

**MMA 831**

**Marketing Analytics**

**Masoum Mosmer**

**Team Project**

**Aug 9, 2020 11:59 PM**

**Team Kipling**

**Ian Noriega | Melody Xu | Grace Guma**

**Rakesh Dhara | Shangeri Sivalingam**

**Shivanand Solomon | Tom Lee**

**Orderoffiles:**

|  |  |  |
| --- | --- | --- |
| **Filename** | **Pages** | **Comments and/or Instructions** |
|  |  |  |
|  |  |  |

**Additional Comments:**

Contents

[Executive Summary 3](#_Toc47888687)

[Introduction 5](#_Toc47888688)

[Olist and the Ecommerce Industry 5](#_Toc47888689)

[Olist’s Value Proposition 5](#_Toc47888690)

[Our Solution 6](#_Toc47888691)

[Data Description 7](#_Toc47888692)

[Olist E-R diagram and database 7](#_Toc47888693)

[Exploratory Data Analysis 8](#_Toc47888694)

[Olist Orders Analysis 8](#_Toc47888695)

[Data Preprocessing & Feature Engineering 9](#_Toc47888696)

[Customer Level Features 10](#_Toc47888697)

[Product Level Features 11](#_Toc47888698)

[Results 12](#_Toc47888699)

[Clustering 12](#_Toc47888700)

[Customer Level Clustering 12](#_Toc47888701)

[Product Level Clustering 14](#_Toc47888702)

[Recommender System 15](#_Toc47888703)

[Overall Approach 15](#_Toc47888704)

[Algorithm used - Singular Value Decomposition (SVD) & KNNbasic 16](#_Toc47888705)

[Modelling Results 17](#_Toc47888706)

[Recommender System Results 18](#_Toc47888707)

[Recommendations 18](#_Toc47888708)

[Create a Personalised Shopping Experience 18](#_Toc47888709)

[Run Targeted Promotions and Sales 19](#_Toc47888710)

[Improve Suppliers’ Inventory Management 19](#_Toc47888711)

[Produce Insights Report 20](#_Toc47888712)

[Appendix 21](#_Toc47888713)

[References 31](#_Toc47888714)

# Executive Summary

Olist is a Brazilian eCommerce startup that helps primarily small retailers reach a wide customer base by making their products available on a large-scale **online marketplace.** Founded in 2015, it now hosts 7,000 small retailers on its platform. It received $46MM in funding from SoftBank in 2019 and now aims to **reach 100,000 retailers on its platform by 2022**. For this to happen, the retail business itself needs to grow to entice new retailers.

The solution devised by Team Kipling is to **employ marketing analytics techniques** to increase sales and average transaction size on the platform. Specifically, we used two forms of **unsupervised machine learning** to personalize the customer experience and target customers more successfully. First, we **segmented customers using clustering** techniques to better understand purchase behaviors. Subsequently, we created a **recommender system** to personalise Olist’s product recommendations to boost customer spend and stickiness to the platform.

Below is a summary of the key findings and recommendations, with detailed explanations provided in the remainder of the report:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Recommendation** | **Explanation/Justification** | **Benefits & Marketing Metrics** | |
| **1** | **Personalize the shopping experience:**  **Recommender system** | Implement a recommender system to provide a personalized shopping experience for customers and boost platform sales. Push optimal products to the customer segments (i.e., clusters) who have displayed an affinity towards that product. | * Customer Satisfaction Ratings * Revenue Lift | |
| **2** | **Promotions & sales: Customer and product segmentation** | Run promotions & sales equipped with customer segment knowledge to up-sell and cross-sell to customers successfully. In particular, focus on customers who have only made one purchase with Olist (64% of all customers)- see details in Result section. | * Customer Satisfaction Ratings * Revenue Lift | |
| **ADDITIONAL RECOMMENDATIONS** | | | |
| **3** | **Inventory Management for Suppliers** | Use sales data to help suppliers manage their inventory more accurately. Identify top-selling items to prevent stockouts and increase customer satisfaction, as well as identify struggling/low-rated products to improve inventory efficiency for suppliers to save costs. | * Customer Satisfaction Ratings * Reduced Lost Sales Opportunities * Reduced Inventory Costs | |
| **4** | **Insight Reports** | Monetize the rich array of data by selling reports with industry and purchase trends to retailers. This will also draw suppliers to the platform. | * Revenue Lift * Diversify Revenue Sources | |

For Olist to achieve its stipulated goal of reaching a supplier base of 100,000, it will need to substantially increase its platform revenue via increased transaction numbers and increased average order size. The above recommendations will enable Olist to achieve these results.

