CUSTOMER BEHAVIOR MODELING

Applied Statistics and Experimental Design
Assoc.Prof Nguyen Linh Giang
Semester 20232

Group 13

DAO VAN TUNG

20204932

NGUYEN TRI THANH

20225457

TRIEU KINH QUOC

20225524

NGUYEN XUAN THANH

20225460

Outline

OUTLINE OF DISCUSSION TOPICS

1. Problems approaches

2.Data overview and exploration

3.Association rule using Apriori

4.Predicting users' next products

5. Conclusion

Problems Approaches

Our approach is to find and extract business-benefit data. There are two ways we'll direct our course:

- The first strategy uses a classification model to determine if a client will make a subsequent purchase of an item they previously purchased.
- Utilizing statistical association rules called the Apriori algorithm to identify a group of objects or times (the purchasing habit rule effect) that go together.

Both tactics will provide us with some insights into how consumers think and act, as well as the trends that each cart reveals

II

Data exploratory analysis and feature engineering

Data overview

The dataset for this project is a relational set of files describing customers' order over time. This dataset contains the information of 3 million grocery orders from over 200,000 Instacart users. For each user, there are between 4 and 100 orders with the time the order was placed, the time between different orders. Each entity has an associated unique ID.

- aisle.csv: gives unique id to each aisle available in the market. There are 134 aisle categories in the market.

aisle_id	aisle
1	prepared soups salads
2	specialty cheeses
3	energy granola bars
4	instant foods
5	marinades meat preparation
6	other
7	packaged meat
8	bakery desserts
9	pasta sauce
10	kitchen supplies
11	cold flu allergy

The first 11 categories in aisle.csv

Data Overview

departme	department
1	frozen
2	other
3	bakery
4	produce
5	alcohol
6	international
7	beverages
8	pets
9	dry goods pasta
10	bulk
11	personal care
12	meat seafood
13	pantry
14	breakfast
15	canned goods
16	dairy eggs
17	household
18	babies
19	snacks
20	deli
21	missing

- The departments.csv stores all departments of the products.
- There are total 21 departments in the market.

product_id	product_name	aisle_id	departme
1	Chocolate Sandwich Cookies	61	19
2	All-Seasons Salt	104	13
3	Robust Golden Unsweetened Oolong Tea	94	7
4	Smart Ones Classic Favorites Mini Rigatoni With Vodka Cream Sauce	38	1
5	Green Chile Anytime Sauce	5	13
6	Dry Nose Oil	11	11
7	Pure Coconut Water With Orange	98	7
8	Cut Russet Potatoes Steam N' Mash	116	1
9	Light Strawberry Blueberry Yogurt	120	16
10	Sparkling Orange Juice & Prickly Pear Beverage	115	7
11	Peach Mango Juice	31	7
12	Chocolate Fudge Layer Cake	119	1
13	Saline Nasal Mist	11	11

- The products.csv provides all the informations of products in the market.
- There are 49688 products.

Data Overview

order_id	product_id	add_to_cart_	reordered
1	49302	1	1
1	11109	2	1
1	10246	3	0
1	49683	4	0
1	43633	5	1
1	13176	6	0
1	47209	7	0
1	22035	8	1
36	39612	1	0
36	19660	2	1
36	49235	3	0
36	43086	4	1

- The orders_products.csv contains all the orders which includes: the id of the order, the products, the order that customer added to cart.
- Reorder indicates that the customer has a previous order that contains the products.

order_id	user_id	eval_set	order_number	order_dov	order_hour_of_da	days_since_prior_order
2539329	1	prior	1	2	8	
2398795	1	prior	2	3	7	15
473747	1	prior	3	3	12	21
2254736	1	prior	4	4	7	29
431534	1	prior	5	4	15	28
3367565	1	prior	6	2	7	19
550135	1	prior	7	1	9	20
3108588	1	prior	8	1	14	14
2295261	1	prior	9	1	16	0
2550362	1	prior	10	4	8	30

- The orders.csv represents the full information of each order: order id, customer id, set of the order that it belongs to, the numbers of each order and the time(hour) of the order as well as days since prior order.

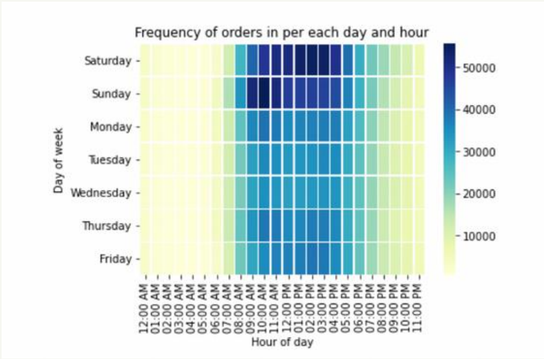
Data exploration and engineering

STEP 1: REMOVE FEATURES

- Remove leaking features

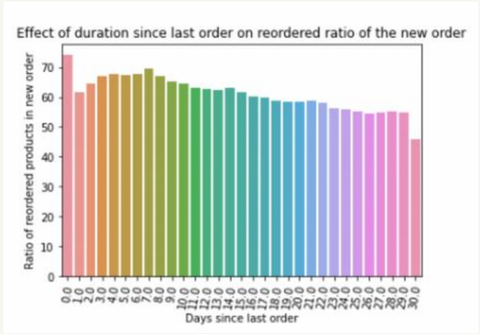
STEP2: DEALING WITH ORDERING TIME FEATURES

