16CAP504D **DATA MINING AND DATA WAREHOUSING** **4H - 4C**

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks: Internal: 40 External: 60** **Total: 100**

Course Objective:

- To introduce students to the basic concepts and techniques of Data Mining.
- To develop skills of using recent data mining software for solving practical problems.
- To gain experience of doing independent study and research.

Learning Outcomes: A student who successfully completes this course should, at a minimum, be able to:

- To introduce students to the basic concepts and techniques of Data Mining.
- To develop skills of using recent data mining software for solving practical problems.
- To gain experience of doing independent study and research.
- Possess some knowledge of the concepts and terminology associated with database systems, statistics, and machine learning

UNIT I

Introduction to Data Mining: Motivation and importance, Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Pattern Classification of Data Mining Systems, Major issues in Data Mining. Pre-process the Data-Data Cleaning, Data Integration and Transformation.

UNIT II

Classification and Regression Algorithms : Naïve Bayes – Multiple Regression Analysis – Logistic Regression – k-Nearest Neighbour Classification – GMDH –Computing and Genetic Algorithms. Support Vector Machines : Linear SVM - SVM with soft margin – Linear kernel – Proximal SVM – Generating Datasets.

Cluster Analysis : Partitional Clusterings – k-medoids – Birch – DBSCAN – Optics – Graph Partitioning – CHAMELEON – COBWEB – GCLuto.

UNIT III

Mining Association rule in large Databases Association Rule Mining, Mining Single -Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Dataware houses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT IV

Mining Complex Types of Data : Mining Spatial Databases – Multimedia Databases – Time-series and Sequence Data – Text Databases – Web Data Mining – Search Engines.

UNIT V

Data Warehouse and OLAP Technology for Data Mining. What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology, Data Ware housing to Data Mining Data Preprocessing Data Warehousing: Failures of past Decision Support System- Operational vs. DSS- Building blocks: features- Data warehouse and Data Mart- Overview of the Components- Metadata Architectural Components: Distinguishing Characteristics- Architectural Framework- Technical Architecture.

SUGGESTED READINGS

1. Jiawei Han and Micheline Kamber.(2011), Data Mining Concepts and Techniques, 3rd Edition, Elsivier,India (Unit I, III, IV, V)
2. Florin Gorunesu. (2011), Data Mining: Concepts, Models and Techniques, Springer. (Unit – IV)
3. Soman.K.P, Shyam Divakar and V. Ajay. (2008), Insight to Data Mining- Theory and Practical, Prentice Hall India, New Delhi. (Unit – II).
4. Kantardzic.(2012). Mining Concepts, Models, Methods and Algorithms, IEEE Press – A John Wiley & Sons.

WEB SITES:

1. www.wikipedia.org/wiki/Data_mining
2. www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.ht
3. www.thearling.com/text/dmwhite/dmwhite.htm

Karpagam Academy of Higher Education

(Deemed to be University)
(Established Under Section 3 of UGC Act,1956)

DEPARTMENT OF COMPUTER APPLICATIONS

Subject Name: Data Mining and Data Warehousing **Subject Code:** 16CAP504D
Class : **III** M.C.A **Semester:** V

LECTURE PLAN

Sl.No	Lecture Duration (Periods)	Topics to be covered	Support Materials
		Unit- I	
1	1	Introduction to Data Mining: Motivation and importance	**W1, T1:** 1-9
2	1	Data Mining, Relational Databases, Data Warehouses, Transactional Databases	**T1:** 10-14,**W1**
3	1	Advanced Database Systems and Advanced Database Applications	**T1:** 15-20
4	1	Data Mining Functionalities	**T1: 21-27**, W1
5	1	Pattern Classification of Data Mining Systems	**T1:** 29-30,J1, W1
6	1	Major issues in Data Mining. Pre-process the Data	**T1:** 36-38,**W1,J1**
7	1	Data Cleaning, Data Integration and Transformation.	**T1: 6**1-70,W1
8	1	Recapitulation & Important Questions Discussion	
Total No.Of Periods Planned	**8**		
Sl.No	**Lecture Duration (Periods)**	**Topics to be covered**	**Support Materials**
		Unit- II	
9	1	Classification and Regression Algorithms : Naïve Bayes – Multiple Regression Analysis	**J1**,W1
10	1	Logistic Regression – k-Nearest Neighbour Classification – GMDH	**J1**,W1
11	1	Computing and Genetic Algorithms. Support Vector Machines : Linear SVM	W1
12	1	SVM with soft margin – Linear kernel – Proximal SVM	**J1**,W1
13	1	Generating Datasets. Cluster Analysis : Partitional Clusterings – k-medoids	**J1**,W1
14	1	Birch – DBSCAN – Optics – Graph Partitioning	**J1**,W1

15	1	CHAMELEON – COBWEB – GCLuto	**J1**,W1
16	1	Recapitulation & Important Questions Discussion	
Total No.Of Periods Planned	**8**		
Sl.No	**Lecture Duration (Periods)**	**Topics to be covered**	**Support Materials**
Unit- III			
17	1	Mining Association rule in large Databases	**T1:** 227-234,W1
18	1	Association Rule Mining	**W1**
19	1	Mining Single	**T1:235-249,W1**
20	1	Dimensional Boolean Association Rules from Transactional Databases	**W1**
21	1	Mining Multilevel Association Rules from Transaction Databases	**T1:250-253,W1**
22	1	Mining Multidimensional Association Rules from Relational Databases	**T1:254-258,W1**
23	1	Data ware houses	**W1**
24	1	From Association Mining to Correlation Analysis	**T1:259-261, W1**
25	1	Constraint-Based Association Mining.	**T1:265-267,W1,J1**
26	1	Recapitulation & Important Questions Discussion	-
Total No.Of Periods Planned	**10**		
Sl.No	**Lecture Duration (Periods)**	**Topics to be covered**	**Support Materials**
Unit- IV			
27	1	Mining Complex Types of Data	**T1: 591-598,W1,J1**
28	1	Mining Spatial Databases	**T1: 600-607,W1,J1**
29	1	Multimedia Databases	**T1:607-613,W1,J1**
30	1	Multimedia Databases Cont..	**T1:607-613,W1,J1**
31	1	Sequence Data	**T1:489-493,**
32	1	Text Databases	**T1: 614-624,W1**
33	1	Time-series	**T1:498-512,W2**
34	1	Web Data Mining	**T1:628-640,W1,J1**
35	1	Search Engines	**T1:628-640**
36	1	Recapitulation & Important Questions Discussion	**-**
Total No.Of Periods Planned	**10**		
Sl.No	**Lecture Duration (Periods)**	**Topics to be covered**	**Support Materials**

Unit- V			
37	1	Data Warehouse and OLAP Technology for Data Mining.	T1:105-109,W1
38		What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture	
39	1	Data Warehouse Implementation, Development of Data Cube Technology	T1: 110-144
40	1	Data Ware housing to Data Mining, Data Preprocessing Data Warehousing	T1:189-192
41	1	Failures of past Decision Support System- Operational vs. DSS	W1,J1
42	1	Building blocks: features- Data warehouse and Data Mart- Overview of the Components	W1,J1
43		Metadata Architectural Components:	W1,J1
44		Distinguishing Characteristics- Architectural Framework- Technical Architecture.	W1,J1
45	1	Recapitulation & Important Questions Discussion	-
46	1	Discussion of previous ESE Question papers	
47	1	Discussion of previous ESE Question papers	
48	1	Discussion of previous ESE Question papers	
Total No.Of Periods Planned	**12**		
Overall Total (All Units)	**48**		

Support Materials:

Text Book:

1. **T1→** Jiawei Han and Micheline Kamber.(2011), Data Mining Concepts and Techniques, 3rd Edition, Elsivier,India

2. **T2→** Soman.K.P, Shyam Divakar and V. Ajay. (2008), Insight to Data Mining- Theory and Practical, Prentice Hall India, New Delhi.

Websites:

1. **W1→**http:// WWW.TUTORIALPOINT.COM/DATA_MINING

Journal

1. **J1→** International Journal of Data warehousing and Data Mining

UNIT I

Introduction to Data Mining: Motivation and importance, Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Pattern Classification of Data Mining Systems, Major issues in Data Mining. Pre-process the Data-Data Cleaning, Data Integration and Transformation.

Introduction to Data Mining: Motivation and importance

In this information age, because we believe that information leads to power and success, and thanks to sophisticated technologies such as computers, satellites, etc., we have been collecting tremendous amounts of information. Initially, with the advent of computers and means for mass digital storage, we started collecting and storing all sorts of data, counting on the power of computers to help sort through this amalgam of information. Unfortunately, these massive collections of data stored on disparate structures very rapidly became overwhelming. This initial chaos has led to the creation of structured databases and database management systems (DBMS). The efficient database management systems have been very important assets for management of a large corpus of data and especially for effective and efficient retrieval of particular information from a large collection whenever needed. The proliferation of database management systems has also contributed to recent massive gathering of all sorts of information. Today, we have far more information than we can handle: from business transactions and scientific data, to satellite pictures, text reports and military intelligence. Information retrieval is simply not enough anymore for decision-making. Confronted with huge collections of data, we have now created new needs to help us make better managerial choices. These needs are automatic summarization of data, extraction of the "essence" of information stored, and the discovery of patterns in raw data.

What kind of information are we collecting?

We have been collecting a myriad of data, from simple numerical measurements and text documents, to more complex information such as spatial data, multimedia channels, and hypertext documents. Here is a non-exclusive list of a variety of information collected in digital form in databases and in flat files.

Business transactions: Every transaction in the business industry is (often) "memorized" for perpetuity.  Such transactions are usually time related and can be inter-business deals such as purchases, exchanges, banking, stock, etc., or intra-business operations such as management of in-house wares and assets. Large department stores, for

example, thanks to the widespread use of bar codes, store millions of transactions daily representing often terabytes of data. Storage space is not the major problem, as the price of hard disks is continuously dropping, but the effective use of the data in a reasonable time frame for competitive decision-making is definitely the most important problem to solve for businesses that struggle to survive in a highly competitive world.

Scientific data:**ð** Whether in a Swiss nuclear accelerator laboratory counting particles, in the Canadian forest studying readings from a grizzly bear radio collar, on a South Pole iceberg gathering data about oceanic activity, or in an American university investigating human psychology, our society is amassing colossal amounts of scientific data that need to be analyzed. Unfortunately, we can capture and store more new data faster than we can analyze the old data already accumulated.

Medical and personal data: From government census to personnel and customer files, very large collections of information are continuously gathered about individuals and groups. Governments, companies and organizations such as hospitals, are stockpiling very important quantities of personal data to help them manage human resources, better understand a market, or simply assist clientele. Regardless of the privacy issues this type of data often reveals, this information is collected, used and even shared. When correlated with other data this information can shed light on customer behaviour and the like.

Surveillance video and pictures: With the amazing collapse of video camera prices, video cameras are becoming ubiquitous. Video tapes from surveillance cameras are usually recycled and thus the content is lost. However, there is a tendency today to store the tapes and even digitize them for future use and analysis.

Satellite sensing: There is a countless number of satellites around the globe: some are geo-stationary above a region, and some are orbiting around the Earth, but all are sending a non-stop stream of data to the surface. NASA, which controls a large number of satellites, receives more data every second than what all NASA researchers and engineers can cope with. Many satellite pictures and data are made public as soon as they are received in the hopes that other researchers can analyze them.

Games: Our society is collecting a tremendous amount of data and statistics about games, players and athletes. From hockey scores, basketball passes and car-racing lapses, to swimming times, boxer**ð**s pushes and chess positions, all the data are stored. Commentators and journalists are using this information for reporting, but trainers and athletes would want to exploit this data to improve performance and better understand opponents.

Digital media: The proliferation of cheap scanners, desktop video cameras and digital cameras is one of the causes of the explosion in digital media repositories. In addition, many radio stations, television channels and film studios are digitizing their audio and video collections to improve the management of their multimedia assets. Associations such as the NHL and the NBA have already started converting their huge game collection into digital forms.

CAD and Software engineering data: There are a multitude of Computer Assisted Design (CAD) systems for architects to design buildings or engineers to conceive system components or circuits. These systems are generating a tremendous amount of data. Moreover, software engineering is a source of considerable similar data with code, function libraries, objects, etc., which need powerful tools for management and maintenance.

Virtual Worlds: There are many applications making use of three-dimensional virtual spaces. These spaces and the objects they contain are described with special languages such as VRML. Ideally, these virtual spaces are described in such a way that they can share objects and places. There is a remarkable amount of virtual reality object and space repositories available. Management of these repositories as well as content-based search and retrieval from these repositories are still research issues, while the size of the collections continues to grow.

Text reports and memos (e-mail messages): Most of the communications within and between companies or research organizations or even private people, are based on reports and memos in textual forms often exchanged by e-mail. These messages are regularly stored in digital form for future use and reference creating formidable digital libraries.

The World Wide Web repositories: Since the inception of the World Wide Web in 1993, documents of all sorts of formats, content and description have been collected and inter-connected with hyperlinks making it the largest repository of data ever built. Despite its dynamic and unstructured nature, its heterogeneous characteristic, and its very often redundancy and inconsistency, the World Wide Web is the most important data collection regularly used for reference because of the broad variety of topics covered and the infinite contributions of resources and publishers. Many believe that the World Wide Web will become the compilation of human knowledge.

What are Data Mining and Knowledge Discovery?

With the enormous amount of data stored in files, databases, and other repositories, it is increasingly important, if not necessary, to develop powerful means for analysis and perhaps interpretation of such data and for the extraction of interesting knowledge that could help in decision-making.

Data Mining, also popularly known as *Knowledge Discovery in Databases* (KDD), refers to the nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases. While data mining and knowledge discovery in databases (or KDD) are frequently treated as synonyms, data mining is actually part of the knowledge discovery process. The following figure (Figure 1.1) shows data mining as a step in an iterative knowledge discovery process.

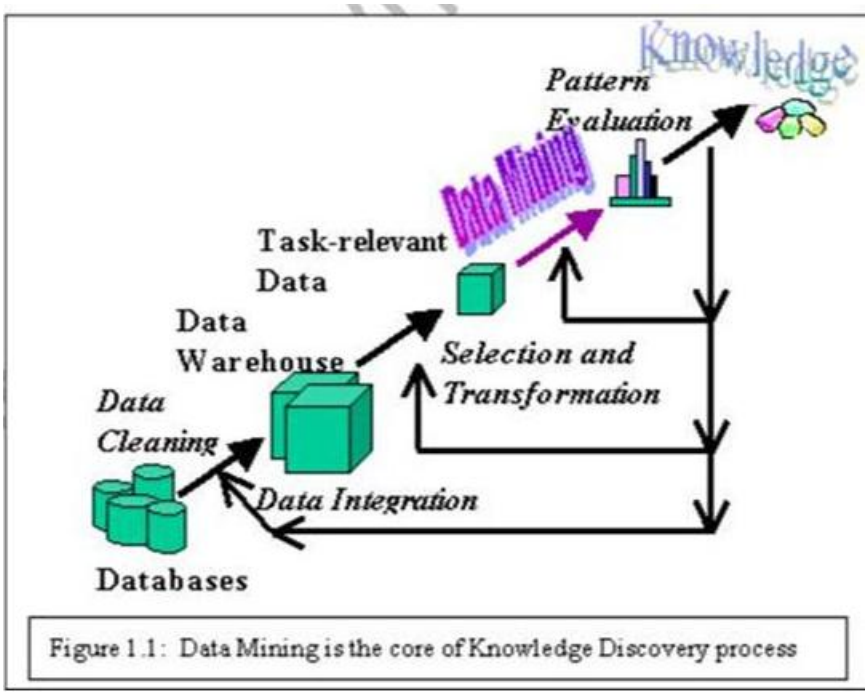


Figure 1.1: Data Mining is the core of Knowledge Discovery process

The Knowledge Discovery in Databases process comprises of a few steps leading from raw data collections to some form of new knowledge. The iterative process consists of the following steps:

Data cleaning: also known as data cleansing, it is a phase in which noise data and irrelevant data are removed from the collection.

Data integration: at this stage, multiple data sources, often heterogeneous, may be combined in a common source.

Data selection:**&#65533;** at this step, the data relevant to the analysis is decided on and retrieved from the data collection.

Data transformation: also known as data consolidation, it is a phase in which the selected data is transformed into forms appropriate for the mining procedure.

Data mining:**&#65533;** it is the crucial step in which clever techniques are applied to extract patterns potentially useful.

Pattern evaluation:**&#65533;** in this step, strictly interesting patterns representing knowledge are identified based on given measures.

Knowledge representation: is the final phase in which the discovered knowledge is visually represented to the user. This essential step uses visualization techniques to help users understand and interpret the data mining results.

It is common to combine some of these steps together. For instance, *data cleaning* and *data integration* can be performed together as a pre-processing phase to generate a data warehouse. *Data selection* and *data transformation* can also be combined where the consolidation of the data is the result of the selection, or, as for the case of data warehouses, the selection is done on transformed data.

The KDD is an iterative process. Once the discovered knowledge is presented to the user, the evaluation measures can be enhanced, the mining can be further refined, new data can be selected or further transformed, or new data sources can be integrated, in order to get different, more appropriate results.

Data mining derives its name from the similarities between searching for valuable information in a large database and mining rocks for a vein of valuable ore. Both imply either sifting through a large amount of material or ingeniously probing the material to exactly pinpoint where the values reside. It is, however, a misnomer, since mining for gold in rocks is usually called "gold mining" and not "rock mining", thus by analogy, data mining should have been called "knowledge mining" instead. Nevertheless, data mining became the accepted customary term, and very rapidly a trend that even overshadowed more general terms such as knowledge discovery in databases (KDD) that describe a more complete process. Other similar terms referring to data mining are: data dredging, knowledge extraction and pattern discovery.

What kind of Data can be mined?

In principle, data mining is not specific to one type of media or data. Data mining should be applicable to any kind of information repository. However, algorithms and approaches may differ when applied to different types of data. Indeed, the challenges presented by different types of data vary significantly. Data mining is being put into use and studied for databases, including relational databases, object-relational databases and object-oriented databases, data warehouses, transactional databases, unstructured and semi-structured repositories such as the World Wide Web, advanced databases such as spatial databases, multimedia databases, time-series databases and textual databases, and even flat files. Here are some examples in more detail:

Flat files: Flat files are actually the most common data source for data mining algorithms, especially at the research level. Flat files are simple data files in text or binary format with a structure known by the data mining algorithm to be applied. The data in these files can be transactions, time-series data, scientific measurements, etc.

Relational Databases

Relational Databases: Briefly, a relational database consists of a set of tables containing either values of entity attributes, or values of attributes from entity relationships. Tables have columns and rows, where columns represent attributes and rows represent tuples. A tuple in a relational table corresponds to either an object or a relationship between objects and is identified by a set of attribute values representing a unique key. These relations are just a subset of what could be a database for the video store and is given as an example.
The most commonly used query language for relational database is SQL, which allows retrieval and manipulation of the data stored in the tables, as well as the calculation of aggregate functions such as average, sum, min, max and count. For instance, an SQL query toselect the videos grouped by category would be:

SELECT count(*) FROM Items WHERE type=video GROUP BY category.

Data mining algorithms using relational databases can be more versatile than data mining algorithms specifically written for flat files, since they can take advantage of the structure inherent to relational databases. While data mining can benefit from SQL for data selection, transformation and consolidation, it goes beyond what SQL could provide, such as predicting, comparing, detecting deviations, etc.

Data Warehouses

Data Warehouses: A data warehouse as a storehouse, is a repository of data collected from multiple data sources (often heterogeneous) and is intended to be used as a whole under the same unified schema. A data warehouse gives the option to analyze data from different sources under the same roof. Let us suppose that OurVideoStore becomes a franchise in North America. Many video stores belonging to OurVideoStore company may have different databases and different structures. If the executive of the company wants to access the data from all stores for strategic decision-making, future direction, marketing, etc., it would be more appropriate to store all the data in one site with a homogeneous structure that allows interactive analysis. In other words, data from the different stores would be loaded, cleaned, transformed and integrated together. To facilitate decision-making and multi-dimensional views, data warehouses are usually modeled by a multi-dimensional data structure. Figure 1.3 shows an example of a three dimensional subset of a data cube structure used for OurVideoStore data warehouse.

The figure shows summarized rentals grouped by film categories, then a cross table of summarized rentals by film categories and time (in quarters).  The data cube gives the summarized rentals along three dimensions: category, time, and city. A cube contains cells that store values of some aggregate measures (in this case rental counts), and special cells that

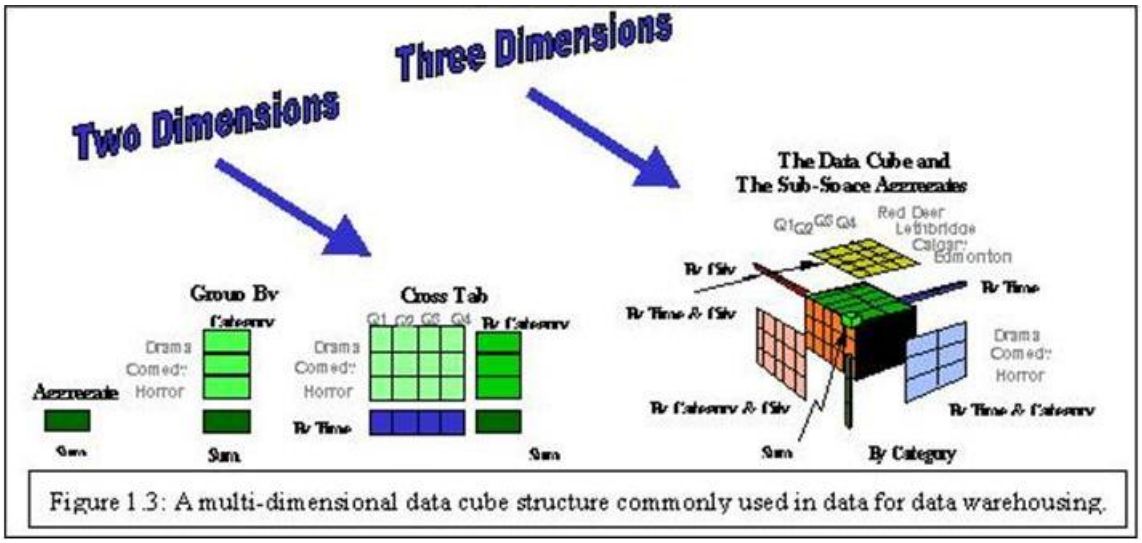


Figure 1.3: A multi-dimensional data cube structure commonly used in data for data warehousing.

store summations along dimensions. Each dimension of the data cube contains a hierarchy of values for one attribute.

Because of their structure, the pre-computed summarized data they contain and the hierarchical attribute values of their dimensions, data cubes are well suited for fast interactive querying and analysis of data at different conceptual levels, known as On-Line Analytical Processing (OLAP). OLAP operations allow the navigation of data at different levels of abstraction, such as drill-down, roll-up, slice, dice, etc. Figure 1.4 illustrates the drill-down (on the time dimension) and roll-up (on the location dimension) operations.

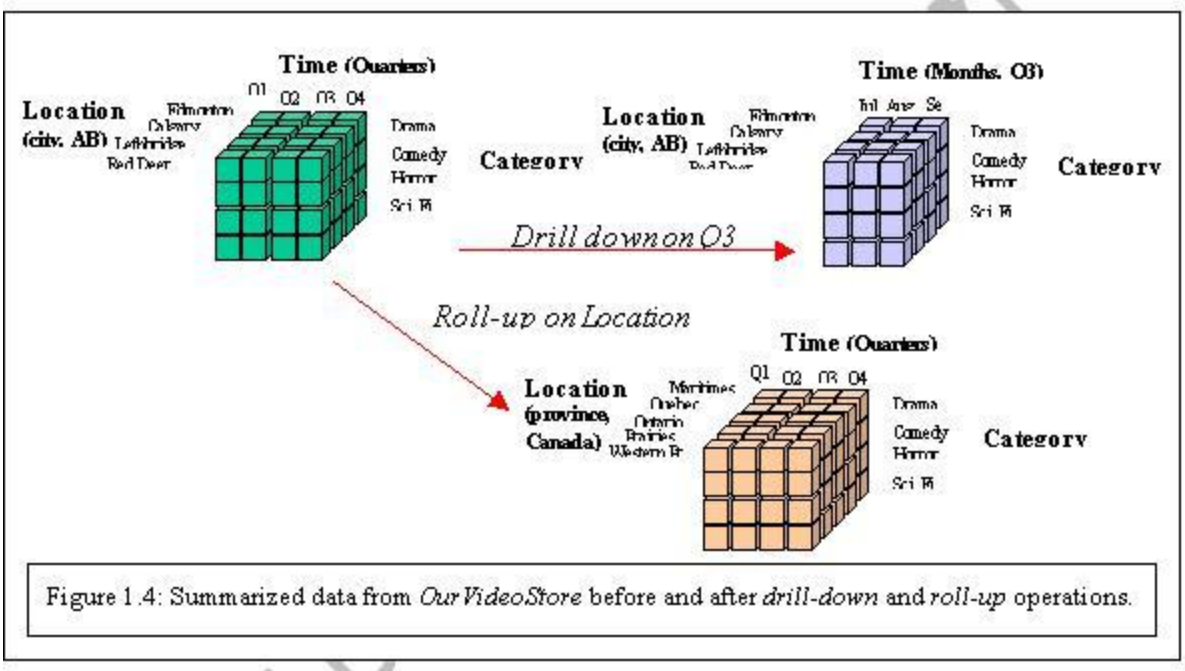


Figure 1.4: Summarized data from *OurVideoStore* before and after *drill-down* and *roll-up* operations.

Transactional Database

Transaction Databases: A transaction database is a set of records representing transactions, each with a time stamp, an identifier and a set of items. Associated with the transaction files could also be descriptive data for the items. For example, in the case of the video store, the rentals table such as shown in Figure 1.5, represents the transaction database. Each record is a rental contract with a customer identifier, a date, and the list of items rented (i.e. video tapes, games, VCR, etc.). Since relational databases do not allow nested tables (i.e. a set as attribute value), transactions are usually stored in flat files or stored in two normalized transaction tables, one for the transactions and one for the transaction items. One typical data mining analysis on such data is the so-called market basket analysis or association rules in which associations between items occurring together or in sequence are studied.

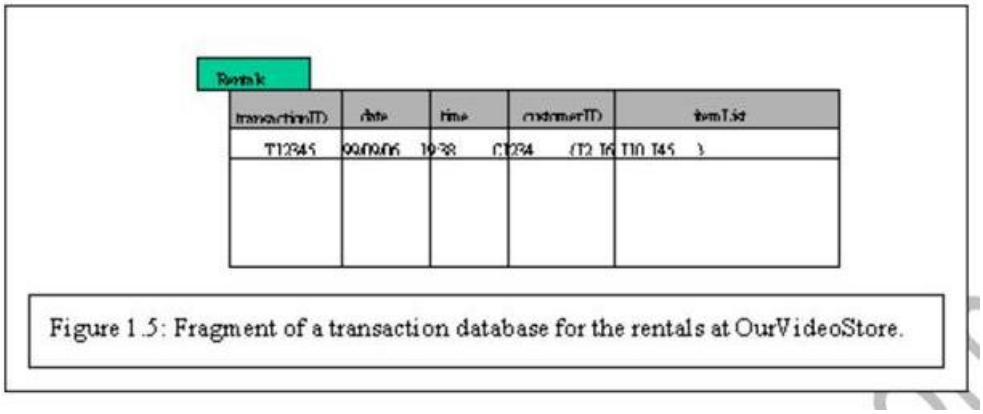


Figure 1.5: Fragment of a transaction database for the rentals at OurVideoStore.

Advanced Database Systems and Advanced Database Applications

Multimedia Databases: Multimedia databases include video, images, audio and text media. They can be stored on extended object-relational or object-oriented databases, or simply on a file system. Multimedia is characterized by its high dimensionality, which makes data mining even more challenging. Data mining from multimedia repositories may require computer vision, computer graphics, image interpretation, and natural language processing methodologies.

Spatial Databases: Spatial databases are databases that, in addition to usual data, store geographical information like maps, and global or regional positioning. Such spatial databases present new challenges to data mining algorithms.

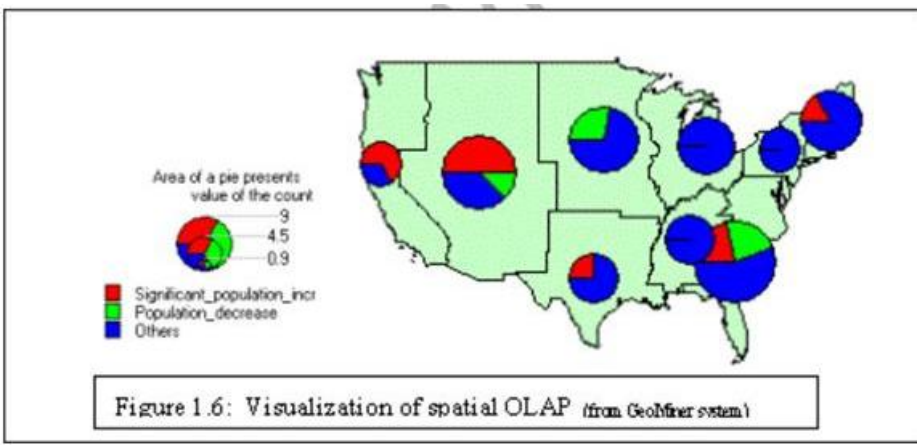


Figure 1.6: Visualization of spatial OLAP (from GeoMiner system)

Time-Series Databases: Time-series databases contain time related data such stock market data or logged activities. These databases usually have a continuous flow of new data coming in, which sometimes causes the need for a challenging real time analysis. Data mining in such databases commonly includes the study of trends and correlations between evolutions of different variables, as well as the prediction of trends and movements of the variables in time. Figure 1.7 shows some examples of time-series data.

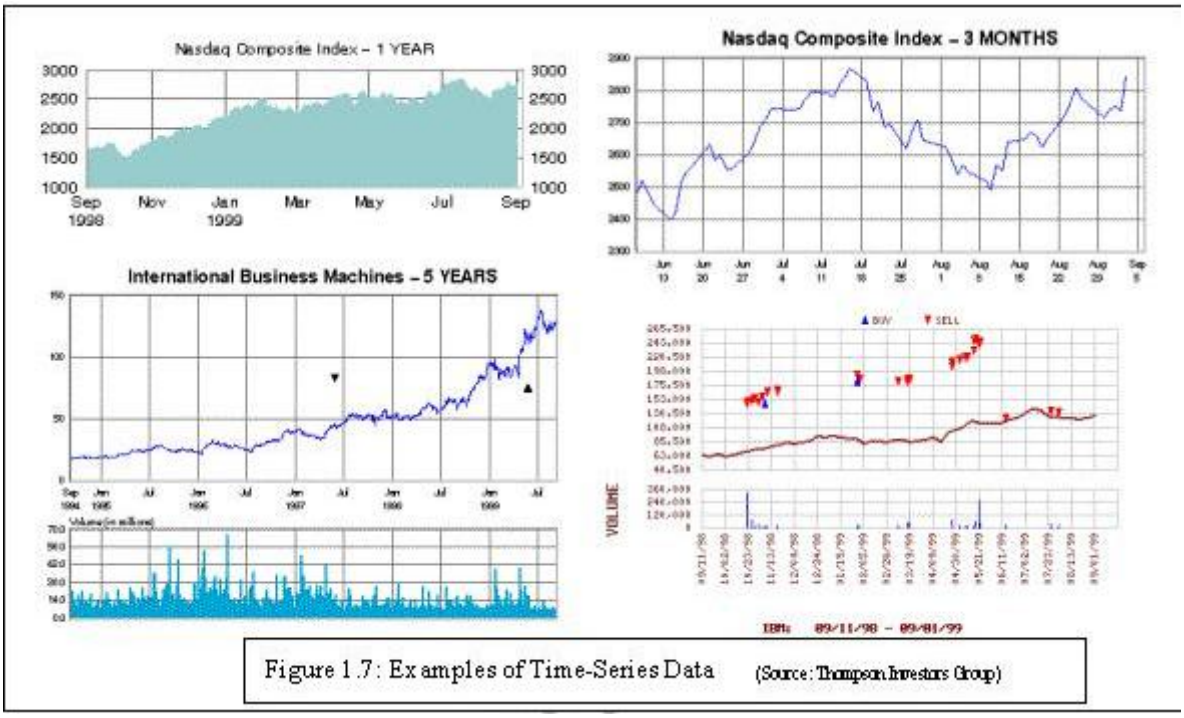


Figure 1.7: Examples of Time-Series Data (Source: Thompson Investors Group)

World Wide Web: The World Wide Web is the most heterogeneous and dynamic repository available. A very large number of authors and publishers are continuously contributing to its

growth and metamorphosis, and a massive number of users are accessing its resources daily. Data in the World Wide Web is organized in inter-connected documents. These documents can be text, audio, video, raw data, and even applications. Conceptually, the World Wide Web is comprised of three major components: The content of the Web, which encompasses documents available; the structure of the Web, which covers the hyperlinks and the relationships between documents; and the usage of the web, describing how and when the resources are accessed. A fourth dimension can be added relating the dynamic nature or evolution of the documents. Data mining in the World Wide Web, or web mining, tries to address all these issues and is often divided into web content mining, web structure mining and web usage mining.

Data Mining Functionalities

What can be discovered?

The kinds of patterns that can be discovered depend upon the data mining tasks employed. By and large, there are two types of data mining tasks: *descriptive data mining* tasks that describe the general properties of the existing data, and *predictive data mining* tasks that attempt to do predictions based on inference on available data.

The data mining functionalities and the variety of knowledge they discover are briefly presented in the following list:

Characterization: Data characterization is a summarization of general features of objects in a target class, and produces what is called *characteristic rules*. The data relevant to a user-specified class are normally retrieved by a database query and run through a summarization module to extract the essence of the data at different levels of abstractions. For example, one may want to characterize the OurVideoStore customers who regularly rent more than 30 movies a year. With concept hierarchies on the attributes describing the target class,

the *attribute-oriented induction* method can be used, for example, to carry out data summarization. Note that with a data cube containing summarization of data, simple OLAP operations fit the purpose of data characterization.

Discrimination: Data discrimination produces what are called *discriminant rules* and is basically the comparison of the general features of objects between two classes referred to as the *target class* and the *contrasting class*. For example, one may want to compare the general characteristics of the customers who rented more than 30 movies in the last year with those whose rental account is lower than 5. The techniques used for data discrimination are very similar to the techniques used for data characterization with the exception that data discrimination results include comparative measures.

Association analysis: Association analysis is the discovery of what are commonly

called *association rules*. It studies the frequency of items occurring together in transactional databases, and based on a threshold called *support*, identifies the frequent item sets. Another

threshold, *confidence*, which is the conditional probability than an item appears in a transaction when another item appears, is used to pinpoint association rules. Association analysis is commonly used for market basket analysis. For example, it could be useful for the OurVideoStore manager to know what movies are often rented together or if there is a relationship between renting a certain type of movies and buying popcorn or pop. The discovered association rules are of the form: P -> Q [s,c], where P and Q are conjunctions of attribute value-pairs, and s (for support) is the probability that P and Q appear together in a transaction and c (for confidence) is the conditional probability that Q appears in a transaction when P is present. For example, the hypothetic association rule:

RentType(X, "game") AND Age(X, "13-19") -> Buys(X, "pop") [s=2% ,c=55%]

would indicate that 2% of the transactions considered are of customers aged between 13 and 19 who are renting a game and buying a pop, and that there is a certainty of 55% that teenage customers who rent a game also buy pop.

Classification: Classification analysis is the organization of data in given classes. Also known as *supervised classification*, the classification uses given class labels to order the objects in the data collection. Classification approaches normally use a *training set* where all objects are already associated with known class labels. The classification algorithm learns from the training set and builds a model. The model is used to classify new objects. For example, after starting a credit policy, the OurVideoStore managers could analyze the customers' behaviours vis-à-vis their credit, and label accordingly the customers who received credits with three possible labels "safe", "risky" and "very risky". The classification analysis would generate a model that could be used to either accept or reject credit requests in the future.

Prediction: Prediction has attracted considerable attention given the potential implications of successful forecasting in a business context. There are two major types of predictions: one can either try to predict some unavailable data values or pending trends, or predict a class label for some data. The latter is tied to classification. Once a classification model is built based on a training set, the class label of an object can be foreseen based on the attribute values of the object and the attribute values of the classes. Prediction is however more often referred to the forecast of missing numerical values, or increase/ decrease trends in time related data. The major idea is to use a large number of past values to consider probable future values.

Clustering: Similar to classification, clustering is the organization of data in classes. However, unlike classification, in clustering, class labels are unknown and it is up to the clustering algorithm to discover acceptable classes. Clustering is also called *unsupervised classification*, because the classification is not dictated by given class labels. There are many clustering approaches all based on the principle of maximizing the similarity between objects in a same class (*intra-class similarity*) and minimizing the similarity between objects of different classes (*inter-class similarity*).

Outlier analysis: Outliers are data elements that cannot be grouped in a given class or cluster. Also known as *exceptions* or *surprises*, they are often very important to identify. While outliers can be considered noise and discarded in some applications, they can reveal important knowledge in other domains, and thus can be very significant and their analysis valuable.

Evolution and deviation analysis: Evolution and deviation analysis pertain to the study of time related data that changes in time. Evolution analysis models evolutionary trends in data, which consent to characterizing, comparing, classifying or clustering of time related data. Deviation analysis, on the other hand, considers differences between measured values and expected values, and attempts to find the cause of the deviations from the anticipated values.

It is common that users do not have a clear idea of the kind of patterns they can discover or need to discover from the data at hand. It is therefore important to have a versatile and inclusive data mining system that allows the discovery of different kinds of knowledge and at different levels of abstraction. This also makes interactivity an important attribute of a data mining system.

Is all that is discovered interesting and useful?

Data mining allows the discovery of knowledge potentially useful and unknown. Whether the knowledge discovered is new, useful or interesting, is very subjective and depends upon the application and the user. It is certain that data mining can generate, or discover, a very large number of patterns or rules. In some cases the number of rules can reach the millions. One can even think of a meta-mining phase to mine the oversized data mining results. To reduce the number of patterns or rules discovered that have a high probability to be non-interesting, one has to put a measurement on the patterns. However, this raises the problem of completeness. The user would want to discover ***all*** rules or patterns, ***but only*** those that are ***interesting***. The measurement of how interesting a discovery is, often called ***interestingness***, can be based on quantifiable objective elements such as *validity* of the patterns when tested on new data with some degree of *certainty*, or on some subjective depictions such as *understandability* of the patterns, *novelty* of the patterns, or *usefulness*.