The rest of the report will provide an overview of Olist, its industry and the challenge at hand; a description of the data available for addressing the challenge; the method used to perform the analytical solution; the results of the analysis; and lastly, the quantified benefits of deploying our recommendations.

# 

# Introduction

## Olist and the Ecommerce Industry

Over the past decade, the ecommerce business has exploded onto the retail scene. Ecommerce refers to the buying and selling of products and/or services over the internet. Several ecommerce companies, such as Amazon and Alibaba, have emerged and established themselves as giants in the industry. As seen in Appendix, Figure 1 (from Statista), and ecommerce’s share of the global retail market has continually increased over recent years, a trend that is expected to continue. This developing trend has pushed many ambitious startups around the world towards the industry over the past few years.

Olist is a Brazilian ecommerce startup that was founded in 2015. Olist primarily aims to serve smaller merchants, typically standard brick-and-mortar retailers, by making their products available on a larger-scale online marketplace. Olist also handles all logistics from the merchants to the purchaser, making them an attractive partner for burgeoning retailers who are eager to get their product recognized and out to consumers. Olist recently (near the end of 2019) received $46.65M in funding from a Japanese company called SoftBank (Mandl, 2019).

## Olist’s Value Proposition

Being an ecommerce platform, Olist’s primary objective is giving small retailers an avenue to effortlessly market, sell and distribute their product, as well as giving customers access to a wide array of products that they otherwise would have had limited visibility to, all on an easy-to-access online marketplace.

Looking forward, Olist’s primary focus is on expansion and scaling upwards – hoping to increase the total number of sellers it serves. They are also exploring the possibilities of offering a wider range of services including integration with ERP systems and offering working capital, in order to become a more complete, self-service platform for its clients (Menezes, 2019).

With these goals in mind, Olist should understand the importance of maximizing the likelihood that any one of its customer’s converts (i.e., purchases product) while browsing through the marketplace. To increase this likelihood, Olist should look to create a personal one-to-one relationship with each of its customers. Understanding that it primarily supports small, developing merchants who have not yet established a lengthy reach, Olist should place a focus on marketing the most applicable merchants and product directly to the most pertinent consumers. Not only will this drive conversion on Olist’s marketplace, but it will generate increased revenues for the merchants and add to a more personalized and efficient customer experience. Additionally, the in-depth customer insight that stands to be gained will provide Olist with yet another avenue to aid its growing merchants, while continuing to effectively scale its operations in the long term.

## Our Solution

Olist has managed to amass a deep collection of data that includes customer data points, product data, and order data (e.g., reviews), amongst others. Our goal is to leverage this data to gain a detailed understanding of the various customer segments that Olist helps its many merchants serve. Using this data, we aim to formalize customer and product segmentations using clustering methods. This exercise will provide insight into what the different types of customers that Olist and its merchants are serving, along with the unique characteristics and preferences that set these segments apart. Additionally, a recommender system will be developed that will allow Olist to suggest the best-suited products for customers based on the clusters, thus helping them establish a more personal marketing approach rather than the traditional “spray and pray” technique.

# Data Description

The data was obtained from Kaggle - Brazilian E-Commerce Public Dataset by Olist[[1]](#footnote-1).

## Olist E-R diagram and database

The Entity Relationship (E-R) diagram of Olist database has been used to identify the tables that will be used within the scope of this project as well as in identifying the inter-relationships among these entities. The diagram can be referenced in the Appendix, Figure 2. As such, to help solve the business goal at hand, attention was paid to analyzing the data on both a customer level and a product level. The following are the key tables used along with a brief description of the data contained therein:

|  |  |
| --- | --- |
| Key Tables used | Description |
| Olist\_orders | Contains details of each order such as its unique Order ID, who made the order (customer\_id), timestamps on when it was made, approved & delivered etc. This table is tied to the Olist\_order\_customers on “customer\_id”. |
| Olist\_orders\_reviews | Contains the reviews of each order (provided a review was done) with the review score ranging from 1-5 by the customer. |
| Olist\_order\_customers | Has information on the customer i.e. mainly their location. Used to identify unique customers (customer\_unique\_id). Each time a customer made an order, a new customer\_id was generated such that linking with the order table was possible. However customer\_unique\_id remained the same to identify repurchases by the same customer. |
| Olist\_order\_items | Contains details on the items (product\_id) purchased within each order such as the item’s price and who sells it (seller\_id) |
| Olist\_products | Contains details about the products sold and include the product\_id, category, # of photos and dimensions. |
| Olist\_order\_payments | Has details of how an order is paid. An order can be paid with a combination of different payment methods (for e.g. an order can be paid with both a credit card and vouchers). The number of installments and total payment value is also captured here. |

