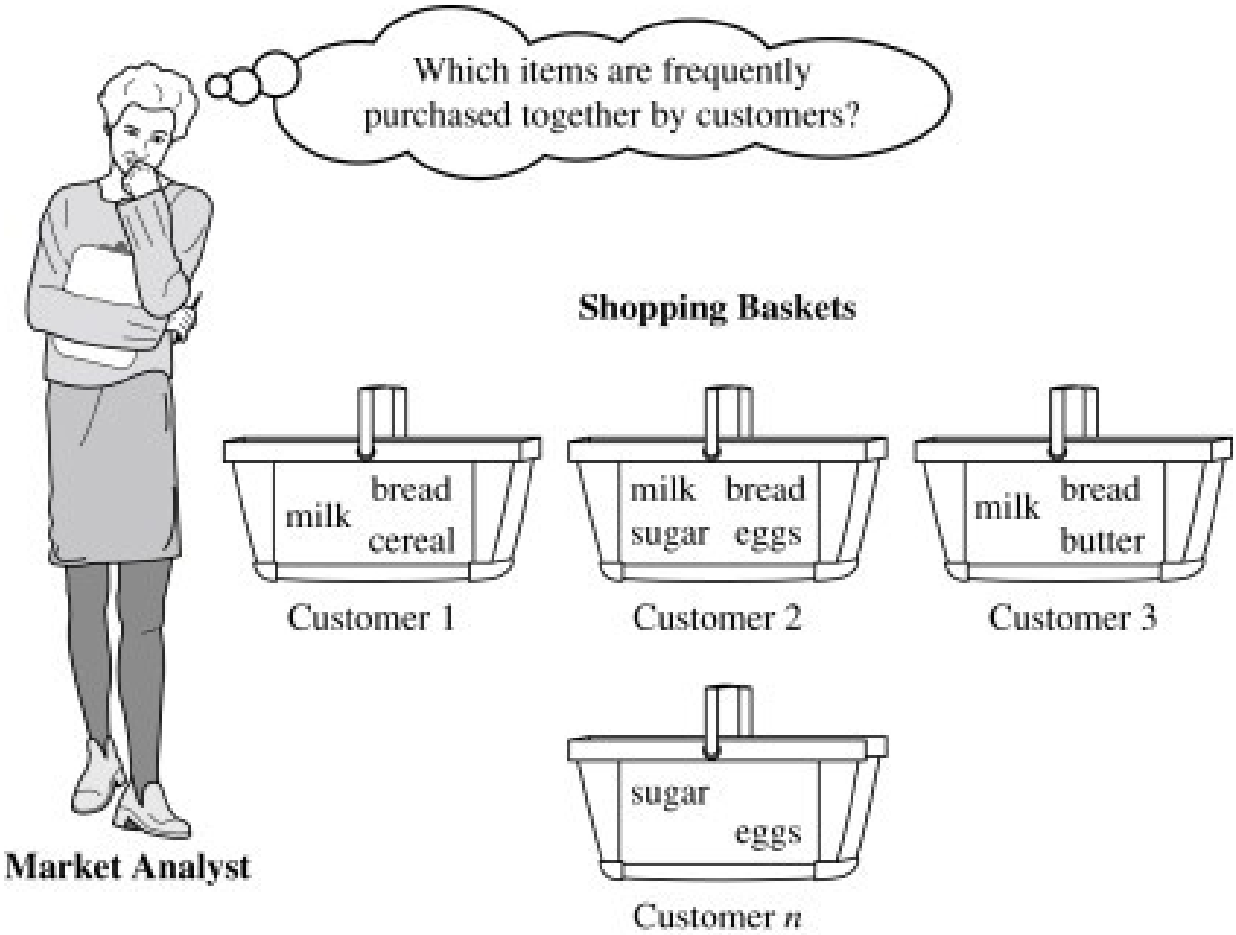
MARKET BASKET ANALYSIS

PHASE 3 SUBMISSION DOCUMENT

DEVELOPMENT PART 1

TOPIC : Start building the Market basket analysis model by loading and pre-processing the dataset.



Introduction :

Market Basket Analysis (MBA), also known as Association Rule Mining or Affinity Analysis, is a data mining and analytical technique used in retail and e-commerce to discover patterns, relationships, and associations among products that customers frequently purchase together.