Discovered patterns can also be found interesting if they confirm or validate a hypothesis sought to be confirmed or unexpectedly contradict a common belief. This brings the issue of describing what is interesting to discover, such as meta-rule guided discovery that describes forms of rules before the discovery process, and interestingness refinement languages that interactively query the results for interesting patterns after the discovery phase. Typically, measurements for interestingness are based on thresholds set by the user. These thresholds define the completeness of the patterns discovered.

Identifying and measuring the interestingness of patterns and rules discovered, or to be discovered, is essential for the evaluation of the mined knowledge and the KDD process as a whole. While some concrete measurements exist, assessing the interestingness of discovered knowledge is still an important research issue.

Pattern Classification of Data Mining Systems

How do we categorize data mining systems?

There are many data mining systems available or being developed. Some are specialized systems dedicated to a given data source or are confined to limited data mining functionalities, other are more versatile and comprehensive. Data mining systems can be categorized according to various criteria among other classification are the following:

Classification according to the type of data source mined: this classification categorizes data mining systems according to the type of data handled such as spatial data, multimedia data, time-series data, text data, World Wide Web, etc.

Classification according to the data model drawn on: this classification categorizes data mining systems based on the data model involved such as relational database, object-oriented database, data warehouse, transactional, etc.

Classification according to the king of knowledge discovered: this classification categorizes data mining systems based on the kind of knowledge discovered or data mining functionalities, such as characterization, discrimination, association, classification, clustering, etc. Some systems tend to be comprehensive systems offering several data mining functionalities together.

Classification according to mining techniques used: Data mining systems employ and provide different techniques. This classification categorizes data mining systems according to the data analysis approach used such as machine learning, neural networks, genetic algorithms, statistics, visualization, database-oriented or data warehouse-oriented, etc. The classification can also take into account the degree of user interaction involved in the data mining process such as query-driven systems, interactive exploratory systems, or autonomous systems. A comprehensive system would provide a wide variety of data mining techniques to fit different situations and options, and offer different degrees of user interaction.

Major issues in Data Mining

What are the issues in Data Mining?

Data mining algorithms embody techniques that have sometimes existed for many years, but have only lately been applied as reliable and scalable tools that time and again outperform older classical statistical methods. While data mining is still in its infancy, it is becoming a trend and ubiquitous. Before data mining develops into a conventional, mature and trusted discipline, many still pending issues have to be addressed. Some of these issues are addressed below. Note that these issues are not exclusive and are not ordered in any way.

Security and social issues: Security is an important issue with any data collection that is shared and/or is intended to be used for strategic decision-making. In addition, when data is collected for customer profiling, user behaviour understanding, correlating personal data with other information, etc., large amounts of sensitive and private information about individuals or companies is gathered and stored. This becomes controversial given the confidential nature of some of this data and the potential illegal access to the information. Moreover, data mining could disclose new implicit knowledge about individuals or groups that

could be against privacy policies, especially if there is potential dissemination of discovered information. Another issue that arises from this concern is the appropriate use of data mining. Due to the value of data, databases of all sorts of content are regularly sold, and because of the competitive advantage that can be attained from implicit knowledge discovered, some important information could be withheld, while other information could be widely distributed and used without control.

User interface issues: The knowledge discovered by data mining tools is useful as long as it is interesting, and above all understandable by the user. Good data visualization eases the interpretation of data mining results, as well as helps users better understand their needs. Many data exploratory analysis tasks are significantly facilitated by the ability to see data in an appropriate visual presentation. There are many visualization ideas and proposals for effective data graphical presentation. However, there is still much research to accomplish in order to obtain good visualization tools for large datasets that could be used to display and manipulate mined knowledge.  The major issues related to user interfaces and visualization are "screen real-estate", information rendering, and interaction. Interactivity with the data and data mining results is crucial since it provides means for the user to focus and refine the mining tasks, as well as to picture the discovered knowledge from different angles and at different conceptual levels.

Mining methodology issues: These issues pertain to the data mining approaches applied and their limitations. Topics such as versatility of the mining approaches, the diversity of data available, the dimensionality of the domain, the broad analysis needs (when known), the assessment of the knowledge discovered, the exploitation of background knowledge and metadata, the control and handling of noise in data, etc. are all examples that can dictate mining methodology choices. For instance, it is often desirable to have different data mining methods available since different approaches may perform differently depending upon the data at hand. Moreover, different approaches may suit and solve user's needs differently.

Most algorithms assume the data to be noise-free. This is of course a strong assumption. Most datasets contain exceptions, invalid or incomplete information, etc., which may complicate, if not obscure, the analysis process and in many cases compromise the accuracy of the results. As a consequence, data preprocessing (data cleaning and transformation) becomes vital. It is often seen as lost time, but data cleaning, as time-consuming and frustrating as it may be, is one of the most important phases in the knowledge discovery process. Data mining techniques should be able to handle noise in data or incomplete information.

More than the size of data, the size of the search space is even more decisive for data mining techniques. The size of the search space is often depending upon the number of dimensions in the domain space. The search space usually grows exponentially when the number of dimensions increases. This is known as the *curse of dimensionality*. This "curse" affects so badly the performance of some data mining approaches that it is becoming one of the most urgent issues to solve.

Performance issues: Many artificial intelligence and statistical methods exist for data analysis and interpretation. However, these methods were often not designed for the very large data sets data mining is dealing with today. Terabyte sizes are common. This raises the issues of scalability and efficiency of the data mining methods when processing considerably large data. Algorithms with exponential and even medium-order polynomial complexity cannot be of practical use for data mining. Linear algorithms are

usually the norm. In same theme, sampling can be used for mining instead of the whole dataset. However, concerns such as completeness and choice of samples may arise. Other topics in the issue of performance are *incremental updating*, and parallel programming. There is no doubt that parallelism can help solve the size problem if the dataset can be subdivided and the results can be merged later. Incremental updating is important for merging results from parallel mining, or updating data mining results when new data becomes available without having to re-analyze the complete dataset.

Data source issues: There are many issues related to the data sources, some are practical such as the diversity of data types, while others are philosophical like the data glut problem. We certainly have an excess of data since we already have more data than we can handle and we are still collecting data at an even higher rate. If the spread of database management systems has helped increase the gathering of information, the advent of data mining is certainly encouraging more data harvesting. The current practice is to collect as much data as possible now and process it, or try to process it, later. The concern is whether we are collecting the right data at the appropriate amount, whether we know what we want to do with it, and whether we distinguish between what data is important and what data is insignificant. Regarding the practical issues related to data sources, there is the subject of heterogeneous databases and the focus on diverse complex data types. We are storing different types of data in a variety of repositories. It is difficult to expect a data mining system to effectively and efficiently achieve good mining results on all kinds of data and sources. Different kinds of data and sources may require distinct algorithms and methodologies. Currently, there is a focus on relational databases and data warehouses, but other approaches need to be pioneered for other specific complex data types. A versatile data mining tool, for all sorts of data, may not be realistic. Moreover, the proliferation of heterogeneous data sources, at structural and semantic levels, poses important challenges not only to the database community but also to the data mining community.

Pre-process the Data- Data Cleaning, Data Integration and Transformation

There is a huge amount of data available in the Information Industry. This data is of no use until it is converted into useful information. It is necessary to analyze this huge amount of data and extract useful information from it.

Extraction of information is not the only process we need to perform; data mining also involves other processes such as Data Cleaning, Data Integration, Data Transformation, Data Mining, Pattern Evaluation and Data Presentation. Once all these processes are over, we would be able to use this information in many applications such as Fraud Detection, Market Analysis, Production Control, Science Exploration, etc.

Why preprocessing ?

1. Real world data are generally
    - Incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - Noisy: containing errors or outliers

- Inconsistent: containing discrepancies in codes or names

2. Tasks in data preprocessing
   - Data cleaning: fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.
   - Data integration: using multiple databases, data cubes, or files.
   - Data transformation: normalization and aggregation.
   - Data reduction: reducing the volume but producing the same or similar analytical results.
   - Data discretization: part of data reduction, replacing numerical attributes with nominal ones.

Data cleaning

1. Fill in missing values (attribute or class value):
   - Ignore the tuple: usually done when class label is missing.
   - Use the attribute mean (or majority nominal value) to fill in the missing value.
   - Use the attribute mean (or majority nominal value) for all samples belonging to the same class.
   - Predict the missing value by using a learning algorithm: consider the attribute with the missing value as a dependent (class) variable and run a learning algorithm (usually Bayes or decision tree) to predict the missing value.
2. Identify outliers and smooth out noisy data:
   - Binning
     - Sort the attribute values and partition them into bins (see "Unsupervised discretization" below);
     - Then smooth by bin means, bin median, or bin boundaries.
   - Clustering: group values in clusters and then detect and remove outliers (automatic or manual)
   - Regression: smooth by fitting the data into regression functions.
3. Correct inconsistent data: use domain knowledge or expert decision.

Data transformation

1. Normalization:
   - Scaling attribute values to fall within a specified range.
     - Example: to transform V in [min, max] to V' in [0,1], apply V'=(V-Min)/(Max-Min)
   - Scaling by using mean and standard deviation (useful when min and max are unknown or when there are outliers): V'=(V-Mean)/StDev
2. Aggregation: moving up in the concept hierarchy on numeric attributes.
3. Generalization: moving up in the concept hierarchy on nominal attributes.
4. Attribute construction: replacing or adding new attributes inferred by existing attributes.

Data reduction

1. Reducing the number of attributes
   - Data cube aggregation: applying roll-up, slice or dice operations.
   - Removing irrelevant attributes: attribute selection (filtering and wrapper methods), searching the attribute space (see Lecture 5: Attribute-oriented analysis).
   - Principle component analysis (numeric attributes only): searching for a lower dimensional space that can best represent the data..
2. Reducing the number of attribute values
   - Binning (histograms): reducing the number of attributes by grouping them into intervals (bins).
   - Clustering: grouping values in clusters.
   - Aggregation or generalization
3. Reducing the number of tuples
   - Sampling

Discretization and generating concept hierarchies

1. Unsupervised discretization - class variable is not used.
   - Equal-interval (equiwidth) binning: split the whole range of numbers in intervals with equal size.
   - Equal-frequency (equidepth) binning: use intervals containing equal number of values.
2. Supervised discretization - uses the values of the class variable.
   - Using class boundaries. Three steps:
     - Sort values.
     - Place breakpoints between values belonging to different classes.
     - If too many intervals, merge intervals with equal or similar class distributions.
   - Entropy (information)-based discretization. Example:
     - Information in a class distribution:
       - Denote a set of five values occurring in tuples belonging to two classes (+ and -) as [+,+,+,-,-]
       - That is, the first 3 belong to "+" tuples and the last 2 - to "-" tuples
       - Then, Info([+,+,+,-,-]) = -(3/5)*log(3/5)-(2/5)*log(2/5) (logs are base 2)
       - 3/5 and 2/5 are relative frequencies (probabilities)
       - Ignoring the order of the values, we can use the following notation: [3,2] meaning 3 values from one class and 2 - from the other.
       - Then, Info([3,2]) = -(3/5)*log(3/5)-(2/5)*log(2/5)
     - Information in a split (2/5 and 3/5 are weight coefficients):
       - Info([+,+],[+,-,-]) = (2/5)*Info([+,+]) + (3/5)*Info([+,-,-])
       - Or, Info([2,0],[1,2]) = (2/5)*Info([2,0]) + (3/5)*Info([1,2])
     - Method:
       - Sort the values;
       - Calculate information in all possible splits;
       - Choose the split that minimizes information;
       - Do not include breakpoints between values belonging to the same class (this will increase information);

- Apply the same to the resulting intervals until some stopping criterion is satisfied.

3. Generating concept hierarchies: recursively applying partitioning or discretization methods.

.

Possible Questions

PART – A (20 Marks)

(Q.No 1 to 20 Online Examinations)

PART – B (2 Marks)

1. What is data mining?
2. Define : Relational database
3. Define : Transactional database
4. List the advanced database applications.
5. What is the purpose of preprocess in data Mining?

PART – C (6 Marks)

1. Discuss about the major issues in Data mining.
2. Discuss about data warehouse in detail.
3. Discuss in detail about data mining functionalities.
4. Illustrate data integration and data transformation with suitable examples.
5. Discuss in detail about advanced database system and advanced database applications.
6. Explain about data cleaning in detail.
7. Explain about Relational databases with suitable examples
8. Discuss about the Classification of data mining systems
9. List and explain the advanced database systems
10. Briefly discuss the transaction database systems.

Karpagam Academy of Higher Education

Coimbatore - 641021

(For the candidates admitted from 2017 onwards)

Department of CS,CA & IT

JNIT - I:(Objective Type/Multiple Choice Questions Carries one mark)

Data mining and Data Warehousing 16CAP504D

UNIT -1						
Q.No	**Questions**	**Option1**	**Option2**	**Option3**	**Option4**	**Answer**
1	____to identify the truly interesting patterns representing knowledge based on some interestingness measures	Knowledge Presentation	Pattern Evaluation	Data Mining	Data Transformation	Pattern Evaluation
2	____refers to extracting or mining of knowledge from large amount of data	Data Mining	Data model	Data Cube	Data relation	Data Mining
3	_____ maps data into predefined groups or classes	regression	classification	clustering	prediction	classification
4	_____ is used to map a data item to a real valued prediction variable.	regression	classification	clustering	prediction	regression
5	Classification of input patterns by using its similarity to the predefined classes is called____	prediction	classification	pattern recognition	regression	pattern recognition
6	A special type of clustering is called _____	classification	segmentation	regression	prediction	segmentation
7	_____ maps data into subsets with associated simple description.	classification	segmentation	pattern recognition	summarization	summarization
8	Which rule is used to identify the specific type of data association	delta rule	association rule	classification rule	segmentation rule	association rule

9	Which is used to determine sequential patterns in data?	sequence discovery	segmentation	regression	random analysis	sequence discovery
10	_____ is the process of finding useful information and patterns in data	datamining	KDD	selection	transfermation	KDD
11	_____ is the use of algorithms to extract the information and pattern derived by the knowledge discovery	datamining	KDD	selection	transfermation	datamining
12	The KDD process is often said to be _____	trivial	nontrivial	significant	consequential	nontrivial
13	Extreme values that occur infrequently, may actually be removed is called _____	outliers	outlayers	statistics	frequency	outliers
14	Tuples is also known as	Records	Rows	All the above	None of the above	Records
15	The major goal of _____ is to be able to describe the result in meaningful manner.	datamining	KDD	selection	transfermation	KDD
16	_____ techniques often involve sophisticated multimedia and graphics presentations.	summarization	conceptualization	visualization	decription	visualization
17	Describing a large database can be viewed as using _____ to help uncover hidden information about the data	approximation	search	induction	compression	approximation
18	_____ is used to proceed from very specific knowledge to more general information.	approximation	search	induction	compression	induction
19	_____ occurs when the model does not fit future states.	outliers	interception	overfitting	qureying	overfitting
20	Expand E-R	Entity relation	Entity record	Entry relationship	Entity relationship	Entity
21	Process of remove noise andinconsistent data is referred____	Data cleaning	Data Integration	Data Transformation	Data mining	Data Cleaning
22	Find out predictive model datamining task from the followings	clustering	summarization	association rule	regression	regression

23	Find out descriptive model datamining task from the followings	classification	sequence discovery	prediction	regression	sequence discovery
24	Which one is the example of data streams	Scientific and engineering data	Banking database	Multimedia	None	Scientific and engineering data
25	Which one is the correct binning methods	Equal frequency	Means	Boundaries	all of the above	Boundaries
26	Which one of the following is discrepancy tool	Data auditing	Data	Data cleaning tool	None of the above	Data auditing
27	Distance measures used to identify the _____ of different items in the database.	unlikeness	matches	alikeness	difference	unlikeness
28	Decision tree is otherwise called as _____ tree	clustering	classification	regression	correlation	classification
29	The _____ phase might remove redundant comparisons or remove subtrees to achieve better performance.	classification	clustering	splitting	pruning	pruning
30	The _____ technique to building a decision tree is based on information theory and attempts to minimize the espected number of comparisons.	bayes rule	backpropagation	ID3	perceptron	ID3
31	The process of combining multiple resources is called_____	Data cleaning	Data Integration	Data Transformation	Data mining	Data Integration
32	_____refer to data relavant to the analysis tasks are retrived from the database	Data cleaning	Data Integration	Data Transformation	Data Selection	Data Selection
33	____is an essential process where intelligence methods are applied in order to extract data.	Data cleaning	Data Integration	Data Transformation	Data mining	Data Mining
34	_____ is a learning technique that adjusts weights in the NN by propagating weight changes backward from the sink to the source nodes.	backpropagation	propagation	regression	correlation	backpropagation

35	A _____ is a class of function whose value decreases or increases with the distance from a central point.	propagation	association	radial basis function	bayes rule	radial basis function
36	A _____ is a single neuron with multiple inputs and one output.	CART	ID3	bayes rule	perceptron	perceptron
37	The _____ contains a predicate that can be evaluated as true or false against each tuple in the database.	consequent	antecedent	propagation	probability	antecedent
38	Data objects whose class label is known	Training	Training set	Training data	Training objects	Training set
39	_____ is a technique to estimate the likelihood of a property given the set of data as evidence or input.	bayes rule	ID3	backpropagation	CART	bayes rule
40	The initial hypothesis is called the _____ hypothesis	start	source	alternative	null	null
41	Rejection of the null hypothesis causes _____hypothesis.	not null	alternative	null	sink	alternative
42	The _____ is a procedure used to test the association between 2 observed variable values and to determine whether the values are statistically significant.	prediction	regression	chi-squared statistics	bayesian	chi-squared statistics
43	_____ is used to examine the degree to which the values for 2 variables behave similarly.	correlation	regression	similarity	dissimilarity	correlation
44	How do you overcome the missing value?	Use a global	Ignore the	Use the attribute	All the above	All the above
45	_____ can be viewed as a directed graph with source, sink and internal nodes	decision tree	clustering	classification	neural network	neural network
46	The _____ of an estimator is the difference between the espected value of the estimator and the actual value	unbias	bias	predict	unpredict	bias

47	The examples for presentation and visualization of data mining is___	Graphs	Curves	Crosstabs	all of the above	all of the above
48	The _____ is defined as the expected value of the squared difference between the estimate and the actual value	root mean square	root mean square error	mean squared error	jackknife	mean squared error
49	_____is called Meta data.	Data about data	Knowledge	Mining about mining	Process about process	Data about data
50	Data mining also known	Knowledge	Knowledge	Pattern analysis	All the above	likelihood
51	The _____ algorithm is an approach that solves the estimation problem with incomplete data	genetic algorithm	expectation maximization	crossover	rule-based class	expectation maximization
52	The _____ diagram shows the distribution of the data	scatter	box plot	histogram	graph	histogram
53	In box plot the total range of the data values is divided into four equal parts called _____	quartiles	quarter	quartales	quad	quartiles
54	Summary is also called____	Smoothing	Aggregation	Generalization	Normalization	Aggregation
55	Correlation analysis is achieved by	Correlation co efficient	Smoothing	Data scrubbing	Data auditing	Correlation co efficient
56	_____ testing attempts to find a model that explains the observed data by first creating a hypothesis and then testing against the data	regression	correlation	logarithmic	hypothesis	hypothesis
57	Which one is the smoothing technique	Binning	regression	clustering	all of the above	all of the above
58	In chi-square, _____ tables are used to evaluate the actual value in order to determine its significance	statistical	logarithmic	algebric	analytical	statistical
59	___is called low level data are replaced by higherlevel data	Smoothing	Aggregation	Generalization	Normalization	Generalization

60	___Performs a linear transformation on the original data	Normalization	Min-max Normalization	Attribute Selection	Smoothing	Min-max Normalization

UNIT II

Classification and Regression Algorithms : Naïve Bayes – Multiple Regression Analysis – Logistic Regression – k-Nearest Neighbour Classification – GMDH –Computing and Genetic Algorithms. Support Vector Machines : Linear SVM - SVM with soft margin – Linear kernel – Proximal SVM – Generating Datasets.

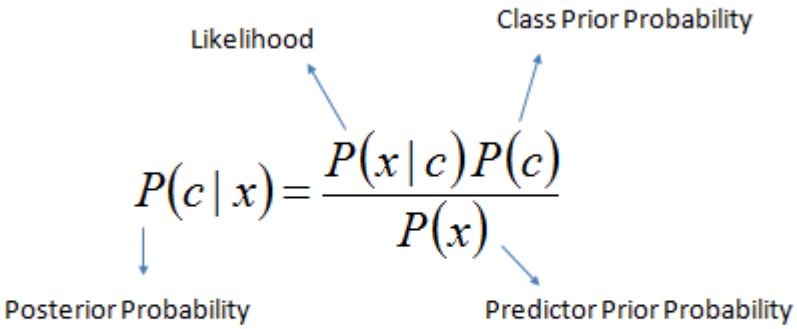
Cluster Analysis : Partitional Clusterings – k-medoids – Birch – DBSCAN – Optics – Graph Partitioning – CHAMELEON – COBWEB – GCLuto.

Naïve Bayes

The Naive Bayesian classifier is based on Bayes' theorem with independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naive Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods.

Algorithm

Bayes theorem provides a way of calculating the posterior probability, $P(c|x)$, from $P(c)$, $P(x)$, and $P(x|c)$. Naive Bayes classifier assume that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. This assumption is called class conditional independence.



$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

- $P(c|x)$ is the posterior probability of *class* (*target*) given *predictor* (*attribute*).
- $P(c)$ 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*.

Example:

The posterior probability can be calculated by first, constructing a frequency table for each attribute against the target. Then, transforming the frequency tables to likelihood tables and finally use the Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

$P(x \mid c) = P(Sunny \mid Yes) = 3/9 = 0.33$

Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

Likelihood Table		Play Golf		
		Yes	No	
Outlook	Sunny	3/9	2/5	5/14
	Overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

$P(x) = P(Sunny) = 5/14 = 0.36$

$P(c) = P(Yes) = 9/14 = 0.64$

Posterior Probability: $P(c \mid x) = P(Yes \mid Sunny) = 0.33 \times 0.64 \div 0.36 = 0.60$

The zero-frequency problem

Add 1 to the count for every attribute value-class combination (*Laplace estimator*) when an attribute value (*Outlook=Overcast*) doesn't occur with every class value (*Play Golf=no*).

Numerical Predictors

Numerical variables need to be transformed to their categorical counterparts (binning) before constructing their frequency tables. The other option we have is using the distribution of the numerical variable to have a good guess of the frequency. For example, one common practice is to assume normal distributions for numerical variables.

The probability density function for the normal distribution is defined by two parameters (mean and standard deviation).

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$ Mean

$$\sigma = \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2\right]^{0.5}$$ Standard deviation

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$ Normal distribution

Example:

		Humidity									*Mean*	*StDev*
Play Golf	yes	86	96	80	65	70	80	70	90	75	79.1	10.2
	no	85	90	70	95	91					86.2	9.7

$$P(\text{humidity} = 74 \mid \text{play} = \text{yes}) = \frac{1}{\sqrt{2\pi}(10.2)} e^{-\frac{(74-79.1)^2}{2(10.2)^2}} = 0.0344$$

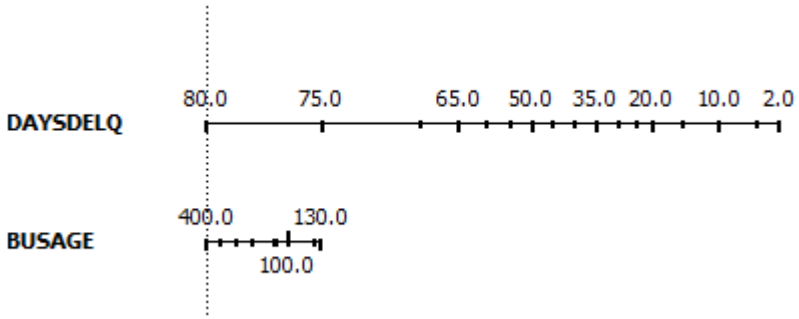
$$P(\text{humidity} = 74 \mid \text{play} = \text{no}) = \frac{1}{\sqrt{2\pi}(9.7)} e^{-\frac{(74-86.2)^2}{2(9.7)^2}} = 0.0187$$

Predictors Contribution

Kononenko's *information gain* as a sum of information contributed by each attribute can offer an explanation on how values of the predictors influence the class probability.

$$log_2 P(c|x) - log_2 P(c)$$

The contribution of predictors can also be visualized by plotting *nomograms*. Nomogram plots log odds ratios for each value of each predictor. Lengths of the lines correspond to spans of odds ratios, suggesting importance of the related predictor. It also shows impacts of individual values of the predictor.



Multiple Regression Analysis

The regression functions are used to determine the relationship between the dependent variable (target field) and one or more independent variables. The dependent variable is the one whose values you want to predict, whereas the independent variables are the variables that you base your prediction on.

A **RegressionModel** defines three types of regression models: linear, polynomial, and logistic regression. The **modelType** attribute indicates the type of regression used.

Linear and stepwise-polynomial regression are designed for numeric dependent variables having a continuous spectrum of values. These models should contain exactly one regression table. The attributes **normalizationMethod** and **targetCategory** are not used in that case.

Logistic regression is designed for categorical dependent variables. These models should contain exactly one regression table for each **targetCategory**. The **normalizationMethod** describes whether/how the prediction is converted into a probability.

For **linear** and **stepwise regression**, the regression formula is:

*Dependent variable = intercept + Sum$_i$ (coefficient$_i$ * independent variable$_i$) + error*

For **logistic regression** the formula is:

*y = intercept + Sum$_i$ (coefficient$_i$ * independent variable$_i$)*
p = 1/(1+exp(-y))

p is the predicted value. In many cases p can be interpreted as the confidence or the probability of an individual belonging to the category of interest, as defined by targetCategory. There can be multiple regression equations. With n classes/categories there are n equations of the form
*y_j = intercept_j + Sum$_i$ (coefficient_j$_i$ * independent variable$_i$)*

A confidence value for category j can be computed by the softmax function
p_j = exp(y_j) / (sum[i in 1..n](exp(y_i)))
Another method, called simplemax, uses a simple quotient
p_j = y_j / (sum[i in 1..n](y_i))
These confidence values are similar to statistical probabilities but they only mimic probabilities by post-processing the values y_i.

```
<!ELEMENT RegressionModel (Extension*, MiningSchema?, ModelStats?,
                           RegressionTable+, Extension*) >
<!ATTLIST RegressionModel
  modelName          CDATA                    #IMPLIED
  functionName       CDATA                    #REQUIRED
  algorithmName      CDATA                    #IMPLIED
  modelType          (linearRegression |
                      stepwisePolynomialRegression |
                      logisticRegression)      #REQUIRED
  targetFieldName    %FIELD-NAME;             #REQUIRED
  normalizationMethod  (none | simplemax | softmax)   "none"
>

<!ELEMENT RegressionTable (NumericPredictor*), (CategoricalPredictor*))>
<!ATTLIST RegressionTable
   intercept          %REAL-NUMBER;      #REQUIRED
   targetCategory     CDATA              #IMPLIED
>

<!ELEMENT NumericPredictor EMPTY>
<!ATTLIST NumericPredictor
   name               %FIELD-NAME;       #REQUIRED
   exponent           %INT-NUMBER;       #REQUIRED
   coefficient        %REAL-NUMBER;      #REQUIRED
   mean               %REAL-NUMBER;      #IMPLIED
>
<!ELEMENT CategoricalPredictor EMPTY>
<!ATTLIST CategoricalPredictor
   name               %FIELD-NAME;       #REQUIRED
   value              CDATA              #REQUIRED
   coefficient        %REAL-NUMBER;      #REQUIRED
>
```

RegressionModel: The root element of an XML regression model. Each instance of a regression model must start with this element.

modelName: This is a unique identifier specifying the name of the regression model.

functionName: Can be **regression** or **classification**.

algorithmName: Can be any string describing the algorithm that was used while creating the model.

modelType: Specifies the type of a regression model. This information is used to select the appropriate mathematical formulas during the scoring phase. The supported regression algorithms are listed.

targetFieldName: The name of the target field (also called response variable).

RegressionTable: A table that lists the values of all predictors or independent variables. If the model is used to predict a numerical field, then there is only one RegressionTable and the attribute targetCategory may be missing. If the model is used to predict a categorical field, then there are two or more RegressionTables and each one must have the attribute targetCategory defined with a unique value.

NumericPredictor: Defines a numeric independent variable. The list of valid attributes comprises the name of the variable, the exponent to be used, and the coefficient by which the valuesof this variable must be multiplied. If the independent variable contains missing values, the mean attribute is used to replace the missing values with the mean value.

CategoricalPredictor : Defines a categorical independent variable. The list of attributes comprises the name of the variable, the **value** attribute, and the coefficient by which the values of this variable must be multiplied. To do a regression analysis with categorical values, some means must be applied to enable calculations. If the specified value of an independent value occurs, the term **variable_name(value)** is replaced with 1. Thus the coefficient is multiplied by 1. If the value does not occur, the term **variable_name(value)** is replaced with 0 so that the product **coefficient** **variable_name(value)** yields 0. Consequently, the product is ignored in the ongoing analysis. If the input value is missing then variable_name(v) yields 0 for any 'v'.

Example:

The following regression formula is used to predict the number of insurance claims:

*number_of_claims = 132.37 + 7.1*age + 0.01*salary + 41.1*car_location('carpark') + 325.03*car_location('street')*

If the value *carpark* was specified for *car location* in a particular record, you would get the following formula:

*number of claims = 132.37 + 7.1 age + 0.01 salary + 41.1 * 1 + 325.03 * 0*

Linear Regression Sample

This is a linear regression equation predicting a number of insurance claims on prior knowledge of the values of the independent variables age, salary and car location. Car location is the only categorical variable. Its value attribute can take on two possible values, carpark and street.

number of claims = 132.37 + 7.1 age + 0.01 salary + 41.1 car location(carpark) + 325.03 car location(street)

The corresponding XML model is:

```
<RegressionModel
  functionName="regression"
  modelName="Sample for linear regression"
  modelType="linearRegression"
  targetFieldName="number of claims">

  <RegressionTable intercept="132.37">
    <NumericPredictor name="age"
                exponent="1" coefficient="7.1"/>
    <NumericPredictor name="salary"
                exponent="1" coefficient="0.01"/>
    <CategoricalPredictor name="car location"
                value="carpark" coefficient="41.1"/>
    <CategoricalPredictor name="car location"
                value="street" coefficient="325.03"/>
   </RegressionTable>

</RegressionModel>
```

Stepwise Polynomial Regression Sample

This is a stepwise polynomial regression equation predicting a number of insurance claims on prior knowledge of the values of the independent variables salary and car location. Car location is a categorical variable. Its value attribute can take on two possible values, carpark and street.

*number of claims = 3216.38 - 0.08 salary + 9.54E-7 salary**2 - 2.67E-12 salary**3 + 93.78 car location(carpark) + 288.75 car location(street)*

```
<RegressionModel
  functionName="regression"
  modelName="Sample for stepwise polynomial regression"
  modelType="stepwisePolynomialRegression"
  targetFieldName="number of claims">

  <RegressionTable intercept="3216.38">
    <NumericPredictor name="salary"
                exponent="1" coefficient="-0.08"/>
    <NumericPredictor name="salary"
                exponent="2" coefficient="9.54E-7"/>
    <NumericPredictor name="salary"
                exponent="3" coefficient="-2.67E-12"/>
    <CategoricalPredictor name="car location"
                value="carpark" coefficient="93.78"/>
    <CategoricalPredictor name="car location"
```

```
                value="street" coefficient="288.75"/>
    </RegressionTable>

</RegressionModel>
```

Logistic Regression Sample:

*y_clerical = 46.418 -0.132*age +7.867E-02*work -20.525*sex('0') +0*sex('1') -19.054*minority('0') +0*minority('1')*

*y_professional = 51.169 -0.302*age +.155*work -21.389*sex('0') +0*sex('1') -18.443*minority('0') + 0*minority('1')*

*y_trainee = 25.478 -.154*age +.266*work -2.639*sex('0') +0*sex('1') -19.821*minority('0') +0*minority('1')*

The model below defines no normalization, so p_j = y_j.

Note that the terms such as 0*minority('1') are superfluous but it's valid to use the same field with different indicator values such as '0' and '1'. Though, a RegresstionTable must not have multiple numeric predictors with the same name and it must not have multiple categorical predictors with the same pair of name and value.

The corresponding XML model is:

```
<RegressionModel
  functionName="classification"
  modelName="Sample for logistic regression"
  modelType="logisticRegression"
  normalizationMethod="none"
  targetFieldName="jobcat">

  <RegressionTable intercept="46.418"
      targetCategory="clerical">
    <NumericPredictor name="age" exponent="1"
                  coefficient="-0.132"/>
    <NumericPredictor name="work" exponent="1"
                  coefficient="7.867E-02"/>

    <CategoricalPredictor name="sex" value="0"
                  coefficient="-20.525"/>
    <CategoricalPredictor name="sex" value="1"
                  coefficient="0"/>
    <CategoricalPredictor name="minority" value="0"
                  coefficient="-19.054"/>
    <CategoricalPredictor name="minority" value="1"
                  coefficient="0"/>
  </RegressionTable>

  <RegressionTable intercept="51.169"
      targetCategory="professional">
    <NumericPredictor name="age" exponent="1"
```

```
                    coefficient="-0.302"/>
    <NumericPredictor name="work" exponent="1"
                    coefficient=".155"/>

    <CategoricalPredictor name="sex" value="0"
                    coefficient="-21.389"/>
    <CategoricalPredictor name="sex" value="1"
                    coefficient="0"/>
    <CategoricalPredictor name="minority" value="0"
                    coefficient="-18.443"/>
    <CategoricalPredictor name="minority" value="1"
                    coefficient="0"/>
  </RegressionTable>

  <RegressionTable intercept="25.478"
        targetCategory="trainee">
    <NumericPredictor name="age" exponent="1"
                    coefficient="-.154"/>
    <NumericPredictor name="work" exponent="1"
                    coefficient=".266"/>

    <CategoricalPredictor name="sex" value="0"
                    coefficient="-2.639"/>
    <CategoricalPredictor name="sex" value="1"
                    coefficient="0"/>
    <CategoricalPredictor name="minority" value="0"
                    coefficient="-19.821"/>
    <CategoricalPredictor name="minority" value="1"
                    coefficient="0"/>
  </RegressionTable>

</RegressionModel>
```

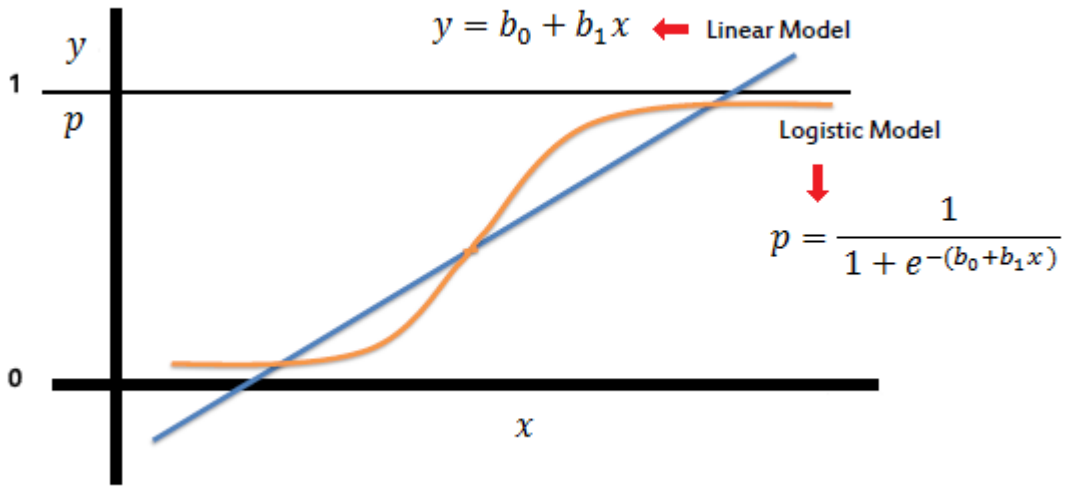
Logistic Regression

Logistic Regression

Logistic regression predicts the probability of an outcome that can only have two values (i.e. a dichotomy). The prediction is based on the use of one or several predictors (numerical and categorical). A linear regression is not appropriate for predicting the value of a binary variable for two reasons:

- A linear regression will predict values outside the acceptable range (e.g. predicting probabilities outside the range 0 to 1)
- Since the dichotomous experiments can only have one of two possible values for each experiment, the residuals will not be normally distributed about the predicted line.

On the other hand, a logistic regression produces a logistic curve, which is limited to values between 0 and 1. Logistic regression is similar to a linear regression, but the curve is constructed using the natural logarithm of the "odds" of the target variable, rather than the probability. Moreover, the predictors do not have to be normally distributed or have equal variance in each group.



In the logistic regression the constant (b_0) moves the curve left and right and the slope (b_1) defines the steepness of the curve. By simple transformation, the logistic regression equation can be written in terms of an odds ratio.

$$\frac{p}{1-p} = \exp(b_0 + b_1 x)$$

Finally, taking the natural log of both sides, we can write the equation in terms of log-odds (logit) which is a linear function of the predictors. The coefficient (b_1) is the amount the logit (log-odds) changes with a one unit change in x.

$$ln\left(\frac{p}{1-p}\right) = b_0 + b_1 x$$

As mentioned before, logistic regression can handle any number of numerical and/or categorical variables.

$$p = \frac{1}{1 + e^{-(b_0 + b_1 x_1 + b_2 x_2 + \cdots + b_p x_p)}}$$

There are several analogies between linear regression and logistic regression. Just as ordinary least square regression is the method used to estimate coefficients for the best fit line in linear regression, logistic regression uses **maximum likelihood estimation** (MLE) to obtain the model coefficients that relate predictors to the target. After this initial function is estimated, the process is repeated until LL (Log Likelihood) does not change significantly.

$$\boldsymbol{\beta^1 = \beta^0 + [X^T W X]^{-1} . X^T (y - \mu)}$$

$\boldsymbol{\beta}$ is a vector of the logistic regression coefficients.

$\boldsymbol{W}$ is a square matrix of order N with elements $n_i \pi_i (1 - \pi_i)$ on the diagonal and zeros everywhere else.

$\boldsymbol{\mu}$ is a vector of length N with elements $\mu_i = n_i \pi_i$.

A **pseudo R^2** value is also available to indicate the adequacy of the regression model. **Likelihood ratio test** is a test of the significance of the difference between the likelihood ratio for the baseline model minus the likelihood ratio for a reduced model. This difference is called "model chi-square". **Wald test** is used to test the statistical significance of each coefficient (*b*) in the model (i.e., predictors contribution).

