Analysis of eCommerce User Behavior

Wenbei Zheng, Yuxin Yang

Abstract

Problem

As an online e-commerce platform or, any merchandiser in this digitalized era, being able to identify the high-value customers and the pattern of this group of people is key to a profitable business. And not only for the high-value groups, but a deep-dive user behavior analysis can also help with segmenting customers into different groups based on relative dimensions more accurately and for better-personalized ad contents.

While there are quite a few approaches for user clusterings, for instance, the RFM model, which will be covered later in this report, there seems a lack of utilization of such findings. For instance, how to understand the churn properly and how to reduce it. Companies usually apply a re-engagement strategy accordingly based on different types of churns, basically reaching out to customers by email with some recovery promotions or new customers incentives. But with deep dive into the churn group, the analysis can tell what's the activity churn, i.e. how many customers became inactive this month. In this way, a more efficient impact can be made in a timely manner. When it comes to eCommerce, a lower price at the first glance is attractive for new customer acquisition, but store owner needs to balance between promotion and branding. Limitless sales can only harm the business as well as the market as a whole.

From this online grocery store user behavior dataset, we wanted to find out:

1. If there is a specific pattern of users' purchasing behavior. The dataset contains information such as user_id, category_code, event_type, event_time, etc.

2. If there is a relation between the number of orders made by users and the event_time of the “purchase” event_type. What would be the most active time period of the day for this store?
3. The dataset is rich for helping us to classify the users into different groups. For instance, which type of users would be the core value customers of the store, and which group of customers would be the churns.
4. The brand and category fields can be applied for association rules. We can apply the apriori model to see what itemsets are within the top popular buys.

Methodology

Briefly, we applied four approaches to tackle the aforementioned issues. To define the users’ purchasing behavior, we applied the AARRR model to see all five stages for this online store. For instance, in the first letter “A” which stands for acquisition, we divided by date and to see the new customers acquired for each day. The event_type field of this dataset is helpful to calculate the conversion rate between each event type. From the time dimensions of the DateTime value, we divided the data by week, date, and hour, to find if there are any specific patterns on purchasing, viewing, and the click rate. To define segments for users, we applied the RFM model, which calculates recency, frequency, and monetary scores separately in the first place, and then sum up all three scores and obtain a total score. Customers can then be filtered by this final score into different groups. Lastly, to see if there’s any association among items, we applied the Apriori rule with tested minimum support value, minimum confidence value, and the minimum length required for each itemset. We assume that these approaches are reasonable to analyze this eCommerce grocery store.

Code

Steps:

Part I - RFM Model

1. As the column information shows that columns “category_code”, “brand”, and “user_session” contain missing values, which impacts the dataset as a whole for further analysis. We decided to first drop all NaN values for both datasets.
2. Union the two data frames as the raw data
3.
    a. Drop unnecessary columns
    b. create a new data frame
    c. query only the purchase event type
    d. Convert event_time to DateTime
4. Aggregate data by user_session
5. RFM analysis - recency, frequency, and monetary calculation
6. RFM data description
7. Apply RFM user segmentation
8. Define user segmentation
9. Visualization of RFM user segmentation

Part II - Apriori Model

1.
    Generate a new data frame for Apriori analysis only
    Split the category_code column by the period, ‘.’, and create a new column named items
    Create a new data frame for user_id and items
2. Find the frequency of the items bought
3. Visualize the frequency of the items bought
4.

Create a new data frame with event_time added

Group the data frame by user_id and event_time, and concatenate the items bought

Apply apriori model with:

i. Min_support = 0.005: so that the itemset occurs in 0.5% of the transactions.

ii. Min_confidence = 0.2: so that we know 20% of the times a customer also bought such item before and after.

iii. Min_length = 2: so the least number of items that a rule should have is 2.

5. Display all the RelationRecord of the association rule.

Part I - AARRR funnel

1. Drop the unnecessary data like “price” = 0 and drop the column which has null
2. Due to the limited memory, iterate through all the columns of a data frame and modify the datatype to reduce memory usage
3. Split the “event_time” to the “date” and “time”
4.

a.Set the people who made a purchase for the first time as a new user

b.creating the new data frame which contains “date” and “purchase _num”

c.From 10/2, the data of each day will be contacted together with the previous data

d. Subtract each day's data with the previous data with the same id to get the number of new users each day

e. Visualize the result

5. Extract the number of unique visitors for each behavior, calculate the transformation of each part and make the funnel chart.
6. put the user who never operates “cart” and “purchase” as a jump user and put them in a list. Calculate the bouncing rate by using the length of the list to divide all users.

