DATA WAREHOUSING AND DATA MINING

NOTE:

MAKAUT course structure and syllabus of 6th semester has been changed from 2021. Previously **DATA WAREHOUSING AND DATA MINING** was in 7th semester. This subject has been redesigned and shifted in 6th semester as per present curriculum. Subject organization has been changed slightly. Taking special care of this matter we are providing the relevant MAKAUT university solutions and some model questions & answers for newly introduced topics, so that students can get an idea about university questions patterns.

INTRODUCTION TO DATA WAREHOUSING

Multiple Choice Type Questions

1. A data warehouse is an integrated collection of data because [WBUT 2009, 2015]
   a) It is a collection of data of different types
   b) It is a collection of data derived from multiple sources
   c) It is a relational database
   d) It contains summarized data

Answer: (b)

2. A data warehouse is said to contain a 'subject oriented' collection of data because [WBUT 2009, 2013]
   a) Its contents have a common theme
   b) It is built for a specific application
   c) It cannot support multiple subjects
   d) It is a generalization of 'object-oriented'

Answer: (a)

3. A Data warehouse is said to be contain in *time-varying* collection of data because [WBUT 2010, 2013, 2015]
   a) Its content vary automatically with time
   b) Its life-span is very limited
   c) Every key structure of data warehouse contains either implicitly or explicitly an element of time
   d) Its content has explicit time-stamp

Answer: (c)

4. Data Warehousing is used for [WBUT 2010, 2012]
   a) Decision Support System
   b) OLTP applications
   c) Database applications
   d) Data Manipulation applications

Answer: (a)

5. Which of the following is TRUE? [WBUT 2010, 2012]
   a) Data warehouse can be used for analytical processing only
   b) Data warehouse can be used for information processing (query, report) and analytical processing
   c) Data warehouse can be used for data mining only
   d) Data warehouse can be used for information processing (query, report), analytical processing and data mining

Answer: (d)

6. Data warehouse architecture is just an over guideline. It is not a blueprint for the data warehouse [WBUT 2011]
   a) True
   b) False

Answer: (b)

7. The most distinguishing characteristic of DSS data is [WBUT 2011]
a) Granularity b) Timespan c) Dimensionality d) Data currency

Answer: (c)

8. A data warehouse is built as a separate repository of data, different from the operational data of an enterprise because [WBUT 2012, 2013]
a) It is necessary to keep the operational data free of any warehouse operations
b) A data warehouse cannot afford to allow corrupted data within it
c) A data warehouse contains summarized data whereas the operational database contains transactional data
d) None of these

Answer: (c)

9. Dimension data within a warehouse exhibits which one of the following properties? [WBUT 2012, 2015]
a) Dimension data consists of the minor part of the warehouse
b) The aggregated information is actually dimension data
c) It contains historical data
d) Dimension data is the information that is used to analyze the elemental transaction

Answer: (b)

10. A data warehouse is said to contain a 'time-varying' collection of data because
a) its content has explicit time-stamp [WBUT 2014, 2015, 2016, 2017]
b) its life-span is very limited
c) it contains historical data
d) it contents vary automatically with time

Answer: (c)

11. The important aspect of the data warehouse environment is that data found within the data warehouse is [WBUT 2016, 2018]
a) subject-oriented b) time-variant
c) integrated d) all of these

Answer: (a)

12. Data warehousing is used for [WBUT 2016]
a) decision support system b) OLAP applications
c) Database applications d) Data manipulation applications

Answer: (a)

13. ..... is a subject–oriented, integrated, time-variant, non-volatile collection of data [WBUT 2017]
a) Data Mining b) Data Warehousing
c) Document Mining d) Text Mining

Answer: (b)

Short Answer Type Questions

1. Define Data Marts. [WBUT 2009, 2010, 2011, 2015, 2018]
Define the types of Data Marts. [WBUT 2009, 2010, 2011, 2018]

Answer:

1st Part:

A data mart is a group of subjects that are organized in a way that allows them to assist departments in making specific decisions. For example, the advertising department will have its own data mart, while the finance department will have a data mart that is separate from it. In addition to this, each department will have full ownership of the software, hardware, and other components that make up their data mart.

2nd Part:

There are two types of Data Marts:

- **Independent** data marts – sources from data captured form OLTP system, external providers or from data generated locally within a particular department or geographic area.
- **Dependent** data mart – sources directly form enterprise data warehouses.

2. Define data mining. What is the advantages data mining over traditional approaches? [WBUT 2009]

Answer:

1st Part:

Data mining, which is also known as knowledge discovery, is one of the most popular topics in information technology. It concerns the process of automatically extracting useful information and has the promise of discovering hidden relationships that exist in large databases. These relationships represent valuable knowledge that is crucial for many applications. Data mining is not confined to the analysis of data stored in data warehouses. It may analyze data existing at more detailed granularities than the summarized data provided in a data warehouse. It may also analyze transactional, textual, spatial, and multimedia data which are difficult to model with current multidimensional database technology.

2nd Part:

With the help of data mining, organizations are in a better position to predict the future regarding the business trends, the possible amount of revenue that could be generated, the orders that could be expected and the type of customers that could be approached. The traditional approaches will not be able to generate such accurate results as they use simpler algorithms. One major advantage of data mining over a traditional statistical approach is its ability to deal directly with heterogeneous data fields.

The advantages of data mining helps the businesses grow help the customers be happy, and help in a lot of other areas like data management.

3. What is the importance of Association Rules in Data mining? [WBUT 2009]

OR,

Explain support, confidence, frequent item set and give a formal definition of association rule. [WBUT 2013]

OR,

What is an Association Rule? Define Support, Confidence, Item set and Frequent item set in Association Rule Mining? [WBUT 2017]

Answer:

To illustrate the concepts, we use a small example from the supermarket domain. The set of items is I = {milk, bread, butter, beer} and a small database containing the items is shown in Table below.

Transaction ID	Items
1	Milk, bread
2	Bread, butter
3	Beer
4	Milk, bread, butter
5	Bread, butter

An example supermarket database with five transactions.

An example rule for the supermarket could be {milk, bread} ➔ {butter} meaning that if milk and bread is bought, customers also buy butter. To select interesting rules from the set of all possible rules, constraints on various measures of significance and interest can be used. The best-known constraints are minimum thresholds on support and confidence.

The **support** supp(X) of an itemset X is defined as the proportion of transactions in the data set which contain the itemset. In the example database in Table 1, the itemset {milk, bread} has a support of 2/5 = 0.4 since it occurs in 40% of all transactions (2 out of 5 transactions).

The **confidence** of a rule is defined conf(X) Y) = supp(X [Y)/supp(X). For example, the rule {milk, bread}) {butter} has a confidence of 0.2/0.4 = 0.5 in the database in the Table, which means that for 50% of the transactions containing milk and bread the rule is correct. Confidence can be interpreted as an estimate of the probability P(Y |X), the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.

In many (but not all) situations, we only care about association rules or causalities involving sets of items that appear frequently in baskets. For example, we cannot run a good marketing strategy involving items that no one buys anyway. Thus, much data mining starts with the assumption that we only care about sets of items with high support; i.e., they appear together in many baskets. We then find association rules or causalities only involving a high-support set of items must appear in at least a certain percent of the baskets, called the support threshold. We use the term **frequent itemset** for "a set S that appears in at least fraction s of the baskets," where s is some chosen constant, typically 0.01 or 1%.

Association rules are statements of the form $\{X_1, X_2, \ldots, X_n\} \rightarrow Y$, meaning that if we find all of $X_1, X_2, \ldots X_n$ in the market basket, then we have a good chance of nding Y . The probability of finding Y for us to accept this rule is the condence of the rule. We normally would search only for rules that had confidence above a certain threshold.

4. State the differences between Data Mart & Warehouse. **[WBUT 2010, 2015]**

Answer:

A data warehouse has a structure which is separate from a data mart, even though they may look similar. Because of this, it is difficult to coordinate the data across multiple departments. Each department will have its own view of how a data mart should look. The data mart that they use will be specific to them. In contrast, a data warehouse is designed around the organization as a whole. Instead of being owned by one department, a data warehouse will generally be owned by the entire company. While the data contained in data warehouses are granular, the information contained in data marts is not very granular at all.

Another thing that separates data warehouses from data marts is that data warehouses contain larger amounts of information. The information that is held by data marts are often summarized. Data warehouses will not contain information that is biased on the part of the department. Instead, it will demonstrate the information that is analyzed and processed by the organization. Much of the information that is held in data warehouses is historical in nature, and they are designed to process this information.

5. When is a Data Mart appropriate? **[WBUT 2011]**

Answer:

Data Marts are created for:

a) Speeding up queries by reducing the volume of data to be scanned
b) Structuring data in a form suitable for user access tools
c) Partitioning data to impose access control strategies
d) Segmenting data into different hardware platforms.

Data marts should not be used in other cases as the operational costs of data marting can be high and once a strategy in place, it can be difficult to change it without incurring substantial cost.

6. What is sequence mining? **[WBUT 2011]**

Answer:

Sequence mining is a type of structured data mining in which the database and administrator look for sequences or trends in the data. This data mining is split into two fields. Itemset sequence mining typically is used in marketing, and string sequence mining is used in biology research. Sequence mining is different from regular trend mining, because the data are more specific, which makes building an effective database difficult for database designers, and it can sometimes go awry if the sequence is any different from the common sequence.

7. Differentiate among Enterprise Warehouse, Data mart and Virtual warehouse. **[WBUT 2013]**

Answer:

Enterprise warehouse - collects all of the information about subjects spanning the entire organization

Data Mart - a subset of corporate-wide data that is of value to specific groups of users. Its scope is confined to specific, selected groups, such as marketing data mart.
Virtual warehouse - A set of views over operational databases. Only some of the possible summary views may be materialized.

8. How is data warehouse different from a database? [WBUT 2013, 2016, 2018]

Answer:

A database is used to store data while a data warehouse is mostly used to facilitate reporting and analysis. Basically, database is just where the data is stored; in order to access this data or analyze it a database management system is required. However, a data warehouse does not necessarily require a DBMS. The purpose of a data warehouse is for easy access to the data for a user. The data warehouse may also be used to analyze the data; however the actual process of analysis is called data mining.

Some differences between a database and a data warehouse:

- A database is used for Online Transactional Processing (OLTP) but can be used for other purposes such as Data Warehousing.
- A data warehouse is used for Online Analytical Processing (OLAP). This reads the historical data for the Users for business decisions.
- In a database the tables and joins are complex since they are normalized for RDMS. This reduces redundant data and saves storage space.
- In data warehouse, the tables and joins are simple since they are de-normalized. This is done to reduce the response time for analytical queries.
- Relational modeling techniques are used for RDMS database design, whereas modeling techniques are used for the Data Warehouse design.
- A database is optimized for write operation, while a data warehouse is optimized for read operations.
- In a database, the performance is low for analysis queries, while in a data warehouse, there is high performance for analytical queries.
- A data warehouse is a step ahead of a database. It includes a database in its structure.

9. What are the issues relating to the diversity of database types? [WBUT 2016]

Answer:

- **Handling of relational and complex types of data** – The database may contain complex data objects, multimedia data objects, spatial data, temporal data etc. It is not possible for one system to mine all these kind of data.
- **Mining information from heterogeneous databases and global information systems** – The data is available at different data sources which may be structured, semi structured or unstructured. Therefore mining the knowledge from them adds challenges to data mining.

10. Introduce the idea of Data Mining with example(s). What are the steps involved in Knowledge discovery in Database (KDD) process? [WBUT 2016]

Answer:

Data Mining: ***Refer to Question No. 2(1st Part) of Short Answer Type Questions.***

Data mining example:

Mobile phone and utilities companies use Data Mining and Business Intelligence to predict 'churn', the terms they use for when a customer leaves their company to get their phone/gas/broadband from another provider. They collate billing information, customer services interactions, website visits and other metrics to give each customer a probability score, then target offers and incentives to customers whom they perceive to be at a higher risk of churning.

Steps involved in KDD process

1. Identify the goal of the KDD process from the customer's perspective.
2. Understand application domains involved and the knowledge that's required
3. Select a target data set or subset of data samples on which discovery is be performed.
4. Cleanse and pre-process data by deciding strategies to handle missing fields and alter the data as per the requirements.
5. Simplify the data sets by removing unwanted variables. Then, analyze useful features that can be used to represent the data, depending on the goal or task.
6. Match KDD goals with data mining methods to suggest hidden patterns.
7. Choose data mining algorithms to discover hidden patterns. This process includes deciding which models and parameters might be appropriate for the overall KDD process.
8. Search for patterns of interest in a particular representational form, which include classification rules or trees, regression and clustering.
9. Interpret essential knowledge from the mined patterns.
10. Use the knowledge and incorporate it into another system for further action.
11. Document it and make reports for interested parties.

11. Does a data warehouse involve a transaction? Explain. **[WBUT 2018]**

Answer:

A data warehouse is a relational database that is designed for query and analysis rather than for transaction processing. It usually contains historical data derived from transaction data, but it can include data from other sources. It separates analysis workload from transaction workload and enables an organization to consolidate data from several sources.

Long Answer Type Questions

1. Define Data Warehouse and briefly describe its characteristics. **[WBUT 2009, 2018]**

Answer:

A data warehouse consists of a computer database responsible for the collection and storage of information for a specific organization. This collection of information is then used to manage information efficiently and analyze the collected data. Although data

warehouses vary in overall design, majority of them are subject oriented, meaning that the stored information is connected to objects or events that occur in reality. The data provided by the data warehouse for analysis provides information on a specific subject, rather than the functions of the company and is collected from varying sources into one unit having time-variant.

A data warehouse has significantly different features from other enterprise-wide systems, particularly in how data is stored, managed and manipulated.

There are four key characteristics which separate the data warehouse from other major operational systems:

1. **Subject Orientation:** Data organized by subject
2. **Integration:** Consistency of defining parameters
3. **Non-volatility:** Stable data storage medium
4. **Time-variance:** Timeliness of data and access terms

2. a) What are the different tiers in a typical 3-tier data warehousing architecture?
b) What do you mean by warehousing schema? Explain. **[WBUT 2009]**

Answer:

a) The bottom tier is a **warehouse database server** which is almost always a relational database system.

The middle tier is an **OLAP server** which is typically implemented using either (1) a Relational OLAP (ROLAP) model, i.e., an extended relational DBMS that maps operations on multidimensional data to standard relational operations; or (2) a Multidimensional OLAP (MOLAP) model, i.e., a special purpose server that directly implements multidimensional data and operations.

The top tier is a **client**, which contains query and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on).

b) The entity-relationship data model is commonly used in the design of relational databases, where a database schema consists of a set of entities or objects, and the relationships between them. Such a data model is appropriate for online transaction processing. Data warehouses, however, require a concise, subject-oriented schema which facilitates on-line data analysis.

3. a) Introduce the concept of data mining and cite two application areas. **[WBUT 2010, 2012]**

b) What are the different steps of a data mining task? **[WBUT 2010, 2012]**

c) Suppose that the data mining task is to cluster the following eight points (with (x, y) representing location) into 3 clusters.

$$A_1(2,10), A_2(2,5), A_3(8,4), B_1(5,8), B_2(7,5), B_3(6,4), C_1(1,2), C_2(4,9)$$

The distance function is Euclidian distance. Initially we assign A_1, B_1 and C_1 as the centre of each cluster. Use k-means algorithm to determine the three clusters. **[WBUT 2010, 2013, 2014, 2016, 2018]**

Answer:

a) 1st Part: *Refer to of Question No. 2(1st Part) of Short Answer Type Questions.*

2nd Part:

An **application** of data mining is market segmentation. With market segmentation, a company will be able to find behaviors that are common among its customers. One can look for patterns among customers that seem to purchase the same products at the same time. Another application of data mining is called customer churn. Customer churn will allow a company to estimate which customers are the most likely to stop purchasing its products or services and go to one of its competitors.

b) The Following Steps Are Usually Followed In Data Mining. These steps are iterative, with the process moving backward whenever needed.

1. Develop an understanding of the application, of the relevant prior knowledge, and of the end user's goals.
2. Create a target data set to be used for discovery.
3. Clean and preprocess data (including handling missing data fields, noise in the data, accounting for time series, and known changes).
4. Reduce the number of variables and find invariant representations of data if possible.
5. Choose the data-mining task (classification, regression, clustering, etc.).
6. Choose the data-mining algorithm.
7. Search for patterns of interest (this is the actual data mining).
8. Interpret the pattern mined. If necessary, iterate through any of steps 1 through 7.
9. Consolidate knowledge discovered and prepare a report.

c) The given eight points were:

A1: (2, 10)
A2: (2, 5)
A3: (8, 4)
B1: (5, 8)
B2: (7, 5)
B3: (6, 4)
C1: (1, 2)
C2: (4, 9)

The points A1, B2, and C1 were assumed to be the initial three cluster centres. There are two possible ways of clustering the given points using the k-means algorithm, depending on to which cluster we assign the point B1:

The first way:

1.1. Round One:

- Compute for each of the eight points its distance from each cluster centre, and assign it to the cluster to which it is most similar.

=> The clustering after the first round execution is:

Cluster 1: A1, B1, C2
Cluster 2: B2, A3, B3
Cluster 3: C1, A2

- update the cluster centres:
The new centre of Cluster 1: (3.67, 9)
The new centre of Cluster 2: (7, 4.33)
The new centre of Cluster 3: (1.5, 3.5)

1.2. Round Two:
- No changes

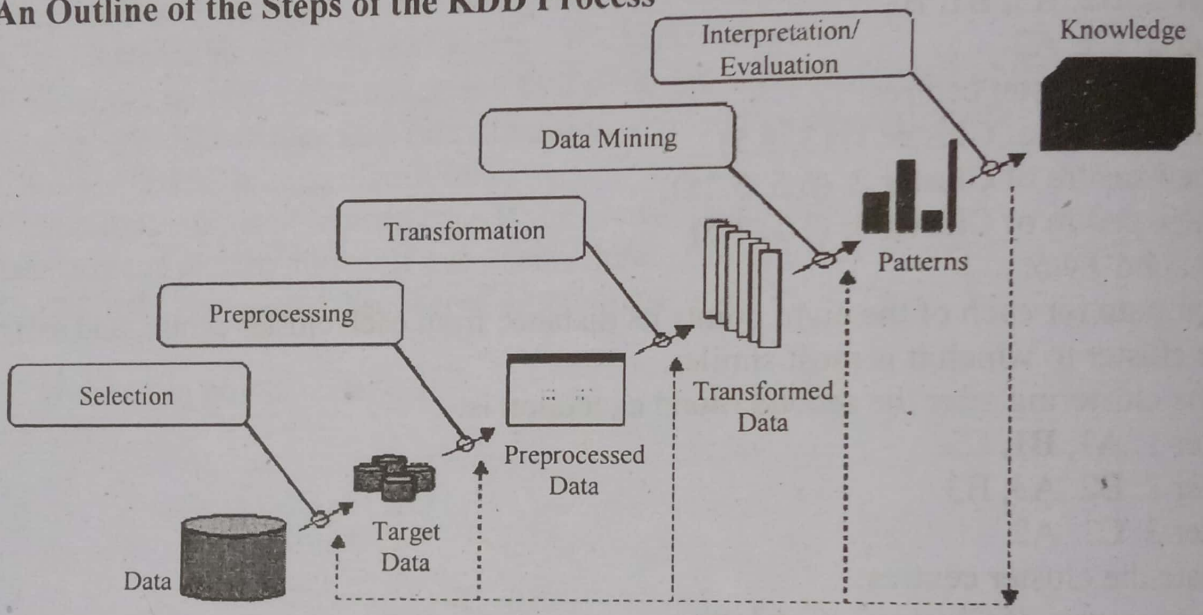
The second way:
2.1. Round One:
- Compute for each of the eight points its distance from each cluster centre, and assign it to the cluster to which it is most similar.
=> The clustering after the first round execution is:
Cluster 1: A1, C2
Cluster 2: B2, A3, **B1,** B3
Cluster 3: C1, A2
- update the cluster centres:
The new centre of Cluster 1: (3, 9.5)
The new centre of Cluster 2: (6.5, 5.25)
The new centre of Cluster 3: (1.5, 3.5)
2.2. Round Two:
- recompute for each of the eight points its distance from each cluster centre, and assign it to the cluster to which it is most similar.
=> The clustering after the second round execution is:
Cluster 1: A1, **B1,** C2
Cluster 2: B2, A3, B3
Cluster 3: C1, A2
- update the cluster centres:
The new centre of Cluster 1: (3.67, 9)
The new centre of Cluster 2: (7, 4.33)
The new centre of Cluster 3: (1.5, 3.5)
2.3. Round Three:
- No changes

4. a) Write down the design steps for a typical data warehouse. **[WBUT 2015]**
b) Explain the flowchart for KDD process.
c) Define Roll-up and Drill-down process with a suitable example.
Answer:
a) In general, building any data warehouse consists of the following steps:

1. **Extracting the transactional data from the data sources into a staging area -** The Extract step covers the data extraction from the source system and makes it accessible for further processing. The main objective of the extract step is to retrieve all the required data from the source system with as little resources as possible. The extract step should be designed in a way that it does not negatively affect the source system in terms or performance, response time or any kind of locking.

