Purpose of recommendation system

# Role of recommendation system in online retail

Of all the industries in the world today, the online retail sector is considered to be among the fastest developing. This is rather so because Globally it is growing at different paces, but it is certainly ingraining itself in our everyday life. It is very likely that Online Retail will keep this place in the coming days.

Rhodes C., and Brien P.,(2015) have drawn attention to the development of online retail in the UK. Sales increased from a level of 2.7% in the year 2007 to approximately 11.1% in the year 2013. Rao P., et al. (2021) also predicted that from the year 2020 to 2027, the growth rate each year would be 14% since in the year 2019, the gross market was $9 trillion.

There are several major factors which contribute this rapid growth, such as cost savings, better product selection, and ease of use.

As it was summarised by Fayyaz Z., et al., (2020), recommendation systems reduce significant and improve customer experiences. They make product suggestions based on user activity, purchase history and features of products. This personalisation helps in creating additional interest among the clients and ensuring their loyalty. It is a mutually beneficial strategy in which businesses acquire dedicated clients while consumers enjoy individualised services.

Apart from personalisation, there are other advantages of recommendation systems:

• **Stock Management:** Companies can sell goods which are related to consumer interests or are low on stock.

* **Upselling:** Businesses can sell more by recommending other related or higher value items.
* **Expansion of Customer Segments:** Users can be integrated to a larger targeted section with precision depending on their needs.

Considering mentioned advantages, we can say that recommendation systems are indispensable in ecommerce as they assist in making purchase decisions as well as boosting income.

## Role of macine learning in recommendation systems

Modern recommendation engines owe a lot of their capabilities to machine learning. It helps in understanding and studying enormous volumes of user’s behavior, profiles, purchase patterns, ratings, and even text describing the items, so the demands of the consumers can be catered to individually. Due to this, business can get more clients, customize their service for various audiences, and respond to the fast changing and competitive market.

Among the advantages of machine learning, one stands out – the ability to read both explicit and implicit feedback:

• **Explicit feedback:** This is data given in the form of product rating of the product such as reviews, left by users on purpose.

• **Implicit feedback:** This is data left by clients while they are engaging with a company for instance in their buying patterns, searches on the platform, or even clicks.

Collaborative and content-based filtering as machine learning models utilize the above data to understand relations among users or products, thus, creating the databases used in giving recommendations.

When potential consumers have numerous options available, the actual business profit can only be gained by connecting the systems and customising them according to business objectives. In support, Liu (2022) argues that creating such systems in practice involves testing different combinations and models in order to satisfy target customers.

## Approaches to build a recommendation system

### Content Based

Users of these types of recommendation systems are targeted based on their recent activities which are determined by the products they usually buy or search for.

Javed et al. (2021) make a thorough distinction between supervised and unsupervised approaches to recommendation systems based on the type of available data, taking into account NLP techniques as well.

For designing a content-based recommendation system, we employ two basic methods:

#### TF-IDF transformation

This technique evaluates words of a description by taking into account relative importance for frequency while disregarding irrelevant words. It facilitates the comparison of product descriptions by locating key similarities making it most useful for more describable products such as books or movies.

In our assessment we are going to use a **TfidfVectorizer** package provided by **sklearn** library.

#### Cosine Similarity

The nature of two items is defined with respect to their image by calculating the cosine angle of their feature vectors, measured from 0, which means no similarity to 1 that implies full similarity. We will utilise cosine\_similarity from the sklearn library for these calculations.

#### Collabarative Filtering

Collaborative filtering is a popular recommendation strategy where the preferences of a customer are inferred from those with similar patterns of behavior. In this case, a cosine similarity matrix of the users’ activity is constructed and a K-Nearest Neighbors (KNN) model is fitted to suggest the most similar users in high-dimensional space.

This was the same model that Cui, B.B., (2017) used in his work in which he builds a recommendation system based on a movie dataset.

# Implementation

## Data overviewImage

----- Unique values ------  
Unique customers: 1468  
Unique products IDs: 1145  
Unique products descriptions: 404  
Unique locations: 5  
Unique transactions: 25061  
DF shape: (52955, 21)

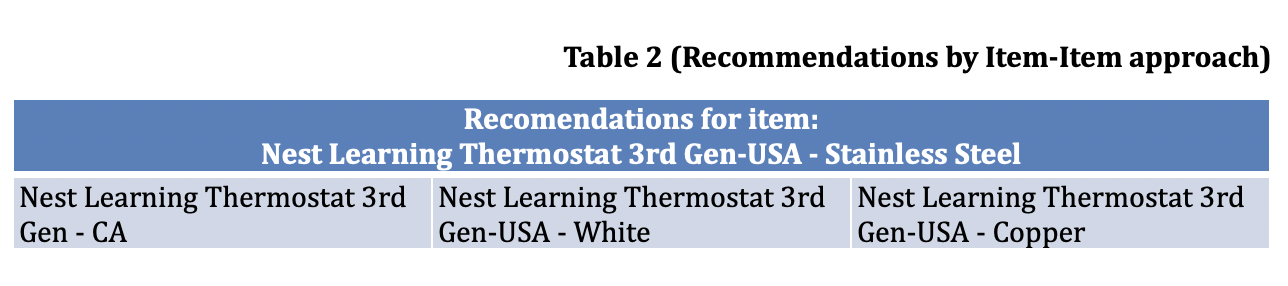
As we can see, the dataset provides a number of attractive components for building a recommendation system. Depending on the chosen approach, specific features will be employed. For example, with a content-based approach, we only have to avail ourselves of data concerning the item in question which includes the item description. While in a user-user approach, it is the buying habits of their customers that must be the main focus.