## Exploratory Data Analysis

Before an agglomerative dataset was assembled to analyze the two major aspects of the business case (i.e. the customer and product), the following Exploratory Data Analysis (EDA) was performed to understand the dataset such that the creation of features would be more organic. It should be mentioned that all of the id’s in the conglomerative data set were non readable and consisted of alphanumeric code.

### Olist Orders Analysis

Key insights gleaned from the data are as follows:

* **Payment method:** Olist’s customers paid off their orders in up to 24 installments, although the majority was paid off in just 1 installment. It was later learned that this sort of payment structure is common practice in Latin America that allows less affluent consumers to pay off their debts in installments throughout the year.[[2]](#footnote-2) The hypothesized reason why most customers on the Olist platform utilized just one payment is that their customers represent a certain tier in society and the products are affordable to them. There were also instances where different payment types were used to effect payments and we believe this reflects the type of customer making the purchase.
* **Timing:** It was also observed that most of the orders are made in the afternoon (Appendix, Figure 4). However, the only real discernable difference in the number of orders made across the days of the week are that Mondays and Tuesdays have the highest order counts while the order counts on the weekend are not that favorable (Appendix, Figure 5).
* **Geography:** It was observed that most orders originated from the state of Sao Paulo (SP). However, the state of Paraiba (PB) had the highest average spend per order while SP had the lowest. This could be attributed to the fact that SP houses 22% of the population while PB can only boast a meager 2%; or because SP is Brazil’s business center[[3]](#footnote-3) meaning orders can be processed quicker than in PB and customers do not feel compelled to buy multiple items in one order. Further validity can be attributed to the latter as the states of Acre (AC) and Alagoas (AL) only contain 0.4% and 1.6% of the population respectively and must therefore carry out their online purchases in ‘bulk’ rather that single shipments. Another reason could be that the people who do shop from these remote, less inhabited states are more affluent than the general personas in their respective states so they can afford to make big purchases.
* **Sales:** The top two selling product categories were found to be bed\_bath\_table (cama\_mesa\_banho) & health\_beauty (beleza\_saude) and are shown in the Appendix, Figure 7. They both have sales ranging above the 9.5K unit benchmark while the closest competitor had 8.5K. It would not be surprising if these two product categories are highly recommended by the recommender system.

These are all preliminary observations made during the Exploratory Data Analysis phase.

## Data Preprocessing & Feature Engineering

As a professional Kaggle dataset was used, not much data cleaning was needed. There were no missing values in any of the datasets, however, the Olist\_order\_reviews dataset had review comments that contained emoji and other special characters. This was handled by the code snippet as shown in the Appendix, Figure 10. Comprehensive details are in the python script - “Olist\_order\_reviews ETL.ipynb”. Each dataset was loaded into the MySQL database (Appendix, Figure 3) using python scripts (<table\_name> ETL.ipynb).

A combination of SQL and Python was used for feature engineering. Relevant tables were joined to create product level features (a sample snippet can be seen in Appendix, Figure 11). Details of the queries used to join the relevant tables can be found in the SQL script - “831\_Project\_Product\_Customer\_Features\_SQL.sql”. Due to the size of the datasets, feature engineering was performed in Python after joining the relevant tables in SQL (a code sample is found in Appendix, Figure 12). For example, to build the customer level features the customer dataset contained over 99,441 rows, so merging on the primary key “customer\_id” and grouping by “customer\_unique\_id” caused MySQL Workbench to crash. More details of the code to build the customer features can be found in the Python script -“831\_Project\_Customer\_Features.ipynb”. Details of the engineered features for both the Customer and Product Levels are shown below.

