CBCS SCHEME

USN | | | | | | | | | | |

21IS643

Sixth Semester B.E. Degree Examination, June/July 2024
Data Mining and Data Warehousing

Time: 3 hrs. Max. Marks: 100

Note: *Answer any FIVE full questions, choosing ONE full question from each module.*

Important Note : 1. On completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages.
2. Any revealing of identification, appeal to evaluator and /or equations written eg, 42+8 = 50, will be treated as malpractice.

Module-1

1	a.	Differentiate Operational Database system and Data ware house.	(08 Marks)
	b.	List Data Warehouse characteristics.	(04 Marks)
	c.	With a neat diagram, explain a Three - Tier Architecture of Data Warehouse.	(08 Marks)
		OR	
2	a.	Explain the schemas of Multidimensional Data models.	(08 Marks)
	b.	Explain OLAP Operations	(06 Marks)
	c.	Differentiate OLAP and OLTP.	(06 Marks)

Module-2

3	a.	What is Data Mining? Explain KDD process in Data Mining.	(10 Marks)
	b.	What kinds of pattern can be mined? Explain in brief.	(10 Marks)
		OR	
4	a.	Explain the Data Preprocessing techniques in brief.	(10 Marks)
	b.	Explain the Dimensionality Reduction with its significance.	(10 Marks)

Module-3

5	a.	Explain the Apriori Algorithm with an example.	(10 Marks)
	b.	Explain the Frequent Pattern growth algorithm and mention its advantages.	(10 Marks)
		OR	
6	a.	Explain Alternative methods for generating frequency item sets in brief.	(10 Marks)
	b.	Explain Evaluation of Association Pattern in brief.	(10 Marks)

Module-4

7	a.	How does Decision tree induction algorithm works? Explain with an example.	(10 Marks)
	b.	Describe the methods for comparing classifiers.	(10 Marks)
		OR	
8	a.	Explain Direct methods and Indirect methods of Rule Extraction in brief.	(10 Marks)
	b.	Explain Nearest Neighbour Classifier. List its characteristics.	(10 Marks)

Module-5

9	a.	Describe K means Clustering algorithm. What are its limitations?	(10 Marks)
	b.	Explain DBSCAN algorithm with an example.	(10 Marks)
		OR	
10	a.	Explain the following in brief : i) Density Based Clustering ii) Graph Based Clustering.	(10 Marks)
	b.	Explain the BRICH Scalable Algorithm.	(10 Marks)

* * * * *

USN										

VTU Examination – JUN/JULY 2024
Scheme of Evaluation

Sub:	Data warehousing and Data Mining					Sub Code:	21IS643	Branch:	ISE
Date:	/08/2024	Duration:	3 hrs	Max Marks:	100	Sem/Sec:	VI/ C		OBE

Answer any FIVE FULL Questions | MARKS | CO | RBT

MODULE -1

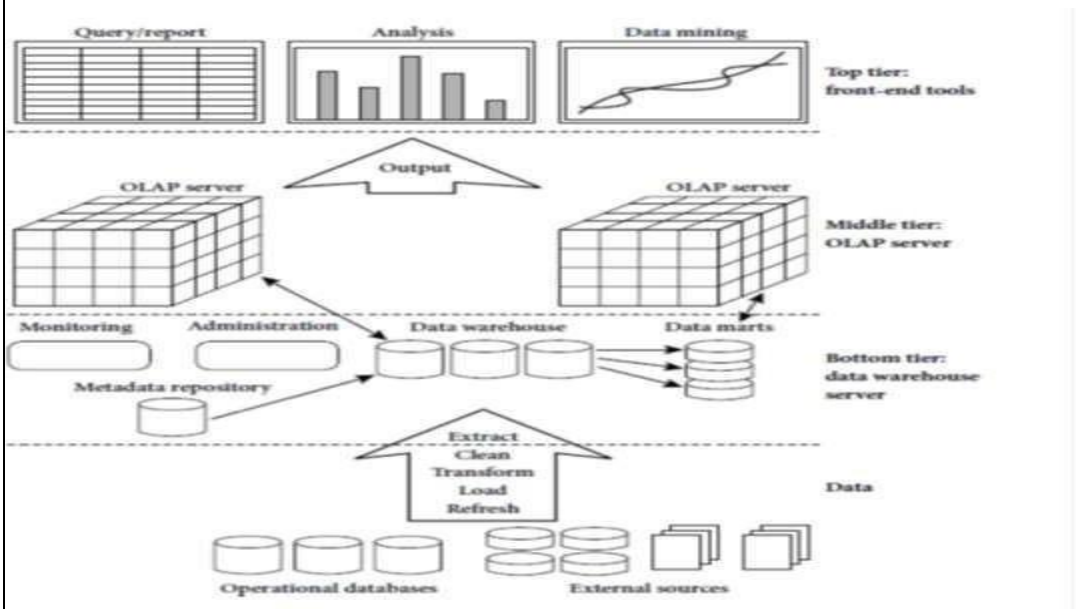
1. **a. Differentiate operational database system and data warehouse** | 8 | 1 | L2