The dataset includes 1,468 unique customers, suggesting that some customers made multiple transactions or included multiple items in a single transaction.

## Content-Based Recomendation System

#### Based on the product discription

In this section, we are going to design a recommendation model based on the product descriptions by performing the following tasks:

1. Read the file and select only the relevant features. Drop any duplicates and missing values.
2. Use the TF-IDF Vectorizer to represent the descriptive feature by represensting the columnar slots of its distinct words.
3. Using a cosine similarity model, a matrix is then created in which each SKU is matched against all other SKUs based on the number of words that are included in their descriptions.
4. The last step is to assign each row of the DataFrame to the respective product description.
5. The last function is such that it accepts a product and issues recommendataions related to that product.

#### Based on the product popularity

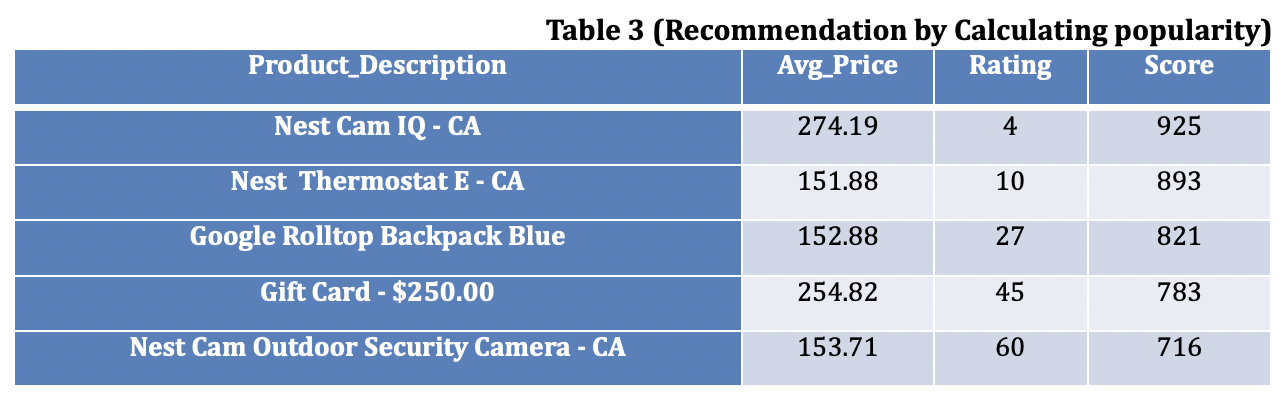
The dataset we chose for this CA contains a rating information which is missing for the content filtering based on ratings. Nevertheless, we can create this data by assuming that purchase count accurately correlates with satisfaction. What this means is that if a customer buys more than one item, we take it as rude to assume that the customer does not like the item. The intention is to develop new features that show men’s views regarding the products they bought before.

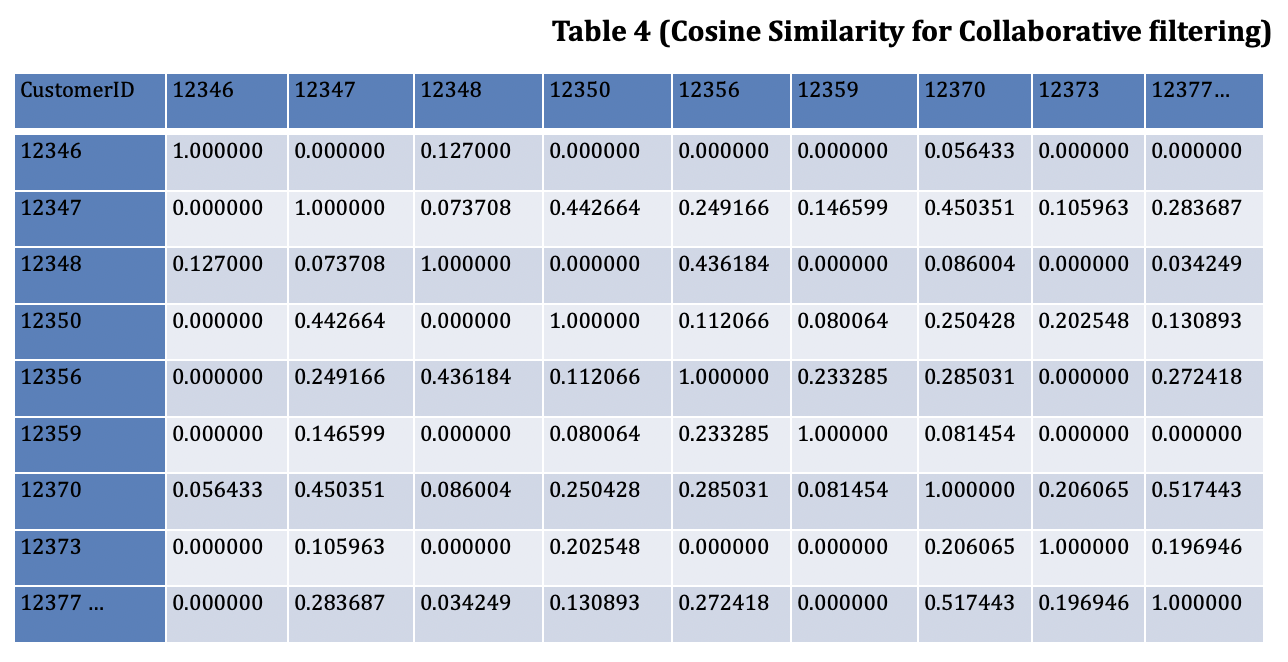
In this stage, we plan for using a procedure analogous to IMDB’s weight average rating calculation to assess representative figures. However, since our data is quite different, it is imperative to modify this procedure in order to create a simulation of the ratings.

