Q.1 (a) State the need of business intelligence. [2 marks]

Business Intelligence (BI) is needed because:

- It enables data-driven decision making by transforming raw data into meaningful insights
- It helps organizations identify market trends and competitive advantages
- It supports strategic planning through historical, current, and predictive analysis
- It improves operational efficiency by identifying process bottlenecks
- It provides real-time monitoring of key performance indicators (KPIs)

Q.1 (c) Define Business Intelligence [2 marks]

Business Intelligence refers to the technologies, applications, practices, and processes that collect, integrate, analyze, and present business information to support better decision-making. It encompasses a range of tools and methodologies that transform raw data into meaningful and actionable insights to help organizations make more informed business decisions.

Q.1 (d) Differentiate between Supervised and Unsupervised learning with examples [2 marks]

Supervised Learning:

- Uses labeled training data with known output values
- Algorithm learns to map inputs to correct outputs
- Performance can be measured against known correct answers
- Examples: Decision trees, linear regression, support vector machines

Unsupervised Learning:

- Uses unlabeled data without predefined outputs
- Algorithm discovers patterns and relationships within data
- No explicit correct answers to measure against
- Examples: Clustering algorithms, association rules, dimensionality reduction

Q.1 (e) How are Classification and Regression different? [2 marks]

Classification:

- Predicts categorical (discrete) output labels or classes
- Output is a class membership (e.g., yes/no, spam/not spam)
- Evaluation metrics include accuracy, precision, recall, F1-score
- Example: Predicting whether an email is spam or not

Regression:

- Predicts continuous numerical values
- Output is a numeric value within a range
- Evaluation metrics include RMSE, MAE, R-squared
- Example: Predicting house prices or temperature

Q.1 (f) Explain types of Hierarchical Clustering algorithm [2 marks]

There are two main types of Hierarchical Clustering algorithms:

1. **Agglomerative (Bottom-up) Approach:**
    - Starts with each data point as a separate cluster
    - Progressively merges the closest clusters until only one remains
    - Uses linkage criteria (single, complete, average, Ward's) to determine cluster proximity
2. **Divisive (Top-down) Approach:**
    - Starts with all data points in one cluster
    - Recursively splits clusters until each data point is in its own cluster
    - Less commonly used but can be more accurate in some cases

Q.1 (g) How is Density-Based Clustering different from other approaches [2 marks]

Density-Based Clustering differs from other clustering approaches in several key ways:

1. **Shape Flexibility**: Unlike k-means which finds spherical clusters, density-based methods can discover clusters of arbitrary shapes by connecting dense regions.
2. **No Predefined Cluster Count**: Does not require specifying the number of clusters beforehand, unlike k-means or k-medoids.
3. **Noise Handling**: Explicitly identifies outliers or noise points that don't belong to any cluster, which partitioning methods like k-means cannot do.
4. **Density Definition**: Forms clusters based on dense regions of points separated by sparse regions, whereas hierarchical methods use distance-based merging or splitting.
5. **Parameter Focus**: Uses density parameters (epsilon and minPoints) rather than cluster count, making it suitable for datasets where the number of natural groupings is unknown.

Q.2 (a) Create a Decision Tree for the dataset [5 marks]

For the given dataset:

| Attribute 1 | Attribute 2 | Class Label |

|------------|------------|------------|

| High | Low | Yes |

| Low | Medium | No |

| Medium | High | Yes |

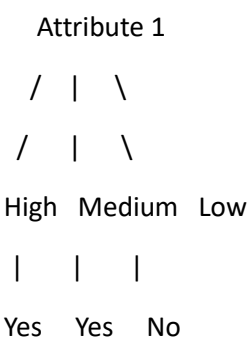
| Low | Low | No |

| High | High | Yes |

Decision Tree Construction:

1. Root node: Best attribute to split on is Attribute 1
    - Information gain calculation shows Attribute 1 is more decisive
2. For Attribute 1 = "High": 2/2 are "Yes" → Leaf node: "Yes"
3. For Attribute 1 = "Medium": 1/1 is "Yes" → Leaf node: "Yes"
4. For Attribute 1 = "Low": 2/2 are "No" → Leaf node: "No"

The resulting decision tree is:



This simple tree perfectly classifies the given dataset, with Attribute 1 being the sole deciding feature.

