KIET GROUP OF INSTITUTIONS

Connecting Life with Learning

www.kiet.edu
Delhi-NCR, Ghaziabad

“Market Basket Analysis: Use association rule mining to classify customer purchasing patterns for targeted marketing strategies.

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY

DEGREE

SESSION 2024-25 in

Name of discipline

CSE-AIML

Navishka Sharma(202401100400123)

Under the supervision of

“Mr. Abhishek Shukla”

KIET Group of Institutions, Ghaziabad

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

🧩 THE PROBLEM: Market Basket Analysis for Targeted Marketing

A retail store (physical or online) wants to understand **customer purchasing behavior** — specifically, **which products are bought together**.

By understanding **purchase patterns**, they can:

- Design **targeted marketing campaigns**
- Create **product bundles or combos**
- Improve **store layout** (e.g., put items near each other)
- Make better **inventory decisions**

🧠 APPROACH USED TO SOLVE THE PROBLEM

We used **Association Rule Mining** with the **Apriori Algorithm** to uncover relationships between items frequently bought together. Here's a step-by-step breakdown of the approach:

- We used a CSV file named **"Market Basket Analysis.csv"**.
- The dataset contains transactions — each row lists products purchased together by a customer.

- Remove missing values (NaN) to clean the data.
- **Converted all items to strings** to ensure consistency.
- **Formatted data as transactions** (list of lists) where each sub-list is a customer's basket.

- Used `TransactionEncoder` to **one-hot encode** the dataset.
- This step transforms it into a **binary format** where:
    - Rows = transactions
    - Columns = items
    - Values = `True` or `False` (whether the item was purchased)

- Applied the **Apriori algorithm** to find frequent combinations of items.
- Set `min_support = 0.005` to only keep itemsets that appear in **at least 0.5% of all transactions**.
- This step gives us combinations of items that occur frequently together.

- Generated **association rules** from frequent itemsets using metrics:
    - **Support**: Frequency of the itemset
    - **Confidence**: Likelihood of buying B if A is bought
    - **Lift**: How much more likely B is bought when A is bought, compared to random chance
- Filtered rules with `lift >= 1.0` for meaningful relationships.

- Analyzed rules like:

If a customer buys `{Milk}`, then they are likely to also buy `{Bread}`.

- Used these rules to make **business decisions**, such as:
    - Product recommendations
    - Combo deals
    - Targeted promotions

Interpretation: 80% of customers who bought Bread also bought Butter, and this relationship is 1.25x stronger than random.

- A list of rules that describe **customer purchasing patterns**.
- These rules can be used for **targeted marketing strategies** and improving customer experience.

CODE :

```python
!pip install mlxtend --quiet

# Step 2: Upload the file from your computer
from google.colab import files
uploaded = files.upload()

# Step 3: Load dataset using the uploaded filename
import pandas as pd
import io

# Automatically detect the filename
filename = list(uploaded.keys())[0]
df = pd.read_csv(io.BytesIO(uploaded[filename]))

# Preview data
df.head()
# Step 4: Preprocess the data
# Step 4: Clean and prepare transactions
transactions = df.dropna().values.tolist()

# Convert all items to strings to avoid type comparison issues
transactions = [[str(item) for item in row if str(item).lower() != 'nan'] for row in transactions]

# Step 5: Encode transactions
from mlxtend.preprocessing import TransactionEncoder

te = TransactionEncoder()
te_ary = te.fit(transactions).transform(transactions)
df_encoded = pd.DataFrame(te_ary, columns=te.columns_)
# Step 6: Find frequent itemsets
from mlxtend.frequent_patterns import apriori, association_rules
```

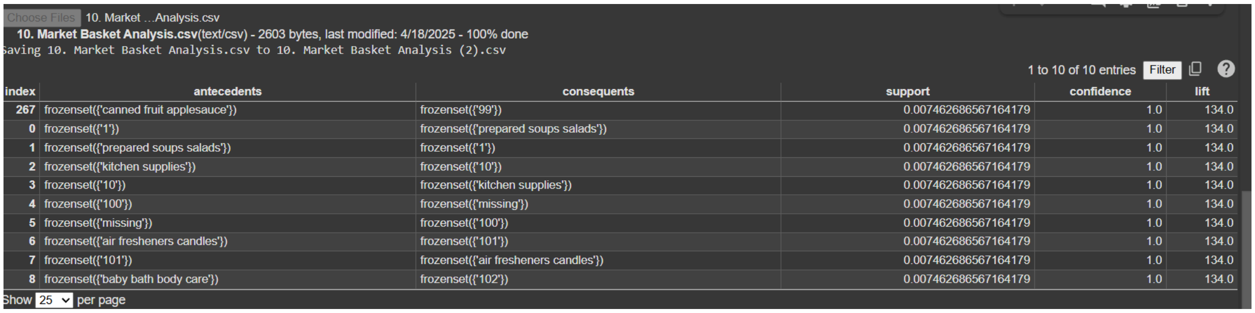
```

frequent_itemsets = apriori(df_encoded, min_support=0.005, use_colnames=True)


# Step 7: Generate rules
rules = association_rules(frequent_itemsets, metric="lift", min_threshold=1.0)

# Step 8: Show top rules
rules[['antecedents', 'consequents', 'support', 'confidence', 'lift']].sort_values(by='lift', ascending=False).head(10)
```