### Customer Level Features

|  |  |
| --- | --- |
| **Features** | **Description** |
| count (payment type credit card) | For each customer, the count of the number of times each payment type was used to make payments for orders was calculated. These features were used to help explain spending patterns of customers at the customer level clustering. There could be a relationship between the number of times a payment type was used to pay for orders and the purchase pattern of a customer. A possible theory, based on experience with online shopping is customers that mostly pay with credit cards would rather buy now and pay later, whereas, those that pay with mostly debit cards would be prudent spenders and spend within their means. |
| count (payment type debit card) |
| count (payment type boleto) |
| count (payment type voucher) |
| count (payment type not defined) |
| % of weekday orders (from datetime) | For each customer, the percentage of time the customer ordered during the day (defined by 6am – 6pm) and the percentage of time the customer ordered during the weekday were calculated. These features were used to help explain purchase patterns of customers at the customer level clustering. It will thus help in discerning if there is a relationship between the time-of-day and day of purchase. If there is a trend, this can be used to introduce promotions during certain periods of the day/week to boost sales. |
| % of day orders (from datetime) |
| total\_orders | Count of orders for each customer. This feature helped underscore the level of affluency of the customer. |
| Avg\_payment\_installments | Average payment installments a customer made.  This feature tends to reflect a customer’s payment ability. |
| Avg\_rating | Average rating a customer gave to past orders |
| Total\_delivered\_orders | Numbers of orders that have been delivered to the customer. |
| Value\_per\_installment | Average payment amount per installment. |
| Avg\_value\_over\_rating | Average of order value over customer rating. This feature means to indicate the relationship between price and customer satisfaction. |
| Avg\_order\_time | Average time between two orders for each customer. |
| Avg\_voucher\_per\_order | Average vouchers used per order for each customer. |

### Product Level Features

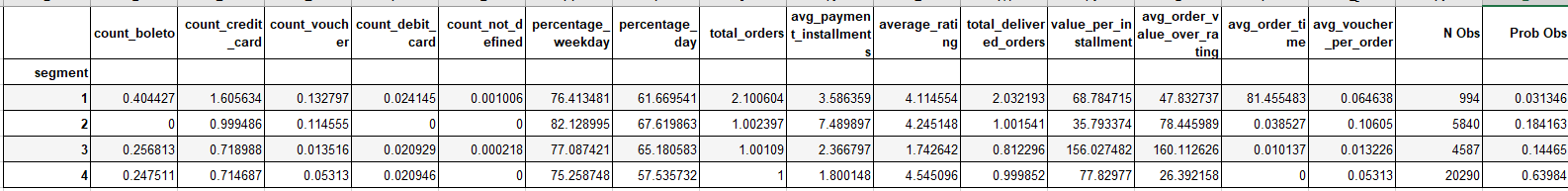
|  |  |
| --- | --- |
| **Features** | **Description** |
| % of weekday orders (from datetime) | For each product, the percentage of time the product was ordered during the day (defined by 06:00 – 18:00) and the percentage of time the product was ordered during the weekday was calculated. These features were used to help form product clusters based on the popularity of the product and when it was ordered It will thus help in discerning if there is a relationship between the time-of-day and day of purchase. If there is a trend, this can be used to introduce promotions during certain periods of the day/week to boost sales. |
| % of day orders (from datetime) |
| Product\_super\_category\_name | Of the 71 unique product categories that exist for the dataset, these were condensed into 5 major groups that aided in simplifying the clustering process. |
| Units\_sold | The total number of units sold for each product – indication of affinity. |
| Revenue | The total revenue generated per product. |
| Average\_price | The average price of the product on the platform across all sellers. |
| Average\_delivery\_time | The average time taken to deliver the product among all order instances. |
| Sellers | The number of sellers that stock the item. |
| Average\_review\_score | The average review score provided by the customers that did rate the product. |
| Photo\_effect | Developed to capture the propensity that the customer had towards purchasing based on photos present and units already sold. |

# Results

## Clustering

### Customer Level Clustering

We clustered customers based on their order and spending patterns. To do this effectively, we created features such as percentage of weekday orders, number of payment installments per order, number of vouchers used, etc. As we do not have in-depth domain knowledge for Olist and Ecommerce in general, we selected a Hierarchical Clustering algorithm. After analyzing the resulting dendrogram, we decided to cluster the customers into four segments because this produced the most desirable performance metrics such as silhouette scores. The table below shows the mean value of each customer level features for each segment, as well as number and percentage of observations.



