



LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY
A UGC Autonomous Institution | Approved by AICTE | Recognized by Govt. Of Telangana
Accredited by NBA& NAAC 'A' Grade | Included of 2F of UGC



Department of Computer Science And Engineering

Academic Year: 2026-2027

Student Roll Number:

160923733152

Student Name:

MOHAMMED UFRAAN

Year:

04

Semester:

07

Branch & Section:

CSE - A2
(c)

Course Name:

DATA MINING

Course Code:

U23CS706

Assignment Number:

01

Date of Assignment:

06/08/26

Due date:

12/08/26

Date of Submission:

Marks Obtained:

10

Remarks:

Signature of Faculty with date



Mohammed Ufraan
160923733152

LORDS INSTITUTE OF ENGINEERING & TECHNOLOGY
(UGC Autonomous)

Approved by AICTE | Affiliated to Osmania University | Estd.2003.
Department of Computer Science Engineering-

Name of the Subject: DATA MINING

Assignment – 1

Date : 06/08/26

Answer all the Questions.

CSE-C

Max.Marks:10

Q.No.	Question	MARKS	CO	BTL
1.	Illustrate the architecture of typical data mining system	2 M	CO1	BTL2
2.	Describe the steps involved in data mining. When viewed as a process of knowledge discovery with neat diagram.	2 M	CO1	BTL2
3	List and explain the different Types of Attributes?	2 M	CO1	BTL2
4	Compare the advantages of FP growth algorithm over Apriori algorithm.	2 M	CO2	BTL3
5.	List the different types of data-sets used for data analysis?	2 M	CO2	BTL2

Bloom's Taxonomy Levels (BTL)		
BTL1 –Remember	BTL2 Understand	BTL3 –Apply
BTL4-Analyze	BTL5 –Evaluate	BTL6 –Create

NOTE:

- 1.Answer all Questions.
- 2.Answer to the assignment questions are to be written on A4 size papers only.
- 3.Each page of the assignment paper should contain full roll number of the student.
- 4.The hard copy of the assignment is to be submitted on or before 12/08/26.

FACULTY
D.HARIKA

01.) Illustrate the architecture of typical data mining system.

Ans:

→ Data Sources:

- * Multiple heterogeneous data sources: relational databases, data warehouses, web data, sensor data, legacy systems.
- * Collect and aggregate raw data various origins.

→ Data Preprocessing Module:

- * Data cleaning: handle missing values, remove inconsistencies, treat outliers.
- * Data integration: combine data from multiple sources, resolve conflicts.
- * Data transformation: normalise, standardise, encode categorical variables.
- * Data reduction: remove irrelevant features, dimensionality reduction.

→ Data Warehouse:

- * Centralised repository of cleaned and integrated data.
- * Subject-oriented, integrated, time-variant, non-volatile storage.
- * Provides structured access to processed data.
- * Maintains metadata and historical information.

→ OLAP Server:

- * Online Analytical Processing.
- * Enables interactive exploration and multidimensional analysis.
- * Pre-mining exploratory analysis.

→ Data Mining Engine:

- * Core computational component implementing mining algorithms.
- * Supports multiple mining tasks: association rules, classification, clustering, regression, anomaly detection.
- * Discovers patterns, relationships, and knowledge from data.

→ Pattern Evaluation Module:

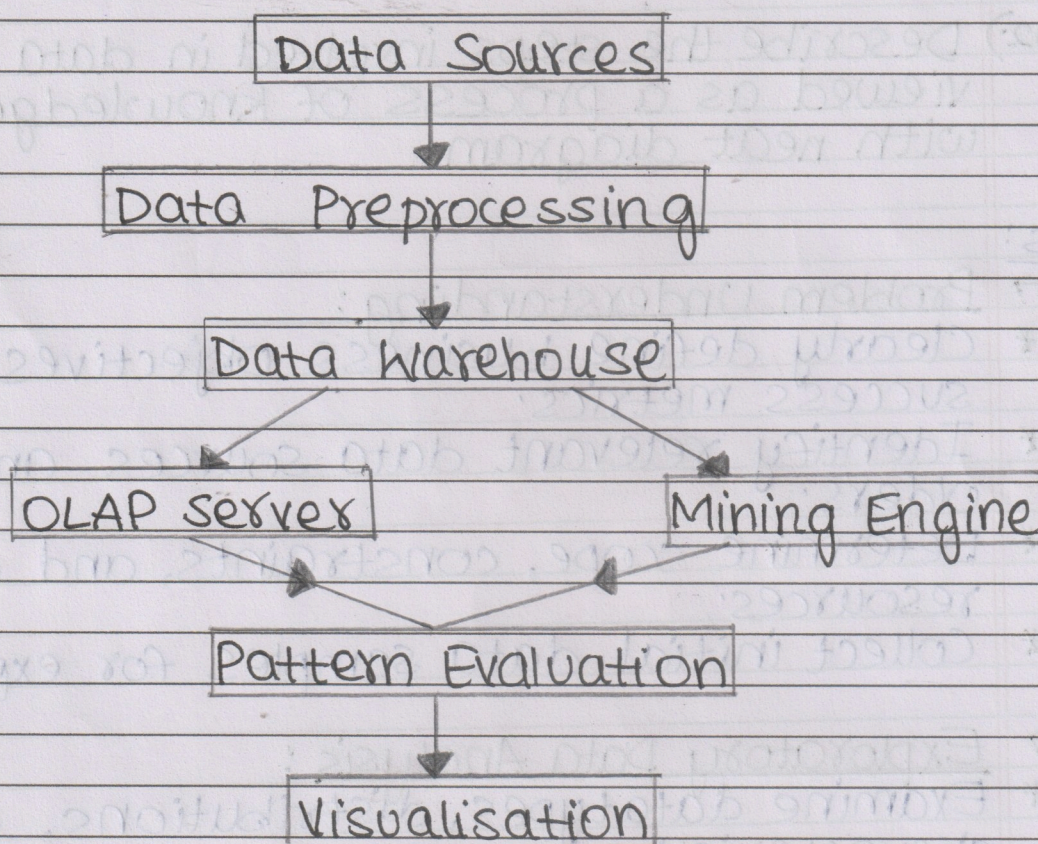
- * Assesses quality and significance of discovered patterns.
- * Uses interestingness measures: support, confidence, lift, chi-squared tests.
- * Filters redundant and insignificant patterns.
- * Validates patterns against domain knowledge.

→ Visualisation Layer:

- * Communicates findings to stakeholders using visual representations.
- * Charts, graphs, decision trees, heat maps, interactive dashboards.
- * Makes complex patterns understandable and actionable.

→ User Interface:

- * Allows users to specify mining parameters, constraints, and preferences.
- * Provides query interfaces for interactive exploration.
- * Domain-specific applications for targeted use cases.



Q2) Describe the steps involved in data mining. When viewed as a process of knowledge discovery with neat diagram.

Ans:

→ Problem Understanding:

- * Clearly define business objectives and success metrics.
- * Identify relevant data sources and stakeholders.
- * Determine scope, constraints, and available resources.
- * Collect initial data samples for exploration.

→ Exploratory Data Analysis:

- * Examine datatypes, distributions, and characteristics.
- * Identify missing values, outliers and anomalies.
- * Compute descriptive statistics.
- * Visualise data relationships using plots and charts.
- * Understand data quality issues.

→ Data Preprocessing:

- * Data Cleaning: handle missing values, correct errors, remove duplicates.
- * Data transformation: normalisation (0-1 range), standardisation (mean = 0), encoding categorical variables.
- * Data integration: combine multiple data sources, resolve naming conflicts.
- * Data reduction: remove irrelevant features, reduce dimensionality.

→ Model Development:

- * Define mining task type: classification, clustering, regression, association, anomaly detection.
- * Select appropriate algorithms for the task.
- * Split data into training (70%) and testing (30%) sets.
- * Train models and perform hyperparameter tuning.

→ Pattern Evaluation:

- * Evaluate model performance using appropriate metrics (accuracy, precision, recall, F1 score, RMSE).
- * Perform statistical significance testing.
- * Validate patterns against domain knowledge and business context.
- * Check for overfitting and generalisation capability.
- * Select best-performing model.

