Instacart Market Basket Analysis Dataset

COURSE PRESENTER

(DR. Omaima Fallatah)

Submitted by:

Name	ID
Manar Ali Alsubhi	444003523
Jana Abdulraouf Allihyani	444001382

DEPARTMENT OF (INFORMATION SCIENCE-DATA SCIENCE)

COLLEGE OF COMPUTER AND INFORMATION SYSTEMS

UMM AL-QURA UNIVERSITY

Table of Contents

- # Introduction

The dataset for this competition is a relational set of files describing customers' orders over time. The goal of the competition is to predict which products will be in a user's next order. The dataset is anonymized and contains a sample of over 3 million grocery orders from more than 200,000 Instacart users. For each user, we provide between 4 and 100 of their orders, with the sequence of products purchased in each order. We also provide the week and hour of day the order was placed, and a relative measure of time between orders. For more information, see the blog post accompanying its public release.

- # Goals of the Study

Increase Sales

Identify frequently purchased product combinations to create targeted promotions and cross-selling strategies.

Improve Store Layout

Optimize product placement by positioning frequently bought together items near each other to enhance convenience and sales.

Enhance Customer Experience

Personalize recommendations based on previous purchasing patterns to improve customer satisfaction and loyalty.

Optimize Inventory Management

Forecast demand for certain products based on related sales, ensuring optimal stock levels and reducing waste.

1. Exploratory Data Analysis

Reviewing the initial rows of these dataframes helps understand the structure and content of each dataset.

Displays the first few rows of the products dataframe.

	product_id	product_name	aisle_id	department_id
0	1	Chocolate Sandwich Cookies	61	19
1	2	All-Seasons Salt	104	13
2	3	Robust Golden Unsweetened Oolong Tea	94	7
3	4	Smart Ones Classic Favorites Mini Rigatoni Wit...	38	1
4	5	Green Chile Anytime Sauce	5	13

Displays the first few rows of the order_products__train dataframe.

	order_id	product_id	add_to_cart_order	reordered
0	1	49302.0	1.0	1.0
1	1	11109.0	2.0	1.0
2	1	10246.0	3.0	0.0
3	1	49683.0	4.0	0.0
4	1	43633.0	5.0	1.0

Displays the first few rows of the order_products_prior dataframe.

	order_id	product_id	add_to_cart_order	reordered
0	2	33120	1.0	1.0
1	2	28985	2.0	1.0
2	2	9327	3.0	0.0
3	2	45918	4.0	1.0
4	2	30035	5.0	0.0

Displays the first few rows of the orders dataframe.

	order_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	\
0	2539329	1	prior	1	2	8	
1	2398795	1	prior	2	3	7	
2	473747	1	prior	3	3	12	
3	2254736	1	prior	4	4	7	
4	431534	1	prior	5	4	15	

	days_since_prior_order
0	NaN
1	15.0
2	21.0
3	29.0
4	28.0

2. Preprocessing

Merge Datasets

combining two or more datasets into a single dataset. This process is often used to bring together related information that is stored separately.

fundamental part of data preprocessing in data analysis and machine learning workflows.

1. **Combining Datasets**

combines the order_products__train and order_products_prior datasets into a single dataset called order_products_all.

2. **Merging with Product Details:**

merges order_products_all with the products dataset to add the product_name based on the product_id. ensures that all rows from order_products_all are kept, even if there isn't a matching product_id in products.

3. **Merging with Order Details:**

merges order_products_all with the orders dataset to add the user_id based on order_id.

These steps help consolidate product and order information into a single dataset, allowing easier analysis, such as identifying patterns in purchasing behavior across users and products.