Part II - Different distribution based on event_type

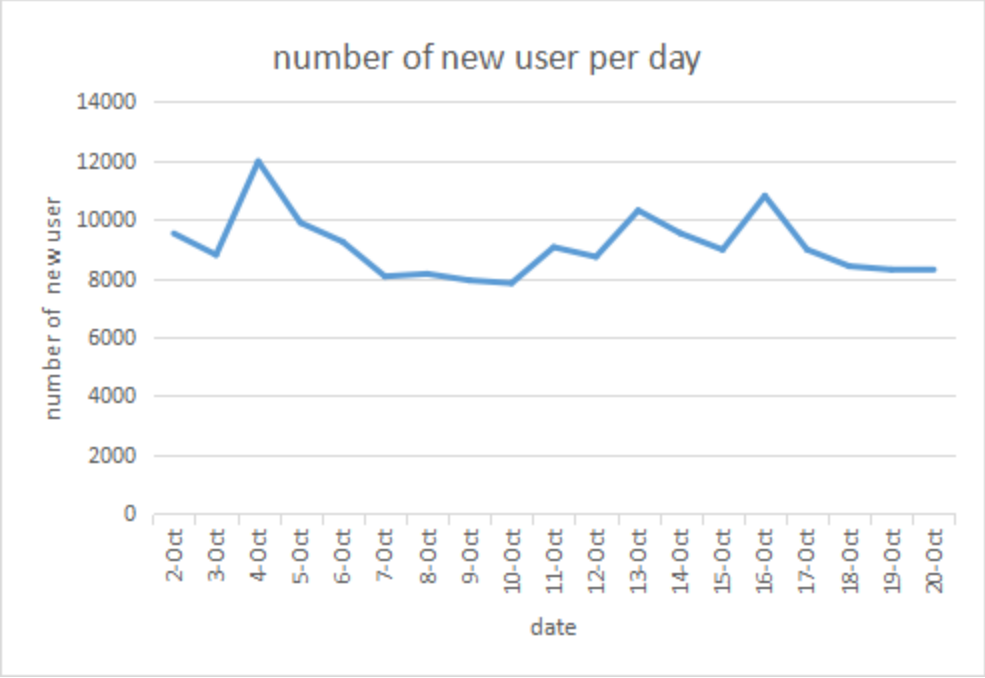
1. Drop the unnecessary data like “price” = 0 and drop the column which has null
2. Due to the limited memory, iterate through all the columns of a data frame and modify the datatype to reduce memory usage
3. Split the “event_time” by using “to_datetime”, split to “daily”, “weekday”, “date”
4. Sum up day by the user that divided by event type

5. Visualize various daily activities and combine them into one figure

6. Sum up hour by the user that divided by event type and visualizes to one figure

Results

The first part for me is going to use the AARRR funnel which is a method to understanding customers, their journey, and optimizing the funnel as well as setting some valuable and actionable metric goals



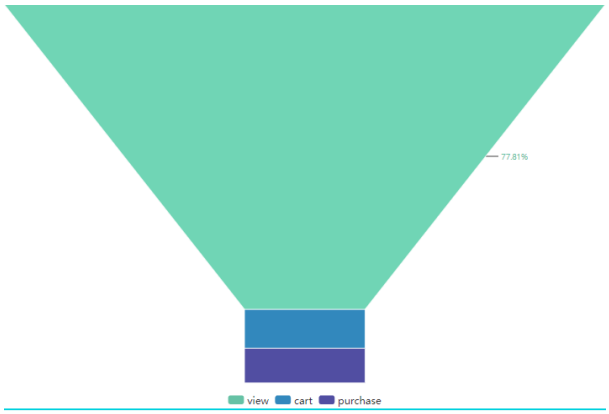
(1) New visitors: 2019-10-01 is selected as the first day, and the new users are defined as the users with the first purchase behavior. The figure for obtaining new users every day is as follows:

Since the data set only contains October's, we hypothesize 2019-10-1 as the first login date. The analysis shows that there are new visitors becoming customers every day. Especially in the first few days of the month.

(2)The conversion and loss of users' behaviors

The following chart shows the total number of users among different behaviors in October.

