Q.1 Differentiate between data mining and data warehouse.

Data Mining and Data Warehousing both are used to holds business intelligence and enable decision making. But both, data mining and data warehousing have different aspects of operating on an enterprise's data. On the one hand, the **data warehouse** is an environment where the data of an enterprise is gathering and stored in a aggregated and summarized manner. On the other hands, **data mining** is a process; that apply algorithms to extract knowledge from the data that you even don't know exist in the database.

Data Warehouse is a central location where information **gathered from multiple sources are stored under a single unified schema**. The data is initially gathered, different sources of enterprise then cleaned and transformed and stored in a data warehouse. Once data is entered in a data warehouse, it stays there for a long time and can be accessed overtimes.

Data Warehouse is a perfect blend of technologies like **data modelling, data acquisition, data management, metadata management, development tools store managements**. All these technologies support functions like **data extraction, data transformation, data storage, providing user interfaces for accessing the data**.

Key Differences between Data Mining and Data Warehousing

There is a basic difference that separates data mining and data warehousing that is data mining is a process of extracting meaningful data from the large database or data warehouse. However, data warehouse provides an environment where the data is stored in an integrated form which eases data mining to extract data more efficiently.

Q.2 Explain the DBMS and Data warehouse.

1. A database stores current data while a data warehouse stores historical data.
2. A database often changes due to frequent updates done on it, and hence, it cannot be used for analysis or reaching decision. A data warehouse extracts data and reports them to analyze and reach decisions.

3. A general database is used for Online Transactional Processing while a data warehouse is used for Online Analytical Processing.
4. Tables in a database are normalized to achieve efficient storage while a data warehouse is usually demoralized to achieve faster querying.
5. Analytical queries are much faster on a data warehouse than on a database.
6. A database contains highly detailed data while a data warehouse contains summarized data.
7. A database provides a detailed relational view while a data warehouse provides a summarized multidimensional view.
8. A database can do a lot of concurrent transactions while a data warehouse is not designed for such tasks.

Q.3 Explain the K-mean and K-Mediod Algorithm with example.

k-means is one of the simplest unsupervised learning algorithms that solve the well-known 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 apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest center. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center. A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more. Finally, this algorithm aims at minimizing an objective function knows as squared error function given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} \left(\left\| x_i - v_j \right\| \right)^2$$

where,

'$||x_i - v_j||$' is the Euclidean distance between x_i and v_j.

'c_i' is the number of data points in i^{th} cluster.

'c' is the number of cluster centers.

Algorithmic steps for k-means clustering

Let $X = \{x_1, x_2, x_3, \ldots\ldots, x_n\}$ be the set of data points and $V = \{v_1, v_2, \ldots\ldots, v_c\}$ be the set of centers.

1) Randomly select 'c' cluster centers.

2) Calculate the distance between each data point and cluster centers.

3) Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers..

4) Recalculate the new cluster center using:

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$$

where, *'c_i'* represents the number of data points in i^{th} cluster.

5) Recalculate the distance between each data point and new obtained cluster centers.

6) If no data point was reassigned then stop, otherwise repeat from step 3).

The ***k*-medoids algorithm** is a clustering algorithm related to the *k*-means algorithm and the medoidshift algorithm. Both the *k*-means and *k*-medoids algorithms are partitional (breaking the dataset up into groups). *K*-means attempts to minimize the total squared error, while *k*-medoids minimizes the sum of dissimilarities between points labeled to be in a cluster and a point designated as the center of that cluster. In contrast to the *k*-means algorithm, *k*-medoids chooses data points as centers (medoids or exemplars).