2. **Transforming the transactional data** - The transform step applies a set of rules to transform the data from the source to the target. This includes converting any measured data to the same dimension (i.e. conformed dimension) using the same units so that they can later be joined. The transformation step also requires joining data from several sources, generating aggregates, generating surrogate keys, sorting, deriving new calculated values, and applying advanced validation rules.
3. **Loading the transformed data into a dimensional database** - During the load step, it is necessary to ensure that the load is performed correctly and with as little resources as possible. The target of the Load process is often a database. In order to make the load process efficient, it is helpful to disable any constraints and indexes before the load and enable them back only after the load completes. The referential integrity needs to be maintained by ETL tool to ensure consistency.

b) An Outline of the Steps of the KDD Process



The overall process of finding and interpreting patterns from data involves the repeated application of the following steps:

1. Developing an understanding of
   - the application domain
   - the relevant prior knowledge
   - the goals of the end-user
2. Creating a target data set: selecting a data set, or focusing on a subset of variables, or data samples, on which discovery is to be performed.
3. Data cleaning and preprocessing.
   - Removal of noise or outliers.
   - Collecting necessary information to model or account for noise.
   - Strategies for handling missing data fields.
   - Accounting for time sequence information and known changes.
4. Data reduction and projection.
   - Finding useful features to represent the data depending on the goal of the task.

- Using dimensionality reduction or transformation methods to reduce the effective number of variables under consideration or to find invariant representations for the data.

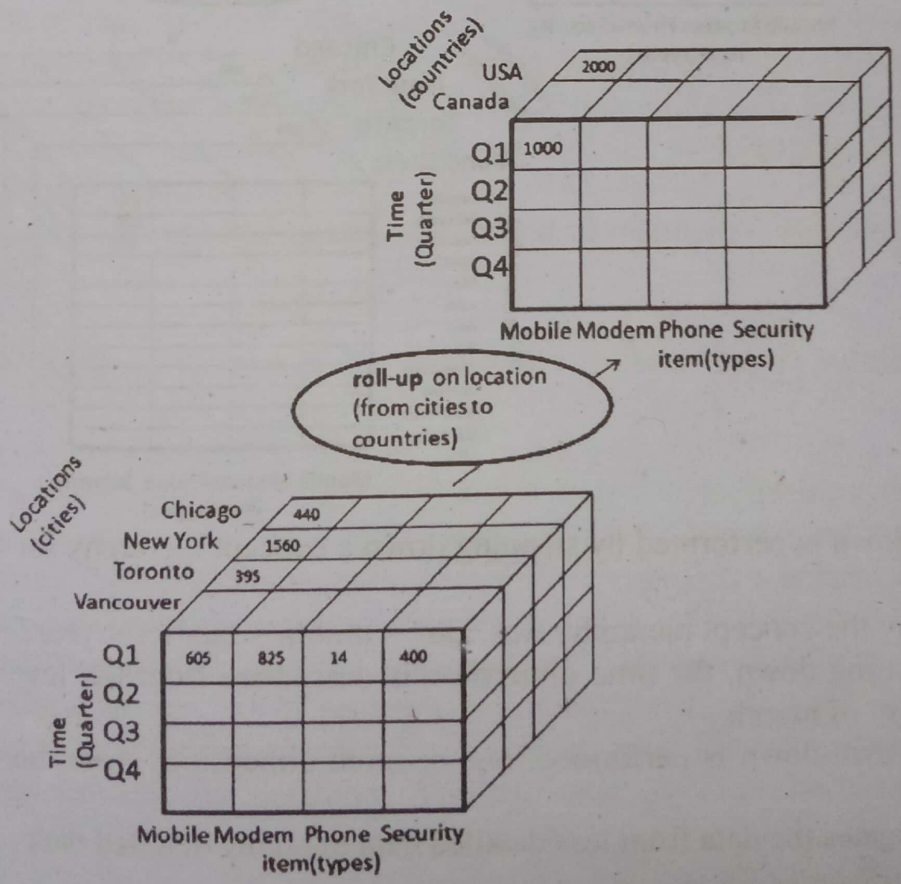
5. Choosing the data mining task.
   - Deciding whether the goal of the KDD process is classification, regression, clustering, etc.
6. Choosing the data mining algorithm(s).
   - Selecting method(s) to be used for searching for patterns in the data.
   - Deciding which models and parameters may be appropriate.
   - Matching a particular data mining method with the overall criteria of the KDD process.
7. Data mining.
   - Searching for patterns of interest in a particular representational form or a set of such representations as classification rules or trees, regression, clustering, and so forth.
8. Interpreting mined patterns.
9. Consolidating discovered knowledge.

c) Roll-up

Roll-up performs aggregation on a data cube in any of the following ways:

- By climbing up a concept hierarchy for a dimension
- By dimension reduction

The following diagram illustrates how roll-up works.



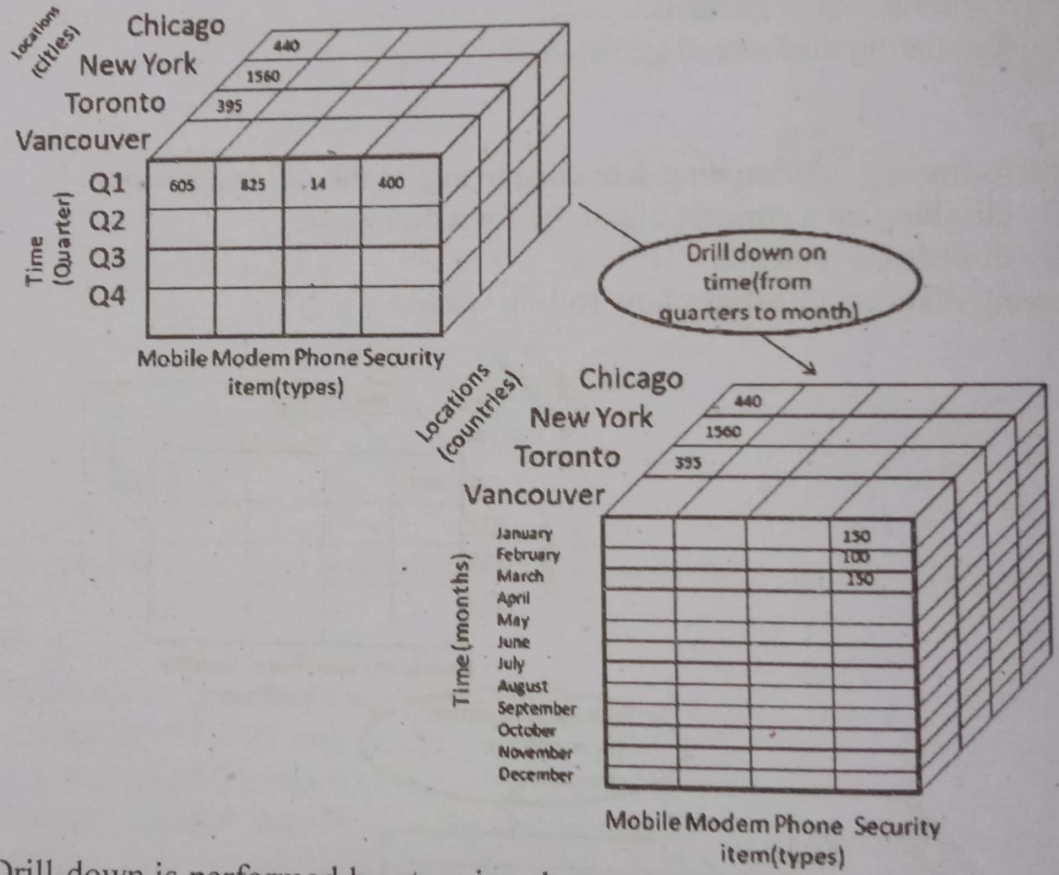
- Roll-up is performed by climbing up a concept hierarchy for the dimension location.
- Initially the concept hierarchy was "street < city < province < country".
- On rolling up, the data is aggregated by ascending the location hierarchy from the level of city to the level of country.
- The data is grouped into cities rather than countries.
- When roll-up is performed, one or more dimensions from the data cube are removed.

Drill-down

Drill-down is the reverse operation of roll-up. It is performed by either of the following ways:

- By stepping down a concept hierarchy for a dimension
- By introducing a new dimension.

The following diagram illustrates how drill-down works:



- Drill-down is performed by stepping down a concept hierarchy for the dimension time.
- Initially the concept hierarchy was "day < month < quarter < year."
- On drilling down, the time dimension is descended from the level of quarter to the level of month.
- When drill-down is performed, one or more dimensions from the data cube are added.
- It navigates the data from less detailed data to highly detailed data.

5. Write short notes on the following:
a) Strategic information **[WBUT 2014, 2016]**
b) Data Mart **[WBUT 2017]**
b) Issues and challenges in data mining **[WBUT 2009]**
c) Text mining **[WBUT 2009, 2012, 2015, 2017]**
d) DBMS vis-à-vis Data Mining **[WBUT 2010]**

Answer:

a) Strategic information:

Strategic information is needed for the day to day operations, meeting of government requirements as well long and short range planning. Strategic information systems are used by organizations to achieve a competitive advantage for the organization.

A Strategic Information System (SIS) is a system that helps companies change or otherwise alter their business strategy and/or structure. It is typically utilized to streamline and quicken the reaction time to environmental changes and aid it in achieving a competitive advantage. Strategic information systems relies on data gathered at all organizational levels and from all information systems whether it was transaction processing systems, management information systems, decision support systems or executive information systems.

Key features of the Strategic Information Systems are the following:

i) Decision support systems that enable to develop a strategic approach to align Information Systems (IS) or Information Technologies (IT) with an organization's business strategies

ii) Primarily Enterprise resource planning solutions that integrate/link the business processes to meet the enterprise objectives for the optimization of the enterprise resources

iii) Database systems with the "data mining" capabilities to make the best use of available corporate information for marketing, production, promotion and innovation. The SIS systems also facilitate identification of the data collection strategies to help optimize database marketing opportunities.

iv) The real-time information Systems that intend to maintain a rapid-response and the quality indicators.

b) Data Mart: ***Refer to Question No. 1 of Short Answer Type Questions.***

c) Issues and challenges in data mining:

One of the key issues raised by data mining technology is not a business or technological one, but a social one. It is the issue of individual privacy. Data mining makes it possible to analyze routine business transactions and glean a significant amount of information about individuals buying habits and preferences.

Another issue is that of data integrity. Clearly, data analysis can only be as good as the data that is being analyzed. A key implementation challenge is integrating conflicting or redundant data from different sources. For example, a bank may maintain credit cards accounts on several different databases. The addresses (or even the names) of a single

cardholder may be different in each. Software must translate data from one system to another and select the address most recently entered.
A hotly debated technical issue is whether it is better to set up a relational database structure or a multidimensional one. In a relational structure, data is stored in tables, permitting ad hoc queries. In a multidimensional structure, on the other hand, sets of cubes are arranged in arrays, with subsets created according to category. While multidimensional structures facilitate multidimensional data mining, relational structures thus far have performed better in client/server environments. And, with the explosion of the Internet, the world is becoming one big client/server environment.
Finally, there is the issue of cost. While system hardware costs have dropped dramatically within the past five years, data mining and data warehousing tend to be self-reinforcing. The more powerful the data mining queries, the greater the utility of the information being gleaned from the data, and the greater the pressure to increase the amount of data being collected and maintained, which increases the pressure for faster, more powerful data mining queries. This increases pressure for larger, faster systems, which are more expensive.

d) Text mining:
Text Mining is the discovery by computer of new, previously unknown information, by automatically extracting information from different written resources. A key element is the linking together of the extracted information together to form new facts or new hypotheses to be explored further by more conventional means of experimentation. In text mining, the goal is to discover heretofore unknown information, something that no one yet knows and so could not have yet written down.
The difference between regular data mining and text mining is that in text mining the patterns are extracted from natural language text rather than from structured databases of facts. Databases are designed for programs to process automatically; text is written for people to read. We do not have programs that can "read" text and will not have such for the forseeable future. Many researchers think it will require a full simulation of how the mind works before we can write programs that read the way people do.

e) DBMS vis-à-vis Data Mining:
DBMS and data mining answer different queries. Data mining helps in predicting future whereas DBMS gives reports.
Example DBMS Reports

- Last months sales for each service type
- Sales per service grouped by customer sex or age bracket
- List of customers who lapsed their policy

Questions answered using Data Mining

- What characteristics do customers that lapse their policy have in common and how do they differ from customers who renew their policy?
- Which motor insurance policy holders would be potential customers for my House Content Insurance policy?

6. What are data mining primitives?

Answer: **[MODEL QUESTION]**

Data mining primitives

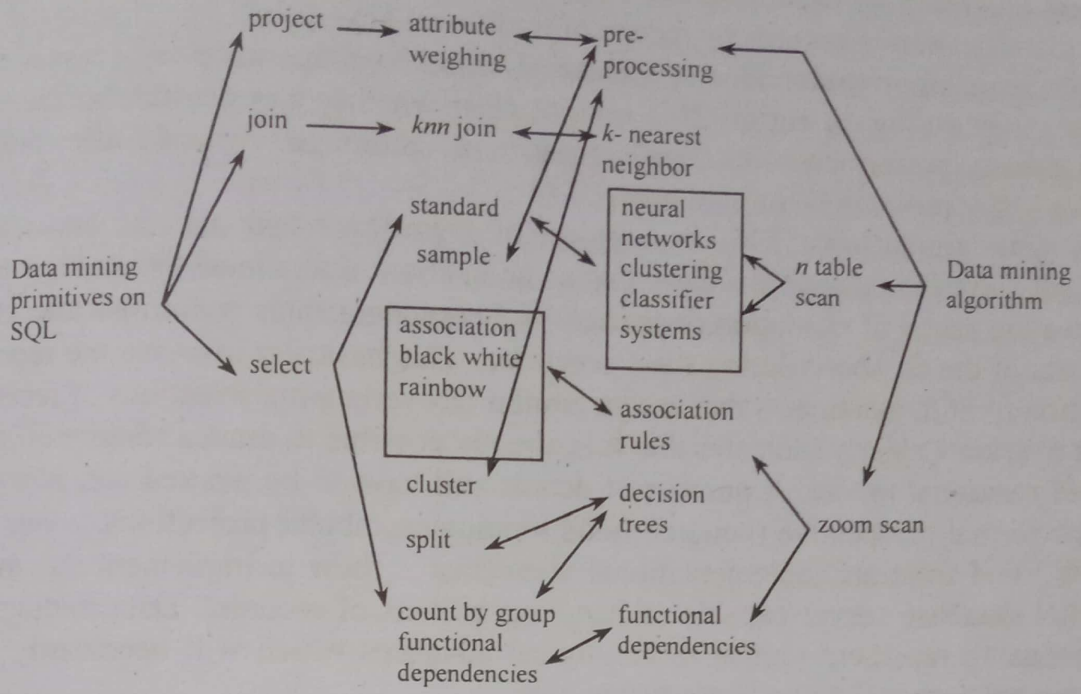


Fig: Data mining primitives in SQL

Contrary to current belief, data mining is first and foremost a server technology. Until recently, most data mining tools operated on flat files and in-memory databases. With the shift of attention to larger databases, the need for implementing data mining algorithms on top of relational databases has been recognized. However, current relational technology, with its emphasis on efficient updates and exact identification of records, is ill-suited to the support of data mining algorithms.

Data mining algorithms require:

- efficient sampling methods
- storage of intermediate results during execution of queries
- capacity to handle large amounts of data
- geometric indexes that allow users to search in the neighborhood of record and bit-mapped operations.

Data mining is supported by client/server technology, with the server technology being of great importance when working with large databases. All these functions can in principle be realized on top of existing implementations, as they are based on generalizations of the same mathematical structure, that is, first-order logic and algebra.

Can these ideas be brought into a general framework for the revision of the relational model? Following figure describes a number of fundamental data mining routines in relation to SQL primitives. In general, data mining routines can be divided into four groups:

1. **Pre-processing:** Routines in this group take samples, compress attribute values into clusters, and code tables into appropriate input formats for algorithms (that is,

transform historical files into time series, multi-valued attributes into sets of binary attributes, tables into bitmaps, etc.).

2. **Nearest neighbor search:** Neighbor search can be interpreted as a generalization of exact identification of records by means of traditional keys.
3. **N table scan algorithms:** This is a class of algorithms that need to access a table several times during execution. It is not yet clear whether it is possible to speed up the database performance of these algorithms, although enriched user-defined functions may prove to be of help here.
4. **Zoom scan algorithms:** This is a group of algorithms that also access a table repeatedly, but in this case it is clear that an adaptation of relational technology could improve the speed of execution dramatically. These algorithms zoom into interesting sub-parts of the database during their execution. This basically involves the repeated execution of SQL statements that access similar or overlapping selections of records.

This short overview clearly indicates that it is already possible to draw a rough outline of an enriched relational model. A number of details still have to be worked out, however, both from a formal perspective (neighborhood topologies, tabular projections, zoom scan queries, etc.) and from an implementational view (that is, how to implement this model on a parallel database server capable of handling billions of records). Data mining and data warehousing represent radical technological advances which will necessarily alter our fundamental approach to database technology.

7. Explain scalable methods for mining sequential patterns.

[MODEL QUESTION]

Answer:

Sequential pattern mining is computationally challenging because such mining may generate and/or test a combinatorically explosive number of intermediate subsequences.

"*How can we develop efficient and scalable methods for sequential pattern mining*?"

Recent developments have made progress in two directions: (1) efficient methods for mining the ful set of sequential patterns, and (2) efficient methods for mining only the set of closed sequential patterns, where a sequential pattern s is closed if there exists no sequential pattern s' where s' where s' is a proper super-sequence of s, and s' has the same (frequency) support as s. because all of the subsequences of a frequent sequence are also frequent, mining the set of closed sequential patterns may avoid the generation of unnecessary subsequences and thus lead to more compact results as well as more efficient methods than mining the full set. We will first examine methods for mining the full set and then study how they can be extended for mining the closed set. In addition, we discuss modifications for mining multilevel, multidimensional sequential patterns (i.e., with multiple levels of granularity).

We can give example of three such approaches for sequential pattern mining, represented by the algorithms GSP, SPADE, and PrefixSpan, respectively. GSP adopts a candidate generate-and-test approach using horizongal data format (where the data are represented as *<sequence_ID : sequence_of_itemsets>*, as usual, where each itemset is an event). SPADE adopts a candidate generate-and-test approach using vertical data format (where

the data are represented as <*itemst* : (*sequence_ID, event_ID*)>. The vertical data format can be obtained by transforming from a horizontally formatted sequence database in just one scan. Prefix Span is a pattern growth method, which does not require candidate generation.