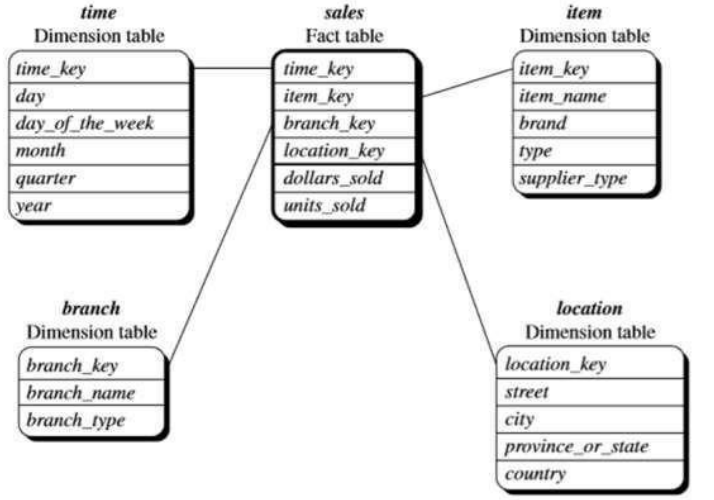
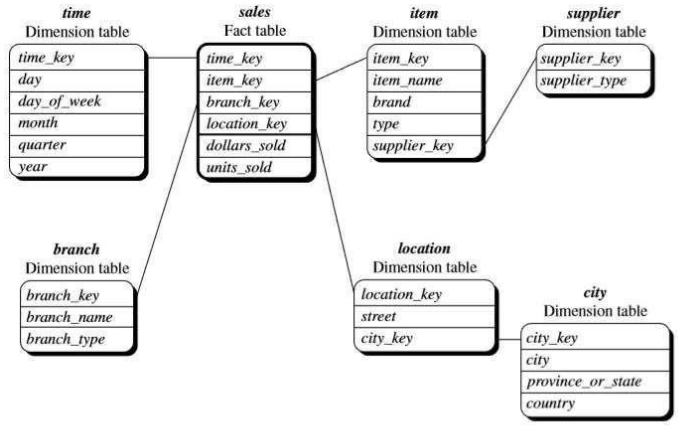
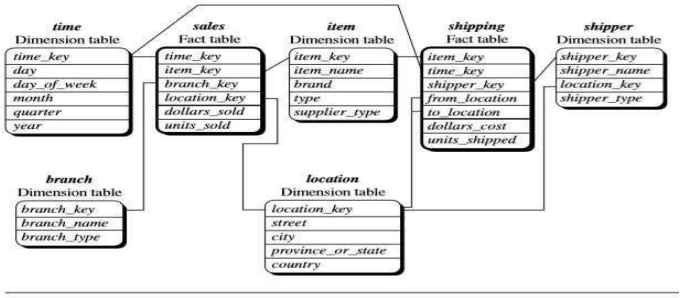
Scheme: Differences between ODS and Data warehouse ---4+4marks

Solution: The major task of online operational database systems is to perform online transaction and query processing. These systems are called **online transaction processing (OLTP)** systems. They cover most of the day-to-day operations of an organization such as purchasing, inventory, manufacturing, banking, payroll, registration, and accounting.

Data warehouse systems, on the other hand, serve users or knowledge workers in the role of data analysis and decision-making. Such systems can organize and present data in various formats in order to accommodate the diverse needs of different users. These systems are known as **online analytical processing(OLAP**) systems. The major distinguishing features of OLTP and OLAP are summarized as follows:

Operational Database	Data Warehouse
Operational frameworks are outlined to back high-volume exchange preparing.	Data warehousing frameworks are regularly outlined to back high-volume analytical processing (i.e., OLAP).
operational frameworks are more often than not concerned with current data.	Data warehousing frameworks are ordinarily concerned with verifiable information.
Data inside operational frameworks are basically overhauled frequently agreeing to need.	Non-volatile, unused information may be included routinely. Once Included once in a while changed.
It is planned for real-time commerce managing and processes.	It is outlined for investigation of commerce measures by subject range, categories, and qualities.
Relational databases are made for on-line value-based Preparing (OLTP)	Data Warehouse planned for on-line Analytical Processing (OLAP)
Operational frameworks are ordinarily optimized to perform quick embeds and overhauls of cooperatively little volumes of data.	Data warehousing frameworks are more often than not optimized to perform quick recoveries of moderately tall volumes of information.

1	**b. List Data warehouse characteristics** **Scheme : characteristics to be listed ---4 Marks** **1. Solution : Subject-Oriented**: A data warehouse is organized around major subjects such as cus- tomer, supplier, product, and sales A data warehouse can be used to analyze a particular subject area. For example, "sales" can be a particular subject. **2. Integrated**: A data warehouse integrates data from multiple data sources. For example, source A and source B may have different ways of identifying a product, but in a data warehouse, there will be onlya single way of identifying a product. **3. Time-Variant**: Historical data is kept in a data warehouse. For example, one can retrieve data from 3 months, 6 months, 12 months, or even older data from a data warehouse. This contrasts with a transactions system, where often only the most recent data is kept. For example, a transaction system may hold the most recent address of a customer, where a data warehouse can hold all addresses associated with a customer. **4. Non-volatile**: Once data is in the data warehouse, it will not change. So, historical data in a data warehouse should never be altered.	4	1	L2
1	**C. With neat diagram ,Explain Three Tier Architecture of Data warehouse** **Scheme** : Diagram + Explanation -----2+6 marks **Solution :**  **Tier-1** **A Three Tier Data Warehouse Architecture:** The bottom tier is a warehouse database server that is almost always a relational database system. Back- end tools and utilities are used to feed data into the bottom tier from operational databases or other external sources (such as customer profile information provided by external consultants). ***Tier-2:*** The middle tier is an OLAP server that is typically implemented using either a relational OLAP (ROLAP)model or a multidimensional OLAP. OLAP model is an extended relational DBMS that maps operations on multidimensional data tostandard relational operations. A multidimensional OLAP (MOLAP) model, that is, a special-purpose server that directlyimplements multidimensional data and operations.	8	1	L2