As in the previous approach, transformations performed on the data will include the following:

1. Create a new feature, "Rating," based on the quantity of each specific item purchased by a customer.
2. Calculate static properties for the scoring equation.
3. Define a function to apply the equation.
4. Apply the function to the data and store results in a new column, "Score."

The score value reflects product popularity.

This actually indicates the popularity of that product. Ratings will be based on the sales volume of an item. The assumption is, 1 purchase equals 1 vote.

**C****ollaborative filtering**

# Modelling

Recommendations for user 15838.0: ['Nest Cam IQ - USA’]

# Interpretation

We have tested three approaches for recommendation systems using collaborative filtering:

#### Item-Item Filtering Based on Product Description

Best suited for online shops that deal with many goods of similar use. It can also propose some newer or expensive alternatives.

**Key Consideration:** Product descriptions must be concise and well-written, covering essential attributes

**Benefits:** There is no requirement for having a history of customer participation or activity which makes it a good fit for such early stage businesses.

#### Recommendations Based on Product Popularity or Ratings

This approach ranks products according to customer ratings or customer satisfaction scores, which takes into account both average scores and count of reviews. Enterprises can also use more demographic information in targeting such recommendations.

**Benefits:** Provides instant recommendations without requiring user activity.

**Drawbacks:** There is a dependence on the aggregate opinions of other users and the requirement of sufficient dataset coverage to be able to provide accurate estimates.

#### User-User Collaborative Filtering

Is built around user behavior or purchases the key selling point for businesses with a wide range of products and built-in databases of customers. It matches users based on like activities rather than descriptions of the products in question.

**Advantages:** This is incredibly precise and not dependent on the descriptions of the products.

**Disadvantages:** It requires lots of information and may be ineffective for users who are new or have not been active for some time.

#### Justification

The appropriate approach in this case to consider for our dataset is User-User Collaborative Filtering due to the following factors:

• **Sufficient Customer Activity Patterns:** The market basket dataset contains adequate purchase convergence that enables cross-recommendation of customers with identical concerns. More patterns deepen the strength of these bonds.

* **Limited Product Descriptions:** Item descriptions are given but not to the level that sufficiently allow for item item filtering or content approaches hence user user approach is preferred since no metadata is sought for.
* **Ability to Recommend wide range of products:** User-user recommendation can suggest items of assorted kinds owing to the overlaps among users thereby broadening the scope of offerings and recommending several unanticipated by customers.

• **Introducing new products:** Recommendations are based on customers’ trends rather than products’ characteristics and therefore it is possible to recommend unlaunched products without the need to possess a lot of metadata or purchase database unlike item- item filtering.

# Market Basket Analysis (MBA)

## Role of MBA in online retail

The Market Basket Analysis (MBA) is a very good technique of ascertaining the buying behavior of customers and assisting them in making decisions. Implicit in conceptual frameworks such as Collaborative Filtering or content based approaches, as the identification of suitable clients and products, hampered in MBA is the analysis of routines of purchasing items that are connected as well as the interrelations between those items, without constantly referring to exactly particular customers or products.

As highlighted in a study by Zamil, A. (2020), the vast selection in online retail often overwhelms customers. MBA helps online retailers provide clear and accurate recommendations, enhancing customer experience and loyalty.

MBA assumes certain items are linked to common customer needs. For instance, if many customers purchase a drill and safety glasses together, the algorithm infers a connection and recommends safety glasses to customers buying a drill.

The MBA approach relies on several key metrics worth noting.

### Support

Support describes probability of each item or combination of item appears in the data set.

### Confidence

Confidence is another probability metric which discribes the likelihood of item X appears within set Y.

