In-Class Exercise: Association Rule Mining Project Guide

Project Overview

## In this project, you will explore real-world applications of association rule mining using transaction data. You will either:

- Select a real dataset

You will apply techniques such as Apriori or FP-Growth to discover meaningful patterns and rules, interpret your findings, and explore how these insights can be used in practice.

Project Goals

- Understand the principles of association rule mining

- Handle categorical and continuous variables (e.g., total amount)

- Implement and customize rule mining with constraints (support, confidence, lift)

- Explore and present meaningful results with real-world relevance

Dataset Options

## 1. You may select a public dataset from:

- Kaggle (e.g., groceries, online retail)

- UCI Repository

- Data.gov or OpenML

**2. Generate your own using Python, Excel, or Mockaroo.**

Your Task (Step-by-Step)

1. Explore the Dataset

- What kinds of features exist?

- Are they categorical, continuous, or timestamped?

2. Draft a Proposal (Due: today)

- Submit a 1–2 page document explaining your approach and goals.

3. Preprocess the Data

- Clean it, bin continuous variables, and convert to transaction format.

4. Implement Mining

- Use Apriori or FP-Growth to extract frequent itemsets and rules.

5. Analyze and Interpret Results

- Focus on top 5–10 rules.

- Use support, confidence, lift to prioritize.

- Visualize your findings.

6. Connect to Real-World Use

- Marketing, customer segmentation, fraud detection, etc.

Proposal Template (Due: Today)

## Team Members:

## Project Title:

Dataset Chosen: (synthetic/real + name/source)

Goal: (What will you investigate? What questions are you trying to answer?)

Features of Interest: (e.g., income bracket, product types, spending levels)

## Plan to Handle Continuous Variables:

Proposed Rule Mining Approach: (Apriori / FP-Growth / other)

Expected Outcome: (What kind of patterns do you hope to discover?)

Why This Matters: (Real-world relevance – who would use this and how?)

Deliverables

- Proposal (due [Insert Date])

- Final report (3–5 pages)

- Python/R notebooks

- Data cleaning and processing steps

- Visuals of mined rules (graphs/tables)

- Interpretation + real-world insight

Guidance on Your Thought Process

Step 1: Find or Generate a Dataset

Option A: Find a Real Dataset

## Some recommended sources:

- UCI Machine Learning Repository

- Kaggle Datasets

- Data.gov

- OpenML.org

- Google Dataset Search

## Examples:

- Online Retail Transaction Data

- Grocery Store Sales (e.g., Instacart or Foodmart)

- Hospital Patient Records (de-identified)

- Smart Home Energy Usage

- Health or Fitness App Data

Option B: Generate a Synthetic Dataset

## Tools:

- Python’s faker or numpy.random

- Custom basket simulations (items, transaction IDs, timestamps)

Step 2: Preprocess the Data

## Tasks:

- Convert continuous variables to categorical (binning, quantiles, etc.)

- Handle missing values

- Normalize column names

- Discretize timestamps or IDs

- Remove duplicates/noise

## Example (Python):

import pandas as pd

df = pd.read\_csv("your\_data.csv")

df['age\_group'] = pd.qcut(df['age'], q=4, labels=['Young', 'Adult', 'Middle-Age', 'Senior'])

Step 3: Transform the Data for Mining

## Transaction format:

- Each row = transaction

- Each column = binary indicator

## Example (Python with mlxtend):

from mlxtend.preprocessing import TransactionEncoder

transactions = [['milk', 'bread'], ['bread', 'diaper', 'beer'], ['milk', 'diaper', 'bread']]

encoder = TransactionEncoder()

encoded = encoder.fit(transactions).transform(transactions)

df = pd.DataFrame(encoded, columns=encoder.columns\_)

Step 4: Apply Association Rule Mining

## Example:

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

frequent\_itemsets = apriori(df, min\_support=0.05, use\_colnames=True)

rules = association\_rules(frequent\_itemsets, metric="lift", min\_threshold=1.0)

print(rules[['antecedents', 'consequents', 'support', 'confidence', 'lift']])

Step 5: Analyze and Interpret Rules

- Identify the top 5 most interesting rules

- Use support, confidence, lift to evaluate

- Recommend an action based on findings

Step 6: Visualize Findings

## Suggested Tools:

- NetworkX graphs

- Heatmaps

- Scatter plots

## Example (NetworkX):

import networkx as nx

import matplotlib.pyplot as plt

G = nx.from\_pandas\_edgelist(rules, source='antecedents', target='consequents')

nx.draw(G, with\_labels=True)

plt.show()

Step 7: Real-World Applications

- Retail: Cross-selling, product bundling

- Healthcare: Co-occurrence of symptoms

- Finance: Suspicious activity detection

- Energy: Smart meter usage patterns