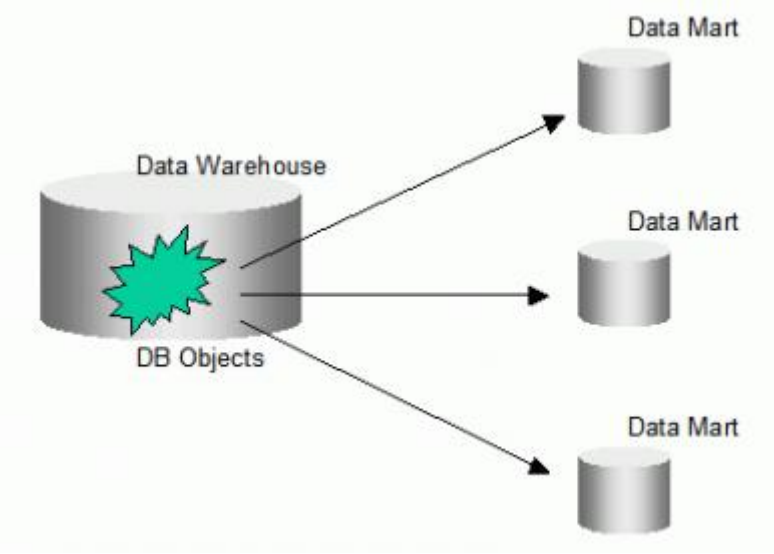
K-medoids is also a partitioning technique of clustering that clusters the data set of *n* objects into *k* clusters with *k* known *a priori*. A useful tool for determining *k* is the silhouette. It could be more robust to noise and outliers as compared to *k*-means because it minimizes a sum of general pairwise dissimilarities instead of a sum of squared Euclidean distances. The possible choice of the dissimilarity function is very rich but in our applet we used the Euclidean distance. A medoid of a finite dataset is a data point from this set, whose average dissimilarity to all the data points is minimal i.e. it is the most centrally located point in the set. The most common realization of *k*-medoid clustering is the **Partitioning Around Medoids (PAM)** algorithm and is as follows:

1. Initialize: randomly select *k* of the *n* data points as the medoid
2. **Assignment step**: Associate each data point to the closest medoid.
3. **Update step**: For each medoid *m* and each data point *o* associated to *m* swap *m* and *o* and compute the total cost of the configuration (that is, the average dissimilarity of *o* to all the data points associated to *m*). Select the medoid *o* with the lowest cost of the configuration.

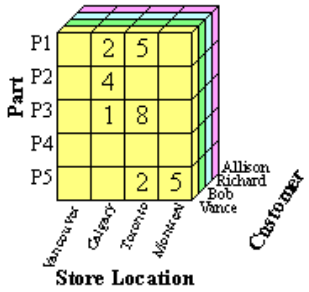
Repeat alternating steps 2 and 3 until there is no change in the assignments.

Q.4 Differentiate between Data marks and Meta Data.

A data mart is a *concept*, whereas a cube is an *implementation option*. A data mart is a problem-specific data store, designed to hold information for reporting / analysis / insight around a specific organizational function (typically), and contains a subset of data from a data warehouse. A Data Mart is the staging area for data that serves the needs of a particular segment or business unit. It is a subset of the data in the data warehouse that focuses the information to a particular subject or operational department, fitted to the purpose of the users without redundancy.



The 'cube' metaphor refers to a non-relational data store which represents many dimensions of related data. Cube is normally a shortening of 'hypercube' - a reference to a multidimensional concept from geometry (and wider mathematics) - and highlights that a 'cube' can have more than 3 dimensions.



Data marts are often implemented using cubes, although some data marts are implemented using relational databases (using star and snowflake schemas). More recently, data marts are being implemented in columnar databases and on non-relational / non-cube technologies such as Hadoop.

Q.5 What do you mean by Virtual data ware house?

A virtual warehouse is another term for a data warehouse. A data warehouse is a computing tool designed to simplify decision-making in business management. It collects and displays business data relating to a specific moment in time, creating a snapshot of the condition of the business at that moment. Virtual warehouses often collect data from a wide variety of sources.