Pseudo R^2

There are several measures intended to mimic the R^2 analysis to evaluate the goodness-of-fit of logistic models, but they cannot be interpreted as one would interpret an R^2 and different pseudo R^2 can arrive at very different values. Here we discuss three pseudo R^2measures.

Pseudo R^2	Equation	Description
Efron's	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - p_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$	*'p'* is the logistic model predicted probability. The model residuals are squared, summed, and divided by the total variability in the dependent variable.
McFadden's	$R^2 = 1 - \frac{LL_{full\ model}}{LL_{intercept}}$	The ratio of the log-likelihoods suggests the level of improvement over the intercept model offered by the full model.
Count	$R^2 = \frac{\#\ Corrects}{Total\ Count}$	The number of records correctly predicted, given a cutoff point of .5 divided by the total count of cases. This is equal to the accuracy of a classification model.

Likelihood Ratio Test

The likelihood ratio test provides the means for comparing the likelihood of the data under one model (e.g., full model) against the likelihood of the data under another, more restricted model (e.g., intercept model).

$$LL = \sum_{i=1}^{n} y_i ln(p_i) + (1 - y_i) ln(1 - p_i)$$

where '*p'* is the logistic model predicted probability. The next step is to calculate the difference between these two log-likelihoods.

$$2(LL_1 - LL_2)$$

The difference between two likelihoods is multiplied by a factor of 2 in order to be assessed for statistical significance using standard significance levels (Chi^2 test). The degrees of freedom for the test will equal the difference in the number of parameters being estimated under the models (e.g., full and intercept).

Wald test

A Wald test is used to evaluate the statistical significance of each coefficient (*b*) in the model.

$$W_j = \frac{b_j}{SE_{b_j}}$$

where *W* is the Wald's statistic with a normal distribution (like Z-test), *b* is the coefficient and *SE* is its standard error. The *W* value is then squared, yielding a Wald statistic with a chi-square distribution.

$$SE_j = sqrt(diag(\boldsymbol{H^{-1}})_j)$$

$$\boldsymbol{H} = [\boldsymbol{X^T W X}]$$

Predictors Contributions

The Wald test is usually used to assess the significance of prediction of each predictor. Another indicator of contribution of a predictor is *exp*(*b*) or **odds-ratio** of coefficient which is the amount the logit (log-odds) changes, with a one unit change in the predictor (*x*).

k-Nearest Neighbour Classification

K Nearest Neighbors - Classification

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970's as a non-parametric technique.

Algorithm

A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor.

Distance functions

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$

Minkowski $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

Hamming Distance

$$D_H = \sum_{i=1}^{k}|x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

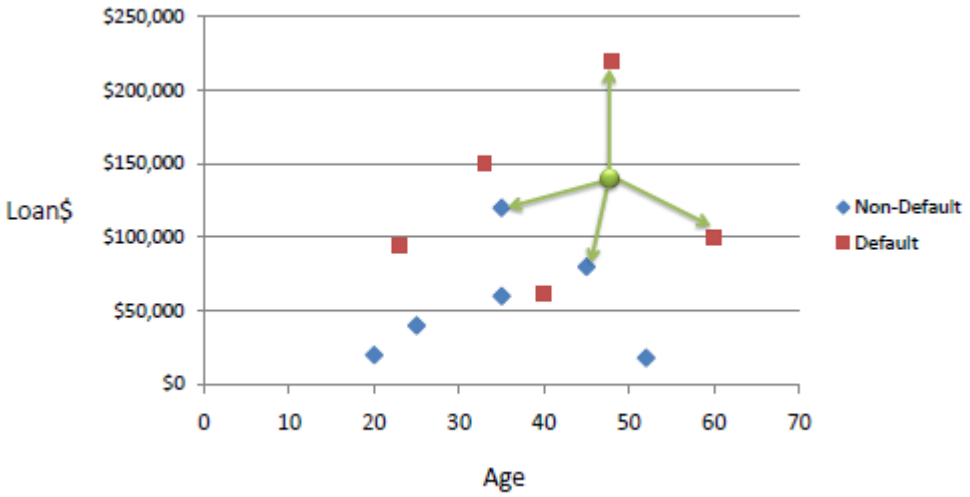
$$x \neq y \Rightarrow D = 1$$

X	Y	Distance
Male	Male	0
Male	Female	1

Choosing the optimal value for K is best done by first inspecting the data. In general, a large K value is more precise as it reduces the overall noise but there is no guarantee. Cross-validation is another way to retrospectively determine a good K value by using an independent dataset to validate the K value. Historically, the optimal K for most datasets has been between 3-10. That produces much better results than 1NN.

Example:

Consider the following data concerning credit default. Age and Loan are two numerical variables (predictors) and Default is the target.



We can now use the training set to classify an unknown case (Age=48 and Loan=$142,000) using Euclidean distance. If K=1 then the nearest neighbor is the last case in the training set with Default=Y.

D = Sqrt[(48-33)^2 + (142000-150000)^2] = 8000.01 >> Default=Y

Age	Loan	Default	Distance	
25	$40,000	N	102000	
35	$60,000	N	82000	
45	$80,000	N	62000	
20	$20,000	N	122000	
35	$120,000	N	22000	2
52	$18,000	N	124000	
23	$95,000	Y	47000	
40	$62,000	Y	80000	
60	$100,000	Y	42000	3
48	$220,000	Y	78000	
33	$150,000	Y	8000	1
48	**$142,000**	**?**		

Euclidean Distance $D = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$

With K=3, there are two Default=Y and one Default=N out of three closest neighbors. The prediction for the unknown case is again Default=Y.

Standardized Distance

One major drawback in calculating distance measures directly from the training set is in the case where variables have different measurement scales or there is a mixture of numerical and categorical variables. For example, if one variable is based on annual income in dollars, and the other is based on age in years then income will have a much higher influence on the distance calculated. One solution is to standardize the training set as shown below.

Age	Loan	Default	Distance
0.125	0.11	N	0.7652
0.375	0.21	N	0.5200
0.625	0.31	N	0.3160
0	0.01	N	0.9245
0.375	0.50	N	0.3428
0.8	0.00	N	0.6220
0.075	0.38	Y	0.6669
0.5	0.22	Y	0.4437
1	0.41	Y	0.3650
0.7	1.00	Y	0.3861
0.325	0.65	Y	0.3771
0.7	0.61	?	

Standardized Variable

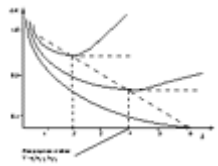
$$X_s = \frac{X - Min}{Max - Min}$$

Using the standardized distance on the same training set, the unknown case returned a different neighbor which is not a good sign of robustness.

GMDH Algorithm

Group Method of Data Handling* was applied in a great variety of areas for data mining and knowledge discovery, forecasting and systems modeling, optimization and pattern recognition. Inductive GMDH algorithms give possibility to find automatically interrelations in data, to select an optimal structure of model or network and to increase the accuracy of existing algorithms.

This original self-organizing approach is substantially different from deductive methods used commonly for modeling. It has inductive nature - it finds the best solution by sorting-out of possible variants.



By sorting of different solutions GMDH networks aims to minimize the influence of the author on the results of modeling. Computer itself finds the structure of the optimal model or laws that act in a system.

Group Method of Data Handling is a set of several algorithms for different problems solution. It consists of parametric, clusterization, analogues complexing, rebinarization and probability algorithms. This inductive approach is based on sorting-out of gradually complicated models and selection of the optimal solution by minimum of external criterion characteristic. Not only polynomials but also non-linear, probabilistic functions or clusterizations are used as basic models.

GMDH approach can be useful because:

- Optimal complexity of the model structure is found, adequate to the level of noise in data sample. For real problems, with noised or short data, a simplified optimal models are more accurate.
- The number of layers and neurons in hidden layers, model structure and other optimal neuran networks parameters are determined automatically.
- It guarantees that the most accurate or unbiased models will be found - method doesn't miss the best solution during sorting of all variants (in the given class of functions).
- As input variables are used any non-linear functions or features, which can influence the output variable.
- It automatically finds interpretable relationships in data and selects effective input variables.
- GMDH sorting algorithms are rather simple for programming.
- Twice-multilayered neural nets can be used to increase the accuracy of another modelling algorithms.
- Method get information directly from data sample and minimizes influence of apriori author assumptions about results of modeling.

- Approach gives possibility to find unbiased physical model of object (law or clusterization) - one and the same for future samples.

It was implemented in the many commercial software tools.

Computing and Genetic Algorithms

Genetic Algorithm (GA) is a search-based optimization technique based on the principles of **Genetics and Natural Selection**. It is frequently used to find optimal or near-optimal solutions to difficult problems which otherwise would take a lifetime to solve. It is frequently used to solve optimization problems, in research, and in machine learning.

Introduction to Optimization

Optimization is the process of **making something better**. In any process, we have a set of inputs and a set of outputs as shown in the following figure.

Optimization refers to finding the values of inputs in such a way that we get the “best” output values. The definition of “best” varies from problem to problem, but in mathematical terms, it refers to maximizing or minimizing one or more objective functions, by varying the input parameters.

The set of all possible solutions or values which the inputs can take make up the search space. In this search space, lies a point or a set of points which gives the optimal solution. The aim of optimization is to find that point or set of points in the search space.

What are Genetic Algorithms?

Nature has always been a great source of inspiration to all mankind. Genetic Algorithms (GAs) are search based algorithms based on the concepts of natural selection and genetics. GAs are a subset of a much larger branch of computation known as **Evolutionary Computation**.

GAs were developed by John Holland and his students and colleagues at the University of Michigan, most notably David E. Goldberg and has since been tried on various optimization problems with a high degree of success.

In GAs, we have a **pool or a population of possible solutions** to the given problem. These solutions then undergo recombination and mutation (like in natural genetics), producing new children, and the process is repeated over various generations. Each individual (or candidate solution) is assigned a fitness value (based on its objective function value) and the fitter individuals are given a higher chance to mate and yield more "fitter" individuals. This is in line with the Darwinian Theory of "Survival of the Fittest".

In this way we keep "evolving" better individuals or solutions over generations, till we reach a stopping criterion.

Genetic Algorithms are sufficiently randomized in nature, but they perform much better than random local search (in which we just try various random solutions, keeping track of the best so far), as they exploit historical information as well.

Advantages of GAs

GAs have various advantages which have made them immensely popular. These include −

- Does not require any derivative information (which may not be available for many real-world problems).
- Is faster and more efficient as compared to the traditional methods.
- Has very good parallel capabilities.
- Optimizes both continuous and discrete functions and also multi-objective problems.
- Provides a list of "good" solutions and not just a single solution.
- Always gets an answer to the problem, which gets better over the time.
- Useful when the search space is very large and there are a large number of parameters involved.

Limitations of GAs

Like any technique, GAs also suffer from a few limitations. These include −

- GAs are not suited for all problems, especially problems which are simple and for which derivative information is available.
- Fitness value is calculated repeatedly which might be computationally expensive for some problems.
- Being stochastic, there are no guarantees on the optimality or the quality of the solution.
- If not implemented properly, the GA may not converge to the optimal solution.

GA – Motivation

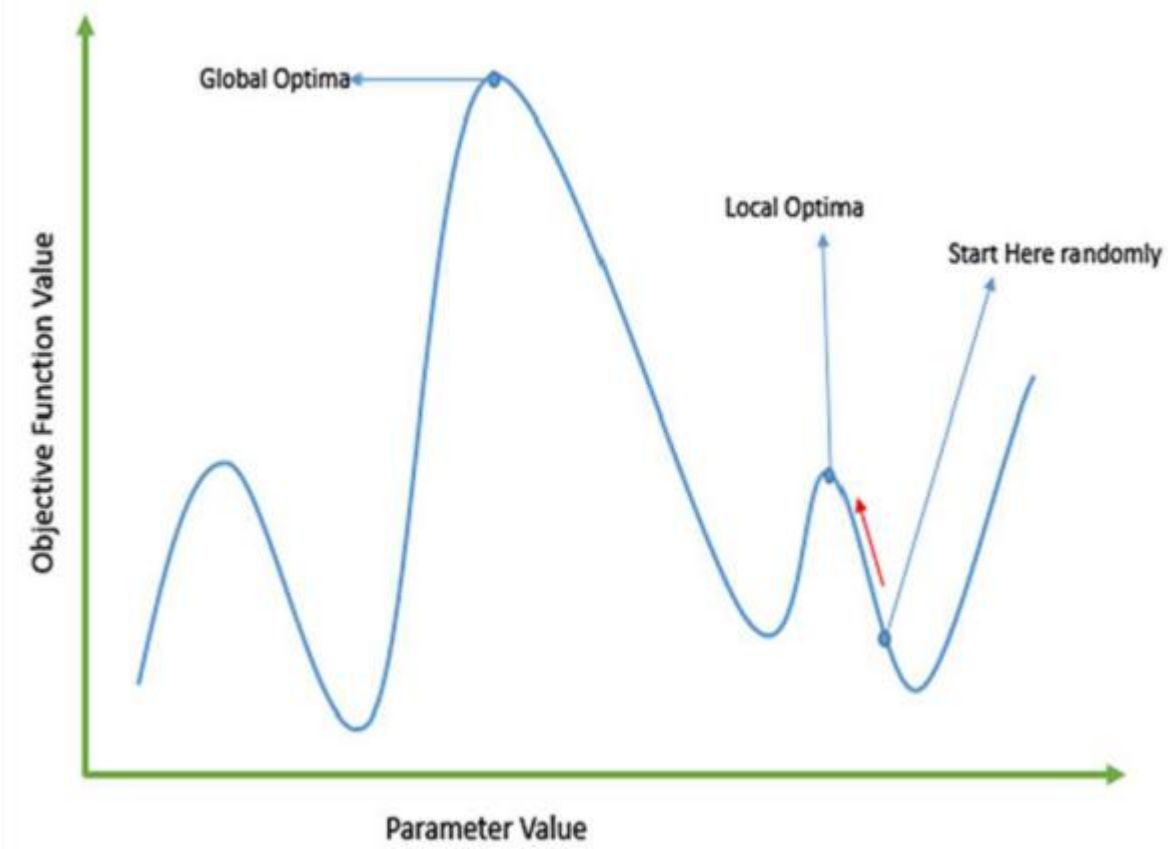
Genetic Algorithms have the ability to deliver a "good-enough" solution "fast-enough". This makes genetic algorithms attractive for use in solving optimization problems. The reasons why GAs are needed are as follows −

Solving Difficult Problems

In computer science, there is a large set of problems, which are **NP-Hard**. What this essentially means is that, even the most powerful computing systems take a very long time (even years!) to solve that problem. In such a scenario, GAs prove to be an efficient tool to provide **usable near-optimal solutions** in a short amount of time.

Failure of Gradient Based Methods

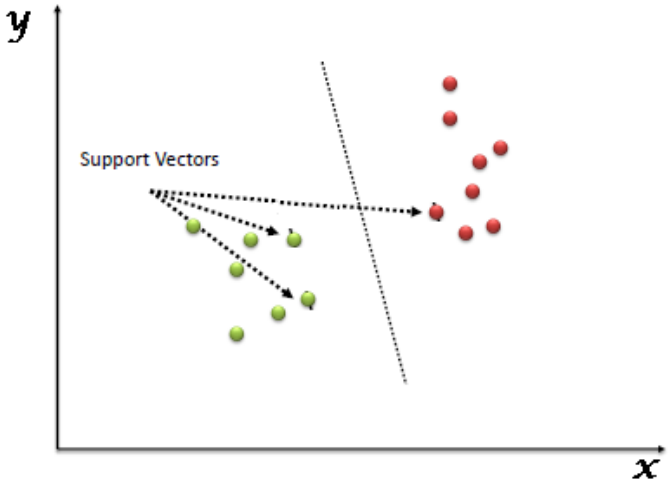
Traditional calculus based methods work by starting at a random point and by moving in the direction of the gradient, till we reach the top of the hill. This technique is efficient and works very well for single-peaked objective functions like the cost function in linear regression. But, in most real-world situations, we have a very complex problem called as landscapes, which are made of many peaks and many valleys, which causes such methods to fail, as they suffer from an inherent tendency of getting stuck at the local optima as shown in the following figure.



Support Vector Machines

What is Support Vector Machine?

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot).



Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

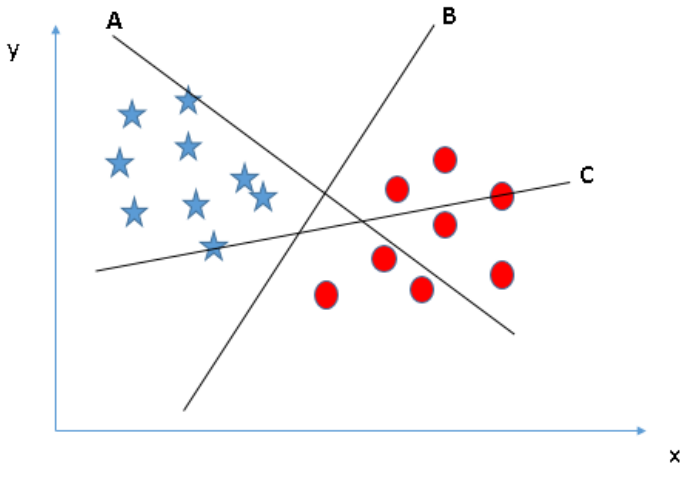
You can look at definition of support vectors and a few examples of its working here.

How does it work?

Above, we got accustomed to the process of segregating the two classes with a hyper-plane. Now the burning question is "How can we identify the right hyper-plane?". Don't worry, it's not as hard as you think!

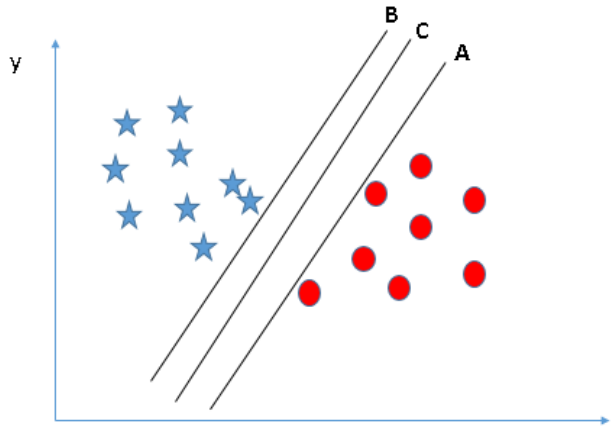
Let's understand:

- **Identify the right hyper-plane (Scenario-1):** Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

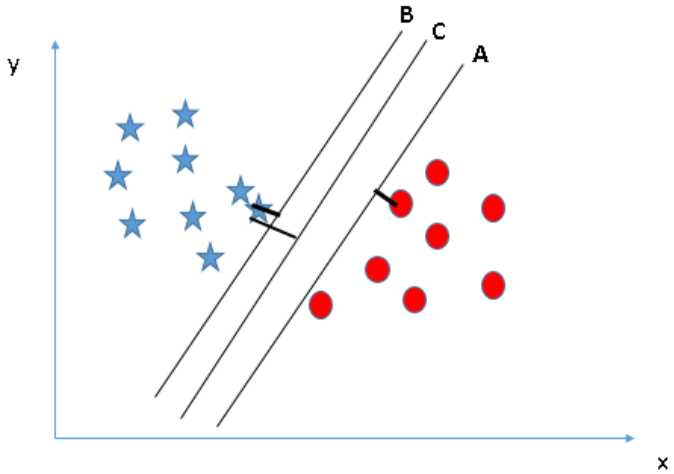


You need to remember a thumb rule to identify the right hyper-plane: "Select the hyper-plane which segregates the two classes better". In this scenario, hyper-plane "B" has excellently performed this job.

- **Identify the right hyper-plane (Scenario-2):** Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, How can we identify the right hyper-plane?

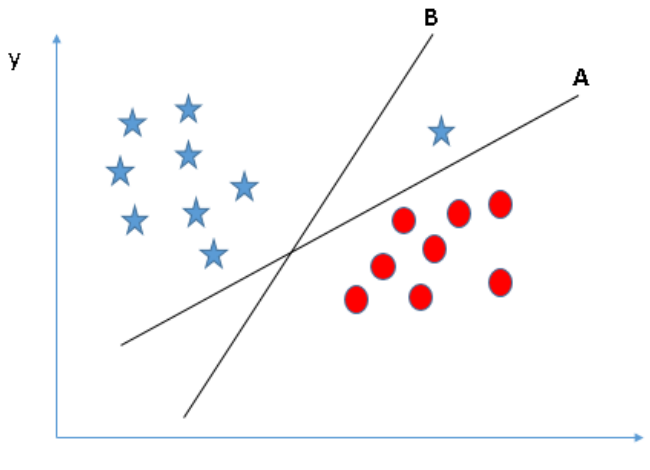


× Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**. Let's look at the below snapshot:



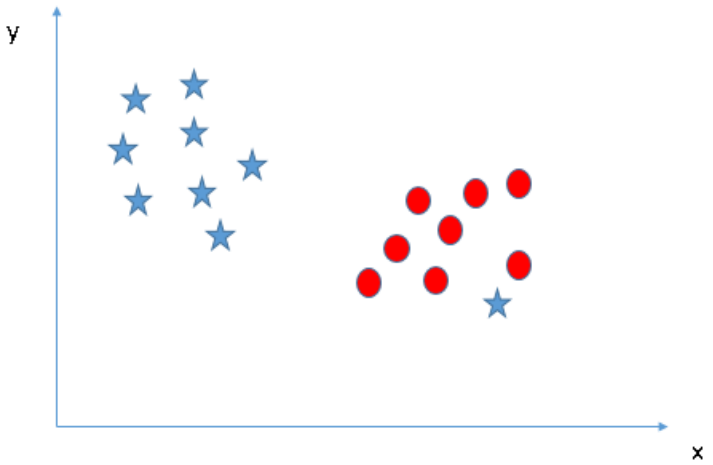
Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

- **Identify the right hyper-plane (Scenario-3):**Hint: Use the rules as discussed in previous section to identify the right hyper-plane



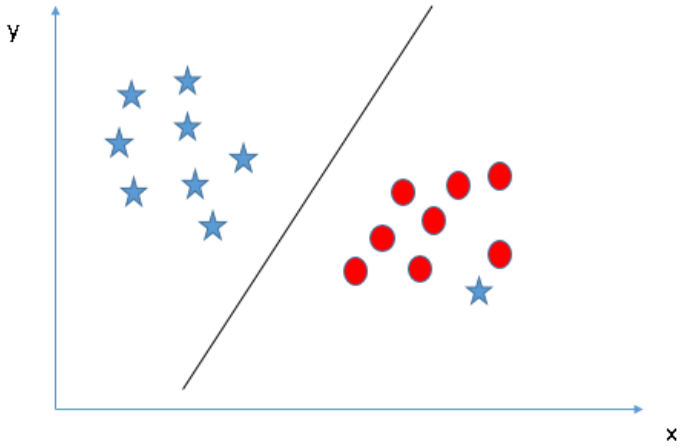
Some of you may have selected the hyper-plane **B** as it has higher margin compared to **A.** But, here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore, the right hyper-plane is **A.**

- **Can we classify two classes (Scenario-4)?:** Below, I am unable to segregate the two classes using a straight line, as one of star lies in the territory of other(circle) class as an outlier.

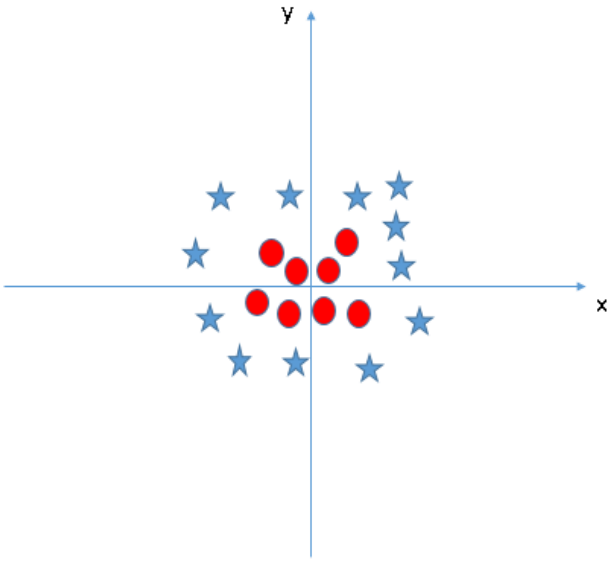


As I have already mentioned, one star at other end is like an outlier for star class. SVM has a feature to ignore outliers and find the

hyper-plane that has maximum margin. Hence, we can say, SVM is robust to outliers.

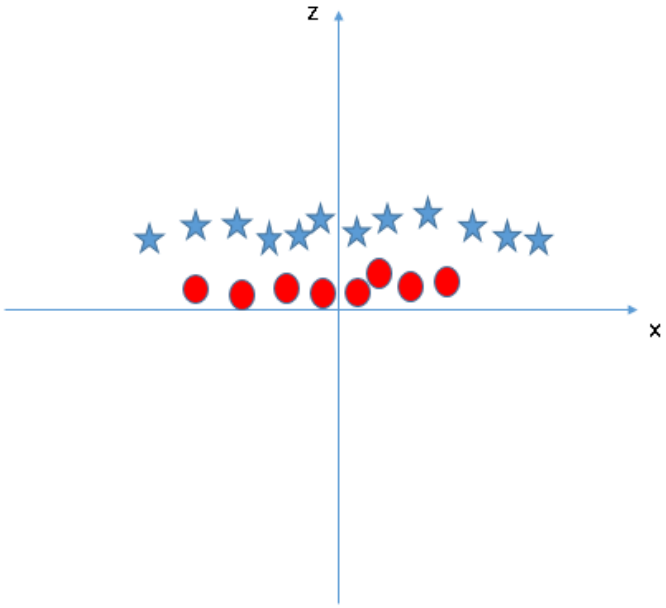


- **Find the hyper-plane to segregate to classes (Scenario-5):** In the scenario below, we can't have linear hyper-plane between the two classes, so how does SVM classify these two classes? Till now, we have only looked at the linear hyper-plane.



SVM can solve this problem. Easily!

It solves this problem by introducing additional feature. Here, we will add a new feature z=x^2+y^2. Now, let's plot the data points on axis x and z:

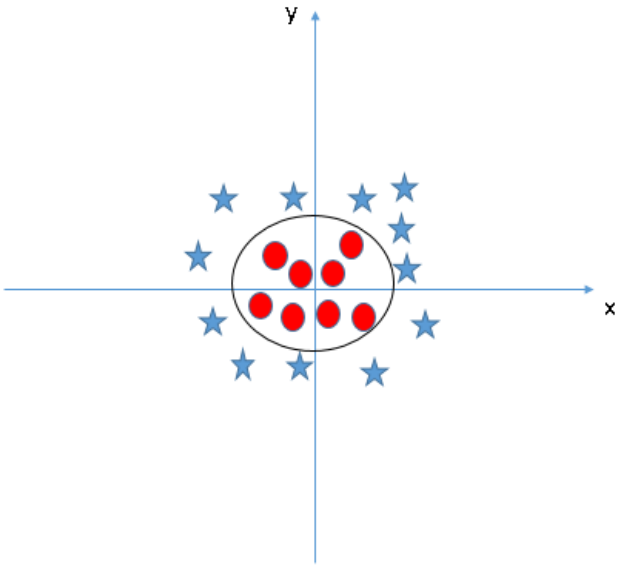


In above plot, points to consider are:

- All values for z would be positive always because z is the squared sum of both x and y
- In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.

In SVM, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, SVM has a technique called the **kernel trick**. These are functions which takes low dimensional input space and transform it to a higher dimensional space i.e. it converts not separable problem to separable problem, these functions are called kernels. It is mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then find out the process to separate the data based on the labels or outputs you've defined.

When we look at the hyper-plane in original input space it looks like a circle:



Now, let's look at the methods to apply SVM algorithm in a data science challenge.

How to implement SVM in Python and R?

In Python, scikit-learn is a widely used library for implementing machine learning algorithms, SVM is also available in scikit-learn library and follow the same structure (Import library, object creation, fitting model and prediction). Let's look at the below code:

```
#Import Library
from sklearn import svm
#Assumed you have, X (predictor) and Y (target) for training data set and x_test(predictor) of test_dataset
# Create SVM classification object
model = svm.svc(kernel='linear', c=1, gamma=1)
# there is various option associated with it, like changing kernel, gamma and C value. Will discuss more # about it in next section.Train the model using the training sets and check score
model.fit(X, y)
model.score(X, y)
#Predict Output
predicted= model.predict(x_test)
```

The e1071 package in R is used to create Support Vector Machines with ease. It has helper functions as well as code for the Naive Bayes Classifier. The creation of a support vector machine in R and Python follow similar approaches, let's take a look now at the following code:

```
#Import Library
require(e1071) #Contains the SVM
Train <- read.csv(file.choose())
Test <- read.csv(file.choose())
# there are various options associated with SVM training; like changing kernel, gamma and C value.

# create model
model <- svm(Target~Predictor1+Predictor2+Predictor3,data=Train,kernel='linear',gamma=0.2,cost=100)

#Predict Output
preds <- predict(model,Test)
table(preds)
```

How to tune Parameters of SVM?

Tuning parameters value for machine learning algorithms effectively improves the model performance. Let's look at the list of parameters available with SVM.

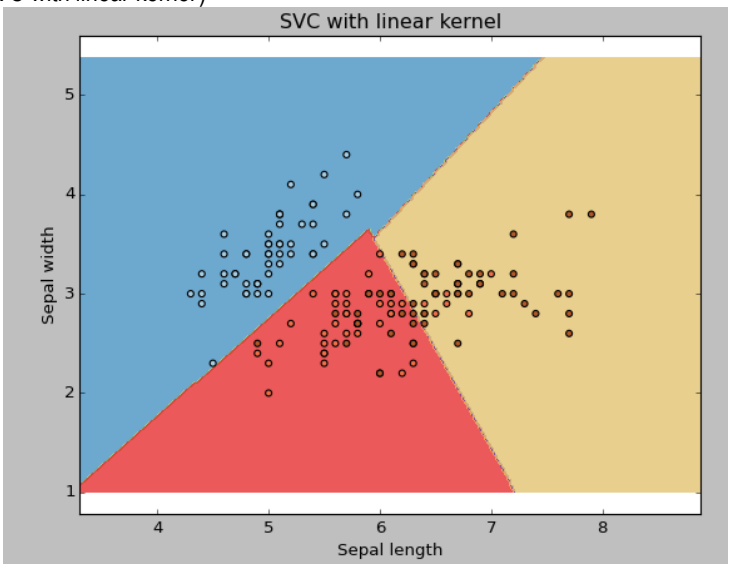
sklearn.svm.SVC(*C=1.0*, *kernel='rbf'*, *degree=3*, *gamma=0.0*, *coef0=0.0*, *shrinking=True*, *probability=False*,*tol=0.001*, *cache_size=200*, *class_weight=None*, *verbose=False*, *max_iter=-1*, *random_state=None*)

I am going to discuss about some important parameters having higher impact on model performance, "kernel", "gamma" and "C".

kernel: We have already discussed about it. Here, we have various options available with kernel like, "linear", "rbf","poly" and others (default value is "rbf"). Here "rbf" and "poly" are useful for non-linear hyper-plane. Let's look at the example, where we've used linear kernel on two feature of iris data set to classify their class.

Example: Have linear kernel

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import svm, datasets
# import some data to play with
iris = datasets.load_iris()
X = iris.data[:, :2] # we only take the first two features. We could
 # avoid this ugly slicing by using a two-dim dataset
y = iris.target
# we create an instance of SVM and fit out data. We do not scale our
# data since we want to plot the support vectors
C = 1.0 # SVM regularization parameter
svc = svm.SVC(kernel='linear', C=1,gamma=0).fit(X, y)
# create a mesh to plot in
x_min, x_max = X[:, 0].min() - 1, X[:, 0].max() + 1
y_min, y_max = X[:, 1].min() - 1, X[:, 1].max() + 1
h = (x_max / x_min)/100
xx, yy = np.meshgrid(np.arange(x_min, x_max, h),
 np.arange(y_min, y_max, h))
plt.subplot(1, 1, 1)
Z = svc.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)
plt.contourf(xx, yy, Z, cmap=plt.cm.Paired, alpha=0.8)
plt.scatter(X[:, 0], X[:, 1], c=y, cmap=plt.cm.Paired)
plt.xlabel('Sepal length')
plt.ylabel('Sepal width')
plt.xlim(xx.min(), xx.max())
plt.title('SVC with linear kernel')
```

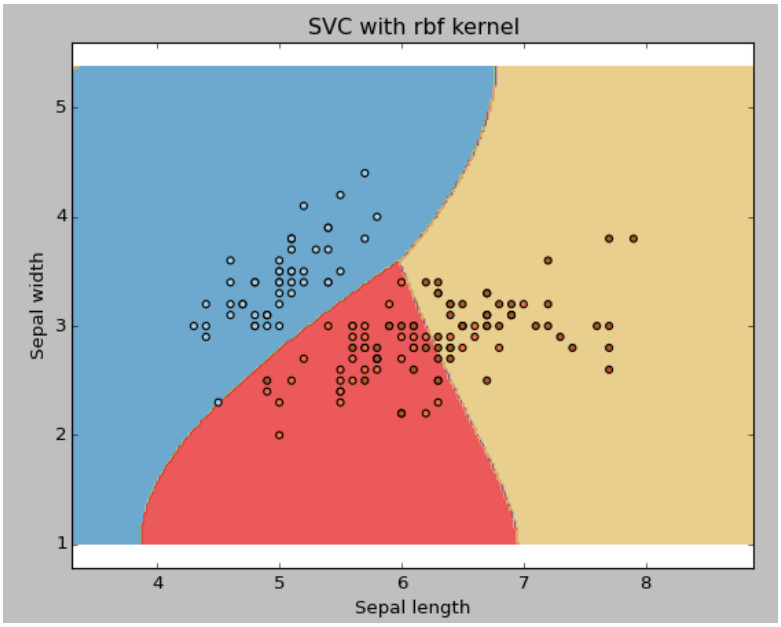