The primary goal of MBA is to uncover insights into customer buying behavior, enhance marketing strategies, and optimize product placement in stores.

Given detaset :

	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...
195	536389	CHRISTMAS LIGHTS 10 REINDEER	6	2010-12-01 10:03:00	8.50	12431.0	Australia
196	536389	VINTAGE UNION JACK CUSHION COVER	8	2010-12-01 10:03:00	4.95	12431.0	Australia
197	536389	VINTAGE HEADS AND TAILS CARD GAME	12	2010-12-01 10:03:00	1.25	12431.0	Australia
198	536389	SET OF 3 COLOURED FLYING DUCKS	6	2010-12-01 10:03:00	5.45	12431.0	Australia
199	536389	SET OF 3 GOLD FLYING DUCKS	4	2010-12-01 10:03:00	6.35	12431.0	Australia

200 rows × 7 columns

1.Import libraries :

```
import pandas as pd
import numpy as np
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules
```

2. Load the Dataset:

You can use various data formats like CSV, Excel, or a database. Python libraries like Pandas make it easy to read data. Here's an example to load a CSV file:

```
import pandas as pd

# Load the dataset
df = pd.read_csv("sales_data.csv")
```

3. Exploratory Data Analysis (EDA):

Perform initial data exploration to understand the dataset, including checking for missing values, unique items, and general statistics.

```
# Check for missing values
print(df.isnull().sum())

# Display the first few rows
print(df.head())

# Get unique items
unique_items = df['Item'].unique()
```

3. Data Preprocessing:

In MBA, you typically need to transform the data into a suitable format. The most common format is a one-hot encoded DataFrame, where each row represents a transaction, and each column

represents an item, with binary values indicating the presence of an item in the transaction.

```
# Perform one-hot encoding
basket = pd.get_dummies(df['Item'])

# Group transactions by their index
basket = basket.groupby(level=0).sum()
```

4. Market Basket Analysis (Apriori Algorithm):

Now that your data is in the right format, you can use an MBA algorithm like Apriori to discover associations between items. You can use libraries like mlxtend in Python for this.

```
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules

# Find frequent item sets
frequent_item_sets = apriori(basket, min_support=0.1, us

# Find association rules
rules = association_rules(frequent_item_sets, metric="li
```

5. Interpret the Results:

Analyze the generated association rules, considering metrics like support, confidence, and lift. These rules represent item associations and can be used for business recommendations.

```
# Display association rules
print(rules)
```

Importance of loading and processing dataset :

Loading and processing a dataset is a crucial initial step in any data analysis or machine learning task, including Market Basket Analysis. Here are the key reasons why loading and processing a dataset is important:

1. **Data Availability:** Without loading the dataset, you cannot access or analyze the data. It's the first step in making the data available for your analysis.
2. **Data Understanding:** By loading the data, you gain an initial understanding of its structure, size, and format. You can see the raw information and start to identify any potential issues or anomalies.
3. **Data Cleaning:** Datasets are rarely perfect. Loading the data allows you to identify and address missing values, duplicates, inconsistent formatting, and other data quality issues. Clean data is essential for accurate analysis.
4. **Data Transformation:** Depending on the analysis you want to perform, you may need to transform the data. For Market Basket Analysis, this often includes converting transactional data into a suitable format, like one-hot encoding, where items are represented as binary variables indicating presence or absence.
5. **Data Exploration (EDA):** Once the data is loaded, you can perform exploratory data analysis to gain insights. EDA involves examining summary statistics, visualizing the data, and identifying trends, patterns, and outliers. This helps you make informed decisions about the analysis approach.

6. **Preparation for Analysis:** Data processing is critical for preparing the data to work with specific analysis techniques or machine learning algorithms. For Market Basket Analysis, this could involve creating transaction-item matrices, which serve as the foundation for finding associations.
7. **Efficiency:** Processing the data, such as optimizing data structures and cleaning, ensures that your analysis runs efficiently. It can save computational resources and time during the analysis phase.
8. **Accuracy and Reliability:** Properly processing the data helps ensure that the results of your analysis are accurate and reliable. By addressing issues like missing values and duplicates, you reduce the risk of drawing incorrect conclusions.
9. **Data Privacy and Compliance:** Processing data can also involve handling sensitive information appropriately, ensuring that you comply with privacy regulations and protect personal data.
10. **Customization:** Depending on the specific requirements of your analysis, you may need to customize the data processing steps. This customization can address unique data challenges and improve the quality of the analysis.