Q.2 (b) Explain the accuracy and coverage in Rule based Classifier. Calculate the same for the given table. [5 marks]

In rule-based classification:

Accuracy: The probability that a rule correctly classifies a random sample satisfying the rule's precondition. It measures the reliability of the prediction.

- Formula: Accuracy = Number of correct predictions / Total number of cases where rule applies

Coverage: The proportion of records in the dataset that satisfy the rule's condition. It measures how widely applicable the rule is.

- Formula: Coverage = Number of records that satisfy rule condition / Total number of records

For the given data table:

Customer ID	Gender	Age	Purchases Last Month	Purchase Prediction
1	Male	30	2	No
2	Female	25	4	Yes
3	Male	40	1	No
4	Female	30	5	Yes
5	Female	35	3	Yes
6	Male	28	0	No
7	Female	22	2	No
8	Male	45	3	Yes
9	Female	32	6	Yes
10	Male	38	4	Yes

Let's evaluate some potential rules:

Rule 1: IF Purchases > 3 THEN Prediction = Yes

- Applies to: IDs 2, 4, 5, 8, 9, 10 (6 records)
- Correct predictions: IDs 2, 4, 5, 9, 10 (5 records)
- Accuracy = 5/6 = 0.833 or 83.3%
- Coverage = 6/10 = 0.6 or 60%

Rule 2: IF Gender = Female THEN Prediction = Yes

- Applies to: IDs 2, 4, 5, 7, 9 (5 records)
- Correct predictions: IDs 2, 4, 5, 9 (4 records)
- Accuracy = 4/5 = 0.8 or 80%
- Coverage = 5/10 = 0.5 or 50%

Rule 3: IF Purchases ≤ 2 THEN Prediction = No

- Applies to: IDs 1, 3, 6, 7 (4 records)
- Correct predictions: IDs 1, 3, 6, 7 (4 records)

- Accuracy = 4/4 = 1.0 or 100%
- Coverage = 4/10 = 0.4 or 40%

The analysis shows Rule 3 has perfect accuracy but lower coverage, while Rule 1 has high accuracy and better coverage, making it likely the better overall classifier for this dataset.

Q.3 (a) Create Clusters using DBSCAN Algorithm [5 marks]

For the DBSCAN algorithm with MinPoints = 4 and Epsilon = 1.9, I'll use the distance matrix provided to identify clusters:

Step 1: Identify core points (points with at least MinPoints = 4 neighbors within Epsilon = 1.9)

- P1: Not a core point (less than 4 neighbors within 1.9)
- P2: Not a core point
- P3: Core point (P2, P4, P5, P7 within 1.9)
- P4: Core point
- P5: Core point
- P6: Not a core point
- P7: Core point

Step 2: Connect core points to form clusters

- Cluster 1: {P3, P4, P5, P7}

Step 3: Assign border points to clusters

- P2 is a border point for Cluster 1 (within epsilon of P3)
- P6 is not within epsilon of any core point

Step 4: Identify noise points

- P1 and P6 are noise points (not core points and not within epsilon of any core point)

Final clusters:

- Cluster 1: {P2, P3, P4, P5, P7}
- Noise: {P1, P6}

Q.3 (b) Explain the concept of Dendrogram in Hierarchical Clustering [5 marks]

A dendrogram is a tree-like diagram that visualizes the hierarchical relationship between clusters in hierarchical clustering. Key aspects include:

1. **Structure**: A tree-like visualization where the y-axis represents the distance or dissimilarity between clusters, and the x-axis represents the individual data points or clusters.

2. **Interpretation**: Each branch point (node) represents a cluster merger, with the height indicating the dissimilarity level at which clusters were combined.
3. **Benefits**:
    - Provides a complete picture of the clustering process at all levels
    - Helps determine the optimal number of clusters by observing where the longest vertical lines occur
    - Preserves the hierarchical relationships between data points
4. **Reading the Dendrogram**:
    - The lower in the tree a merger occurs, the more similar the clusters
    - Cutting the dendrogram horizontally at different heights yields different cluster configurations
    - The height of vertical lines indicates the dissimilarity between merged clusters
5. **Applications**:
    - Taxonomic classification in biology
    - Document clustering
    - Customer segmentation
    - Gene expression analysis

Dendrograms are particularly valuable for understanding the natural grouping structure in data and for determining an appropriate number of clusters when this is not known in advance.