The clustering presents distinctive customer behaviors as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Segment** | **Title** | **Key Points** | **Takeaways for Olist** |
| **1** | **Frequent Customers** | * Only segment that placed 2+ orders on average. * 3% of all customers. | * Only 3% of customers are frequent buyers. This presents a huge opportunity to further understand this cluster to try to move more customers into this category. |
| **2** | **Live in the Moment** | * Pay in 7.5 installments per order on average (highest amongst the segments). * Use the most vouchers. * At the same time, have a high average order value. * 18% of all customers. | * These are desirable customers who are willing to spend on more expensive products even if it means financing the purchase over a longer time horizon. These customers should be targeted with higher value products and financing solutions. |
| **3** | **Unsatisfied Customers** | * Give the lowest ratings (average 1.74/5). * Highest average order value. * Probable reasons for dissatisfaction:   Mismatch between quality and price.  20% of orders were not delivered.   * 15% of all customers. | * This is important feedback for Olist who manages the delivery of products. 15% of its customer base is overall unsatisfied and this is seems to be in large part due to orders not being delivered. Olist must act on this information to reduce the number of customers who fall into this cluster. |
| **4** | **The Average Joe/Jill** | * Only bought once with Olist. * Pay in less than 2 installments. * Lowest average order value. * Give the highest ratings. * 64% of all customers. | * This is the most important cluster for Olist to pay attention. The vast majority of its customers have only made one purchase with Olist and at the same time give the highest ratings. These are happy customers who need to be targeted with the appropriate marketing techniques to encourage a follow-up purchase. |

A few more important observations are that credit card is the most popular payment method, followed by boleto. Across all clusters, 75+% of customers place orders on weekdays and ~60% do so in the daytime. These are other important considerations for Olist to take into account when formulating its targeted marketing strategy.

### Product Level Clustering

Three clustering algorithms (Kmeans, Hierarchical, DB Scan) were run to identify product clusters. The hierarchical clustering output was selected for cluster analysis as it gave the highest silhouette score of 0.188. At this silhouette score, 5 product clusters were created.

Product clustering was performed by identifying features that explained how popular and how expensive a product was. The features included in clustering are to help understand how expensive and how popular a product is. Appendix figures 18, 19 contain the dendrogram and the mean values for each of the 5 product clusters respectively.

Common behaviours across clusters are – 77% of the products are sold on weekdays, 60% are sold during the day (6AM-6PM), most products are stocked by only 1 seller and get delivered in around 12 days. The customers are generally satisfied with product purchases as they have rated products quite high at around 4 out of 5. As seen above, while some features exhibit similar behaviours across clusters, other features differentiate clusters. The standout cluster among the 5 is the 3rd cluster I.e. Cluster #2. A quick overview of each cluster is shared below.

|  |  |  |
| --- | --- | --- |
| **Cluster** | **Cluster Name** | **Key Points** |
| 0 | **Lifestyle** | * Fashion and Tools/Repair. * High customer satisfaction rates. * 26% of all products on Olist. |
| 1 | **Technology** | * Mainly tech-based products (70%). * 3x sales of other products (avg 6.5 units per product vs 2). * Priciest products (avg 250 Reals vs 100-150 Reals). * 23% of all products on Olist. |
| 2 | **The Stars - Highest Selling** | * Most popular products with highest sales (avg 370 units per product vs 2-3) * Contains only 11 products, stocked by more stores than other products * <1% of all products on Olist. |
| 3 | **For the Homebodies** | * Household products * 40% of all products on Olist |
| 4 | **The Entertainers** | * Entertainment-based products * 12% of all products on Olist |

## 

## Recommender System

### Overall Approach

The recommender system was used on the data received from the customer level clustering. The recommender system is built on the premise that in order to recommend more personalized products to a customer, the recommender system will be run only on the data of the cluster that the customer belongs to. For example, as seen in Appendix, Figure 16, customer with “customer\_unique\_id” = 8c9ca7a5718a991b4f765b8c3d569b85, belongs to cluster 4 (based on customer level clustering), so the recommender system was run on all the instances of customers that belong to cluster 4.

### Algorithm used - Singular Value Decomposition (SVD) & KNNbasic

The recommender system was tested using the ‘Surprise’ python library that implements Singular Value Decomposition (SVD)[[4]](#footnote-4) and KNNbasic. In this context, both KNNbasic and SVD are collaborative based filtering algorithms that predict a rating for a customer and product pair based on the history of ratings given by the customer and ratings given to the product. However, SVD has additional complexity built into it as it uses matrix factorization in the prediction. SVD creates a matrix where each row is a customer and each column corresponds to a product[[5]](#footnote-5). The matrix is filled with ratings each user has ever given a product as shown in Appendix, Figure 17. Furthermore, it can be seen from Appendix, Figure 17 that there are scoring matrix is sparsely populated because in reality, a customer would not purchase every product nor would they rate every product they purchased. The goal is to fill in those missing values in the matrix.