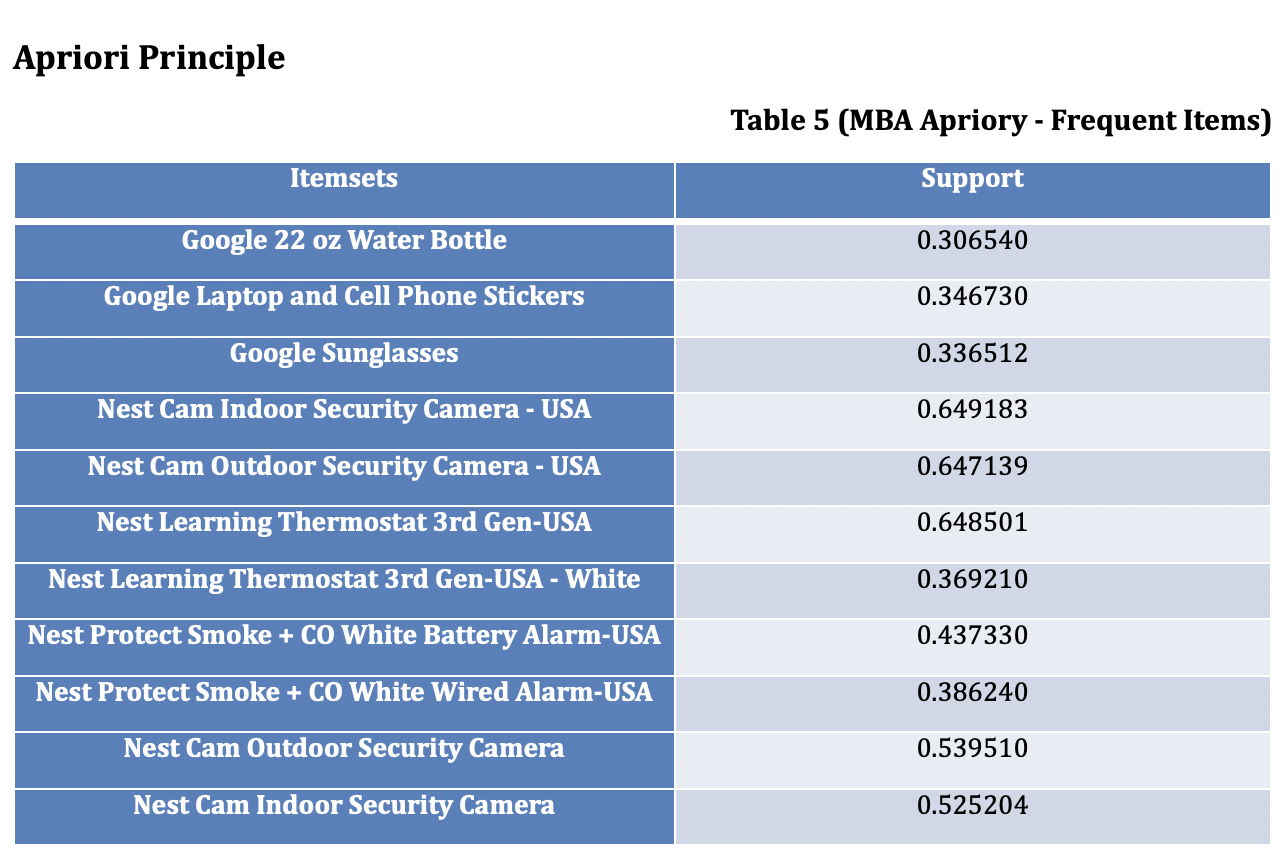
### Lift

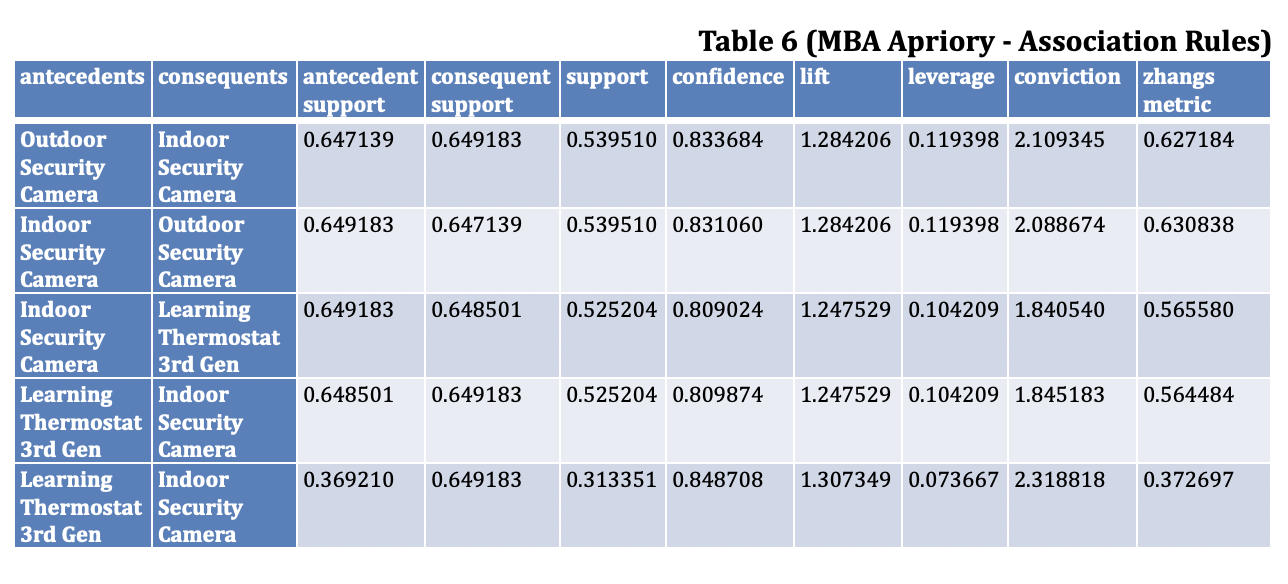
Lift measures the association of two items or sets of items and how likely they can be purchased together.