2 a.	***Tier-3:*** The top tier is a front-end client layer, which contains query and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on). **(OR)** **a.Explain the Schemas of Multidimensional Model** **Scheme** : schemas +Diagram +Explanation **Solution :** Star schema: A fact table in the middle connected to a set of dimension tables Snowflake schema: Arefinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake Fact constellations: Multiple fact tables share dimension tables, viewed as a collection of stars, thereforecalled galaxy schema or fact constellation.  Star schema of *sales* data warehouse.  Snowflake schema of a *sales* data warehouse.  Fact constellation schema of a sales and shipping data warehouse. **2 b.Explain OLAP Operations**	8	1	L2

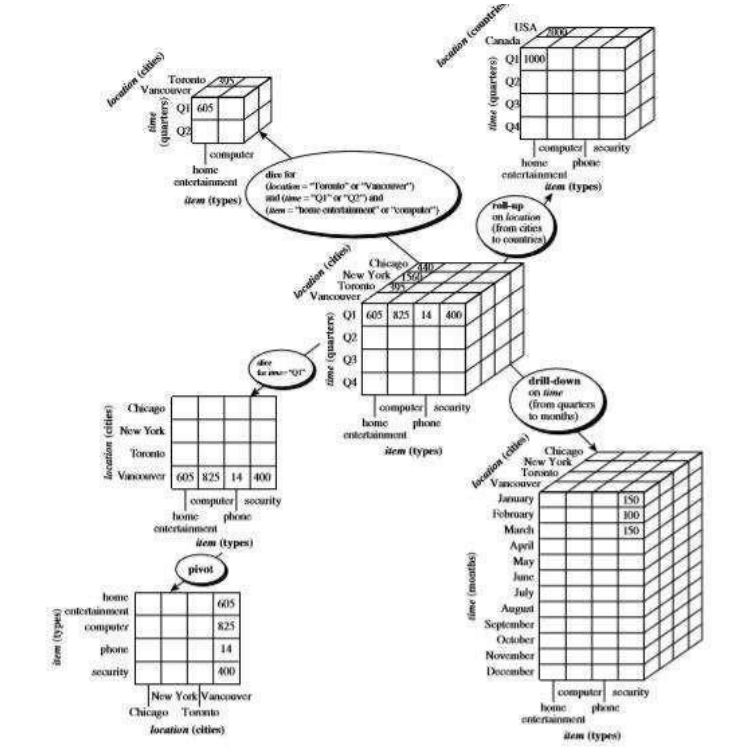
	Scheme : Different set of OLPA operation with diagram---------2+4 Marks **Solution : Typical OLAP Operations** **ROLL-UP** This is like zooming-out on the data-cube This is required when the user needs further abstraction or less detail. • Initially, the location-hierarchy was "street < city < province < country". • On rolling up, the data is aggregated by ascending the location-hierarchy from the level-of city to level-of- country. **DRILL DOWN** This is like zooming-in on the data. This is the reverse of roll-up. • This is an appropriate operation → when the user needs further details or → when the user wants to partition more finely or → when the user wants to focus on some particular values of certain dimensions. • This adds more details to the data. • Initially, the time-hierarchy was "day < month < quarter < year". • On drill-up, the time dimension is descended from the level-of-quarter to the level-of-month. **PIVOT (OR ROTATE)** This is used when the user wishes to re-orient the view of the data-cube. This may involve → swapping the rows and columns or → moving one of the row-dimensions into the column-dimension. **SLICE & DICE** These are operations for browsing the data in the cube. • These operations allow ability to look at information from different viewpoints. • A slice is a subset of cube corresponding to a single value for 1or more members of dimensions.. A dice operation is done by performing a selection of 2 or more dimensions.  **Figure 4.12** Examples of typical OLAP operations on multidimensional data.	6	1	L2
	c. Differentiate OLAP and OLTP **Scheme : Difference between OLTP and OLAP ---3+3 marks** **Solution :**	6	1	L2

Table 4.1 Comparison of OLTP and OLAP Systems

Feature	*OLTP*	*OLAP*
Characteristic	operational processing	informational processing
Orientation	transaction	analysis
User	clerk, DBA, database professional	knowledge worker (e.g., manager, executive, analyst)
Function	day-to-day operations	long-term informational requirements decision support
DB design	ER-based, application-oriented	star/snowflake, subject-oriented
Data	current, guaranteed up-to-date	historic, accuracy maintained over time
Summarization	primitive, highly detailed	summarized, consolidated
View	detailed, flat relational	summarized, multidimensional
Unit of work	short, simple transaction	complex query
Access	read/write	mostly read
Focus	data in	information out
Operations	index/hash on primary key	lots of scans
Number of records accessed	tens	millions
Number of users	thousands	hundreds
DB size	GB to high-order GB	≥ TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

Module 2:

3 a. **What is data mining ,Explain the KDD Process in Data mining**

Scheme : Def'n of Data mining + KDD Steps and its Associated meaning. ---- 2+4+4 marks

- **Solution :** Data mining is the process of automatically discovering useful information in large data repositories. Finding hidden information in a database
- Data mining techniques are deployed to scour large databases in order to find novel and useful patterns that might otherwise remain unknown. They also provide capabilities to predict e outcome of a future observation.

10 | 2 | L2

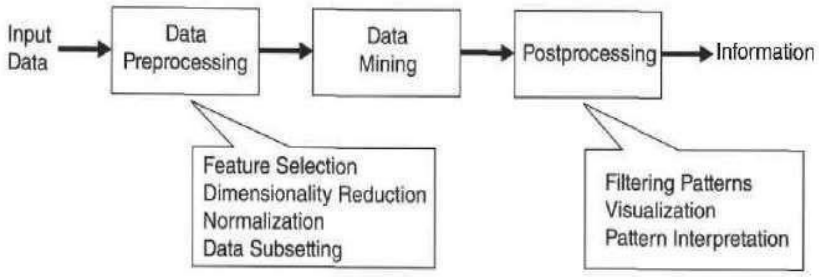


Figure 1.1. The process of knowledge discovery in databases (KDD).

