# Final Project Part 4

### Aaron Vo

### BIA 678

### July 29, 2024

# Calling the Apriori Algorithm

|  |  |  |
| --- | --- | --- |
|  | support | itemsets |
| 48 | 0.021864 | (Banana) |
| 45 | 0.010357 | (Bag of Organic Bananas) |
| 422 | 0.008055 | (Organic Hass Avocado) |
| 470 | 0.006904 | (Organic Raspberries) |
| 39 | 0.005754 | (Asparagus) |
| 529 | 0.005754 | (Organic Zucchini) |
| 623 | 0.005754 | (Russet Potato) |
| 476 | 0.004603 | (Organic Red Onion) |
| 502 | 0.004603 | (Organic Strawberries) |
| 494 | 0.004603 | (Organic Sour Cream) |

Association rule mining is used to identify relationships between the Instacart products and items in a dataset. The key metrics used are Support, Confidence, and Lift.

* **Support**: The frequency with which an itemset appears in the dataset.
* **Confidence**: How often items in the right-hand side of the rule appear when items on the left-hand side are present.
* **Lift**: The ratio of the observed support to the expected support if the items were independent.

Given the data, one can gather some key insights regarding the initial support values provided.

* **Banana (0.021864)**: Bananas have the highest support value, meaning they are present in about 2.18% of the transactions in your dataset. This indicates that bananas are a frequently purchased item and could be a strong candidate for product recommendations.
* **Bag of Organic Bananas (0.010357)**: This item has a support of 1.04%, suggesting that it is also commonly bought but not as frequently as regular bananas.
* **Organic Hass Avocado (0.008055)**: This product has a support of 0.81%, indicating that it is a popular organic item but less common than bananas.
* **Organic Raspberries (0.006904)**: With a support of 0.69%, organic raspberries are bought less frequently than the previous items but still have a notable presence.
* **Asparagus (0.005754)** and **Organic Zucchini (0.005754)**: Both have the same support value of 0.57%, indicating that they are equally common in the dataset.
* **Russet Potato (0.005754)**: Similar in frequency to asparagus and organic zucchini.
* **Organic Red Onion (0.004603)**, **Organic Strawberries (0.004603)**, and **Organic Sour Cream (0.004603)**: These items have the lowest support values among those listed, at 0.46%, meaning they are less frequently purchased but still relevant in the dataset.

Overall, this means that the items like Bananas and Bag of Organic Bananas, with high support, are frequently bought and could be recommended to users who purchase similar products. Alternatively. items with lower support, such as Organic Red Onion and Organic Sour Cream, might be recommended to users who have a history of purchasing organic or less common items. If we calculate *confidence* and *lift* for these *itemsets*, we might find strong associations, such as people who buy Organic Hass Avocado might also frequently purchase Organic Raspberries. These associations could be used to suggest complementary products.

# Association Rules

## Confidence

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | antecedents | consequents | antecedent support | consequent support | support | confidence | lift | leverage | conviction |
| 0 | (Large Brown Eggs, Honey Wheat Bread) | (Hamburger Buns) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 1 | (Large Brown Eggs, Hamburger Buns) | (Honey Wheat Bread) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 2 | (Honey Wheat Bread, Hamburger Buns) | (Large Brown Eggs) | 0.001151 | 0.002301 | 0.001151 | 1.0 | 434.5 | 0.001148 | inf |
| 3 | (Large Brown Eggs) | (Honey Wheat Bread, Hamburger Buns) | 0.002301 | 0.001151 | 0.001151 | 0.5 | 434.5 | 0.001148 | 1.997699 |

Let’s take a look at the association rule where it is measured by confidence with min threshold of 0.5. Firstly, the items Large Brown Eggs, Honey Wheat Bread are antecedent of the Hamburger Buns. These items have a confidence of 1.0, indicating that 100% of the transactions containing both Large Brown Eggs and Honey Wheat Bread also contain Hamburger Buns. It’s also had a lift of 869.0, showing that the presence of Large Brown Eggs and Honey Wheat Bread increases the likelihood of purchasing Hamburger Buns by 869 times. The rules suggest that there is a very strong association between Large Brown Eggs, Honey Wheat Bread.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | antecedents | consequents | antecedent support | consequent support | support | Confidence | lift | leverage | conviction |
| 6 | (Organic Light Agave Nectar, Sharp Cheddar Che... | (Newman O's Creme Filled Mint Chocolate Cookies) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 7 | (Organic Light Agave Nectar, Newman O's Creme ... | (Sharp Cheddar Cheese) | 0.001151 | 0.002301 | 0.001151 | 1.0 | 434.5 | 0.001148 | Inf |
| 8 | (Newman O's Creme Filled Mint Chocolate Cookie... | (Organic Light Agave Nectar) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | Inf |
| 9 | (Organic Light Agave Nectar) | (Newman O's Creme Filled Mint Chocolate Cookie... | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 10 | (Sharp Cheddar Cheese) | (Organic Light Agave Nectar, Newman O's Creme ... | 0.002301 | 0.001151 | 0.001151 | 0.5 | 434.5 | 0.001148 | 1.997699 |

Further down the list, some more interesting items appear in the confidence rulesets. Items light Organic Light Agave Nectar have a strange connect with Cheddar Cheese and Cream Filled Mint Chocolate cookies. This could be useful in the recommendation tool if Instacart decides to use said data for their basis of their model.

