Online Retailer : Summary Report

Group 5

Our chosen dataset is a transactional data set from an online e-commerce system obtained from UCI's machine learning repository. Our objectives were:

(i) Customer Segmentation and Prediction using Recency, Frequency & Monetary (RFM) Value

(ii) Predict Customer Lifetime Value (LTV) to calculate the total revenue a business can reasonably expect from a single customer account.

(iii) Perform product clustering to better understand customer spending patterns to create an personalized retail experience

Data cleaning was more challenging that anticipated as some identifiers or behaviors were difficult to interpret as we have limited knowledge of the POS system, eg. stock codes, or behaviors of cancellations, refunds, discounts, etc. However we incorporated intricate logic to identify such records and limit our cleaned data to only transactions with a (net) positive revenue.

For our first objective of customer segmentation, we used a semi-supervised learning approach. We first used unsupervised algorithms - K-means, hierarchical clustering and DBSCAN to segment customers to clearly identify the customer's purchase profile. All three algorithms revealed an optimal k value of 3, with k-means outperforming others in forming distinct customer segments. Once clustering was performed, we used various classifiers including - Gradient Boost, Random Forest, Decision Tree, AdaBoost, Extra Trees, K Neighbours to predict their segments based on the RFM scores. Here, we found the Extra Trees Classifier performed the best. Other models were either overfitted or under fitted.

Predict LTV: we invest in customers (like discounts and promotions) to generate revenue.These actions make some customers more valuable in terms of lifetime value but there are some customers who pull down this profitability. So we need to segment customers and act accordingly. To implement this we split the dataset into 2 parts, we took 3 months of data, calculated RFM Scores. RFM Scores that we calculated are our predictors.Then we calculated 6 months LTV for each customer, which we used to train our model.

Lifetime Value **("Golden Metric")** *: Total Gross Revenue - Total Cost*

Since there is no cost, Revenue becomes our lifetime value. Finally we merged 3 months and 6 months dataframe to see correlation between LTV and our predictors.we found positive correlation between them, so our alternate hypothesis was correct means High RFM scores leads to high LTV. Model can predict LTV (money form) but we want segments so that business can target customers based on their predicted LTV. So we applied clustering and have three segments low, medium and high. The Extra Tree model was chosen as the best model out of many models for making predictions, the model was evaluated on many metrics

	Prediction_Time	Accuracy_Score	f1	Precision	Recall
ExtraTreesClassifier	0.021827	79.920000	0.792593	0.809061	0.822222

For our last objective, we analyzed consumer spending patterns to generate recommendations and a purchase prediction model. For generating recommendations, we used the product description values as input for the TF-IDF Vectorizer and used K-means to form clusters. The optimal k was 16, however visualizing the results in a 2-D graph using t-SNE showed formation of some distinct clusters, with one cluster being sparsely distributed. The output of K-means assigned each product a cluster. For each cluster the product that was the most popular among other customers and the product that generated maximum revenue was found. Based on a customer's top spending cluster, the above generated product recommendations were made.

We also applied a combination of product clustering and customer segmentation to develop a matrix of customers and their total purchases, average purchase amount, total amount spent, and total amounts spent in the respective product category clusters. Customer purchase prediction models were created using these clusters, however, the VotingClassifier accuracy results against the actual data was approximately ~77%. Individually the classifiers scored much higher with the Training/Test data set with Random Forest algorithm scoring the highest at 91.4%.

To improve the model, we would request a larger dataset size, as the current dataset only contained 13 months of transactional data. Insights into the descriptors included in the dataset (eg. stock codes, discounts, refunds, or cancellations) and how they are handled within the e-commerce system would be helpful in creating a cleaner data file. Finally, a part of the analysis that would be redundant in the real-world is product clustering. As the dataset is of an e-commerce system, the underlying product catalogue would have already been built from product information provided by suppliers. Hence, had this been readily available more efforts could've been focused on the recommendation and prediction models.