- Deep analysis about ordered time



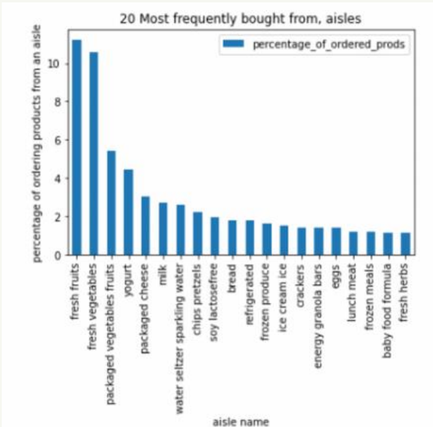
STEP3: DEALING WITH PRODUCTS FEATURES

- Explore more about products

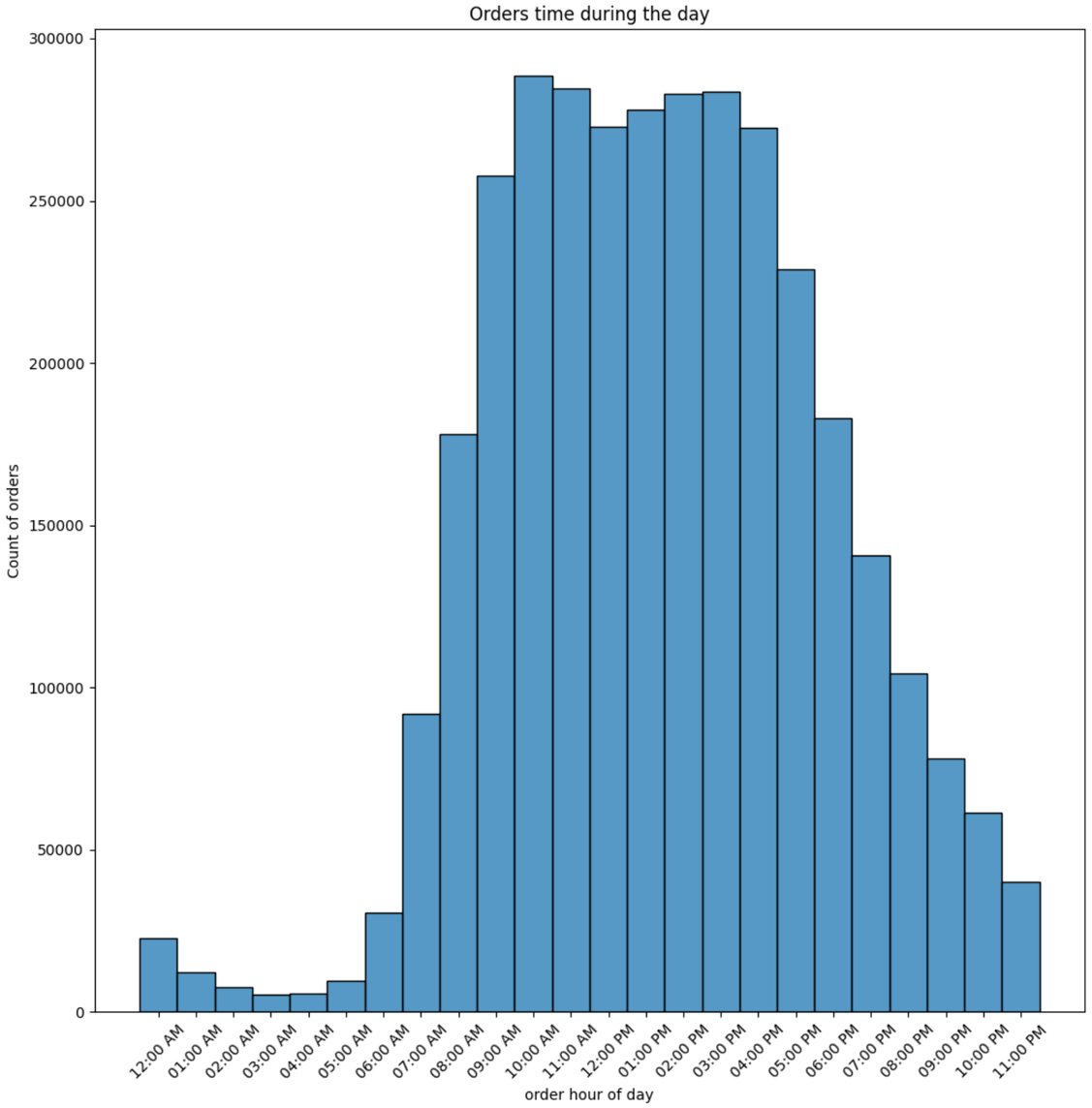


STEP4: DEALING WITH PRODUCTS FEATURES

- Deep dive in Departments and Aisles

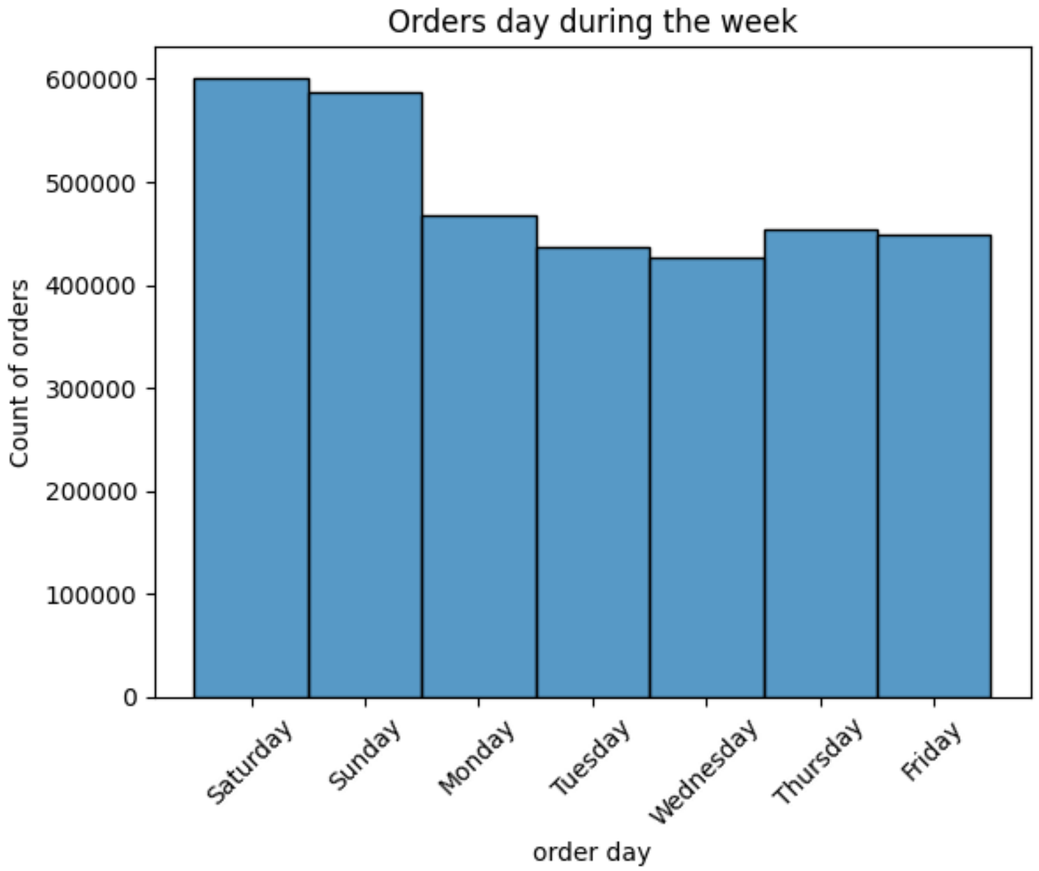


ORDERING TIME FEATURES



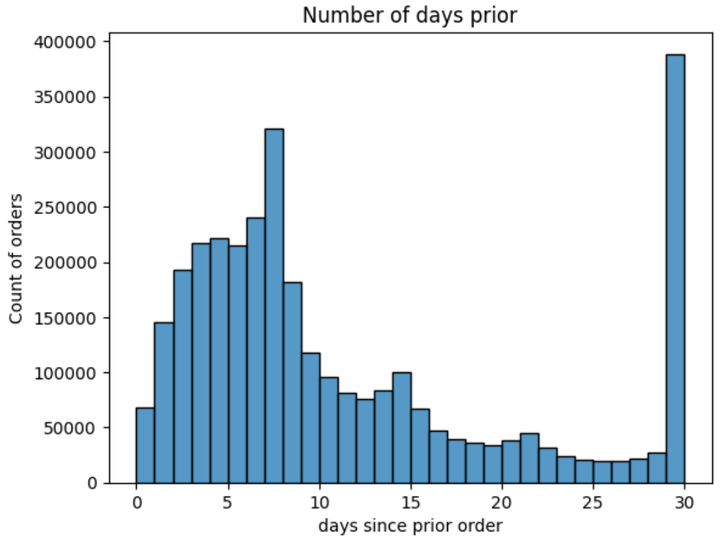
- Most products are ordered around 9:00 AM to 5:00 PM
- After 5:00 PM, the number of customers decreases
- Very few customers place orders at midnight

