DSC 630 – Predictive Analytics

Course Project

Kimberly Cable

(Individual)

Fall, 2022

Data Selection and Project Proposal

Overview

Olist is a Brazilian e-commerce company that connects small businesses to a larger marketplace. It gives these small businesses a way to manage their products, shipping, and online payments. They have approximately 200,000 users in about 180 countries.

Business Problem

With any online retailer, retaining customers is key but knowing who may leave and who stays could be guesswork. Understanding how and why customers stay and shop and why they leave is pivotal to a company's business.

Knowing who your customers are and how and why they shop or do not return plays a big part in customer service which ultimately enhances a customer's satisfaction level. A high satisfaction level could lead to overall better reviews and with these good reviews, their sellers will see more new sales.

This study will hope to find similar traits among its customers. This will help the marketing team know who best to send offers to or to whom might they need to send a discount coupon because it looks like they haven't purchased in a while.

Data

The data has been sourced from Kaggle and has 100,000 online orders from 2016 to 2018. The data consists of records with products, customers, and review information for each transaction provided by Olist.

The data consists of the following datasets (see Figure 1):

- Customers: This dataset has information about the customer and their location.
- Geolocation: This dataset has information about the Brazilian zip codes and their latitude/longitude coordinates.
- Order Items: This dataset has information about the items purchased within each order.
- Order Payments: This dataset has information about the order payment options.
- Order Reviews: This dataset has information about the reviews made by the customers.
- Orders: This dataset has information about all customer orders.
- Products: This dataset has information about all the products sold by Olist.
- Sellers: This dataset has information about the sellers that fulfilled the orders made at Olist.
- Category Name Translation: This dataset has English/Portuguese translations for all products sold at Olist.

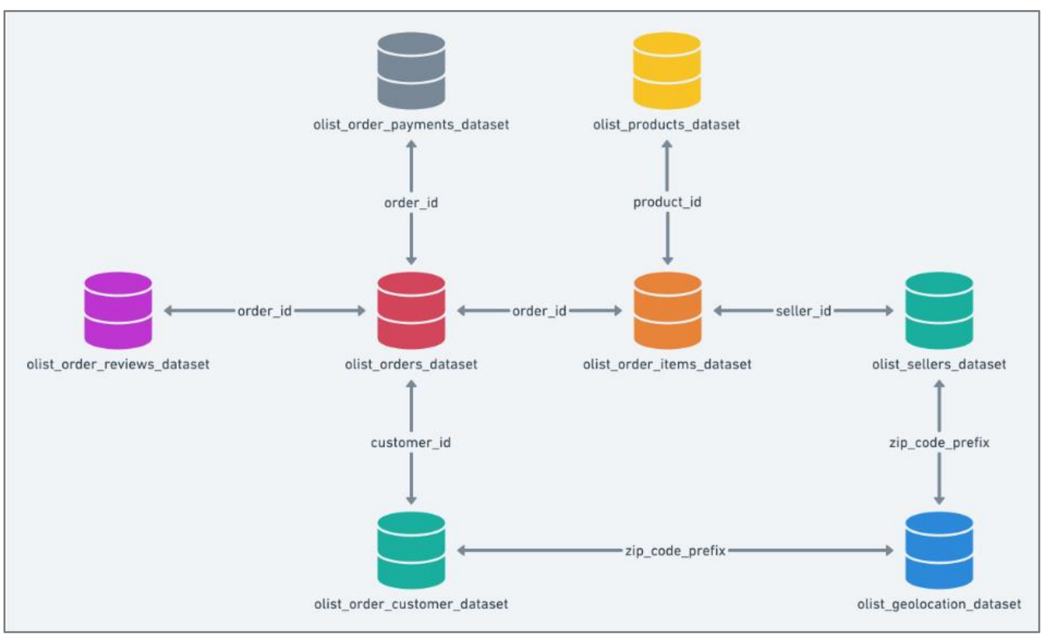

Figure 1: Data Schema

Model Selection

I plan to use the k-means clustering method. K-means is used to help identify clusters or groups within your dataset. This model will help group customers based on previous purchases and reviews that they made. I will also investigate RFM (recency, frequency, monetary) analysis. It is a marketing technique used to rank and group customers based on the number of times the customer has purchased, the last time the customer has made a purchase, and the total dollar amount the customer has spent.

Model Evaluation

To make sure my model is performing correctly, I will use the elbow method and/or the silhouette coefficient to make sure I am using the correct number of clusters. I hope to use a decision tree using the clusters found through the k-means clustering analysis. This will help analyze the results of the clusters.

Learning Objective

I hope to learn more about the customers who shop with Olist. This will help sellers target different types of individuals by grouping them into different categories to know whom they need to target with promotions and whom they need to go after to bring them back.