First thing SVD does is to create latent factors. Latent factors in this context are typical customers and typical products. It is similar to PCA where the dimensions (the number of customers) are reduced to a few “typical customers”, for example, typical customers that buys health products (health product buyer). Each customer can then be represented by a combination of the latent factors. As shown in the sample equations (for customer1 and customer2) below:

The same can be done with each product. Each product can be represented by a combination of latent factors as shown in the sample equation (for product 1) below:

Once the latent factors are created[[6]](#footnote-6), 2 vectors are created. The first vector represents the affinity of each user to each of the latent factors as shown in the example equations below:

The second vector represents the affinity of each product to each of the latent factors as shown in the example equations below:

Finally, the rating can be calculate with the dot product of and. So when using SVD, the rating for each customer and product pair is given by the equation below:



Furthermore, if a customer has high affinity for the latent factors that are endorsed by a product, the rating for that customer and product pair will be high and vice versa.

### Modelling Results

The SVD and KNNbasic model was trained on 67% of data. Root Mean Squared Error (RMSE) was the metric used to compare the model. The final model was used to predict the top 5 product recommendations for a particular user and top 5 product recommendations for the whole cluster. As shown in the summary table below, SVD was preferred for the recommender system:

|  |  |
| --- | --- |
| SVD | KNNbasic |
| 0.5858 | 0.6194 |

Table 1: Summary of RMSE score for KNNbasic and SVD

### Recommender System Results

From Appendix, Figure 13 to Figure 16, the different products were recommended to customers based on the cluster they belong to. The recommender displays the results such that, products under the “Your top product recommendations” table are personalized product recommendations to that particular customer. Products under the “You may also like” table are products that are mostly recommended to all customers in that cluster. These cluster level recommendations can also serve as the first step of displaying personalized recommendations to a new customer who has visited the site and has not made enough purchases but who can be identified under a cluster based on their demographics and location information.

As expected from the EDA section, the product category “bed\_bath\_table”, as the most sold product is recommended to all clusters except surprisingly cluster 1. It could be because they represent only 3% of the Olist customer base.

# Recommendations

To achieve its stated goal of expanding its retailer base from 7,000 to 100,000 retailers in 2 years’ time, Olist should implement the following recommendations. To onboard retailers in the stipulated time, Olist needs to increase the overall number of transactions on its platform as well as the average transaction value to attract new retailers.

## Create a Personalised Shopping Experience

Olist should implement the recommender system built by Team Kiping to provide a personalized shopping experience for customers. This would recommend products based on previous purchases and on purchases made by similar customers. The primary objective is to increase the average transaction value for customers but also promote repeat purchase behavior among the existing customer base given the improved customer experience. New customers visiting Olist can also be treated a personalized experience, albeit less personal, by identifying certain information such as demographics and location to categorize them into a Customer Segment and provide segment-level product recommendations.

## Run Targeted Promotions and Sales

A vast array of insights were gleaned from the customer and product level clustering as outlined in the Results section. Customer level clustering should be used to understand which customers should be targeted and how. The greatest area of opportunity the 64% one-time yet satisfied purchasers who should be actively targeted with promotions to encourage a return purchase. The customers who often make large purchases should be targeted with higher value products. Product level clustering will enable Olist to identify and promote sales of top selling products by location. Furthermore, it will also identify struggling products which Olist can suggest suppliers bundle with other products to improve turnover. This will help suppliers with their inventory management.

From the recommender system, the top products recommended to each cluster can be obtained and used as part of a marketing campaign to other customers falling in that cluster (e.g. products in the bed\_bath\_table product category). Furthermore, according to the EDA, consumers tend to purchase products in the afternoon. Olist can recommend to sellers that sell the top products to host sales and promotions during the afternoon periods to boost sales.

We also provide two additional recommendations which were not part of the analytical solutions discussed in this report.

## Improve Suppliers’ Inventory Management

While this recommendation is not a direct outcome of clustering or of the recommender system, sales can be predicted for each product and that prediction can be used to estimate inventory needed to avoid stockouts. Olist can help retailers in avoiding potential stockouts by proactively informing them of the expected sales of products. Managing stockouts is a key aspect of maintaining revenue flows and increasing customer satisfaction – again leading to improved stickiness to Olist. Furthermore, it can help to indicate which products have lower turnover in order to bundle them with other products or to focus on restocking those products which are higher selling.