```
plt.show()
```

Example: Have rbf kernel

Change the kernel type to rbf in below line and look at the impact.

svc = svm.SVC(kernel='rbf', C=1,gamma=0).fit(X, y)



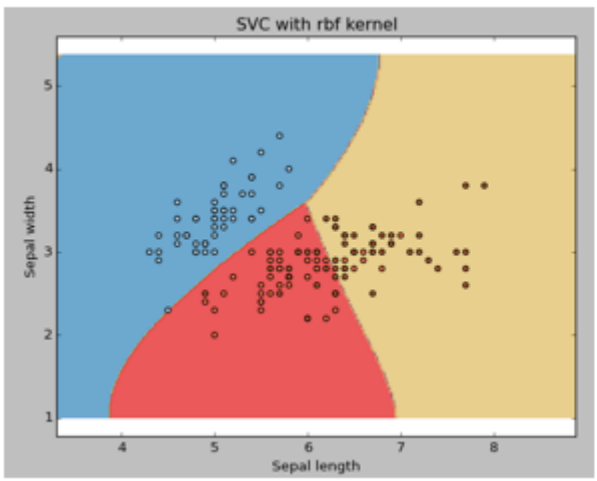
I would suggest you to go for linear kernel if you have large number of features (>1000) because it is more likely that the data is linearly separable in high dimensional space. Also, you can RBF but do not forget to cross validate for its parameters as to avoid over-fitting.

gamma: Kernel coefficient for 'rbf', 'poly' and 'sigmoid'. Higher the value of gamma, will try to exact fit the as per training data set i.e. generalization error and cause over-fitting problem.

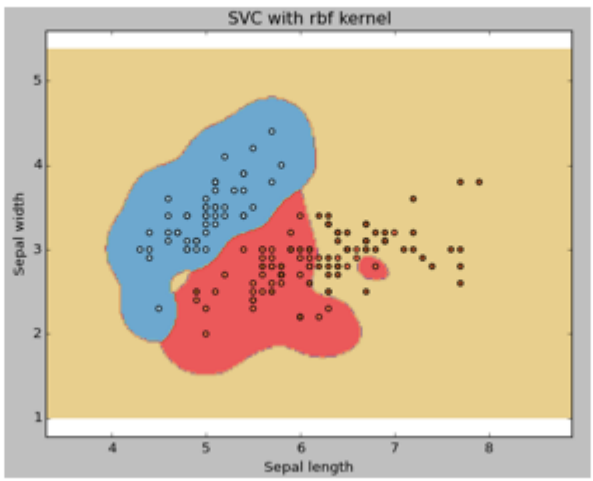
Example: Let's difference if we have gamma different gamma values like 0, 10 or 100.

svc = svm.SVC(kernel='rbf', C=1,gamma=0).fit(X, y)

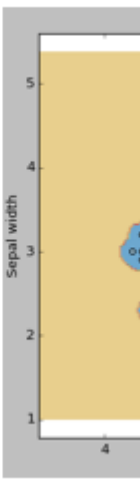
gamma =0



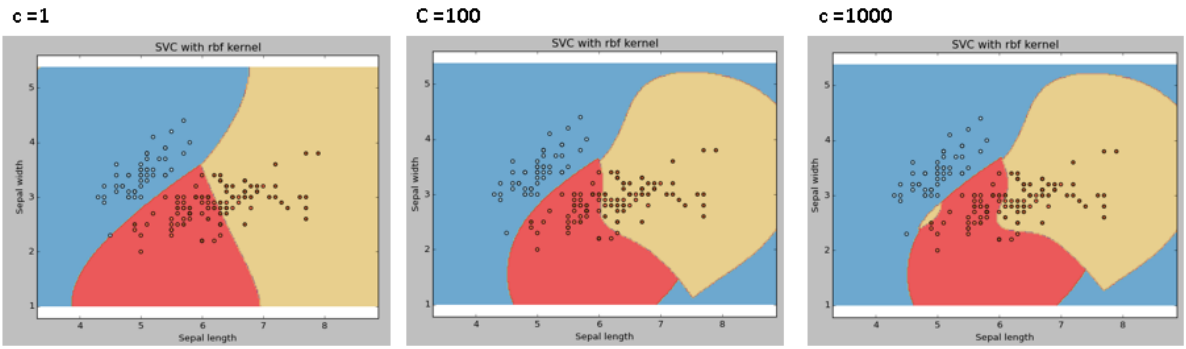
gamma =10



gamma



C: Penalty parameter C of the error term. It also controls the trade off between smooth decision boundary and classifying the training points correctly.



We should always look at the cross validation score to have effective combination of these parameters and avoid over-fitting.

In R, SVMs can be tuned in a similar fashion as they are in Python. Mentioned below are the respective parameters for e1071 package:

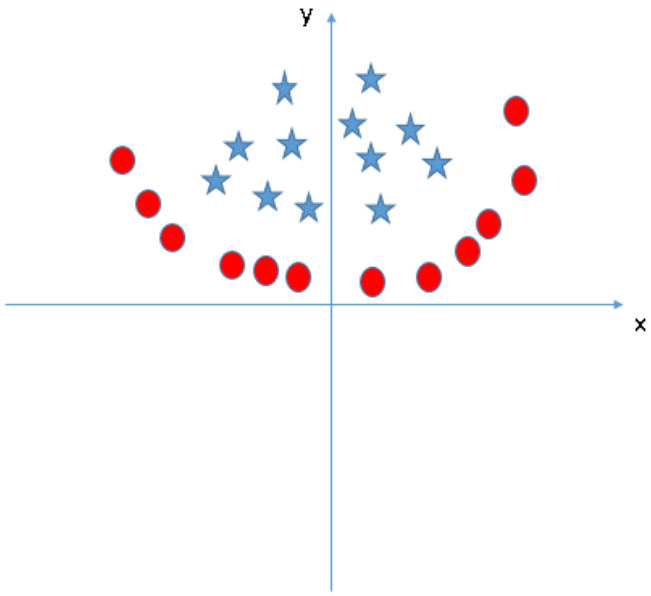
- The kernel parameter can be tuned to take “Linear”,”Poly”,”rbf” etc.
- The gamma value can be tuned by setting the “Gamma” parameter.
- The C value in Python is tuned by the “Cost” parameter in R.

Pros and Cons associated with SVM

- **Pros:**
  - It works really well with clear margin of separation
  - It is effective in high dimensional spaces.
  - It is effective in cases where number of dimensions is greater than the number of samples.
  - It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- **Cons:**
  - It doesn’t perform well, when we have large data set because the required training time is higher
  - It also doesn’t perform very well, when the data set has more noise i.e. target classes are overlapping
  - SVM doesn’t directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. It is related SVC method of Python scikit-learn library.

Practice Problem

Find right additional feature to have a hyper-plane for segregating the classes in below snapshot:



Cluster Analysis

Partitional Clusterings

Cluster is a group of objects that belongs to the same class. In other words, similar objects are grouped in one cluster and dissimilar objects are grouped in another cluster.

What is Clustering?

Clustering is the process of making a group of abstract objects into classes of similar objects.

Points to Remember

- A cluster of data objects can be treated as one group.
- While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups.
- The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups.

Applications of Cluster Analysis

- Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
- Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
- In the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations.

- Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
- Clustering also helps in classifying documents on the web for information discovery.
- Clustering is also used in outlier detection applications such as detection of credit card fraud.
- As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.

Requirements of Clustering in Data Mining

The following points throw light on why clustering is required in data mining −

- **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.

Clustering Methods

Clustering methods can be classified into the following categories −

- Partitioning Method
- Hierarchical Method
- Density-based Method
- Grid-Based Method
- Model-Based Method
- Constraint-based Method

Partitioning Method

Suppose we are given a database of ‘n’ objects and the partitioning method constructs ‘k’ partition of data. Each partition will represent a cluster and k ≤ n. It means that it will classify the data into k groups, which satisfy the following requirements −

- Each group contains at least one object.
- Each object must belong to exactly one group.

Points to remember −

- For a given number of partitions (say k), the partitioning method will create an initial partitioning.
- Then it uses the iterative relocation technique to improve the partitioning by moving objects from one group to other.

Hierarchical Methods

This method creates a hierarchical decomposition of the given set of data objects. We can classify hierarchical methods on the basis of how the hierarchical decomposition is formed. There are two approaches here −

- Agglomerative Approach
- Divisive Approach

Agglomerative Approach

This approach is also known as the bottom-up approach. In this, we start with each object forming a separate group. It keeps on merging the objects or groups that are close to one another. It keep on doing so until all of the groups are merged into one or until the termination condition holds.

Divisive Approach

This approach is also known as the top-down approach. In this, we start with all of the objects in the same cluster. In the continuous iteration, a cluster is split up into smaller clusters. It is down until each object in one cluster or the termination condition holds. This method is rigid, i.e., once a merging or splitting is done, it can never be undone.

Approaches to Improve Quality of Hierarchical Clustering

Here are the two approaches that are used to improve the quality of hierarchical clustering −

- Perform careful analysis of object linkages at each hierarchical partitioning.
- Integrate hierarchical agglomeration by first using a hierarchical agglomerative algorithm to group objects into micro-clusters, and then performing macro-clustering on the micro-clusters.

Density-based Method

This method is based on the notion of density. The basic idea is to continue growing the given cluster as long as the density in the neighborhood exceeds some threshold, i.e., for each data point within a given cluster, the radius of a given cluster has to contain at least a minimum number of points.

Grid-based Method

In this, the objects together form a grid. The object space is quantized into finite number of cells that form a grid structure.

Advantage

- The major advantage of this method is fast processing time.
- It is dependent only on the number of cells in each dimension in the quantized space.

Model-based methods

In this method, a model is hypothesized for each cluster to find the best fit of data for a given model. This method locates the clusters by clustering the density function. It reflects spatial distribution of the data points.

This method also provides a way to automatically determine the number of clusters based on standard statistics, taking outlier or noise into account. It therefore yields robust clustering methods.

Constraint-based Method

In this method, the clustering is performed by the incorporation of user or application-oriented constraints. A constraint refers to the user expectation or the properties of desired clustering results. Constraints provide us with an interactive way of communication with the clustering process. Constraints can be specified by the user or the application requirement.

k-medoids

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 datapoints 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 realisation 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 medoids
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.

Birch

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies)

- It is a scalable clustering method.
- Designed for very large data sets
- Only one scan of data is necessary
- It is based on the notation of CF (Clustering Feature) a CF Tree.
- CF tree is a height balanced tree that stores the clustering features for a hierarchical clustering.
- Cluster of data points is represented by a triple of numbers (N,LS,SS) Where

  N= Number of items in the sub cluster

  LS=Linear sum of the points

  SS=sum of the squared of the points

A CF Tree structure is given as below:

- Each non-leaf node has at most B entries.
- Each leaf node has at most L CF entries which satisfy threshold T, a maximum diameter of radius
- P(page size in bytes) is the maximum size of a node
- Compact: each leaf node is a subcluster, not a data point

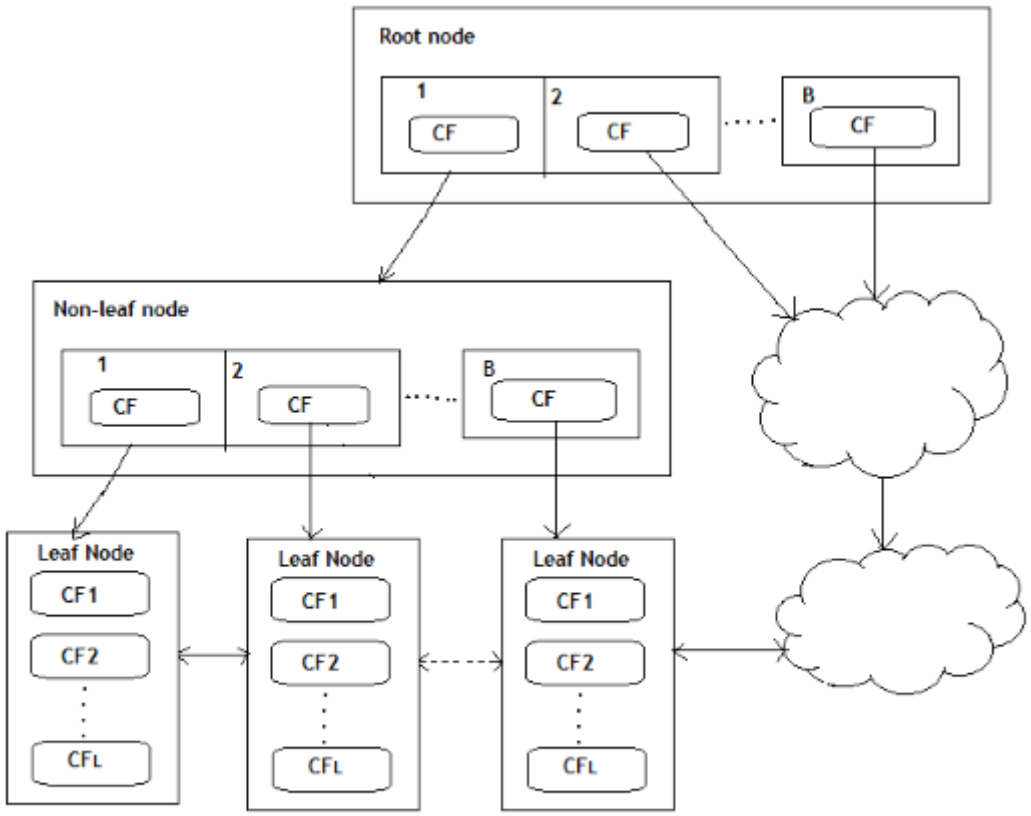


Fig: A CF tree structure

Basic Algorithm:

- Phase 1: Load data into memory

  Scan DB and load data into memory by building a CF tree. If memory is exhausted rebuild the tree from the leaf node.

- Phase 2: Condense data

  Resize the data set by building a smaller CF tree

  Remove more outliers

  Condensing is optional

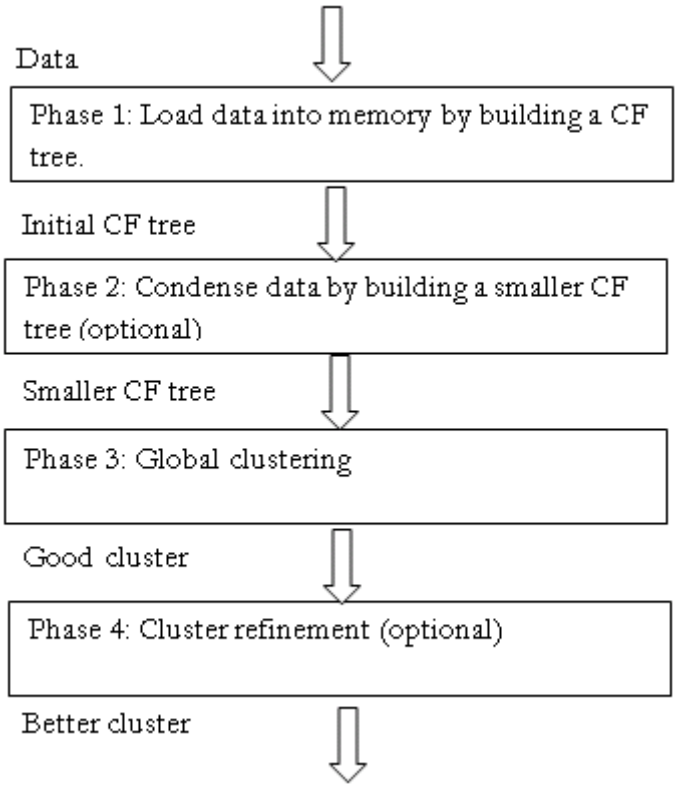
- Phase 3: Global clustering

  Use existing clustering algorithm (e.g. KMEANS, HC) on CF entries

- Phase 4: Cluster refining

  Refining is optional

Fixes the problem with CF trees where same valued data points may be assigned to different leaf entries.



Example:

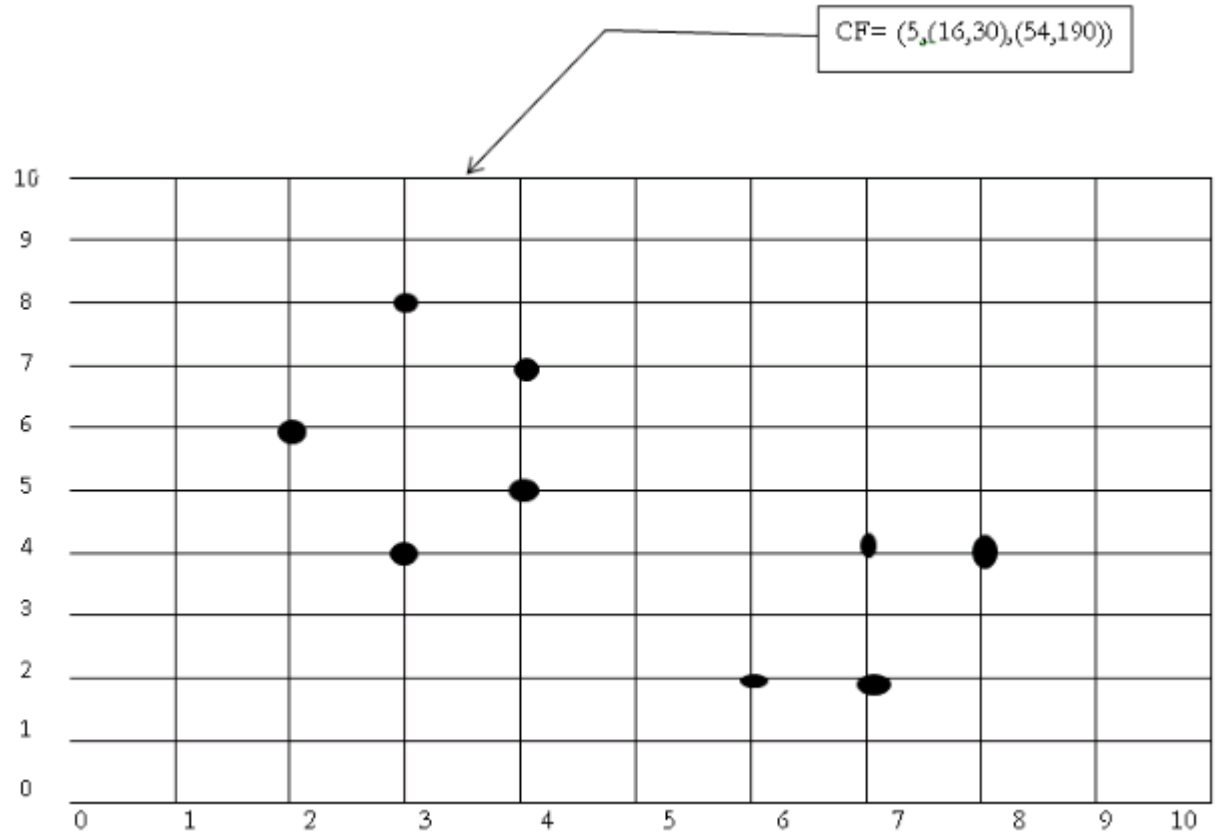
Clustering feature:

CF= (N, LS, SS)

N: number of data points

LS: $\sum_{Ni=1}=X_i$

$$SS:\sum_{Ni=1}=X_{2I}$$



(3,4) (2,6)(4,5)(4,7)(3,8)

N=5

NS= (16, 30) i.e. 3+2+4+4+3=16 and 4+6+5+7+8=30

SS=(54,190)=3_2+2_2+4_2+4_2+3_2=54 *and* 4_2+6_2+5_2+7_2+8_2=190

- Advantages: Finds a good clustering with a single scan and improves the quality with a few additional scans
- Disadvantages: Handles only numeric data
- Applications:

  Pixel classification in images

  Image compression

  Works with very large data sets

DBSCAN

DBSCAN: Density Based Clustering of Applications with noise

- DBSCAN is a density-based algorithm.
- DBSCAN requires two parameters: epsilon (Eps) and minimum points (MinPts).It starts with an arbitrary starting point that has not been visited .It then finds all the neighbour points within distance Eps of the starting point.
- If the number of neighbours is greater than or equal to MinPts, a cluster is formed. The starting point and its neighbours are added to this cluster and the starting point is marked as visited. The algorithm then repeats the evaluation process for all the neighbours recursively.
- If the number of neighbours is less than MinPts, the point is marked as noise.
- If a cluster is fully expanded (all points within reach are visited) then the algorithm proceeds to iterate through the remaining unvisited points in the dataset.
- Major features:

  Discover clusters of arbitrary shape

  Handle noise

  One scan

  Need density parameters

- Basic concept:

  For any cluster we have:

- A central point (p) i.e. core point
- A distance from the core point(Eps)
- Minimum number of points within the specified distance (MinPts)

DBSCAN Algorithm

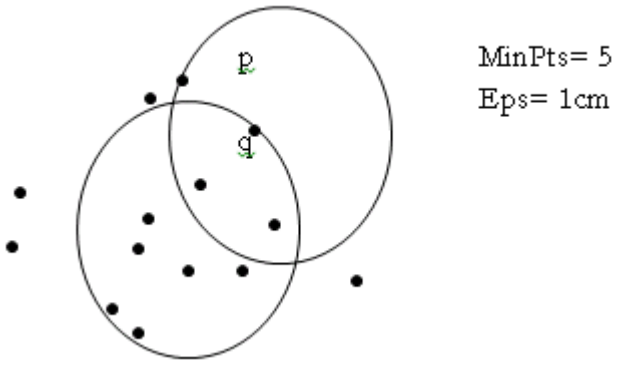
1. Create a graph whose nodes are the points to be clustered
2. -neighborhood of cεFor each core-point c create an edge from c to every point p in the
3. Set N to the nodes of the graph;
4. If N does not contain any core points terminate
5. Pick a core point c in N
6. Let X be the set of nodes that can be reached from c by going forward;

   a. create a cluster containing X∪

{c}

b. N=N/(X ∪

6. {c})
7. Continue with step 4



- MinPts: Minimum number of points in any cluster
- ϵ

 F.or each point in cluster there must be another point in it less than this distance away.

 distance of appoint neighborhood: Points within ε

 N(p) :{q belongs to D |dist(p,q) <=ϵ
)
 Core point:
 - neighborhood dense enough (MinPts)

 Conditions: p belongs to N ϵ

(q)

|Nϵ

(q)| > = MinPts

Optics

Ordering points to identify the clustering structure (**OPTICS**) is an algorithm for finding density-based[1] clusters in spatial data. It was presented by Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel and Jörg Sander.[2] Its basic idea is similar to DBSCAN,[3] but it addresses one of DBSCAN's major weaknesses: the problem of detecting meaningful clusters in data of varying density. In order to do so, the points of the database are (linearly) ordered such that points which are spatially closest become neighbors in the ordering. Additionally, a special distance is stored for each point that represents the density that needs to be accepted for a cluster in order to have both points belong to the same cluster. This is represented as a dendrogram.

Graph Partitioning

The fundamental problem that is trying to solve is that of splitting a large irregular graphs into *k* parts. This problem has applications in many different areas including, parallel/distributed computing (load balancing of computations), scientific computing (fill-reducing matrix re-orderings), EDA algorithms for

VLSI CAD (placement), data mining (clustering), social network analysis (community discovery), pattern recognition, relationship network analysis, etc.

The *partitioning* is usually done so that it satisfies certain **constraints** and optimizes certain **objectives**. The most common constraint is that of producing equal-size partitions, whereas the most common objective is that of minimizing the number of cut edges (i.e., the edges that straddle partition boundaries). However, in many cases, different application areas tend to require their own type of constraints and objectives; thus, making the problem all that more interesting and challenging!

The research in the lab is focusing on a class of algorithms that have come to be known as **multilevel graph partitioning algorithms**. These algorithms solve the problem by following an *approximate-and-solve* paradigm, which is very effective for this as well as other (combinatorial) optimization problems.

Over the years we focused and produced good solutions for a number of graph-partitioning related problems. This includes partitioning algorithms for graphs corresponding to finite element meshes, multilevel nested dissection, parallel graph/mesh partitioning, dynamic/adaptive graph repartitioning, multi-constraint and multi-objective partitioning, and circuit and hypergraph partitioning.

Our latest research is focusing on three key areas:

- Mesh/graph partitioning algorithms that take into the fine-grain characteristics of the underlying parallel computer and can deal with heterogeneous computing and communication capabilities.
- Partitioning/load-balancing algorithms for mesh-less or mesh/particles scientific simulations.
- Partitioning algorithms for scale-free graphs and/or graphs whose degree distribution follows a power-low curve.

CHAMELEON

Chameleon Clustering

Combines initial partition of data with hierarchical clustering techniques it modifies clusters dynamically

Step1:

- Generate a KNN graph
- because it's local, it reduces influence of noise and outliers
- provides automatic adjustment for densities

Step2:

- use METIS: a graph partitioning algorithm
- get equally-sized groups of well-connected vertices
- this produces "sub-clusters" - something that is a part of true clusters

Step3:

- recombine sub-clusters
- combine two clusters if
  - they are relatively close
  - they are relatively interconnected
- so they are merged only if the new cluster will be similar to the original ones
- i.e. when "self-similarity" is preserved (similar to the join operation in Scatter/Gather)

But

- Curse of Dimensionality makes similarity functions behave poorly
- distances become more uniform as dimensionality grows
- and this makes clustering difficult

Similarity between two point of high dimensionality can be misleading

- often points may be similar even though they should belong to different clusters

COBWEB

COBWEB is an incremental system for hierarchical conceptual clustering. COBWEB was invented by Professor Douglas H. Fisher, currently at Vanderbilt University.[1][2]

COBWEB incrementally organizes observations into a classification tree. Each node in a classification tree represents a class (concept) and is labeled by a probabilistic concept that summarizes the attribute-value distributions of objects classified under the node. This classification tree can be used to predict missing attributes or the class of a new object.[3]

There are four basic operations COBWEB employs in building the classification tree. Which operation is selected depends on the category utility of the classification achieved by applying it. The operations are:

- Merging Two Nodes
  Merging two nodes means replacing them by a node whose children is the union of the original nodes' sets of children and which summarizes the attribute-value distributions of all objects classified under them.
- Splitting a node
  A node is split by replacing it with its children.
- Inserting a new node
  A node is created corresponding to the object being inserted into the tree.
- Passing an object down the hierarchy
  Effectively calling the COBWEB algorithm on the object and the subtree rooted in the node.

The COBWEB Algorithm

COBWEB(*root*, *record*):
Input: A COBWEB node *root*, an instance to insert *record*
if root has no children *then*
  children := {*copy*(*root*)}
  newcategory(*record*) \\ adds child with record's feature values.
  insert(*record*, *root*) \\ update root's statistics
else