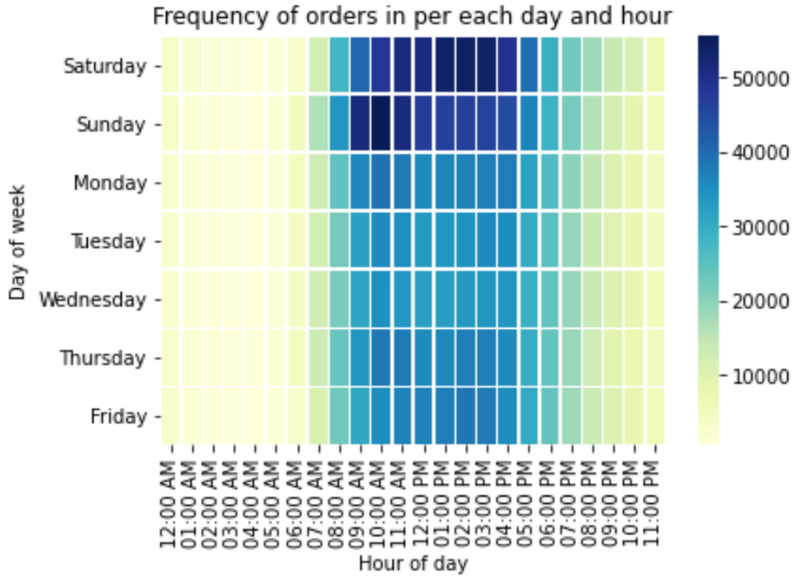
ORDERING TIME FEATURES



- The number of orders remained stable during the whole week
- Slightly high in the weekends

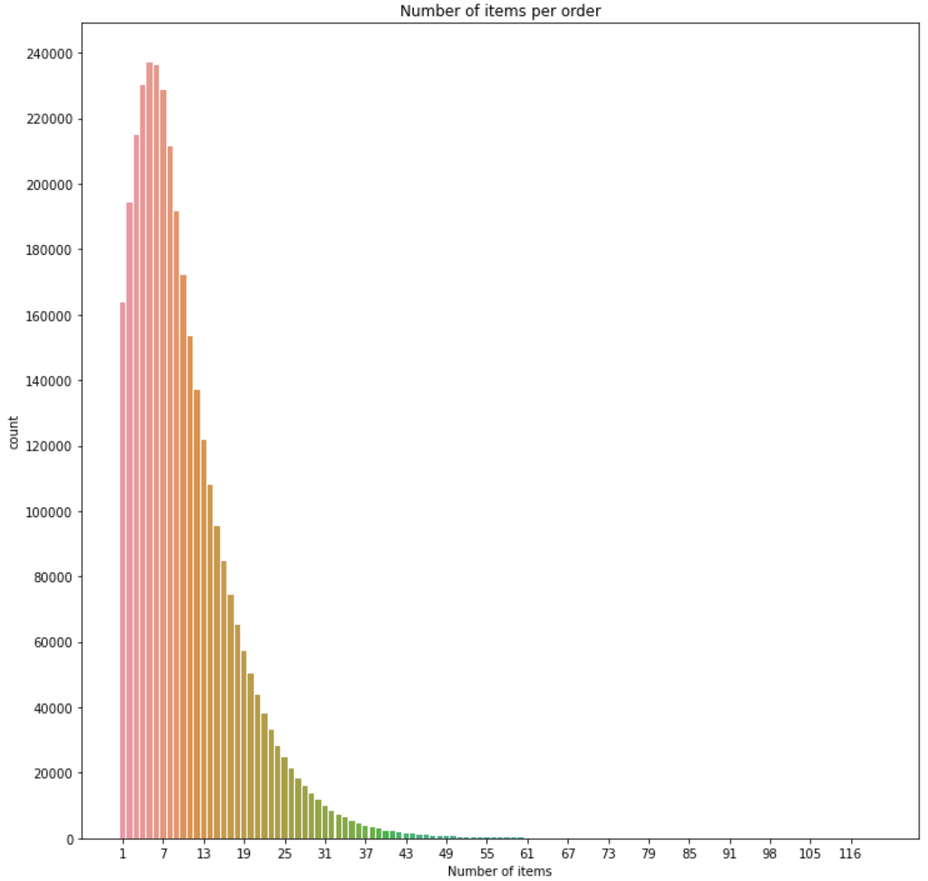
ORDERING TIME FEATURES





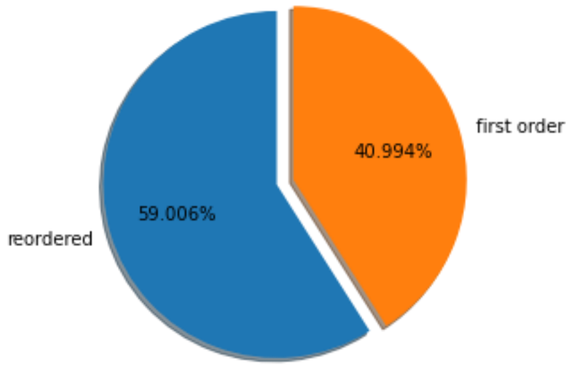
- Most customers place their orders around 3 to 8 days after their previous one.
- Customers seem to reorder after one week.
- Many orders are 30 days since prior order as all the values greater than 30 are coerced to 30.
- Saturday afternoon and Sunday morning seem to be the prime time for ordering.