Database

A virtual warehouse is essentially a business database. The data found in a virtual warehouse is usually copied from multiple sources throughout a production system. This is done so related data can be searched quickly and without accessing the entire system. Performing a search of an entire production system at one time could potentially compromise the system's performance. Using a data warehouse removes this operating risk and speeds up the overall access process.

Data Mart

Depending on the type of information being stored in a virtual warehouse, a single warehouse can become overburdened with data from dozens of different sources relating to any number of potential topics. To prevent the warehouse from becoming impossible to navigate, subdivisions called data marts are sometimes used. These data marts divide the information saved in the warehouse into categories that can be individually selected and searched based on what the user is seeking.

Features

Data warehouses are generally updated regularly to ensure they reflect the current condition of the business whose data they store. They can often be set to update on a daily, weekly or customized timetable, depending on the desires of the user. The data stored in a virtual warehouse is static. This means new data is stored alongside existing data rather than over it, allowing you to access historical information as well as current information.

Integration

Another advantage to using a virtual warehouse to store and catalog data is integration. In computing terms, integration is when data pulled from two or more sources that label their information differently is stored using a single means of identification. Integration is useful for data retrieval purposes, because it makes it possible to search all stored data at one time, as opposed to performing individual searches of each source dictated by the source's specific identification method.

Q.6 Explain the tuning and testing of data warehouse.

A data warehouse keeps evolving and it is unpredictable what query the user is going to post in the future. Therefore it becomes more difficult to tune a data warehouse system. In this chapter, we will discuss how to tune the different aspects of a data warehouse such as performance, data load, queries, etc.

- There is little that can be done to **tune** any business rules enforced by constraints. If the rules are enforced by using SQL or by triggers code, that can needs to be tuned to maximal efficiency.
- The load can be also improved by using parallelism.
- The data warehouse will contain two types of query. There will be fixed queries that are clearly defined and well understood, such as regular reports, common aggregations, etc.
- Often the correct tuning choice for such eventualities will be to allow an infrequently used index or aggregation to exist to catch just those sorts of query.
- To create those sorts of indexes or aggregations, you must have an understanding that such queries are likely to be run.
- Before you can tune the data warehouse you must have some objective measures of performance to work with. Measures such as
  - Average query response times
  - Scan rates
  - I/O throughput rates
  - Time used per query

- Memory usage per process
- These measure should be specified in the service level agreement (SLA).

ETL or Data warehouse **testing** is categorized into four different engagements:
New Data Warehouse Testing – New DW is built and verified from scratch. Data input is taken from customer requirements and different data sources and new data warehouse is build and verified with the help of ETL tools.
Migration Testing – In this type of project customer will have an existing DW and ETL performing the job but they are looking to bag new tool in order to improve efficiency.
Change Request – In this type of project new data is added from different sources to an existing DW. Also, there might be a condition where customer needs to change their existing business rule or they might integrate the new rule.
Report Testing – Report are the end result of any Data Warehouse and the basic propose for which DW is build. Report must be tested by validating layout, data in the report and calculation.

Q.7 Differentiate between KDD and data mining.

KDD is a field of computer science, which deals with extraction of previously unknown and interesting information from raw data. KDD is the whole process of trying to make sense of data by developing appropriate methods or techniques. This process deal with the mapping of low-level data into other forms those are more compact, abstract and useful. This is achieved by creating short reports, modeling the process of generating data and developing predictive models that can predict future cases. Due to the exponential growth of data, especially in areas such as business, KDD has become a very important process to convert this large wealth of data in to business intelligence, as manual extraction of patterns has become seemingly impossible in the past few decades.

Data Mining is only a step within the overall KDD process. There are two major Data Mining goals as defined by the goal of the application, and they are namely verification or discovery. Verification is verifying the user's hypothesis about data, while discovery is automatically finding interesting patterns. There are four major data mining task: clustering, classification, regression, and association (summarization). Clustering is identifying similar groups from unstructured data. Classification is learning rules that can be applied to new data. Regression is finding functions with minimal error to model data. And association is looking for relationships between variables.