All three approaches either directly or indirectly explore the Apriori property, stated as follows: every nonempty subsequence of a sequential pattern is a sequential pattern. (Recall that for a pattern to be called sequential, it must be frequent. That is, it must satisfy minimum support.) the Apriori property is antimonotonic (or downward-closed) in that, if a sequence cannot pass a test (e.g., regarding minimum support), all of its super-sequences will also fail the test. Use of this property to prune the search space can help make the discovery of sequential patterns more efficient.

CLASSIFICATION AND PREDICTION OF DATA WAREHOUSING

Multiple Choice Type Questions

1. Which of the following techniques are appropriate for data warehousing? [WBUT 2009, 2013, 2018]
   a) Hashing on primary keys
   b) Indexing on foreign keys of the fact table
   c) Bit-map indexing
   d) Join indexing

Answer: (c)

2. ..... is an example of predictive type of data mining whereas ..... is an example of descriptive type of data mining. [WBUT 2010, 2012]
   a) Association Rule, Clustering
   b) Association Rule, Classification
   c) Classification, Clustering
   d) Clustering, Classification

Answer: (c)

3. What is Metadata? [WBUT 2017]
   a) Summarized data
   b) Operational data
   c) Data about data
   d) None of these

Answer: (c)

4. The full form of OLAP is [WBUT 2017]
   a) Online Analytical Processing
   b) Online Advanced Processing
   c) Online Advanced preparation
   d) Online Analytical Performance

Answer: (a)

5. The apriori algorithm is a [WBUT 2017]
   a) top – down search
   b) breadth first search
   c) depth first search
   d) bottom-up search

Answer: (d)

6. Classification rules are extracted from [WBUT 2017]
   a) Root node
   b) Decision tree
   c) Siblings
   d) Branches

Answer: (b)

7. Which of the following is a predictive model? [WBUT 2017]
   a) Clustering
   b) Regression
   c) Summarization
   d) Association rules

Answer: (b)

8. All set of items whose support is greater than the user-specified minimum support are called as [WBUT 2017]

a) Border set
b) Frequent set
c) Maximal frequent set
d) Lattice

Answer: (b)

Short Answer Type Questions

1. Explain the use of Dynamic itemset counting algorithm. **[WBUT 2009]**

OR,

Describe the principle of Dynamic Itemset Counting technique for Frequent Itemset generation. **[WBUT 2010]**

Answer:

Dynamic Itemset Counting is an alternative to Apriori Itemset Generation. Here Itemsets are dynamically added and deleted as transactions are read and it relies on the fact that for an itemset to be frequent, all of its subsets must also be frequent, so we only examine those itemsets whose subsets are all frequent.

Algorithm stops after every M transactions to add more itemsets.

- **Train analogy:** There are stations every M transactions. The passengers are itemsets. Itemsets can get on at any stop as long as they get off at the same stop in the next pass around the database. Only itemsets on the train are counted when they occur in transactions. At the very beginning we can start counting 1-itemsets, at the first station we can start counting some of the 2-itemsets. At the second station we can start counting 3-itemsets as well as any more 2-itemsets that can be counted and so on.

Itemsets are marked in four different ways as they are counted:

- **Solid box:** ☐ confirmed frequent itemset - an itemset we have finished counting and exceeds the support threshold *minsupp*
- **Solid circle:** ◯ confirmed infrequent itemset - we have finished counting and it is below *minsupp*
- **Dashed box:** ⬚ suspected frequent itemset - an itemset we are still counting that exceeds *minsupp*
- **Dashed circle:** ◌ suspected infrequent itemset - an itemset we are still counting that is below *minsupp*

DIC Algorithm

1. Mark the empty itemset with a solid square. Mark all the 1-itemsets with dashed circles. Leave all other itemsets unmarked.
2. While any dashed itemsets remain:
   a) Read M transactions (if we reach the end of the transaction file, continue from the beginning). For each transaction, increment the respective counters for the itemsets that appear in the transaction and are marked with dashes.
   b) If a dashed circle's count exceeds *minsupp*, turn it into a dashed square. If any immediate superset of it has all of its subsets as solid or dashed squares, add a new counter for it and make it a dashed circle.

Once a dashed itemset has been counted through all the transactions, make it solid and stop counting it.

2. State Apriori Algorithm for frequent item set generation. **[WBUT 2010]**

Answer:

The Apriori Algorithm is an influential algorithm for mining frequent itemsets for boolean association rules.

Key Concepts:

- **Frequent Itemsets:** The sets of item which has minimum support (denoted by L_i for ith-Itemset).
- **Apriori Property:** Any subset of frequent itemset must be frequent.
- **Join Operation:** To find L_k, a set of candidate k-itemsets is generated by joining L_{k-1} with itself.

Algorithm

- Find the *frequent itemsets*: the sets of items that have minimum support
  - A subset of a frequent itemset must also be a frequent itemset i.e., if $\{AB\}$ is a frequent itemset, both $\{A\}$ and $\{B\}$ should be a frequent itemset
  - Iteratively find frequent itemsets with cardinality from 1 to *k (k*-itemset*)*
- Use the frequent itemsets to generate association rules.

Pseudo code

- Join Step: C_k is generated by joining L_{k-1} with itself
- Prune Step: Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset

Pseudo-code:

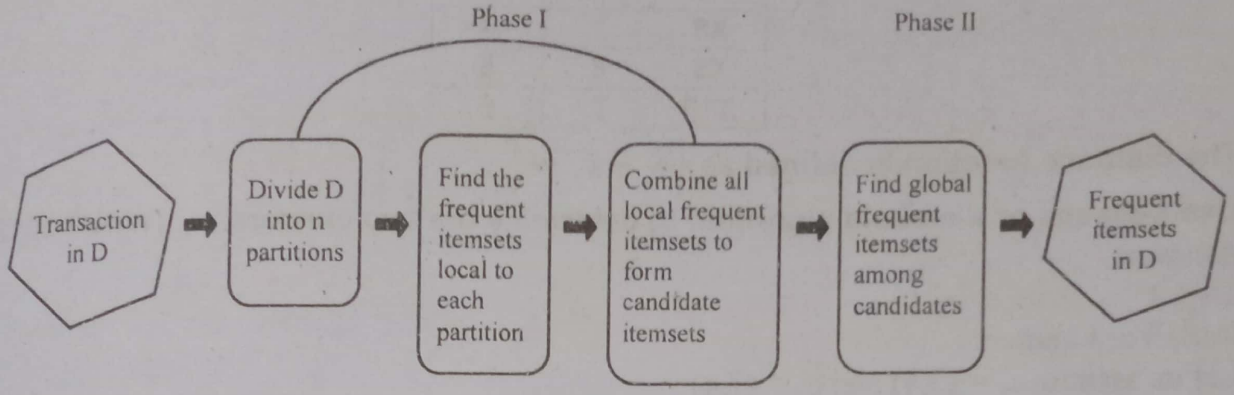
C_k: Candidate itemset of size k
L_k: frequent itemset of size k
L_1= {frequent items};
For (k= 1; L_k!=∅; k++) do begin
    C_{k+1}= candidates generated from L_k;
    for each transaction tin database do
        increment the count of all candidates in C_{k+1} that are contained in t
    L_{k+1}= candidates in C_{k+1} with min_support
end
return $\cup_k L_k$;

3. Describe the principle of partitioning technique for Frequent itemset generation and justify how it improves the efficiency of Frequent item set generation compared to a priori Algorithm. **[WBUT 2010, 2012]**

Answer:

A partitioning technique can be used that requires just two database scans to mine the frequent itemsets. It consists of two phases. In Phase I, the algorithm subdivides the

transactions of D into n nonoverlapping partitions. If the minimum support threshold for transactions in D is *min sup*, then the minimum support count for a partition is *minsup, the number of transactions in that partition*. For each partition, all frequent itemsets within the partition are found. These are referred to as local frequent itemsets. In Phase II, a second scan of D is conducted in which the actual support of each candidate is assessed in order to determine the global frequent itemsets. Partition size and the number of partitions are set so that each partition can fit into main memory and therefore be read only once in each phase. The following figure describes the phases.



Apriori algorithm needs many database scans and for each scan, frequent itemsets are searched by pattern matching, which is especially time-consuming for large frequent itemsets with long patterns. Partition algorithm uses the intersection of transaction ids (tids) to determine the support count. Because it performs a breadth-first search, it partitions the database into blocks in which local frequent itemsets are found to overcome the memory limitations. An extra database scan is needed to determine the global frequency of local frequent itemsets.

4. What is clustering? Discuss two main methods of clustering. [WBUT 2010]

Answer:

1st Part: ***Refer to Question No. 1(a) (1st Part) of Long Answer Type Questions.***

2nd Part:

The non-hierarchical techniques in general are faster to create from the historical database but require that the user make some decision about the number of clusters desired or the minimum "nearness" required for two records to be within the same cluster. These non-hierarchical techniques are often run multiple times starting off with some arbitrary or even random clustering and then iteratively improving the clustering by shuffling some records around. These techniques sometimes create clusters that are created with only one pass through the database adding records to existing clusters when they exist and creating new clusters when no existing cluster is a good candidate for the given record. Because the definition of which clusters are formed can depend on these initial choices of which starting clusters should be chosen or even how many clusters these techniques can be less repeatable than the hierarchical techniques and can sometimes create either too many or too few clusters because the number of clusters is predetermined by the user not determined solely by the patterns inherent in the database.

5. Suppose that the data mining task is to cluster the following ten points (with (x,y)) representing location into two clusters:

X1	2	6
X2	3	4
X3	3	8
X4	4	7
X5	6	2
X6	6	4
X7	7	3
X8	7	4
X9	8	5
X10	7	6

The distance function is defined as $\left[x_i - x_j\right] + \left[y_i - y_j\right]$

Use k-means or k-medoid algorithm to determine the two clusters. **[WBUT 2012]**

Answer:

Step 1

Initialize *k* centers.

Let us assume $c_1 = (3,4)$ and $c_2 = (7,4)$

So here c_1 and c_2 are selected as medoids.

Costs to the nearest medoid are shown bold in the table.

i	c_1		Data objects (X_i)		Cost (distance)
1	3	4	2	6	**3**
3	3	4	3	8	**4**
4	3	4	4	7	**4**
5	3	4	6	2	5
6	3	4	6	4	3
7	3	4	7	3	5
9	3	4	8	5	6
10	3	4	7	6	6

i	c_2		Data objects (X_i)		Cost (distance)
1	7	4	2	6	7
3	7	4	3	8	8
4	7	4	4	7	6
5	7	4	6	2	**3**
6	7	4	6	4	**1**
7	7	4	7	3	**1**
9	7	4	8	5	**2**
10	7	4	7	6	**2**

Then the clusters become:

$\text{Cluster}_1 = \{(3,4)(2,6)(3,8)(4,7)\}$

$\text{Cluster}_2 = \{(7,4)(6,2)(6,4)(7,3)(8,5)(7,6)\}$

Since the points (2,6) (3,8) and (4,7) are closer to c_1 hence they form one cluster whilst remaining points form another cluster.
So the total cost involved is 20.
Where cost between any two points is found using formula

$$\text{cost}(x,c) = \sum_{i=1}^{d} |x - c|$$

where x is any data object, c is the medoid, and d is the dimension of the object which in this case is 2.
Total cost is the summation of the cost of data object from its medoid in its cluster so here:

Total cost = {cost((3,4), (2,6)) + cost((3,4), (3,8)) + cost((3,4), (4,7))}
+ {cost((7,4), (6,2)) + cost((7,4), (6,4)) + cost((7,4), (7,3))
+ cost((7,4), (8,5)) + cost((7,4), (7,6))}
= (3 + 4 + 4) + (3 + 1 + 1 + 2 + 2)
= 20

Select one of the nonmedoids O′
Let us assume O′ = (7,3)
So now the medoids are c_1(3,4) and O′(7,3)
If c1 and O′ are new medoids, calculate the total cost involved
By using the formula in the step 1

i	c_1		Data objects (X_i)		Cost (distance)
1	3	4	2	6	3
3	3	4	3	8	4
4	3	4	4	7	4
5	3	4	6	2	5
6	3	4	6	4	3
7	3	4	7	4	4
9	3	4	8	5	6
10	3	4	7	6	4

i	O′		Data objects (X_i)		Cost (distance)
1	7	3	2	6	8
3	7	3	3	8	9
4	7	3	4	7	7
5	7	3	6	2	2
6	7	3	6	4	2
7	7	3	7	4	1
9	7	3	8	5	3
10	7	3	7	6	3

Total cost = 3 + 4 + 4 + 2 + 2 + 1 + 3 + 3 = 22

So cost of swapping medoid from c_2 to O′ is
S = current total cost – past total cost = 22 – 20 = 2 > 0
So moving to O′ would be a bad idea, so the previous choice was good. So we try other nonmedoids and found that our first choice was the best. So the configuration does not change and algorithm terminates here (i.e. there is no change in the medoids).

6. What is metadata in Data warehousing? What is metadata catalog? Discuss the different categories of metadata used in data warehouse. **[WBUT 2014]**

OR,

a) What is metadata? **[WBUT 2015]**
b) What is the typical content of metadata of a data warehouse? **[WBUT 2015]**

OR,

What is Metadata? Explain different types of Metadata. **[WBUT 2017]**

Answer:
Metadata is simply defined as data about data. The data that is used to represent other data is known as metadata. In terms of data warehouse, we can define metadata as follows.

- Metadata is the road-map to a data warehouse.
- Metadata in a data warehouse defines the warehouse objects.
- Metadata acts as a directory. This directory helps the decision support system to locate the contents of a data warehouse.

A collection of metadata records combined with data management and search tools forms a data catalogue. The architecture of these data catalogues can vary from tightly held internal data indexes to widely accessible distributed catalogues spread across the Internet.
Metadata can be broadly categorized into three categories:

- **Business Metadata:** It has the data ownership information, business definition, and changing policies.
- **Technical Metadata:** It includes database system names, table and column names and sizes, data types and allowed values. Technical metadata also includes structural information such as primary and foreign key attributes and indices.
- **Operational Metadata:** It includes currency of data and data lineage. Currency of data means whether the data is active, archived, or purged. Lineage of data means the history of data migrated and transformation applied on it.

Long Answer Type Questions

1. a) What is clustering? Why is it difficult to handle categorical data for clustering? Distinguish between partition clustering and hierarchical clustering. **[WBUT 2009]**

OR,

What is clustering? What are the features of good cluster? What do you mean by hierarchical clustering technique? **[WBUT 2013, 2014, 2016, 2018]**

Answer:

1st Part:

Clustering is a data mining (machine learning) technique used to place data elements into related groups without advance knowledge of the group definitions.

There are two main types of clustering techniques, those that create a **hierarchy** of clusters and **non-hierarchy** of clusters. The hierarchical clustering techniques create a hierarchy of clusters from small to big. With a hierarchy of clusters defined it is possible to choose the number of clusters that are desired. At the extreme it is possible to have as many clusters as there are records in the database. In this case the records within the cluster are optimally similar to each other (since there is only one) and certainly different from the other clusters. The advantage of hierarchical clustering methods is that they allow the end user to choose from either many clusters or only a few.

2nd Part:

Since, the process of grouping a set of physical or abstract objects into classes of similar objects is clustering, similarity metric is important here because it is used for outlier detection. The clustering algorithm which is main memory based can operate only on the following two data structures namely,

a) Data Matrix

b) Dissimilarity Matrix

So it is difficult to handle categorical data.

In partition clustering, every data sample is initially assigned to a cluster in some (possibly random) way. Samples are then iteratively transferred from cluster to cluster until some criterion function is minimized. Once the process is complete, the samples will have been partitioned into separate compact clusters. Examples of partition clustering methods are k-means and Lloyd's method.

In hierarchical clustering, each sample is initially considered a member of its own cluster, after which clusters are recursively combined in pairs according to some predetermined condition until eventually every point belongs to a single cluster. The resulting hierarchical structure may be represented by a binary tree or "dendrogram", from which the desired clusters may be extracted. Examples of hierarchical clustering methods are the single-link, Ward's, centroid, complete-link, group average, median, and parametric Lance Williams methods.

b) Describe the working of the PAM algorithm. Compare its performance with CLARA and CLARANS. [WBUT 2009]

Answer:

The PAM k-medoids clustering algorithm, for example, evaluates a set of k objects considered to be representative objects (medoids) of k clusters within T objects such that the non-selected objects are clustered with the medoid to which it is the most similar (i.e. closest in terms of the provided distance metric). The process operates by swapping one of the medoids with one of the objects iteratively such that the total distance between non-selected objects and their medoid is reduced. The algorithm can be depicted as follows:

Step 1: Initialization - choose k medoids from T objects randomly.
Step 2: Evaluation - calculate the cost $D'_t - D_t$ for each swap of one medoid with one object, where D_t is the total distance before the swap and D'_t is the total distance after the swap.
Step 3: Selection - accept the swap with the best cost and if the cost is negative, go to step 2; otherwise record the medoids and terminate the program.

The computational complexity of the PAM algorithm is $O((1 + \beta)k(T - k)^2)$ which is based on the number of partitions per object, where β is the number of successful swaps. It can also be expressed as $O'((1 + \beta)k^2(T - k)^2)$ based on the number of distance calculations, i.e., one partition per object is equivalent to k distances calculations. Clearly, this is time consuming even for the moderate number of objects and a small number of medoids.
The computational complexity of the CLARA algorithm is $O(q(ks^2 + (T - k)) + \beta ks^2)$ based on the number of partitions per object or $O'(q(k^2s^2 + k(T - k)) + \beta k^2s^2)$ based on the number of distance calculations, where q, s, k, β and T are the number of samples, object size per sample, number of medoids, the number of successful swaps for all samples tested and the total number of objects, respectively. Clearly, the CLARA algorithm can deal with a larger number objects than can PAM algorithm if $s \ll T$.
The computational complexity of CLARANS is $O((\beta + \text{numlocal})(T - k))$ based on the number of partitions per object or $O'((\beta + \text{numlocal})k(T - k))$ based on the number of distance calculations, where β is the number of test moves between nodes.

2. a) What is Clustering? Why is it required for data warehousing and data mining? What are the differences between partitioning clustering and Hierarchical clustering?
b) Suppose the following table shows the distances between different cities. Construct a hierarchical tree using Hierarchical clustering method.

	BA	FA	MI	NA	RM	TO
BA	0	662	877	255	412	996
FI	662	0	295	468	268	400
MI	877	295	0	754	564	138
NA	255	468	754	0	219	869
RM	412	268	564	219	0	669
To	996	400	138	869	669	0

c) Discuss *K*-mean algorithm with a suitable example. **[WBUT 2017]**
Answer:
a) 1st Part: ***Refer to Question No. 1(a) (1st Part) of Long Answer Type Questions.***

2nd Part:
Clustering is required for data warehousing and data mining for the following reasons:
- Scalability – We need highly scalable clustering algorithms to deal with large databases.

- Ability to deal with different kinds of attributes – Algorithms should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data.
- Discovery of clusters with attribute shape – The clustering algorithm should be capable of detecting clusters of arbitrary shape. They should not be bounded to only distance measures that tend to find spherical cluster of small sizes.
- High dimensionality – The clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space.
- Ability to deal with noisy data – Databases contain noisy, missing or erroneous data. Some algorithms are sensitive to such data and may lead to poor quality clusters.
Interpretability – The clustering results should be interpretable, comprehensible, and usable.

3rd Part: ***Refer to Question No. 1(a) (2nd Part) of Long Answer Type Questions.***

b) The nearest pair of cities is MI and TO, at distance 138. These are merged into a single cluster called "MI/TO". The level of the new cluster is L(MI/TO) = 138 and the new sequence number is m = 1.

Then we compute the distance from this new compound object to all other objects. In single link clustering the rule is that the distance from the compound object to another object is equal to the shortest distance from any member of the cluster to the outside object. So the distance from "MI/TO" to RM is chosen to be 564, which is the distance from MI to RM, and so on.