3 b. **What kind of patterns can be mined, Explain in brief.**

Scheme: The types of pattern and their explanation----------4+6 marks

Solution :

- These patterns can be broadly categorized into two main groups, each offering valuable insights:
- **Descriptive Patterns**
- **Predictive Patterns**
- Types of Descriptive patterns are :
  Data Characterization and Data Discrimination.
  Types of Predictive patterns are : Classification and regression and elaborate on the points.

10 | 2 | L2

(OR)

4.a. **Explain Data preprocessing technique in brief**

Scheme : The different steps of preprocessing ------- 5+5 Marks

Solution : Preprocessing steps should be applied to make the data more suitable for data mining

10 | 2 | L2

The most important ideas and approaches are

- Aggregation
- Sampling
- Dimensionality Reduction
- Feature subset selection
- Feature creation
- Discretization and Binarization
- Attribute Transformation

Aggregation

- ✓ Combining two or more attributes (or objects) into a single attribute (or object) Purpose:
  - Data reduction
  - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc
  - More ―stable‖ data
  - Aggregated data tends to have less variability.

Sampling

- ✓ Sampling is the main technique employed for data selection.
- ✓ It is often used for both the preliminary investigation of the data and the final data analysis.
- ✓ Statisticians sample because obtaining the entire set of data of interest is too expensive or time consuming.
- ✓ Sampling is used in data mining because processing the entire set of data of interest is too expensive or time consuming.

Types of Sampling

- ✓ Simple Random Sampling

There is an equal probability of selecting any particular item

- ✓ Sampling without replacement

As each item is selected, it is removed from the population

- ✓ Sampling with replacement

Objects are not removed from the population as they are selected for the sample. In sampling with replacement, the same object can be picked up more than once

- ✓ Stratified sampling

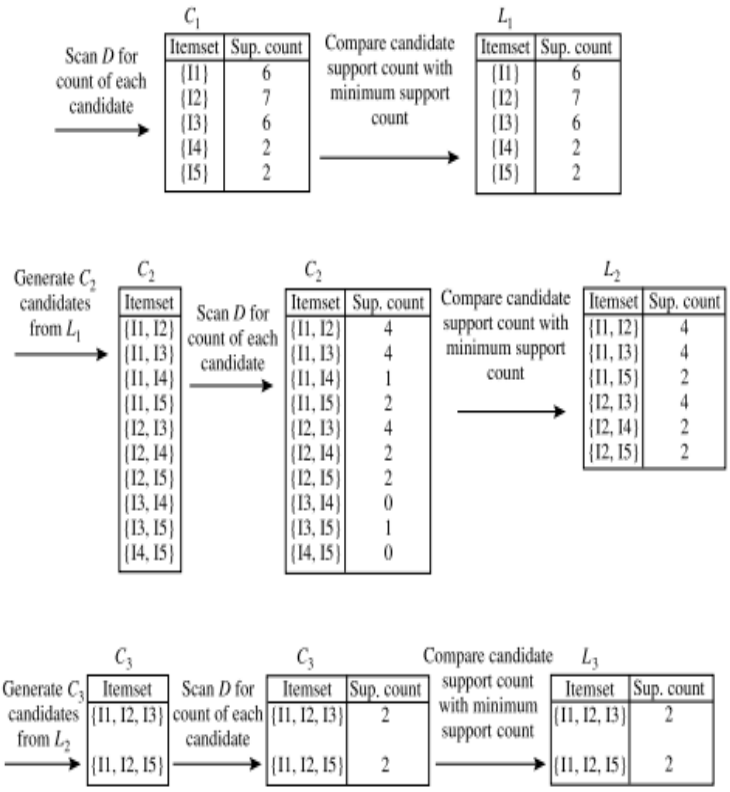
Split the data into several partitions; then draw random samples from each partition.

4 b	**Explain the Dimensionality reduction with example** **Scheme :** Def'n of Dimensionality reduction and explanation with example----------4+6 marks **Solution :** To Avoid curse of dimensionality Reduce amount of time and memory required by data mining algorithms. Allow data to be more easily visualized May help to eliminate irrelevant features or reduce noise The Curse of Dimensionality: the curse of dimensionality refers to the phenomenon that many types of data analysis become significantly harder as the dimensionality of the data increases. Specifically, as dimensionality increases, the data becomes increasingly sparse in the space that it occupies.	10	2	L2
	Module 3			

5 a.	**Explain a Priori algorithm with example**
	Scheme : Algorithm + Example ---------- 5+5 marks
	Solution :
	Algorithm 3: Apriori algorithm
	F_1 = {frequent items of size 1};
	for (k = 1; F_k !=ϕ; k++) do begin
	C_{k+1} = apriori-gen (F_k); // New candidates generated from F_k
	for all transactions t in database do begin
	C'_t = subset (C_{k+1},t); // Candidates contained in t
	for all candidate $c \in C'_t$ do
	$c.count$++; // Increment the count of all candidates in C_{k+1} that are contained in t
	end
	F_{k+1} = {$C \in C_{k+1} \mid c.count \geq$ minimum suport}
	//Candidates in C_{k+1} with minimum support
	end
	end
	Answer $\bigcup_k F_k$;
		10	3	L3

Transactional Data for an *AllElectronics* Branch

TID	*List of item_IDs*
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3



5 b. **Explain the frequent pattern growth algorithm and mention its advantages.**

Scheme : Algorithm +advantages----------7+3 marks

Solution : Use a compressed representation of the database using an FP-tree.

