**MARKET BASKET ANALYSIS WITH APRIORI ALGORITHM**

**Problem Statement:**

The problem is to perform market basket analysis on a provided dataset to unveil hidden patterns and associations between products. The goal is to understand customer purchasing behavior and identify potential cross-selling opportunities for a retail business. This project involves using association analysis techniques, such as Apriori algorithm, to find frequently co-occurring products and generate insights for business optimization.

Innovative techniques that can be used to improve the prediction system's accuracy and robustness are

* Ensemble Learning
* Deep Learning Architectures
* Transfer Learning
* AutoML
* Bayesian Optimization
* Reinforcement Learning
* Graph Neural Networks (GNNs)
* Self-Supervised Learning, etc.

**1) ENSEMBLE LEARNING:**

Ensemble learning is a machine learning technique that enhances accuracy and resilience in forecasting by merging predictions from multiple models. It aims to mitigate errors or biases that may exist in individual models by leveraging the collective intelligence of the ensemble.

The underlying concept behind ensemble learning is to combine the outputs of diverse models to create a more precise prediction. By considering multiple perspectives and utilizing the strengths of different models, ensemble learning improves the overall performance of the learning system. This approach not only enhances accuracy but also provides resilience against uncertainties in the data. By effectively merging predictions from multiple models, ensemble learning has proven to be a powerful tool in various domains, offering more robust and reliable forecasts.

More generally, ensemble models can be applied to any base learner beyond trees, in averaging methods such as

• Gradient-boosted trees

• Random forests and other randomized tree ensembles

• Bagging meta-estimator

• Voting Classifier

• Voting Regressor

• Stacked generalization

• AdaBoost

**1. Random Forest:**

**Accuracy Improvement:** Random Forest combines multiple decision trees trained on different subsets of the data and features. By averaging their predictions, it reduces the risk of overfitting and captures complex relationships in the data.

**Robustness:** Random Forest is less prone to overfitting than individual decision trees, making it more robust to noise and outliers in the data. It's also less sensitive to changes in the training data.

**2. Gradient Boosting:**

**Accuracy Improvement:** Gradient boosting builds an ensemble of decision trees sequentially, where each tree corrects the errors of the previous ones. This iterative process results in improved prediction accuracy, as it focuses on difficult-to-predict examples.

**Robustness:** By continuously refining the model, gradient boosting can adapt to complex patterns in the data, making it robust to noisy and complex datasets.

**3. Bagging Meta-Estimator:**

**Accuracy Improvement:** Bagging creates multiple subsets of the training data through bootstrapping and trains individual models on each subset. By averaging their predictions (for regression) or taking a majority vote (for classification), it reduces variance and improves accuracy.

**Robustness**: Bagging helps mitigate the impact of outliers and noisy data points by averaging the predictions of multiple models.

**4. Voting Classifier and Voting Regressor:**

**Accuracy Improvement:** Voting combines predictions from multiple base models (classifiers or regressors) using a majority vote or weighted average, respectively. It can improve accuracy by reducing bias and variance.

**Robustness:** Voting can be more robust than individual models since it benefits from the diversity of the ensemble.

**5. Stacked Generalization:**

**Accuracy Improvement:** Stacking combines predictions from multiple machine learning models (base models) using a meta-model. It leverages the strengths of different models, potentially improving accuracy. The meta-model learns to assign weights to the base models, which can adapt to the data.

**Robustness:** Stacking can reduce overfitting by combining models with different biases, leading to a more robust and stable prediction system.

**6. AdaBoost:**

**Accuracy Improvement**: AdaBoost focuses on training weak learners sequentially and assigns higher weights to misclassified examples. This process improves accuracy by emphasizing difficult-to-classify instances.

**Robustness:** AdaBoost adapts to the data by iteratively adjusting the model's emphasis on different examples, making it robust to noisy data.

**2) REINFORCEMENT LEARNING (RL):**

Reinforcement learning is a machine learning paradigm that focuses on making a sequence of decisions to maximize a cumulative reward. While RL is often associated with applications like game playing and robotics, it can also be applied to prediction systems in various domains.

Advantages of using RL:

* Improved Model Selection
* Hyperparameter Tuning
* Exploration of Feature Engineering
* Dynamic Model Ensembles

**3) AUTOML (AUTOMATED MACHINE LEARNING):**

AutoML refers to the automation of the end-to-end machine learning pipeline, from data preprocessing to model selection and hyperparameter tuning. It streamlines the process of building predictive models, making it accessible to non-experts and experts alike.