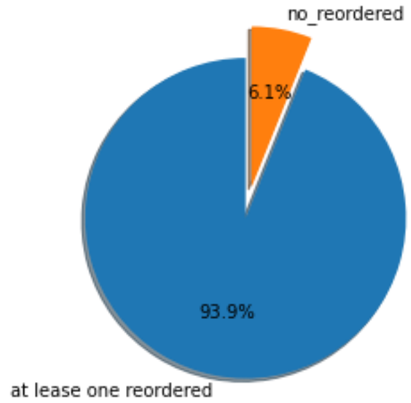
PRODUCTS FEATURES



The number of items per order of all customers

PRODUCTS FEATURES

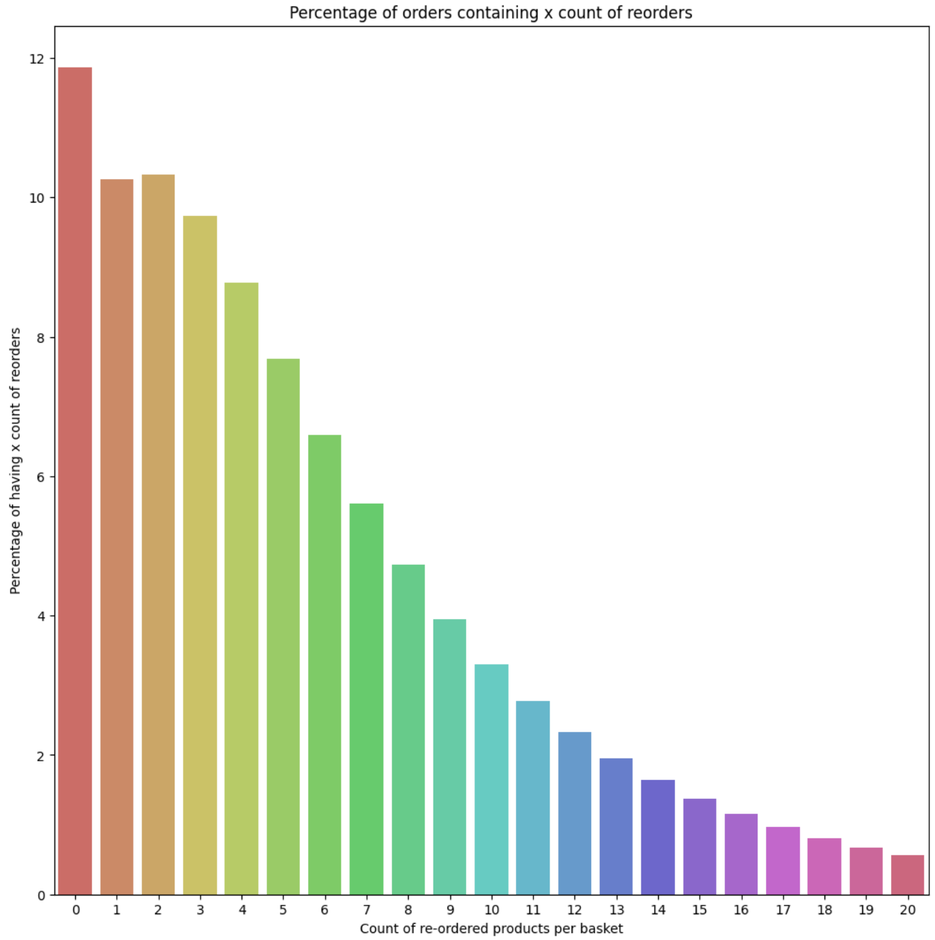




- 59% of products are reordered.
- 41% of products are the first time being ordered.
- 93,9% of orders contain at least one reordered product.

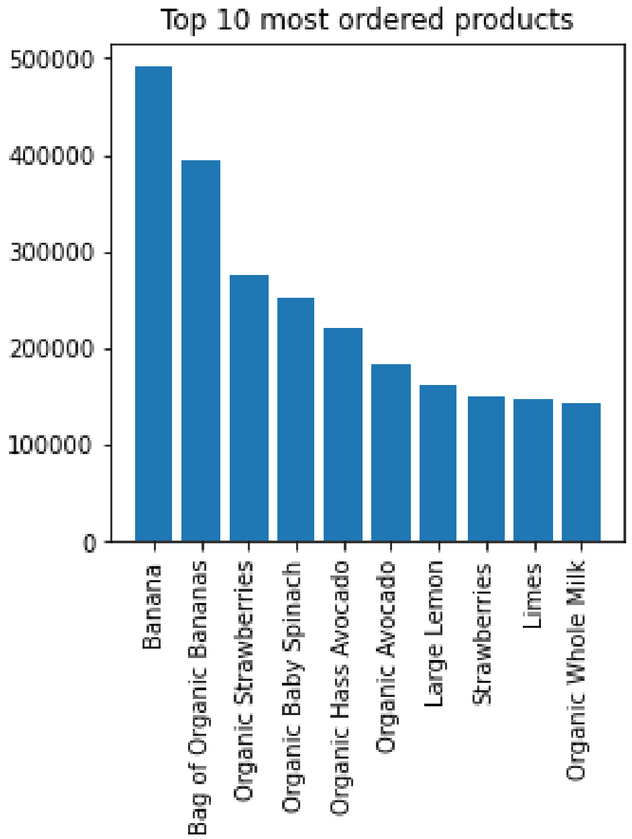
➔ Customers have a high tendency to rebuy products that they have already ordered and used before