After merging MI with TO we obtain the following matrix:

	BA	FI	MI/TO	NA	RM
BA	0	662	877	255	412
FI	662	0	295	468	268
MI/TO	877	295	0	754	564
NA	255	468	754	0	219
RM	412	268	564	219	0

min d(i,j) = d(NA,RM) = 219 => merge NA and RM into a new cluster called NA/RM

L(NA/RM) = 219

m = 2

	BA	FI	MI/TO	NA/RM
BA	0	662	877	255
FI	662	0	295	268
MI/TO	877	295	0	564
NA/RM	255	268	564	0

min d(i,j) = d(BA,NA/RM) = 255 => merge BA and NA/RM into a new cluster called BA/NA/RM

L(BA/NA/RM) = 255

m = 3

	BA/NA/RM	FI	MI/TO

BA/NA/RM	0	268	564
FI	268	0	295
MI/TO	564	295	0

min d(i,j) = d(BA/NA/RM,FI) = 268 => merge BA/NA/RM and FI into a new cluster called BA/FI/NA/RM

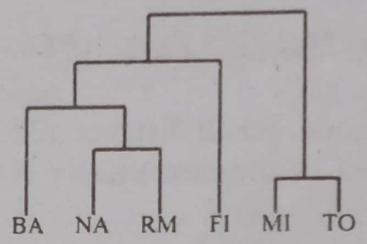
L(BA/FI/NA/RM) = 268

m = 4

	BA/NA/RM	**MI/TO**
BA/NA/RM	0	295
MI/TO	295	295

Finally, we merge the last two clusters at level 295.

The process is summarized by the following hierarchical tree:



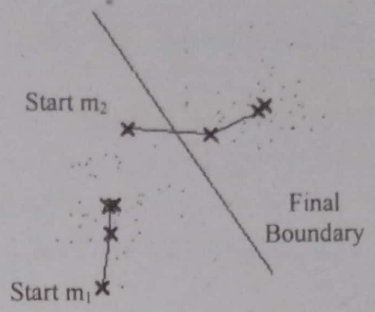
c) K-means is an unsupervised learning algorithms that solve the clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The algorithm is composed of the following steps:

1. *Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.*
2. *Assign each object to the group that has the closest centroid.*
3. *When all objects have been assigned, recalculate the positions of the K centroids.*
4. *Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.*

Suppose that we have n sample feature vectors $x_1, x_2, ..., x_n$ all from the same class, and we know that they fall into k compact clusters, $k < n$. Let mi be the mean of the vectors in cluster i. If the clusters are well separated, we can use a minimum-distance classifier to separate them. That is, we can say that x is in cluster i if $|| x - m_i ||$ is the minimum of all the k distances. This suggests the following procedure for finding the k means:

- Make initial guesses for the means $m_1, m_2, ..., m_k$
- Until there are no changes in any mean
  - Use the estimated means to classify the samples into clusters
  - For i from 1 to k
    - Replace mi with the mean of all of the samples for cluster i
  - end_for
- end_until

Here is an example showing how the means m1 and m2 move into the centers of two clusters.



3. a) Introduce the concept of Support, Confidence and Frequent Itemset and then give a formal definition of Association Rule.

b) Generate all Frequent Itemsets from the following transaction data given minimum support = 0.3.

TID	Items	TID	Items
1	A, B, C, E	6	B, C
2	B, D, E	7	A, C, E
3	B, C	8	A, B, C, E
4	A, B, D	9	A, B, C
5	A, C	10	C, D, E

c) Find the Association Rules from the above Frequent sets at min. 50% confidence. **[WBUT 2010]**

Answer:

a) *Refer to Question No. 2 of Short Answer Type Questions.*

b) Pass1

itemset	support
{A}	6/10
{B}	7/10
{C}	8/10
{D}	3/10
{E}	5/10

Pass2

itemset	support
{A,B}	4/10
{A,C}	5/10
{A,D}	1/10
{A,E}	3/10
{B,C}	5/10
{B,D}	2/10
{B,E}	3/10
{C,D}	1/10
{C,E}	4/10
{D,E}	2/10

large itemset	support
{A,B}	4/10
{A,C}	5/10
{A,E}	3/10
{B,C}	5/10
{B,E}	3/10
{C,E}	4/10

Pass3

itemset	support
{A,B,C}	3/10
{A,B,E}	2/10
{A,C,E}	3/10
{B,C,E}	2/10

frequent itemset	support
{A,B,C}	3/10
{A,C,E}	3/10

Pass3

itemset	support
{A,B,C,E}	2/10

Therefore frequent itemsets are {A,B,C} and {A,C,E} with min_support 0.3.

c) Association Rules:

{A,B} →C, confidence(c) = σ(A,B,C}/σ(A,B) = 3/4=0.75
{B,C}→A, confidence(c) =σ(A,B,C)/σ(B,C)=3/5 = 0.6
{A,C}→B, confidence(c) =σ(A,B,C)/σ(A,C)=3/5 = 0.6
{A,C}→E, confidence(c) =σ(A,C,E)/σ(A,C)=3/5=0.6
{A,E}→C, confidence(c) =σ(A,C,E)/σ(A,E)=3/3=1
{C,E}→A, confidence(c) =σ(A,C,E)/σ(C,E)=3/4=0.75

4. Write short notes on the following:

a) Dimensional Modeling **[WBUT 2018]**
b) Generalized Association Rule **[WBUT 2010]**
c) PAM Clustering Technique **[WBUT 2010]**
d) CLARANS clustering algorithm vis-à-vis PAM and CLARA **[WBUT 2010]**
e) *K*-Medoid Algorithm **[WBUT 2017]**

Answer:

a) Dimensional Modeling:

A dimensional model is a data structure technique optimized for Data warehousing tools. The concept of Dimensional Modelling was developed by Ralph Kimball and is comprised of "fact" and "dimension" tables.

A Dimensional model is designed to read, summarize, analyze numeric information like values, balances, counts, weights, etc. in a data warehouse. In contrast, relation models

are optimized for addition, updating and deletion of data in a real-time Online Transaction System.
These dimensional and relational models have their unique way of data storage that has specific advantages.
For instance, in the relational mode, normalization and ER models reduce redundancy in data. On the contrary, dimensional model arranges data in such a way that it is easier to retrieve information and generate reports.
Hence, Dimensional models are used in data warehouse systems and not a good fit for relational systems.

b) Generalized Association Rule:
In most real-life applications taxonomies (is-a hierarchies) over the items are available. In general, a taxonomy can be represented as a directed acyclic graph (DAG). Given a set of transactions T each of which is a set of items, and a taxonomy Tax, the problem of mining generalized association rules is to discover all rules of the form $X \rightarrow Y$, with the user-specified minimum support and minimum confidence. X and Y can be sets of items at any level of the taxonomy, such that no item in Y is an ancestor of any item in X. For example, there might be a rule which says that "50% of transactions that contain Soft drinks also contain Snacks; 5% of all transactions contain both these items"

c) PAM Clustering Technique:
PAM stands for "partition around medoids". The algorithm is intended to find a sequence of objects called *medoids* that are centrally located in clusters. Objects that are tentatively defined as medoids are placed into a set S of *selected objects*.
If O is the set of objects that the set $U = O - S$ is the set of *unselected objects. The goal of the algorithm is to minimize the average dissimilarity of objects to their closest selected object*. Equivalently, we can minimize the sum of the dissimilarities between object and their closest selected object.
The algorithm has two phases:
(i) In the first phase, BUILD, a collection of k objects are selected for an initial set S.
(ii) In the second phase, SWAP, one tries to improve the quality of the clustering by exchanging selected objects with unselected objects.
The goal of the algorithm is to minimize the average dissimilarity of objects to their closest selected object. Equivalently, we can minimize the sum of the dissimilarities between object and their closest selected object.

d) CLARANS clustering algorithm vis-à-vis PAM and CLARA:
The mediods searching in PAM or CLARA is abstracted as searching k subgraphs from n points graph, and based on this understanding, a PAM-like clustering algorithm called CLARANS (Clustering Large Applications based upon RANdomized Search) is defined. While PAM searches the whole graph and CLARA searches some random sub-graphs, CLARANS randomly samples a set and selects k medoids in climbing sub-graph mountains. CLARANS selects the neighboring objects of medoids as candidates of new medoids. It samples subsets to verify medoids in multiple times to avoid bad samples.

Obviously, multiple time sampling of medoids verification is time consuming which limits CLARANS from clustering very large datasets in an acceptable time period.

e) *K*-Medoid Algorithm:

Partitioning Around Medoids or the K-medoids algorithm is a partitional clustering algorithm which is slightly modified from the K-means algorithm. The basic idea of this algorithm is to first compute the K representative objects which are called as medoids. After finding the set of medoids, each object of the data set is assigned to the nearest medoid. That is, object i is put into cluster v_i, when medoid mv_i is nearer than any other medoid m_w.

The algorithm proceeds in two steps:

- **BUILD-step:** This step sequentially selects k "centrally located" objects, to be used as initial medoids
- **SWAP-step:** If the objective function can be reduced by interchanging (swapping) a selected object with an unselected object, then the swap is carried out. This is continued till the objective function can no longer be decreased.

The algorithm is as follows:

1. Initially select k random points as the medoids from the given n data points of the data set.
2. Associate each data point to the closest medoid by using any of the most common distance metrics.
3. For each pair of non-selected object h and selected object i, calculate the total swapping cost TC_{ih}.

   If $TC_{ih} < 0$, i is replaced by h
5. Repeat the steps 2-3 until there is no change of the medoids.

MINING TIME SERIES DATA

Multiple Choice Type Questions

1. To optimize data warehouse design, which one is done? [WBUT 2011]
   a) Normalization of fact tables and demoralization of dimension tables
   b) Normalization of fact tables and dimension tables
   c) Demoralization of fact tables and dimension tables
   d) Normalization of dimension tables and demoralization of fact tables

Answer: (d)

2. _____, is the value of an attribute is examined as it varies over time.
   a) Time series Analysis
   b) Classification [MODEL QUESTION]
   c) Association
   d) Prediction

Answer: (d)

Long Answer Type Questions

1. a) Define decision tree. [WBUT 2009, 2010]
b) What are the advantages and disadvantages of the decision tree approach over other approaches for data mining? [WBUT 2009]
c) Discuss briefly the tree construction principle. [WBUT 2009]

OR,

Describe the basic algorithm for decision tree induction. [WBUT 2013]
d) What is a classification problem? What is the difference between supervised and unsupervised classification? [WBUT 2009]

Answer:

a) Decision trees are powerful and popular tools for classification and prediction. The attractiveness of decision trees is due to the fact that, in contrast to neural networks, decision trees represent rules.

Decision tree is a classifier in the form of a tree structure (see Figure 1), where each node is either:

- **a leaf node** - indicates the value of the target attribute (class) of examples, or
- **a decision node** - specifies some test to be carried out on a single attribute-value, with one branch and sub-tree for each possible outcome of the test.

A decision tree can be used to classify an example by starting at the root of the tree and moving through it until a leaf node, which provides the classification of the instance.

Information gain, is simply the expected reduction in entropy caused by partitioning the examples according to this attribute. More precisely, the information gain, Gain (S, A) of an attribute A, relative to a collection of examples S, is defined as

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

where Values(A) is the set of all possible values for attribute A, and S_v is the subset of S for which attribute A has value v (i.e., $S_v = \{s \varepsilon S \mid A(s) = v\}$). Note the first term in the

equation for Gain is just the entropy of the original collection S and the second term is the expected value of the entropy after S is partitioned using attribute A. The expected entropy described by this second term is simply the sum of the entropies of each subset Sv, weighted by the fraction of examples $|S_v|/|S|$ that belong to S_v. Gain (S,A) is therefore the expected reduction in entropy caused by knowing the value of attribute A. Put another way, Gain(S,A) is the information provided about the target attribute value, given the value of some other attribute A. The value of Gain(S,A) is the number of bits saved when encoding the target value of an arbitrary member of S, by knowing the value of attribute A.

A decision tree can be constructed top-down using the information gain in the following way:

1. begin at the root node
2. determine the attribute with the highest information gain which is not already used as an ancestor node
3. add a child node for each possible value of that attribute
4. attach all examples to the child node where the attribute values of the examples are identical to the attribute value attached to the node
5. if all examples attached to the child node can be classified uniquely add that classification to that node and mark it as leaf node
6. go back to step two if there are unused attributes left, otherwise add the classification of most of the examples attached to the child node

b) ***Advantages of Decision Tree model***

- Decision tree methods tend to produce models that are easy to interpret. At each non-terminal node, a decision is based upon just one predictor variable and this makes it easy to follow. For example, to explain particular classification one need only look at the series of simple decisions that led to it. The final tree model can in-fact be cast into a set of rules one can follow to classify a given case. In comparison, generalized linear models use linear combinations of variables that can be difficult to interpret or explain.
- Tree models make no assumptions about the distribution of the underlying data and they are thus a non-parametric procedure. This can be especially useful if the distribution of the data is indeed unknown, something that happens a lot in practice.
- Decision tree methods are easily able to handle both categorical and continuous variables.
- Decision tree methods have a built-in feature selection method that makes them immune to the presence of useless variables. Such variables are ignored and they do no affect the tree building process. This is a common problem with over-parametized datasets.
- Tree models are very adept at revealing complex interactions between variables. Each branch of a tree can contain different combinations of variables and the same variable can appear more than once in different parts of the tree. This can reveal how a variable can depend on another and in what specific context this dependency exists.

- Decision tree models are extremely robust to the effect of outliers. This is so because the models are constructed in a frequency-based technique where one is counting the instances in a split. Outliers that occur in the independent variables do not affect the tree growing process because the values used to split each node are not likely to be on the outlier values. Outliers in the dependent variable go into their own nodes and do not affect the rest of tree.
- Tree models offer several ways of dealing with missing values that can often minimize or eliminate the effect of such values on model performance. In many other methods the whole set of instances with missing values would be omitted from analysis. This advantage is present in both the when training the tree model or when applying the model on new data for class prediction.

Disadvantages of Decision Tree Models:

Decision trees are indeed powerful data mining tools, but they are not perfect and are suited to certain types of problems. The main weaknesses in decision tree modeling are listed below:

- Classification trees can be unstable and small variations in the data (such as that made by randomization) can cause very different looking trees being generated. This is especially so in cases where the goodness of splits for the different variables are close to each other in value. In this case, a small variation in the data is enough to influence the picking of on split over another.
  - Many of these problems are dealt with by using stratified sampling when creating testing and training datasets. In this case, the training and testing datasets have the same class distribution.
  - Such problems are also a symptom of a dataset that is too small. In such cases, it is important to use the entire dataset to train and test the model. This is the idea behind the cross-validation technique. Since the final model is built on the whole dataset, randomization of this same data will not cause too much instability.
- Some the tree models generated may be very large and complex. This can make the model difficult to interpret, understand or justify.
- Tree models really shine when applied to classification problems; unfortunately, they are not very good at estimation tasks(as in regression), i.e. having numerical dependent variable.
- Decision tree models are computationally expensive to train

c) Most algorithms that have been developed for learning decision trees are variations on a core algorithm that employs a top-down, greedy search through the space of possible decision trees. Decision tree programs construct a decision tree T from a set of training cases.

Constructing Decision Tree using ID3

```
function ID3
Input:   (R: a set of non-target attributes,
          C: the target attribute,
          S: a training set) returns a decision tree;
```

```
begin
   If S is empty, return a single node with
      value Failure;
   If S consists of records all with the same
      value for the target attribute,
      return a single leaf node with that value;
   If R is empty, then return a single node
      with the value of the most frequent of the
      values of the target attribute that are
      found in records of S; [in that case
      there may be be errors, examples
      that will be improperly classified];
   Let A be the attribute with largest
      Gain(A,S) among attributes in R;
   Let {aj| j=1,2, .., m} be the values of
      attribute A;
   Let {Sj| j=1,2, .., m} be the subsets of
      S consisting respectively of records
      with value aj for A;
   Return a tree with root labeled A and arcs
      labeled a1, a2, .., am going respectively
      to the trees (ID3(R-{A}, C, S1), ID3(R-{A}, C, S2),
      .....,ID3(R-{A}, C, Sm);
   Recursively apply ID3 to subsets {Sj| j=1,2, .., m}
      until they are empty
end
```

d) Classification is the process of dividing a dataset into mutually exclusive groups such that the members of each group are as "close" as possible to one another, and different groups are as "far" as possible from one another, where distance is measured with respect to specific variable(s) one is trying to predict. For example, a typical classification problem is to divide a database of companies into groups that are as homogeneous as possible with respect to a creditworthiness variable with values "Good" and "Bad."

With supervised classification, we identify examples of the Information classes (i.e., land cover type) of interest in an image(say). These are called "training sites". The image processing software system is then used to develop a statistical characterization of the reflectance for each information class. This stage is often called "signature analysis" and may involve developing a characterization as simple as the mean or the rage of reflectance on each bands, or as complex as detailed analyses of the mean, variances and covariance over all bands. Once a statistical characterization has been achieved for each information class, the image is then classified by examining the reflectance for each pixel and making a decision about which of the signatures it resembles most.

Unsupervised classification is a method which examines a large number of unknown pixels and divides into a number of classed based on natural groupings present in the image values. unlike supervised classification, unsupervised classification does not require analyst-specified training data. The basic premise is that values within a given

cover type should be close together in the measurement space (i.e. have similar gray levels), whereas data in different classes should be comparatively well separated (i.e. have very different gray levels)

2. a) What is the use of Regression? What may be the reasons for not using the linear regression model to estimate the output data? [WBUT 2016]

Answer:

Regression analysis is widely used for prediction and forecasting, where its use has substantial overlap with the field of machine learning. Regression analysis is also used to understand which among the independent variables are related to the dependent variable, and to explore the forms of these relationships.

Linear Regression Disadvantages:

- By its nature, linear regression only looks at linear relationships between dependent and independent variables. That is, it assumes there is a straight-line relationship between them. Sometimes this is incorrect.
- Linear regression looks at a relationship between the mean of the dependent variable and the independent variables.
- Linear Regression is sensitive to Outliers
- Linear regression assumes that the data are independent. This is often, but not always, sensible. Two common cases where it does not make sense are clustering in space and time.

b) How is time series data used in pattern analysis? Give the formula for Pearson's r. [WBUT 2016]

Answer:

Time series data have a temporal order that makes analysis distinctly different from other data analysis. The goal of time series analysis can be divided into characterization or prediction. There is a consistent pattern contaminated with random noise, which typically requires filtering to aid in identifying the underlying pattern. The pattern itself can be divided into the main trend and a seasonal component.

Pearson's correlation coefficient when applied to a population is

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

where:

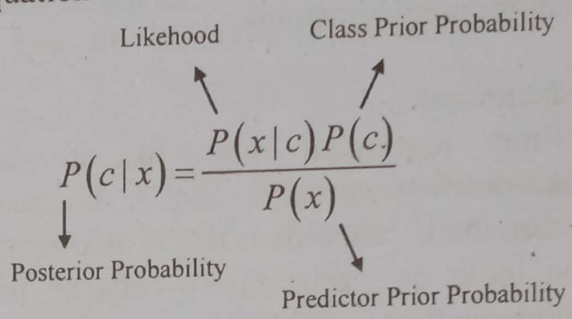
- cov is the covariance
- σ_X is the standard deviation of X
- σ_Y is the standard deviation of Y

c) Explain Bayesian classification. [WBUT 2016]

Answer:

Bayesian classification technique is based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that

the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.
Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.
Bayes theorem provides a way of calculating posterior probability $P(c|x)$ from $P(c)$, $P(x)$ and P(x|c). Look at the equation below:



$$P(c|X) = P(x_1|c) \times P(x_2|c) \times .... \times P(x_n|c) \times P(c)$$

Above,

- $P(c|x)$ is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
- $P(c)$ is the prior probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

3. a) What are the uses of training data set and test data set for a decision tree classification scheme? **[WBUT 2010, 2012, 2018]**
b) Discuss the principle of FP- tree Growth algorithm. **[WBUT 2010, 2013]**

OR,

Discuss the different phases of FP-tree growth algorithm. **[WBUT 2012]**

OR,

Define FP tree. Discuss the method of computing FP tree. **[WBUT 2018]**

Answer:

a) Decision trees need two kinds of data: Training and Testing.
Training data, which are usually the bigger part of data, are used for constructing trees. The more training data collected, the higher the accuracy of the results. The other group of data, testing, is used to get the accuracy rate and misclassification rate of the decision tree.

b) FP-Growth algorithm allows frequent itemset discovery without candidate itemset generation. It is a two step approach:
Step 1: Build a compact data structure called the FP-tree
Built using 2 passes over the data-set.