## Produce Insights Report

Another recommendation for attracting retailers to Olist is to create an internal “Sales Insights” team to identify industry trends. Retailers could then purchase these insights to improve their product mix. Olist would create an alternate revenue stream for itself by selling these insights to the retailers as well. This will ensure retailer stickiness to the Olist ecosystem.

# Appendix

* All significant plots, tables, etc.

A close up of a map

Description automatically generated

Figure 1: Ecommerce share of Global Retail Sales

(Note – projections are prior to COVID-19…share growth has accelerated since)

A screenshot of a cell phone

Description automatically generated

Figure 2: E-R diagram of Olist data

A screenshot of a social media post

Description automatically generated

Figure 3: Kipling’s 831 Olist Database

A screenshot of a cell phone

Description automatically generated

Figure 4: EDA - Olist orders analysis, time of day orders are made

A screenshot of a cell phone

Description automatically generated

Figure 5: Day of week orders are purchased in Brazil

A screenshot of a social media post

Description automatically generated

Figure 6: EDA – Olist orders analysis, distribution of installments customers tend to pay of their orders

A screenshot of a cell phone

Description automatically generated

Figure 7: EDA – Olist orders analysis, snippet of some of the product sold

A screenshot of a cell phone

Description automatically generated

Figure 8: EDA – Olist orders around Brazil, snippet of some states Olist orders are made

A screenshot of a cell phone

Description automatically generated

Figure 9: EDA – Olist orders around Brazil, average amount spent by customer per state

A screenshot of a social media post

Description automatically generated

Figure 10: Code snippet handling emoticons in review comments in Olist\_order\_reviews\_dataset

A screenshot of a cell phone

Description automatically generated

Figure 11: SQL snippet to create product level features

A screenshot of a social media post

Description automatically generated

Figure 12: Python snippet to complete the feature engineering after joining the relevant tables in SQL

A screenshot of a cell phone

Description automatically generated

Table 13: Recommender System – customer in cluster 1

A screenshot of a cell phone

Description automatically generated

Table 14: Recommender System – customer in cluster 2

A screenshot of a cell phone

Description automatically generated

Table 15: Recommender System – customer in cluster 3

A screenshot of a cell phone

Description automatically generated

Figure 16: Recommender System – customer in cluster 4

A screenshot of a cell phone

Description automatically generated

Figure 17: Example of matrix factorization created by SVD algorithm

A picture containing door

Description automatically generated

Figure 18: Product Clustering Dendrogram – NOTE that we took 5 clusters for analysis



Figure 19: Cluster Means

A close up of a sign

Description automatically generated

Figure 20: Recommender System - RMSE output for SVD()

A screenshot of a cell phone

Description automatically generated

Figure 21: Recommender System - RMSE output for KNNbasic()

# References

Mandl, Carolina. “Japan’s SoftBank invests in Brazilian marketplace integrator Olist.” Reuters, 23 Oct. 2019, uk.reuters.com/article/uk-softbank-latam-Olist/japans-softbank-invests-in-brazilian-marketplace-integrator-Olist-idUKKBN1X20TJ

Menezes, Fabiane Ziolla. “Why Do Brazilian Startup Olist Opted for SoftBank's Investment.” LABS, 24 Oct. 2019, labs.ebanx.com/en/articles/technology/interview-Olist-softbank/.

1. [https://www.kaggle.com/Olistbr/brazilian-ecommerce](https://www.kaggle.com/olistbr/brazilian-ecommerce) [↑](#footnote-ref-1)
2. <https://www.ecommerce-nation.com/digital-payment-solutions-latin-america-offer/> [↑](#footnote-ref-2)
3. <https://www.britannica.com/place/Sao-Paulo-Brazil> [↑](#footnote-ref-3)
4. <https://medium.com/@m_n_malaeb/singular-value-decomposition-svd-in-recommender-systems-for-non-math-statistics-programming-4a622de653e9> [↑](#footnote-ref-4)
5. <http://nicolas-hug.com/blog/matrix_facto_1> [↑](#footnote-ref-5)
6. <http://nicolas-hug.com/blog/matrix_facto_2> [↑](#footnote-ref-6)