OUTPUT:

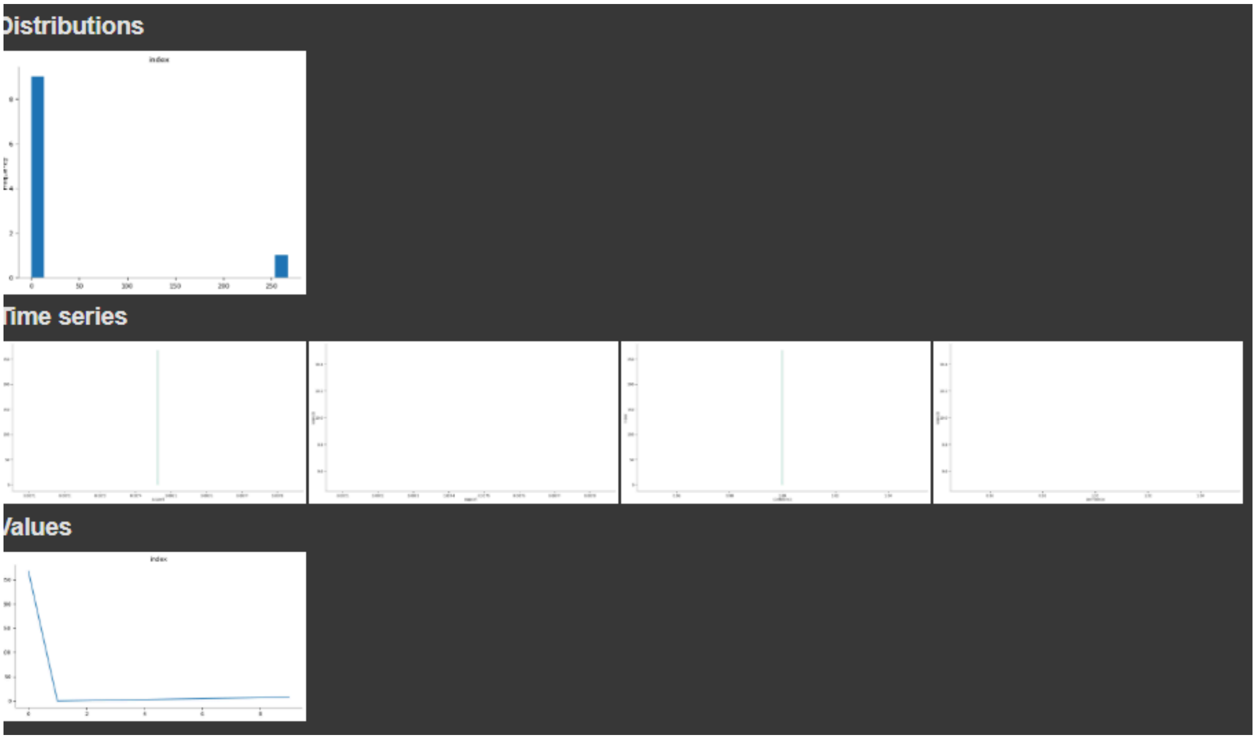


```
Choose Files  10. Market ...Analysis.csv
  10. Market Basket Analysis.csv(text/csv) - 2603 bytes, last modified: 4/18/2025 - 100% done
Saving 10. Market Basket Analysis.csv to 10. Market Basket Analysis (2).csv
```

1 to 10 of 10 entries Filter

index	antecedents	consequents	support	confidence	lift
267	frozenset({'canned fruit applesauce'})	frozenset({'99'})	0.007462686567164179	1.0	134.0
0	frozenset({'1'})	frozenset({'prepared soups salads'})	0.007462686567164179	1.0	134.0
1	frozenset({'prepared soups salads'})	frozenset({'1'})	0.007462686567164179	1.0	134.0
2	frozenset({'kitchen supplies'})	frozenset({'10'})	0.007462686567164179	1.0	134.0
3	frozenset({'10'})	frozenset({'kitchen supplies'})	0.007462686567164179	1.0	134.0
4	frozenset({'100'})	frozenset({'missing'})	0.007462686567164179	1.0	134.0
5	frozenset({'missing'})	frozenset({'100'})	0.007462686567164179	1.0	134.0
6	frozenset({'air fresheners candles'})	frozenset({'101'})	0.007462686567164179	1.0	134.0
7	frozenset({'101'})	frozenset({'air fresheners candles'})	0.007462686567164179	1.0	134.0
8	frozenset({'baby bath body care'})	frozenset({'102'})	0.007462686567164179	1.0	134.0

Show 25 per page



REFERENCE/CREDITS :

Libraries & Tools

- `mlxtend` Library Documentation:
  https://rasbt.github.io/mlxtend/user_guide/frequent_patterns/apriori/

- `pandas` Library:
  https://pandas.pydata.org/

- Google Colab:
  https://colab.research.google.com/

Dataset

- *" Market Basket Analysis.csv"* – a sample market basket dataset used for transaction-based learning.