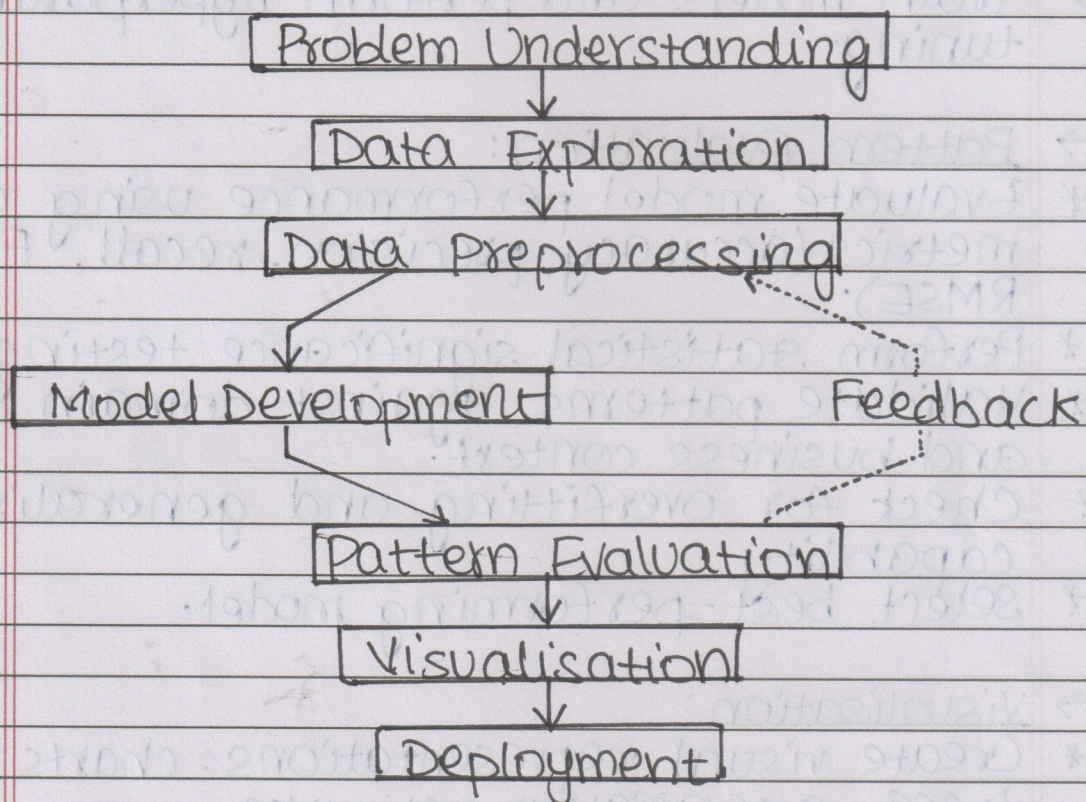
→ Visualisation:

- * Create visual representations: charts, decision trees, association networks.
- * Prepare business-focused reports and executive summaries.
- * Develop interactive dashboards for stakeholder exploration.
- * Present insights with actionable recommendations.

→ Deployment and Monitoring:

- * Deploy validated models into production environment.

- * Integrate with existing business systems and processes.
- * Set up performance monitoring and alerting systems.
- * Retrain models periodically as new data arrives.



Q3) List and explain the different types of attributes.

Ans: Attributes are classified into two main categories:

(A) Categorical (Qualitative) Attributes:

→ Nominal Attributes:

- * No inherent ordering between values.
- * Only equality/inequality comparisons are meaningful.
- * Values represent distinct categories or classes.
- * Example: Color (Red, Blue, Green), Gender (Male, Female), Country (USA, India, Japan).
- Distance measure: Hamming distance (count mismatches).
- * Suitable algorithms: Decision Trees, Naive Bayes, K-Means with categorical distance.

→ Ordinal Attributes:

- * Values have meaningful ordering/ranking.
- * Differences between values are not measurable or equal.
- * Comparison operations (greater, less) are meaningful.
- * Example: Education Level (High School < Bachelor < Master < PhD), Rating (Poor < Fair < Good < Excellent).
- * Distance measure: Ordinal distance = $|\text{rank}_i - \text{rank}_j|$.
- * Preserve ordering in preprocessing and analysis.

(B) Numeric (Quantitative) Attributes :

→ Internal Attributes :

- * Ordered with equal intervals between the consecutive values.
- * Meaningful differences but NO true zero point (zero is arbitrary).
- * Addition and subtraction are meaningful; multiplication/division are NOT.
- * Example: Temperature in Celcius (0°C $\neq$ no heat), Year (1990, 2000, 2020).
- * Preprocessing: standardisation (z-score), normalisation.

→ Ratio Attributes :

- * Ordered with meaningful ratios and a true zero point (zero = absence).
- * ALL arithmetic operations are valid and meaningful.
- * Differences AND ratios are meaningful.
- * Example: Height (0cm = no height), weight (0kg = no weight).
- * Most flexible type; supports all mathematical operations.

Type	Ordering	Arithmetic	Example
Nominal	No	No	Color
Ordinal	Yes	No	Rating
Interval	Yes	Yes	Temperature
Ratio	Yes	Yes	Height

04.) Compare the advantages of FP-growth algorithm over Apriori algorithm.

Ans:

→ Apriori algorithm:

- * Level wise iterative approach generating candidate itemsets.
- * Applies apriori principle: if itemset is frequent, all subsets are frequent.
- * Prunes candidates with support below minimum support threshold.
- * Time complexity: $O(2^I)$ in worst case.