### Leverage

Leverage is another mertic for measuring association of two items or sets. But in this case they are considered as independent items which do not belong to the same set.

### Conviction

Conviction measures the degree of implication or dependence between items in an association rule. It reflects the strength of the rule.



### Explanation of recommendation

**Recommendations for customer with basket:** Nest Cam Outdoor Security Camera:

1. Nest Cam Indoor Security Camera

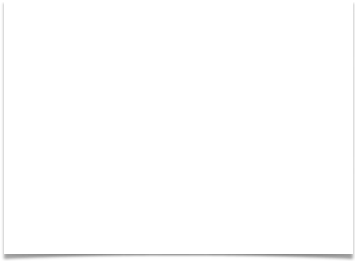
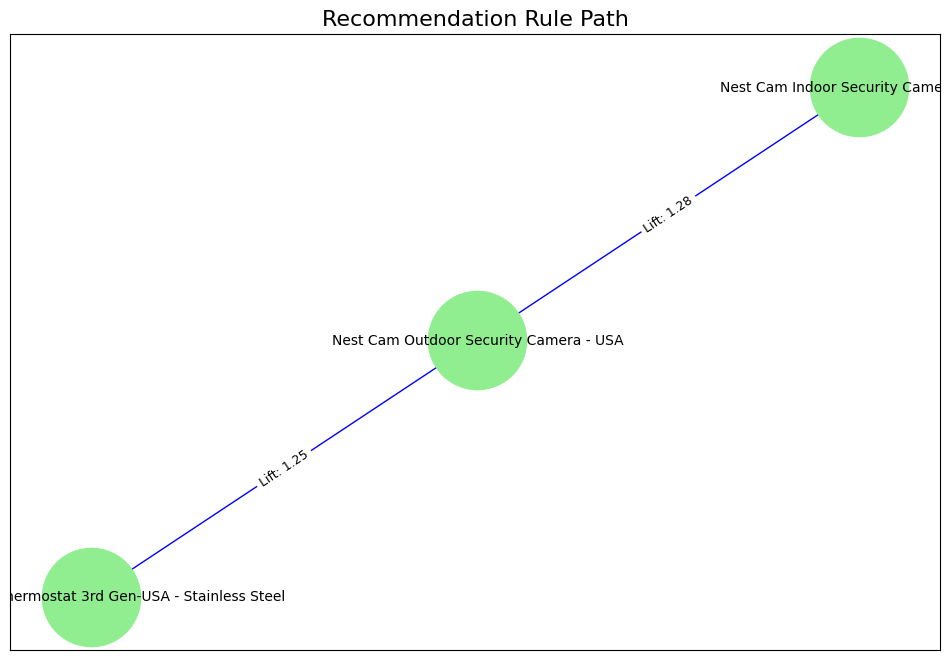
• **Explanation**: Because you bought Nest Cam Outdoor Security Camera, we recommend Nest Cam Indoor Security Camera (Support: 0.54, Confidence: 0.83, Lift: 1.28)

2. Nest Learning Thermostat 3rd Gen-USA - Stainless Steel

• **Explanation:** Because you bought Nest Cam Outdoor Security Camera, we recommend Nest Learning Thermostat 3rd Gen. (Support: 0.52, Confidence: 0.81, Lift: 1.25)