Step 2: Extracts frequent itemsets directly from the FP-tree
Traversal through FP-Tree

The frequent-pattern tree (FP-tree) is a compact structure that stores quantitative information about frequent patterns in a database.

1) One root labeled as "null" with a set of item-prefix subtrees as children, and a frequent-item-header table (presented in the right side of Figure 1);
2) Each node in the item-prefix subtree consists of three fields:
   a) **Item-name:** registers which item is represented by the node;
   b) **Count:** the number of transactions represented by the portion of the path reaching the node;
   c) **Node-link:** links to the next node in the FP-tree carrying the same item-name, or null if there is none.
3) Each entry in the frequent-item-header table consists of two fields:
   a) **Item-name:** as the same to the node;
   b) **Head of node-link:** a pointer to the first node in the FP-tree carrying the item-name.

Additionally the frequent-item-header table can have the count support for an item.

Algorithm 1: FP-tree construction

Input: A transaction database DB and a minimum support threshold.

Output: FP-tree, the frequent-pattern tree of DB.

Method: The FP-tree is constructed as follows.

1. Scan the transaction database DB once. Collect F, the set of frequent items, and the support of each frequent item. Sort F in support-descending order as FList, the list of frequent items.
2. Create the root of an FP-tree, T, and label it as "null". For each transaction Trans in DB do the following:
   - Select the frequent items in Trans and sort them according to the order of FList. Let the sorted frequent-item list in Trans be [p | P], where p is the first element and P is the remaining list. Call insert tree([p | P], T).
   - The function insert tree([p | P], T) is performed as follows. If T has a child N such that N.item-name = p.item-name, then increment N 's count by 1; else create a new node N , with its count initialized to 1, its parent link linked to T , and its node-link linked to the nodes with the same item-name via the node-link structure. If P is nonempty, call insert tree (P, N) recursively.

By using this algorithm, the FP-tree is constructed in two scans of the database. The first scan collects and sort the set of frequent items, and the second constructs the FP-Tree.

After constructing the FP-Tree it is possible to mine it to find the complete set of frequent patterns.

Algorithm 2: FP-Growth

Input: A database DB, represented by FP-tree constructed according to Algorithm 1, and a minimum support threshold.

Output: The complete set of frequent patterns.

Method: call FP-growth(FP-tree, null).
Procedure FP-growth(Tree, a) {
(01) if Tree contains a single prefix path then // Mining single prefix-path FP-tree {
(02) let P be the single prefix-path part of Tree;
(03) let Q be the multipath part with the top branching node replaced by a null root;
(04) for each combination (denoted as ß) of the nodes in the path P do
(05) generate pattern ß ? a with support = minimum support of nodes in ß;
(06) let freq pattern set(P) be the set of patterns so generated;
}
(07) else let Q be Tree;
(08) for each item ai in Q do { // Mining multipath FP-tree
(09) generate pattern ß = ai ? a with support = ai .support;
(10) construct ß's conditional pattern-base and then ß's conditional FP-tree Tree ß;
(11) if Tree ß = Ø then
(12) call FP-growth(Tree ß , ß);
(13) let freq pattern set(Q) be the set of patterns so generated;
}
(14) return(freq pattern set(P) ? freq pattern set(Q) ? (freq pattern set(P) × freq pattern set(Q)))
}

4. What is time series data mining? **[MODEL QUESTION]**

Answer:

A time series is a sequence of data points recorded at specific time points - most often in regular time intervals (seconds, hours, days, months etc.). Every organization generates a high volume of data every single day – be it sales figure, revenue, traffic, or operating cost. Time series data mining can generate valuable information for long-term business decisions, yet they are underutilized in most organizations. Below is a list of few possible ways to take advantage of time series datasets:

Trend analysis: Just plotting data against time can generate very powerful insights. One very basic use of time-series data is just understanding temporal pattern/trend in what is being measured. In businesses it can even give an early indication on the overall direction of a typical business cycle.

Outlier/anomaly detection: An outlier in a temporal dataset represents an anomaly. Whether desired (e.g. profit margin) or not (e.g. cost), outliers detected in a dataset can help prevent unintended consequences.

Examining shocks/unexpected variation: Time-series data can identify variations (expected or unexpected) and abnormalities, detect signals in the noise.

Association analysis: By plotting bivariate/multivariate temporal data it is easy (just visually) to identify associations between any two features (e.g., profit vs sales). This association may or may not imply causation, but this is a good starting point in selecting input features that impact output variables in more advanced statistical analysis.

Forecasting: Forecasting future values using historical data is a common methodological approach – from simple extrapolation to sophisticated stochastic methods such as ARIMA.

Predictive analytics: Advanced statistical analysis such as panel data models (fixed and random effects models) rely heavily on multi-variate longitudinal datasets. These types of analysis help in business forecasts, identify explanatory variables, or simply help understand associations between features in a dataset.

5. Explain Periodicity Analysis for time related sequence data. [MODEL QUESTION]

Answer:

"What is periodicity analysis?" Periodicity analysis is the mining of periodic patterns, that is, the search for recurring patterns in time-related sequence data. Periodicity analysis can be applied to many important areas. For example, seasons, tides, planet trajectories, daily power consumptions, daily traffic patterns, and weekly TV programs all present certain periodic patterns. Periodicity analysis is often performed over time-series data, which consists of sequences of values or events typically measured at equal time intervals (e.g., hourly, daily, weekly). It can also be applied to other time-related sequence data where the value or event may occur at a nonequal time interval or at any time (e.g., on-line transactions). Moreover, the items to be analyzed can be numerical data, such as daily temperature or power consumption fluctuations, or categorical data (events), such as purchasing a product or watching a game.

The problem of mining periodic patterns can be viewed from different perspectives. Based on the coverage of the pattern, we can categorize periodic patterns into full versus partial periodic patterns:

- **A full periodic pattern** is a pattern where every point in time contributes (precisely or approximately) to the cyclic behavior or a time related sequence. For example, all of the days in the year approximately contribute to the season cycle of the year.
- **A partial periodic pattern** specifies the periodic behavior of a time-related sequence at some but not all of the points in time. For example, Sandy reads the New York Times from 7:00 to 7:30 every weekday morning, but her activities at other times do not have much regularity. Partial periodicity is a looser form of periodicity than full periodicity and occurs more commonly in the real world.

Based on the precision of the periodicity, a pattern can be either synchronous or asynchronous, where the former requires that an event occur at a relatively fixed offset in each "stable" period, such as 3 p.m. every day, whereas the latter allows that the event fluctuates in a somewhat loosely defined period. A pattern can also be either precise or approximate, depending on the data value or the offset within a period. For example, if Sandy reads the newspaper at 7:00 on some days, but at 7:10 or 7:15 on others, this is an approximate periodic pattern.

Techniques for full periodicity analysis for numerical values have been studied in signal analysis and statistics. Methods lime FFT (Fast Fourier Transformation) are commonly used to transform data from the time domain to the frequency domain in order to facilitate such analysis.

Mining partial, categorical, and asynchronous periodic patterns poses more challenging problems in regards to the development of efficient data mining solutions. This is because most statistical methods or those relying on time-to-frequency domain transformations are either inapplicable or expensive at handling such problems.
Take mining partial periodicity as an example. Because partial periodicity mixes periodic events and non-periodic events together in the same period, a time-to-frequency transformation method, such as FFT, becomes ineffective because it treats the time series as an inseparable flow of values. Certain periodicity detection methods can uncover some partial periodic patterns, but only if the period, length, and timing of the segment (subsequence of interest) in the partial patterns have certain behaviors and are explicitly specified. For the newspaper reading example, we need to explicitly specify details such as "Find the regular activities of Sandy during the half-hour after 7:00 for a period of 24 hours." A naïve adaptation of such methods to the partial periodic pattern mining problem would be prohibitively expensive, requiring their application to a huge number of possible combinations of the three parameters of period, length, and timing.
Most of the studies on mining partial periodic patterns apply the Apriori property heuristic and adopt some variations of Apriori-like mining methods. Constraints can also be pushed deep into the mining process. Studies have also been performed on the efficient mining of partially periodic events patterns or asynchronous periodic patterns with unknown or with approximate periods.
Mining partial periodicity may lead to the discovery of cyclic or periodic association rules, which are rules that associate a set of events that occur periodically. An example of a periodic association rule is "Based on day-to-day transactions, if afternoon tea is well received between 3:00 to 5:00 p.m., dinner will sell well between 7:00 to 9:00 p.m. on weekends."
Due to the diversity of applications of time-related sequence data, further development of efficient algorithms for mining various kinds of periodic patterns in sequence databases is desired.

6. What are applications of data mining? **[MODEL QUESTION]**

Answer:

Data mining is widely used in diverse areas. There are a number of commercial data mining system available today and yet there are many challenges in this field. In this tutorial, we will discuss the applications and the trend of data mining.

Data Mining Applications:

Here is the list of areas where data mining is widely used –

- Financial Data Analysis
- Retail Industry
- Telecommunication Industry
- Biological Data Analysis
- Other Scientific Applications
- Intrusion Detection

Financial Data Analysis

The financial data in banking and financial industry is generally reliable and of high quality which facilitates systematic data analysis and data mining. Some of the typical cases are as follows –

- Design and construction of data warehouses for multidimensional data analysis and data mining.
- Loan payment prediction and customer credit policy analysis.
- Classification and clustering of customers for targeted marketing.
- Detection of money laundering and other financial crimes.

Retail Industry

Data Mining has its great application in Retail Industry because it collects large amount of data from on sales, customer purchasing history, goods transportation, consumption and services. It is natural that the quantity of data collected will continue to expand rapidly because of the increasing ease, availability and popularity of the web.

Data mining in retail industry helps in identifying customer buying patterns and trends that lead to improved quality of customer service and good customer retention and satisfaction. Here is the list of examples of data mining in the retail industry –

- Design and Construction of data warehouses based on the benefits of data mining.
- Multidimensional analysis of sales, customers, products, time and region.
- Analysis of effectiveness of sales campaigns.
- Customer Retention.
- Product recommendation and cross-referencing of items.

Telecommunication Industry

Today the telecommunication industry is one of the most emerging industries providing various services such as fax, pager, cellular phone, internet messenger, images, e-mail, web data transmission, etc. Due to the development of new computer and communication technologies, the telecommunication industry is rapidly expanding. This is the reason why data mining is become very important to help and understand the business.

Data mining in telecommunication industry helps in identifying the telecommunication patterns, catch fraudulent activities, make better use of resource, and improve quality of service. Here is the list of examples for which data mining improves telecommunication services –

- Multidimensional Analysis of Telecommunication data.
- Fraudulent pattern analysis.
- Identification of unusual patterns.
- Multidimensional association and sequential patterns analysis.
- Mobile Telecommunication services.
- Use of visualization tools in telecommunication data analysis.

Biological Data Analysis

In recent times, we have seen a tremendous growth in the field of biology such as genomics, proteomics, functional Genomics and biomedical research. Biological data mining is a very important part of Bioinformatics. Following are the aspects in which data mining contributes for biological data analysis –

- Semantic integration of heterogeneous, distributed genomic and proteomic databases.
- Alignment, indexing, similarity search and comparative analysis multiple nucleotide sequences.
- Discovery of structural patterns and analysis of genetic networks and protein pathways.
- Association and path analysis.
- Visualization tools in genetic data analysis.

Other Scientific Applications

The applications discussed above tend to handle relatively small and homogeneous data sets for which the statistical techniques are appropriate. Huge amount of data have been collected from scientific domains such as geosciences, astronomy, etc. A large amount of data sets is being generated because of the fast numerical simulations in various fields such as climate and ecosystem modeling, chemical engineering, fluid dynamics, etc. Following are the applications of data mining in the field of Scientific Applications –

- Data Warehouses and data preprocessing.
- Graph-based mining.
- Visualization and domain specific knowledge.

Intrusion Detection

Intrusion refers to any kind of action that threatens integrity, confidentiality, or the availability of network resources. In this world of connectivity, security has become the major issue. With increased usage of internet and availability of the tools and tricks for intruding and attacking network prompted intrusion detection to become a critical component of network administration. Here is the list of areas in which data mining technology may be applied for intrusion detection –

- Development of data mining algorithm for intrusion detection.
- Association and correlation analysis, aggregation to help select and build discriminating attributes.
- Analysis of Stream data.
- Distributed data mining.
- Visualization and query tools.

Data Mining System Products

There are many data mining system products and domain specific data mining applications. The new data mining systems and applications are being added to the previous systems. Also, efforts are being made to standardize data mining languages.

Choosing a Data Mining System

The selection of a data mining system depends on the following features –

- **Data Types** – The data mining system may handle formatted text, record-based data, and relational data. The data could also be in ASCII text, relational database data or data warehouse data. Therefore, we should check what exact format the data mining system can handle.
- **System Issues** – We must consider the compatibility of a data mining system with different operating systems. One data mining system may run on only one operating system or on several. There are also data mining systems that provide web-based user interfaces and allow XML data as input.
- **Data Sources** – Data sources refer to the data formats in which data mining system will operate. Some data mining system may work only on ASCII text files while others on multiple relational sources. Data mining system should also support ODBC connections or OLE DB for ODBC connections.
- **Data Mining functions and methodologies** – There are some data mining systems that provide only one data mining function such as classification while some provides multiple data mining functions such as concept description, discovery-driven OLAP analysis, association mining, linkage analysis, statistical analysis, classification, prediction, clustering, outlier analysis, similarity search, etc.
- **Coupling data mining with databases or data warehouse systems** – Data mining systems need to be coupled with a database or a data warehouse system. The coupled components are integrated into a uniform information processing environment. Here are the types of coupling listed below –
  - No coupling
  - Loose Coupling
  - Semi tight Coupling
  - Tight Coupling
- **Scalability** – There are two scalability issues in data mining –
  - Row (Database size) Scalability – A data mining system is considered as row scalable when the number or rows are enlarged 10 times. It takes no more than 10 times to execute a query.
  - Column (Dimension) Salability – A data mining system is considered as column scalable if the mining query execution time increases linearly with the number of columns.
- **Visualization Tools** – Visualization in data mining can be categorized as follows –
  - Data Visualization
  - Mining Results Visualization
  - Mining process visualization
  - Visual data mining
- **Data Mining query language and graphical user interface** – An easy-to-use graphical user interface is important to promote user-guided, interactive data mining. Unlike relational database systems, data mining systems do not share underlying data mining query language.

MINING DATA STREAMS

Multiple Choice Type Questions

1. The 'Dice' operation is concerned with [WBUT 2009, 2014]
   a) Multiple runs of slice
   b) slice on more than one dimension
   c) selecting certain cells of more than one dimension
   d) two consecutive slice operations in two different dimensions

Answer: (d)

2. The major drawback of CLARANS algorithms is [WBUT 2009, 2011]
   a) it cannot handle very large volumes of data
   b) it assumes that all objects fit into the main memory, and the result is very sensitive to input order
   c) it cannot find the best clustering if any sampled medoit is not among the best *k* methods

Answer: (b)

3. A drill-down operation is concerned with [WBUT 2009, 2016, 2018]
   a) which merges cells of two dimension
   b) which merges cells of any one dimension based on the characteristics of the dimension
   c) which splits cells of two dimensions
   d) which splits cells of any one dimension based on the characteristics of the dimension

Answer: (d)

4. Parameters used for association Rule Mining are [WBUT 2010, 2018]
   a) Confidence and Support
   b) Confidence and Itemcount
   c) Support and Itemcount
   d) Support, Confidence and Itemcount

Answer: (a)

5. Two main types of clustering techniques in data mining are [WBUT 2010, 2018]
   a) Serial clustering and parallel clustering
   b) Hierarchical clustering and partitioning clustering
   c) Homogeneous clustering and heterogeneous clustering
   d) *k* –medoids clustering and *K* –means clustering

Answer: (b)

6. The algorithm which uses the concept of a train running over data to find associations of items in data mining known as [WBUT 2011]
   a) Apriority Algorithm
   b) Partition Algorithm
   c) Dynamic Item-set Counting Algorithm
   d) FP-Tree growth Algorithm

Answer: (c)

7. If we know exactly what information we need then.....would suffice, but if we vaguely know the possible patterns then.....are useful. [WBUT 2012]
   a) Data Warehouse, Data Mining techniques
   b) DBMS Query, Data Mining techniques
   c) DBMS Query, Data Warehouse applications
   d) Data Warehouse applications, Data Mining techniques

Answer: (b)

8. Association analysis is used for [WBUT 2014]
   a) transaction data analysis b) olap c) molap d) none of these

Answer: (a)

9. Which one is not a data mining task? [WBUT 2014, 2015]
   a) indexing b) classification c) clustering d) regression

Answer: (a)

10. An example of hierarchical clustering algorithm is [WBUT 2014, 2018]
   a) clarans b) C4.5 c) average linkage d) rock

Answer: (d)

11. Which frequent pattern mining technique mines without candidate generation?
   a) Partitioning b) Apriori [WBUT 2018]
   c) FP-growth d) Dynamic intensive counting

Answer: (c)

Short Answer Type Questions

1. What are the different methods of computing the best split? What are entropy gain and gain ratio? [WBUT 2012]

Answer:

Various measures are available to determine which test condition provides the best split. These are:

- Gini Index
- Entropy / Information Gain
- Classification Error

A measure used from Information Theory in the ID3 algorithm and many others used in decision tree construction is that of Entropy. Informally, the entropy of a dataset can be considered to be how disordered it is. It has been shown that entropy is related to information, in the sense that the higher the entropy, or uncertainty, of some data, then the more information is required in order to completely describe that data. In building a decision tree, we aim to decrease the entropy of the dataset until we reach leaf nodes at which point the subset that we are left with is pure, or has zero entropy and represents instances all of one class (all instances have the same value for the target attribute).

We measure the entropy of a dataset, S, with respect to one attribute, in this case the target attribute, with the following calculation:

Entropy (S) = $\sum_{i=1}^{C} p_i \log_2 p_i$ where Pi is the proportion of instances in the dataset that take the ith value of the target attribute, which has C different values

In order to reduce the effect of the bias resulting from the use of information gain, a variant known as **Gain Ratio** has been introduced. Gain Ratio is defined by the formula *Gain Ratio = Information Gain/Split Information* where Split Information is a value based on the column sums.

2. How is CLARANS different from CLARA? Illustrate this using small example. **[WBUT 2012]**

Answer:

CLARA (Clustering Large Applications), is an implementation of PAM in a subset of the dataset. It draws multiple samples of the dataset, applies PAM on samples, and then outputs the best clustering out of these samples.

CLARANS (Clustering Large Applications based on Randomized Search), combines the sampling techniques with PAM. The clustering process can be presented as searching a graph where every node is a potential solution, that is, a set of k medoids. The clustering obtained after replacing a medoid is called the neighbour of the current clustering. CLARANS selects a node and compares it to a user-defined number of their neighbours searching for a local minimum. If a better neighbour is found (i.e., having lower-square error), CLARANS moves to the neighbour's node and the process start again; otherwise the current clustering is a local optimum. If the local optimum is found, CLARANS starts with a new randomly selected node in search for a new local optimum.

Basically, CLARA draws multiple random samples of the input points, runs algorithm {PAM} on each sample set, and outputs the clustering with the lowest clustering cost (in terms of the sum of distances of the points to the closest medoid). Like CLARA the algorithm CLARANS is also a randomized version of PAM. In each local improvement step, CLARANS does not consider all possible swaps of one medoid and one non-medoid like {PAM} does, but it only considers a randomly chosen subset of all possible swaps. CLARANS computes a certain fixed number of local minima in this manner and outputs the clustering with the minimum clustering cost among all these local minima.