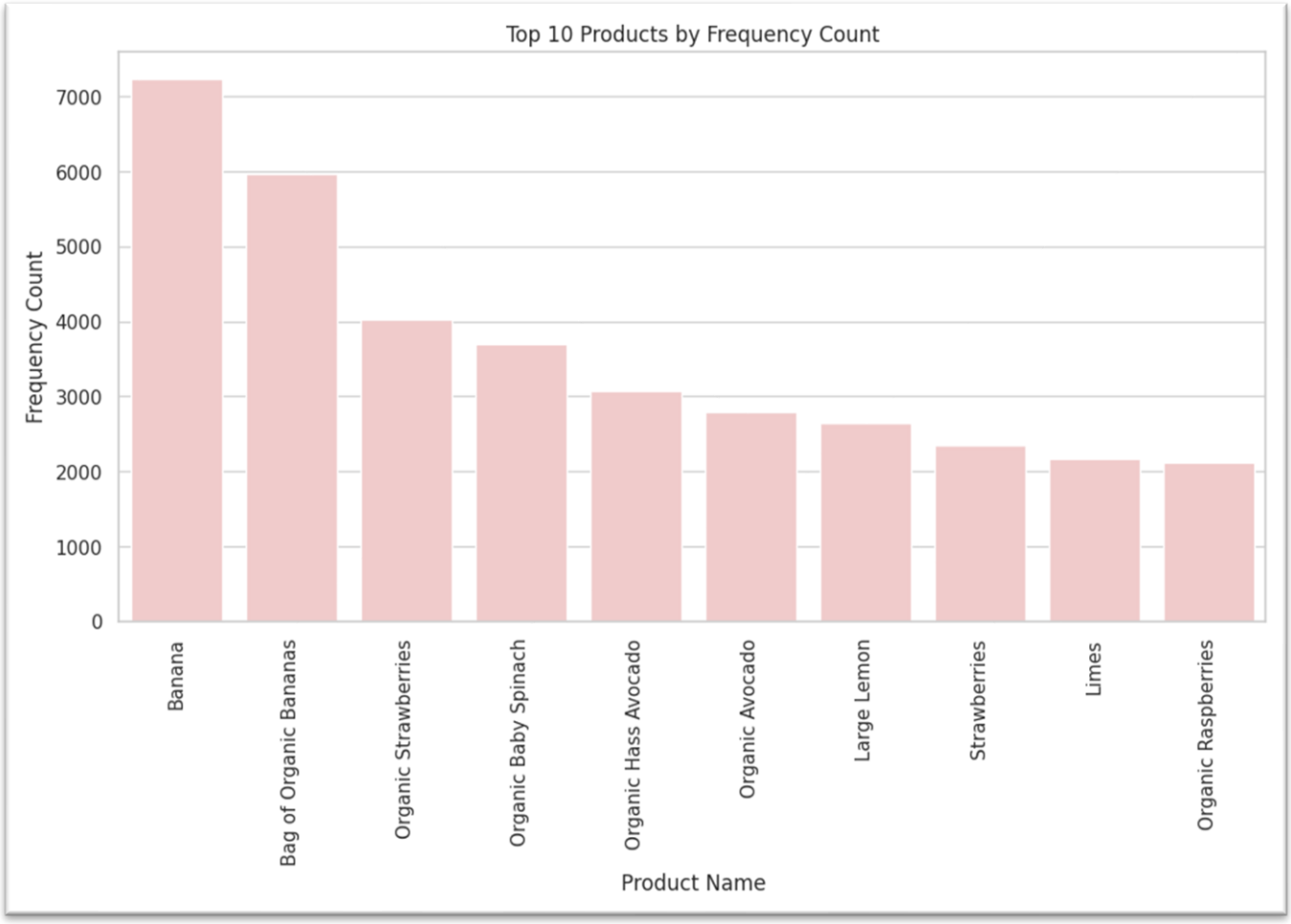
- **Frequency count of each unique value in the 'product_name' column**

product_name	
Banana	7240
Bag of Organic Bananas	5968
Organic Strawberries	4034
Organic Baby Spinach	3700
Organic Hass Avocado	3066
Organic Avocado	2791
Large Lemon	2648
Strawberries	2357
Limes	2171
Organic Raspberries	2124

- This table shows a list of the top 10 products and their corresponding counts representing the most purchased or most frequent items

- fresh produce, especially fruits like bananas and berries, are among the most popular items.

visualize the frequency of the top 10 products



- **Convert Data into Transactional Format**

This operation creates a "basket" view of each order, showing all products in that order as a list.

- Groups the order_products_all dataframe by order_id.
- Converts the product_name values for each order_id into a list.

	order_id	product_name
0	1	[Bulgarian Yogurt, Organic 4% Milk Fat Whole M...
1	2	[Organic Egg Whites, Michigan Organic Kale, Ga...
2	3	[Total 2% with Strawberry Lowfat Greek Straine...
3	4	[Plain Pre-Sliced Bagels, Honey/Lemon Cough Dr...
4	5	[Bag of Organic Bananas, Just Crisp, Parmesan,...

- **Convert the Transaction Data into a One-Hot Encoded Format**

Represents the presence of products in each transaction with binary values (1,0)

```
usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWa
 and should_run_async(code)
  #2 Coffee Filters  #2 Cone White Coffee Filters  #2 Mechanical Pencils  \
              False                         False                  False
              False                         False                  False
              False                         False                  False
              False                         False                  False
              False                         False                  False

  #4 Natural Brown Coffee Filters  #NAME?  \
                            False   False
                            False   False
                            False   False
                            False   False
                            False   False

  & Go! Hazelnut Spread + Pretzel Sticks  \
                                   False
                                   False
                                   False
                                   False
                                   False
```

3. Implement Basket Analysis Algorithms

the Apriori Algorithm

The Apriori algorithm is a popular method used in data mining for finding frequent item sets and generating association rules.

```
usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `
 and should_run_async(code)
   support                    itemsets
 0.010864    (100% Raw Coconut Water)
 0.018078    (100% Whole Wheat Bread)
 0.011654       (2% Reduced Fat Milk)
 0.021909  (Apple Honeycrisp Organic)
 0.024811                (Asparagus)
```

Support: this indicates the proportion of transactions that include the item. It's a measure of how frequently the item appears in the dataset. products that are commonly bought by customers.