# Visualisation

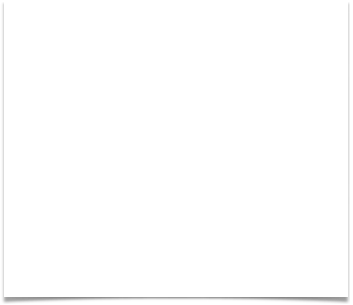
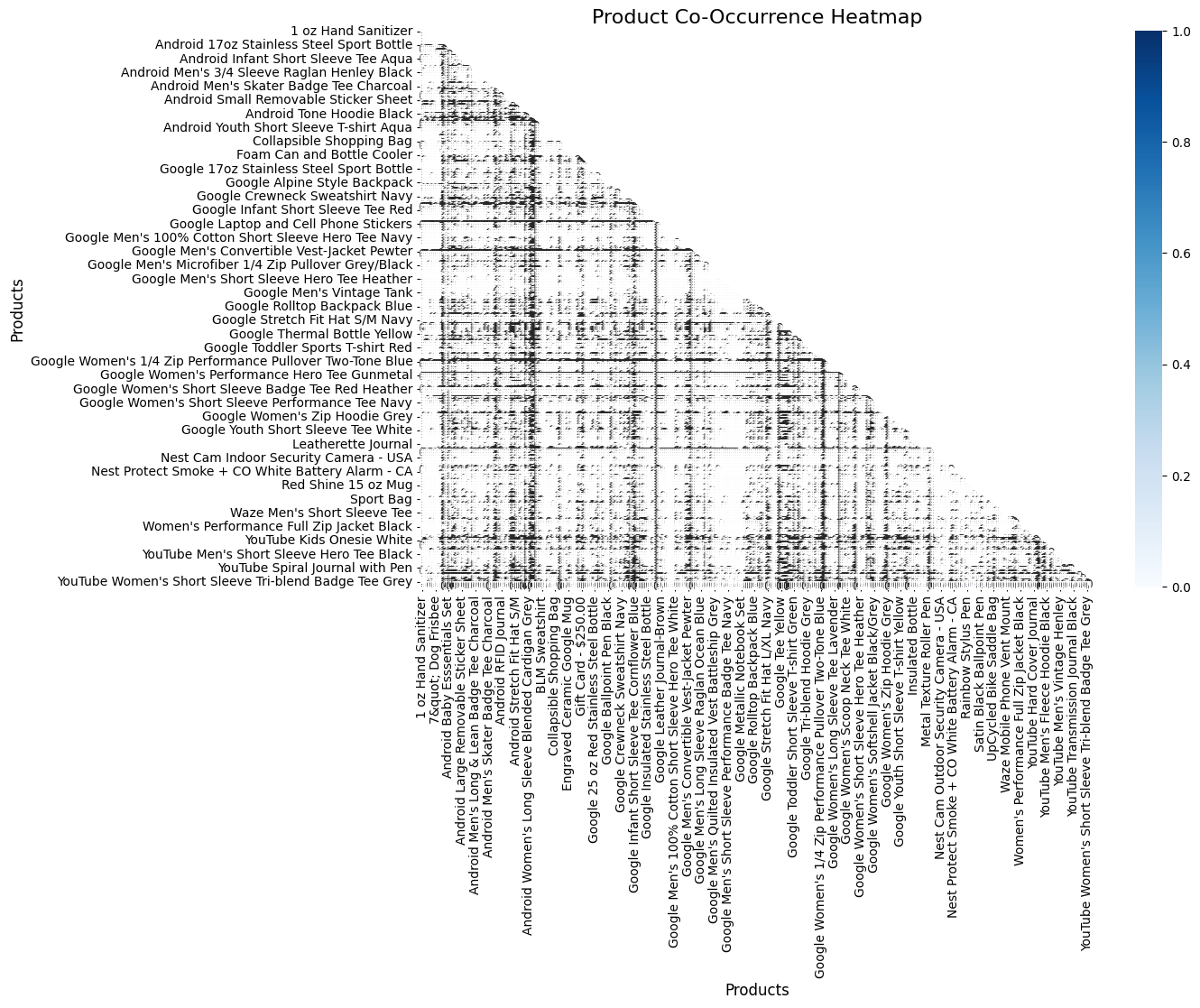
##### Figure 1 (Recommendation justification by displaying network plot)



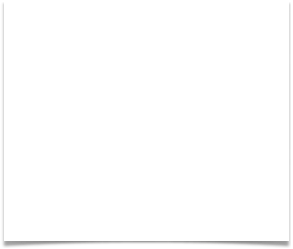
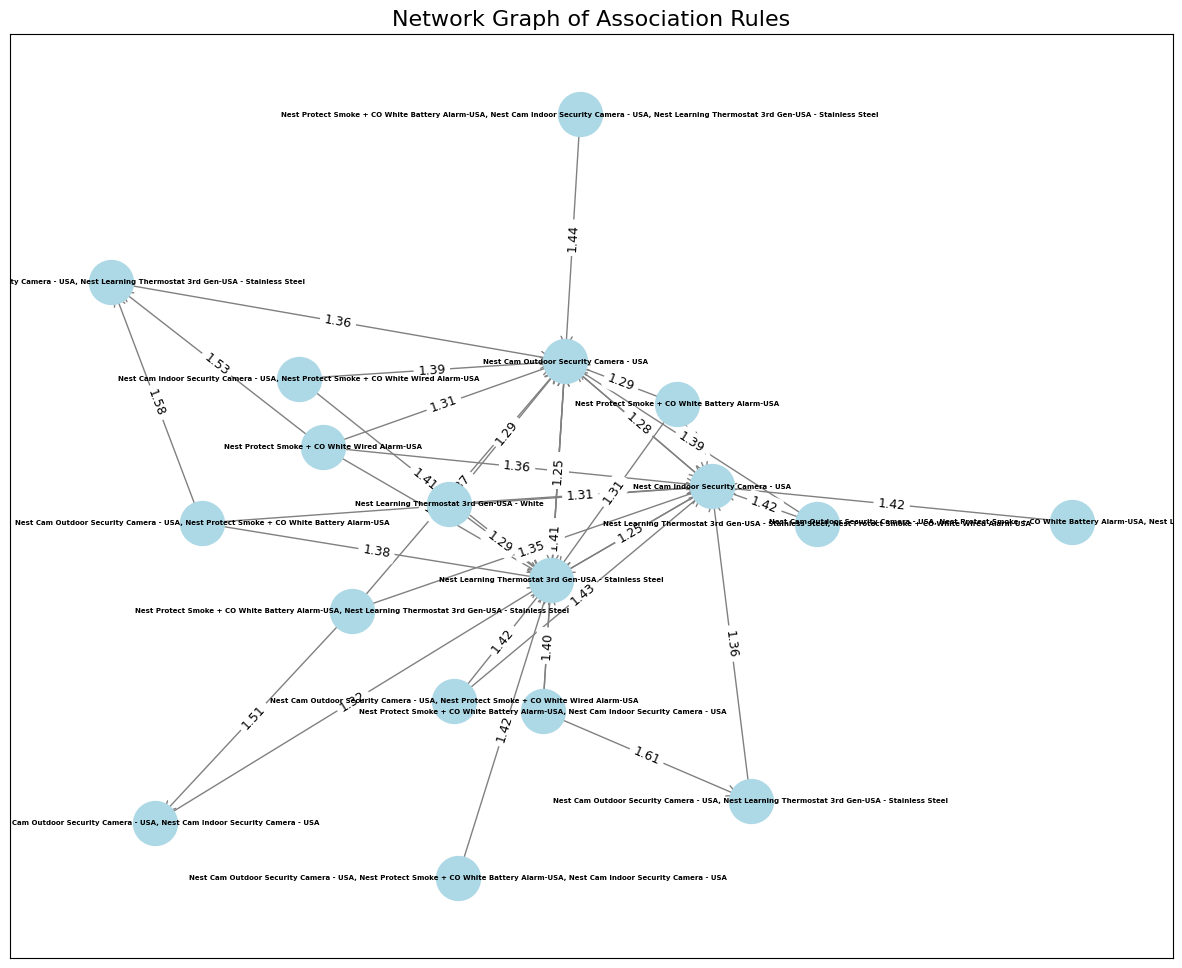
##### Figure 2 (Top 10 Rules based on Lift mertic)