```
  insert(record, root)
  for child in root's children do
    calculate Category Utility for insert(record, child),
    set best1, best2 children w. best CU.
  end for
  if newcategory(record) yields best CU then
    newcategory(record)
  else if merge(best1, best2) yields best CU then
    merge(best1, best2)
    COBWEB(root, record)
  else if split(best1) yields best CU then
    split(best1)
    COBWEB(root, record)
  else
    COBWEB(best1, record)
  end if
end
```

GCLuto.

CLUTO is a software package for clustering low- and high-dimensional datasets and for analyzing the characteristics of the various clusters. CLUTO is well-suited for clustering data sets arising in many diverse application areas including information retrieval, customer purchasing transactions, web, GIS, science, and biology.

CLUTO's distribution consists of both stand-alone programs and a library via which an application program can access directly the various clustering and analysis algorithms implemented in CLUTO.

Features

- Multiple classes of clustering algorithms:
  - partitional, agglomerative, & graph-partitioning based.
- Multiple similarity/distance functions:
  - Euclidean distance, cosine, correlation coefficient, extended Jaccard, user-defined.
- Numerous novel clustering criterion functions and agglomerative merging schemes.
- Traditional agglomerative merging schemes:
  - single-link, complete-link, UPGMA
- Extensive cluster visualization capabilities and output options:
  - postscript, SVG, gif, xfig, etc.
- Multiple methods for effectively summarizing the clusters:
  - most descriptive and discriminating dimensions, cliques, and frequent itemsets.
- Can scale to very large datasets containing hundreds of thousands of objects and tens of thousands of dimensions.

Possible Questions

PART – A (20 Marks)

(Q.No 1 to 20 Online Examinations)

PART – B (2 Marks)

1. What is classification?
2. Define : Graph Partitioning
3. What is logistic regression?
4. What is Birch algorithm?
5. What is clustering?

PART – C (6 Marks)

1. Demonstrate naïve bayes algorithms in detail.
2. Illustrate clustering based graph partitioning in detail.
3. Demonstrate logistic regression algorithms in detail.
4. Illustrate BIRCH clustering method in detail.
5. Demonstrate GMDH in detail.
6. Discuss in detail about phases of CHAMELEON algorithm.
7. Explain about multiple regression analysis.
8. Discuss in detail about partitional clustering with neat sketch.
9. Explain the phases of genetic algorithm.
10. Explain about the overview of COBWEB cluster analysis.

Karpagam Academy of Higher Education

Coimbatore - 641021

(For the candidates admitted from 2017 onwards)

Department of CS,CA & IT

UNIT - II:(Objective Type/Multiple Choice Questions Carries one mark)

Data mining and Data Warehousing 16CAP504D

UNIT -2						
Q.No	**Questions**	**Option1**	**Option2**	**Option3**	**Option4**	**Answer**
1	In decision tree, each _____ node represents a prediction of a solution to the problem	root	internal	leaf	middle	leaf
2	Bayesian Classifiers are_____classifiers	Numerical	Statistical	Floating	String	Statistical
3	PAM refers to____	Partitioning Around Medoids	Part Around Model	Partitioning Area Model	None	Partitioning Around Medoids
4	Expand SVM	Support vector motor	Self vector machine	Support vector machine	ort vertical mac	Support vector machine
5	Expand DBSCAN	Density based spatial clustering of applications with noise	Density based spatial clustering of approach with noise	Density based spatial clustering of applications with node	al classes of ap	Density based spatial clustering of applications with noise
6						
7	____is a clustering algorithm that uses dynamic modeling to determine the similarity between pairs ofclusters	CHAMELEON	BIRCH	ROCK	COBWEB	CHAMELEON

8	_____is designed for clustering a large amount of numerical data by integration of hierarchicalclustering and other clustering methods	CHAMELEON	BIRCH	ROCK	COBWEB	BIRCH
9	The _____ consists of many processing elements	decision tree	neural network	genetic algorithm	ctivation functio	neural network
10	In neural network, the output nodes are called as _____ node	end	terminal	sink	resultant	sink
11	Expand OPTICS	Ordering points to identify the clustering structure	Ordering points to identify the clustering structure	Ordering points to identify the clustering structure	Ordering points to identify the clustering structure	internal
12	In neural network all the three nodes has its own respective _____	states	stages	protocol	layers	layers
13	Neural network work only with _____ data	numeric	alphabetic	alphanumeric	special symbols	numeric
14	In neural network, the arcs are labeled with _____	values	rates	weights	heights	weights
15	One type of connectivity in neural network is _____ where some links are back to earlier layers	forward	feedback	backward	reback	feedback
16	_____ occurs when the neural network is trained to fit one set of data almost exactly	underfitting	upperbound	overfitting	lowerbound	overfitting
17	A _____ activiation function produces a linear output value based on the input	linear	threshold	guassian	sigmoid	linear
18	In _____ activation function, the output value is depend on the sum of the products of the input values and their associated weights	linear	threshold	guassian	sigmoid	threshold

19	The _____ function is a bell-shaped	linear	threshold	guassian	sigmoid	guassian
20	The _____ function is an S shaped curve with output values between -1 and 1	linear	threshold	guassian	sigmoid	sigmoid
21	Nodes in neural networks often have an exra input called a _____	unbias	bias	predict	outlier	bias
22	The _____ algorithm indicate how to combine the given set of individuals to produce new ones.	crossover	rule-based class	expectation maximization	genetic algorithm	crossover
23	The _____ operation randomly changes characters in the offspring	skew	fitness	mutation	crossover	mutation
24	A _____ function is used to determine the best individuals in a population	fitness	activation	mutation	skew	fitness
25	_____ data values cause problems during both the training phase and to the clasification process itself	training	invalid	incorrect	missing	missing
26	In operating characteristic curve the horizontal axis has the percentage of _____ positives	TRUE	FALSE	null	zero	FALSE
27	In operating characteristic curve the vertical axis has the percentage of _____ positives	TRUE	FALSE	null	zero	TRUE
28	_____ is erroneous data	outlier	invalid	noise	wrong	noise
29	Logistic regression uses a logistic _____	box	square	curve	line	curve
30	The _____ technique assumes that the entire training set includes not only the data in the set but also the desired classification for each item	C4.5	KNN	ID3	CART	KNN

31	The decision tree approach is most useful in _____ problems	classification	association	clustering	regression	classification
32	The decision tree approach to classification is to divide the search space into _____ regions	square	oval	rectangle	polygon	rectangle
33	The concept used in ID3 is to quantify information is called _____	pruning	entropy	splitting	divide and conquer	entropy
34	How many primary pruning strategies are proposed in C4.5	one	two	three	four	two
35	In subtree _____, a subtree is replaced by a leaf node	replacement	raising	placement	upraisal	replacement
36	In subtree _____, a subtree is replaced by its most used subtree	replacement	raising	placement	upraisal	raising
37	In _____, a gini index is used to find the best split	C4.5	ID3	SPRINT	CART	SPRINT
38	Neural network are more robust than _____ of the wieghts	genetic algorithm	similarity measure	dissimilarity measure	decision tree	decision tree
39	_____ learning is performed if the out is not known	unsupervised	nonsupervised	desupervised	supervised	unsupervised
40	The _____ can then be used to find a total error over all nodes in the network or over only the outpupt nodes	root mean square	root mean square error	mean squared error	jackknife	mean squared error
41	In order to modify the input weights, _____ rule examines not only the output value but also the desired value for output	learning rule	hebb rule	delta rule	backpropagation	delta rule
42	In _____ approach, the weights are changed once after all tuples in the training set are applied and a total MSE is found	online	offline	inline	endline	offline

43	In _____ approach, the weights are changed after each tuple in the training set is applied	online	offline	inline	endline	online
44	A radial basis function has a _____ shape	U	S	curve	gaussian	gaussian
45	The simplest neural network is called a _____	perception	perceptron	pipeline	predictors	perceptron
46	In linear regression, the n input variables are called _____	generators	response	prediction	predictors	predictors
47	In linear regression, the one output variables are called _____	generators	response	prediction	predictors	response
48	_____ is the problem of determining how much alike the two variables actually are	regression	generators	correlation	predictors	correlation
49	A _____ is a predictive modeling technique used in classification, clustering and prediction tasks.	activation function	genetic algorithm	neural network	decision tree	decision tree
50	A decision tree is a tree where the root and each internal node is labeled with a _____	question	answer	hint	decision	question

UNIT III

Mining Association rule in large Databases Association Rule Mining, Mining Single -Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Dataware houses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

Mining Frequent Patterns Associations and Correlations

Basic Concepts:

Mining frequent patterns is probably one of the most important concepts in data mining. A lot of other data mining tasks and theories stem from this concept. It should be the beginning of any data mining technical training because, on one hand, it gives a very well shaped idea about what data mining is and, on the other, it is not extremely technical.

Efficient and Scalable Frequent item set Mining Methods

Association Rule Mining:

• Search patterns given as association rules of the form

Body ⇒ Head [support, confidence]

Body: property of an object x e.g. a transaction, a person

Head: property probable to be implied by Body support, confidence: measures on validity of the rule.

• Examples

– buys(x, “diapers”) ⇒ buys(x, “beers”) [0.5%, 60%]

– major(x, “CS”) ∧ takes(x, “DB”) ⇒ grade(x, “A”) [1%, 75%]

• Problem: Given

(1) database of transactions

(2) each transaction is a list of items

Find: all rules that correlate the presence of one set of items with that

of another set of items.

Association rule mining is a technique for discovering unsuspected data

dependencies and is one of the best known data mining techniques. The basic idea is to identify from a given database, consisting of itemsets (e.g. shopping baskets), whether the occurrence of specific items, implies also the occurrence of other items with a relatively high probability.

Mining Various kinds of Association rules:

Single vs. Multidimensional Association Rules

- Single-dimensional rules

  buys(X, "milk") ⇒ buys(X, "bread")

- Multi-dimensional rules: more than 2 dimensions or predicates

  age(X,"19-25") ∧ buys(X, "popcorn") ⇒ buys(X, "coke")

- Transformation into single-dimensional rules:
  use predicate/value pairs as items

  customer(X, [age, "19-25"]) ∧ customer(X, [buys, "popcorn"])
  ⇒ customer(X, [buys,"coke"])

- Simplified Notation for single dimensional rules

  {milk} ⇒ {bread}
  {[age, "19-25"], [buys, "popcorn"]} ⇒ {[buys,"coke"]}

Definition of Association Rules

Terminology and Notation

Set of all items I, subset of I is called *itemset*

Transaction (tid, T), $T \subseteq I$ itemset, transaction identifier tid

Set of all transactions D (database), Transaction $T \in D$

Definition of Association Rules $A \Rightarrow B$ [s, c]

A, B itemsets ($A, B \subseteq I$)

$A \cap B$ empty

support s = probability that a transaction contains $A \cup B$

$= P(A \cup B)$

confidence c = conditional probability that a transaction having A also contains B

$= P(B|A)$

Example: Items I = {apple, beer, diaper, eggs, milk}

Transaction (2000, {beer, diaper, milk})

Association rule {beer} $\Rightarrow$ {diaper} [0.5, 0.66]

Mining Multidimensional Association Rules

- Single-dimensional rules
  buys(X, "milk") ⇒ buys(X, "bread")
- Multi-dimensional rules: more than 2 dimensions or predicates
  - Inter-dimension association rules (*no repeated predicates*)
    age(X,"19-25") ∧ occupation(X,"student") ⇒ buys(X,"coke")
  - hybrid-dimension association rules (*repeated predicates*)
    age(X,"19-25") ∧ buys(X, "popcorn") ⇒ buys(X, "coke")
- Transformation into single-dimensional rules
  - Items are predicate/value pairs
    customer(X, [age, "19-25"]) ∧
    customer(X, [occupation, "student"])
    ⇒ customer(X,[buys,"coke"])
    customer(X, [age, "19-25"]) ∧ customer(X, [buys, "popcorn"])
    ⇒ customer(X, [buys,"coke"])

Multidimensional association rules can be mined using the same method by transforming the problem. The items and the corresponding item values are encoded into a tuple. This results again in a finite number of possible (modified) item values, and therefore the same techniques as for single-dimensional rules apply.

Mining Quantitative Association Rules:

• Categorical Attributes

– finite number of possible values, no ordering among values

• Quantitative Attributes

– numeric, implicit ordering among values

• Quantitative attributes are transformed into categorical attributes by

– Static discretization of quantitative attributes

• Quantitative attributes are statically discretized by using predefined concept hierarchies.

Dynamic discretization

• Quantitative attributes are dynamically discretized into “bins” based on the distribution of the data. For quantitative attributes the situation is more complex. A simple approach is to statically or dynamically discretize them into categorical attributes. However, the rules that can be found depend on the discretization chosen. It may happen that the bins are for example too fine-grained, and a rule that could be more
efficiently be expressed at a coarser granularity is split into multiple rules.

Components of a Data Mining Algorithm:

• Model or pattern structure
– Which kind of global model or local pattern is searched
– Vector representation of documents
– Association Rules
• Score function
– Determine how well a given data set fits a model or pattern
– Similarity of vectors
– Support and confidence
• Optimization and search method
– Finding best parameters for a global model: optimization problem
– Finding data satisfying a pattern: search problem
– Search the k nearest neighbours
– Joining and pruning
• Data management strategy
– Handling very large datasets
– Inverted files
– Sampling, partitioning and transaction elimination

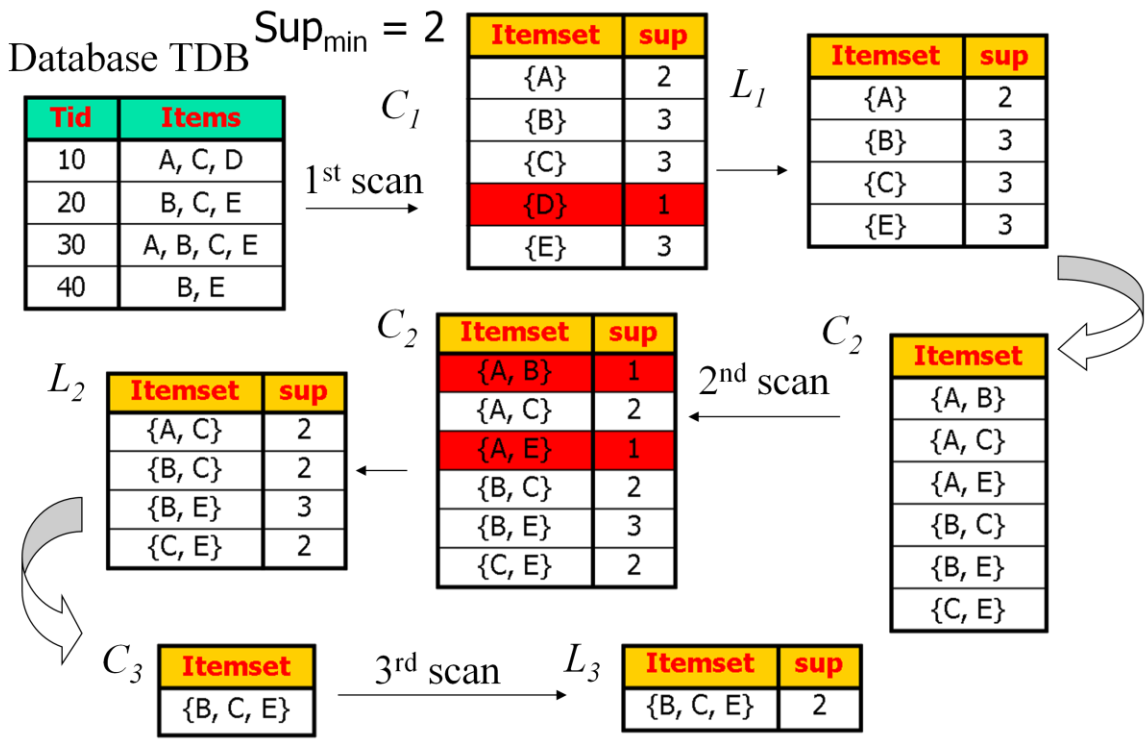
We illustrate here of how the four main components of data mining algorithms, are instantiated with association rule mining. Compare also to the corresponding methods used for vector space retrieval.

Apriori Algorithm:

- Apriori pruning principle: If there is any item set which is infrequent, its superset should not be generated/tested!
- Method:
  - Initially, scan DB once to get frequent 1-itemset
  - Generate length (k+1) candidate itemsets from length k frequent itemsets
  - Test the candidates against DB

  Terminate when no frequent or candidate set can be generated

The Apriori Algorithm—An Example



February 27, 2014 Data Mining: Concepts and Techniques 12

Pseudo-code for Apriori Algorithm:

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 t in 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 $\cup_k L_k$;

From Association Mining to Correlation Analysis:

Interestingness Measure: Correlations (Lift)

- *play basketball* $\Rightarrow$ *eat cereal* [40%, 66.7%] is misleading
  - The overall % of students eating cereal is 75% > 66.7%.
- *play basketball* $\Rightarrow$ *not eat cereal* [20%, 33.3%] is more accurate, although with lower support and confidence
- Measure of dependent/correlated events: lift

$$lift = \frac{P(A \cup B)}{P(A)P(B)}$$

	Basketball	Not basketball	Sum (row)
Cereal	2000	1750	3750
Not cereal	1000	250	1250
Sum(col.)	3000	2000	5000

$$lift(B,C) = \frac{2000/5000}{3000/5000 * 3750/5000} = 0.89 \qquad lift(B,\neg C) = \frac{1000/5000}{3000/5000 * 1250/5000} = 1.33$$

***lift* and χ^2 Good Measures of Correlation:**

$$lift = \frac{P(A \cup B)}{P(A)P(B)}$$

$$all\_conf = \frac{\sup(X)}{\max\_item\_\sup(X)}$$

$$coh = \frac{\sup(X)}{|universe(X)|}$$

Correlation analysis

Association rule mining often generates a huge number of rules, but a majority of them either are redundant or do not reflect the true correlation relationship among data objects.

•Some strong association rules (based on support and confidence) can be misleading.

•Correlation analysis can reveal which strong association rules are interesting and useful.

Eg: play basketball eat cereal[40%, 66.7%] is misleading

–The overall % of students eating cereal is 75% > 66.7%.

•play basketball not eat cereal[20%, 33.3%] is more accurate, although with lower support and confidence

Basketball	Not basketball		Sum (row)
Cereal	2000 (40%)	1750 (35%)	3750 (75%)
Not cereal	1000 (20%)	250 (5%)	1250 (25%)

Sum(col.)	3000 (60%)	2000 (40%)	5000 (100%)

Correlation analysis The lift score as follows,

•Lift = 1 A and B are independent

•Lift > 1 A and B are positively correlated

•Lift < 1 A and B are negatively correlated.

Basketball	Not basketball		Sum (row)
Cereal	2000 (40%)	1750 (35%)	3750 (75%)
Not cereal	1000 (20%)	250 (5%)	1250 (25%)
Sum(col.)	3000 (60%)	2000 (40%)	5000 (100%)

Correlation analysis The χ2 test:

•Lift calculates the correlation value, but we could not tell whether the value is statistically significant.

•Pearson Chi-squareis the most common test for significance of the relationship between categorical variables

•If this value is larger than a cutoff value at a significance level (e.g. at 95% significance level), then we say all the variables are dependent (correlated), else we say all the variables are independent.

Other correlation/interestingness measure: cosine, all confidence, IG…

Correlation analysis disadvantages:

Problem: Evaluate each rule individually!

Pr(CHD)=30%

R2: Family history=yes ∧Race=Caucasian ⇒CHD

[sup=20%, conf=55%]

R1: Family history=yes ⇒CHD

[sup=50%, conf=60%]

We should consider the nested structure of the rules!

To solve this problem, we proposed the MDR framework.

Constraint-based Mining:

- Knowledge type constraint:
  - Classification, association, etc.
- Data constraint — using SQL-like queries
  - find product pairs sold together in stores in Chicago in Dec.'02
- Dimension/level constraint
  - in relevance to region, price, brand, customer category
- Rule (or pattern) constraint
  - small sales (price < $10) triggers big sales (sum > $200)
- Interestingness constraint
  - strong rules: min_support ≥ 3%, min_confidence ≥ 60%

Constrained mining vs. constraint-based search/reasoning:

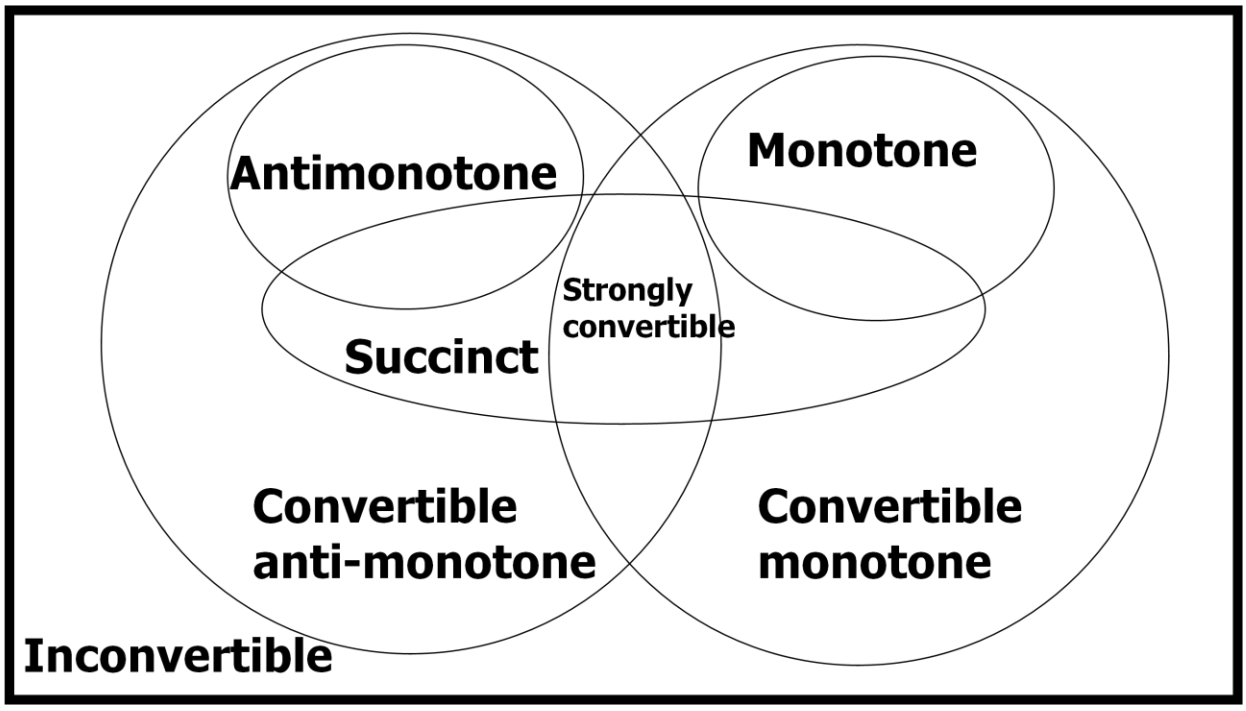
- Constrained mining vs. constraint-based search/reasoning
  - Both are aimed at reducing search space
  - Finding all patterns satisfying constraints vs. finding some (or one) answer in constraint-based search in AI
  - Constraint-pushing vs. heuristic search
  - It is an interesting research problem on how to integrate them

- Constrained mining vs. query processing in DBMS
  - Database query processing requires to find all
  - Constrained pattern mining shares a similar philosophy as pushing selections deeply in query processing

A Classification of Constraints:

In a constrained Hamiltonian system, a dynamical quantity is called a first class constraint if its Poisson bracket with all the other constraints vanishes on the constraint surface. A second class constraint is one that is not first class.

a cutoff value at a significance level (e.g. at 97% significance level), then we say all the variables are dependent (correlated), else we say all the variables are independent.



Handling Multiple Constraints

- Different constraints may require different or even conflicting item-ordering
- If there exists an order R s.t. both C_1 and C_2 are convertible w.r.t. R, then there is no conflict between the two convertible constraints
- If there exists conflict on order of items

- Try to satisfy one constraint first
- Then using the order for the other constraint to mine frequent itemsets in the corresponding projected database

What Constraints Are Convertible in Data Mining:

Constraint	Convertible anti-monotone	Convertible monotone	Strongly convertible
avg(S) ≤ , ≥ v	Yes	Yes	Yes
median(S) ≤ , ≥ v	Yes	Yes	Yes
sum(S) ≤ v (items could be of any value, v ≥ 0)	Yes	No	No
sum(S) ≤ v (items could be of any value, v ≤ 0)	No	Yes	No
sum(S) ≥ v (items could be of any value, v ≥ 0)	No	Yes	No
sum(S) ≥ v (items could be of any value, v ≤ 0)	Yes	No	No
……			

Then using the order for the other constraint to mine frequent item sets in the corresponding projected database

However, the rules that can be found depend on the discretization chosen. It may happen that the bins are for example too fine-grained, and a rule that could be more

Constraint-Based Mining—A General Picture :

Constraint	Antimonotone	Monotone	Succinct
$v \in S$	No	yes	yes
$S \supseteq V$	No	yes	yes
$S \subseteq V$	Yes	no	yes
$\min(S) \leq v$	No	yes	yes
$\min(S) \geq v$	Yes	no	yes
$\max(S) \leq v$	Yes	no	yes
$\max(S) \geq v$	No	yes	yes
$\text{count}(S) \leq v$	yes	no	weakly
$\text{count}(S) \geq v$	No	yes	weakly
$\text{sum}(S) \leq v$ ($a \in S, a \geq 0$)	Yes	no	no
$\text{sum}(S) \geq v$ ($a \in S, a \geq 0$)	No	yes	no
$\text{range}(S) \leq v$	Yes	no	no
$\text{range}(S) \geq v$	No	yes	no
$\text{avg}(S)\ \theta\ v, \theta \in \{=, \leq, \geq\}$	convertible	convertible	no
$\text{support}(S) \geq \xi$	Yes	no	no
$\text{support}(S) \leq \xi$	No	yes	no

Frequent-Pattern Mining:

- Frequent pattern mining—an important task in data mining
- Scalable frequent pattern mining methods
  - Apriori (Candidate generation & test)
  - Projection-based (FPgrowth, CLOSET+, ...)
  - Vertical format approach (CHARM, ...)
- Mining a variety of rules and interesting patterns
- Constraint-based mining
- Mining sequential and structured patterns
- Extensions and applications

Frequent-Pattern Mining: Research Problems:

- Mining fault-tolerant frequent, sequential and structured patterns
  - Patterns allows limited faults (insertion, deletion, mutation)
- Mining truly interesting patterns
  - Surprising, novel, concise, …
- Application exploration
  - E.g., DNA sequence analysis and bio-pattern classification
  - "Invisible" data mining

Possible Questions

PART – A (20 Marks)

(Q.No 1 to 20 Online Examinations)

PART – B (2 Marks)

1. What is association rule?
2. Define : Transactional Database
3. What is constraint based association mining?
4. What is multi dimensional rule?
5. Define : Correlation analysis

PART – C (6 Marks)

1. Discuss in detail about apriori algorithm.
2. Explain about meta-rule guided mining of association rules.
3. Discuss in detail about mining multilevel association rules from transaction databases.
4. Explain Multidimensional association rules with examples.
5. Discuss in detail about mining Multi-D association rule from data warehouses.
6. Explain about the single dimensional Boolean association rules.
7. Discuss the techniques used to improve the efficiency of Apriori algorithm.
8. Explain how correlation analysis supports association rule analysis in Data Mining.
9. Explain the Mining Frequent Item sets without Candidate Generation.
10. Explain about Mining quantitative association rules.

Karpagam Academy of Higher Education

Coimbatore - 641021

(For the candidates admitted from 2017 onwards)

Department of CS,CA & IT

UNIT - III:(Objective Type/Multiple Choice Questions Carries one mark)

Data mining and Data Warehousing 16CAP504D

UNIT -3						
Q.No	**Questions**	**Option1**	**Option2**	**Option3**	**Option4**	**Answer**
1	Association rules generated from mining data at multiple levels of abstraction are called	Multilevel association rule	Single level association rule	Classification rule	Clustering rule	Multilevel association rule
2	Using uniform minimum support for all levels refers to__	Uniform support	Multi support	Single Support	Mining Support	Uniform support
3	Using reduced minimum support at lower levels refers to_____	Uniform support	Multi support	Single Support	Reduced Support	Reduced Support
4	Using item or group based support refer to_____	Group based support	Multi support	Single Support	Reduced Support	Group based support
5	Intradimensional association rule also refers to___	Multi dimensional	Single Dimensional	Both multi and single	None	Single Dimensional
6	A____is a set containing k conjunctive predicates	Predicate set	Non-Predicate set	Single predicate	Null predicate	Predicate set
7	In_____the interval size of each bit is the same	Equal width binning	Equal Frequency Binning	Clustering based binning	Non Clustering based binning	Equal width binning
8	In____each bin has approximately the samenumber of tuples assigned to it	Equal width binning	Equal Frequency Binning	Clustering based binning	None	Equal Frequency Binning

9	in ____clustering is performed on the quantitative attribute to group neighboring points	Equal width binning	Equal Frequency Binning	Clustering based binning	None	Clustering based binning
10	____specifies the typeof knowledge to be mined such as association or correlations	Rule	Interestingness	Data	Knowledge	Knowledge
11	____the set of task relavant data	Rule	Interestingness	Data	Knowledge	Data
12	____specifies the desired dimensions of the data or level of the concept hierarchies.	Rule	Interestingness	Data	Dimension	Dimension
13	_____specifies the thresholds on statistical measures of rule interestingness,such as support, confidence and correlation	Rule	Interestingness	Data	Knowledge	Interestingness
14	_____specifies the form of rules to be mined	Rule	Interestingness	Data	Knowledge	Rule
15	_____ model makes a prediction about values of data using known results found from different data.	predictive	descriptive	preference	process	predictive
16	_____ model identifies patterns or relationships in data.	predictive	descriptive	process	preference	descriptive
17	_____ maps data into predefined groups or classes	regression	classification	clustering	prediction	classification
18	_____ is used to map a data item to a real valued prediction variable.	regression	classification	clustering	prediction	regression
19	Classification of input patterns by using its similarity to the predefined classes is called____	prediction	classification	pattern recognition	regression	pattern recognition
20	A special type of clustering is called _____	classification	segmentation	regression	prediction	segmentation
21	_____ maps data into subsets with associated simple description.	classification	segmentation	pattern recognition	summarizatio n	summarization
22	Which rule is used to identify the specific type of data association	delta rule	association rule	classification rule	segmentation rule	association rule

23	Which is used to determine sequential patterns in data?	sequence discovery	segmentation	regression	random analysis	sequence discovery
24	_____ is the process of finding useful information and patterns in data	datamining	KDD	selection	transfermation	KDD
25	_____ is the use of algorithms to extract the information and pattern derived by the knowledge discovery	datamining	KDD	selection	transfermation	datamining
26	The KDD process is often said to be _____	trivial	nontrivial	significant	consequential	nontrivial
27	Extreme values that occur infrequently, may actually be removed is called _____	outliers	outlayers	statistics	frequency	outliers
28	Traditional graph structure is a _____ visualization techniqe	graphical	geometric	icon-based	pixel-based	graphical
29	The major goal of _____ is to be able to describe the result in meaningful manner.	datamining	KDD	selection	transfermation	KDD
30	_____ techniques often involve sophisticated multimedia and graphics presentations.	summarization	conceptualization	visualization	decription	visualization
31	Describing a large database can be viewed as using _____ to help uncover hidden information about the data	approximation	search	induction	compression	approximation
32	_____ is used to proceed from very specific knowledge to more general information.	approximation	search	induction	compression	induction
33	_____ occurs when the model does not fit future states.	outliers	interception	overfitting	qureying	overfitting
34	The problem of increasing over all complexity and decrease the efficiency of an algorithms is _____	Single dimensionality	Multi-Dimensionality	Dimensionality curse	Recursion	Dimensionality curse
35	A _____ is the one that involves categorical and quantitative data.	Generalized association rule	Increemental rule	Multiple level association rule	Quantitative association rule	Quantitative association rule

36	Find out predictive model datamining task from the followings	clustering	summarization	association rule	regression	regression
37	Find out descriptive model datamining task from the followings	classification	sequence discovery	prediction	regression	sequence discovery
38	Flooding, speech recognition, machine learning are_____	classification applicationa	prediction applications	clustering application	regression applications	prediction applications
39	Link analysis alternatively referred to as a _____	rule analysis	affinity analysis	master analysis	report analysis	affinity analysis
40	Clustering is similar to _____	Summarization	Classification	Association	Regression	Classification
41	Smallest distance between an element in one cluster and an element in the other is called as_____.	Multiple link	Complete link	Single link	Centroid	Single link
42	Largest distance between an element in one cluster and an element in the other is called as_____.	Multiple link	Complete link	Single link	Centroid	Complete link
43	If clusters have a representative centroid, then the centroid distance is defined as the distance between the _____.	Medoid	Outliers	Clusters	Centroid	Centroid
44	_____ are sample points with values much different from those of the remaning set of data.	Outlier	Centroid	Medoid	Cluster	Outlier
45	_____ is the process of identifying outlieers in a set of data.	Discordancy tests	Outlier detection	Statistical Technique	sequential test	Outlier detection
46	A tree data structure, called a _____ can be used to illustrate clustering technique.	Centroid	Outlier	Dendrogram	information gain	Dendrogram
47	The root in the dendrogram tree contains one cluster where all elements are _____.	Same	Different	Unique	Together	Together
48	The leaves in the dendrogram consists of a _____ element cluster.	Single	Same	Unique	Distinct	Single

49	Association rule that involves more than one dimension is called	Interdimensional Association rule	Intradimensional association rule	Multi dimensional association rule	Single diamensional association rule	Multi dimensional association rule
50	Multidimensional association rule with no repeated predicates are called____	Interdimensional Association rule	Intradimensional association rule	Null rule	Single diamensional association rule	Interdimensional Association rule

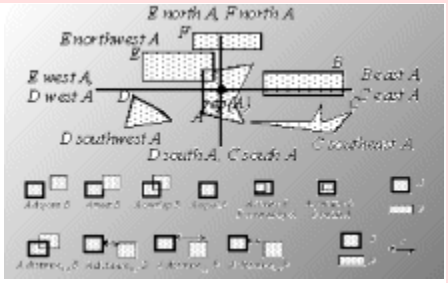
UNIT IV

Mining Complex Types of Data : Mining Spatial Databases – Multimedia Databases – Time-series and Sequence Data – Text Databases – Web Data Mining – Search Engines.

Mining Spatial Databases

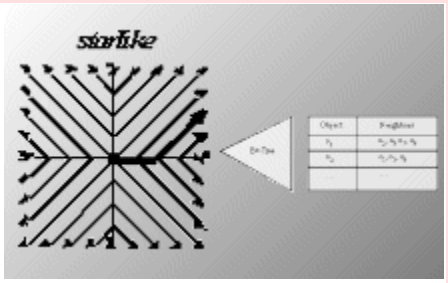
The main difference between data mining in relational DBS and in spatial DBS is that attributes of the neighbors of some object of interest may have an influence on the object and therefore have to be considered as well. The explicit location and extension of spatial objects define implicit relations of spatial neighborhood (such as topological, distance and direction relations) which are used by spatial data mining algorithms. Therefore, new techniques are required for effective and efficient data mining.

Techniques



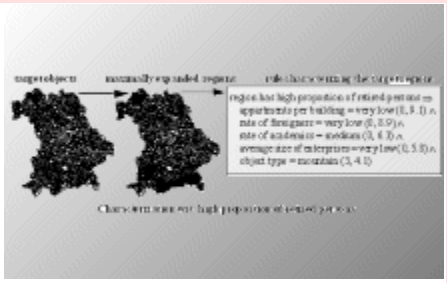
Database Primitives for Spatial Data Mining

We have developed a set of database primitives for mining in spatial databases which are sufficient to express most of the algorithms for spatial data mining and which can be efficiently supported by a DBMS. We believe that the use of these database primitives will enable the integration of spatial data mining with existing DBMS's and will speed-up the development of new spatial data mining algorithms. The database primitives are based on the concepts of neighborhood graphs and neighborhood paths.



Efficient DBMS Support

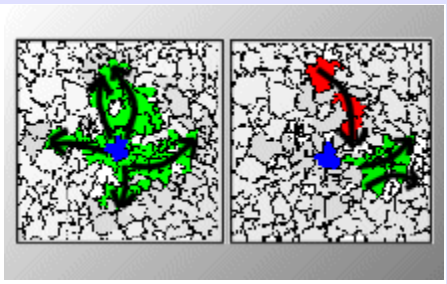
Effective filters allow to restrict the search to such neighborhood paths "leading away" from a starting object. Neighborhood indices materialize certain neighborhood graphs to support efficient processing of the database primitives by a DBMS. The database primitives have been implemented on top of the DBMS Illustra and are being ported to Informix Universal Server.



Algorithms for Spatial Data Mining

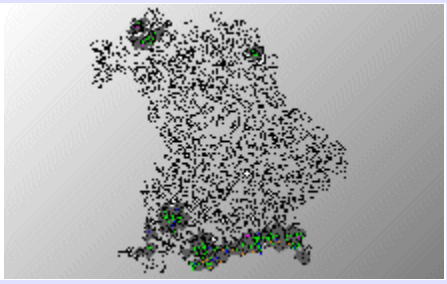
New algorithms for spatial characterization and spatial trend analysis were developed. For spatial characterization it is important that class membership of a database object is not only determined by its non-spatial attributes but also by the attributes of objects in its neighborhood. In spatial trend analysis, patterns of change of some non-spatial attributes in the neighborhood of a database object are determined.

Applications



Spatial Trend Detection in GIS

Spatial trends describe a regular change of non-spatial attributes when moving away from certain start objects. Global and local trends can be distinguished. To detect and explain such spatial trends, e.g. with respect to the economic power, is an important issue in economic geography.



Spatial Characterization of Interesting Regions

Another important task of economic geography is to characterize certain target regions such as areas with a high percentage of retirees. Spatial characterization does not only consider the attributes of the target regions but also neighboring regions and their properties.

Multimedia Databases

Multimedia data typically means digital images, audio, video, animation and graphics together with text data. The acquisition, generation, storage and processing of multimedia data in computers and transmission over networks have grown tremendously in the recent past.

This astonishing growth is made possible by three factors. Firstly, personal computers usage becomes widespread and their computational power gets increased. Also technological advancements resulted in high-resolution devices, which can capture and display multimedia data (digital cameras, scanners, monitors, and printers). Also there came high-density storage devices. Secondly high-speed data communication networks are available nowadays. The Web has wildly proliferated and software for manipulating multimedia data is now available. Lastly, some specific applications (existing) and future applications need to live with multimedia data. This trend is expected to go up in the days to come.

Multimedia data are blessed with a number of exciting features. They can provide more effective dissemination of information in science, engineering , medicine, modern biology, and social sciences. It also facilitates the development of new paradigms in distance learning, and interactive personal and group entertainment.

The huge amount of data in different multimedia-related applications warranted to have databases as databases provide consistency, concurrency, integrity, security and availability of data. From an user perspective, databases provide functionalities for the easy manipulation, query and retrieval of highly relevant information from huge collections of stored data.

MultiMedia Databases (MMDBs) have to cope up with the increased usage of a large volume of multimedia data being used in various software applications. The applications include digital libraries, manufacturing and retailing, art and entertainment, journalism and so on. Some inherent qualities of multimedia data have both direct and indirect influence on the design and development of a multimedia database. MMDBs are supposed to provide almost all the functionalities, a traditional database provides. Apart from those, a MMDB has to provide some new and enhanced functionalities and features. MMDBs are required to provide unified frameworks for storing, processing, retrieving, transmitting and presenting a variety of media data types in a wide variety of formats. At the same time, they must adhere to numerical constraints that are normally not found in traditional databases.

Contents of MMDB

An MMDB needs to manage several different types of information pertaining to the actual multimedia data. They are:

- Media data - This is the actual data representing images, audio, video that are captured, digitized, processes, compressed and stored.