Although, the two terms KDD and Data Mining are heavily used interchangeably, they refer to two related yet slightly different concepts. KDD is the overall process of extracting knowledge from data while Data Mining is a step inside the KDD process, which deals with identifying patterns in data. In other words, Data Mining is only the application of a specific algorithm based on the overall goal of the KDD process.

Q.8 Explain the data mining query language.

The Data Mining Query Language is actually based on the Structured Query Language (SQL). Data Mining Query Languages can be designed to support ad hoc and interactive data mining. This DMQL provides

commands for specifying primitives. The DMQL can work with databases and data warehouses as well. DMQL can be used to define data mining tasks. Particularly we examine how to define data warehouses and data marts in DMQL.

Here is the syntax of DMQL for specifying task-relevant data −

```
use database database_name

or

use data warehouse data_warehouse_name
in relevance to att_or_dim_list
from relation(s)/cube(s) [where condition]
order by order_list
group by grouping_list
```

Characterization

The syntax for characterization is −

```
mine characteristics [as pattern_name]
   analyze  {measure(s) }
```

The analyze clause, specifies aggregate measures, such as count, sum, or count%.

Discrimination

The syntax for Discrimination is −

```
mine comparison [as {pattern_name]}
For {target_class } where  {t arget_condition }
{versus  {contrast_class_i }
where {contrast_condition_i}}
analyze  {measure(s) }
```

Association

The syntax for Association is−

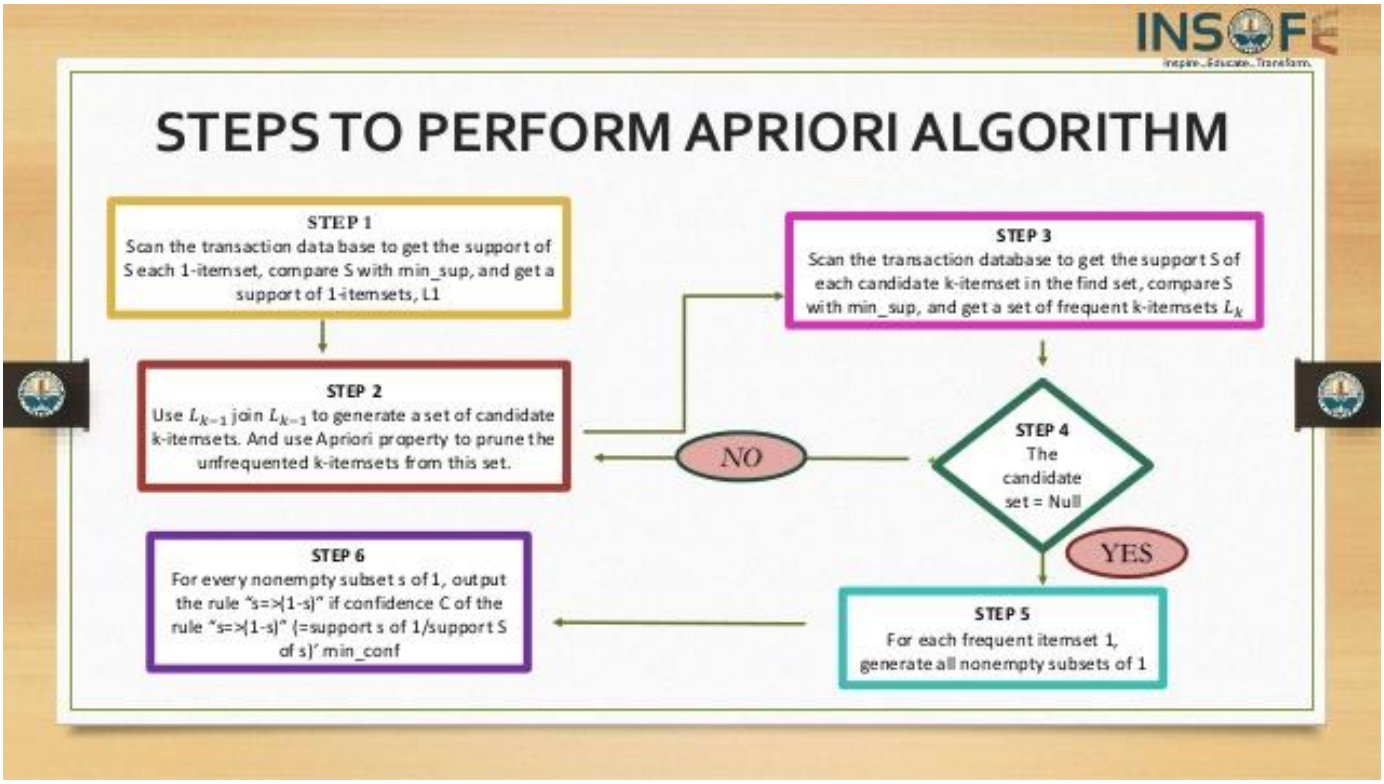
```
mine associations [ as {pattern_name} ]
{matching {metapattern} }
```