PRODUCTS FEATURES

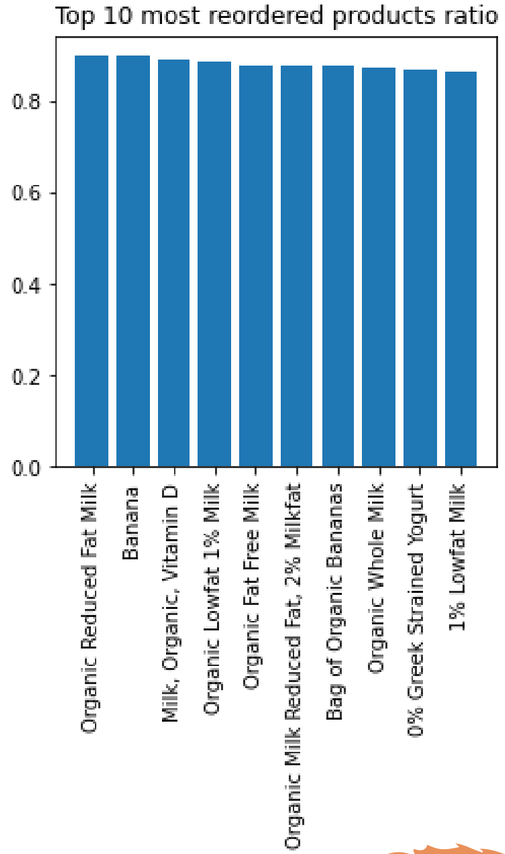


Distribution of the number of reordered products per basket

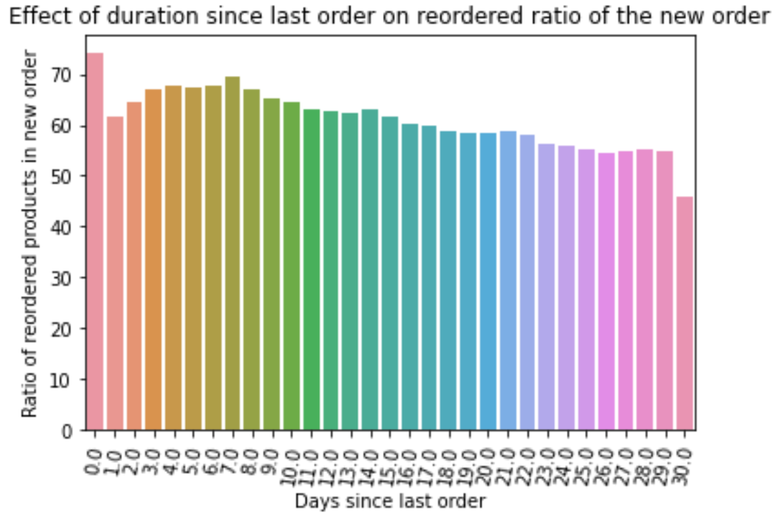
PRODUCTS FEATURES

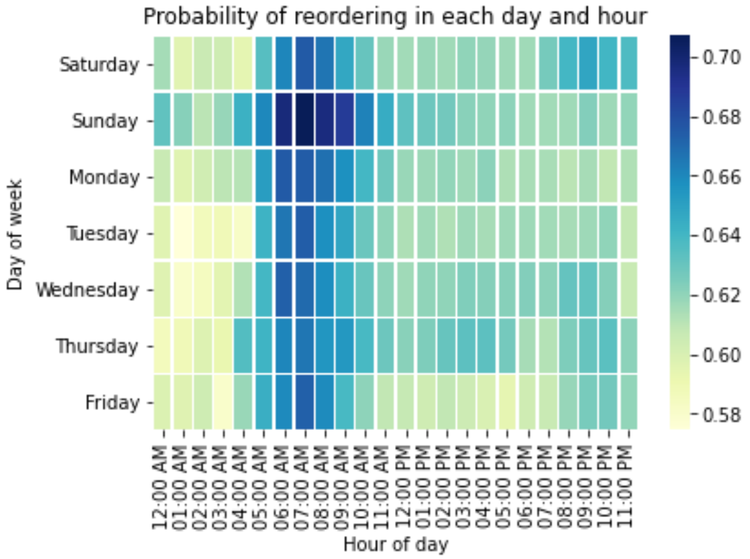


- Banana is the bestseller, followed by Organic straw - berries and organic baby spinach.
- Organic reduced fat milk is the product with the highest probability of being reordered



PRODUCTS FEATURES

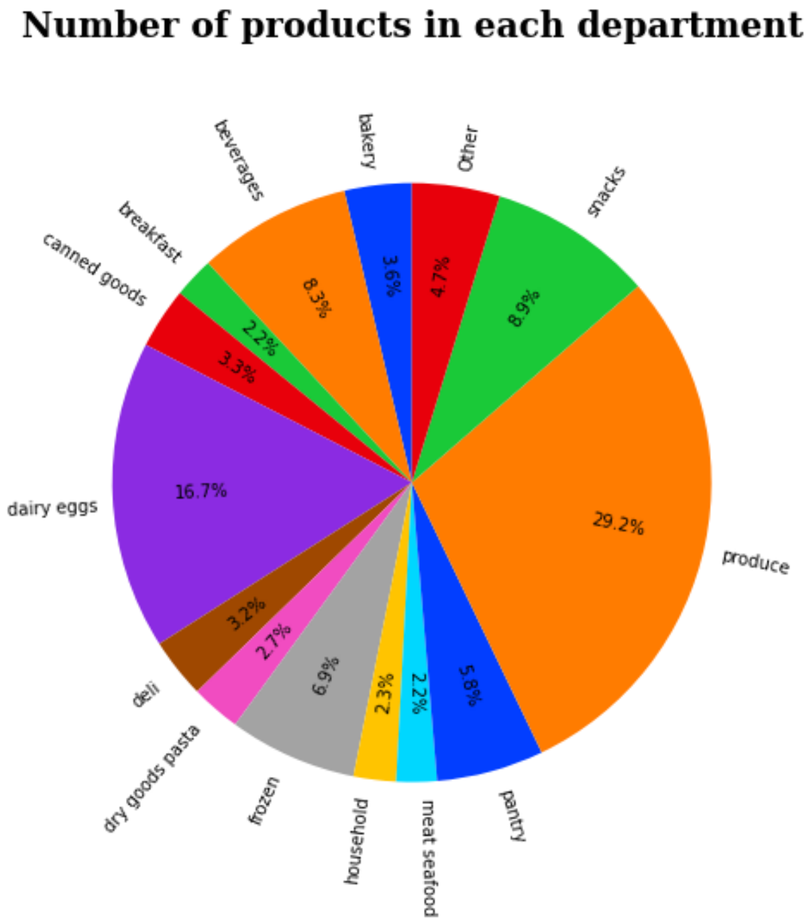


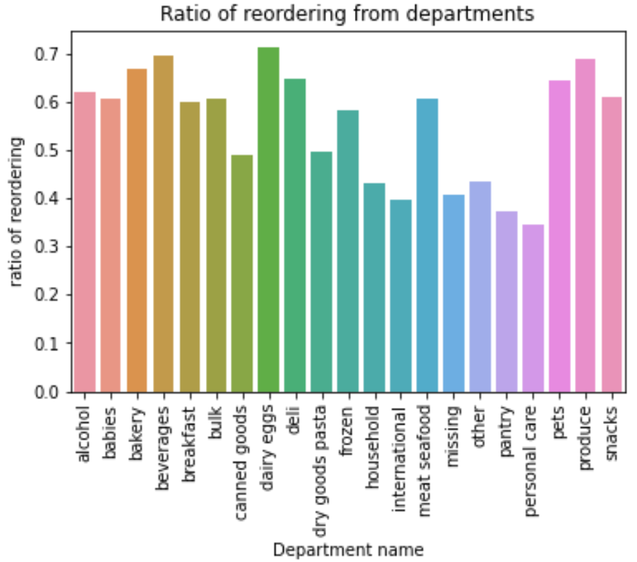


- 74% of products bought at the same day of previous order are reordered.
- 69% of products bought after 1 week of previous order, are reordered.
- Customers tend to reorder products around 6AM to 8AM, especially on Sunday

➔ These are the time we should recommend customers with products that they have bought before