- Media format data - This contains information pertaining to the format of the media data after it goes through the acquisition, processing, and encoding phases. For instance, this consists of information such as the sampling rate, resolution, frame rate, encoding scheme etc.
- Media keyword data - This contains the keyword descriptions, usually relating to the generation of the media data. For example, for a video, this might include the date, time, and place of recording , the person who recorded, the scene that is recorded, etc This is also called as content descriptive data.
- Media feature data - This contains the features derived from the media data. A feature characterizes the media contents. For example, this could contain information about the distribution of colors, the kinds of textures and the different shapes present in an image. This is also referred to as content dependent data.

  The last three types are called meta data as they describe several different aspects of the media data. The media keyword data and media feature data are used as indices for searching purpose. The media format data is used to present the retrieved information.

Designing MMDBs

Many inherent characteristics of multimedia data have direct and indirect impacts on the design of MMDBs. These include : the huge size of MMDBs, temporal nature, richness of content, complexity of representation and subjective interpretation. The major challenges in designing multimedia databases arise from several requirements they need to satisfy such as the following:

1. Manage different types of input, output, and storage devices. Data input can be from a variety of devices such as scanners, digital camera for images, microphone, MIDI devices for audio, video cameras. Typical output devices are high-resolution monitors for images and video, and speakers for audio.
2. Handle a variety of data compression and storage formats. The data encoding has a variety of formats even within a single application. For instance, in medical applications, the MRI images of brain has lossless or very stringent quality of lossy coding technique, while the X-ray images of bones can be less stringent. Also, the radiological image data, the ECG data, other patient data, etc. have widely varying formats.

3. Support different computing platforms and operating systems. Different users operate computers and devices suited to their needs and tastes. But they need the same kind of user-level view of the database.
4. Integrate different data models. Some data such as numeric and textual data are best handled using a relational database model, while some others such as video documents are better handled using an object-oriented database model. So these two models should coexist together in MMDBs.
5. Offer a variety of user-friendly query systems suited to different kinds of media. From a user point of view, easy-to-use queries and fast and accurate retrieval of information is highly desirable. The query for the same item can be in different forms. For example, a portion of interest in a video can be queried by using either

   1) a few sample video frames as an example,

   2) a clip of the corresponding audio track or

   3) a textual description using keywords.

6. Handle different kinds of indices. The inexact and subjective nature of multimedia data has rendered keyword-based indices and exact and range searches used in traditional databases ineffective. For example, the retrieval of records of persons based on social security number is precisely defined, but the retrieval of records of persons having certain facial features from a database of facial images requires, content-based queries and similarity-based retrievals. This requires indices that are content dependent, in addition to key-word indices.
7. Develop measures of data similarity that correspond well with perceptual similarity. Measures of similarity for different media types need to be quantified to correspond well with the perceptual similarity of objects of those data types. These need to be incorporated into the search process
8. Provide transparent view of geographically distributed data. MMDBs are likely to be a distributed nature. The media data resides in many different storage units possibly spread out geographically. This is partly due to the changing nature of computation and computing resources from centralized to networked and distributed.
9. Adhere to real-time constraints for the transmission of media data. Video and audio are inherently temporal in nature. For example, the frames of a video need to be presented at the rate of at least 30 frames/sec. for the eye to perceive continuity in the video.
10. Synchronize different media types while presenting to user. It is likely that different media types corresponding to a single multimedia object are stored in different formats, on different devices, and have different rates of transfer. Thus they need to be periodically synchronized for presentation.

The recent growth in using multimedia data in applications has been phenomenal. Multimedia databases are essential for efficient management and effective use of huge amounts of data. The diversity of applications using multimedia data, the rapidly changing technology, and the inherent complexities in the semantic representation, interpretation and comparison for similarity pose many challenges. MMDBs are still in their infancy. Today's MMDBs are closely bound to narrow application areas. The experiences acquired from developing and using novel multimedia applications will help advance the multimedia database technology.

Time-series and Sequence Data

A **time series** is a series of data points indexed (or listed or graphed) in time order. Most commonly, a time series is a sequence taken at successive equally spaced points in time. Thus it is a sequence of discrete-time data. Examples of time series are heights of ocean tides, counts of sunspots, and the daily closing value of the Dow Jones Industrial Average.

Time series are very frequently plotted via line charts. Time series are used in statistics, signal processing, pattern recognition, econometrics, mathematical finance, weather forecasting, earthquake prediction, electroencephalography, control engineering, astronomy, communications engineering, and largely in any domain of applied science and engineering which involves temporal measurements.

Time series *analysis* comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. **Time series *forecasting*** is the use of a model to predict future values based on previously observed values. While regression analysis is often employed in such a way as to test theories that the current values of one or more independent time series affect the current value of another time series, this type of analysis of time series is not called "time series analysis", which focuses on comparing values of a single time series or multiple dependent time series at different points in time.[1] Interrupted time series analysis is the analysis of interventions on a single time series

Time series data have a natural temporal ordering. This makes time series analysis distinct from cross-sectional studies, in which there is no natural ordering of the observations (e.g. explaining people's wages by reference to their respective education levels, where the individuals' data could be entered in any order). Time series analysis is also distinct from spatial data analysis where the observations typically relate to geographical locations (e.g. accounting for house prices by the location as well as the intrinsic characteristics of the houses). A stochastic model for a time series will generally reflect the fact that observations close together in time will be more closely related than observations further apart. In addition, time series models will often make use of the natural one-way ordering of time so that values for a given period will be expressed as deriving in some way from past values, rather than from future values (see time reversibility.)

What is sequential pattern mining?

Data mining consists of extracting information from data stored in databases to understand the data and/or take decisions. Some of the most fundamental data mining tasks are clustering, classification, outlier analysis, and pattern mining. **Pattern mining** consists of discovering interesting, useful, and unexpected patterns in databases Various types of patterns can be discovered in databases such as frequent itemsets, associations, subgraphs, sequential rules, and periodic patterns.

The task of **sequential pattern mining** is a data mining task specialized for analyzing **sequential data,** to discover **sequential patterns**. More precisely, it consists of discovering interesting subsequences in **a set of sequences**, where the interestingness of a subsequence can be measured in terms of various criteria such as its occurrence frequency, length, and profit. Sequential pattern mining has numerous real-life applications due to the fact that data is naturally encoded as **sequences of symbols** in many fields such as bioinformatics, e-learning, market basket analysis, texts, and webpage click-stream analysis.

I will now explain the task of **sequential pattern mining** with an example. Consider the following **sequence database**, representing the purchases made by customers in a retail store.

SID	Sequence
1	$\langle\{a,b\},\{c\},\{f,g\},\{g\},\{e\}\rangle$
2	$\langle\{a,d\},\{c\},\{b\},\{a,b,e,f\}\rangle$
3	$\langle\{a\},\{b\},\{f,g\},\{e\}$
4	$\langle\{b\},\{f,g\}\rangle$

This database contains four sequences. Each **sequence** represents the items purchased by a customer at different times. A sequence is an ordered list of itemsets (sets of items bought together). For example, in this database, the first sequence (SID 1) indicates that a customer bought some items *a* and *b* together, then purchased an item *c*, then purchased items *f* and *g* together, then purchased an item *g*, and then finally purchased an item *e.*

Traditionally, sequential pattern mining is being used to find subsequences that appear often in a sequence database, i.e. that are common to several sequences. Those subsequences are called the **frequent sequential patterns**. For example, in the context of our example, sequential pattern mining can be used to find the sequences of items frequently bought by customers. This can be useful to understand the behavior of customers to take marketing decisions.

To do **sequential pattern mining**, a user must provide a sequence database and specify a parameter called the **minimum support threshold**. This parameter indicates a minimum number of sequences in which a pattern must appear to be considered frequent, and be shown to the user. For example, if a user sets the minimum support threshold to 2 sequences, the task of **sequential pattern mining** consists of finding all subsequences appearing in at least 2 sequences of the input database. In the example database, 29 subsequences met this requirement. These sequential patterns are shown in the table below,

where the number of sequences containing each pattern (called the *support*) is indicated in the right column

Sequential patterns found

Pattern	Sup.
$\langle\{a\}\rangle$	3
$\langle\{a\},\{g\}\rangle$	2
$\langle\{a\},\{g\},\{e\}\rangle$	2
$\langle\{a\},\{f\}\rangle$	3
$\langle\{a\},\{f\},\{e\}\rangle$	2
$\langle\{a\},\{c\}\rangle$	2
$\langle\{a\},\{c\},\{f\}\rangle$	2
$\langle\{a\},\{c\},\{e\}\rangle$	2
$\langle\{a\},\{b\}\rangle$	2
$\langle\{a\},\{b\},\{f\}\rangle$	2
$\langle\{a\},\{b\},\{e\}\rangle$	2
$\langle\{a\},\{e\}\rangle$	3
$\langle\{a,b\}\rangle$	2
$\langle\{b\}\rangle$	4
$\langle\{b\},\{g\}\rangle$	3
$\langle\{b\},\{g\},\{e\}\rangle$	2
$\langle\{b\},\{f\}\rangle$	4
$\langle\{b\},\{f,g\}\rangle$	2
$\langle\{b\},\{f\},\{e\}\rangle$	2
$\langle\{b\},\{e\}\rangle$	3
$\langle\{c\}\rangle$	2
$\langle\{c\},\{f\}\rangle$	2
$\langle\{c\},\{e\}\rangle$	2
$\langle\{e\}\rangle$	3
$\langle\{f\}\rangle$	4
$\langle\{f,g\}\rangle$	2
$\langle\{f\},\{e\}\rangle$	2
$\langle\{g\}\rangle$	3
$\langle\{g\},\{e\}\rangle$	2

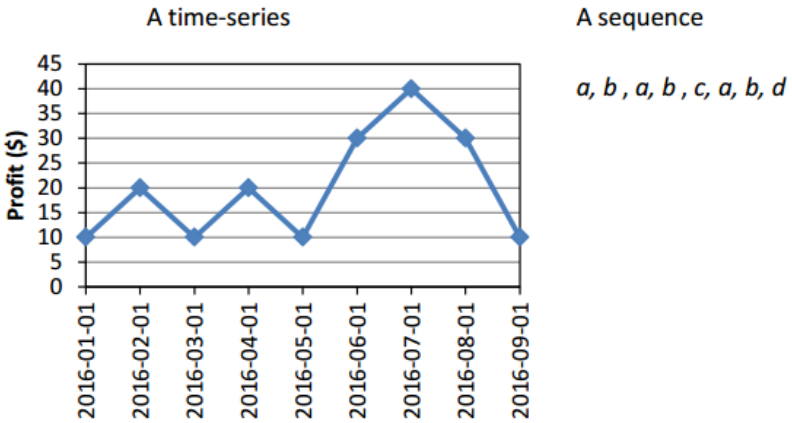
of the table.

For example, the patterns <{a}> and <{a}, {g}> are frequent and have a support of 3 and 2 sequences, respectively. In other words, these patterns appears in 3 and 2 sequences of the input database, respectively. The pattern <{a}> appears in the sequences 1, 2 and 3, while the pattern <{a}, {g}> appears in sequences 1 and 3. These patterns are interesting as they represent some behavior common to several customers. Of course, this is a toy example. Sequential pattern mining can actually be applied on database containing hundreds of thousands of sequences.

Another example of application of sequential pattern mining is text analysis. In this context, a set of sentences from a text can be viewed as sequence database, and the goal of sequential pattern mining is then to find subsequences of words frequently used in the text. If such sequences are contiguous, they are called "ngrams" in this context. If you want to know more about this application, you can read this blog post, where sequential patterns are discovered in a Sherlock Holmes novel.

Can sequential pattern mining be applied to time series?

Besides sequences, **sequential pattern mining** can also be applied to **time series** (e.g. stock data), when discretization is performed as a pre-processing step. For example, the figure below shows a **time series** (an ordered list of numbers) on the left. On the right, a **sequence** (a sequence of symbols) is shown representing the same data, after applying a transformation. Various transformations can be done to transform a time series to a sequence such as the popular SAX transformation. After performing the transformation, any sequential pattern mining algorithm can be applied.



A time-series (left) and a sequence (right)

Where can I get Sequential pattern mining implementations?

To try sequential pattern mining with your datasets, you may try the open-source **SPMF data mining software**, which provides implementations of numerous **sequential pattern mining algorithms:** http://www.philippe-fournier-viger.com/spmf/

It provides implementations of several algorithms for sequential pattern mining, as well as several variations of the problem such as discovering **maximal sequential patterns**, **closed sequential patterns** and sequential rules. Sequential rules are especially useful for the purpose of performing predictions, as they also include the concept of confidence.

What are the current best algorithms for sequential pattern mining?

There exists several sequential pattern mining algorithms. Some of the classic algorithms for this problem are **PrefixSpan, Spade, SPAM,** and **GSP**. However, in the recent decade, several novel and

more efficient algorithms have been proposed such as **CM-SPADE** and **CM-SPAM** (2014), **FCloSM** and **FGenSM** (2017), to name a few. Besides, numerous algorithms have been proposed for extensions of the problem of sequential pattern mining such as finding the sequential patterns that generate the most profit (high utility sequential pattern mining).

Text Databases

Text databases consist of huge collection of documents. They collect these information from several sources such as news articles, books, digital libraries, e-mail messages, web pages, etc. Due to increase in the amount of information, the text databases are growing rapidly. In many of the text databases, the data is semi-structured.

For example, a document may contain a few structured fields, such as title, author, publishing_date, etc. But along with the structure data, the document also contains unstructured text components, such as abstract and contents. Without knowing what could be in the documents, it is difficult to formulate effective queries for analyzing and extracting useful information from the data. Users require tools to compare the documents and rank their importance and relevance. Therefore, text mining has become popular and an essential theme in data mining.

Information Retrieval

Information retrieval deals with the retrieval of information from a large number of text-based documents. Some of the database systems are not usually present in information retrieval systems because both handle different kinds of data. Examples of information retrieval system include −

- Online Library catalogue system
- Online Document Management Systems
- Web Search Systems etc.

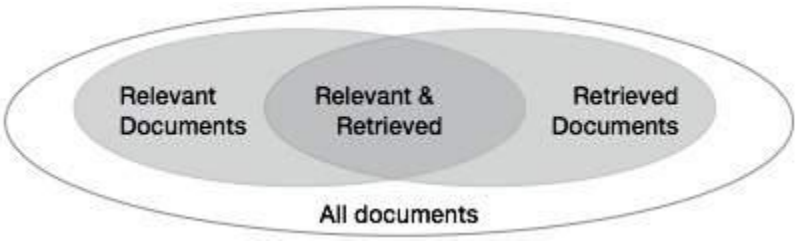
Note − The main problem in an information retrieval system is to locate relevant documents in a document collection based on a user's query. This kind of user's query consists of some keywords describing an information need.

In such search problems, the user takes an initiative to pull relevant information out from a collection. This is appropriate when the user has ad-hoc information need, i.e., a short-term need. But if the user has a long-term information need, then the retrieval system can also take an initiative to push any newly arrived information item to the user.

This kind of access to information is called Information Filtering. And the corresponding systems are known as Filtering Systems or Recommender Systems.

Basic Measures for Text Retrieval

We need to check the accuracy of a system when it retrieves a number of documents on the basis of user's input. Let the set of documents relevant to a query be denoted as {Relevant} and the set of retrieved document as {Retrieved}. The set of documents that are relevant and retrieved can be denoted as {Relevant} ∩ {Retrieved}. This can be shown in the form of a Venn diagram as follows −



There are three fundamental measures for assessing the quality of text retrieval −

- Precision
- Recall
- F-score

Precision

Precision is the percentage of retrieved documents that are in fact relevant to the query. Precision can be defined as −

Precision= |{Relevant} ∩ {Retrieved}| / |{Retrieved}|

Recall

Recall is the percentage of documents that are relevant to the query and were in fact retrieved. Recall is defined as −

Recall = |{Relevant} ∩ {Retrieved}| / |{Relevant}|

F-score

F-score is the commonly used trade-off. The information retrieval system often needs to trade-off for precision or vice versa. F-score is defined as harmonic mean of recall or precision as follows −

F-score = recall x precision / (recall + precision) / 2

Web Data Mining

Web Data Mining

The term **Web Data Mining** is a technique used to crawl through various web resources to collect required information, which enables an individual or a company to promote business, understanding marketing dynamics, new promotions floating on the Internet, etc. There is a growing trend among companies, organizations and individuals alike to gather information through web data mining to utilize that information in their best interest.

Data Mining is done through various types of data mining software. These can be simple data mining software or highly specific for detailed and extensive tasks that will be sifting through more information to pick out finer bits of information. For example, if a company is looking for information on doctors including their emails, fax, telephone, location, etc., this information can be mined through one of these data mining software programs. This information collection through data mining has allowed companies to make thousands and thousands of dollars in revenues by being able to better use the internet to gain business intelligence that helps companies make vital business decisions.

Before this data mining software came into being, different businesses used to collect information from recorded data sources. But the bulk of this information is too much too daunting and time consuming to gather by going through all the records, therefore the approach of computer based data mining came into being and has gained huge popularity to now become a necessity for the survival of most businesses.

This collected information is used to gain more knowledge and based on the findings and analysis of the information make predictions as to what would be the best choice and the right approach to move toward on a particular issue. Web data mining is not only focused to gain business information but is also used by various organizational departments to make the right predictions and decisions for things like business development, work flow, production processes and more by going through the business models derived from the data mining.

A strategic analysis department can undermine their client archives with data mining software to determine what offers they need to send to what clients for maximum conversions rates. For example, a company is thinking about launching cotton shirts as their new product. Through their client database, they can clearly determine as to how many clients have placed orders for cotton shirts over the last year and how much revenue such orders have brought to the company.

After having a hold on such analysis, the company can make their decisions about which offers to send both to those clients who had placed orders on the cotton shirts and those who had not. This makes sure that the organization heads in the right direction in their marketing and not goes through a trial and error phase to learn the hard facts by spending money needlessly. These analytical facts also shed light as to what the percentage of customers is who can move from your company to your competitor.

The data mining also empowers companies to keep a record of fraudulent payments which can all be researched and studied through data mining. This information can help develop more advanced and protective methods that can be undertaken to prevent such events from happening. Buying trends shown through web data mining can help you to make forecast on your inventories as well. This is a direct

analysis, which will empower the organization to fill in their stocks appropriately for each month depending on the predictions they have laid out through this analysis of buying trends.

The data mining technology is going through a huge evolution and new and better techniques are made available all the time to gather whatever information is required. Web data mining technology is opening avenues on not just gathering data but it is also raising a lot of concerns related to data security. There is loads of personal information available on the internet and web data mining had helped to keep the idea of the need to secure that information at the forefront.

Search Engines.

Search Engine refers to a huge database of internet resources such as web pages, newsgroups, programs, images etc. It helps to locate information on World Wide Web.

User can search for any information by passing query in form of keywords or phrase. It then searches for relevant information in its database and return to the user.

Search Engine Components

Generally there are three basic components of a search engine as listed below:

1. Web Crawler
2. Database
3. Search Interfaces

Web crawler

It is also known as **spider** or **bots.** It is a software component that traverses the web to gather information.

Database

All the information on the web is stored in database. It consists of huge web resources.

Search Interfaces

This component is an interface between user and the database. It helps the user to search through the database.

Search Engine Working

Web crawler, database and the search interface are the major component of a search engine that actually makes search engine to work. Search engines make use of Boolean expression AND, OR, NOT to restrict and widen the results of a search. Following are the steps that are performed by the search engine:

- The search engine looks for the keyword in the index for predefined database instead of going directly to the web to search for the keyword.
- It then uses software to search for the information in the database. This software component is known as web crawler.
- Once web crawler finds the pages, the search engine then shows the relevant web pages as a result. These retrieved web pages generally include title of page, size of text portion, first several sentences etc.

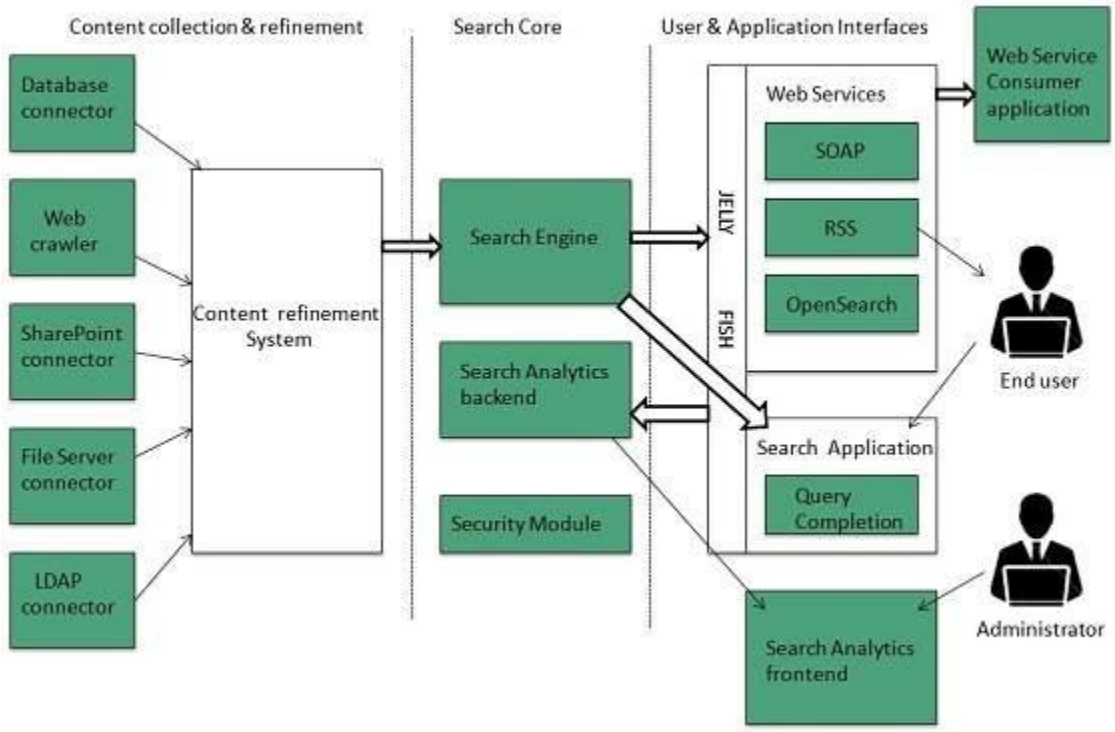
These search criteria may vary from one search engine to the other. The retrieved information is ranked according to various factors such as frequency of keywords, relevancy of information, links etc.

- User can click on any of the search results to open it.

Architecture

The search engine architecture comprises of the three basic layers listed below:

- Content collection and refinement.
- Search core
- User and application interfaces



Search Engine Processing

Indexing Process

Indexing process comprises of the following three tasks:

- Text acquisition
- Text transformation
- Index creation

Text acquisition

It identifies and stores documents for indexing.

Text Transformation

It transforms document into index terms or features.

Index Creation

It takes index terms created by text transformations and create data structures to suport fast searching.

Query Process

Query process comprises of the following three tasks:

- User interaction
- Ranking
- Evaluation

User interaction

It supporst creation and refinement of user query and displays the results.

Ranking

It uses query and indexes to create ranked list of documents.

Evaluation

It monitors and measures the effectiveness and efficiency. It is done offline.

Examples

Following are the several search engines available today:

Search Engine	Description
Google	It was originally called **BackRub.** It is the most popular search engine globally.
Bing	It was launched in 2009 by **Microsoft.** It is the latest web-based search engine that also delivers Yahoo's results.
Ask	It was launched in 1996 and was originally known as **Ask Jeeves.** It includes support for match, dictionary, and conversation question.
AltaVista	It was launched by **Digital Equipment Corporation** in 1995. Since 2003, it is powered by Yahoo technology.
AOL.Search	It is powered by Google.
LYCOS	It is top 5 internet portal and 13th largest online property according to Media Matrix.
Alexa	It is subsidiary of Amazon and used for providing website traffic information.

Possible Questions

PART – A (20 Marks)

(Q.No 1 to 20 Online Examinations)

PART – B (2 Marks)

1. What is multimedia database?
2. Define : Search engine
3. What is the use of time series data?
4. What is web data mining?
5. Define :Sequence data

PART – C (6 Marks)

1. Explain about Multidimensional analysis of multimedia data.
2. Discuss in detail about text data analysis and information retrieval.
3. Explain about similarity search in time series analysis.
4. Discuss in detail about text mining approaches.
5. Discuss in detail spatial classification and spatial trend analysis.
6. Explain dimensionality reduction with text databases.
7. Discuss in detail about Spatial clustering methods.
8. Explicate the association mining in multimedia data.
9. Explain the followings
   a. Spatial data cube construction.
   b. Spatial OLAP.
   c. Spatial classification.
10. Discuss in detail about web page layout structure.

Karpagam Academy of Higher Education

Coimbatore - 641021

(For the candidates admitted from 2017 onwards)

Department of CS,CA & IT

UNIT - IV:(Objective Type/Multiple Choice Questions Carries one mark)

Data mining and Data Warehousing 16CAP504D

UNIT -4						
Q.No	**Questions**	**Option1**	**Option2**	**Option3**	**Option4**	**Answer**
1	_____stores a large amount of spatial related data	Temporal DB	Spatial DB	Relational DB	None	Spatial DB
2	____refers to the extraction of knowledge and spatial relationship	Spatial data mining	Temporal data mining	Textual data mining	None	Spatial data mining
3	A spatial data warehouse is____	Subject oriented	Integrated	Time variant	All the above	All the above
4	_____contains only nonspatial data	nonspatial dimension	spatial dimension	temporal diamension	None	spatial dimension
5	_____is a dimension whose primitives level and all of its high	Spatial to single	Spatial to spatial	Single to single	Single to spatial	Spatial to spatial
6	_____ are frequently used by retail stores to assist in marketing, advertising,….	Classification	Association rules	Clustering	Neural Networks	Association rules
7	_____ are used to show the relationship between data items.	Classification	Association rules	Clustering	Neural Networks	Association rules
8	A _____ is an itemset whose number of occurrences is above a threshold.	Large Itemset	Small Itemset	Equal Itemset	least Update	Large Itemset
9	The _____ is the most well known association rule algorithm.	Priori Algorithm	Posteriori Algorithm	Apriori Algorithm	Aposteriori Algorithm	Apriori Algorithm

10	The basic idea of the _____ is to generate candidate itemsets of a particular size and then scan the database to count these to see if they are large.	Priori Algorithm	Posteriori Algorithm	Apriori Algorithm	Aposteriori Algorithm	Apriori Algorithm
11	The itemset in sampling algorithm is viewed as ____ itemsets.	Potentially Small	Potentially Large	Potentially Equal	None	Potentially Large
12	The candidates in the sampling algorithm is determined by applying the _____ function.	Positive Border	Equal Border	Largest Border	Negative Border	Negative Border
13	_____stores and manages a large collection of multimedia data such as audio, video and etc.	Multimedia DB	Media DB	Relational DB	Non Relational DB	Multimedia DB
14	In____approach the signature of an image includes color histograms.	Color histogram	Multifeature composed	Wavletbased	Wavlet based signature and region based granularity	Color histogram
15	In ___approach the signature ofan image includes a composition of multiple features	Color histogram	Multifeature composed	Wavletbased	Wavlet based signature and region based granularity	Multifeature composed
16	_____uses the dominant wavelet coefficients of an image as its signature	Color histogram	Multifeature composed	Wavletbased	Wavlet based signature and region based granularity	Wavletbased
17	A_____can contain additional dimensions and measures for multimedia information	Multimedia data cube	Temporal data mining	Temporal data cube	Data cube	Multimedia data cube
18	A _____ is a maximal graph in which there is an edge between any two vertices.	Cluster	Dendrogram	Clique	None	Clique

19	The _____ link technique merges two clusters if the average distance between any two points in the two target clusters is below the distance threshold.	Single	Complete	Multiple	Average	Average
20	In _____ clustering, all items are initially placed in one cluster.	Divisive	Partitional	Agglomerative	Hierarchical	Divisive
21	Partitional clustering is otherwise called as _____ .	Hierarchical	Non-Hierarchical	Divisive	Agglomerative	Non-Hierarchical
22	In _____ clustering only one set of clusters may be created internally within the various algorithms.	Hierarchical	Partitional	Divisive	Agglomerative	Partitional
23	The common measure is _____ which measures the squared distance from each point to the centroid for the associated cluster.	Squared error metric	Non Squared error Metric	metroid	chi square	Squared error metric
24	The time complexity of K-means is _____.	O(n)	$O(n^2)$	O(kn)	O(tkn)	O(tkn)
25	An algorithm similar to the single link technique is called as _____.	Nearest Neighbor algorithm	Multiple link algorithm	Farthest neighbor algorithm	Mutual link algorithm	Nearest Neighbor algorithm
26	Which one is the correct text retrival method	Document selection	Document Ranking	Boolean Retrivalmodel	All the above	All the above
27	_____methods the query is regarded as specifying constraints for selecting relavant documents	Document selection	Document Ranking	Boolean Retrivalmodel	All the above	Document selection
28	_____method is used to rank all documents	Document selection	Document view	Boolean Retrivalmodel	None	None
29	Which one is the correct indexing techniques	Inverted indicies	Signature file	All of the above	None of the above	All of the above
30	An _____ is a network of perceptrons	multisection	multiframe	multilayer	multiscene	multilayer

31	_____ theorem states that a mapping between two sets of numbers can be performed using an neural network with only one hidden layer.	boolean	gaussian	bayes	kolmogorov	kolmogorov
32	A _____ rule consists of if part and then part	classification	association	bayes	incremental	classification
33	The if part in classification rule is called as _____	antecedent	consequent	preposition	composition	antecedent
34	The then part in classification rule is called as _____	antecedent	consequent	preposition	composition	consequent
35	The _____ algorithm is to cluster output values with the associated hidden nodes and input	genetic algorithm	expectation maximization	RX	crossover	RX
36	_____ algorithm attempt to generate rules exactly cover a specific class	genetic algorithm	covering	expectation maximization	RX	covering
37	A _____ of approaches takes multiple techniques and blemd them into a new approach	combined	semantic	analysis	synthesis	synthesis
38	One simple approach is called _____ which generates a simple set of rules that are equivalent to a DT with only one level	1R	2R	3R	4R	1R
39	_____ algorithm generate a rule for each leaf node in the decision tree	genetic algorithm	gen	expectation maximization	RX	gen
40	DOM Refers to	Document object model	Document opproach model	Document object method	Dataobject model	Document object model
41	LSI Refers to	Latent Semantic item	Latent Semantic Indexing	Linear Semantic Indexing	Latent System Indexing	Latent Semantic Indexing

42	VIPS refers to	Vision based problem system	Vision based page system	Vision based problem segmentation	Vision based page segmentation	Vision based page segmentation
43	_____ is basec on Apriori algorithm.	First UPdate	Fast UPdate	Firm UPdate	least Update	Fast UPdate
44	A generalized association rule, _____ is defined like regular association rule the restriction that no item in Y may be above any item in X.	X=Y	X.Y	X!=Y	X=> Y	X=> Y
45	A _____, X=> Y is defined like regular association rule the restriction that no item in Y may be above any item in X.	Generalized association rule	Increemental rule	Multiple level association rule	single level association rule	Generalized association rule
46	A variation of generalized rules are _____.	Generalized association rule	Increemental rule	Multiple level association rule	single level association rule	Multiple level association rule
47	When large itemsets are found at level I, large itemsets are generated for level _____.	i-1	i=1	i<1	i+1	i+1
48	In ____ where the input is a set of keyword or terms in the document	Keyboard based approach	Tagging approach	Information extraction approach	None	Keyboard based approach
49	In____where the input is set of tags	Keyboard based approach	Tagging approach	Information extraction approach	None	Tagging approach
50	____consists of set semantic information as input	Keyboard based approach	Tagging approach	Information extraction approach	None	Information extraction approach

UNIT V

Data Warehouse and OLAP Technology for Data Mining. What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology, Data Ware housing to Data Mining Data Preprocessing Data Warehousing: Failures of past Decision Support System- Operational vs. DSS- Building blocks: features- Data warehouse and Data Mart- Overview of the Components- Metadata Architectural Components: Distinguishing Characteristics- Architectural Framework- Technical Architecture.

Data Warehousing - Overview

The term "Data Warehouse" was first coined by Bill Inmon in 1990. According to Inmon, a data warehouse is a subject oriented, integrated, time-variant, and non-volatile collection of data. This data helps analysts to take informed decisions in an organization.

An operational database undergoes frequent changes on a daily basis on account of the transactions that take place. Suppose a business executive wants to analyze previous feedback on any data such as a product, a supplier, or any consumer data, then the executive will have no data available to analyze because the previous data has been updated due to transactions.

A data warehouses provides us generalized and consolidated data in multidimensional view. Along with generalized and consolidated view of data, a data warehouses also provides us Online Analytical Processing (OLAP) tools. These tools help us in interactive and effective analysis of data in a multidimensional space. This analysis results in data generalization and data mining.

Data mining functions such as association, clustering, classification, prediction can be integrated with OLAP operations to enhance the interactive mining of knowledge at multiple level of abstraction. That's why data warehouse has now become an important platform for data analysis and online analytical processing.

Understanding a Data Warehouse

A data warehouse is a database, which is kept separate from the organization's operational database.

There is no frequent updating done in a data warehouse.

It possesses consolidated historical data, which helps the organization to analyze its business.

A data warehouse helps executives to organize, understand, and use their data to take strategic decisions.