3. Explain temporal data mining with an example. How is it different from spatial data mining? **[WBUT 2015]**

Answer:

Temporal data mining refers to the extraction of implicit, non-trivial, and potentially useful abstract information from large collections of temporal data. Temporal data are sequences of a primary data type, most commonly numerical or categorical values and sometimes multivariate or composite information. Examples of temporal data are regular time series (e.g., stock ticks, EEG), event sequences (e.g., sensor readings, packet traces, medical records, weblog data), and temporal databases (e.g., relations with time stamped tuples, databases with versioning). The common factor of all these sequence types is the total ordering of their elements. They differ on the type of primary information, the

regularity of the elements in the sequence, and on whether there is explicit temporal information associated to each element (e.g., timestamps).
Spatial Data Mining is the process of discovering interesting and previously unknown, but potentially useful patterns from large spatial datasets. Extracting interesting and useful patterns from spatial datasets is more difficult than extracting the corresponding patterns from traditional numeric and categorical data due to the complexity of spatial data types, spatial relationships, and spatial autocorrelation.

4. Introduce the concept of splitting attribute and splitting criterions. [WBUT 2018]

Answer:

Drawing a decision tree from the available dataset involves **splitting attribute** of each node. Each branch will have a possible value of the corresponding attribute. Splitting attribute is the most informative attribute among all the attributes. To select the most informative attribute, an algorithm uses a factor called Entropy. Goodness of a split is determined by information gain. Attribute with the maximum information gain is considered to split. Dataset is split for all the attributes values.
The **splitting criterion** tells us which attribute to test at node N by determining the "best" way to separate or partition the tuples in D into individual classes. The splitting criterion also tells us which branches to grow from node N with respect to the outcomes of the chosen test.

5. Explain about Filters for Mining Models. [MODEL QUESTION]

Answer:

Data-based model filtering helps you create mining models that use subsets of data in a mining structure. Filtering gives you flexibility when you design your mining structures and data sources, because you can create a single mining structure, based on a comprehensive data source view. You can then create filters to use only a part of that data for training and testing a variety of models, instead of building a different structure and related model for each subset of data.
For example, you define the data source view on the Customers table and related tables. Next, you define a single mining structure that includes all the fields you need. Finally, you create a model that is filtered on a particular customer attribute, such as Region. You can then easily make a copy of that model, and change just the filter condition to generate a new model based on a different region.
Some real-life scenarios where you might benefit from this feature include the following:

- Creating separate models for discrete values such as gender, regions, and so forth. For example, a clothing store might use customer demographics to build separate models by gender, even though the sales data comes from a single data source for all customers.
- Experimenting with models by creating and then testing multiple groupings of the same data, such as ages 20-30 vs. ages 20-40 vs. ages 20-25.
- Specifying complex filters on nested table contents, such as requiring that a case be included in the model only if the customer has purchased at least two of a particular item.

Long Answer Type Questions

1. In data warehouse technology explain ROLAP, MOLAP and HOLAP techniques of implementing a multidimensional view. [WBUT 2009]

OR,

Compare between HOLAP, ROLAP and MOLAP. [WBUT 2013]

Answer:

Relational OLAP (ROLAP) servers: These are the intermediate servers that stand in between a relational back-end server and client front-end tools. They use a relational or extended-relational DBMS to store and manage warehouse data, and OLAP middleware to support missing pieces. ROLAP servers include optimization for each DBMS back-end, implementation of aggregation navigation logic, and additional tools and services. ROLAP technology tends to have greater scalability than MOLAP technology.

Multidimensional OLAP (MOLAP) servers: These servers support multidimensional views of data through array-based multidimensional storage engines. They map multidimensional views directly to data cube array structures. For example, Essbase of Arbor is a MOLAP server. The advantage of using a data cube is that it allows fast indexing to precomputed summarized data. Notice that with multidimensional data stores, the storage utilization may be low if the data set is sparse. In such cases, sparse matrix compression techniques should be explored. Many OLAP servers adopt a two-level storage representation to handle sparse and dense data sets: the dense subcubes are identified and stored as array structures, while the sparse subcubes employ compression technology for efficient storage utilization.

Hybrid OLAP (HOLAP) servers: The hybrid OLAP approach combines ROLAP and MOLAP technology, benefiting from the greater scalability of ROLAP and the faster computation of MOLAP. For example, a HOLAP server may allow large volumes of detail data to be stored in a relational database, while aggregations are kept in a separate MOLAP store. The Microsoft SQL Server 7.0 OLAP Services supports a hybrid OLAP server.

2. a) Why do we need to have separate Data Warehouse for OLAP applications?
b) Starting with the base cuboid [day, item, branch] what specific OLAP operations (e.g. slice for time = 'year') should be performed in order to list the total sales of each branch in the year 2008? [WBUT 2010]

Answer:

a) A major reason for using a data warehouse in OLAP is to help promote the high performance of both systems. An operational database is designed and tuned from known tasks and workloads, such as indexing and hashing using primary keys, searching for particular records, and optimizing "canned" queries. On the other hand, data warehouse queries are often complex. They involve the computation of large groups of data at summarized levels, and may require the use of special data organization, access, and implementation methods based on multidimensional views. Processing OLAP queries in operational databases would substantially degrade the performance of operational tasks.

Moreover, an operational database supports the concurrent processing of several transactions. An OLAP query often needs read-only access of data records for summarization and aggregation. Concurrency control and recovery mechanisms, if applied for such OLAP operations, may jeopardize the execution of concurrent transactions and thus substantially reduce the throughput of an OLTP system.

b)

1. roll up from day to month to year
2. slice for year = "2008"
3. roll up on item from individual item to all
4. slice for item = "all"
5. get the list of total sales by each branch in 2008

3. a) What is tree pruning? What are the different tree pruning techniques?
[WBUT 2013, 2014]

b) Evaluate Information Gain and Gain Ratio with suitable example. [WBUT 2013]

Answer:

a) When a decision tree is built many of the branches will reflect anomalies in the training data due to noise or outliers. Tree pruning methods address this problem of over fitting the data. There are two common approaches to tree pruning.

- **Pre-pruning:** In the pre-pruning approach, a tree is "pruned" by halting its construction early. Upon halting, the node becomes leaf. The leaf may hold most frequent class among the subsets tuples or the probability distribution of those tuples. When constructing a tree, attribute selection measures such as statistical significance, information gain, gini index and so on can be used to access the goodness of a split. If partitioning the tuple at node would result in a split that falls below a pre-specified threshold, then further partitioning of a given subset is halted. There are difficulties, however in choosing an appropriate threshold. High threshold could result in oversimplified trees; whereas low thresholds could result in very little simplification.
- **Post-pruning:** Post-pruning removes sub trees from a "fully grown" tree. A sub tree at a given node is pruned by removing its branches and replacing it with leaf. The leaf is labeled with the most frequent class among the sub tree being replaced. The "best" pruned tree is one that minimizes the encoding bits. This method adopts MDL (Minimum Description Length) principle. The basic idea is that the simplest solution is preferred. Alternatively, pre-pruning and post-pruning may be interleaved for a combined approach. Post-pruning requires more computation than pre-pruning, yet generally leads to a more reliable tree. Although pruned tree tends to be more compact than their unpruned counterparts, they may still be rather large and complex. Decision tree can suffer from repetition and replication.

b) Information gain (IG) is utilized by the ID3/C4.5/YaDT family of algorithms as a measure of the effectiveness of an attribute in classifying the training data. Information gain is computed by measuring the difference between the entropy of the dataset before

the split and the overall entropy of the dataset after the split. The attribute with the highest information gain is assumed to be the best splitting attribute and is the first attribute to split the dataset. Discrete attributes normally appear once in a decision tree, however, continuous attributes may appear more than once along any path through the tree. Thus IG = entropy(parent) – [average entropy(children)].
In order to reduce the effect of the bias resulting from the use of information gain, a variant known as **Gain Ratio** has been introduced. Gain Ratio is defined by the formula *Gain Ratio = Information Gain/Split Information* where Split Information is a value based on the column sums.

4. a) Define a frequent set. Show that every subset of any item set must contain either a frequent set or a border set.
b) Define support and confidence.
c) Find all the frequent sets using Apriori algorithm of the following database:

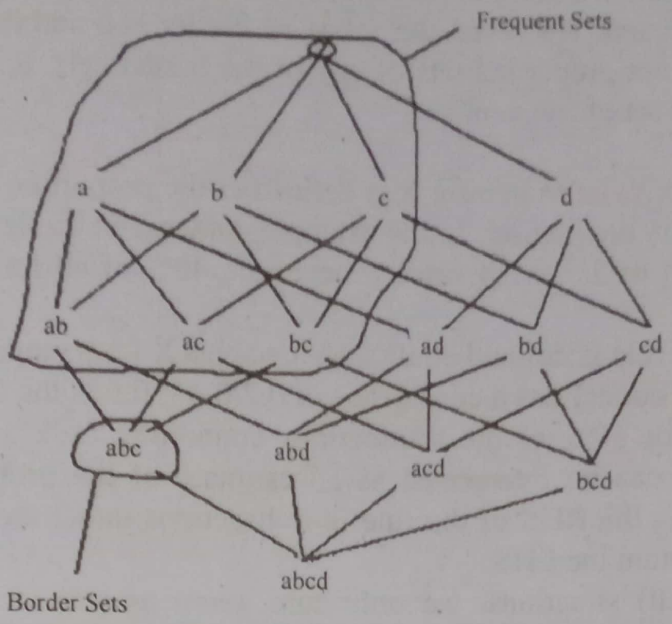
A1	A2	A3	A4	A5	A6	A7	A8	A9
1	0	0	0	1	1	0	1	0
0	1	0	1	0	0	0	1	0
0	0	0	1	1	0	1	0	0
0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	0	0
0	1	1	1	0	0	0	0	0
0	1	0	0	0	1	1	0	1
0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	0
0	0	1	0	1	0	1	0	0
0	0	0	0	1	1	0	1	0
0	1	0	1	0	1	1	0	0
1	0	1	0	1	0	1	0	0
0	1	1	0	0	0	0	0	1
0	0	1	0	1	0	1	0	0

Assume σ = 20%. **[WBUT 2015]**

Answer:

a) 1st Part:

A set of one or more items belonging to set I is termed as an itemset. An itemset X is a frequent itemset if support of X is greater than the user specified support threshold.
An itemset X is called a border set if X is not frequent but all of its subsets are frequent.



2nd Part:

An itemset is a border set if it is not a frequent set, but all its proper subsets are frequent sets.

One can see that if X is an infrequent itemset, then it must have a subset (not necessarily a proper subset) that is a border set. It is easy to derive a proof for this. Since X is not frequent, it is possible that it is a border set. In that case, the proof is done. Let us assume that X is not a border set too. Hence, there exists at least one proper subset of cardinality $|X|-1$ that is not frequent, say X'. If X' is a border set, then the proof is complete. Let us, hence, assume that X' is not a border set. We recursively construct $X, X', X'', \ldots,$ and so on, having the common property that neither of these is a frequent set nor a border set and this construction process terminates when we get a set which is a border. This construction process must terminates in a finite number of steps as we are decreasing the size of the sets by 1 in every step. In most peculiar case, we may land up in a singleton itemset (the empty itemset is always considered to be a frequent set).

Note that if we know the set of all maximal frequent sets of a given T with respect to a σ, then we can find the set of all frequent sets without any extra scan of the database. Thus, the set of all maximal frequent sets can act as a compact representation of the set of all frequent sets. However, if we require the frequent sets together with their respective support values in T, then we have to make one more database pass to derive the support values when the set of all maximal frequent sets is known. It is thus easy to characterize the class of frequent sets and the class of infrequent sets in terms of the boundary sets between these two classes. Note that some maximal frequent sets are proper subsets of some border sets. Similarly, it is possible that a proper subset of a border set, of cardinality one less than the border set, is not necessarily always maximal.

Thus, we cannot establish a definite relationship between the set of maximal frequent sets and the set of border sets. However, the set of all border sets and the set of the macimal frequent, which are not proper subsets of any of the border sets, jointly provide a better representation of the set of frequent sets.

b) The **support** supp(X) of an itemset X is defined as the proportion of transactions in the data set which contain the itemset. In the example database in Table 1, the itemset {milk, bread} has a support of 2/5 = 0.4 since it occurs in 40% of all transactions (2 out of 5 transactions).
The **confidence** of a rule is defined conf(X) Y) = supp(X [Y)/supp(X). For example, the rule {milk, bread}) {butter} has a confidence of 0.2/0.4 = 0.5 in the database in the Table, which means that for 50% of the transactions containing milk and bread the rule is correct. Confidence can be interpreted as an estimate of the probability P(Y |X), the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.
In many (but not all) situations, we only care about association rules or causalities involving sets of items that appear frequently in baskets. For example, we cannot run a good marketing strategy involving items that no one buys anyway. Thus, much data mining starts with the assumption that we only care about sets of items with high support; i.e., they appear together in many baskets. We then find association rules or causalities only involving a high-support set of items must appear in at least a certain percent of the baskets, called the support threshold. We use the term **frequent itemset** for "a set S that appears in at least fraction s of the baskets," where s is some chosen constant, typically 0.01 or 1%.

c) The item count for the transactions for different passes given below:
Pass 1:

Item Id	Count
1	2
2	6
3	6
4	4
5	8
6	5
7	7
8	4
9	2

Since σ = 20%., we discard item 1 and 9.
So our set is L1 :={ 2, 3, 4, 5, 6, 7, 8}
Pass 2:
Generate candidate for k=2
C2:={{2,3},{2,4},{2,5},{2,6},{2,7},{2,8},{3,4},{3,5},{3,6},{3,7},{3,8},{4,5},{4,6},{4,7},{4,8},{5,6},{5,7},{5,8}, {6, 7}, {6, 8}, {7, 8}}

The table below shows only those for which σ = 20%. holds.

Item Id	Count
2,3	3
2,4	3
3,5	3
3,7	3
5,6	3
5,7	5
6,7	3

Pass 3:

Generate Candidate for k: = 3

C3:={{2,3,4},{2,3,5}, {2,3,6}, {2,3,7}, {2,4,6}, {2, 4, 7}, {2, 5, 6}, {2, 5, 7}, {3,4,6}, {3,5,7}, {4,5,6}, {4,5,7},{5,6,7}}

Finally we see that for {3,5,7} σ = 20%. holds.

5. a) What is strong rule? Derive strong rules from the following transaction database considering minimum support as 40% and minimum confidence as 70%.

TID	Item _ID
1	$\{I_1, I_2, I_3, I_4, I_5\}$
2	$\{I_2, I_3,\}$
3	$\{I_1, I_2, I_6\}$
4	$\{I_2, I_1, I_7\}$
5	$\{I_1, I_6, I_8,\}$

b) Explain CLS algorithm to construct a decision tree. **[WBUT 2017]**

Answer:

a) 1st part:

A strong rule is one that satisfies both minimum support and minimum confidence.

2nd part:

For support of 40%, threshold is at least 2 transaction.

Pass(k)	Candidate k itemset	Frequent k-itemsets
K=1	I_1=4, I_2=4, I_3=2, I_4=1, I_5=1, I_6=2, I_7=1, I_8=1	I_1, I_2, I_3, I_6
K=2	$\{I_1, I_2\}$(3), $\{I_1, I_3\}$(1), $\{I_1, I_6\}$(2), $\{I_2, I_3,\}$(2), $\{I_2, I_6\}$(1), $\{I_3, I_6\}$(0)	$\{I_1, I_2\}$, $\{I_2, I_3\}\{I_1, I_6\}$
K=3	$\{I_1, I_2, I_3\}$(1), $\{I_1, I_2, I_6\}$(1)	-

All frequent itemsets: $\{I_1, I_2\}$, $\{I_2, I_3\}\{I_1, I_6\}$, I_1, I_2, I_3, I_6

Association rules:

$\{I_1, I_2\}$ would generate: $I_1 \rightarrow I_2$(3/5, 3/4)and$I_2 \rightarrow I_1$(3/5, 3/4)

$\{I_2, I_3\}$ would generate: $I_2 \rightarrow I_3$(2/5, 2/4)$I_3 \rightarrow I_2$(2/5, 2/2)

$\{I_1, I_6\}$ would generate: $I_1 \rightarrow I_6$(2/5, 2/4)$I_6 \rightarrow I_1$(2/5, 2/2)

Strong rules are:
With the confidence threshold set to 70%, the Strong Association Rules are:
$I_1 \rightarrow I_2$ (0.6, .75),
$I_2 \rightarrow I_1$ (0.6, 0.75),
$I_3 \rightarrow I_2$ (0.4, 1)
$I_6 \rightarrow I_1$ (0.4, 1)

b) CLS constructs a decision tree that attempts to minimize the cost of classifying an object. This cost has components of two types: the measurement cost of determining the value of property A exhibited by the object, and the misclassification cost of deciding that the object belongs to class J when its real class is K. CLS uses a look-ahead strategy similar to minimax. At each stage, CLS explores the space of possible decision trees to a fixed depth, chooses an action to minimize cost in this limited space, then moves one level down in the tree. Depending on the depth of look-ahead chosen, CLS can require a substantial amount of computation, but has been able to unearth subtle patterns in the objects shown to it.

6. Explain Slicing, Dicing, Roll-up and Drill-down with a suitable example.
[WBUT 2017]

Answer:
Consider the following cube illustrating temperature of certain days recorded weekly:

Temperature	64	65	68	69	70	71	72	75	80	81	83	85
Week 1	1	0	1	0	1	0	0	0	0	0	1	0
Week 2	0	0	0	1	0	0	1	2	0	1	0	0

Slice performs a selection on one dimension of the given cube, thus resulting in a subcube.
In the cube example above, if we make the selection, temperature=cool we will obtain the following cube:

	cool
day 1	0
day 2	0
day 3	0
day 4	0
day 5	1
day 6	0
day 7	1
day 8	0
day 9	1
day 10	0
day 11	0
day 12	0
day 13	0
day 14	0

The **dice** operation defines a subcube by performing a selection on two or more dimensions. Applying the selection (time = day 3 OR time = day 4) AND (temperature =

cool OR temperature = hot) to the original cube we get the following subcube (still two-dimensional):

	cool	hot
day 3	0	1
day 4	0	0

The **roll-up** operation (also called drill-up or aggregation operation) performs aggregation on a data cube, either by climbing up a concept hierarchy for a dimension or by climbing down a concept hierarchy, i.e. dimension reduction.
Assume we want to set up levels (hot(80-85), mild(70-75), cold(64-69)) in temperature from the above cube. To do this we have to group columns and add up the values according to the concept hierarchy. By doing this we obtain the following cube:

temperature	cool	mild	hot
Week 1	2	1	1
Week 2	1	3	1

The **roll down** operation (also called drill down) is the reverse of roll up. It navigates from less detailed data to more detailed data. It can be realized by either stepping down a concept hierarchy for a dimension or introducing additional dimensions.
Performing roll down operation on the same cube mentioned above

	cool	mild	hot
day 1	0	0	0
day 2	0	0	0
day 3	0	0	1
day 4	0	1	0
day 5	1	0	0
day 6	0	0	0
day 7	1	0	0
day 8	0	0	0
day 9	1	0	0
day 10	0	1	0
day 11	0	1	0
day 12	0	1	0
day 13	0	0	1
day 14	0	0	0

7. Write short notes on the following:

a) GSP algorithm [WBUT 2009]
b) ROCK vs. CACTUS [WBUT 2011]
c) GSP vs. SPADE [WBUT 2011]
d) Decision tree construction with presorting [WBUT 2011]
e) WUM [WBUT 2012, 2018]
f) C4.5 [WBUT 2014]
g) Top-down approach [WBUT 2014]

[WBUT 2016]
[WBUT 2017]

h) Backpropagation algorithm
i) Supervised vs. unsupervised learning

Answer:

a) GSP algorithm:

The GSP algorithm consists of two phases and is described below:

Phase 1:

- Scan over the database to identify all the frequent items, i.e., 1-element sequences

Phase 2:

- Iteratively scan over the database to discover all frequent sequences. Each iteration discovers all the sequences with the same length.
- In the iteration to generate all *k*-sequences
  - Generate the set of all candidate *k*-sequences, C_k, by joining two (*k-1*)-sequences if only their first and last items are different
  - Prune the candidate sequence if any of its *k-1* contiguous subsequence is not frequent
  - Scan over the database to determine the support of the remaining candidate sequences
- Terminate when no more frequent sequences can be found

b) ROCK vs. CACTUS:

The ROCK (Robust Clustering using links) algorithm is based on the principle of hierarchical clustering. First, a random sample of objects is chosen. These objects are clustering to the desired number of clusters, and then the remaining objects are assigned to the created clusters. The method uses a graph concept, whose main terms are neighbors and links. A neighbor of a certain object is such an object to which similarity with the investigated object is equal to or greater than a predefined threshold. A link between two objects is the number of common neighbors of these objects. The principle of the ROCK methods lies in maximization of the function which takes into account both maximization of sums of links for the objects from the same cluster, and minimization of sums of links for the objects from different clusters.