- It allows frequent itemset discovery without candidate generation.
- Once an FP-tree has been constructed, it uses a recursive divide-and-conquer approach to mine the frequent itemsets.
- Two step:
- 1.Build a compact data structure called the FP-tree
- 2 passes over the database
- 2.extracts frequent itemsets directly from the FP-tree
- Traverse through FP-tree
- Nodes correspond to items and have a counter
- FP-Growth reads 1 transaction at a time and maps it to a path
- Fixed order is used, so paths can overlap when transactions share items (when they have the same prex).
- In this case, counters are incremented
- Pointers are maintained between nodes containing the same item, creating singly linked lists (dotted lines)
- The more paths that overlap, the higher the compression.
- Frequent itemsets extracted from the FP-Tree.

10	3	L3

(OR)

6 a. **Explain the alternative method of generating the frequent item sets in brief**

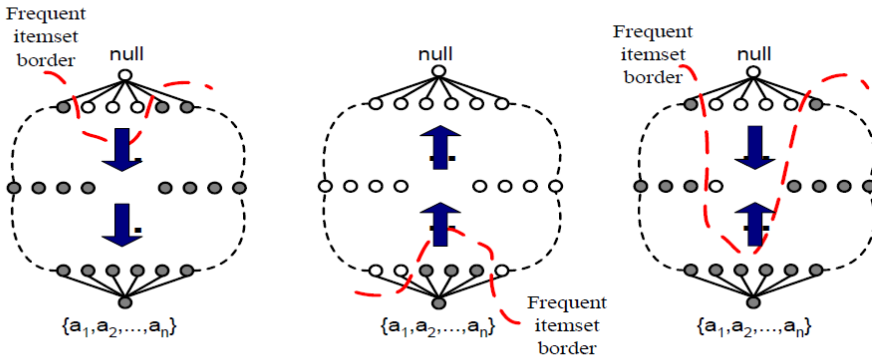
Scheme : Different methods + Explanation about the methods -----4+6marks

Solution :

10	3	L3

Traversal of Itemset Lattice

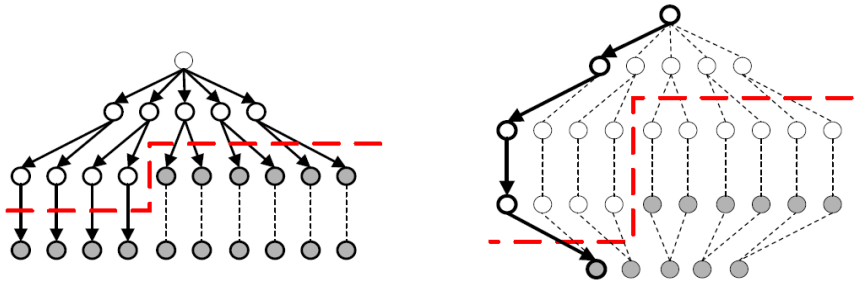
- General-to-specific vs Specific-to-general



(a) General-to-specific (b) Specific-to-general (c) Bidirectional

Traversal of Itemset Lattice

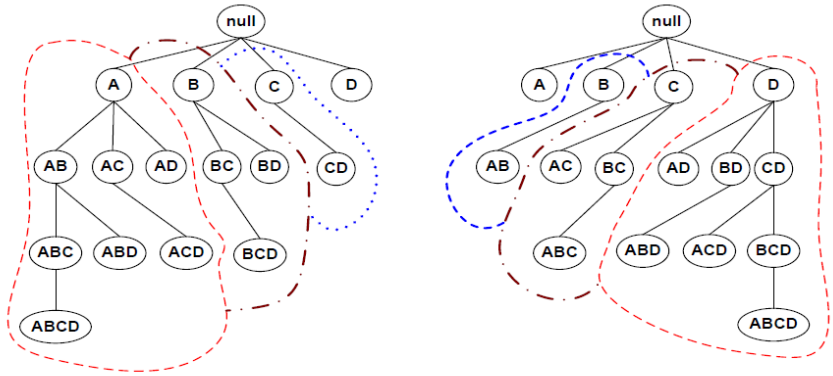
- Breadth-first vs Depth-first



(a) Breadth first (b) Depth first

Traversal of Itemset Lattice

- Equivalent Classes



(a) Prefix tree (b) Suffix tree

6.b.	**Explain Evaluation of Association pattern in brief** **Scheme :** Different methods of evaluation **Solution :**	10	3	L3

- Association rule algorithms tend to produce too many rules
- – many of them are uninteresting or redundant
- – Redundant if {A,B,C} –> {D} and {A,B} –> {D} have same support & confidence

- Interestingness measures can be used to prune/rank the derived patterns

Computing Interestingness Measure

Given X → Y or {X,Y}, information needed to compute interestingness can be obtained from a contingency table

Contingency table

	Y	$\overline{Y}$	
X	f_{11}	f_{10}	f_{1+}
$\overline{X}$	f_{01}	f_{00}	f_{o+}
	f_{+1}	f_{+0}	N

f_{11}: support of X and Y
f_{10}: support of X and $\overline{Y}$
f_{01}: support of $\overline{X}$ and Y
f_{00}: support of $\overline{X}$ and $\overline{Y}$

Used to define various measures

- support, confidence, Gini, entropy, etc.