Data warehouse systems help in the integration of diversity of application systems.

A data warehouse system helps in consolidated historical data analysis.

Why a Data Warehouse is Separated from Operational Databases

A data warehouses is kept separate from operational databases due to the following reasons:

An operational database is constructed for well-known tasks and workloads such as searching particular records, indexing, etc. In contract, data warehouse queries are often complex and they present a general form of data.

Operational databases support concurrent processing of multiple transactions. Concurrency control and recovery mechanisms are required for operational databases to ensure robustness and consistency of the database.

An operational database query allows reading and modifying operations, while an OLAP query needs only **read only** access of stored data.

An operational database maintains current data. On the other hand, a data warehouse maintains historical data.

Data Warehouse Features

The key features of a data warehouse are discussed below:

Subject Oriented - A data warehouse is subject oriented because it provides information around a subject rather than the organization's ongoing operations. These subjects can be product, customers, suppliers, sales, revenue, etc. A data warehouse does not focus on the ongoing operations, rather it focuses on modelling and analysis of data for decision making.

Integrated - A data warehouse is constructed by integrating data from heterogeneous sources such as relational databases, flat files, etc. This integration enhances the effective analysis of data.

Time Variant - The data collected in a data warehouse is identified with a particular time period. The data in a data warehouse provides information from the historical point of view.

Non-volatile - Non-volatile means the previous data is not erased when new data is added to it. A data warehouse is kept separate from the operational database

and therefore frequent changes in operational database is not reflected in the data warehouse.

Data Warehouse Applications

As discussed before, a data warehouse helps business executives to organize, analyze, and use their data for decision making. A data warehouse serves as a sole part of a plan-execute-assess "closed-loop" feedback system for the enterprise management. Data warehouses are widely used in the following fields:

Financial services

Banking services

Consumer goods

Retail sectors

Controlled manufacturing

Types of Data Warehouse

Information processing, analytical processing, and data mining are the three types of data warehouse applications that are discussed below:

Information Processing - A data warehouse allows processing the data stored in it. The data can be processed by means of querying, basic statistical analysis, reporting using crosstabs, tables, charts, or graphs.

Analytical Processing - A data warehouse supports analytical processing of the information stored in it. The data can be analyzed by means of basic OLAP operations, including slice-and-dice, drill down, drill up, and pivoting.

Data Mining - Data mining supports knowledge discovery by finding hidden patterns and associations, constructing analytical models, performing classification and prediction. These mining results can be presented using the visualization tools.

Given below are the issues to be taken into account while determining the functional split:

The structure of the department may change.

The products might switch from one department to other.

The merchant could query the sales trend of other products to analyze what is happening to the sales.

Identify User Access Tool Requirements

We need data marts to support **user access tools** that require internal data structures. The data in such structures are outside the control of data warehouse but need to be populated and updated on a regular basis.

There are some tools that populate directly from the source system but some cannot. Therefore additional requirements outside the scope of the tool are needed to be identified for future.

Identify Access Control Issues

There should to be privacy rules to ensure the data is accessed by authorized users only. For example a data warehouse for retail banking institution ensures that all the accounts

belong to the same legal entity. Privacy laws can force you to totally prevent access to information that is not owned by the specific bank.

Data marts allow us to build a complete wall by physically separating data segments within the data warehouse. To avoid possible privacy problems, the detailed data can be removed from the data warehouse. We can create data mart for each legal entity and load it via data warehouse, with detailed account data.

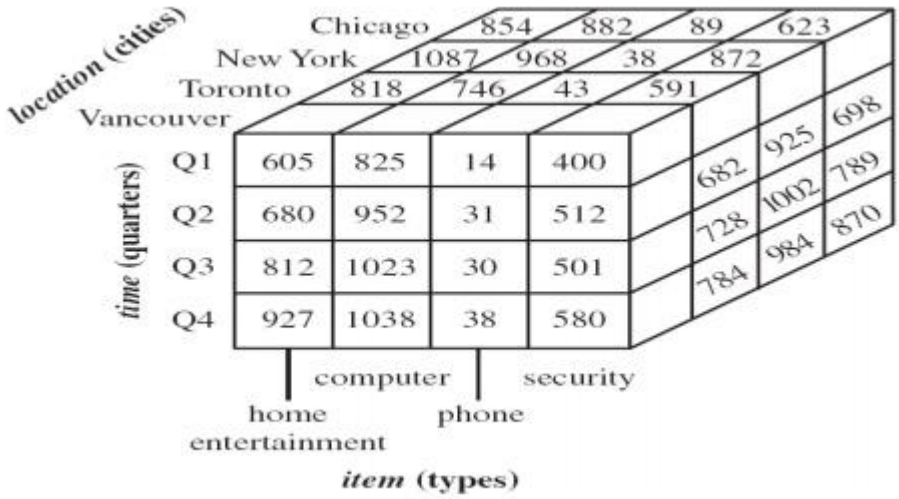
Multidimensional Data Model

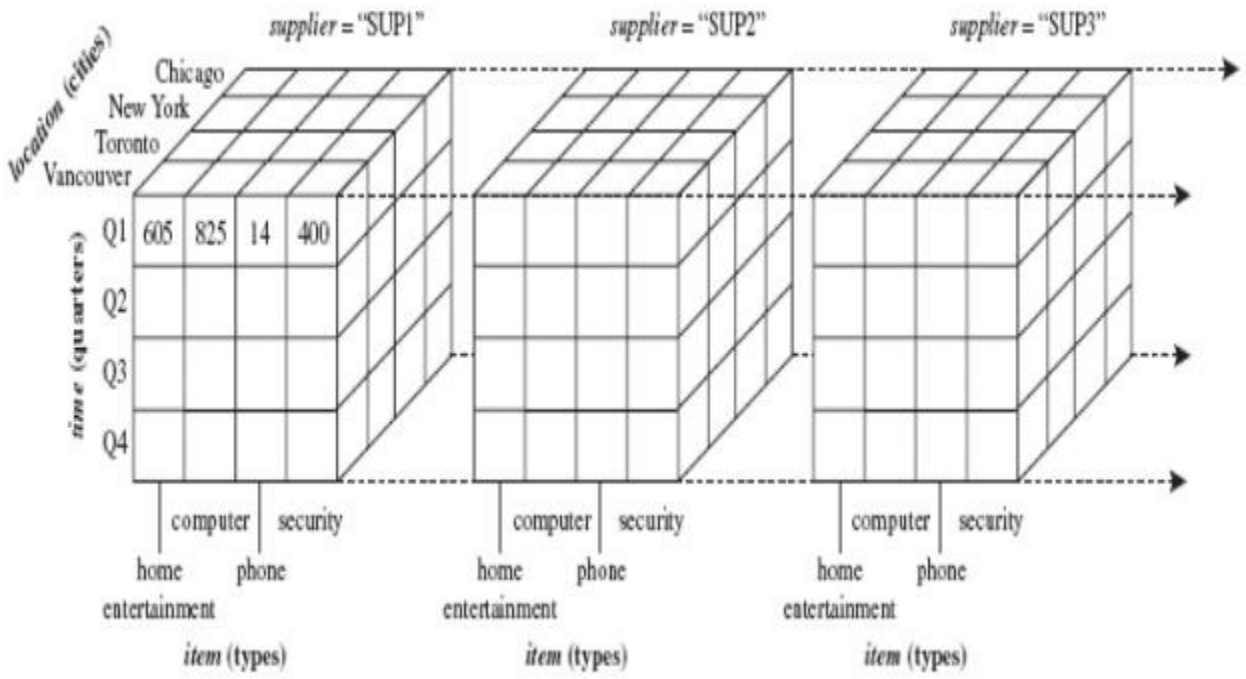
The multidimensional data model is an integral part of On-Line Analytical Processing, or OLAP

Multidimensional data model is to view it as a cube. The cable at the left contains detailed sales data by product, market and time. The cube on the right associates sales number (unit sold) with dimensions-product type, market and time with the unit variables organized as cell in an array.

This cube can be expended to include another array-price-which can be associates with all or only some dimensions. As number of dimensions increases number of cubes cell increase exponentially.

Dimensions are hierarchical in nature i.e. time dimension may contain hierarchies for years, quarters, months, weak and day. GEOGRAPHY may contain country, state, city etc.

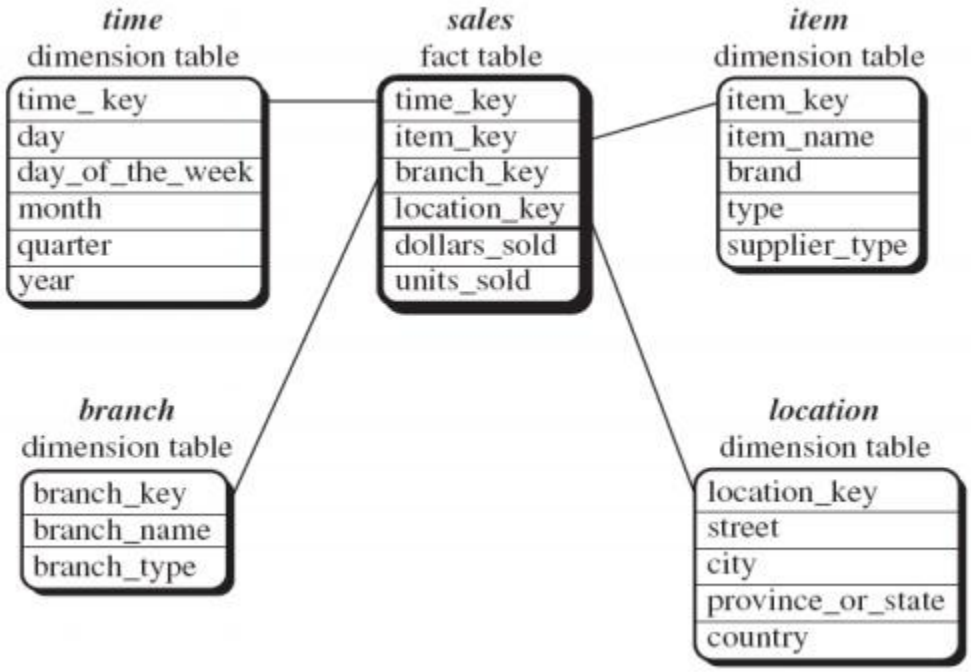




Star schema: A fact table in the middle connected to a set of dimension tables It contains:

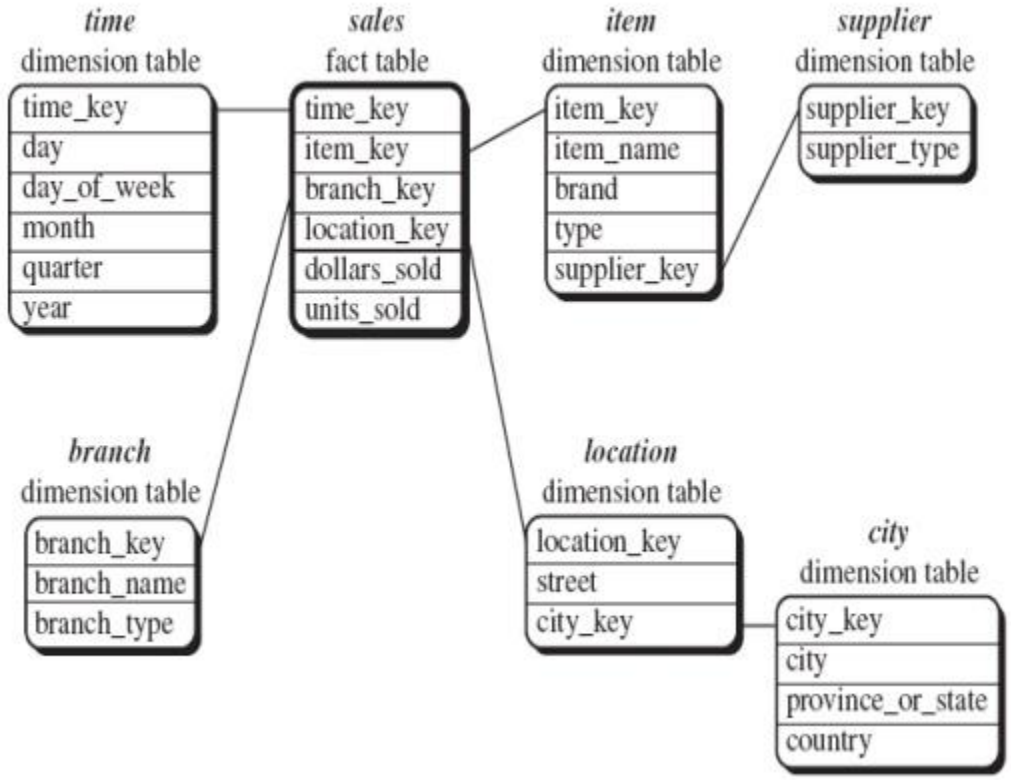
A large central table (fact table)

A set of smaller attendant tables (dimension table), one for each dimension

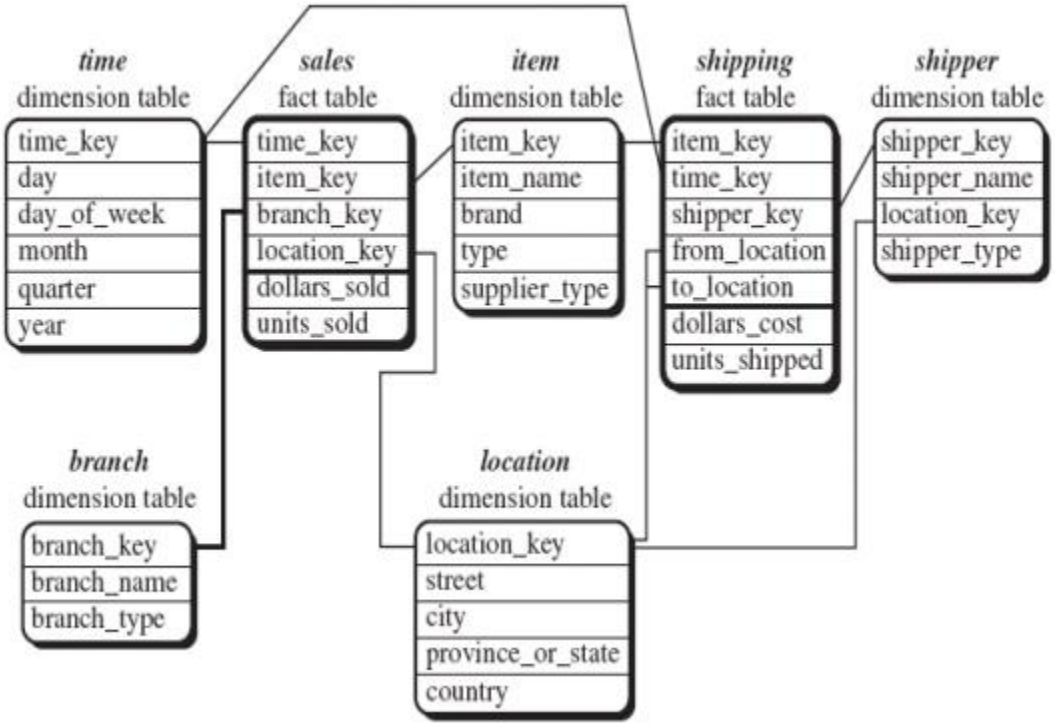


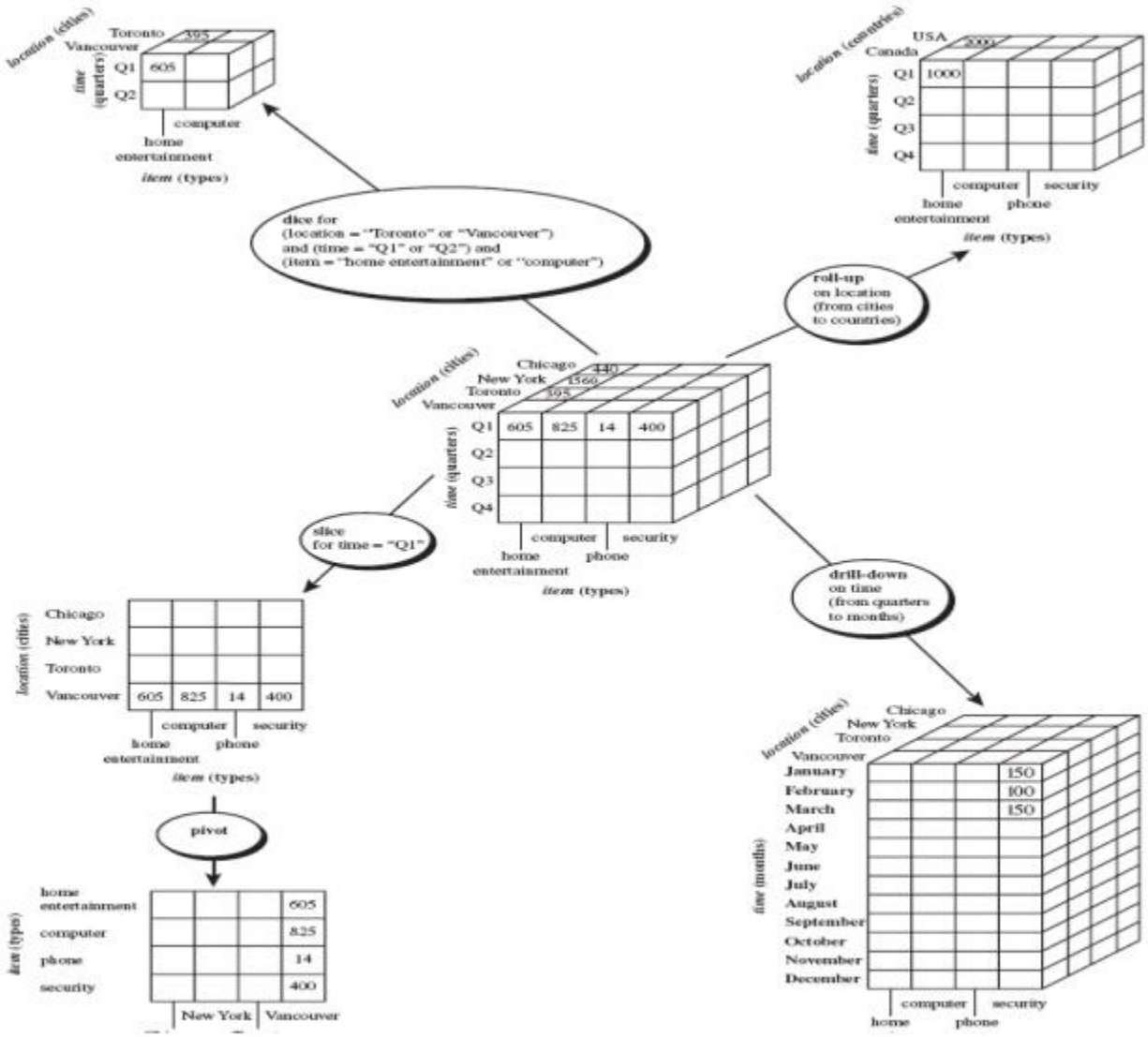
Snowflake schema: A refinement of star schema where some dimensional hierarchy is further splitting (normalized) into a set of smaller dimension tables, forming a shape similar to snowflake

However, the snowflake structure can reduce the effectiveness of browsing, since more joins will be needed



Fact constellations: Multiple fact tables share dimension tables, viewed as a collection of stars, therefore called galaxy schema or fact constellation


time
dimension table
time_key
day
day_of_week
month
quarter
year
sales
fact table
time_key
item_key
branch_key
location_key
dollars_sold
units_sold
item
dimension table
item_key
item_name
brand
type
supplier_type
shipping
fact table
item_key
time_key
shipper_key
from_location
to_location
dollars_cost
units_shipped
shipper
dimension table
shipper_key
shipper_name
location_key
shipper_type
branch
dimension table
branch_key
branch_name
branch_type
location
dimension table
location_key
street
city
province_or_state
country



In this cube we can observe, that each side of the cube represents one of the elements of the question. The x-axis represents the time, the y-axis represents the products and the z-axis represents different centers. The cells of in the cube represents the number of product sold or can represent the price of the items.

This Figure also gives a different understanding to the drilling down operations. The relations defined must not be directly related, they related directly.

The size of the dimension increase, the size of the cube will also increase exponentially. The time response of the cube depends on the size of the cube.

Operations in Multidimensional Data Model:

Aggregation (*roll-up*)

dimension reduction: e.g., total sales by city

summarization over aggregate hierarchy: e.g., total sales by city and year -> total sales by region and by year

Selection (*slice*) defines a subcube

e.g., sales where city = Palo Alto and date = 1/15/96

Navigation to detailed data (*drill-down*)

e.g., (sales – expense) by city, top 3% of cities by average income

Visualization Operations (e.g., Pivot or dice)

Data Warehouse Architecture

Overall Architecture

The data warehouse architecture is based on a relational database management system server that functions as the central repository for informational data. Operational data and processing is completely separated from data warehouse processing. This central information repository is surrounded by a number of key components designed to make the

entire environment functional, manageable and accessible by both the operational systems that source data into the warehouse and by end-user query and analysis tools.

Typically, the source data for the warehouse is coming from the operational applications. As the data enters the warehouse, it is cleaned up and transformed into an integrated structure and format.

The transformation process may involve conversion, summarization, filtering and condensation of data. Because the data contains a historical component, the warehouse must be capable of holding and managing large volumes of data as well as different data structures for the same database over time.

Data Warehouse Database

The central data warehouse database is the cornerstone of the data warehousing environment. This database is almost always implemented on the relational database management system (RDBMS) technology. However, this kind of implementation is often constrained by the fact that traditional RDBMS products are optimized for transactional database processing. Certain data warehouse attributes, such as very large database size, ad hoc query processing and the need for flexible user view creation including aggregates, multi-table joins and drill-downs, have become drivers for different technological approaches to the data warehouse database. These approaches include:

- Parallel relational database designs for scalability that include shared-memory, shared disk, or shared-nothing models implemented on various multiprocessor configurations (symmetric multiprocessors or SMP, massively parallel processors or MPP, and/or clusters of uni- or multiprocessors).
- An innovative approach to speed up a traditional RDBMS by using new index structures to bypass relational table scans.
- Multidimensional databases (MDDBs) that are based on proprietary database technology; conversely, a dimensional data model can be implemented using a familiar RDBMS. Multi-dimensional databases are designed to overcome any limitations placed on the warehouse by the nature of the relational data model. MDDBs enable on-line analytical processing (OLAP) tools that architecturally belong to a group of data warehousing components jointly categorized as the data query, reporting, analysis and mining tools.

Sourcing, Acquisition, Cleanup and Transformation Tools

A significant portion of the implementation effort is spent extracting data from operational systems and putting it in a format suitable for informational applications that run off the data warehouse.

The data sourcing, cleanup, transformation and migration tools perform all of the conversions, summarizations, key changes, structural changes and condensations needed to transform disparate data into information that can be used by the decision support tool. They produce the programs and control statements, including the COBOL programs, MVS job-control language (JCL), UNIX scripts, and SQL data definition language (DDL) needed to move data into the data warehouse for multiple operational systems. These tools also maintain the meta data. The functionality includes:

- Removing unwanted data from operational databases
- Converting to common data names and definitions
- Establishing defaults for missing data
- Accommodating source data definition changes

The data sourcing, cleanup, extract, transformation and migration tools have to deal with some significant issues including:

- Database heterogeneity. DBMSs are very different in data models, data access language, data navigation, operations, concurrency, integrity, recovery etc.
- Data heterogeneity. This is the difference in the way data is defined and used in different models – homonyms, synonyms, unit compatibility (U.S. vs metric), different attributes for the same entity and different ways of modeling the same fact.

These tools can save a considerable amount of time and effort. However, significant shortcomings do exist. For example, many available tools are generally useful for simpler data extracts.

Frequently, customized extract routines need to be developed for the more complicated data extraction procedures.

Meta data

Meta data is data about data that describes the data warehouse. It is used for building, maintaining, managing and using the data warehouse. Meta data can be classified into:

- Technical meta data, which contains information about warehouse data for use by warehouse designers and administrators when carrying out warehouse development and management tasks.
- Business meta data, which contains information that gives users an easy-to-understand perspective of the information stored in the data warehouse.

Equally important, meta data provides interactive access to users to help understand content and find data. One of the issues dealing with meta data relates to the fact that many data extraction tool capabilities to gather meta data remain fairly immature. Therefore, there is often the need to create a meta data interface for users, which may involve some duplication of effort.

Meta data management is provided via a meta data repository and accompanying software. Meta data repository management software, which typically runs on a workstation, can be used to map the source data to the target database; generate code for data transformations; integrate and transform the data; and control moving data to the warehouse.

As user's interactions with the data warehouse increase, their approaches to reviewing the results of their requests for information can be expected to evolve from relatively simple manual analysis for trends and exceptions to agent-driven initiation of the analysis based on user-defined thresholds. The definition of these thresholds, configuration parameters for the software agents using them, and the information directory indicating where the appropriate sources for the information can be found are all stored in the meta data repository as well.

Access Tools

The principal purpose of data warehousing is to provide information to business users for strategic decision-making. These users interact with the data warehouse using front-end tools. Many of these tools require an information specialist, although many end users develop expertise in the tools. Tools fall into four main categories: query and reporting tools, application development tools, online analytical processing tools, and data mining tools.

Query and Reporting tools can be divided into two groups: reporting tools and managed query tools. Reporting tools can be further divided into production reporting tools and report writers.

Production reporting tools let companies generate regular operational reports or support high-volume batch jobs such as calculating and printing paychecks. Report writers, on the other hand, are inexpensive desktop tools designed for end-users.

Managed query tools shield end users from the complexities of SQL and database structures by inserting a metalayer between users and the database. These tools are designed for easy-to-use, point-and-click operations that either accept SQL or generate SQL database queries.

Often, the analytical needs of the data warehouse user community exceed the built-in capabilities of query and reporting tools. In these cases, organizations will often rely on the tried-and-true approach of in-house application development using graphical development environments such as PowerBuilder, Visual Basic and Forte. These application development platforms integrate well with popular OLAP tools and access all major database systems including Oracle, Sybase, and Informix.

OLAP tools are based on the concepts of dimensional data models and corresponding databases, and allow users to analyze the data using elaborate, multidimensional views. Typical business applications include product performance and profitability, effectiveness of a sales program or marketing campaign, sales forecasting and capacity planning. These tools assume that the data is organized in a multidimensional model.

A critical success factor for any business today is the ability to use information effectively. Data mining is the process of discovering meaningful new correlations, patterns and trends by digging into large amounts of data stored in the warehouse using artificial intelligence, statistical and mathematical techniques.

Data Marts

The concept of a data mart is causing a lot of excitement and attracts much attention in the data warehouse industry. Mostly, data marts are presented as an alternative to a data warehouse that takes significantly less time and money to build. However, the term data mart means different things to different people. A rigorous definition of this term is a data store that is subsidiary to a data warehouse of integrated data. The data mart is directed at a partition of data (often called a subject area) that is created for the use of a dedicated group of users. A data mart might, in fact, be a set of denormalized, summarized, or aggregated data. Sometimes, such a set could be placed on the data warehouse rather than a physically separate store of data. In most instances, however, the data mart is a physically separate store of data and is resident on separate database server, often a local area network serving a dedicated user group. Sometimes the data mart simply comprises relational OLAP technology which creates highly denormalized dimensional model (e.g., star schema) implemented on a relational database. The resulting hypercubes of data are

used for analysis by groups of users with a common interest in a limited portion of the database.

These types of data marts, called dependent data marts because their data is sourced from the data warehouse, have a high value because no matter how they are deployed and how many different enabling technologies are used, different users are all accessing the information views derived from the single integrated version of the data.

Unfortunately, the misleading statements about the simplicity and low cost of data marts sometimes result in organizations or vendors incorrectly positioning them as an alternative to the data warehouse. This viewpoint defines independent data marts that in fact, represent fragmented point solutions to a range of business problems in the enterprise. This type of implementation should be rarely deployed in the context of an overall technology or applications architecture. Indeed, it is missing the ingredient that is at the heart of the data warehousing concept — that of data integration. Each independent data mart makes its own assumptions about how to consolidate the data, and the data across several data marts may not be consistent.

Moreover, the concept of an independent data mart is dangerous — as soon as the first data mart is created, other organizations, groups, and subject areas within the enterprise embark on the task of building their own data marts. As a result, you create an environment where multiple operational systems feed multiple non-integrated data marts that are often overlapping in data content, job scheduling, connectivity and management. In other words, you have transformed a complex many-to-one problem of building a data warehouse from operational and external data sources to a many-to-many sourcing and management nightmare.

Data Warehouse Administration and Management

Data warehouses tend to be as much as 4 times as large as related operational databases, reaching terabytes in size depending on how much history needs to be saved. They are not synchronized in real time to the associated operational data but are updated as often as once a day if the application requires it.

In addition, almost all data warehouse products include gateways to transparently access multiple enterprise data sources without having to rewrite applications to interpret and utilize the data.

Furthermore, in a heterogeneous data warehouse environment, the various databases reside on disparate systems, thus requiring inter-networking tools. The need to manage this environment is obvious.

Managing data warehouses includes security and priority management; monitoring updates from the multiple sources; data quality checks; managing and updating meta data; auditing and reporting data warehouse usage and status; purging data; replicating, subsetting and distributing data; backup and recovery and data warehouse storage management.

Information Delivery System

The information delivery component is used to enable the process of subscribing for data warehouse information and having it delivered to one or more destinations according to some user-specified scheduling algorithm. In other words, the information delivery system distributes warehouse-stored data and other information objects to other data warehouses and end-user products such as spreadsheets and local databases. Delivery of information may be based on time of day or on the completion of an external event. The rationale for the delivery systems component is based on the fact that once the data warehouse is installed and operational, its users don't have to be aware of its location and maintenance. All they need is the report or an analytical view of data at a specific point in time. With the proliferation of the Internet and the World Wide Web such a delivery system may leverage the convenience of the Internet by delivering warehouse-enabled information to thousands of end-users via the ubiquitous world wide network.

In fact, the Web is changing the data warehousing landscape since at the very high level the goals of both the Web and data warehousing are the same: easy access to information. The value of data warehousing is maximized when the right information gets into the hands of those individuals who need it, where they need it and they need it most. However, many corporations have struggled with complex client/server systems to give end users the access they need. The issues become even more difficult to resolve when the users are physically remote from the data warehouse location. The Web removes a lot of these issues by giving users universal and relatively inexpensive access to data. Couple this access with the ability to deliver required information on demand and the result is a web-enabled information delivery system that allows users dispersed across continents to perform a sophisticated business-critical analysis and to engage in collective decision-making.

Data Warehouse Implementation

Data Warehouse Load Manager

Load manager performs the operations required to extract and load the data into the database. The size and complexity of a load manager varies between specific solutions from one data warehouse to another.

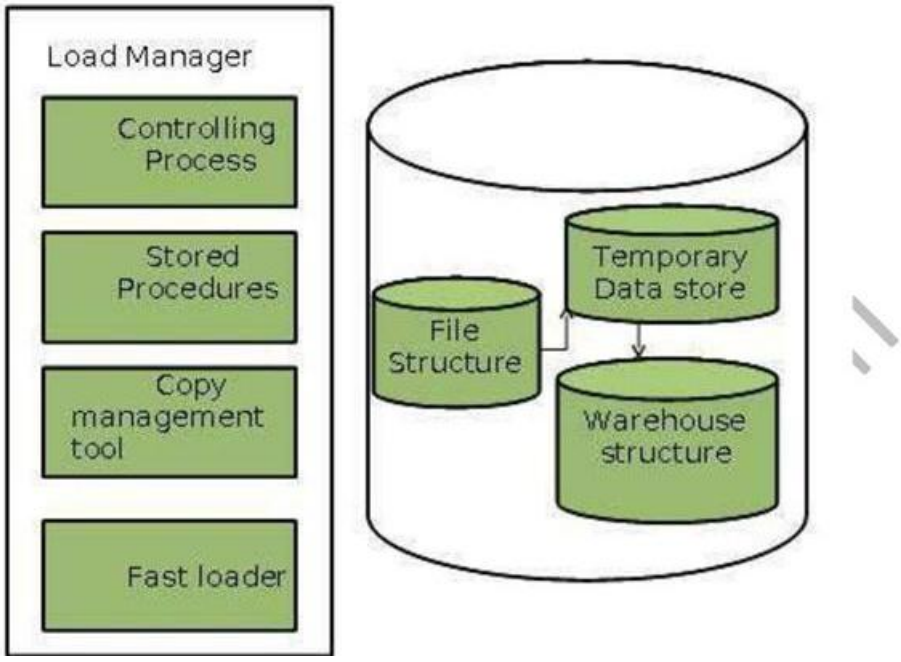
Load Manager Architecture

The load manager does perform the following functions:

Extract data from the source system.

Fast load the extracted data into temporary data store.

Perform simple transformations into structure similar to the one in the data warehouse.



Extract Data from Source

The data is extracted from the operational databases or the external information providers. Gateways are the application programs that are used to extract data. It is supported by underlying DBMS and allows the client program to generate SQL to be executed at a server. Open Database Connection (ODBC) and Java Database Connection (JDBC) are examples of gateway.

Fast Load

In order to minimize the total load window, the data needs to be loaded into the warehouse in the fastest possible time.

Transformations affect the speed of data processing.

It is more effective to load the data into a relational database prior to applying transformations and checks.

Gateway technology is not suitable, since they are inefficient when large data volumes are involved.

Simple Transformations

While loading, it may be required to perform simple transformations. After completing simple transformations, we can do complex checks. Suppose we are loading the EPOS sales transaction, we need to perform the following checks:

Strip out all the columns that are not required within the warehouse.

Convert all the values to required data types.

Warehouse Manager

The warehouse manager is responsible for the warehouse management process. It consists of a third-party system software, C programs, and shell scripts. The size and complexity of a warehouse manager varies between specific solutions.

Warehouse Manager Architecture

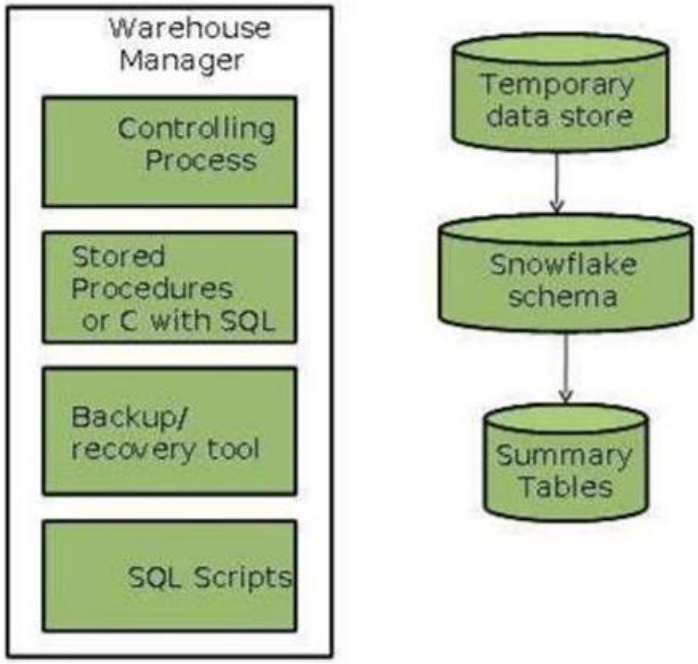
A warehouse manager includes the following:

The controlling process

Stored procedures or C with SQL

Backup/Recovery tool

SQL scripts



Functions of Warehouse Manager

A warehouse manager performs the following functions:

Analyzes the data to perform consistency and referential integrity checks.

Creates indexes, business views, partition views against the base data.

Generates new aggregations and updates the existing aggregations.

Generates normalizations.

Transforms and merges the source data of the temporary store into the published data warehouse.

Backs up the data in the data warehouse.

Archives the data that has reached the end of its captured life.

Query Manager

The query manager is responsible for directing the queries to suitable tables. By directing the queries to appropriate tables, it speeds up the query request and response process. In addition, the query manager is responsible for scheduling the execution of the queries posted by the user.

Query Manager Architecture

A query manager includes the following components:

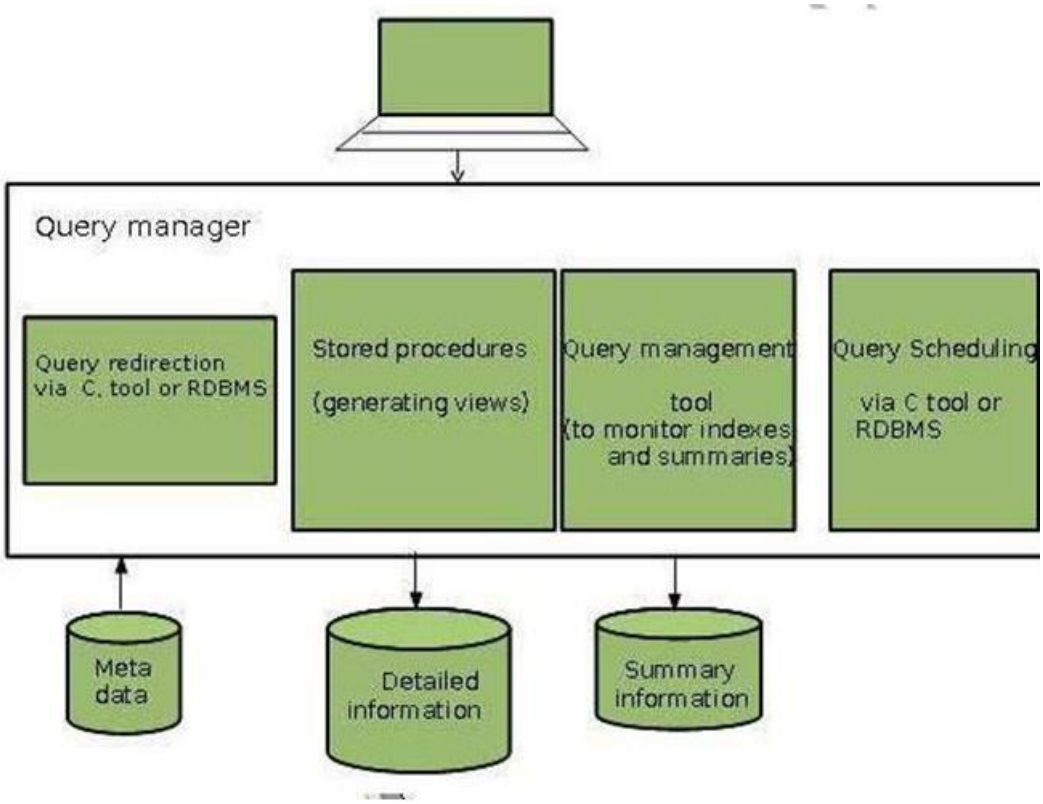
Query redirection via C tool or RDBMS

Stored procedures

Query management tool

Query scheduling via C tool or RDBMS

Query scheduling via third-party software



Functions of Query Manager

It presents the data to the user in a form they understand.

It schedules the execution of the queries posted by the end-user.

It stores query profiles to allow the warehouse manager to determine which indexes and aggregations are appropriate.

Development of Data Cube Technology

Users of decision support systems often see data in the form of ***data cubes***. The cube is used to represent data along some measure of interest. Although called a "cube", it can be 2-dimensional, 3-dimensional, or higher-dimensional. Each dimension represents some attribute in the database and the cells in the data cube represent the measure of interest. For example, they could contain a count for the number of times that attribute combination occurs in the database, or the minimum, maximum, sum or average value of some attribute. Queries are performed on the cube to retrieve decision support information.

Example: We have a database that contains transaction information relating company sales of a part to a customer at a store location. The data cube formed from this database is a 3-dimensional representation, with each cell (*p*,*c*,*s*) of the cube representing a combination of values from *part*, *customer* and *store-location*. A sample data cube for this combination is shown in Figure 1. The contents of each cell is the count of the number of times that specific combination of values occurs together in the database. Cells that appear blank in fact have a value of zero. The cube can then be used to retrieve information within the database about, for example, which store should be given a certain part to sell in order to make the greatest sales.

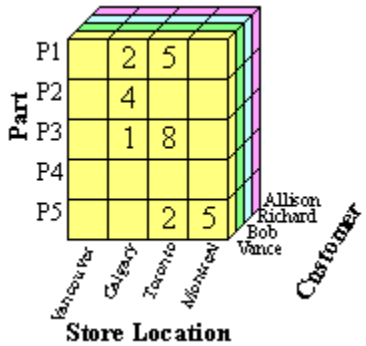


Figure 1(a): Front View of Sample Data Cube

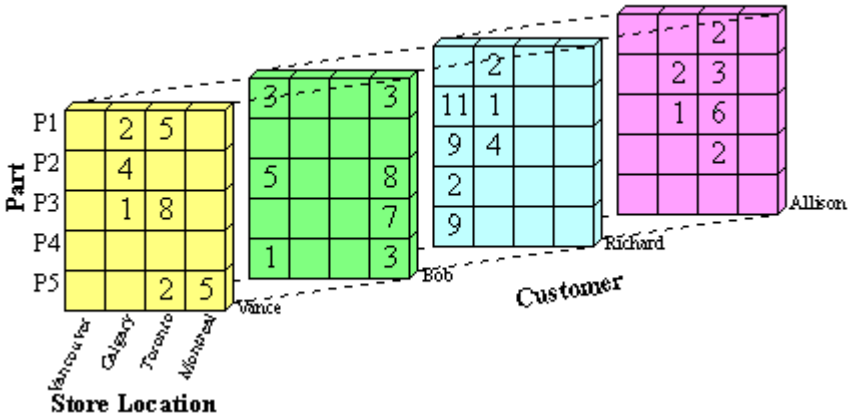


Figure 1(b): Entire View of Sample Data Cube

Computed versus Stored Data Cubes

The goal is to retrieve the decision support information from the data cube in the most efficient way possible. Three possible solutions are:

1. Pre-compute all cells in the cube
2. Pre-compute no cells
3. Pre-compute some of the cells

If the whole cube is pre-computed, then queries run on the cube will be very fast. The disadvantage is that the pre-computed cube requires a lot of memory. The size of a cube for *n* attributes $A_1,...,A_n$ with cardinalities $|A_1|,...,|A_n|$ is $\pi|A_i|$. This size increases exponentially with the number of attributes and linearly with the cardinalities of those attributes.

To minimize memory requirements, we can pre-compute none of the cells in the cube. The disadvantage here is that queries on the cube will run more slowly because the cube will need to be rebuilt for each query.

As a compromise between these two, we can pre-compute only those cells in the cube which will most likely be used for decision support queries. The trade-off between memory space and computing time is called the ***space-time trade-off***, and it often exists in data mining and computer science in general.

Representation

m-Dimensional Array:

A data cube built from *m* attributes can be stored as an *m*-dimensional array. Each element of the array contains the measure value, such as count. The array itself can be represented as a 1-dimensional array. For example, a 2-dimensional array of size *x* x *y* can be stored as a 1-dimensional array of size *x*y*, where element (*i*,*j*) in the 2-D array is stored in location (*y*i+j*) in the 1-D array. The disadvantage of storing the cube directly as an array is that most data cubes are sparse, so the array will contain many empty elements (zero values).

List of Ordered Sets:

To save storage space we can store the cube as a sparse array or a list of ordered sets. If we store all cells in the data cube from Figure 1, then the resulting datacube will contain ($card_{Part} * card_{StoreLocation} * card_{Customer}$) combinations, which is 5 * 4 * 4 = 80 combinations. If we eliminate cells in the cube that contain zero, such as {P1, Vancouver, Allison}, only 27 combinations remain, as seen in Table 1.

Table 1 shows an ***ordered set representation*** of the data cube. Each attribute value combination is paired with its corresponding count. This representation can be easily stored in a database table to facilitate queries on the data cube.

Combination	Count		
{P1, Calgary, Vance}	2		
{P2, Calgary, Vance}	4	{P3, Vancouver, Richard}	9
{P3, Calgary, Vance}	1	{P4, Vancouver, Richard}	2
{P1, Toronto, Vance}	5	{P5, Vancouver, Richard}	9
{P3, Toronto, Vance}	8	{P1, Calgary, Richard}	2
{P5, Toronto, Vance}	2	{P2, Calgary, Richard}	1
{P5, Montreal, Vance}	5	{P3, Calgary, Richard}	4
{P1, Vancouver, Bob}	3	{P2, Calgary, Allison}	2
{P3, Vancouver, Bob}	5	{P3, Calgary, Allison}	1
{P5, Vancouver, Bob}	1	{P1, Toronto, Allison}	2
{P1, Montreal, Bob}	3	{P2, Toronto, Allison}	3
{P3, Montreal, Bob}	8	{P3, Toronto, Allison}	6
{P4, Montreal, Bob}	7	{P4, Toronto, Allison}	2
{P5, Montreal, Bob}	3		
{P2, Vancouver, Richard}	11		

Table 1: Ordered Set Representation of a Data Cube

Representation of Totals

Another aspect of data cube representation which can be considered is the representation of totals. A simple data cube does not contain totals. The storage of totals increases the size of the data cube but can also decrease the time to make total-based queries. A simple way to represent totals is to add an additional layer on *n* sides of the *n*-dimensional datacube. This can be easily visualized with the 3-dimensional data cube introduced in Figure 1. Figure 2 shows the original cube with an additional layer on each of three sides to store total values. The totals represent the sum of all values in one horizontal row, vertical row (column) or depth row of the data cube.

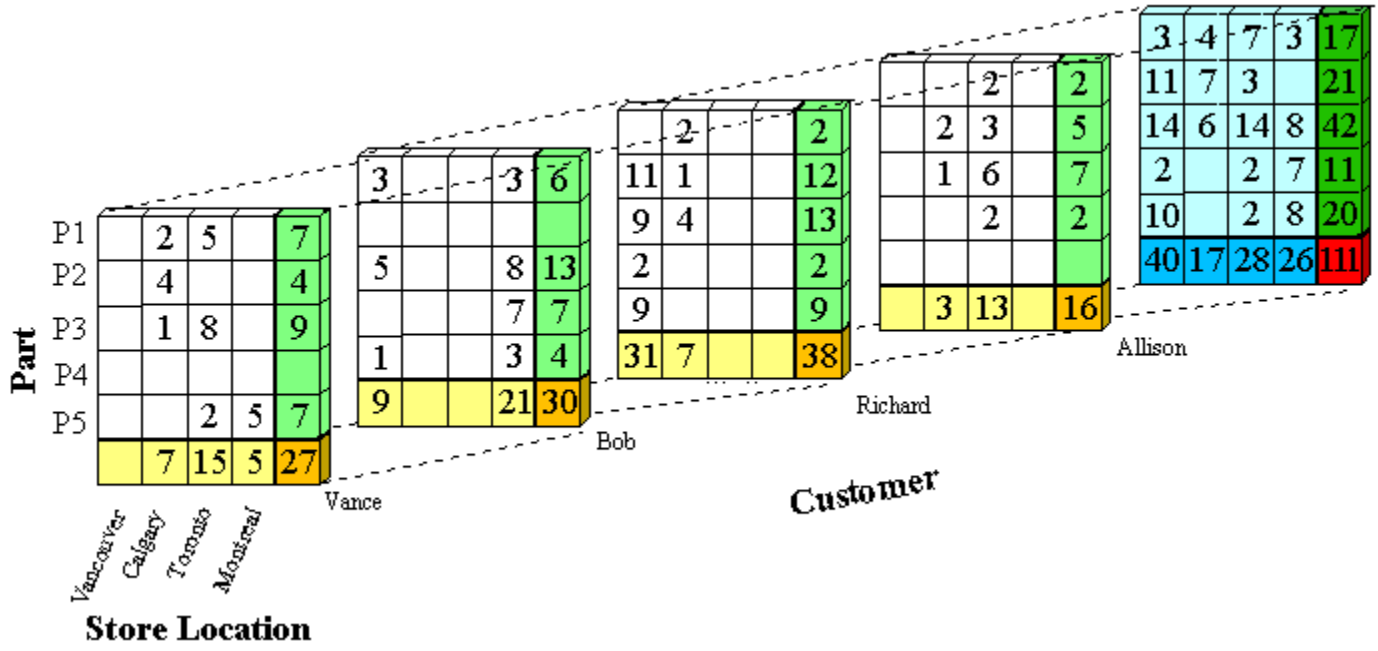


Figure 2: Cube with Totals

The color coding used in Figure 2 is as follows:

- White: Original values
- Light yellow: Total for one customer and one store location
- Light green: Total for one customer and one part
- Light blue: Total for one part and one store location
- Dark yellow: Total for one customer
- Dark green: Total for one part
- Dark blue: Total for one store location
- Red: Total number of transactions in all

To store these totals in ordered set representation the value ***ANY*** can be used. For example, there are 15 transactions where Vance buys a part in Toronto. The ordered set representation of this is ({ANY, Toronto, Vance},15), because it could be any part. The ordered set representation of all of Vance's transactions is ({ANY, ANY, Vance},27), that is all transactions at all store locations for Vance. The total number of transactions in the whole cube is found in the red cell and is 111. This is represented as ({ANY, ANY, ANY}, 111).

Operations on Data Cubes

Summarization or Rollup

Rollup or ***summarization*** of the data cube can be done by traversing upwards through a ***concept hierarchy***. A concept hierarchy maps a set of low level concepts to higher level, more general concepts. It can be used to summarize information in the data cube. As the values are combined, cardinalities shrink and the cube gets smaller. Generalizing can be thought of as computing some of the summary total cells that contain *ANY*s, and storing those in favour of the original cells.

Figures 2 through 8 show an example of summarizing a data cube built from two attributes, Province and Grant_Amount. The measure of interest stored in the cube is Vote. It contains the total number of votes for each combination of Province and Grant_Amount that occurred in the input table. The concept hierarchies for the Province and Grant_Amount attributes are shown in Figure 2. In Figure 2(a), each province represents a location and some can be mapped to more general regions, such as the Prairies or the Maritimes. Those regions can be further mapped to Western Canada and Atlantic Canada. The top level of the hierarchy is "ANY", representing any location. Western and Atlantic Canada are higher level, more general concepts than, for example, Alberta and Nova Scotia. The concept hierarchy in Figure 2(b) represents the Grant_Amount dimension of the database. Grant_Amount is originally stored as specific numbers, such as $34000. The concept hierarchy generalizes the values by grouping them into categories of multiples of $10000, then $20000, and finally including all the amounts into ANY. Ordinarily, concept hierarchies are provided by a domain expert, because then the resulting general concepts will make sense to people familiar with the domain. Concept hierarchies might also be formed automatically by clustering.

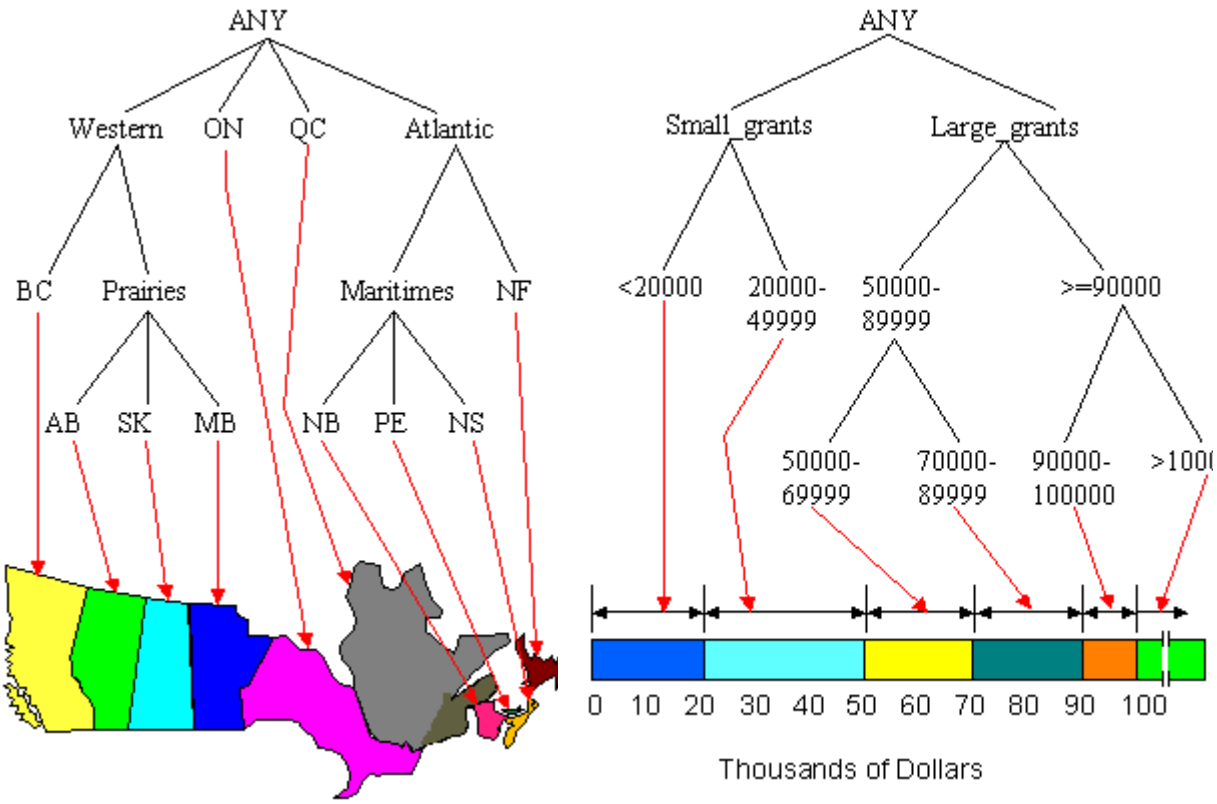


Figure 2(a): Concept Hierarchy for Province

Figure 2(b): Concept Hierarchy for Grant_Amount

To reduce the size of the data cube, we can summarize the data by computing the cube at a higher level in the concept hierarchy. A non-summarized cube would be computed at the lowest level, for example, the province level in Figure 2(a). If we compute the cube at the second level, there are only six categories, B.C., Prairies, Ont., Que., Maritimes and Nfld., and the data cube will be much smaller. Figure 3 shows a sample generalization of the Province attribute for those provinces that can be grouped under the concept Prairies and those that can be grouped under the concept Maritimes. For example, for Sask., the province, or location name, changes to Prairies, but the other attribute values remain unchanged because they are not summarized at this point. The new, summarized concept hierarchy is shown in Figure 4.

Province	Grant_Amount	Votes
B.C.	>100000	4
B.C.	90000-100000	8
B.C.	70000-89999	34
B.C.	50000-69999	41
Ont.	>100000	9
Ont.	90000-100000	13
Ont.	70000-89999	53
Ont.	20000-49999	234
N.B.	20000-49999	38
N.S.	<20000	15
P.E.I.	20000-49999	6
P.E.I.	<20000	65
Alb.	90000-100000	4
Alb.	<20000	21
Man.	<20000	112
Sask.	<20000	98
Que.	>100000	12
Que.	90000-100000	43
Que.	70000-89999	65

Generalize

Maritimes	20000-49999	38
Maritimes	<20000	15
Maritimes	20000-49999	6
Maritimes	<20000	65

Generalize

Prairies	90000-100000	4
Prairies	<20000	21
Prairies	<20000	112
Prairies	<20000	41

Figure 3: Generalizing the Province Attribute

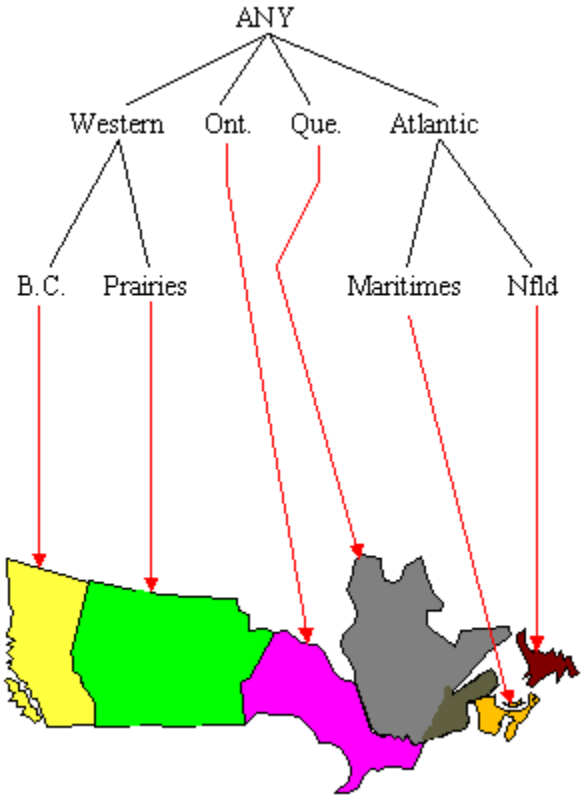


Figure 4: After Generalization of Province Attribute

The next step in the process is to remove duplicate tuples from the data. The measure of interest, Votes, is summed for tuples that have the same Province and Grant_Amount values. For example, in Figure 5, three tuples contain the combination {Prairies, <20000}. The tuples are removed, their individual Votes values are summed, and a new tuple replaces all three, with a Votes value of 231.

Province	Grant_Amount	Votes
B.C.	>=90000	12
B.C.	>=90000	8
B.C.	50000-89999	34
B.C.	50000-89999	41
Ont.	>=90000	9
Ont.	>=90000	13
Ont.	50000-89999	53
Ont.	20000-49999	234
N.B.	20000-49999	38
N.S.	<20000	15
P.E.I.	20000-49999	6
P.E.I.	<20000	65
Prairies	90000-100000	4
Prairies	<20000	21
Prairies	<20000	112
Prairies	<20000	98
Que.	>=90000	12
