**Implementation of Association Rule Mining algorithm (Apriori)**

import pandas as pd

import numpy as np

from mlxtend.frequent\_patterns import apriori, association\_rules

np.random.seed(42)

data = pd.DataFrame(np.random.randint(0, 2, (100, 6)),

columns=['Milk', 'Bread', 'Butter', 'Cheese', 'Eggs', 'Yogurt'])

print("Transaction Data:")

print(data.head(10))

print(f"\nTotal transactions: {len(data)}")

frequent\_itemsets = apriori(data, min\_support=0.2, use\_colnames=True)

print("\nFrequent Itemsets:")

print(frequent\_itemsets.sort\_values('support', ascending=False))

rules = association\_rules(frequent\_itemsets, metric="confidence", min\_threshold=0.5)

print("\nAssociation Rules:")

print(rules[['antecedents', 'consequents', 'support', 'confidence', 'lift']].sort\_values('lift', ascending=False))

print(f"\nSummary:")

print(f"Number of frequent itemsets: {len(frequent\_itemsets)}")

print(f"Number of association rules: {len(rules)}")

print(f"Minimum support: 0.2")

print(f"Minimum confidence: 0.5")

high\_lift\_rules = rules[rules['lift'] > 1.5]

if len(high\_lift\_rules) > 0:

print(f"\nStrong rules (lift > 1.5):")

for idx, row in high\_lift\_rules.iterrows():

print(f"{set(row['antecedents'])} -> {set(row['consequents'])} (lift: {row['lift']:.3f})")