→ FP Growth algorithm:

- * Pattern growth approach building Frequent Pattern (FP) tree.
Extracts frequent patterns directly from tree structure.
No candidate generation or testing phase.
Time complexity: $O(N \log N)$ to $O(N)$.

→ Advantages of FP growth:

- * No Candidate Generation:
FP-Growth eliminates expensive candidate generation phase, reducing computational overhead by 5-10x.
- * Fewer database scans:
Apriori requires $k+1$ scans for k itemsets; FP growth needs only 2 scans (one for 1 item-sets, one for tree construction).

- * Memory efficient:
FP tree compresses data through shared prefixes; typically uses 10-20% of database size vs. massive candidate sets in Apriori.
- * Execution Speed:
 $O(N \log N)$ complexity vs. $O(2^{|I|})$ for Apriori; 5-13x faster on typical datasets.
- * Better Scalability:
Suitable for large, dense datasets and big data applications; Apriori impractical for high-dimensional data.
- * Handles long Patterns:
Efficiently mines deep itemsets; Apriori deteriorates rapidly with pattern depth.

Performance Benchmark:
(100K transactions)

Min Support	Apriori	FP-Growth	Speedup
5%	45 sec	8 sec	5.6x
3%	120 sec	15 sec	8x
1%	380 sec	28 sec	13.6x

05.) List the different types of datasets used for data analysis.

Ans:

→ Relational (Tabular) Data:

- * Structured as 2D tables with rows and columns.
- * Each row represents one entity; each column represents one attribute.
- * Example: Customer database (ID, Name, Age, Income), CSV files.
- * Advantages: Well-organised, efficient querying, easy processing.
- * Challenges: Missing values, data normalisation required.

→ Transactional Data:

- * Records of transactions with variable-length itemsets.
- * Each transaction contains set of items purchased together.
- * Example: Market basket data (Milk, Bread, Butter), E-commerce purchases.
- * Sparse representation; support and confidence metrics apply.
- * Analysis: Association rule mining (Apriori, FP-Growth).

→ Graph/Network Data:

- * Nodes represent entities; edges represent relationships.
- * Can be directed (one-way) or undirected (bidirectional), weighted or unweighted.

- * Example: Social networks (users, friendships), Web graph (pages, hyper-links), Citation networks.
- * Analysis: Community detection, centrality analysis, link prediction.

→ Sequential/Time series Data:

- * Records ordered chronologically with temporal dependencies.
- * Values correlate with past values (autocorrelation); exhibits trends and seasonality.
- * Example: Stock prices, weather data, sensor readings, IoT data.
- * Requires time-specific handling (not i.i.d. assumption).

→ Textual Data:

- * Unstructured documents requiring natural language processing.
- * High-dimensional after vectorisation (bag-of-words, TF-IDF).
- * Example: News articles, customer reviews, tweets, emails.
- * Preprocessing: Tokenisation, stop-word removal, stemming, lemmatisation.
- * Analysis: Text classification, sentiment analysis, topic modeling, NER.

→ Multimedia Data:

- * Images, audio and video content.
- * Image: 2D pixel matrices (RGB, grayscale); Audio: sound waves; Video: frame sequences.
- * Example: Medical images (X-rays, CT scans), surveillance footage, music files.

- * Requires specialised feature extraction (CNNs for images, MFCCs for audio).
- * Analysis: Object detection, face recognition, activity recognition, emotion detection.

→ Streaming/online Data:

- * Continuous real-time data flow with potentially unbounded volume.
- * Data arrives sequentially; cannot load entire dataset in memory.
- * Subject to concept drift (data distribution changes over time).
- * Example: Sensor streams, social media feed, network logs, event streams.
- * Requires online learning and incremental processing approaches.

→ Spatial Data:

- * Geographic or location-based data with spatial dimensions.
- * Proximity and topology relationships are significant.
- * Example: GPS trajectories, satellite image, map data, location coordinates.
- * May combine with temporal data (spatio-temporal).
- * Analysis: Spatial clustering, trajectory mining, hotspot analysis, map pattern recognition.

→ Semi-structured Data:

- * Partial structure with embedded schemas/labels in data.

- * Neither fully structured (like relational) nor completely unstructured (like text).
- * Flexible schema; self-describing elements.
- * Example: XML, JSON files, HTML web pages, system logs.
- * Requires parsing before analysis; bridges structured and unstructured data.

Dataset Type	Structure	Key Analysis
Relational	Structured	SQL, Machine Learning
Transactional	Semi-structured	Association Rules
Graph	Network	Network analysis
Sequential	Temporal	Time Series
Textual	Unstructured	NLP
Multimedia	Unstructured	Deep Learning
Streaming	Unbounded	Stream Mining