Challenges involved in loading and preprocessing a market basket dataset :

Loading and preprocessing a house price dataset can be a complex task, as it involves handling various types of data and dealing with several challenges. Here are some of the common challenges involved in this process:

1. **Data Quality:** Datasets may contain missing values, outliers, errors, or inconsistencies. Cleaning and imputing these issues is a crucial step in preprocessing.
2. **Data Size:** Large datasets can be memory-intensive and slow to process. You may need to consider techniques for managing and working with big data.

3. **Data Types:** House price datasets often include a mix of data types, such as numerical (e.g., price, square footage) and categorical (e.g., location, property type). Handling these different data types requires distinct preprocessing steps.
4. **Feature Engineering:** Extracting meaningful features from raw data is essential. This may involve creating new variables, aggregating data, or transforming existing features to improve the model's performance.
5. **Normalization and Scaling:** To ensure that numerical features have the same scale, preprocessing may involve normalization or standardization. This can be important for machine learning models that are sensitive to feature scales.
6. **Categorical Data Handling:** Categorical variables must be encoded (e.g., one-hot encoding) so that they can be used in machine learning algorithms. Handling a large number of categories can be challenging.
7. **Dealing with Outliers:** Outliers can significantly impact the accuracy of house price predictions. Deciding how to handle outliers, whether through removal or transformation, is a critical step.
8. **Dimensionality Reduction:** Some datasets may have high dimensionality, with many features. Dimensionality reduction techniques like Principal Component Analysis (PCA) may be needed.
9. **Data Imbalance:** If the dataset has an imbalanced distribution of target values (e.g., very few luxury properties compared to standard ones), the model may have difficulty learning patterns. Techniques like oversampling, undersampling, or using appropriate evaluation metrics are needed.
10. **Handling Time-Series Data:** Some datasets may include time-series data, such as historical price trends. Special techniques for time-series analysis may be required.

11. **Geospatial Data:** Datasets that include geographical information (e.g., latitude, longitude) may require specialized preprocessing to account for spatial relationships.
12. **Handling Missing Data:** Deciding how to deal with missing data is important. Imputation techniques, such as mean, median, or machine learning-based imputation, need to be chosen.
13. **Data Normalization and Encoding:** Ensuring that data is in a suitable format for machine learning models is crucial. This includes encoding text data, normalizing numeric data, and splitting the dataset into training and testing sets.
14. **Data Privacy and Security:** Handling sensitive information like property addresses and owner details requires safeguarding privacy and complying with data protection regulations.
15. **Data Splitting:** Properly splitting the dataset into training, validation, and test sets is essential for model evaluation and avoiding data leakage.
16. **Feature Selection:** Identifying the most relevant features and eliminating irrelevant ones can improve model performance and reduce complexity.

How to overcome the challenges of loading and preprocessing a house price dataset :

Overcoming the challenges of loading and preprocessing a house price dataset involves a systematic approach and a combination of data preprocessing techniques. Here are some strategies to address these challenges:

1. **Data Quality Issues:**
    - Handle missing data by imputing it with appropriate values, such as the mean, median, or using more advanced imputation methods like K-Nearest Neighbors (KNN) imputation.

- Identify and handle outliers using methods like Z-scores, interquartile range (IQR), or robust statistical methods.
- Correct errors and inconsistencies in the data, such as typos or incorrect entries.

2. **Data Size:**
    - If the dataset is large and memory-intensive, consider using data sampling to work with a subset for exploratory analysis and model building.
    - Use distributed computing frameworks like Apache Spark for big data processing.
3. **Data Types:**
    - Categorize and differentiate between numerical and categorical features, as they require different preprocessing steps.
    - Convert categorical data into a numerical format using techniques like one-hot encoding or label encoding.
4. **Feature Engineering:**
    - Create new features that might be more informative for predicting house prices, such as square footage per bedroom, age of the property, or proximity to amenities.
    - Aggregate or transform existing features to create meaningful variables.
5. **Normalization and Scaling:**
    - Normalize or standardize numerical features to ensure they have a similar scale, especially when using machine learning algorithms that are sensitive to feature scales.
6. **Categorical Data Handling:**
    - Apply one-hot encoding to convert categorical variables into binary (0/1) format.
    - Consider feature selection techniques to reduce dimensionality if one-hot encoding leads to a large number of new binary columns.
7. **Dealing with Outliers:**

- Decide on a strategy for handling outliers, such as truncation (capping), transformation, or treating them as a separate category.