Risks

As with any dataset, there may be inaccuracies with the data, data may be missing or invalid. Also, the model I have chosen may not be the correct choice and another may be more suited for this data. Further analysis may need to be done.

Contingency Plan

If k-means clustering is not showing good accuracies for clustering, I may investigate another method for segmentation. Re-examining the data may also help. Removing outliers, narrowing the feature selection, etc. may help with model building and evaluation.

Preliminary Analysis

Will the data be able to answer the questions?

The data is comprised of 9 datasets from a dump of the company's database tables.

Customers:

- customer_id: key to the orders dataset. Each order has a unique customer_id.
- customer_unique_id: unique identifier of a customer.
- customer_zip_code_prefix: first five digits of customer zip code
- customer_city: customer city name
- customer_state: customer state

Since the customer_id is unique to each customer, I dropped the customer_unique_id. I also dropped customer_zip_code_prefix and customer_city as they will not be needed and just left customer_state to do analysis on where most customers reside in each state in Brazil.

Final dataset: 99,441 rows and 2 columns

Orders:

- order_id: unique identifier of the order
- customer_id: key to the customer dataset. Each order has a unique customer_id.
- order_status: reference to the order status (delivered, shipped, etc).

- order_purchase_timestamp: shows the purchase timestamp.
- order_approved_at: shows the payment approval timestamp.
- order_delivered_carrier_date: shows the order posting timestamp. When it was handled to the logistic partner.
- order_delivered_customer_date: Shows the actual order delivery date to the customer.
- order_estimated_delivery_date: shows the estimated delivery date that was informed to customer at the purchase moment.

I converted all date columns to a datetime type.

Looking at the graph of orders over time, I noticed little to no orders in 2016 and little to no orders after August 2018 so I dropped those orders (see Figure 2).

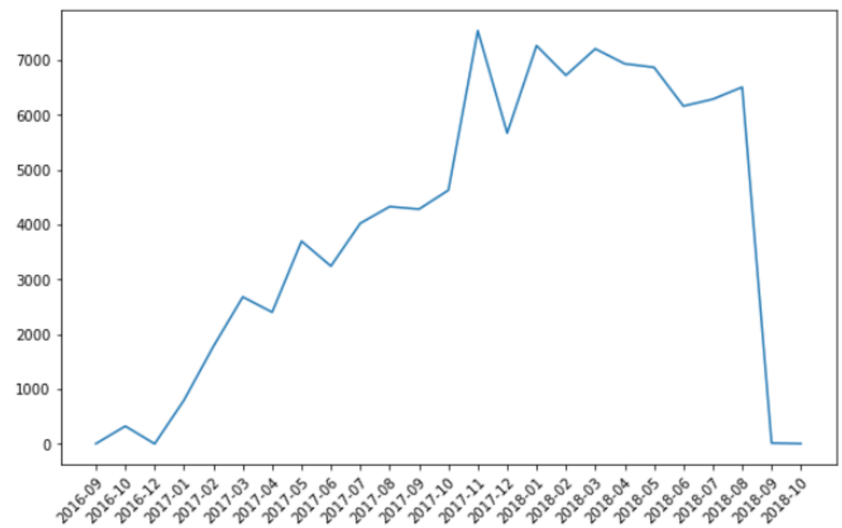

Figure 2: Number of Orders over Time

There were 82 empty values in the order_approved_at column. I looked at the order_status of each of the 82 values. 63 were 'canceled', 14 were 'delivered', and 5 were 'created'. The 'delivered' status should not have any empty order_approved_at values so I replaced the null values with the order_purchase_timestamp.

There were 1,602 empty values in the order_delivered_carrier_date column. The order status for 'delivered' was only 2 and the rest had status' that were okay for this field. I replaced the 2 empty values with the order_approved_at date.

There were 2,727 empty values in the order_delivered_customer_date column. The order status of 'delivered' was only 8 and the rest had status' that were okay for that field. To replace this field, I averaged the difference between the order_delivered_customer_date and the order_delivered_carrier_date to get the average difference between the two. I then added this date to the order_delivered_carrier_date to get the new order_delivered_customer_date for those empty values. In this case the difference was 9 days that were added to the order_delivered_carrier_date.

Final Dataset: 92, 580 rows and 8 columns

Order Items:

- order_id: unique identifier of the order
- order_item_id: sequential number identifying a number of items included in the same order.
- product_id: product unique identifier
- seller_id: seller unique identifier
- shipping_limit_date: Shows the seller shipping limit date for handling the order over to the logistic partner.
- price: item price
- freight_value: item freight value item (if an order has more than one item the freight value is split between items)

I dropped the seller_id and the shipping_limit_date as those two columns were not needed.

Final Dataset: 112,650 rows and 5 columns

Order Reviews:

- review_id: unique review identifier
- order_id: unique identifier of the order
- review_score: Note ranging from 1 to 5 given by the customer on a satisfaction survey.
- review_comment_title: Comment title from the review left by the customer, in Portuguese.
- review_comment_message: Comment message from the review left by the customer, in Portuguese.