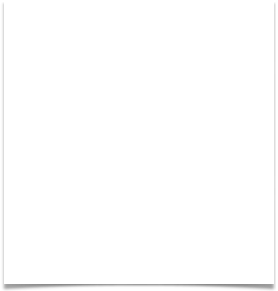
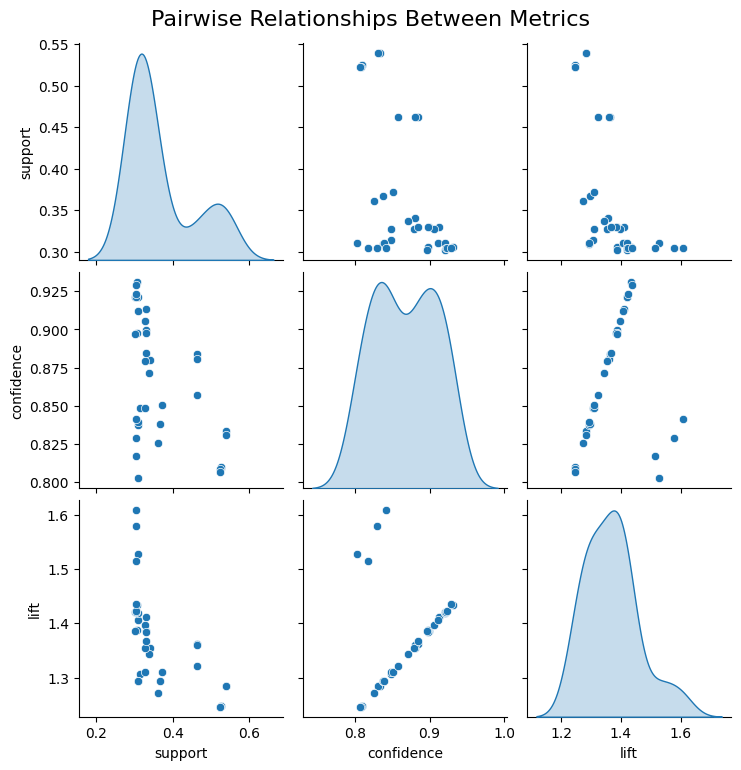
##### Figure 3 (Co-Occurrence Heatmap)



##### Figure 4 (Network Graph of Association Rules)



##### Figure 5 (Correlation between metrics)



## FP-Growth

--------------------  
Input  
['Nest Cam Indoor Security Camera - USA']  
--------------------  
  
Recommendations for customer with basket ['Nest Cam Indoor Security Camera - USA']: ['Nest Cam Outdoor Security Camera - USA', 'Nest Learning Thermostat 3rd Gen-USA - Stainless Steel']

# Data Visualisation

Dashboard is deployed using Heroku service and can be accessed by link:  
<https://sales-dashboard-ca2-1df0b74784d9.herokuapp.com/>

## Data Preparation Explanation

We decided to implement 2 recommendation models:

• MBA to provide a recommendation based on selected products.

• Collaborative User-User approach to provide a recommendation based on the activity of other customers.

Additionally, we implemented the SARIMA model to provide users with a time series prediction of sales for the selected time period.

### Key preparation stages:

**Dataset for general**

1. Feature selection - keep only relevant for dashboard features.
2. Create a new feature with state codes for displaying a map.
3. Time formatting.

**Dataset for User-User Recommendation**

For this dashboard feature, we used models which we developed earlier with slight modifications to meet the format.

1. General preparation included (grouping, dropping duplicates and missing values)
2. Creating a User-Item matrix.
3. Splitting the data on train and test.
4. Defining the KNN model and parameters for searching for the closest customers.
5. Function user\_user:

* • applies trained KNN model to find the most similar customers by purchasing history
* • gets items related to the user
* • excludes items the user has already purchased
* • Return the most relevant items for selected customers.