Interest Factor

$$I(A,B) = \frac{s(A,B)}{s(A) \times s(B)} = \frac{N f_{11}}{f_{1+} f_{+1}}.$$

$$I(A,B) \begin{cases} = 1, & \text{if } A \text{ and } B \text{ are independent;} \\ > 1, & \text{if } A \text{ and } B \text{ are positively correlated;} \\ < 1, & \text{if } A \text{ and } B \text{ are negatively correlated.} \end{cases}$$

7 a.	**Module 4** **How does decision tree induction algorithm works? Explain with an example** **Scheme :** Decision tree Induction algorithm + example **Solution :**	**10**	**4**	**L3**

Algorithm 4.1 A skeleton decision tree induction algorithm.

```
TreeGrowth (E, F)
 1: if stopping_cond(E,F) = true then
 2:    leaf = createNode().
 3:    leaf.label = Classify(E).
 4:    return leaf.
 5: else
 6:    root = createNode().
 7:    root.test_cond = find_best_split(E, F).
 8:    let V = {v|v is a possible outcome of root.test_cond }.
 9:    for each v ∈ V do
10:       E_v = {e | root.test_cond(e) = v and e ∈ E}.
11:       child = TreeGrowth(E_v, F).
12:       add child as descendent of root and label the edge (root → child) as v.
13:    end for
14: end if
15: return root.
```

Entropy at a given node t:

$$Entropy(t) = -\sum_j p(j \mid t) \log p(j \mid t)$$

(NOTE: $p(j \mid t)$ is the relative frequency of class j at node t).

- Maximum (log n_c) when records are equally distributed among all classes implying least information
- Minimum (0.0) when all records belong to one class, implying most information

Computing Entropy of a Single Node

$$Entropy(t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Entropy = – 0 log 0 – 1 log 1 = – 0 – 0 = 0

C1	**1**
C2	**5**

P(C1) = 1/6 P(C2) = 5/6

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (1/6) = 0.65

C1	**2**
C2	**4**

P(C1) = 2/6 P(C2) = 4/6

Entropy = – (2/6) $\log_2$ (2/6) – (4/6) $\log_2$ (4/6) = 0.92

7 b.	**Describe the method for comparing the classifiers** Scheme : Different method + explanation -------4+6 marks Solution :			
	For illustrative purposes, consider a pair of classification models, *MA* and *MB*. - *MA* achieves 85% accuracy when evaluated on a test set containing 30	**10**	**4**	**L3**

	records, - *MB* achieves 75% accuracy on a different test set containing 5000 records. Based on this information, is *MA* a better model than *MB*? Two key issues regarding the statistical significance of the performance metrics: 1.The first question relates to the issue of estimating the confidence interval of a given model accuracy. 2.The second question relates to the issue of testing the statistical significance of the observed deviation. Estimating a Confidence Interval for Accuracy To determine the confidence interval, we need to establish the probability distribution that governs the accuracy measure. Given a test set that contains *N* records, Comparing the Performance of Two Classifiers Suppose we want to compare the performance of two classifiers using the *k*-fold cross-validation approach. The data set *D* is divided into *k* equal-sized partitions. Then apply each classifier to construct a model from *k* - 1 of the partitions and test it on the remaining partition. This step is repeated *k* times, each time using a different partition as the test set. Let - *Mij* denote the model induced by classification technique *Li* during the *jth* iteration. - each pair of models *M*1*j* and *M*2*j* are tested on the same partition *j*. - Let *e*1*j* and *e*2*j* be their respective error rates. - The difference between their error rates during the *jth* fold can be written as *dj* = *e*1*j* - *e*2*j*.			
8 a.	**(OR)** **Explain direct and indirect method of rule extractions in brief** **Scheme :** Direct method +indirect method of rule extraction--------5+5 marks **Solution :** Direct Methods for Rule Extraction The sequential covering algorithm is often used to extract rules directly from data. The algorithm extracts the rules one class at a time for data sets that contain more than two classes. For the vertebrate classification problem, the sequential covering algorithm may generate rules for classifying birds first, followed by rules for classifying mammals, amphibians, reptiles, and finally, fishes. The criterion for deciding which class should be generated first depends on a number of factors, such as - the class prevalence (i.e., fraction of training records that belong to a particular class) or - the cost of misclassifying records from a given class. A summary of the sequential covering algorithm is given in Algorithm .	**10**	**4**	**L3**

Algorithm 5.1 Sequential covering algorithm.

1: Let E be the training records and A be the set of attribute-value pairs, $\{(A_j, v_j)\}$.
2: Let Y_o be an ordered set of classes $\{y_1, y_2, \ldots, y_k\}$.
3: Let $R = \{\ \}$ be the initial rule list.
4: **for** each class $y \in Y_o - \{y_k\}$ **do**
5: **while** stopping condition is not met **do**
6: $r \leftarrow$ Learn-One-Rule (E, A, y).
7: Remove training records from E that are covered by r.
8: Add r to the bottom of the rule list: $R \longrightarrow R \vee r$.
9: **end while**
10: **end for**
11: Insert the default rule, $\{\} \longrightarrow y_k$, to the bottom of the rule list R.