Event type	number
view	2322867
cart	294902
purchase	263445



Conversion rate of cart = cart number/view number = 294902/2322867 = 12.70%

Conversion rate of purchase = purchase number/view number = 263445/2322867 = 11.34%

We can find that the conversion rate of users of the platform from browsing to other behaviors is at a high level, and the activity of users is pretty good, indicating that the platform has enough attraction to users, indicating that the platform has enough attraction to users.

(3) Bounce rate

Definition: The bounce rate refers to the proportion of the number of visits that users enter through the corresponding entrance and leave after visiting only one page, accounting for the total number of visits to the page.

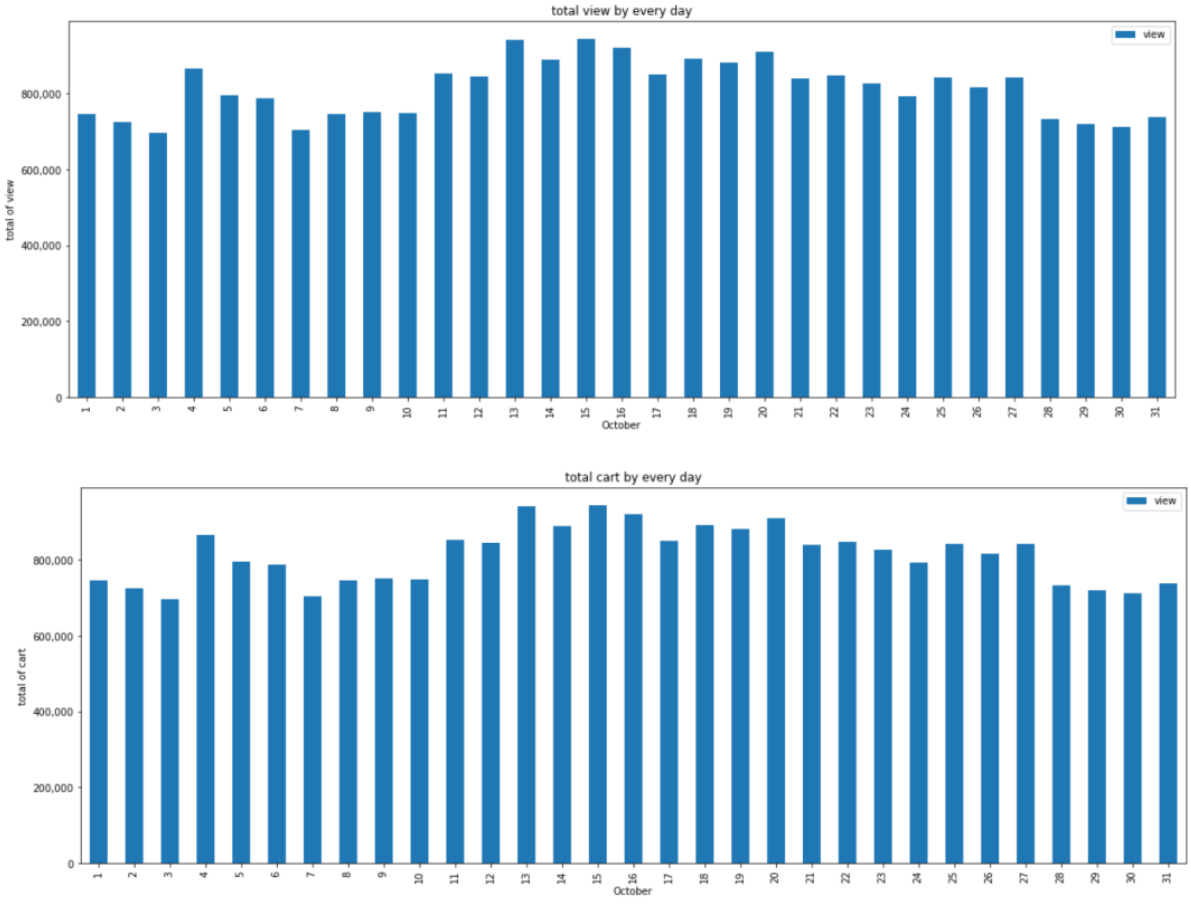
Bounce rate = single page visitors/total visitors = 1940918/2322867 = 83.56%