8. **Dimensionality Reduction:**
    - Use dimensionality reduction techniques like Principal Component Analysis (PCA) to reduce the number of features and avoid multicollinearity.
9. **Data Imbalance:**
    - Implement techniques like oversampling or undersampling for imbalanced datasets to ensure the model can learn from minority classes.
    - Choose appropriate evaluation metrics that account for imbalanced data, such as area under the Receiver Operating Characteristic curve (AUC-ROC) or F1-score.
10. **Handling Time-Series Data:**
    - Utilize time-series analysis techniques if your dataset involves time-dependent variables, such as historical price trends.
11. **Geospatial Data:**
    - Use geospatial analysis methods to extract insights from geographical features like latitude, longitude, and distances to certain locations.
12. **Handling Missing Data:**
    - Choose an imputation method that is suitable for the nature of missing data (e.g., mean for numerical, mode for categorical) and document the imputation process.
13. **Data Normalization and Encoding:**
    - Standardize data formats, ensuring that text data is encoded, numeric data is normalized, and date-time data is formatted consistently.
14. **Data Privacy and Security:**
    - Safeguard sensitive information and adhere to data protection regulations. Consider anonymization techniques when working with personal data.

15. **Data Splitting:**
    - Properly split the dataset into training, validation, and test sets to avoid data leakage and ensure model evaluation is representative of real-world performance.
16. **Feature Selection:**
    - Employ feature selection methods, such as Recursive Feature Elimination (RFE) or feature importance from tree-based models, to reduce the number of features to the most relevant ones.

loading dataset :

Loading a dataset for Market Basket Analysis typically involves reading transactional data from a file, such as a CSV file, into a format suitable for analysis. Below is a Python code example using the Pandas library to load a sample Market Basket dataset from a CSV file

Assuming you have a CSV file named "market_basket_data.csv" with transaction data, here's how you can load it:

```
import pandas as pd

# Load the Market Basket dataset from a CSV file
data = pd.read_csv("market_basket_data.csv")

# Display the first few rows of the dataset
print(data.head())
```

Make sure to replace "market_basket_data.csv" with the actual filename of your dataset. This code uses Pandas to read the data

into a DataFrame, which is a common data structure used for data analysis in Python.

The dataset should have each row representing a transaction, and the columns should represent items or products. For Market Basket Analysis, this is the typical format.

Once you've loaded the dataset, you can further preprocess it and perform Market Basket Analysis to discover associations between items that are frequently purchased together. This may include converting the data into the required format, removing any unnecessary columns, and using specialized libraries like mlxtend for frequent itemset mining and association rule generation.

Program :

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score, mean_absolute_error
from sklearn.linear_model import LinearRegression
from sklearn.linear_model import Lasso
```

Out[4]:

	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...
195	536389	CHRISTMAS LIGHTS 10 REINDEER	6	2010-12-01 10:03:00	8.50	12431.0	Australia
196	536389	VINTAGE UNION JACK CUSHION COVER	8	2010-12-01 10:03:00	4.95	12431.0	Australia
197	536389	VINTAGE HEADS AND TAILS CARD GAME	12	2010-12-01 10:03:00	1.25	12431.0	Australia
198	536389	SET OF 3 COLOURED FLYING DUCKS	6	2010-12-01 10:03:00	5.45	12431.0	Australia
199	536389	SET OF 3 GOLD FLYING DUCKS	4	2010-12-01 10:03:00	6.35	12431.0	Australia

200 rows × 7 columns

Preprocessing the dataset:

Preprocessing in the context of dataset refers to a set of operations and techniques applied to raw data before it is used for analysis, machine learning, or any other data-related task. The goal of preprocessing is to clean, transform, and prepare the data so that it is suitable for the specific task at hand

After we will clear our data frame, will remove missing values.