Algorithm CACTUS (Categorical Clustering Using Summaries) is based on the idea of the common occurrences of certain categories of different variables. If the difference in the number of occurrences for the categories vkt and vlu of the k-th and l-th variable, and the expected frequency (on the assumption of uniform distribution in the frame of the certain categories of the remaining variables, and the assumption of the independency) is greater than a user-defined threshold, the categories are strongly connected. The algorithm has three phases: summarization, clustering and verification. During clustering, the candidates for clusters are chosen from which the final clusters are determined in the verification phase.

c) GSP vs. SPADE:

GSP (Generalized Sequential Pattern) is a mining algorithm which is outlined below
Initially, every item in DB is a candidate of length-1

- for each level (i.e., sequences of length-k) do

- scan database to collect support count for each candidate sequence
- generate candidate length-(k+1) sequences from length-k frequent sequences using Apriori

- repeat until no frequent sequence or no candidate can be found

Its major strength: Candidate pruning by Apriori

SPADE (Sequential Pattern Discovery using Equivalent Class) is a vertical format sequential pattern mining method as outlined below:

- A sequence database is mapped to a large set of
  - Item: <SID, EID>
- Sequential pattern mining is performed by
  - growing the subsequences (patterns) one item at a time by Apriori candidate generation

SPADE uses only simple temporal join operation on id-lists. As the length of a frequent sequence increases, the size of its id-list decreases, resulting in very fast joins.

No complicated hash-tree structure is used, and no overhead of generating and searching of subsequences is incurred. These structures typically have very poor locality. On the other hand SPADE has excellent locality, since a join requires only a linear scan of two lists.

As the minimum support is lowered, more and larger frequent sequences are found. GSP makes a complete dataset scan for each iteration. SPADE on the other hand restricts itself to usually only three scans. It thus cuts down the I/O costs.

d) Decision tree construction with presorting:

SLIQ (Supervised Learning in Quest) developed by IBM's Quest project team, is a decision tree classifier designed to classify large training data. It uses a pre-sorting technique in the tree-growth phase. This helps avoid costly sorting at each node. SLIQ keeps a separate sorted list for each continuous attribute and a separate list called class list. An entry in the class list corresponds to a data item, and has a class label and name of the node it belongs in the decision tree. An entry in the sorted attribute list has an attribute value and the index of data item in the class list. SLIQ grows the decision tree in breadth-first manner. For each attribute, it scans the corresponding sorted list and calculates entropy values of each distinct values of all the nodes in the frontier of the decision tree simultaneously. After the entropy values have been calculated for each attribute, one attribute is chosen for a split for each nodes in the current frontier, and they are expanded to have a new frontier. Then one more scan of the sorted attribute list is performed to update the class list for the new nodes.

While SLIQ handles disk-resident data that are too large to fit in memory, it still requires some information to stay memory-resident which grows in direct proportion to the number of input records, putting a hard-limit on the size of training data. In a newer version called SPRINT (Scalable Parallelizable Induction of decision Trees) the memory restrictions have been removed.

e) WUM:

The ease and speed with which business transactions can be carried out over the Web has been a key driving force in the rapid growth of electronic commerce. Specifically, e-commerce activity that involves the end user is undergoing a significant revolution. The ability to track users' browsing behavior down to individual mouse clicks has brought the vendor and end customer closer than ever before. It is now possible for a vendor to personalize his product message for individual customers at a massive scale, a phenomenon that is being referred to as mass customization. The scenario described above is one of many possible applications of **Web Usage mining (WUM)**, which is the process of applying data mining techniques to the discovery of usage patterns from Web data, targeted towards various applications. Data mining efforts associated with the Web, called Web mining, can be broadly divided into three classes, i.e. content mining, usage mining, and structure mining.

f) C4.5:

C4.5 builds decision trees from a set of training data in the same way as ID3, using the concept of information entropy. The training data is a set S = {s1, s2, ...} of already classified samples. Each sample si consists of a p-dimensional vector (x1,i, x2,i, ...,xp,i), where the xj represent attributes or features of the sample, as well as the class in which si falls.

At each node of the tree, C4.5 chooses the attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. The splitting criterion is the normalized information gain (difference in entropy). The attribute with the highest normalized information gain is chosen to make the decision. The C4.5 algorithm then recurs on the smaller sublists.

This algorithm has a few base cases:

- All the samples in the list belong to the same class. When this happens, it simply creates a leaf node for the decision tree saying to choose that class.
- None of the features provide any information gain. In this case, C4.5 creates a decision node higher up the tree using the expected value of the class.
- Instance of previously-unseen class encountered. Again, C4.5 creates a decision node higher up the tree using the expected value.

Pseudocode

In pseudocode, the general algorithm for building decision trees is:

1. Check for base cases
2. For each attribute a
   - Find the normalized information gain ratio from splitting on a
3. Let a_best be the attribute with the highest normalized information gain
4. Create a decision node that splits on a_best
5. Recur on the sublists obtained by splitting on a_best, and add those nodes as children of node

g) Top-down approach:

A top-down method approaches a problem from the high level view a system.

For example, in a bottom-up business example a company will look at sales and customer data to extract any patterns.

A top-down approach could model a corporation and its strategy and ask the following: Who are the target customers? What is the supply chain? What is the marketing? How are the departments divided? What are the corporate sales policies? How are sales team incentivized?

There are external factors that also need to be considered in this example such as competition from other companies and the overall economy. If all the divisions of the company are not aligned with the corporate strategy then it will be easy to understand that the target sales and customer reach will not be optimal. By starting from the top and then deconstructing the parts and understanding the interactions between the parts one can gain an understanding of what type of customers can be reached and attracted.

h) Backpropagation algorithm:

Backpropagation, is a common method of training artificial neural networks and used in conjunction with an optimization method such as gradient descent. The algorithm repeats a two phase cycle, propagation and weight update. When an input vector is presented to the network, it is propagated forward through the network, layer by layer, until it reaches the output layer. The output of the network is then compared to the desired output, using a loss function, and an error value is calculated for each of the neurons in the output layer. The error values are then propagated backwards, starting from the output, until each neuron has an associated error value which roughly represents its contribution to the original output.

Backpropagation uses these error values to calculate the gradient of the loss function with respect to the weights in the network. In the second phase, this gradient is fed to the optimization method, which in turn uses it to update the weights, in an attempt to minimize the loss function.

The importance of this process is that, as the network is trained, the neurons in the intermediate layers organize themselves in such a way that the different neurons learn to recognize different characteristics of the total input space. After training, when an arbitrary input pattern is present which contains noise or is incomplete, neurons in the hidden layer of the network will respond with an active output if the new input contains a pattern that resembles a feature that the individual neurons have learned to recognize during their training.

Backpropagation requires a known, desired output for each input value in order to calculate the loss function gradient – it is therefore usually considered to be a supervised learning method; nonetheless, it is also used in some unsupervised networks such as autoencoders. It is a generalization of the delta rule to multi-layered feedforward networks, made possible by using the chain rule to iteratively compute gradients for each layer. Backpropagation requires that the activation function used by the artificial neurons (or "nodes") be differentiable.

i) Supervised vs. unsupervised learning:

Supervised: So, if you are training your machine learning task for every input with corresponding target, it is called supervised learning, which will be able to provide target for any new input after sufficient training. Your learning algorithm seeks a function from inputs to the respective targets. If the targets are expressed in some classes, it is called classification problem. Alternatively, if the target space is continuous, it is called regression problem.

Unsupervised: Contrary, if you are training your machine learning task only with a set of inputs, it is called unsupervised learning, which will be able to find the structure or relationships between different inputs. Most important unsupervised learning is clustering, which will create different cluster of inputs and will be able to put any new input in appropriate cluster.

8. What is sequential pattern mining in data stream? **[MODEL QUESTION]**

Answer:

Sequential pattern mining is trying to find relationships between occurrences of sequential events, to find if there exists any specific order of the occurrences. We can find the sequential patterns of specific individual items also we can find the sequential patterns across different items.

Application:

- Sequential pattern mining is widely used in analyzing of DNA sequence.
- Sequential pattern can be widely used in different areas, such as mining user access patterns for the web sites, using the history of symptoms to predict certain kind of disease, also by using sequential pattern mining, the retailers can make the inventory control more efficient.

Challenges on Sequential pattern mining:

- A huge number of possible sequential patterns are hidden in databases.
- A mining algorithm should find the complete set of patterns, be highly efficient, scalable.

9. What is frequent pattern mining in data stream? **[MODEL QUESTION]**

Answer:

Frequent Pattern Mining (AKA Association Rule Mining) is an analytical process that finds frequent patterns, associations, or causal structures from data sets found in various kinds of databases such as relational databases, transactional databases, and other data repositories. Given a set of transactions, this process aims to find the rules that enable us to predict the occurrence of a specific item based on the occurrence of other items in the transaction.

Let's look at an example of Frequent Pattern Mining. First, we will want to understand the terminology used in this type of analysis. While there are numerous metrics and factors used in this technique, for this example, we will only consider two factors namely, Support and Confidence.

Support: The support of a rule x -> y (where x and y are each items/events etc.) is defined as the proportion of transactions in the data set which contain the item set x as

well as y. So, Support (x -> y)= no. of transactions which contain the item set x & y / total no. of transactions.

Confidence: The confidence of a rule x -> y is defined as: Support (x -> y) / support (x). So, it is the ratio of the number of transactions that include all items in the consequent (y in this case), as well as the antecedent (x in this case) to the number of transactions that include all items in the antecedent (x in this case).

In the table below, Support (milk->bread) = 0.4 means milk and bread are purchased together occur in 40% of all transactions. Confidence (milk->bread) = 0.5 means that if there are 100 transactions containing milk then there will be 50 that will also contain bread.

TID	Milk	Bread	Butter	Beer
1	1	0	1	1
2	1	1	1	0
3	0	1	1	0
4	1	0	0	1
5	1	1	1	1

WEB MINING

Multiple Choice Type Questions

1. The mining activity which mines web log records to discover user access patterns of web pages is [WBUT 2011, 2014]

a) web content mining
b) web usage mining
c) web structure mining
d) web search mining

Answer: (b)

2. K-means is based on [WBUT 2011, 2014, 2015]

a) Euclidian distance
b) Hamming distance
c) RMS
d) None of these

Answer: (a)

3. _____ is the application of data mining techniques to discover patterns from the Web. [MODEL QUESTION]

a) Text Mining
b) Multimedia Mining
c) Web Mining
d) Link Mining

Answer: (c)

Short Answer Type Questions

1. What do you understand by Web Mining? Compare Web Mining with Data Mining. [WBUT 2009]

Answer:

Web Mining is the use of the data mining techniques to automatically discover and extract information from web documents/services. It is used for discovering useful information from the World-Wide Web and its usage patterns. In web mining we use data mining techniques to make the web more useful and more profitable (for some) and to increase the efficiency of our interaction with the web.

Web mining can involve all of the traditional data mining processes of classification, segmentation, clustering, association, prediction, and modeling the only difference is that the analyzes result in immediate action. Unlike data mining, web mining is dependent on the use of software agent to trigger targeted offers as events take place in real time. Web mining unlike data mining which existed prior the Internet explosion involves a new paradigm of data collection, integration and analysis. Web mining involves pattern recognition via a seamless stream of activity taking place over a decision network and not a static warehouse. Web mining works by performing data analysis via networks, using software agents to mine, collaborate and discover conditions and features which can lead to increases in sales, cross-selling opportunities and the targeting of specific products or services.

2. What are the differences between OLAP & OLTP? [WBUT 2009, 2011, 2013, 2017]

Answer:

The following table summarizes the major differences between OLTP and OLAP system design.

	OLTP System Online Transaction Processing (Operational System)	**OLAP System Online Analytical Processing (Data Warehouse)**
Source of data	Operational data; OLTPs are the original source of the data.	Consolidation data; OLAP data comes from the various OLTP Databases
Purpose of data	To control and run fundamental business tasks	To help with planning, problem solving, and decision support
What the data	Reveals a snapshot of ongoing business processes	Multi-dimensional views of various kinds of business activities
Inserts and Updates	Short and fast inserts and updates initiated by end users	Periodic long-running batch jobs refresh the data
Queries	Relatively standardized and simple queries Returning relatively few records	Often complex queries involving aggregations
Processing Speed	Typically very fast	Depends on the amount of data involved; batch data refreshes and complex queries may take many hours; query speed can be improved by creating indexes
Space Requirements	Can be relatively small if historical data is archived	Larger due to the existence of aggregation structures and history data; requires more indexes than OLTP
Database Design	Highly normalized with many tables	Typically de-normalized with fewer tables; use of star and/or snowflake schemas
Backup and Recovery	Backup religiously; operational data is critical to run the business, data loss is likely to entail significant monetary loss and legal liability	Instead of regular backups, some environments may consider simply reloading the OLTP data as a recovery method

3. What is Web Structure Mining? [MODEL QUESTION]

Answer:

Web structure mining is the application of discovering structure information from the web. The structure of the web graph consists of web pages as nodes, and hyperlinks as edges connecting related pages. Structure mining basically shows the structured summary of a particular website. It identifies relationship between web pages linked by information or direct link connection. To determine the connection between two commercial websites, Web structure mining can be very useful.

Long Answer Type Questions

1. Discuss about mining multimedia data on the web. [MODEL QUESTION]

Answer:

Multimedia data mining refers to the analysis of large amounts of multimedia information in order to find patterns or statistical relationships. Once data is collected, computer programs are used to analyze it and look for meaningful connections. This information is often used by governments to improve social systems. It can also be used in marketing to discover consumer habits.

Multimedia data mining requires the collection of huge amounts of data. The sample size is important when analyzing data because predicted trends and patterns are more likely to be inaccurate with a smaller sample. This data can be collected from a number of different media, including videos, sound files, and images. Some experts also consider spatial data and text to be multimedia. Information from one or more of these media is the focus of data collection.

Whereas an analysis of numerical data can be straightforward, multimedia data analysis requires sophisticated computer programs which can turn it into useful numerical data. There are a number of computer programs available that make sense of the information gathered from multimedia data mining. These computer programs are used to search for relationships that may not be apparent or logically obvious.

When multimedia is mined for information, one of the most common uses for this information is to anticipate behavior patterns or trends. Information can be divided into classes as well, which allows different groups, such as men and women or Sundays and Mondays, to be analyzed separately. Data can be clustered, or grouped by logical relationship, which can help track consumer affinity for a certain brand over another, for example.

Multimedia data mining has a number of uses in today's society. An example of this would be the use of traffic camera footage to analyze traffic flow. This information can be used when planning new streets, expanding existing streets, or diverting traffic. Government organizations and city planners can use the information to help traffic flow more smoothly and quickly.

While the term data mining is relatively new, the practice of mining data has been around for a long time. Grocery stores, for example, have long used data mining to track consumer behavior by collecting data from their registers. The numerical data relating to sales information can be used by a computer program to learn what people are buying

and when they are likely to buy certain products. This information is often used to determine where to place certain products and when to put certain products on sale.

2. Write short notes on the following:
a) Web-Enabled Data Warehouse **[WBUT 2011]**
b) Arbor Essbase Web **[WBUT 2012, 2015, 2017]**
c) Automatic Classification of Web Documents **[MODEL QUESTION]**
d) Web Usage Mining **[MODEL QUESTION]**

Answer:

a) Web-Enabled Data Warehouse:

Today's data warehouses are no longer confined to a select group of internal users. Under present conditions, corporations need to increase the productivity of all the members in the corporation's value chain. Useful information from the corporate data warehouse must be provided not only to the employees but also to customers, suppliers, and all other business partners. This new delivery method through the web helps the partners to retrieve, analyze, and share information from the company's data warehouse. When one web-enables the data warehouse, from the point of view of the users, the key requirements are: self-service data access, interactive analysis, high availability and performance, zero-administration client (thin client technology such as Java applets), tight security, and unified metadata. Bringing the Web to the warehouse essentially involves capturing the clickstream of all the visitors to your company's Web site and performing all the traditional data warehousing functions.

The company's effort now involves extraction, transformation, and loading of the clickstream data to the Webhouse repository. Clickstream data tracks how people proceeded through the company's Web site, what triggers purchases, what attracts people, and what makes them come back.

b) Arbor Essbase Web:

The Arbor Essbase Web Gateway is used to develop and deploy intranet and internet based Web-enabled applications such as ad-hoc analysis, management reporting, enterprise information systems, budgeting, sales forecasting. Essbase allows corporations to deliver OLAP applications directly from operational systems, or within an overall data warehousing architecture. Standard Web browsers can be used to access OLAP applications developed using the Essbase Web Gateway.

Based upon a true, client/server architecture, Essbase supports multi-user read and write access, large-scale data capacity, robust analytical calculations, flexible data navigation and consistent, rapid response times. The Essbase server's open architecture supports direct data access using standard spreadsheets or leading third-party query, reporting and EIS tool.

c) Automatic Classification of Web Documents:

In the automatic classification of Web documents, each document is assigned a class label from a set of predefined topic categories, based on a set of examples of pre-classified documents. For example, Yahoo's taxonomy and its associated documents can

be used as training and test sets in order to derive a Web document classification scheme. This scheme may then be used to classify new Web documents by assigning categories from the same taxonomy. Keyword-based document classification methods were the methods can be used for Web document classification. Such a term-based classification scheme has shown good results in Web page classification. Since hyperlinks contain high quality semantic clues to a page's topic, it is beneficial to make good use of such semantic information in order to achieve even better accuracy than pure keyword based classification. However, since the hyperlinks surrounding a document may be quite noisy, naïve use of terms in a document's hyperlink neighborhood can even degrade accuracy. The use of robust statistical models such as Markov random fields (MRFs), together with relaxation labeling, has been explored. Such a method has experimentally been shown to substantially improve the accuracy of Web document classification.

d) Web Usage Mining:

Besides mining Web contents and Web linkage mines Web log records to discover user access patterns of Web pages. Analyzing and exploring regularities in Web log records can identify potential customers for electronic commerce, enhance the quality and delivery of Internet information services to the end user, and improve Web server system performance. A Web server usually registers a (Web) log entry, or Weblog entry, for every access of a Web page. It includes the URL requested, the IP address from which the request originated, and a timestamp. For Web-based e-commerce servers, a huge number of Web access log records are being collected. Popular Web sites may register the Weblog records in the order of hundreds of megabytes every day. Weblog databases provide rich information about Web dynamics. Thus it is important to develop sophisticated Weblog mining techniques.

In developing techniques for Web usage mining, we may consider the following. First, although it is encouraging and exciting to imagine the various potential applications of Weblog file analysis, it is important to know that the success of such applications depends on what and how much valid and reliable knowledge can be discovered from the large raw log data. Often, raw Weblog data need to be cleaned, condensed, and transformed in order to retrieve and analyze significant and useful information. Second, with the available URL, time, IP address, and Web page content information, a multidimensional view can be constructed on the Weblog database, and multidimensional OLAP analysis can be performed to find the top N users, top N accessed Web pages, most frequently accessed time periods, and so on, which will help discover potential customers, users, markets, and others.