Que.	>=90000	43
Que.	50000-89999	65

Remove Duplicate Tuples

Prairies	>=90000	4
Prairies	<20000	231

Figure 5: Remove Duplicate Tuples after Province Generalization

We can also generalize the Grant_Amount Attribute. Figure 6 shows summarization of this attribute through the addition of a new category for amounts greater than or equal to 90000. In total, the Grant_Amount value of seven tuples is changed to the new classification. After summarizing the data into groups of <20000, 20000 - 49999, 50000 -89999 and >=90000, the concept hierarchy is as shown in Figure 7. Again, we need to remove the duplicate tuples after generalizating the Grant_Amount attribute. This is shown in Figure 8.

Province	Grant_Amount	Votes
B.C.	>100000	4
B.C.	90000-100000	8
B.C.	70000-89999	34
B.C.	50000-69999	41
Ont.	>100000	9
Ont.	90000-100000	13
Ont.	70000-89999	53
Ont.	20000-49999	234
N.B.	20000-49999	38
N.S.	<20000	15
P.E.I.	20000-49999	6
P.E.I.	<20000	65
Prairies	90000-100000	4
Prairies	<20000	21
Prairies	<20000	112
Prairies	<20000	98
Que.	>100000	12
Que.	90000-100000	43
Que.	70000-89999	65

Generalize

B.C.	>=90000	4
B.C.	>=90000	8

Generalize

Ont.	>=90000	9
Ont.	>=90000	13

Generalize

Prairies	>=90000	4

Generalize

Que.	>=90000	12
Que.	>=90000	43

Figure 6: Generalize Grant_Amount Attribute

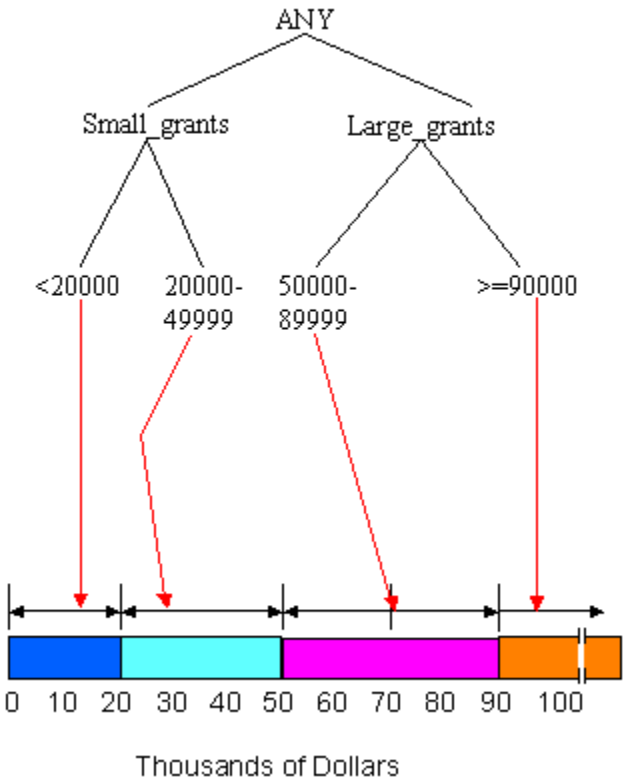


Figure 7: After Generalization of Grant_Amount Attribute

Province	Grant_Amount	Votes
B.C.	>=90000	12
B.C.	>=90000	8
B.C.	50000-89999	34
B.C.	50000-89999	41
Ont.	>=90000	9
Ont.	>=90000	13
Ont.	50000-89999	53
Ont.	20000-49999	234
N.B.	20000-49999	38
N.S.	<20000	15
P.E.I.	20000-49999	6
P.E.I.	<20000	65
Prairies	>=90000	4
Prairies	<20000	231
Que.	>=90000	12
Que.	>=90000	43
Que.	50000-89999	65

Remove Duplicate Tuples

B.C.	>=90000	20
Ont.	>=90000	22
Que.	>=90000	55

Figure 8: Removing Duplicate Tuples after Grant_Amount Generalization

The final result of summarizing the data by introducing the categories Prairies, Maritimes and >=90000 is shown in Figure 9. From here, the process could be continued to further generalize Province or Grant_Amount.

Province	Grant_Amount	Votes
B.C.	>=90000	20
B.C.	50000-89999	75
Ont.	>=90000	22
Ont.	50000-89999	53
Ont.	20000-49999	234
N.B.	20000-49999	38
N.S.	<20000	15
P.E.I.	20000-49999	6
P.E.I.	<20000	65
Prairies	>=90000	4
Prairies	<20000	231
Que.	>=90000	55
Que.	50000-89999	65

Figure 9: Final Result of Generalization

DSS- Building blocks

Decision support systems (DSS) are interactive software-based systems intended to help managers in decision-making by accessing large volumes of information generated from various related information systems involved in organizational business processes, such as office automation system, transaction processing system, etc.

DSS uses the summary information, exceptions, patterns, and trends using the analytical models. A decision support system helps in decision-making but does not necessarily give a decision itself. The decision makers compile useful information from raw data, documents, personal knowledge, and/or business models to identify and solve problems and make decisions.

Programmed and Non-programmed Decisions

There are two types of decisions - programmed and non-programmed decisions.

Programmed decisions are basically automated processes, general routine work, where:

- These decisions have been taken several times.
- These decisions follow some guidelines or rules.

For example, selecting a reorder level for inventories, is a programmed decision.

Non-programmed decisions occur in unusual and non-addressed situations, so:

- It would be a new decision.
- There will not be any rules to follow.
- These decisions are made based on the available information.
- These decisions are based on the manger's discretion, instinct, perception and judgment.

For example, investing in a new technology is a non-programmed decision.

Decision support systems generally involve non-programmed decisions. Therefore, there will be no exact report, content, or format for these systems. Reports are generated on the fly.

Attributes of a DSS

- Adaptability and flexibility
- High level of Interactivity
- Ease of use
- Efficiency and effectiveness
- Complete control by decision-makers
- Ease of development
- Extendibility
- Support for modeling and analysis
- Support for data access
- Standalone, integrated, and Web-based

Characteristics of a DSS

- Support for decision-makers in semi-structured and unstructured problems.
- Support for managers at various managerial levels, ranging from top executive to line managers.
- Support for individuals and groups. Less structured problems often requires the involvement of several individuals from different departments and organization level.
- Support for interdependent or sequential decisions.
- Support for intelligence, design, choice, and implementation.
- Support for variety of decision processes and styles.
- DSSs are adaptive over time.

Benefits of DSS

- Improves efficiency and speed of decision-making activities.
- Increases the control, competitiveness and capability of futuristic decision-making of the organization.
- Facilitates interpersonal communication.
- Encourages learning or training.
- Since it is mostly used in non-programmed decisions, it reveals new approaches and sets up new evidences for an unusual decision.

- Helps automate managerial processes.

Components of a DSS

Following are the components of the Decision Support System:

- **Database Management System (DBMS)**: To solve a problem the necessary data may come from internal or external database. In an organization, internal data are generated by a system such as TPS and MIS. External data come from a variety of sources such as newspapers, online data services, databases (financial, marketing, human resources).
- **Model Management System**: It stores and accesses models that managers use to make decisions. Such models are used for designing manufacturing facility, analyzing the financial health of an organization, forecasting demand of a product or service, etc.

  Support Tools: Support tools like online help; pulls down menus, user interfaces, graphical analysis, error correction mechanism, facilitates the user interactions with the system.

Classification of DSS

There are several ways to classify DSS. Hoi Apple and Whinstone classifies DSS as follows:

- **Text Oriented DSS:** It contains textually represented information that could have a bearing on decision. It allows documents to be electronically created, revised and viewed as needed.
- **Database Oriented DSS:** Database plays a major role here; it contains organized and highly structured data.
- **Spreadsheet Oriented DSS:** It contains information in spread sheets that allows create, view, modify procedural knowledge and also instructs the system to execute self-contained instructions. The most popular tool is Excel and Lotus 1-2-3.
- **Solver Oriented DSS:** It is based on a solver, which is an algorithm or procedure written for performing certain calculations and particular program type.
- **Rules Oriented DSS:** It follows certain procedures adopted as rules.
- **Rules Oriented DSS:** Procedures are adopted in rules oriented DSS. Export system is the example.
- **Compound DSS:** It is built by using two or more of the five structures explained above.

Types of DSS

Following are some typical DSSs:

- **Status Inquiry System:** It helps in taking operational, management level, or middle level management decisions, for example daily schedules of jobs to machines or machines to operators.
- **Data Analysis System:** It needs comparative analysis and makes use of formula or an algorithm, for example cash flow analysis, inventory analysis etc.

- **Information Analysis System:** In this system data is analyzed and the information report is generated. For example, sales analysis, accounts receivable systems, market analysis etc.
- **Accounting System:** It keeps track of accounting and finance related information, for example, final account, accounts receivables, accounts payables, etc. that keep track of the major aspects of the business.
- **Model Based System:** Simulation models or optimization models used for decision-making are used infrequently and creates general guidelines for operation or management.

Data Mart

Why Do We Need a Data Mart?

Listed below are the reasons to create a data mart:

To partition data in order to impose **access control strategies.**

To speed up the queries by reducing the volume of data to be scanned.

- To segment data into different hardware platforms.

To structure data in a form suitable for a user access tool.

Note: Do not data mart for any other reason since the operation cost of data marting could be very high. Before data marting, make sure that data marting strategy is appropriate for your particular solution.

Cost-effective Data Marting

Follow the steps given below to make data marting cost-effective:

Identify the Functional Splits

Identify User Access Tool Requirements

Identify Access Control Issues

Identify the Functional Splits

In this step, we determine if the organization has natural functional splits. We look for departmental splits, and we determine whether the way in which departments use information tend to be in isolation from the rest of the organization. Let's have an example.

Consider a retail organization, where each merchant is accountable for maximizing the sales of a group of products. For this, the following are the valuable information:

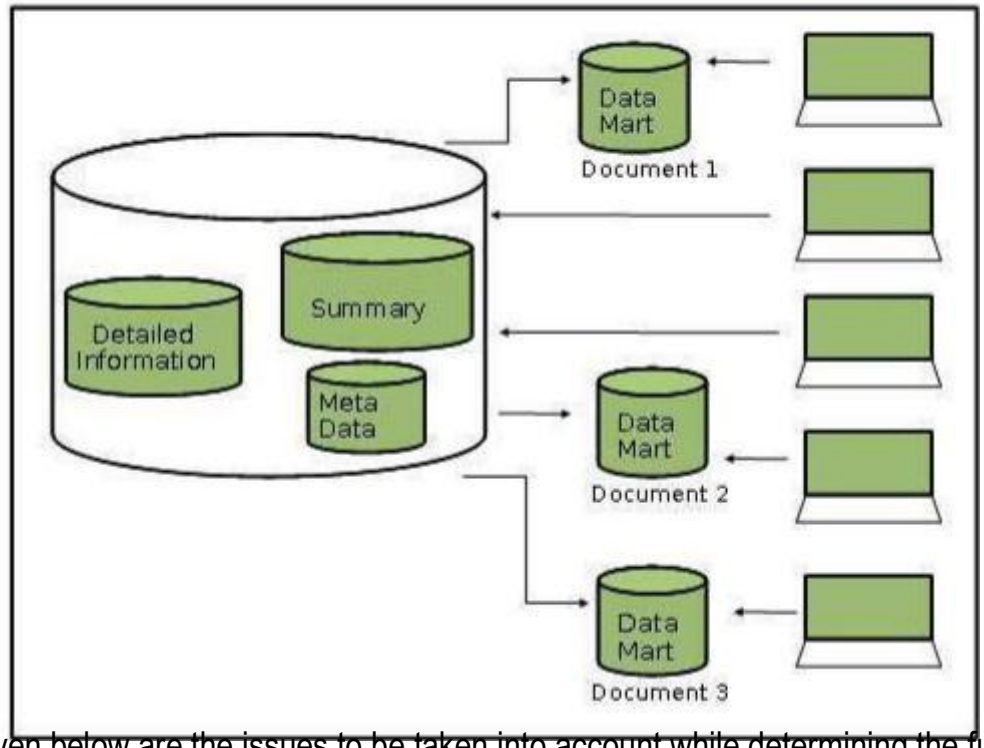
sales transaction on a daily basis

sales forecast on a weekly basis

stock position on a daily basis

stock movements on a daily basis

As the merchant is not interested in the products they are not dealing with, the data marting is a subset of the data dealing which the product group of interest. The following diagram shows data marting for different users.



Given below are the issues to be taken into account while determining the functional split:

The structure of the department may change.

The products might switch from one department to other.

The merchant could query the sales trend of other products to analyze what is happening to the sales.

Note: We need to determine the business benefits and technical feasibility of using a data mart.

Identify User Access Tool Requirements

We need data marts to support **user access tools** that require internal data structures. The data in such structures are outside the control of data warehouse but need to be populated and updated on a regular basis.

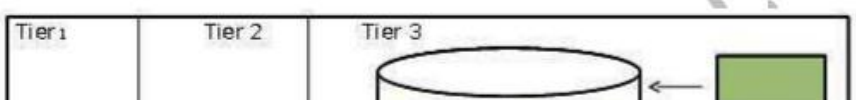
There are some tools that populate directly from the source system but some cannot. Therefore additional requirements outside the scope of the tool are needed to be identified for future.

Identify Access Control Issues

There should to be privacy rules to ensure the data is accessed by authorized users only. For example a data warehouse for retail banking institution ensures that all the accounts belong to the same legal entity. Privacy laws can force you to totally prevent access to information that is not owned by the specific bank.

Data marts allow us to build a complete wall by physically separating data segments within the data warehouse. To avoid possible privacy problems, the detailed data can be removed from the data warehouse. We can create data mart for each legal entity and load it via data warehouse, with detailed account data.

Designing Data Marts



Data marts should be designed as a smaller version of starflake schema within the data warehouse and should match with the database design of the data warehouse. It helps in maintaining control over database instances.

The summaries are data marted in the same way as they would have been designed within the data warehouse. Summary tables help to utilize all dimension data in the starflake schema.

Cost of Data Marting

The cost measures for data marting are as follows:

Hardware and Software Cost

Network Access

Time Window Constraints

Hardware and Software Cost

Although data marts are created on the same hardware, they require some additional hardware and software. To handle user queries, it requires additional processing power and disk storage. If detailed data and the data mart exist within the data warehouse, then we would face additional cost to store and manage replicated data.

Note: Data marting is more expensive than aggregations, therefore it should be used as an additional strategy and not as an alternative strategy.

Network Access

A data mart could be on a different location from the data warehouse, so we should ensure that the LAN or WAN has the capacity to handle the data volumes being transferred within the data mart load process.

Time Window Constraints

The extent to which a data mart loading process will eat into the available time window depends on the complexity of the transformations and the data volumes being shipped. The determination of how many data marts are possible depends on:

Network capacity.

Time window available

Volume of data being transferred

Mechanisms being used to insert data into a data mart

Overview of the Components: Metadata Architectural Components: Architectural Framework, Technical Architecture.

What is Metadata?

Metadata is simply defined as data about data. The data that is used to represent other data is known as metadata. For example, the index of a book serves as a metadata for the contents in the book. In other words, we can say that metadata is the summarized data that leads us to detailed data. In terms of data warehouse, we can define metadata as follows.

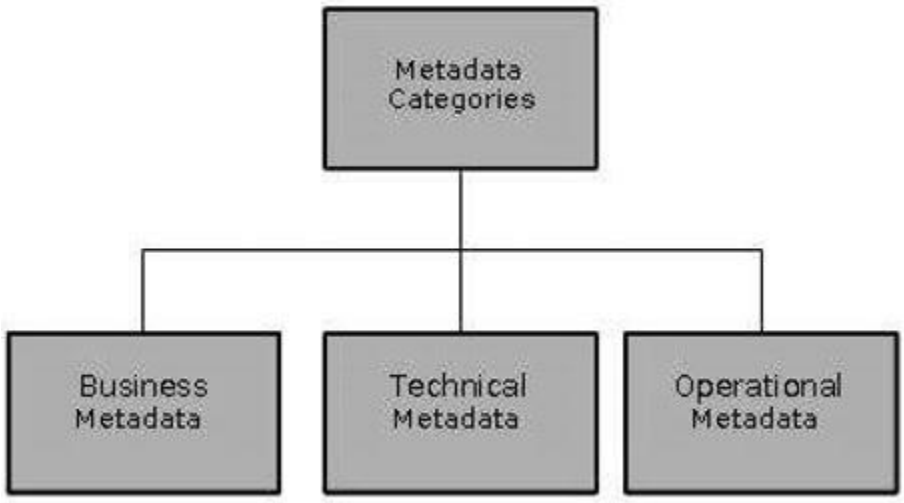
- Metadata is the road-map to a data warehouse.
- Metadata in a data warehouse defines the warehouse objects.
- Metadata acts as a directory. This directory helps the decision support system to locate the contents of a data warehouse.

Note − In a data warehouse, we create metadata for the data names and definitions of a given data warehouse. Along with this metadata, additional metadata is also created for time-stamping any extracted data, the source of extracted data.

Categories of Metadata

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.

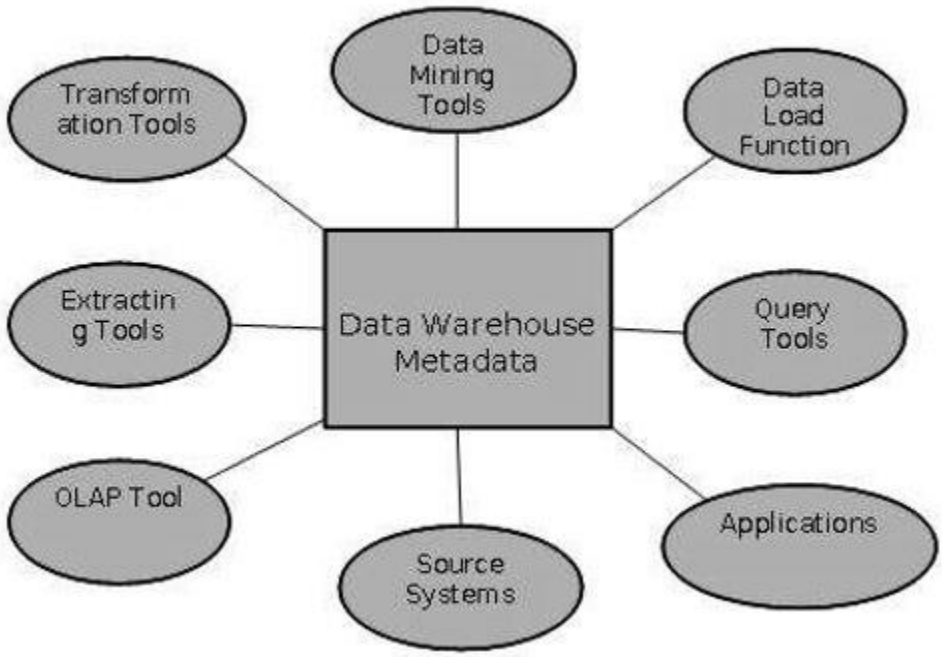


Role of Metadata

Metadata has a very important role in a data warehouse. The role of metadata in a warehouse is different from the warehouse data, yet it plays an important role. The various roles of metadata are explained below.

- Metadata acts as a directory.
- This directory helps the decision support system to locate the contents of the data warehouse.
- Metadata helps in decision support system for mapping of data when data is transformed from operational environment to data warehouse environment.
- Metadata helps in summarization between current detailed data and highly summarized data.
- Metadata also helps in summarization between lightly detailed data and highly summarized data.
- Metadata is used for query tools.
- Metadata is used in extraction and cleansing tools.
- Metadata is used in reporting tools.
- Metadata is used in transformation tools.
- Metadata plays an important role in loading functions.

The following diagram shows the roles of metadata.



Metadata Repository

Metadata repository is an integral part of a data warehouse system. It has the following metadata −

- **Definition of data warehouse** − It includes the description of structure of data warehouse. The description is defined by schema, view, hierarchies, derived data definitions, and data mart locations and contents.
- **Business metadata** − It contains has the data ownership information, business definition, and changing policies.
- **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.
- **Data for mapping from operational environment to data warehouse** − It includes the source databases and their contents, data extraction, data partition cleaning, transformation rules, data refresh and purging rules.
- **Algorithms for summarization** − It includes dimension algorithms, data on granularity, aggregation, summarizing, etc.

Challenges for Metadata Management

The importance of metadata can not be overstated. Metadata helps in driving the accuracy of reports, validates data transformation, and ensures the accuracy of calculations. Metadata also enforces the definition of business terms to business end-users. With all these uses of metadata, it also has its challenges. Some of the challenges are discussed below.

- Metadata in a big organization is scattered across the organization. This metadata is spread in spreadsheets, databases, and applications.
- Metadata could be present in text files or multimedia files. To use this data for information management solutions, it has to be correctly defined.
- There are no industry-wide accepted standards. Data management solution vendors have narrow focus.
- There are no easy and accepted methods of passing metadata.

Possible Questions

PART – A (20 Marks)

(Q.No 1 to 20 Online Examinations)

PART – B (2 Marks)

1. What is data warehouse?
2. Define : OLAP
3. What is data cube?
4. List the failure of decision support system
5. What is data mart?

PART – C (6 Marks)

1. Discuss about failures of past decision support system.
2. Explain the followings
   i) Star Schema.
   ii) Snow flake Schema.
3. Discuss in detail about the development of data cube technology.
4. Explain about the implementation of data warehouse.
5. Discuss about a three tier data warehousing architecture with neat sketch.
6. Explain about technical architecture of data warehouse.
7. Discuss about OLAP operations in the multidimensional data model.
8. Explain in detail about architecture framework.
9. Explain the Types of OLAP servers.
10. Discuss in detail about efficient computing of data cube.

Karpagam Academy of Higher Education

Coimbatore - 641021

(For the candidates admitted from 2017 onwards)

Department of CS,CA & IT

UNIT - V:(Objective Type/Multiple Choice Questions Carries one mark)

Data mining and Data Warehousing 16CAP504D

Q.No	Questions	Option1	Option2	Option3	Option4	Answer
1	_____ takes data from source systems and makes it available to the data warehouse	data cleaning	data abstraction	data extraction	data integration	data extraction
2	clean and _____ the loaded data into a structure that speeds up queries	transform	extract	partition	aggregate	transform
3	A _____ is a collection of data objects that are similar to one another	cluster	outlier	prediction	hashing	cluster
4	The _____ process is the system process that manages the queries and speeds them up by directing queries to the most effective data source	process management	load management	system management	query management	query management
5	The _____ is the system component that performs all the operations necessary to support the extract and load process	process manager	load manager	system manager	query manager	load manager
6	The bulk of the effort to develop a load manager should be planned within the fir _____ phase	analysis	design	load	production	production
7	Meta data describes the _____.	type and format of data	structure of the contents of database	location of data	owner of data	structure of the contents of database

8	In bottom up approach _____ are used by end-users	data mart	large data warehouse	central database	operational database	data mart
9	_____ are used to load the information from the operational database	Statistical Techniques	Visualization Techniques	Windowing mechanism	Replication Techniques	Replication Techniques
10	_____ responses end-users queries in a very short space of time.	Client / Server Technique	Time sharing system	Multiprocessing computer system	Real time system	Multiprocessing computer system
11	Expert system contain _____.	spatial data	knowledge of specialists	knowledge of business logic	transaction records	knowledge of specialists
12	OLAP store their data in _____.	table format	object oriented format	a special multi-dimensional format	points in multi-dimensional space	a special multi-dimensional format
13	OLAP tools	do not learn but create new knowledge	do learn but cannot create new knowledge	do learn and also can create new knowledge	do not learn and cannot create new knowledge	do not learn and cannot create new knowledge
14	Data mining algorithm should not have a complexity that is higher than____.	logn	n^2	nlogn	n	nlogn
15	Association rules are defined on _____.	single attribute	n attributes	binary attributes	data attributes	binary attributes
16	A perceptrons consists of _____.	two layered networks	single layered networks	three layered networks	multiple layered networks	three layered networks
17	Back propagation method gives _____.	answers and idea as to how they arrived at the answers	answers and no clear idea as to how they arrived at the answers	only ideas as to how to obtain answers	answers and poor ideas as to how they arrived at the answers	answers and no clear idea as to how they arrived at the answers
18	A Kohonen's self-organizing map is a collection of _____.	networks	neurons	processors	monitors	neurons
19	Genetic algorithms can be viewed as a kind of _____ strategy.	self learning	meta learning	knowledge discovering	concept learning	meta learning

20	Voroni diagram divide a sample space into _____.	different groups	different divisions	different sub networks	different regions	different regions
21	Neural networks are somewhat better at _____ tasks.	problem solving tasks	classification	knowledge engineering	simple	classification
22	SQL retrieves _____.	hidden knowledge	hidden rules	shallow knowledge	deep knowledge	shallow knowledge
23	_____ can be found by pattern recognition algorithms	Hidden knowledge	Deep knowledge	Encrypted information	Fine grained segmentation	Hidden knowledge
24	_____ give a yes or no answer and no explanation of their responses	Genetic algorithm	Neural network	k-nearest neighbor algorithm	(b) and (c)	(b) and (c)
25	The reporting stage combines _____.	analysis of the results & application of the result to new data	the results & application of the result to new data	analysis of the results & application of the result to existing data	a. analysis of the results & application of the result to new rules	analysis of the results & application of the result to new data
26	The delivery process is staged in order to _____.	reduce the execution time	to minimize error	minimize risk	measure benefits	minimize risk
27	Delivery process is designed to deliver _____.	an enterprise data warehouse	a point solution	maximum information	quality solution	an enterprise data warehouse
28	Delivery process ensures _____.	to reduce the overall delivery time-slice	benefits are delivered incrementally	to reduce the overall delivery time-slice & benefits are delivered incrementally	to reduce investment	to reduce the overall delivery time-slice & benefits are delivered incrementally
29	The technical blueprint phase must deliver an overall architecture that satisfies the _____ requirements.	short term	long term	today's	mid term	long term

30	----------_____ is part of day-to-day management of the data warehouse.	Creating / deleting summaries	Extracting data	Cleaning data	Populating data	Creating / deleting summaries
31	The data extracted from the source systems is loaded into a _____.	data warehouse	data mart	temporary data store	operational database	temporary data store
32	The purpose of business case is to identify the projected _____.	out put structure	business process	business benefits	business process risks	business benefits
33	_____ is a typical example of grid based method	Partitioning	STING	Density based	Model –based	STING
34	SOM stands for	Self-organizing feature map	Simple organizing map	Self oriented map	Summer Opt Machine	Self-organizing feature map
35	_____ is a clustering approach that performs clustering by incorporation of user specified or application oriented constrained	Partitioning	Hierarchical	Density based	Constrained based clustering	Constrained based clustering
36	The cost complexity pruning algorithm used in _____	CART	ID3	splitting rule	printer	CART
37	In linear regression, the n input variables are called _____	generators	response	prediction	predictors	predictors
38	In linear regression, the one output variables are called _____	generators	response	prediction	predictors	response
39	_____ is the problem of determining how much alike the two variables actually are	regression	generators	correlation	predictors	correlation
40	A _____ rule consists of if part and then part	classification	association	bayes	incremental	classification
41	_____ is the count of number of transactions that contain the items in X.	O(X)	$O(n^2)$	O(kn)	O(n)	O(X)
42	In _____ clustering, all items are initially placed in one cluster.	Divisive	Partitional	Agglomerative	Hierarchical	Divisive

43	Partitional clustering is otherwise called as _____ .	Hierarchical	Non-Hierarchical	Divisive	Agglomerative	Non-Hierarchical
44	In _____ clustering only one set of clusters may be created internally within the various algorithms.	Hierarchical	Partitional	Divisive	Agglomerative	Partitional
45	The common measure is _____ which measures the squared distance from each point to the centroid for the associated cluster.	Squared error metric	Non Squared error Metric	metroid	squared	Non Squared error Metric
46	The _____ error clustering algorithm minimizes the squared error.	Squared	Non Squared	metroid	divisive	Squared
47	The squared error for a cluster is the _____ of the squared Euclidean distances between each element in the cluster.	Difference	Multiple	Sum	mean	Sum
48	_____ is an iterative algorithm in which items are moved among sets of clusters until the desired set is reached.	Squared error metric	Partitional	Hierarchical	K- Means	K- Means
49	Back propagation method gives _____.	answers and idea as to how they arrived at the answers	answers and no clear idea as to how they arrived at the answers	only ideas as to how to obtain answers	answers and poor ideas as to how they arrived at the answers	answers and no clear idea as to how they arrived at the answers
50	A Kohonen's self-organizing map is a collection of _____.	networks	neurons	processors	monitors	neurons

Reg No. ……..

[16CAP504D]

KARPAGAM ACADEMY OF HIGHER EDUCATION
(Established Under Section 3 of UGC Act 1956)
COIMBATORE – 64 021
MCA Degree Examination
(For the candidates admitted from 2016 onwards)
Fifth Semester
First Internal Exam August 2018
Data Mining and Data Warehousing

Duration: 2 Hrs **Maximum Marks: 50 Marks**
Date & Session: **Class: III MCA**

Answer Key

Part - A (20 X 1 = 20 Marks)
(Answer all the Questions)

1._____ is to remove noise and inconsistent data.
a. **Data Cleaning** b. Data Mining c. Data Purify d. Data Reduction

2. KDD stands for _____.
a. Knowing Discovery from Data b.**Knowledge Discovery from Data**
c. Knowing Data about Data d. Knight Discovery Database

3._____ is the process of discovering interesting patterns and knowledge from large amounts of data.
a. Data Cleaning b. **Data Mining**
c. Data Purify d. Data Reduction

4._____ is the repository of information collected from multiple sources.
a. Database b. Data Store c. Data warehouse d. **Data Mart**

5. Expand SVM
a. Support vector motor b. Self vector machine
c. **Support vector machine** d. Support vertical machine

6. Tuples is also known as
a) Records b) Rows c) All the above d) None of the above

7. Decision tree is otherwise called as _____ tree
a. clustering **b. classification** c. regression d. correlation

8. Data objects whose class label is known
a. Training characters **b. Training set** c. Training data d. Training objects

9. Summary is also called____
a) Smoothing **b) Aggregation** c) Generalization d) Normalization

10. SQL retrieves _____ knowledge.
a) hidden b)hidden rules **c)shallow** d)deep

11. Expand E-R
a. Entity relation b. Entity record c. Entry relationship
d. Entity relationship

12. Process of remove noise and inconsistent data is referred____
a. Data cleaning b. Data Integration
c. Data Transformation d. Data mining

13._____ is the merging of data from multiple data stores.

a. Cleaning b. Data Mining c. **Data Integration**
d. Data Reduction

14. Which one is the example of data streams
a. Scientific and engineering data b. Banking database
c. Multimedia d.None

15. Which one of the following is discrepancy tool
a. Data auditing tool b. Data smoothing tool
c. Data cleaning tool d.None of the above

16._____ model identifies patterns or relationships in data.
a.Descriptive b. Data mining c. Data Purify d. Data reduction

17._____ process is used to Discover knowledge from Data.
a. KKD b.**KDD** c.KDM d. KMD

18. Data Mining is the process of discovering _____ from large amounts of data.
a. **Interesting Patterns** b. Patents c. model d. interesting samples

19. A special type of clustering is called _____
a. Classification b.Regression c. Prediction d. **Segmentation**

20. Hierarchical clustering and other clustering methods
a. CHAMELEON b. **BIRCH** c. ROCK d. COBWEB

Part - B (3 X 2=6 Marks)
(Answer all the Questions)

21. Define: Data Mining
- Data mining is the process of sorting through large data sets to identify patterns and establish relationships to solve problems through data analysis.

22. What is noise? How do you overcome it?
- Noisy data is data with a large amount of additional meaningless information in it called noise. The term has often been used as a synonym for corrupt data.
- Using binning methods

23. What is classification? Give any two classification algorithms.
- Classification is a data mining function that assigns items in a collection to target categories or classes. The goal of classification is to accurately predict the target class for each case in the data.

K Nearest Neighbors Algorithm
Naïve Bayes Algorithm

Part - C (3 X 8=24 Marks)
(Answer all the Questions)

24. (a) Discuss the major issues in data mining?

Mining methodology and user interaction issues: These reflect the kinds of knowledge mined the ability to mine knowledge at multiple granularities, the use of domain knowledge, ad hoc mining, and knowledge visualization.

- Mining different kinds of knowledge databases: Data mining should cover a wide spectrum of data analysis and knowledge discovery tasks, including data characterization, discrimination, association, classification, clustering, tread and deviation analysis, and similarity analysis.
- Interactive mining of knowledge at multiple levels of abstraction: Incorporation of background knowledge: Data mining query languages and ad hoc mining: Relational query languages

(such as SQL) allow users to pose ad hoc queries for data retrieval.

- Presentation and visualization of data mining results:
- Handling noisy or incomplete data:
- Pattern evaluation--the interestingness problem: A data mining system can uncover thousands of patterns.
- **Performance issues: These include efficiency, scalability, and parallelization of data mining algorithms.**
- Efficiency and scalability of data mining algorithms: To effectively extract information from a huge amount of data in databases, data mining algorithms must be efficient and scalable.
- Parallel, distributed, and incremental mining algorithms: The huge size of many databases, the wide distribution of data, and the computational complexity of some data mining methods are factors motivating the development of algorithms that divide data into partitions that can be processed in parallel.

Issues relating to the diversity of database types:

- Handling of relational and complex types of data: Specific data mining systems should be constructed for mining specific kinds of data.
- Mining information from heterogeneous databases and global information systems: Local- and wide-area computer networks (such as the Internet) connect many sources of data, forming huge, distributed, and heterogeneous databases.

(OR)

(b). Discuss the major components of data mining.

Data Mining Architecture

The major components of any data mining system are data source, data warehouse server, data mining engine, pattern evaluation module, graphical user interface and knowledge base.

a) Data Sources

Database, data warehouse, World Wide Web (WWW), text files and other documents are the actual sources of data. You need large volumes of historical data for data mining to be successful.

Different Processes

The data needs to be cleaned, integrated and selected before passing it to the database or data warehouse server. As the data is from different sources and in different formats, it cannot be used directly for the data mining process because the data might not be complete and reliable.

b) Database or Data Warehouse Server

The database or data warehouse server contains the actual data that is ready to be processed. Hence, the server is responsible for retrieving the relevant data based on the data mining request of the user.

c) Data Mining Engine

The data mining engine is the core component of any data mining system. It consists of a number of modules for performing data mining tasks including association, classification, characterization, clustering, prediction, time-series analysis etc.

d) Pattern Evaluation Modules

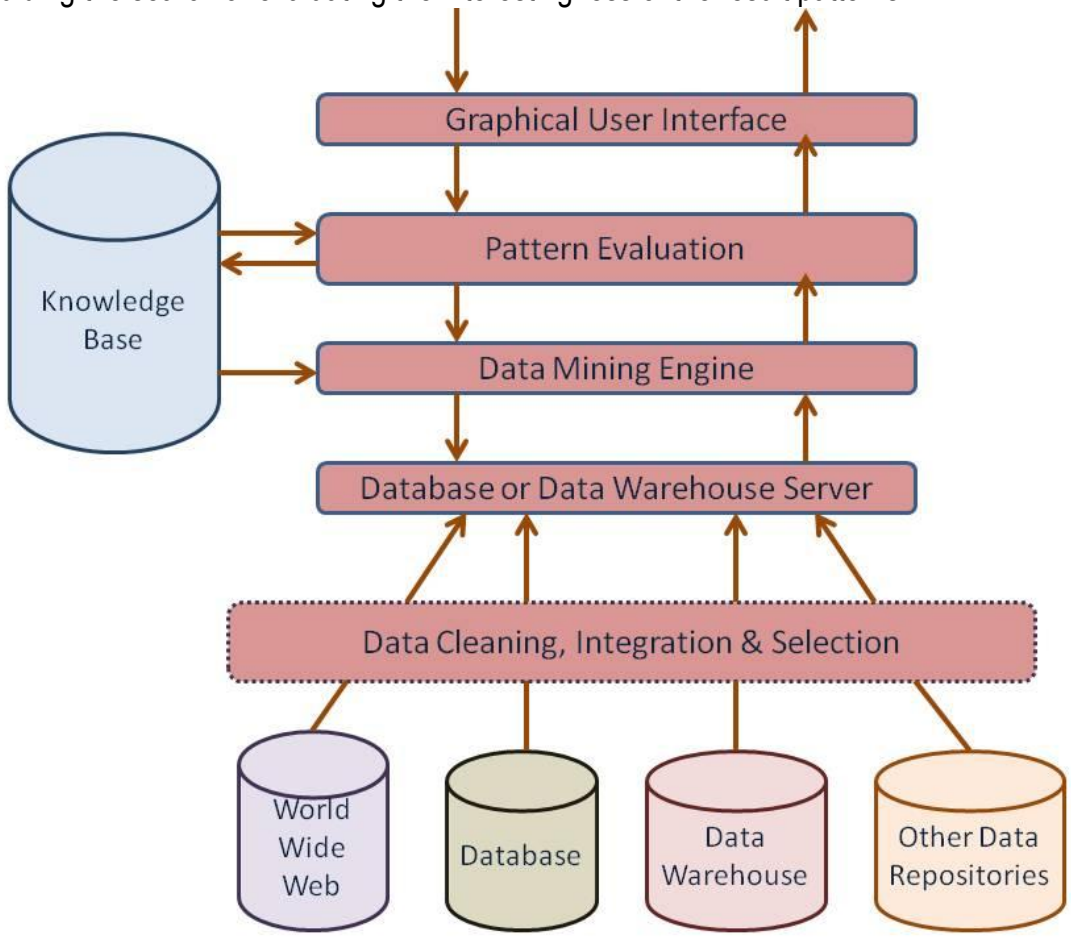
The pattern evaluation module is mainly responsible for the measure of interestingness of the pattern by using a threshold value. It interacts with the data mining engine to focus the search towards interesting patterns.

e) Graphical User Interface

The graphical user interface module communicates between the user and the data mining system.

f) Knowledge Base

The knowledge base is helpful in the whole data mining process. It might be useful for guiding the search or evaluating the interestingness of the result patterns.



25. (a). What is classification? Discuss the k-nearest neighbor classification algorithm.

- Classification is a data mining function that assigns items in a collection to target categories or classes.
- K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970's as a non-parametric technique.
- Algorithm
- A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor.

Distance functions

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$

Minkowski $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

- It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

Hamming Distance

$$D_H = \sum_{i=1}^{k}|x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

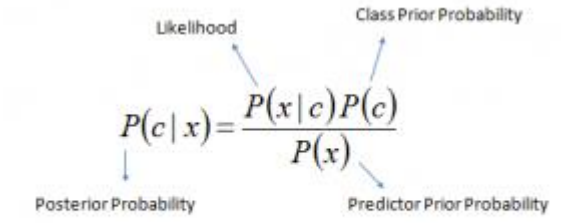
X	Y	Distance
Male	Male	0
Male	Female	1

- Choosing the optimal value for K is best done by first inspecting the data. In general, a large K value is more precise as it reduces the overall noise but there is no guarantee. Cross-validation is another way to retrospectively determine a good K value by using an independent dataset to validate the K value. Historically, the optimal K for most datasets has been between 3-10. That produces much better results than 1NN.

(OR)

(b**). Write a short notes on Naïve Bayes.**

- It is a classification technique 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.
- 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:



$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

- 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.

26. (a). What is data mining? What are all the steps involved in data mining process discuss?

- Data mining is the process of sorting through large data sets to identify patterns and establish relationships to solve problems through data analysis.
- Data mining is a five-step process:
    - Identifying the source information
    - Picking the data points that need to be analyzed
    - Extracting the relevant information from the data
    - Identifying the key values from the extracted data set
    - Interpreting and reporting the results

(OR)

(b).Write a short notes on relational databases and data warehouse.

- A **relational database** (RDB) is a collective set of multiple data sets organized by tables, records and columns. RDBs establish a well-defined relationship between database tables. Tables communicate and share information, which facilitates data searchability, organization and reporting.
- RDBs use Structured Query Language (SQL), which is a standard user application that provides an easy programming interface for database interaction.
- RDB is derived from the mathematical function concept of mapping data sets and was developed by Edgar F. Codd.
- a data warehouse is a relational database housed on an enterprise mainframe server or, increasingly, in the cloud. Data from various online transaction processing (OLTP) applications and other sources are selectively extracted for business intelligence activities, decision support and to answer user inquiries.

Basic components of a data warehouse

- A data warehouse stores data that is extracted from data stores and external sources. The data records within the warehouse must contain details to make it searchable and useful to business users. Taken together, there are three main components of data warehousing:
    - data sources from operational systems, such as Excel, ERP, CRM or financial applications;
    - a data staging area where data is cleaned and ordered; and
    - a presentation area where data is warehoused.