```
13 #complete.cases(data) removing rows with missing values in any column of data frame
14 itemslist <- itemslist[complete.cases(itemslist), ]
```

The summary gives us some useful information:To apply Association Rule mining, we need to convert dataframe into transaction data to make all items that are bought together in one invoice will be in one row. Below lines of code will combine all products from one BillNo and Date and combine all products from that BillNo and Dat

```
18 #ddply(dataframe, variables_to_split_dataframe, function)
19 transaxtionData <- ddply(itemslist,c("BillNo","Date"),
20                          function(df1)paste(df1$Itemname,
21                                             collapse = ","))
```

We don’t need BillNo and Date, we will make it as Null.
Next, you have to store this transaction data into .csv

```
22 transaxtionData$BillNo <- NULL
23 transaxtionData$Date <- NULL
24 #will gave the name to column "item"
25 colnames(transaxtionData) <- c("items")
```

This how should look transaction data before we will go to next step.

```
#quote: If TRUE it will surround character or factor column with double quotes.
#If FALSE nothing will be quoted
#row.names: either a logical value indicating whether the row names of x are to be
#written along with x, or a character vector of row names to be written.
write.csv(transaxtionData, "assigment1_itemslist.csv", quote = FALSE, row.names = FALSE)
```

items			
WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	KNITTED UNION FLAG HOT WATER BOTTLE
HAND WARMER UNION JACK	HAND WARMER RED POLKA DOT		
ASSORTED COLOUR BIRD ORNAMENT	POPPY'S PLAYHOUSE BEDROOM	POPPY'S PLAYHOUSE KITCHEN	FELTCRAFT PRINCESS CHARLOTTE DOLL
JAM MAKING SET WITH JARS	RED COAT RACK PARIS FASHION	YELLOW COAT RACK PARIS FASHION	BLUE COAT RACK PARIS FASHION
BATH BUILDING BLOCK WORD			
ALARM CLOCK BAKELIKE PINK	ALARM CLOCK BAKELIKE RED	ALARM CLOCK BAKELIKE GREEN	PANDA AND BUNNIES STICKER SHEET
PAPER CHAIN KIT 50'S CHRISTMAS			
HAND WARMER RED POLKA DOT	HAND WARMER UNION JACK		
WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	EDWARDIAN PARASOL RED
VICTORIAN SEWING BOX LARGE			
WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	EDWARDIAN PARASOL RED
HOT WATER BOTTLE TEA AND SYMPATHY	RED HANGING HEART T-LIGHT HOLDER		
HAND WARMER RED POLKA DOT	HAND WARMER UNION JACK		
JUMBO BAG PINK POLKADOT	JUMBO BAG BAROQUE BLACK WHITE	JUMBO BAG CHARLIE AND LOLA TOYS	STRAWBERRY CHARLOTTE BAG
JAM MAKING SET PRINTED			
RETROSPOT TEA SET CERAMIC 11 PC	GIRLY PINK TOOL SET	JUMBO SHOPPER VINTAGE RED PAISLEY	AIRLINE LOUNGE

Conclusion :

In conclusion, loading and preprocessing the dataset for Market Basket Analysis is a fundamental and essential step in uncovering valuable insights into customer buying behavior and optimizing sales and marketing strategies. Key takeaways specific to Market Basket Analysis include:

1. **Data Loading:** Importing transactional data into an accessible data structure, such as a Pandas DataFrame in Python, is the first step in the analysis.
2. **Data Exploration:** An initial examination of the dataset helps you understand its structure, identify missing values, and inspect the raw information.
3. **Data Transformation:** Conversion of the data into a suitable format, where transactions become rows, and items become columns with binary values (1 for presence, 0 for absence) is essential. This is typically achieved through one-hot encoding.
4. **Data Summary:** Calculating summary statistics, such as the number of transactions and the average number of items per transaction, provides insights into the dataset's characteristics.

5. **Remove Rare Items:** Consider excluding items that appear infrequently in transactions to improve the relevance of association rules.
6. **Handling Missing Values:** Address any missing values in the dataset, typically through imputation or data cleaning techniques.
7. **Save the Processed Data:** Optionally, you can save the preprocessed data for future use, streamlining Market Basket Analysis in the future.

Through these preprocessing steps, you prepare your dataset for frequent itemset mining and association rule generation. This well-structured and clean data serves as the foundation for discovering item associations and optimizing business strategies in retail and e-commerce.