Prediction

The syntax for prediction is −

```
mine prediction [as pattern_name]
analyze prediction_attribute_or_dimension
{set {attribute_or_dimension_i= value_i}}
```

Q.9 Explain the Aprion Algorithm.



Q.10 Explain the types of regression.

Numeric prediction is the task of predicting continuous (or ordered) values for given input For example: We may wish to predict the salary of college graduates with 10 years of work experience, or the potential sales of a new product given its price. The mostly used approach for numeric prediction is regression A statistical methodology that was developed by Sir Frances Galton (1822-1911), a mathematician who was also a cousin of Charles Darwin. In many texts use the terms "regression" and "numeric prediction" synonymously Regression analysis can be used to model the relationship between one or more independent or predictor variables and a dependent or response variable (which is continuous value). In the context of data mining, the predictor variables are the attributes of interest describing the tuple. The response variable is what we want to predict.

Types of Regression

The types of Regression are as:

¬ Linear Regression

¬ Nonlinear Regression

Linear Regression

Straight-line regression analysis involves a response variable, y, and a single predictor variable, x.

It is the simplest form of regression, and models y as a linear function of x.

That is,

y=b+wx

Where the variance of y is assumed to be constant, and b and w are regression coefficients specifying the Y intercept and slope of the line, respectively.
The regression coefficient, w and b, can also be thought of as weight, so that we can equivalent write,
y=w0+w1x.
The regression coefficient can be estimated using this method with the following equations:
[Refer to write board:]
Example Too:

Multiple Linear Regression

The multiple linear regression is an extension of straight-line regression so as to involve more than one predictor variable. An example of a multiple linear regression model based on two predictor attributes or variables, A1 and A2, is
y=w0+w1x1+w2x2,
Where x1 and x2 are the values of attributes A1 and A2, respectively, in X. Multiple regression problems are instead commonly solved with the use of statistical software packages, such as SPSS(**Statistical Package for the Social Sciences)**, etc..

Nonlinear Regression

The straight-line linear regression case where dependent response variable, y, is modeled as a linear function of a single independent predictor variable, x. If we can get more accurate model using a nonlinear model, such as a parabola or some other higher-order polynomial? Polynomial regression is often of interest when there is just one predictor variable. Consider a cubic polynomial relationship given by
y=w0+w1x+w2xsq2+w3xcu3

Nonlinear Regression

In statistics, **nonlinear regression** is a form of regression analysis in which observational data are modeled by a function which is a nonlinear combination of the model parameters and depends on one or more independent variables. The data are fitted by a method of successive approximations.