUMK776
Group 8

Business Challenge

The Google Merchandise Store is an e-commerce site that sells Google Branded merchandise. The store team uses digital ads to drive consumers to their site and increase sales. The team is primarily interested in acquiring new customers (i.e. first time purchasers) through digital advertising. More specifically, they want to acquire customers that will be the most valuable in the long-term. "Long-term" value will be considered as the customer's value during the first 90 days starting when they make their first purchase. Our task was to build a model to predict the long-term value of first time purchasers. These values will be used as conversion values to inform Google advertising platforms' bidding algorithms and help the Marketing Science teams at Google to improve the algorithm.

Data Exploration and Preparation

Since our business challenge was to build a prediction model to evaluate the customer lifetime value over a period of 90 days, we had to analyze our first purchase sessions starting May 2017. We chose this period as our starting point because the dataset provided to us was limited to September 2017 and we had to ensure that we had enough data to build our prediction model for 90 days.

To analyze this data, we broke down the dataset into 4 segments: Studying the user's first purchase session, the impact of promotional events leading to their first purchase, 7 days data after the first purchase was made to study their activity and engagement with the e-commerce website and studying sessions within 8 to 90 days post-first-purchase to look into future purchases.

Choice of Variables

The initial model shared by Google, covered base with mostly transactional variables. We then recognized variables that could be added to the model to enhance its prediction and efficiency. While some of the variables selected were based on the correlation of these variables with our dependent variable - 'Future Revenue', it was hard to build a model based solely on correlation values since the dataset provided to us had at least 90% missing data(including zeros). This led us to pick and choose variables based on our marketing acumen, and look into factors above and beyond transactions to gain insights about these consumer's engagement patterns. Looking into 'Channel Grouping', we could identify the key channels that led to maximum traffic on the page. To study geonetworks from where the sessions originated, we identified top metros and states in the US(derived from IP addresses) and built dummy variables for these regions to get a better understanding of how relevant these would be to our model. This set included regions such as - BayArea CA, Los Angeles CA, NY, Seattle-Tacoma WA, Chicago IL.

E-commerce action type discusses the ecommerce hits that occurred during the session and categorizes data based on user's activities such as clicking on product details, adding items to cart, removing items from cart, checking out, entering shipping and billing details, reviewing cart, making the final payment to complete the

transaction. Correlation between futureRevenue and Added item to cart was -0.021854; for billing and shipping, it was 0.039882; for payment, it was 0.052794, and for review, it came out to be -0.022763.

To capture the time of the day when the purchase occurred, we segmented the hourly hits data based on midnight(Hour 0 to 6), morning(Hour 7 to 12), noon(Hour 13 to 15), evening(Hour 16 to 19) and night(Hour 20 to 23). The correlation here for hits hours vs Future Revenue ranged from 0.021444 to -0.021627. We identified the first pageview/screen view as 'Hits Entrance' and the last pageview/screen view as 'Hits Exit' so that the model understands where the user's journey starts and ends in a session.

Similar to other e-commerce websites, the Google Merchandise Store also includes 'Product List Names' based on the user's past activities, search histories, etc. Major subcategories included in this section were - 'Category List', 'Related Products List' and 'Search Results List'. The correlation for each of the three vs Future Revenue were -0.056815 , 0.080533 , -0.037181 respectively.

Additionally, we also added a dummy variable to identify whether users came from a social channel or not. This was complemented with our next set of variables identifying the different social channels the user could have possibly come from - 'Youtube' and 'Facebook' since these channels accounted for the highest number of hits on the page.

The last set of variables added were categorized based on Google owned brands - 'YouTube', 'Android', 'Google' and were added as content groups.

Modeling Approach

Following the completion of our data preparation and exploration, our model was ready to be used for implementing various modeling techniques to aid us in determining patterns in our data. Considering the complexity of the dataset and to effectively evaluate our model's performance (accuracy and efficiency), we divided our dataset into a training set (a subset to train a model) and test set (a subset to test the trained model). Additionally, we employed hyperparameter tuning to control the behavior of our XGBoost Regressor model and to maximize the model's predictive accuracy. After experimenting with various combinations when training our data, we found that setting the maximum depth to 8 resulted in more accurate findings. To normalize the model, we removed missing values from variables like 'firstWeekTimeOnSite' and replaced median values for variables like 'checkout' and 'AffiliatesChannelGrouping' and lastly, we removed the outliers from few of the variables.