Advantages of using AutoML:

* Efficient Model Search
* Hyperparameter Optimization
* Feature Engineering
* Robustness and Reliability
* Scalability

**PUTTING THE DESIGN TECHNIQUE INTO IMPLEMENTATION (IMPLEMENTATION ALGORITHM):**

**1. Data Understanding**

Step 1: Load the dataset and explore it to understand its structure, column meanings, and the relationships between variables.

Step 2: Perform exploratory data analysis (EDA) to get insights into the dataset, such as the most popular products, the average order value, and the most common customer purchase patterns.

**2. Data Preprocessing**

Step 1: Handle missing values, if any, through imputation or removal.

Step 2: Remove duplicates.

Step 3: Transform the data into a transactional format where each transaction is represented as a list of purchased items.

Step 4: Convert categorical data, like product categories, into numerical values, if necessary.

**3. Applying the Apriori Algorithm**

Step 1: Define two thresholds: support and confidence.

Support: The percentage of transactions that contain a given itemset.

Confidence: The percentage of transactions that contain an itemset given that they also contain another itemset.

Step 2: Implement the Apriori algorithm to find all the frequent itemsets and association rules that meet the support and confidence thresholds.

**4. Interpretation of Results**

Step 1: Analyze the generated association rules to understand the relationships between products.

Step 2: Identify products that are frequently bought together and the strength of these associations.

Step 3: Pay attention to the lift values of the association rules. Lift is a measure of how strong an association is. A lift value of 1.0 means that the two items are independent of each other. A lift value greater than 1.0 means that the two items are positively correlated, meaning that they tend to occur together more often than expected by chance.

**5. Visualization**

Step 1: Use visualization tools and techniques to represent the discovered patterns and associations effectively. For example, you might create a heatmap to visualize item co-occurrence, a network graph to show item relationships, or a bar chart to display support and confidence values.

**6. Business Recommendations**

Step 1: Based on our interpretation of the results and visualizations, we need to provide actionable recommendations to the retail business.

Step 2: These recommendations may include optimizing product placements in stores, creating bundled offers for frequently associated products, or designing targeted marketing campaigns based on customer purchasing behavior.

**7. Documentation**

Step 1: Document the entire market basket analysis process, including the data preprocessing steps, Apriori algorithm configuration, and interpretation of results.

Step 2: Summarize your findings, insights, and recommendations in a report or presentation format.

**8. Continuous Improvement**

Step 1: Market basket analysis is an ongoing process.

Step 2: Continuously monitor customer behavior and adapt your recommendations as shopping trends change.

Step 3: Consider automating the process for regular updates and insights.

**4) DEEP LEARNING ARCHITECTURE:**

**Feed Forward Neural Network**

Association rule mining aims to discover interesting relationships and patterns within large datasets particularly in transactional databases. The Apriori algorithm specifically focuses on finding frequent itemsets and generating association rules based on these itemsets. The existing Apriori algorithm can be modified by using neural network method in order to optimize the prediction results. Neural networks are a powerful tool for market basket analysis. A Feed forward Neural Network (FFNN) is a type of artificial neural network where information moves in only one direction forward, from the input nodes, through hidden layers (if any), to the output nodes. They can be used to identify patterns in customer behavior and to predict which products are likely to be purchased together.

**1. Embedding Products:**

Each unique item (product) is mapped to a dense vector of real numbers capturing its features and relationships with other items. These embeddings are learned during the training process and allow the network to understand the semantic meaning of items in the context of customer transactions.

**2. Feedforward Neural Network Model:**

Input Layer:

The input layer receives sequences of embedded items. Each item is represented as a vector in the input sequence.

Hidden Layers :

Depending on the complexity of the problem is one or more hidden layers can be added with varying numbers of nodes. These hidden layers allow the network to learn intricate patterns in the data.

Output Layer:

The output layer consists of nodes representing different product pairs. Each node predicts the likelihood of a particular product pair being bought together.

**3. Training the Model:**

The model is trained using historical transactional data. The loss function like categorical cross-entropy for association rule prediction quantifies the difference between predicted and actual sequences.

Optimizer:

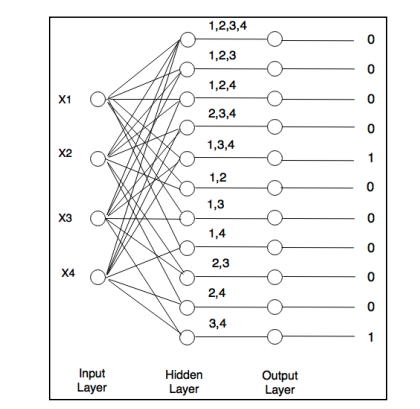
An appropriate optimizer is used to minimize the loss function during training.

The model is trained over multiple epochs, adjusting the batch size and weights.