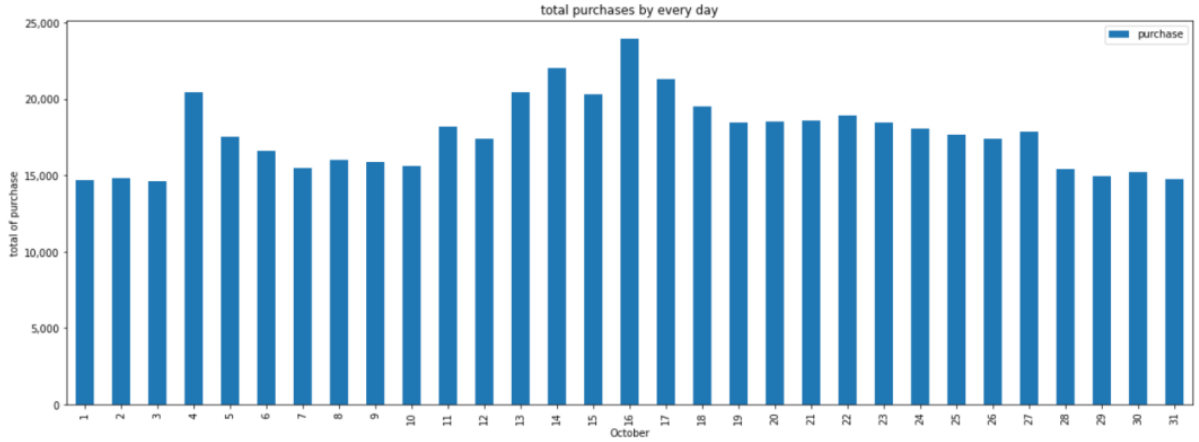
(4) The ratio of the total number of views of browsing-only users to the total number of views

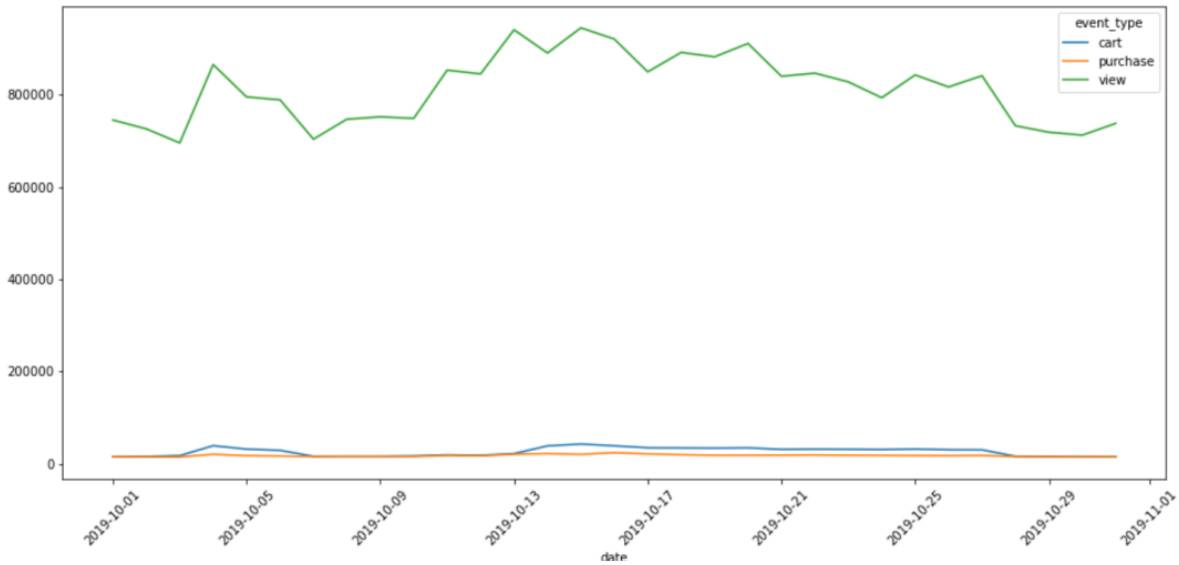
The number of users who only browsed the page without performing other actions/Total number of visitors = 564261 / 2322867= 24.29%

In the second part, I am planning to find the relationship between the number of orders made by users and the ordering time. Analyze user behavior patterns based on the time dimension.