DEPARTMENT AND AISLE

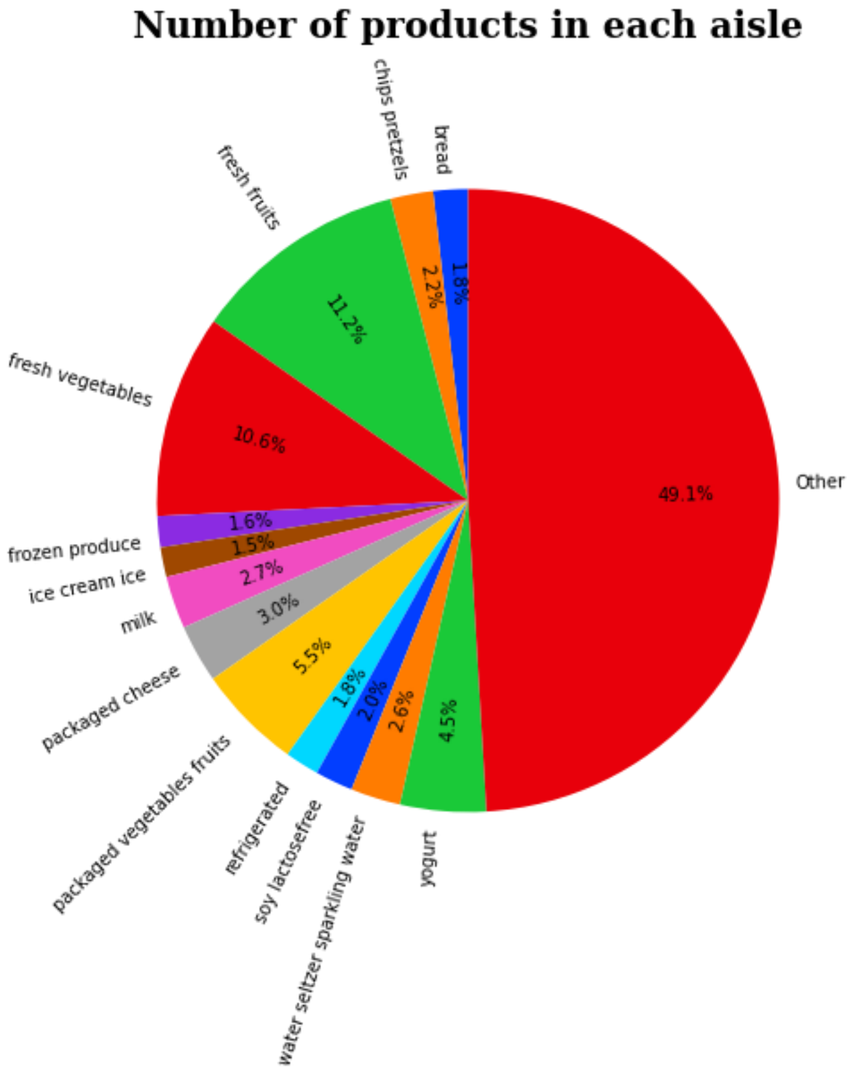




- The **produce** department has the most number of products, almost double the **dairy egg** department in the second place.

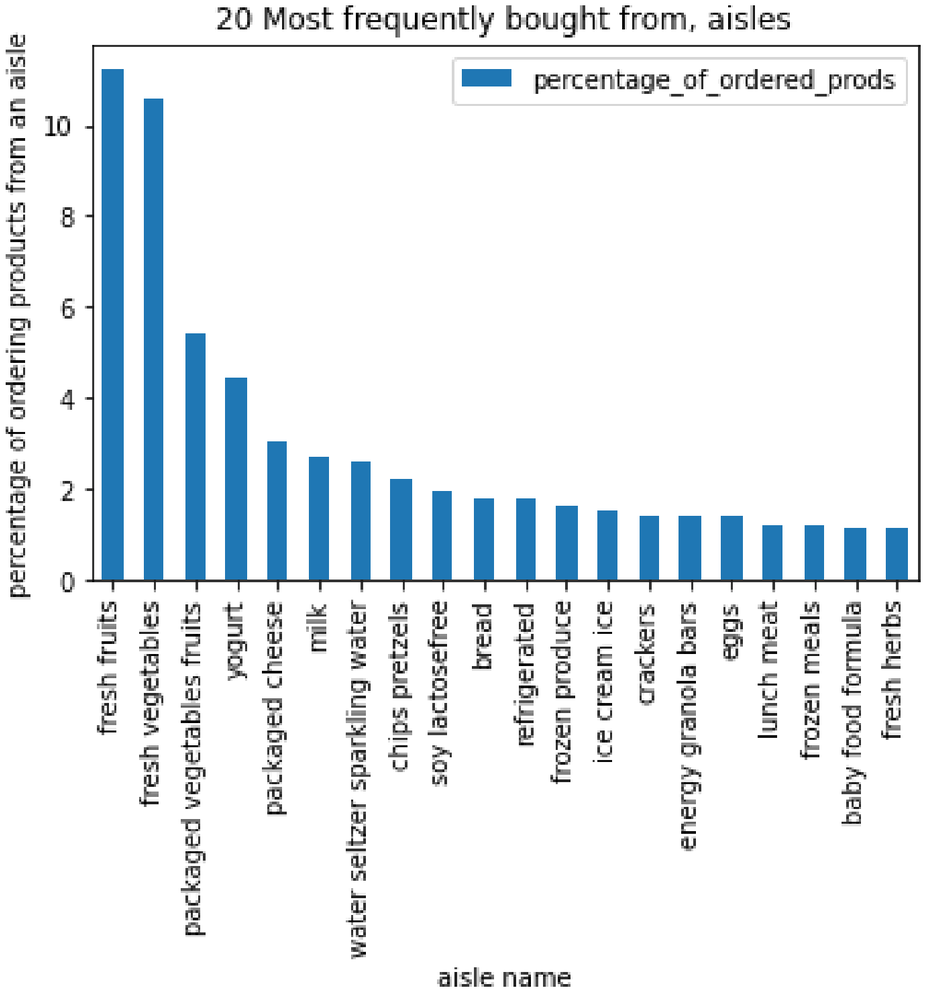
- **Dairy eggs, produce, pets, snacks, beverages, bakery, deli, meat - seafood, bulk, alcohol** are ten departments with the most reordered products.

DEPARTMENT AND AISLE



- 134 aisles in total with 12 aisles accounting for 51% of products.
- Fresh fruits and fresh vegetables are the two aisles with the most products,11.2% and 10.6% respectively

DEPARTMENT AND AISLE



- Fresh fruits and fresh vegetables are the two most frequently bought aisles, with around 14% and 12%, respectively.

- Familiar names like milk, yogurt, cheese, bread, eggs and water are most frequently bought from aisles.