1. Function plot\_customer\_recommendation\_graph explains why recommended features were selected for certain customers by displaying a network plot.

**Dataset for MBA**

1. After all missing values were dropped, we created a User-Item Matrix.
2. Creating an item set with minimum support of 0.3, using apriory approach.
3. Creating rules based on confidence with a threshold of 0.8.

**Forecast**

To make the dashboard more insightful for the user, we added a forecasting model (SARIMA) for daily sales according to specific filtering. Mostly for this feature, we made a preparation directly in the callback:

1. Using IF statements filter df according to user's requirements.
2. Transform filtered df by the sum of each transaction day.
3. Set Transaction Day as an index.
4. Define the parameters of the SARIMA Model and fit the model. Note: Initially we added AutoARIMA to define optimal parameters we each filtered time series. This solution significantly increased the time of response, due to the amount of calculations needed to be done. For those reasons, we excluded that step.
5. Get the prediction based on the user's request and display on the plot actual data and a forecast.

## Displayed Data Explanation

Based on requirements to create a dashboard for young adults (18-35 y.o.) we decided to display several charts and tables aligned with certain data visualisation techniques:

### Tab 1: Customer information

Aims to give details about certain customers or demographic groups.

#### Sub Tab 1: Specific Customer

Using the Drop-down element or Input Window user can filter df and get all transaction info related to the selected customer displayed as a table. Additionally, there is displayed a bar plot to show customer's purchase distribution grouped by product categories.

#### Sub Tab 2: Demographic

In this subtab, users can explore sales data for certain genders and locations.

As an output user can see raw filtered data with all product categories for selected genders and locations, supported by the bar chart.

### Tab 2: Recommendations

#### Sub Tab 1: By Customer

In this sub-tab, we applied Collaborative User-User Filtering. Users can apply the Drop-down element or Input Window to select a customer and the function will provide the top 3 items the most relevant for this customer based on their purchase history. Additionally, we implemented a network plot to explain why the model made that decision.

#### Sub Tab 2: By Product

Similarly, we implemented a recommendation system based on product purchasing frequency using Market Basket Analysis. Users can select a specific product and get a recommended item which has the highest probability of appearing tougher in the dataset.

### Tab 3: Sales

This Tab is dedicated to sales analysis and forecasting.

On the left side, we placed several elements to filter data.

* + Time period selector: The user can choose the specific time period to output the data.
  + Location Drop-down: Users can choose only those states which are relevant to them.
  + Specific product category.
  + Bar time period selector, if a user wants to get a daily sales forecast for a specific time.

Output includes 3 elements:

1. Geographic sales distribution: The map displays selected states and their total sales based on selected time and product categories.
2. Daily sales: The line chart illustrates daily total sales made and forecast based on selected parameters.
3. Detailed table: If the user wants to dive into raw data they can find a table with all filtered transactions.

### Dark mode bottom

As a response to the requirement to create a dashboard for the target audience of young adults, we implemented the dark mode feature which aims to reduce visual load. This function may be appropriated by users who spend a significant amount of time in front screens and could be overwhelmed with bright themes.

### General Dashboard composition

Overall, we tried to follow a familiar target audience structure, which is commonly used nowadays in many apps. The left sidebar is dedicated to managing outputs on the right side.

## Dashboard DocumentationImage

# Image

# Image

# Image

# References

Cui, B.B., 2017. Design and implementation of movie recommendation system based on Knn collaborative filtering algorithm. In ITM web of conferences (Vol. 12, p. 04008). EDP Sciences.

Debnath, S., 2008. Machine learning based recommendation system. Master's thesis, Department of Computer Science and Engineering, Indian Institute of Technology.

Fayyaz, Z., Ebrahimian, M., Nawara, D., Ibrahim, A. and Kashef, R., 2020. Recommendation systems: Algorithms, challenges, metrics, and business opportunities. Applied sciences, 10(21), p.7748.

Javed, U., Shaukat, K., Hameed, I.A., Iqbal, F., Alam, T.M. and Luo, S., 2021. A review of content-based and context-based recommendation systems. International Journal of Emerging Technologies in Learning (iJET), 16(3), pp.274-306.

Liu, L., 2022. e‐Commerce Personalized Recommendation Based on Machine Learning Technology. Mobile Information Systems, 2022(1), p.1761579.

Rao, P., Balasubramanian, S., Vihari, N., Jabeen, S., Shukla, V. and Chanchaichujit, J., 2021. The e-commerce supply chain and environmental sustainability: An empirical investigation on the online retail sector. Cogent Business & Management, 8(1), p.1938377.

Rhodes, C. and Brien, P., 2015. The retail industry: statistics and policy. House of Commons Library Briefing Paper.

Zamil, A.M.A., Al Adwan, A. and Vasista, T.G., 2020. Enhancing customer loyalty with market basket analysis using innovative methods: a python implementation approach. International Journal of Innovation, Creativity and Change, 14(2), pp.1351-1368.

#### Author: Ilia Grishkin

<https://github.com/CCT-College-Dublin/integrated-ca2-50-Ilia-Grishkin>

### Total words: 2715