(1) Different date distribution based on behavior



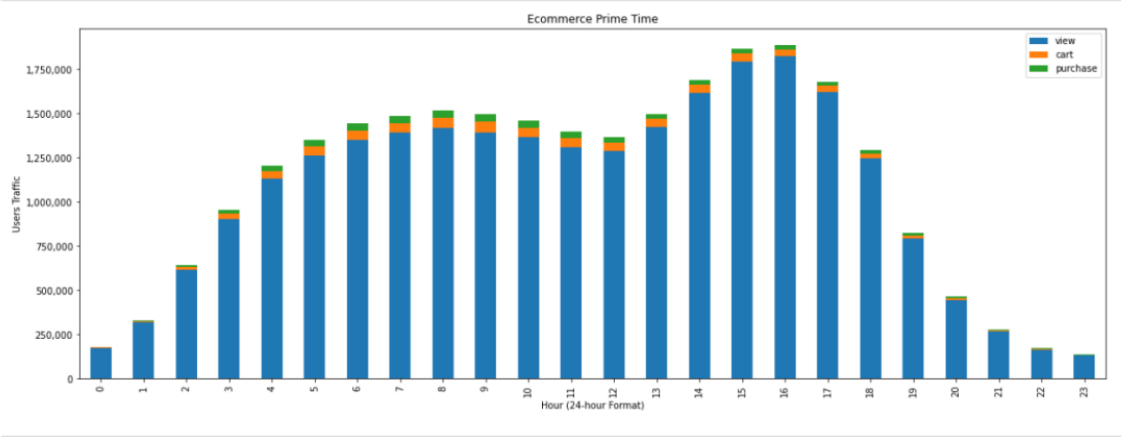




From the figures, we can figure out that the event type view, cart and purchase The general trend of increase and decrease is the same. What we need to pay attention to is there are four weekends which are roughly in line with our daily routines, there will be varying degrees of increase during the weekend. In the first and second weekends, user activity increased only slightly compared with the working day. While in the third weekend, user activity increased significantly. We

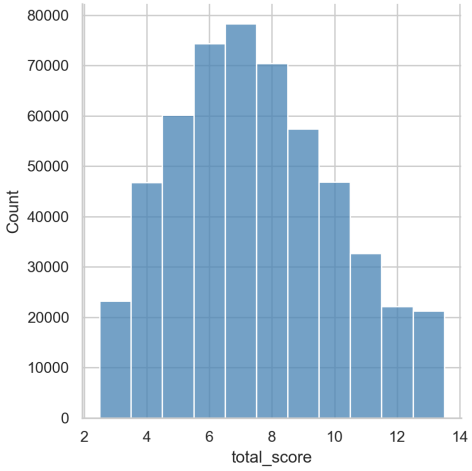
thought this is because of Columbus Day, some stores might have some discount promotions. Therefore, this platform can withdraw from marketing activities at weekends and festivals to tap users' purchase potential.

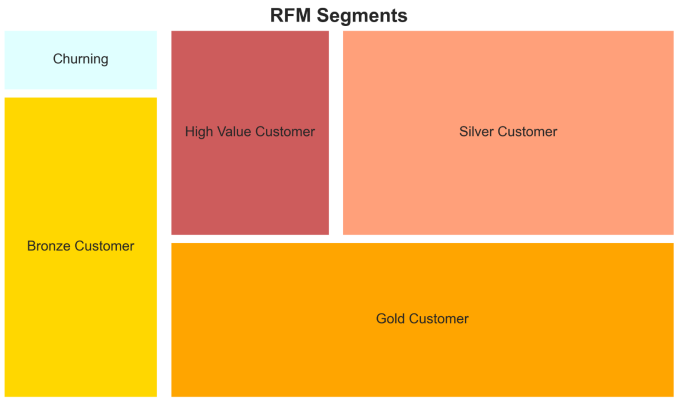
(2)Distribution at different time points based on behavior



The last chart shows the changes of user Various behaviors in 24 hours, with fluctuations close to each other. The peaks appear from 14:00 to 17:00 p.m., and then enter the trough at midnight, which is in line with the work and rest time of most people. From 20 p.m. to 23 p.m., interactive marketing means such as various live broadcasting activities may achieve greater benefits.

My first part is surrounding the topic of user clustering or user segmentation. I would assume that the result is somehow biased, as the data only contains two-month worth of customers and transactions. In the data description of the frequency field, more than 75% of the customers have only purchased once or twice over the two-month period. This means the scores of the frequency section would be the same for nearly 75%. The max number of 328 on frequency data description seems suspectable, such behavior is more of a bot or an individual merchandiser, who intends to track the prices of several items.