III

Association rule using Apriori

Apriori

- **Support:** Refers to the default popularity of any product.

$$\text{support}(A) = \frac{\text{number of transactions containing } A}{\text{total number of transactions}} \tag{1}$$

- **Confidence:** Refers to the possibility that the customers bought both products A and B together.

$$\text{confidence}(A \rightarrow B) = \frac{\text{number of transactions containing } A \text{ and } B}{\text{number of transactions containing } A} \tag{2}$$

- **Lift:** Refers to the increase in the ratio of the sale of A when you sell B. On the other hand, if the lift value is below one it requires that the people are unlikely to buy both the items together. Larger the value, the better the combination.

$$\text{lift}(A \rightarrow B) = \frac{\text{confidence}(A \rightarrow B)}{\text{support}(B)} \tag{3}$$

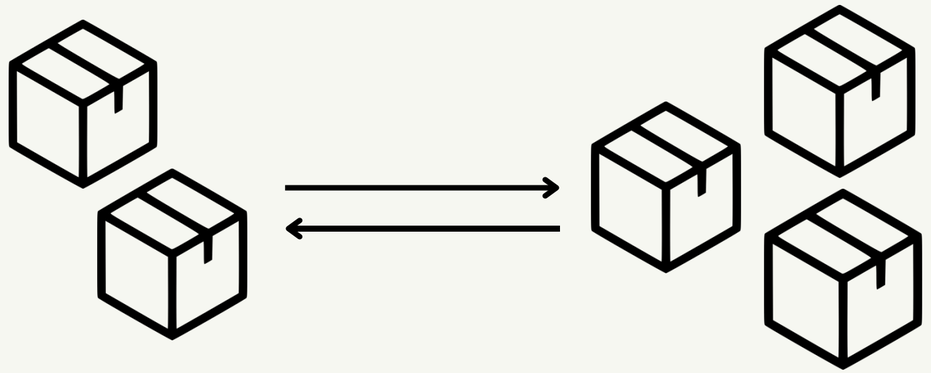
Products - products Apriori

Dataset

Apriori

Products

Lots of product



Data Selection:

- Chose the train set from the orders table.
- Selected products with consumption between 1500 and 3500.
- Considered orders with at least three items to analyze meaningful associations.

Products - products Apriori

Dataset

Apriori

Products

Lots of product

Implement:

1. Start with individual items and calculate their support.
2. Retain itemsets meeting the minimum support threshold (0.003).
3. Generate combinations of itemsets and compute their lift with a threshold of 1.2.
4. Iterate until no new itemsets are generated.

Products - products Apriori

antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
(Soda)	(Hass Avocados, Raspberries)	0.003227	0.004528	0.000340	0.105263	23.244737	0.000325	1.112586
(Hass Avocados, Raspberries)	(Soda)	0.004528	0.003227	0.000340	0.075000	23.244737	0.000325	1.077593
(Small Hass Avocado, Raspberries)	(Blueberries, Organic Blackberries)	0.004472	0.007076	0.000623	0.139241	19.678582	0.000591	1.153544
(Blueberries, Organic Blackberries)	(Small Hass Avocado, Raspberries)	0.007076	0.004472	0.000623	0.088000	19.678582	0.000591	1.091588
(Organic Granny Smith Apple, Jalapeno Peppers)	(Organic Kiwi, Organic Small Bunch Celery)	0.003000	0.005830	0.000340	0.113208	19.416743	0.000322	1.121085

Top Support Itemsets:

- Sparkling Water Grapefruit: 7.91%
- Raspberries: 6.21%
- Organic Fuji Apple: 8.17%
- Small Hass Avocado: 9.01%
- Broccoli Crown: 9.10%

Association Rules:

- Organic foods often purchased with diet foods.
- Example: Organic Fuji Apple with Jalapeno Peppers, Roma Tomatoes.

Products - time Apriori

Dataset

Apriori

Daily timeline

Lots of products

Data Preparation:

- Filtered train set from the orders table.
- Merged with order products train set and product set.
- Limited product range to 600-650 to balance healthy and unhealthy food items.
- Encoded data into categories and time frames for dummy variable creation.

Products - time Apriori

Dataset

Apriori

Daily timeline

Lots of products

Implement:

1. Compute support for all possible combinations among times and items with a minimum threshold of 0.003.
2. Compute lift for combinations among times and items with a minimum threshold of 1.0.
3. Identify top associations and analyze how product purchases relate to specific times.

Products - time Apriori

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	antecedents_length	consequer
0	(weekday, 100% Lactose Free Fat Free Milk)	(early_morning_hours (<7))	0.012808	0.026269	0.000949	0.074074	2.819831	0.000612	1.051630	2	
1	(early_morning_hours (<7))	(weekday, 100% Lactose Free Fat Free Milk)	0.026269	0.012808	0.000949	0.036117	2.819831	0.000612	1.024182	1	
2	(weekday, early_morning_hours (<7))	(100% Lactose Free Fat Free Milk)	0.018797	0.019568	0.000949	0.050473	2.579333	0.000581	1.032548	2	
3	(100% Lactose Free Fat Free Milk)	(weekday, early_morning_hours (<7))	0.019568	0.018797	0.000949	0.048485	2.579333	0.000581	1.031200	1	
4	(Shredded Hash Browns, weekday)	(early_morning_hours (<7))	0.013223	0.026269	0.000830	0.062780	2.389902	0.000483	1.038957	2	
5	(early_morning_hours (<7))	(Shredded Hash Browns, weekday)	0.026269	0.013223	0.000830	0.031603	2.389902	0.000483	1.018979	1	

Early Morning Purchases:

- Associated with healthy foods: Mineral Water, Lactose-Free Milk, Chicken Breast.

Late Night Purchases:

- Associated with unhealthy foods: Pizza, Ramen, Coke.

Implications for Sales Strategy:

- Tailor recommendations based on time of day.
- Promote healthy foods in the morning and indulgent foods at night.

IV

Predicting users' next products

Features engineering

User related features:

- user_total_orders: Total number of orders by user
- user_total_items: Total number of items the user had ordered
- total_distinct_items: Total number of distinct items purchased by each user.
- user_average_days_between_orders: average days between orders
- user_average_basket: average number of item in each order

Features engineering

Order related features:

- order_hour_of_day: The hour the order is placed

- days_since_prior_order: Number of days since the last order

- days_since_ratio: days_since_prior_order / user_average_days_between_orders

Features engineering

Product related features:

- product_orders: total number of orders of a product
- organic: check if the product is organic or not
- product_reorders: total number of reorders of a product
- product_reorder_rate: product_reorders/product_orders

Features engineering

UserXProduct (UP) features:

- UP_orders: total number of a product ordered by a user
- UP_orders_ratio: UP_orders / user_total_orders
- UP_last_order_id: ID of the last order
- UP_average_pos_in_cart: average position of a product when the user purchased it

Features engineering

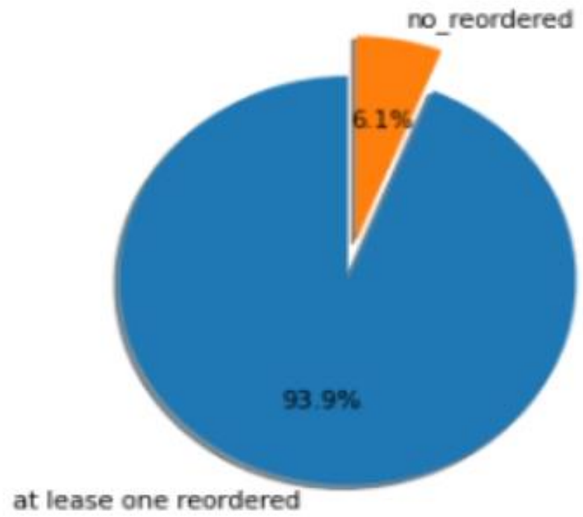
UserXProduct (UP) features:

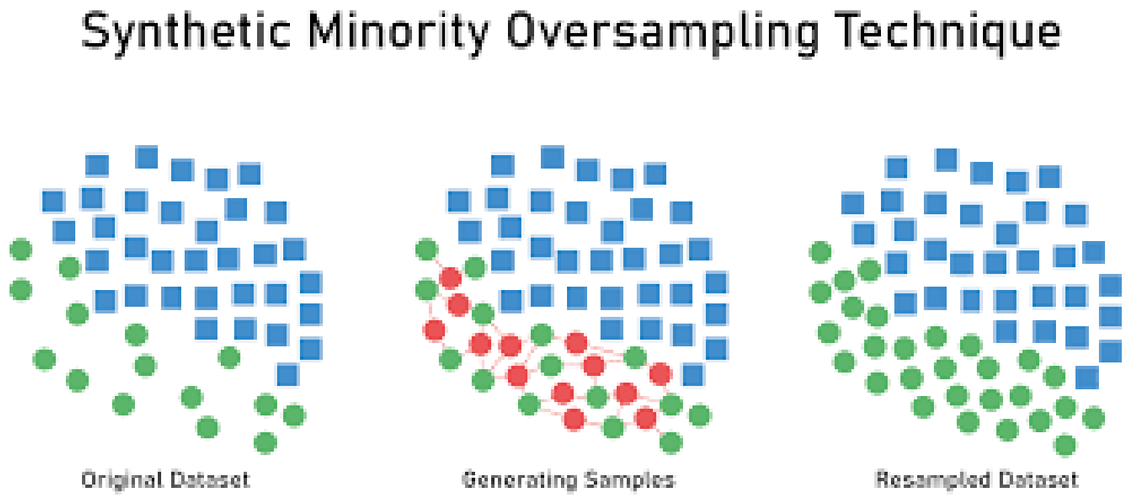
- UP_reorder_rate: UP_orders / user_total_orders

- UP_orders_since_last: user_total_orders - UP_last_order_id

- UP_delta_hour_vs_last: difference between the hour that the user makes the current order and that of the previous order

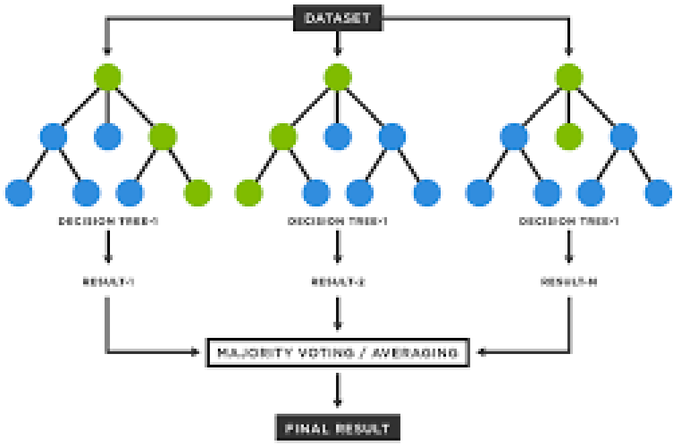
Dealing with imbalance classes

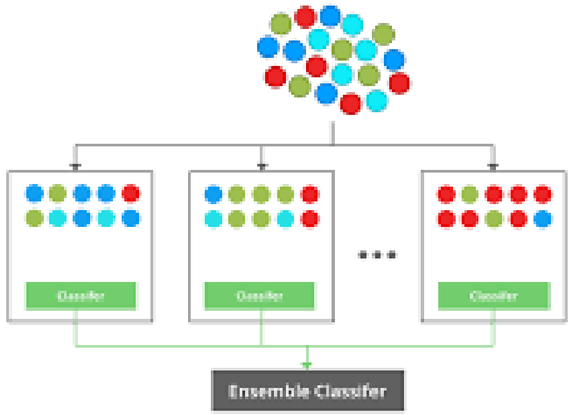


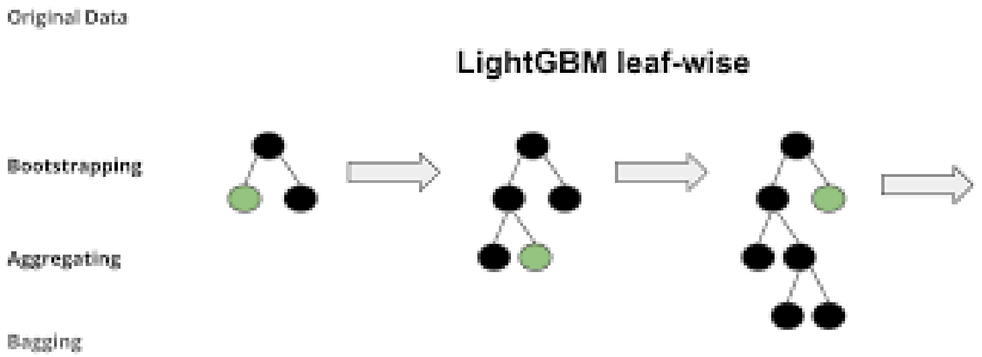


- Due to the fact that our dataset class has a skewed proportion, where the reordered class is around 10% of the whole dataset, SMOTE (Synthetic Minority Oversampling Technique) is chosen to overcome the imbalance problem.
- **Main idea of SMOTE**:
  - First selects a minority class instance at random and finds its k nearest minority class neighbors.
  - The synthetic instance is then created by choosing one of the k nearest neighbors b at random and connecting a and b to form a line segment in the feature space.
  - The synthetic instances are generated as a convex combination of the two chosen instances a and b.

Classification technique







The techniques that will be used to classify the two classes are: RandomForest, Extreme gradient boosting (XGBoost) and Light Gradient Boosting Machine (LGBM). Random Forest is currently one of the most popular and accurate methods that can be implemented easily and efficiently whereas XGBoost and LGBM have become the de-facto algorithm for winning numerous competitions just because they are too powerful.

Experimental results

Table 1: F1-score of all classification methods

Method	F1-score	
	Without SMOTE	With SMOTE
RandomForest	0.18258	0.31546
XGBoost	0.34518	
LGBM	0.23972	0.33209

- XGBoost gives the highest F1-score, followed by LGBM and RandomForest.
- Performance of Random Forest and LGBM improved drastically after oversampling with SMOTE, proving its effectiveness in imbalance class problems.
- XGBoost works well with an imbalanced dataset even when not applying SMOTE by making use of the parameter *scale_pos_weight* to set the suitable weight for each class.
- Three classifiers perform really well in classifying the not reordered class, while underperforming in the reordered class.

Conclusion

Apriori rules and some association rules are used to demonstrate a slightly better way to survey customers' habits. We have shown some new insight about customer's habit when applying these algorithms such as predicting whether or not the customer will reorder an item and customers' behaviour relation to their orders. In the future, we will combine other strategies, Apriori and other well-scored methods to predict customers' movement