

Course: ARTIFICIAL INTELLIGENCE
TITLE: MARKET BASKET INSIGHTS
PHASE 1 SUBMISSION

TEAM MEMBERS:

MOHANA PRIYA .N (au 311521106061)

MOUNICKA G.G (au 311521106062)

PADMA PRIYA .M. (au 311521106069)

YAMINI.H(au 311521106114)

ABSTRACT

Market basket analysis (MBA) is a data mining technique that is used to discover relationships between items that are frequently purchased together. It is a powerful tool that can be used by retailers to understand customer purchase patterns and make more informed decisions about product placement, promotions, and inventory management.

Module

The following is a high-level overview of a module for market basket analysis for groceries:

Data preparation: The first step is to prepare the grocery transaction data. This may involve cleaning the data, removing outliers, and encoding the data into a format that is compatible with the MBA algorithm. **Association rule mining:** The next step is to mine the transaction data for association rules. Association rules are of the form:

IF antecedent THEN consequent where antecedent and consequent are sets of items. The antecedent is the set of items that are purchased together frequently, and the consequent is the set of items that are likely to be purchased when the antecedent items are purchased. There are a number of different algorithms that can be used to mine association rules. One of the most common algorithms is the Apriori algorithm. The minimum confidence parameter specifies the minimum percentage of transactions that contain the antecedent items that must also contain the consequent items in order for the association rule to be considered valid.

Analysis: Once the association rules have been mined, they can be analyzed to identify patterns and trends. For example, a retailer may find that customers who purchase diapers are also likely to purchase baby wipes and formula. This information can then be used to develop targeted promotions or to place these products together in the store.

Benefits

Market basket analysis can provide a number of benefits to retailers, including:

Increased sales: Retailers can use MBA to identify patterns and trends in customer purchase behavior. This information can then be used to develop targeted promotions or to place products together in the store in a way that encourages customers to purchase more items.

Reduced inventory costs: Retailers can use MBA to identify items that are frequently purchased together. This information can then be used to ensure that these items are always in stock, while reducing the amount of inventory that is held for items that are not purchased as frequently.

Improved customer satisfaction: Retailers can use MBA to understand their customers' needs and preferences better. This information can then be used to improve the customer experience, such as by offering more convenient shopping options or by providing more personalized recommendations.

Dataset:

- The Online Retail II data set, which includes the sales data of the UK-based online sales store, was used.
- Sales data between 01/12/2009 - 09/12/2011 are included in the data set.

<https://www.kaggle.com/datasets/aslanahmedov/market-basket-analysis>

Invoice	StockCode	Description	Quantity	InvoiceDate	Price	Customer ID	Country
36365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.550	17850.000	United Kingdom
36365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.390	17850.000	United Kingdom
36365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.750	17850.000	United Kingdom
36365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.390	17850.000	United Kingdom
36365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.390	17850.000	United Kingdom

Problem

Statements:

Suggesting products to users at the basket stage. In this study, we will apply Market Basket analysis using the Apriori algorithm:

1. Import Data & Data Preprocessing
2. Preparing Invoice-Product Matrix for ARL Data Structure
3. Determination of Association Rules
4. Suggesting appropriate product offers to customers at the basket stage
5. Functionalization