## Lift

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | antecedents | | consequents | antecedent support | consequent support | support | confidence | lift | leverage | conviction |
| 0 | | (Large Brown Eggs, Honey Wheat Bread) | (Hamburger Buns) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 1 | | (Large Brown Eggs, Hamburger Buns) | (Honey Wheat Bread) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 2 | | (Honey Wheat Bread, Hamburger Buns) | (Large Brown Eggs) | 0.001151 | 0.002301 | 0.001151 | 1.0 | 434.5 | 0.001148 | inf |
| 3 | | (Large Brown Eggs) | (Honey Wheat Bread, Hamburger Buns) | 0.002301 | 0.001151 | 0.001151 | 0.5 | 434.5 | 0.001148 | 1.997699 |
| 4 | | (Honey Wheat Bread) | (Large Brown Eggs, Hamburger Buns) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 5 | | (Hamburger Buns) | (Large Brown Eggs, Honey Wheat Bread) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 6 | | (Organic Light Agave Nectar, Sharp Cheddar Che... | (Newman O's Creme Filled Mint Chocolate Cookies) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |
| 7 | | (Organic Light Agave Nectar, Newman O's Creme ... | (Sharp Cheddar Cheese) | 0.001151 | 0.002301 | 0.001151 | 1.0 | 434.5 | 0.001148 | inf |
| 8 | | (Newman O's Creme Filled Mint Chocolate Cookie... | (Organic Light Agave Nectar) | 0.001151 | 0.001151 | 0.001151 | 1.0 | 869.0 | 0.001149 | inf |

Like the confidence graph, there is a strong association or high lift values across these rules indicate very strong associations between these specific products. For example, customers who buy Large Brown Eggs and Honey Wheat Bread are highly likely to also purchase Hamburger Buns, and vice versa. Instacart can use this information to build potential Bundling Opportunities where retailers could consider bundling these items together as a promotional strategy to capitalize on these strong associations. Furthermore, Instacart can use this as targeted recommendations to apply these rules. This could be used to make targeted product recommendations in an online shopping platform, suggesting related items that customers are statistically likely to buy together.

# Recommendation Model Visualization

Using the association rule mining results that was found earlier, one can use it to identify products with the highest lift values for the items currently in the cart and sort/prioritize recommendations based on lift values.

For the back end, Instacart can use Python and leverage AWS to deploy their model and datasets to integrate with the front-end system to fetch real-time product data and recommendation results. This ensure the system updates recommendations dynamically as items are added or removed from the cart.

For the front-end, the developers can use up-to-date HTML/CSS or even JavaScript frameworks like React and Angular for layout and design. This could be useful in handling any API calls that hosted on AWS to call the model and perform recommendations data. Finally, Instacart can ensure recommendations are relevant and have testers and customers try the system to see if they enjoy the experience or not. How this might look like should be a feature that is implemented on the checkout page, for when customers are about to complete their orders. A simple “You Might Also Like” section is prominently displayed below the cart summary and includes a visually appealing with clear product images and pricing. Users should be able to add recommendation products to their cart. This interface design aims to enhance the shopping experience by making relevant product suggestions based on data-driven insights, thereby increasing the likelihood of additional purchases.

# Content-Based Recommendations

def recommend\_products(user\_id, similarity\_matrix\_df, user\_item\_matrix, top\_n):

similar\_users = similarity\_matrix\_df[user\_id].sort\_values(ascending=False).index[1:]

user\_purchases = set(user\_item\_matrix.columns[user\_item\_matrix.loc[user\_id] > 0])

recommendations = []

for similar\_user in similar\_users:

similar\_user\_purchases = set(user\_item\_matrix.columns[user\_item\_matrix.loc[similar\_user] > 0])

recommended\_products = similar\_user\_purchases - user\_purchases

recommendations.extend(list(recommended\_products))

if len(recommendations) >= top\_n:

break

return recommendations[:top\_n]

This code snippet describes a recommendation system based on user similarity and collaborative filtering. First, we compute a similarity matrix where we find the similarity between users based on their interactions with items. We use a cosine\_similarity to calculate a matrix where each element represents the cosine similarity between two users, indicating how similar their item preferences are.

The following parameters are:

* user\_id: ID of the user for whom recommendations are to be generated.
* similarity\_matrix\_df: DataFrame containing the similarity scores between users.
* user\_item\_matrix: Matrix where rows represent users, columns represent items, and values indicate interactions (e.g., ratings or purchase frequency).
* top\_n: Number of top recommendations to return.

The function essentially identifies users similar to the target user, then suggests items that those similar users have interacted with but the target user hasn’t. This leverages the assumption that similar users have similar products that they bought. An example products looks like the following:

Top 10 Product Recommendations for the user 202279:

-------------

0 Pure Sparkling Water

1 Half & Half

2 Freeze Dried Strawberry Slices

3 Double Chocolate Cake

4 Tiny Fruits Blueberry Apple

5 Organic Freeze Dried Strawberries

6 Organic Freeze-Dried Mango

7 Berry Medley

8 Organic Garlic

9 Organic Small Bunch Celery