Our development dataset was then divided into a regression and classification model, with the regression model containing all sessions that resulted in a purchase and the classification model predicting the likelihood of a purchase from sessions with no first purchase transaction. We chose to split the data to make the model less complicated and increase accuracy. Our team used the below Modeling techniques*(Appendix: 1)* towards building a model to predict the long-term value of first time purchasers.

As our dependent variable in the regression models (future revenue) is a continuous variable, we started with **Linear regression** being one of the most classic regression algorithms. The model aided us in comprehending the linear relationship between the independent and dependent variables (future revenue). It

served as a good starting point and benchmark for other modeling techniques that we were planning on testing. However, the linear regression model has its limitations especially considering the enourmous dataset and number of variables. Additionally, linear regression is sensitive to outliers which can impact the model. Considering all the limitations and after carefully analyzing the results, the linear regression model was not reliable solely.

After Linear Regression, we employed a **Decision Tree Regressor** modeling technique. Decision tree helps break down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. We used this technique due to the pros of this technique - (1) It works for both continuous as well as categorical output variables (2) Non-linearity has no effect on the model's performance, and the amount of hyper-parameters that need to be tuned is almost null. We employed the decision tree regressor to assess the predictability of our continuous variable - future revenue. However, as compared to other decision predictors, this model is unstable, and minor changes in the data cause significant changes in the decision tree structure.

Further, looking at the limitations of Decision Tree Regressor we employed **Random Forest Regressor**. We employed this technique as it combines multiple decision trees in determining the final output rather than relying on individual decision trees. Higher accuracy was achieved with this model since it allowed for missing values. Random Forest Regressor is a better performance model as it avoids overfitting. However, we cannot rely on the standalone interpretability of this model since it calculates an average over multiple decision trees.

We further employed **XGBoost Regressor** technique and achieved more accurate results. XGBoost regressor uses more accurate approximations to find the best tree model.

We used classification models for the dependent variable repeat purchaser, which is a categorical variable. We began with employing **Logistic Regression** to model probabilities with two possible outcomes while assigning class levels to input data (categorical variables). However, it fails since the model assumes linearity between the dependent variable and the independent variables. Further we used **Decision Tree Classifier** and **XGBoost Classifier** models. However, the classifier in these models was unable to distinguish between Positive and Negative class points. In other words, the classifier predicts either a random class or a constant class for all data points. Finally, we used the **Gaussian Naive Bayes** approach, which makes naive assumptions about the independence of the characteristics. This modeling technique is associated with each class and follows a normal (or Gaussian) distribution. We did not acquire reliable findings with this model, which can be attributed to the model presuming conditional independence, which does not hold true because the feature set displays interdependence to a great extent.

Evaluation and Results

After multiple iterations and tuning in our models, we reached optimal performance values for our models *(Appendix 1)*. Amongst the multiple initial models, XGB regressor model with parameter as max_depth=8, n_estimators=150, reg_alpha=10, reg_lambda=0, objective='reg:squared error', booster='gbtree',

random_state=123, learning_rate=0.20 performed best. However, the R-squared value was very low and it was also clear that most predictions are very close to 0.

Further, after splitting the modeling into regression and classification, we performed 4 models for regression modeling for repeat purchasers. We made predictions using the XGB Regressor model, decision tree regression model, Linear regression model and random forest model. XGBoost regressor model with hyperparameter tuning contributed to the best MAE overall. The best values for parameters are max_depth=5, n_estimators=100, reg_alpha=0, reg_lambda=10, objective='reg:squarederror', booster='gbtree', random_state=123, learning_rate=0.02.

By using Gaussian Naive Bayes model for classification modeling we were able to get ROC AUC value of 0.56. However, we got ROC AUC of 0.507 from XGB Booster Classifier (max_depth=6, n_estimators=150, reg_alpha=0.5, reg_lambda=0, objective="binary:logistic", booster='gbtree', 'random_state=123, learning_rate=0.2) , 0.52036 from Decision Tree Classifier Model and 0.498 from Logistic Regression model. All classification models didn't give strong values to consider in our final calculations *(Appendix 2)*. The values were equivalent to random guess and therefore we calculated final the prediction as product of [XGB Regression Model Prediction] * [Average Repeat Purchase Rate] .