Program

```
df_ = pd.read_excel('../input/online-retail-dataset/online_retail_II.xlsx', sheet_name='Year 2010-2011')
# Import & Filter Data:

def data_filter(dataframe, country=False, Country=""):
    if country:
        dataframe = dataframe[dataframe["Country"] == Country]
    return dataframe

def outlier_thresholds(dataframe, variable):
    quartile1 = dataframe[variable].quantile(0.01)
    quartile3 = dataframe[variable].quantile(0.99)
    interquartile_range = quartile3 - quartile1
    up_limit = quartile3 + 1.5 * interquartile_range
    low_limit = quartile1 - 1.5 * interquartile_range
    return low_limit, up_limit

def replace_with_thresholds(dataframe, variable):
    low_limit, up_limit = outlier_thresholds(dataframe, variable)
    dataframe.loc[(dataframe[variable] < low_limit), variable] = low_limit
    dataframe.loc[(dataframe[variable] > up_limit), variable] = up_limit

def data_prep(dataframe):
    # Data preprocessing:
    dataframe.dropna(inplace=True)

    # Delete if the product name contains "POST":
    dataframe = dataframe[~dataframe["StockCode"].str.contains("POST", na=False)]

    dataframe = dataframe[~dataframe["Invoice"].str.contains("C", na=False)]
```

```

dataframe = dataframe[dataframe["Quantity"] > 0]
dataframe = dataframe[dataframe["Price"] > 0]
replace_with_thresholds(dataframe, "Quantity")
replace_with_thresholds(dataframe, "Price")
return dataframe

# Invoice Product Matrix:
def create_invoice_product_df(dataframe, id=False):
    if id:
        return dataframe.groupby(['Invoice', "StockCode"])[ 'Quantity' ].sum().unstack().fillna(0)
    else:
        return dataframe.groupby(['Invoice', 'Description'])[ 'Quantity' ].sum().unstack().fillna(0)

# Find Product name with Stock Code:
def check_id(dataframe, stockcode):
    product_name = dataframe[dataframe["StockCode"] == stockcode]["Description"].unique()[0]
    return stockcode, product_name

# Apriori Algorithm & ARL Rules:
def apriori_alg(dataframe, support_val=0.01):
    inv_pro_df = create_invoice_product_df(dataframe, id=True)
    frequent_itemsets = apriori(inv_pro_df, min_support=support_val, use_colnames=True)
    rules = association_rules(frequent_itemsets, metric="support", min_threshold=support_val)
    sorted_rules = rules.sort_values("support", ascending=False)
    return sorted_rules

def recommend_product(dataframe, product_id, support_val= 0.01, num_of_products=5):
    sorted_rules = apriori_alg(dataframe, support_val)
    recommendation_list = []
    for idx, product in enumerate(sorted_rules["antecedents"]):
        for j in list(product):
            if j == product_id:
                recommendation_list.append(list(sorted_rules.iloc[idx]["consequents"])[0])
    recommendation_list = list( dict.fromkeys(recommendation_list) )
    return(recommendation_list[0:num_of_products])

# Data Preparation:
df = df_.copy()

df = data_prep(df)
df = data_filter(df, country=True, Country="Germany")
df.head()

```

Output

	Invoice	StockCode	Description	Quantity	InvoiceDate	Price	Customer ID
1109	536527	22809	SET OF 6 T-LIGHTS SANTA	6.000	2010-12-01 13:04:00	2.950	12662.000
1110	536527	84347	ROTATING SILVER ANGELS T-LIGHT HLDR	6.000	2010-12-01 13:04:00	2.550	12662.000
1111	536527	84945	MULTI COLOUR SILVER T-LIGHT HOLDER	12.000	2010-12-01 13:04:00	0.850	12662.000
1112	536527	22242	5 HOOK HANGER MAGIC TOADSTOOL	12.000	2010-12-01 13:04:00	1.650	12662.000
1113	536527	22244	3 HOOK HANGER MAGIC GARDEN	12.000	2010-12-01 13:04:00	1.950	12662.000

Recommendation System Function

Program

```
def recommendation_system_func(dataframe,support_val=0.01, num_of_products= 5 ):
    product_id = input("Enter a product id:")

    if product_id in list(dataframe["StockCode"].astype("str").unique()):
        product_list = recommend_product(dataframe, int(product_id), support_val, num_of_products)

        if len(product_list) == 0:
            print("There is no product can be recommended!")
        else:
            print("Related products with product id -" , product_id , "can be seen below:")

            for i in range(0, len(product_list[0:num_of_products])):
                print(check_id(dataframe, product_list[i]))

    else:
        print("Invalid Product Id, try again!")
```

Output

Enter product id - 2 (22725)

Related products with product id - 22725 can be seen below:

```
(22727, 'ALARM CLOCK BAKELIKE RED ')
(22326, 'ROUND SNACK BOXES SET OF4 WOODLAND ')
(22728, 'ALARM CLOCK BAKELIKE PINK')
(22726, 'ALARM CLOCK BAKELIKE GREEN')
(22729, 'ALARM CLOCK BAKELIKE ORANGE')
```

```
recommendation_system_func(df)
```

There is no product can be recommended!

```
recommendation_system_func(df)
```

Invalid Product Id, try again!

Conclusion

Market basket analysis is a powerful tool that can be used by retailers to understand customer purchase patterns and make more informed decisions about product placement, promotions, and inventory management. By analyzing the items that customers frequently purchase together, retailers can develop strategies to increase sales, reduce inventory costs, and improve customer satisfaction.