- review_creation_date: Shows the date in which the satisfaction survey was sent to the customer.
- review_answer_timestamp: Shows satisfaction survey answer timestamp.

I dropped all columns except the review_id and the review_score. I may add this back with more research.

Final Dataset: 99, 224 rows and 2 columns

Products:

- product_id: unique product identifier
- product_category_name: root category of product, in Portuguese.
- product_name_lenght: number of characters extracted from the product name.
- product_description_lenght: number of characters extracted from the product description.
- product_photos: number of product published photos
- product_weight_g: product weight measured in grams.
- product_length_cm: product length measured in centimeters.
- product_height_cm: product height measured in centimeters.
- product_width_cm: product width measured in centimeters.

I merged the category name translation dataset with the products dataset. There were 2 category names that did not have translations. I googled the Portuguese names and replaced those names with their English translations.

I dropped all columns except the product_id and the product_category_name as the other columns were not needed.

Final Dataset: 32,951 rows and 2 columns

Sellers:

- seller_id: seller unique identifier
- seller_zip_code_prefix: first 5 digits of seller zip code
- seller_city: seller city name

- seller_state: seller state

Final Dataset: 3095 rows and 4 columns

Order Payments:

- order_id: unique identifier of the order.
- payment_sequential: a customer may pay an order with more than one payment method. If he does so, a sequence will be created to
- payment_type: method of payment chosen by the customer.
- payment_installments: number of installments chosen by the customer.
- payment_value: transaction value.

Final Dataset: 103,886 rows and 5 columns

Geolocation:

- geolocation_zip_code_prefix: first 5 digits of zip code
- geolocation_lat: latitude
- geolocation_lng: longitude
- geolocation_city: city name
- geolocation_state: state

I did not use this dataset as all customers were in Brazil.

Product Category Name Translation:

- Product_category_name: category name in Portuguese
- Product_category_name_english: category name in English

I did not use this dataset on its own and merged it into the products dataset.

Overview

Looking at the data initially, it looks like there will be enough columns and rows for me to choose from to perform a customer segmentation analysis.

Visualizations that will be useful

As I am looking to do modeling for customer segmentation, knowing more about how many items a customer purchased and how much they spent will be useful information. Also, how the different features may have played into how the customer scored their order the way they did.

I plan to graph the following:

- Number of orders per year, month, day of the week, and time of day
- Top and bottom categories that customers purchased from. Both in the number of items purchased and the amount spent
- The number of orders per State
- Review scores count per score
- Review scores per order status
- Review score per difference between purchase date and delivered date
- Review score per difference between the estimated delivery date and actual delivery date
- Top and bottom customers in spending
- Top and bottom customers in the number of orders purchased from olist

Do you need to adjust the data and/or driving questions?

After I do some exploratory analysis, I would like to look at RFM (recency, frequency, monetary) analysis to help me look at the buying behavior of the customers and then start building the customer segmentation model. Also, initially, I was thinking of doing either K-means or RFM but

I think I will do RFM first then K-means with the RFM data to help achieve my goals of finding out more about how to keep customers and receive high scores.

Do I need to adjust my model/evaluation choices?

Doing a little research in customer segmentation analysis, I think I will need to do a Recency, Frequency, and Monetary (RFM) analysis first then use this information for a K-means clustering model. This will help qualitatively rank and group the customers so the K-means clustering model can easily distinguish the different clusters. Also, I found that decision trees are good for cluster analysis in interpreting the results so I may try to incorporate this in my process.

Are my original expectations still reasonable?

The original expectations were to understand how and why customers stay and shop and why they leave. I believe with the data I have and the tools I have presented I can still achieve this.

References

Arvai, K. (n.d.). *K-Means Clustering in Python: A Practical Guide*. Retrieved from Real Python: https://realpython.com/k-means-clustering-python/

Begam, S. (2021). *Customer Profiling and Segmentation – An Analytical Approach To Business Strategy In Retail Banking*. Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2021/03/customer-profiling-and-segmentation-an-analytical-approach-to-business-strategy-in-retail-banking/

Kaggle. (n.d.). *Brazilian E-Commerce Public Dataset by Olist*. Retrieved from https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce

Pagan Research. (n.d.). Retrieved from olist: https://paganresearch.io/company/olist

Selvaraj, N. (2022). *How to Build Customer Segmentation Models in Python?* Retrieved from 365 Data Science: https://365datascience.com/tutorials/python-tutorials/build-customer-segmentation-models/

Wright, G. (n.d.). *RFM analysis (recency, frequency, monetary)*. Retrieved from TechTarget: https://www.techtarget.com/searchdatamanagement/definition/RFM-analysis