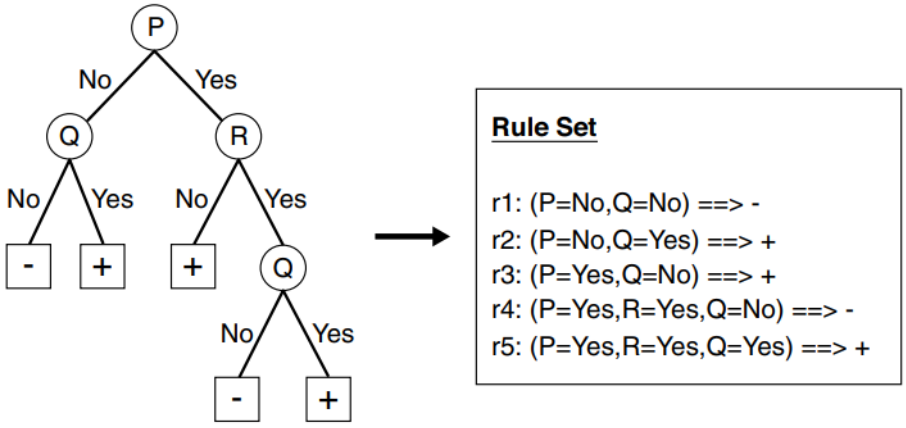
Indirect Methods for Rule Extraction

A method for generating a rule set from a decision tree.

Every path from the root node to the leaf node of a decision tree can be expressed as a classification rule.

The test conditions encountered along the path form the conjuncts of the rule antecedent, while the class label at the leaf node is assigned to the rule consequent.

Figure below shows an example of a rule set generated from a decision tree.



8.b

Explain Nearest Neigbhour classifier , List its characteristics

Scheme: Classifier algorithm + Characteristics--------6+4 marks

Solution :

10	**4**	**L3**

1: Let k be the number of nearest neighbors and D be the set of training examples.
2: **for** each test example $z = (\mathbf{x}', y')$ **do**
3: Compute $d(\mathbf{x}', \mathbf{x})$, the distance between z and every example, $(\mathbf{x}, y) \in D$.
4: Select $D_z \subseteq D$, the set of k closest training examples to z.
5: $y' = \underset{v}{\text{argmax}} \sum_{(\mathbf{x}_i, y_i) \in D_z} I(v = y_i)$
6: **end for**

Characteristics of Nearest-Neighbor Classifiers

The characteristics of the nearest-neighbor classifier are summarized below:

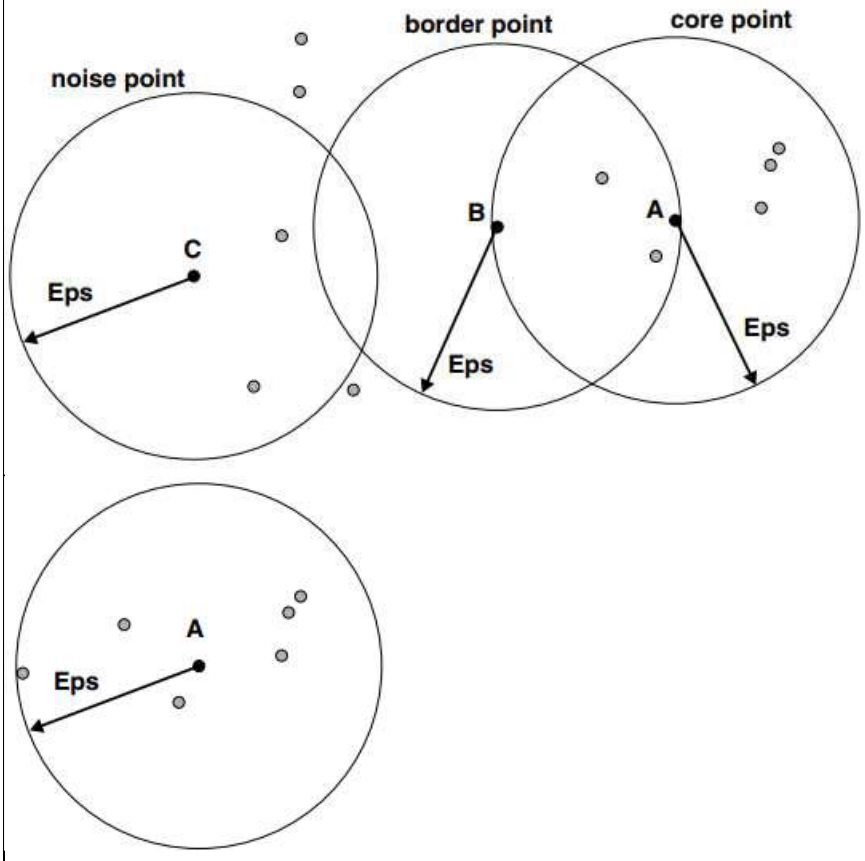
• Nearest-neighbor classification is part of a more general technique known as instance-based learning, which uses specific training instances to make predictions without having to maintain an abstraction (or model) derived from data.

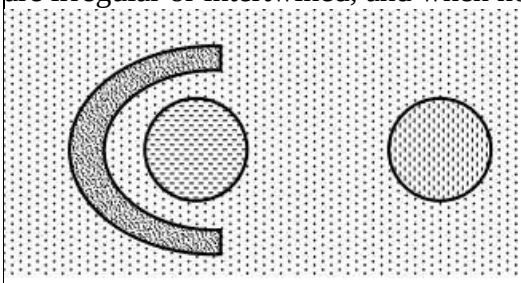
• Lazy learners such as nearest-neighbor classifiers do not require model building.

However, classifying a test example can be quite expensive because we need to compute the proximity values individually between the test and training examples.