Item	Percentage of Transactions
100% Raw Coconut Water	10.864%
100% Whole Wheat Bread	13.078%
2% Reduced Fat Milk	11.654%
Apple Honeycrisp Organic	21.909%
Asparagus	24.811%

Generate Association Rules

Association rules are used in data mining to discover interesting relationships between variables

- Confidence: Measures how often the consequent is purchased when the antecedent is purchased.
- Lift: Evaluates the strength of a rule by comparing the observed support to that expected

Lift > 1: Positive association; items are more likely to be bought together.

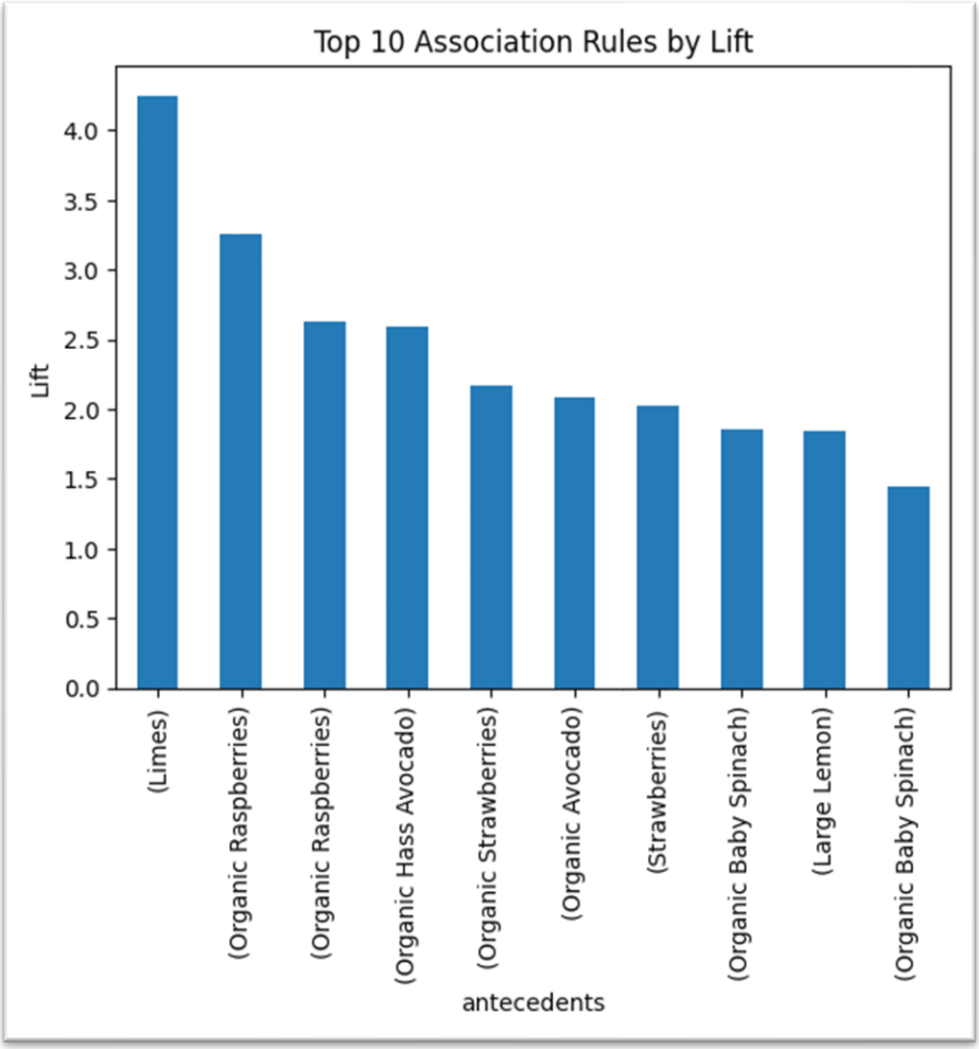
Lift = 1: No association; items are independent.

Lift < 1: Negative association; items are less likely to be bought together.

```
           antecedents              consequents   support  confidence  \
  (Organic Baby Spinach)  (Bag of Organic Bananas)  0.016418    0.218568
  (Organic Hass Avocado)  (Bag of Organic Bananas)  0.018599    0.306739
   (Organic Raspberries)  (Bag of Organic Bananas)  0.013147    0.310557
  (Organic Strawberries)  (Bag of Organic Bananas)  0.021049    0.255713
          (Large Lemon)                 (Banana)  0.014552    0.267515

      lift
  1.850631
  2.597180
  2.629505
  2.165136
  1.841087
usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_cell`
 and should_run_async(code)
```

Visualize the Results



Conclusion: All item pairs have a lift greater than 1, indicating a strong positive relationship

Organic Baby Spinach → Bag of Organic Bananas

Organic Hass Avocado → Bag of Organic Bananas

Organic Raspberries → Bag of Organic Bananas

Organic Strawberries → Bag of Organic Bananas

Large Lemon → Banana