Our best MAE value is 18.838 with RMSE of 69.43 and avg. error of -3.83 using XGB Regressor Model for repeat purchasers *(Appendix 3)*. We also included direction and scope for the Random forest model as it was the second best fitting model with MAE of 22.77 and best avg. error of 1.07. We rejected the initial model and Naive model because of extremely high errors and MAE values.

Learning & Improvements

This project taught us many important lessons about predicting lifetime value and conducting a research study in general. It gave us hands-on Big Query ML experience which we used to further enhance our model. By putting our model to work within the Google smart bidding business challenge, we were able to understand major drivers to improve the system for Google and its clients.

With our improved CLTV prediction, we can now provide the system with data inputs that reflect the eventual impact of each customer and therefore, achieve precise targeting for a potential high value shopper. With this insight, budgets can be allocated to the channels that maximize profit over a longer time period which, in our case, is Direct Channel. Google Ads platform can be a great aid in improving the channel further.

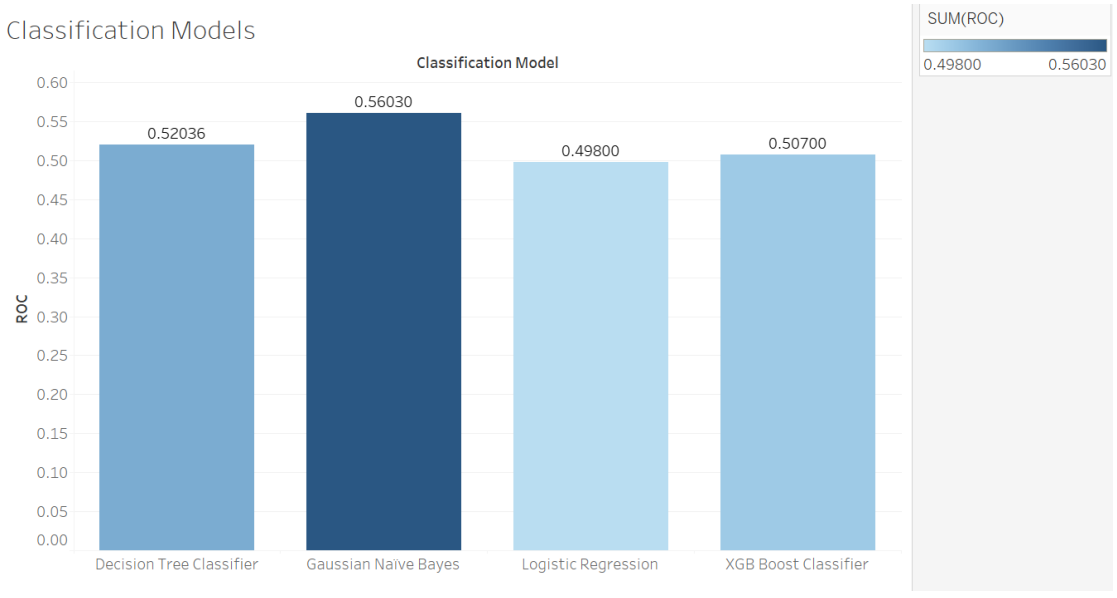
Overall CLV based model provides more valuable insights compared to return on investment model for Google Ads which may not be able to maximize the potential of the ads. With all the insights, techniques and systems put together our model is flexible to perform and adapt to suit multiple client needs.

APPENDIX

1. Model estimations

Model Type	Model Name	Training RMSE	Testing RMSE	Training R-squared	Testing R-Squared
Initial Model	Linear Regression	168.21	143.63	0.11	0.08
	Decision Tree	140.1	212.08	0.2	0.12
	XGB Boost	145.85	219	0.13	0.07
Regression model for repeat purchaser	XGBoost Regressor	340.1	578.93	0.61	0.17
	Random Forest Regression	192.05	362.36	0.9	0.22
	Linear Regression	486.8	357.6	0.34	0.19
	Decision Tree	419.8	303.18	0.2	0.12

2. Classification Model Estimations



3. MAE and Average Error