**4. Presenting Insights:**

After the FFNN is trained, it can predict associations between products. For instance, given a set of products in a basket the model can predict the likelihood of other products being included in the same transaction. The insights of accuracy in association rules predicted by the trained RNN model can be presented through graphical representations.

**EXAMPLE: SINGLE FEED FORWARD NEURAL NETWORK**



The input layer of the neural network would represent the products in the customer's shopping basket. The hidden layer would learn to identify patterns in the data such as which products are often purchased together. The output layer would predict the probability that the customer will purchase a particular product given the products that they have already purchased.

**5) USING VISUALIZING TOOLS FOR ENHANCED INSIGHTS PRESENTATIONS**

**DATA IMPORT:**

**import numpy as np # linear algebra**

**import pandas as pd**

**from matplotlib import pyplot as plt**

**df=pd.read\_excel("/kaggle/input/market-basket-analysis/Assignment-1\_Data.xlsx")**

Firstly, importing necessary python libraries such as numpy, pandas, and matplotlib. Import the datset (Assignment-1\_Data.xlsx).

**DATA UNDERSTANDING AND EXPLORATION**:

**df.info()**

This method prints information about the created data frame “df” including the index data type and columns, non-null values and memory usage. Drop any rows where item name column is null. Drop any rows where item quantity sold is 0 or less.

**df["SumPrice"]=df["Quantity"]\*df["Price"]**

Create a new column, SumPrice, that tells us total sales revenue (Quantity \* Price) of the item.

**TRANSFORMATION OF DATA USING ASSOCIATION RULES:**

Market Basket Analysis using Apriori Algorithm and Association Rule Mining

* Convert the Dataset into transactional format (Each row is one bill number with every item sold in that bill in a list)
* Create a one-hot matrix of the products (Product sold = 1, Not sold = 0)
* Merge the transactional matrix and the one hot matrix
* Import the mlxtend library and perform association mining and generate association rules

**GENERATING ASSOCIATION RULES:**

**rules = association\_rules(frequent\_itemsets, metric="lift", min\_threshold=1)**

**rules**

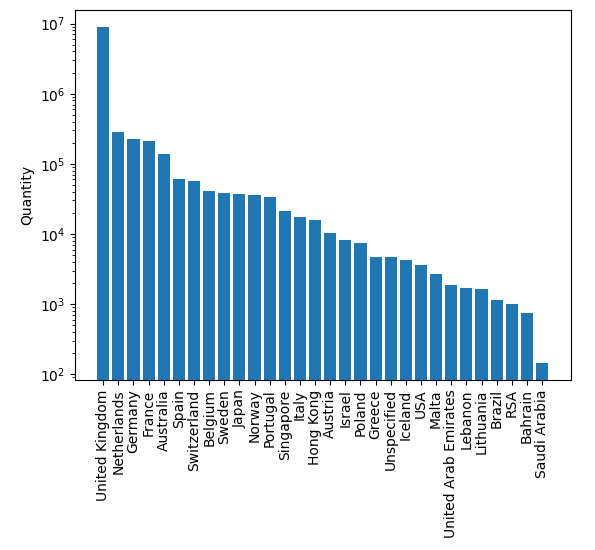
These rules can be used by retailers to make recommendations to customers, or to design marketing campaigns. This is based on “frequent\_itemsets” provided in the given dataset.

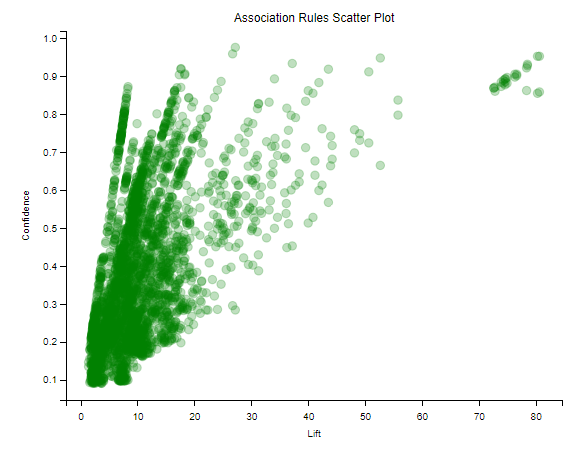
**FILTERING THE GENERATED RULES:**

Association rules are a powerful tool for data mining and machine learning. By filtering association rules, we can extract the most relevant and useful information from the data. Once the rules have been filtered, they can be used for a variety of tasks, such as: recommendation systems, market campaigns and fraud detection.

**VISUALIZATION OF THE RULES:**

The scatter plot shows a number of association rules with high lift and confidence. This means that there are a number of item combinations that are frequently purchased together. Retailers can use this information to make recommendations to customers or to design marketing campaigns.

****

****