Third, data mining can be performed on Weblog records to find association patterns, sequential patterns, and trends of Web accessing. For Web access pattern mining, it is often necessary to take further measures to obtain additional information of user traversal to facilitate detailed Weblog analysis. Such additional information may include user-browsing sequences of the Web pages in the Web server buffer, and so on. With the use of such Weblog files, studies have been conducted on analyzing system performance, improving system design by Web caching.

Web page pre-fetching, and Web pages swapping; understanding the nature of Web traffic and understanding user reaction and motivation.

For example, some studies have proposed adaptive sites; Web sites that improve themselves by learning from user access patterns. Weblog analysis may also help build customized Web services for individual users. Since Weblog data provide information about what kind of users will access what kind of Web pages, Weblog information can be integrated can be integrated with Web content and Web linkage structure mining to help Web page ranking, Web document classification, and the construction of a multilayered Web information base as well.

RECENT TRENDS IN DISTRIBUTED WAREHOUSING

Multiple Choice Type Questions

1. A heterogeneous distributed database is which of the following? [MODEL QUESTION]

a) The same DBMS is used at each location and data are not distributed across all nodes
b) The same DBMS is used at each location and data are distributed across all nodes
c) The different DBMS is used at each location and data are not distributed across all nodes
d) The different DBMS is used at each location and data are distributed across all nodes

Answer: (d)

2. An autonomous homogeneous environment is which of the following? [MODEL QUESTION]

a) The same DBMS is at each node and each DBMS works independently
b) The same DBMS is at each node and a central DBMS coordinates database access
c) The different DBMS is at each node and each DBMS works independently
d) The different DBMS is at each node and a central DBMS coordinates database access

Answer: (a)

Short Answer Type Questions

1. Explain class imbalance problem. [MODEL QUESTION]

Answer:

It is the problem in machine learning where the total number of a class of data (positive) is far less than the total number of another class of data (negative). This problem is extremely common in practice and can be observed in various disciplines including fraud detection, anomaly detection, medical diagnosis, oil spillage detection, facial recognition, etc.

2. What does Social Network Analysis (SNA) mean? [MODEL QUESTION]

Answer:

Social network analysis (SNA) is a process of quantitative analysis of a social network. SNA measures and maps the flow of relationships and relationship changes between knowledge-possessing entities. Simple and complex entities include websites, computers, animals, humans, groups, organizations and nations.

The SNA structure is made up of node entities, such as humans, and ties, such as relationships. The advent of modern thought and computing facilitated a gradual evolution of the social networking concept in the form of highly complex, graph-based networks with many types of nodes and ties. These networks are the key to procedures and initiatives involving problem solving, administration and operations.

3. What is distributed data mining? **[MODEL QUESTION]**

Answer:

This type of data mining is gaining popularity as it involves analyzing the information stored in different company locations or at different organizations. Highly sophisticated algorithms are used to extract data from different locations and provide detailed insights to aid decision making.

4. What are recent trends in data mining? **[MODEL QUESTION]**

Answer:

Trends in Data Mining

Data mining concepts are still evolving and here are the latest trends that we get to see in this field –

- Application Exploration.
- Scalable and interactive data mining methods.
- Integration of data mining with database systems, data warehouse systems and web database systems.
- Standardization of data mining query language.
- Visual data mining.
- New methods for mining complex types of data.
- Biological data mining.
- Data mining and software engineering.
- Web mining.
- Distributed data mining.
- Real time data mining.
- Multi database data mining.
- Privacy protection and information security in data mining.

Long Answer Type Questions

1. Explain the concept of distributed data warehouse. List some of its advantages and disadvantages. **[MODEL QUESTION]**

Answer:

A distributed data warehouse is just like the name if implies. That is, the data are shared across multiple data repositories, for the purpose of OLAP and where each data warehouse may belong to one or more organization. It consists of many local data warehouses with one centralized global data warehouse. Distributed data warehousing covers a complete enterprise data warehouse but have tiny data stores that are built separately. These data stores are connected physically over a network to provide users

access to the relevant reports without affecting performance. Moreover, a distributed data warehouse is said to be the core of all enterprise data which is used to send relevant data to individual data marts. Bu doing so, users can easily access information required for order management, customer billing, sales analysis and other reporting functions. Distributed data warehouses can be categorized into three types, which are as follows:

- **Local and global data warehouse:** In this type, there is a local data warehouse which represents the data unique to the local operating site and, global data warehouse which represents that part of data which is integrated across the business.
- **Technologically distributed data warehouse:** In this type, logically there is a single data warehouse but physically there are many data warehouses which are all related and distributed over multiple processors.
- **Independently evolving distributed data warehouse:** In this type, data warehouse environment builds in an uncoordinated environment. That is, first one data warehouse appears, then second and so on. Therefore, it results in political and organizational differences due to the lack of coordination among different data warehouses.

The advantages of using distributed data warehouse are as follows:

- It is faster to achieve as each local site can control over its design and resources.
- There is no limit for placing the data into each local or global data warehouse. However, additional processors can be added if the volume of data exceeds the limit of distributed processors.
- The entry cost is much less than with centralized structure. The requirement of hardware and software is much less when loaded initially on distributed technology.

The distributed data warehouse has some disadvantages also. These are as follows:

- Issues like metadata, data transfer makes the environment complex and results in more overhead
- Managing multiple development efforts on local sites is an unmanageable task for data warehouse architect.
- In a distributed environment, the roles and responsibilities are not clearly defined.
- Major technological problems could arise by transfer of data and multiple table queries.
- When the warehouse is distributed over multiple servers, excessive network traffic starts to flow from source to destination.
- Coordinating development across the distributed locations becomes complex and less effective.
- Interconnectivity between the different local sites of distributed data warehouse could be problematic in case of traffic congestion.

2. Explain the concept of distributed data mining. **[MODEL QUESTION]**

Answer:

Traditional warehouse-based architectures for data mining suppose to have centralized data repository. Such a centralized approach is fundamentally inappropriate for most of the distributed and ubiquitous data mining applications. In fact, the long response time, lack of proper use of distributed resource, and the fundamental characteristic of centralized data mining algorithms do not work well in distributed environments. A scalable solution for distributed applications calls for distributed processing of data, controlled by the available resources and human factors. For example, let us suppose an ad hoc wireless sensor network where the different sensor nodes are monitoring some time-critical events. Central collection of data from every sensor node may create traffic over the limited bandwidth wireless channels and this may also drain a log of power from the devices. A distributed architecture for data mining is likely aimed to reduce the communication load and also to reduce the battery power more evenly across the different nodes in the sensor network.

One can easily imagine similar needs for distributed computation of data mining primitives in ad hoc wireless networks of mobile devices like PDAs, cellphones and wearable computers. The wireless domain is not the only example. In fact, most of the applications that deal with time-critical distributed data are likely to benefit by paying careful attention to the distributed resources for computation, storage, and the cost of communication. As an other example, let us consider the World Wide Web: it contains distributed data and computing resources. An increasing number of databases (e.g., weather databases, oceanographic data, etc.) and data streams (e.g., financial data, emerging disease information, etc.) are currently made on-line, and many of them change frequently. It is easy to think of many applications that require regular monitoring of these diverse and distributed sources of data. A distributed approach to analyze this data is likely to be more scalable and practical particularly when the application involves a large number of data sites. Hence, in this case we need data mining architectures that pay careful attention to the distribution of data, computing and communication, in order to access and use them in a near optimal fashion. Distributed Data Mining (sometimes referred by the acronym DDM) consider data mining in this broader context.

DDM may also be useful in environments with multiple compute nodes connect over high-speed networks. Even if the data can be quickly centralized using the relatively fast network, proper balancing of computational load among a cluster of nodes may require a distributed approach. The privacy issue is playing an increasingly important role in the emerging data mining applications. For example, let us suppose a consortium of different banks collaborating for detecting frauds. If a centralized solution was adopted, all the data from every bank should be collected in a single location, to be processed by a data mining system. Nevertheless, in such a case a Distributed Data Mining system should be the natural technological choice: both it is able to learn models from distributed data without exchanging the raw data between different repository, and it allows detection of fraud by preserving the privacy of every bank's customer transaction data.

For what concerns techniques and architecture, it is worth noticing that many several other fields influence Distributed Data Mining systems concepts. First, many DDM systems adopt the Multi-Agent System (MAS) architecture, which finds its root in the Distributed Artificial Intelligence (DAI). Second, although Parallel Data Mining often assumes the presence of high-speed network connections among the computing nodes, the development of DDM has also been influenced by the PDM literature. Most DDM algorithms are designed upon the potential parallelism they can apply over the given distributed data. Typically, the same algorithm operates on each distributed data site concurrently, producing one local model per site. Subsequently, all local models are aggregated to produced the final model. In following figure a general Distributed Data Mining framework is presented. In essence, the success of DDM algorithms lies in the aggregation. Each local model represents locally coherent patterns, but lacks details that may be required to induce globally meaningful knowledge. For this reason, many DDM algorithms require a centralization of a subset of local data to compensate it. The ensemble approach has been applied in various domains to increase the accuracy of the predictive model to be learnt. It produces multiple models and combines them to enhance accuracy. Typically, voting (weighted or un-weighted) schema are employed to aggregate base model for obtaining a global model. As we have discussed above, minimum data transfer is another key attribute of the successful DDM algorithm.

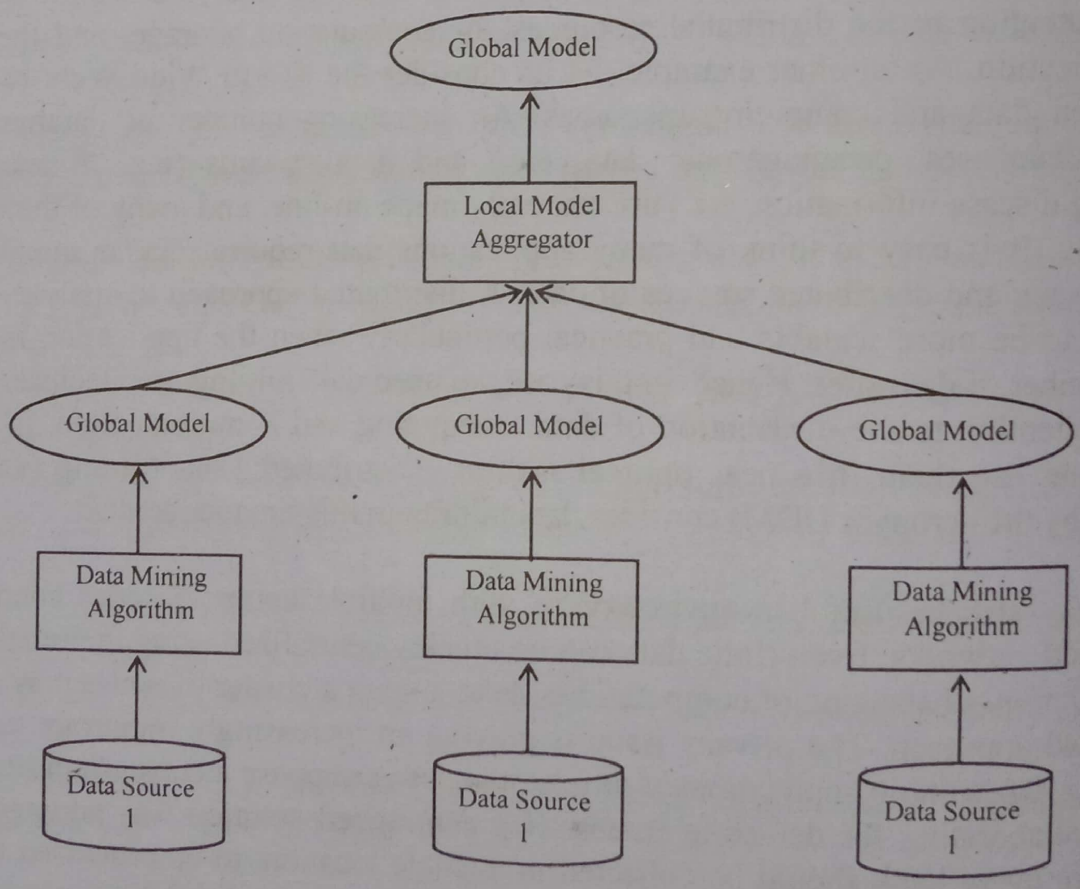


Fig: General Distributed Data Mining Framework

3. Why Class Imbalance is a problem? **[MODEL QUESTION]**

Answer:

Most machine learning algorithms and works best when the number of instances of each classes are roughly equal. When the number of instances of one class far exceeds the other, problems arise. This is best illustrated below with an example.

Given a dataset of transaction data, we would like to find out which are fraudulent and which are genuine ones. Now, it is highly cost to the e-commerce company if a fraudulent transaction goes through as this impacts our customers trust in us, and costs us money. So we want to catch as many fraudulent transactions as possible.

If there is a dataset consisting of 10000 genuine and 10 fraudulent transactions, the classifier will tend to classify fraudulent transactions as genuine transactions. The reason can be easily explained by the numbers. Suppose the machine learning algorithm has two possibly outputs as follows:

1. Model 1 classified 7 out of 10 fraudulent transactions as genuine transactions and 10 out of 10000 genuine transactions as fraudulent transactions.
2. Model 2 classified 2 out of 10 fraudulent transactions as genuine transactions and 100 out of 10000 genuine transactions as fraudulent transactions.

If the classifier's performance is determined by the number of mistakes, then clearly Model 1 is better as it makes only a total of 17 mistakes while Model 2 made 102 mistakes. However, as we want to minimize the number of fraudulent transactions happening, we should pick Model 2 instead which only made 2 mistakes classifying the fraudulent transactions. Of course, this could come at the expense of more genuine transactions being classified as fraudulent transactions, but will be a cost we can bear for now. Anyhow, a general machine learning algorithm will just pick Model 1 than Model 2, which is a problem. In practice, this means we will let a lot of fraudulent transactions go through although we could have stopped them by using Model 2. This translates to unhappy customers and money lost for the company.

4. What do you understand by the term 'Graph Mining'? **[MODEL QUESTION]**

Answer:

Graphs become increasingly important in modeling complicated structures, such as circuits, images, chemical compounds, protein structures, biological networks, social networks, the Web, workflows, and XML documents. Many graph search algorithms have been developed in chemical informatics, computer vision, video indexing, and text retrieval. With the increasing demand on the analysis of large amounts of structured data, graph mining has become an active and important theme in data mining. Among the various kinds of graph patterns, frequent substructures are the very basic patterns that can be discovered in a collection of graphs. They are useful for characterizing graph sets, discriminating different groups of graphs, classifying and clustering graphs, building graph indices, and facilitating similarity search in graph databases. Recent studies have developed several graph mining methods and applied them to the discovery of interesting patterns in various applications.

For example, there have been reports on the discovery of active chemical structures in HIV-screening datasets by contrasting the support of frequent graphs between different classes.
There have been studies on the use of frequent structures as features to classify chemical compounds, on the frequent graph mining technique to study protein structural families, on the detection of considerably large frequent subpathways in metabolic networks, and on the use of frequent graph patterns for graph indexing and similarity search in graph databases.

5. What are recent developments in distributed data warehouses environments? [MODEL QUESTION]

Answer:
Currently data warehouse is used as organizational repository to support business decision making. Mostly the data warehouse systems use centralized approach. Furthermore, the hierarchy of organization and classes of users is not considered in data warehousing systems. Before the iPhone and Xbox, prior to the first Tweet or Facebook "Like," and well in advance of tablets and the cloud, there was the data warehouse. For 30 years, businesses have centrally stored data for analysis and data driven decision making. For all of that time, the data warehouse has been the business-insights workhorse of enterprise computing. The big trend in the mid 1990's was the emergence of data warehouses that were a terabyte in size, which at the time was considered a huge amount of data. Today's leading edge systems are a thousand times larger—measured in peta bytes. Data warehouses have had staying power because the concept of a central data repository—fed by dozens or hundreds of databases, applications, and other source systems—continues to be the best, most efficient way for companies to get an enterprise-wide view of their customers, supply chains, sales, and operations. ASMs support refinement method in developing a data warehouse and OLAP systems. of One strategy Abstract State Machines (ASMs) also be used to design a distributed data warehouses also Abstract State Machines provide a meticulous mathematically tricks for high-level system design, validation and verification at earliest stage of system development. Wehrle et al (2007) also deal with a distributed, grid-aware environment. They apply the Globus Toolkit together with a set of specialized services for grid based data warehouses. Fact table data is partitioned and distributed across participant nodes. Grid computing has emerged as a new technology, whose main challenge is the complete integration of heterogeneous computing systems and data resources with the aim of providing a global computing space. A new Data Mining Grid Architecture, named DMGA, which for their composition in a real scenario. Dimension tables data is replicated. A local data index service provides local information about data stored at each node. A communication service uses the local data index service from the participant grid's nodes to enable that remote data is accessed. The first step in query execution is to search for data at the local node (using the local index service). Missing data is located by the use of the communication service and accessed remotely. Bindia et. al. (2012) A multi agent based system based query cycling process in distributed DWH and also remove the disadvantage of multi agent approach by placing an buffer, so that there will be no need

for client to connect at all the time to get the results. Distributed systems provides support where optimization methods to data design and OLAP queries in the data warehouse environment should be implemented with an objective of supporting the decision makers by providing a single view of data even though that data is physically distributed across multiple data warehouses in multiple systems at different branches . Another work on grid-aware data warehouses is presented in (Lawrence Rau-Chaplin 2006).The OLAP-Enabled Grid considers the scenario where the data of a single organization is distributed across a number of operational databases at remote locations. Each operational database has capabilities for answering OLAP queries, and access to a possible variety of other computational and storage resources which are located close by. Users who are interested in doing OLAP on these databases are distributed over the network.

Below are some new trends and opportunities in data warehousing.

- The "datafication" of the enterprise requires more capable data warehouses.
- Physical and logical consolidation help reduce costs.
- Hadoop optimizes data warehouse environments.
- Customer experience (CX) strategies use realtime analytics to improve marketing campaigns.
- Engineered systems are becoming a preferred approach for large scale information management.
- On-demand analytics environments meet the growing demand for rapid prototyping and information discovery
- Data compression enables higher-volume, higher value analytics
- In-database analytics simplify analysis

6. What are recent trends in data mining? **[MODEL QUESTION]**

Answer:

Businesses which have been slow in adopting the process of data mining are now catching up with the others. Extracting important information through the process of data mining is widely used to make critical business decisions. In the coming decade, we can expect data mining to become as ubiquitous as some of the more prevalent technologies used today.

Some of the key data mining trends for the future include –

1. Multimedia Data Mining:

This is one of the latest methods which is catching up because of the growing ability to capture useful data accurately. It involves the extraction of data from different kinds of multimedia sources such as audio, text, hypertext, video, images, etc. and the data is converted into a numerical representation in different formats. This method can be used in clustering and classifications, performing similarity checks, and also to identify associations.

2. Ubiquitous Data Mining:
This method involves the mining of data from mobile devices to get information about individuals. In spite of having several challenges in this type such as complexity, privacy, cost, etc. this method has a lot of opportunities to be enormous in various industries especially in studying human-computer interactions.

3. Distributed Data Mining:
This type of data mining is gaining popularity as it involves the mining of huge amount of information stored in different company locations or at different organizations. Highly sophisticated algorithms are used to extract data from different locations and provide proper insights and reports based upon them.

4. Spatial and Geographic Data Mining:
This is new trending type of data mining which includes extracting information from environmental, astronomical, and geographical data which also includes images taken from outer space. This type of data mining can reveal various aspects such as distance and topology which is mainly used in geographic information systems and other navigation applications.

5. Time Series and Sequence Data Mining:
The primary application of this type of data mining is study of cyclical and seasonal trends. This practice is also helpful in analyzing even random events which occur outside the normal series of events. This method is mainly being use by retail companies to access customer's buying patterns and their behaviors.