• Nearest-neighbor classifiers make their predictions based on local information,

	whereas decision tree and rule-based classifiers attempt to find a global model that fits the entire input space. - Because the classification decisions are made locally, nearest-neighbor classifiers (with small values of *k*) are quite susceptible to noise. - Choosing the value of k: ‣ If k is too small, sensitive to noise points ‣ If k is too large, neighborhood may include points from other classes • Nearest-neighbor classifiers can produce arbitrarily shaped decision boundaries. Such boundaries provide a more flexible model representation compared to decision tree and rule-based classifiers that are often constrained to rectilinear decision boundaries. • Nearest-neighbor classifiers can produce wrong predictions unless the appropriate proximity measure and data preprocessing steps are taken. **Module -5**			
9.a.	**Describe K- means clustering algorithm, What are its Limitations.** **Scheme :** Algorithm + Limitations-----6+4 marks **Solution :** We first choose K initial centroids, where K is a user specified parameter, namely, the number of clusters desired. Each point is then assigned to the closest centroid, and each collection of points assigned to a centroid is a cluster. The centroid of each cluster is then updated based on the points assigned to the cluster. We repeat the assignment and update steps until no point changes clusters, or equivalently, until the centroids remain the same. K-means is formally described by Algorithm below. 1: Select K points as initial centroids. 2: **repeat** 3: Form K clusters by assigning each point to its closest centroid. 4: Recompute the centroid of each cluster. 5: **until** Centroids do not change. Limitations of K-means clustering K-means is not suitable for all types of data. K-means has problems when clusters are of differing ◦Sizes ◦Densities ◦Non-globular shapes K-means has problems when the data contains outliers. Strength: ◦K-means is simple and can be used for a wide variety of data types. ◦It is also quite efficient, even though multiple runs are often performed.	**10**	**5**	**L3**
9.b	**Explain DBSCAN algorithm with an Example** **Scheme :** DBSCAN Algo+ Explanation---------6+4 marks **Solution :** Density-based clustering locates regions of high density that are separated from one another by regions of low density. DBSCAN is a simple and effective density-based clustering algorithm. In the center-based approach, density is estimated for a particular point in the data set by counting the number of points within a specified radius, Eps, of that point. This includes the point itself. This technique is graphically illustrated by Figure here. The number of points within a radius of Eps of point A is 7, including A itself.	**10**	**5**	**L3**

	This method is simple to implement, but the density of any point will depend on the specified radius. For instance, if the radius is large enough, then all points will have a density of m, the number of points in the data set. If the radius is too small, then all points will have a density of 1. The center-based approach to density allows us to classify a point as being (1) a core point : in the interior of a dense region - These points are in the interior of a density-based cluster. - In Figure below, point A is a core point. (2) a border point: on the edge of a dense region - A border point is not a core point, but falls within the neighborhood of a core point. - In Figure below, point B is a border point. - A border point can fall within the neighborhoods of several core points. (3) a noise or background point: in a sparsely occupied region. - A noise point is any point that is neither a core point nor a border point. - In Figure below, point C is a noise point.  **(OR)**			
10 a.	**Explain the following in brief a) Density Based clustering b) Graph based clustering** **Scheme:** Explanation of Density based and graph based clustering------5+5 marks **Solution :** **Density based clustering** A cluster is a dense region of objects that is surrounded by a region of low density. - Figure (d) shows some density-based clusters for data created by adding noise to the data of Figure (c). - The two circular clusters of Figure (c) are not merged. - A density-based definition of a cluster is often employed when the clusters	**10**	**5**	**L3**

	are irregular or intertwined, and when noise and outliers are present.  (d) Density-based clusters. Clusters are regions of high density separated by regions of low density. **graph based clustering** Graph-Based clustering uses the proximity graph Start with the proximity matrix Consider each point as a node in a graph Each edge between two nodes has a weight which is the proximity between the two points. Initially the proximity graph is fully connected MIN (single-link) and MAX (complete-link) can be viewed as starting with this graph,In the simplest case, clusters are connected components in the graph. Sparsification The m by m proximity matrix for m data points can be represented as a dense graph in which each node is connected to all others and the weight of the edge between any pair of nodes reflects their pairwise proximity. Although every object has some level of similarity to every other object, for most data sets, objects are highly similar to a small number of objects and weakly similar to most other objects. This property can be used to sparsify the proximity graph (matrix), by setting many of these low-similarity (highdissimilarity) values to 0 before beginning the actual clustering process Sparsification has several beneficial effects: • Data size is reduced. • Clustering may work better. • Graph partitioning algorithms can be used.			
10.b.	**Explain the BIRCH Scalable algorithm** **Schem:** Algorithm +Explanation -----6+4 marks **Solution :** BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) is a highly efficient clustering technique for data in Euclidean vector spaces, i'e., data for which averages make sense. BIRCH can efficiently cluster such data with one pass and can improve that clustering with additional passes. BIRCH can also deal effectively with outliers.	**10**	**5**	**L3**

Algorithm 9.13 BIRCH.

1: **Load the data into memory by creating a CF tree that summarizes the data.**
2: **Build a smaller CF tree if it is necessary for phase 3.** T is increased, and then the leaf node entries (clusters) are reinserted. Since T has increased, some clusters will be merged.
3: **Perform global clustering.** Different forms of global clustering (clustering that uses the pairwise distances between all the clusters) can be used. However, an agglomerative, hierarchical technique was selected. Because the clustering features store summary information that is important to certain kinds of clustering, the global clustering algorithm can be applied as if it were being applied to all the points in a cluster represented by the CF.
4: **Redistribute the data points using the centroids of clusters discovered in step 3, and thus, discover a new set of clusters.** This overcomes certain problems that can occur in the first phase of BIRCH. Because of page size constraints and the T parameter, points that should be in one cluster are sometimes split, and points that should be in different clusters are sometimes combined. Also, if the data set contains duplicate points, these points can sometimes be clustered differently, depending on the order in which they are encountered. By repeating this phase multiple times, the process converges to a locally optimum solution.

Faculty : Jayashree M