The histogram for the total score is slightly skewed left, where the range of 6 to 8 is, which means the mainstream customers of this online grocery store do not shop frequently. The treemap of the RFM model further clarifies this pattern, where bronze, silver, and gold customers consist the main part of the whole customer group, while the high-value customer is about half of the silver customer. And there are churning ones who either spent less or shop less.

The second part is about association rules, where I wanted to see if there's any highly repurchased items or itemsets. I applied the apriori model and set the min_support value as 0.005 so that the itemset occurs in 0.5% of the transactions; min_confidence as 0.25 so that we know whenever the customer bought items in the precedent, 25% of the times he/she also bought items in the antecedent; and min_length as 2 so that at least 2 items in an itemset that a rule should have. For the min_confidence, I also tested min_confidence in a range of 0.15 to 0.3, with 0.15 being too many rules and 0.3 being too few rules, and 0.2 has about the same rules as 0.25.

[RelationRecord(items=frozenset({'smartphone'}), support=0.6033093940461643, ordered_statistics=[OrderedStatistic(items_base=frozenset(), items_add=frozenset({'smartphone'}), confidence=0.6033093940461643, lift=1.0)]),

RelationRecord(items=frozenset({'clocks', 'smartphone'}), support=0.013677629260905745, ordered_statistics=[OrderedStatistic(items_base=frozenset({'clocks'}), items_add=frozenset({'smartphone'}), confidence=0.32125720825813264, lift=0.5324916393288425)]),

RelationRecord(items=frozenset({'headphone', 'smartphone'}), support=0.023361420764208017, ordered_statistics=[OrderedStatistic(items_base=frozenset({'headphone'}), items_add=frozenset({'smartphone'}), confidence=0.29669388046557016, lift=0.4917773258522601)]),

RelationRecord(items=frozenset({'notebook', 'smartphone'}), support=0.011331179297264474, ordered_statistics=[OrderedStatistic(items_base=frozenset({'notebook'}), items_add=frozenset({'smartphone'}), confidence=0.2979646148538761, lift=0.49388359901963713)]),

RelationRecord(items=frozenset({'tv', 'smartphone'}), support=0.015407480152631697, ordered_statistics=[OrderedStatistic(items_base=frozenset({'tv'}), items_add=frozenset({'smartphone'}), confidence=0.25285270507181745, lift=0.4191095109194828)])]

As shown above, there are five rules generated and all of them contain smartphone items, which is expected since, within two months period, this online grocery store has sold a quantity of 720067 smartphones, putting the smartphone ranks No. 1 in its sales ranking. The itemsets from these rules show that customers prefer purchasing digital products from this store, such as the rules of {'headphone', 'smartphone'}, {'notebook', 'smartphone'}. As the data implies, the smartphone is the main selling product, with 12,000 or so amounts being sold per day within two months. On the other hand, I would assume that this online store also operates its business with small to mid-sized firms instead of selling to individual customers only. And it is probably operating the OEM model.

Future work possibility discussion

Due to the hardware limitation and the data set being large enough, many previously envisaged results have not been achieved. In the future, we are supposed to deploy this program on the server.

What's more, we can analyze this data set in different dimensions, Through multi-angle analysis, we can provide a more meaningful suggestion for this platform.

Conclusion

Here is some advice we can offer:

1. Optimize the search engine and recommendation algorithm of the platform, and make personalized recommendations according to multiple factors such as users' love, age, and income. Which can make people stay on the page longer.
2. Provide users with recommendations for the same type of goods purchased and cart by users, to facilitate users to browse and determine a purchase.
3. For the personal stores, some preferential activities related to collection and purchase can be launched to help improve the collection and purchase volume.
4. Establish user value levels through the RFM model, and put forward management recommendations for users of various levels. Bronze, silver, and gold customers constitute the main part of the entire customer base that needs to be taken care of.
5. By using the apriori model customers prefer to buy digital products from this store, the platform can try to find the customer base who likes to use the platform to analyze their other characteristics

Reference

- *https://towardsdatascience.com/recency-frequency-monetary-model-with-python-and-how-sephora-uses-it-to-optimize-their-google-d6a0707c5f17*
- *https://matplotlib.org/stable/gallery/color/named_colors.html*
- *https://medium.com/analytics-vidhya/data-visualization-and-rfm-recency-frequency-and-monetary-analysis-using-python-customer-d7e129437aac*
- *https://stackoverflow.com/*
- *eCommerce behavior data from multi category store | Kaggle*
- *https://blog.csdn.net/weixin_43388963/article/details/100149563*