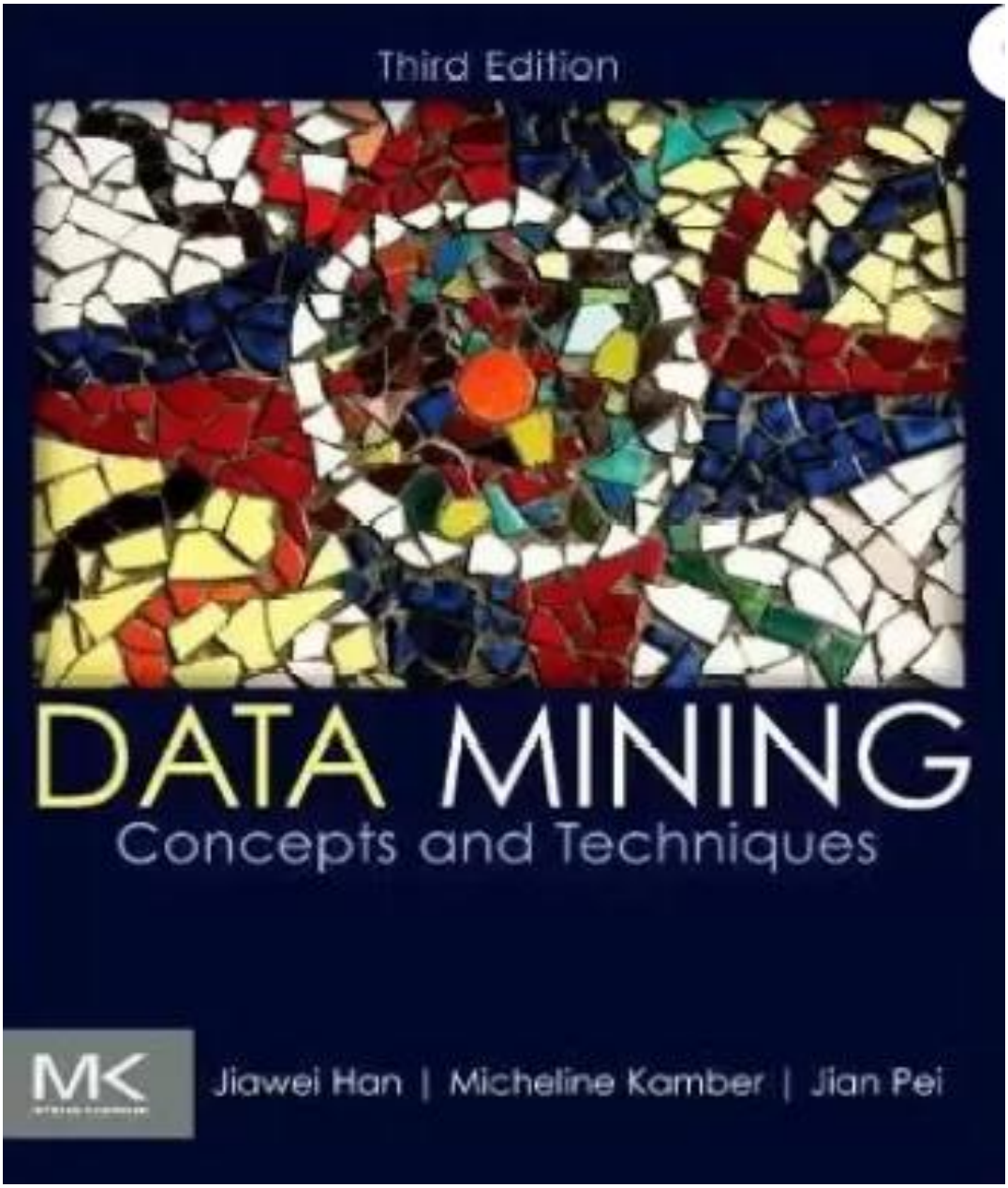
DATA MINING AND WARE HOUSING

P16CS31



STUDY MATERIAL PREPARED BY

Prof. A. NARAYANAN., M.Sc., M.Phil.,

ASSISTANT OF COMPUTER SCIENCE,
SWAMI DAYANANDA COLLEGE OF ARTS & SCIENCE,
MANJAKKUDI

II M.Sc., COMPUTER SCIENCE
Semester: III

CORE COURSE VII-DATA MINING AND WARE HOUSING
P16CS31

Inst. Hours/Week: 5 **Credit: 5**

Objective : On successful completion of the course the students should have: Understood data mining techniques-Concepts and design of data warehousing.

UNIT I

Introduction – What is Data mining – Data Warehouses – Data Mining Functionalities – Basic Data mining tasks – Data Mining Issues – Social Implications of Data Mining– Applications and Trends in Data Mining.

UNIT II

Data Preprocessing : Why preprocess the Data ? –Data Cleaning - Data Integration and Transformation – Data Reduction – Data cube Aggregation – Attribute Subset Selection Classification: Introduction – statistical based algorithms – Bayesian Classification. Distance based algorithms – decision tree based algorithms – ID3.

UNIT III

Clustering: Introduction - Hierarchical algorithms – Partitional algorithms – Minimum spanning tree – K-Means Clustering - Nearest Neighbour algorithm. Association Rules: What is an association rule? – Methods to discover an association rule–APRIORI algorithm – Partitioning algorithm .

UNIT IV

Data Warehousing: An introduction – characteristics of a data warehouse – Data marts – other aspects of data mart .Online analytical processing: OLTP & OLAP systems.

UNIT V

Developing a data warehouse : Why and how to build a data warehouse – Data warehouse architectural strategies and organizational issues – Design consideration – Data content – meta data – distribution of data – tools for data warehousing – Performance considerations

TEXT BOOKS

1. Jiawei Han and Miceline Kamber , “Data Mining Concepts and Techniques “ , Morgan Kaulmann Publishers, 2006. (Unit I – Chapter 1 -1.2, 1.4 , Chapter 11- 11.1) (Unit II – Chapter 2 - 2.1,2.3, 2.4, 2.5.1,2.5.2) 2. Margaret H Dunham , “Data mining Introductory & Advanced Topics”, Pearson Education , 2003.(Unit I – Chapter 1 -1.1 , 1.3, 1.5) , (UNIT II – Chapter 4 – 4.1, 4.2, 4.3, 4.4) (UNIT III – Chapter 5 – 5.1,5.4, 5.5.1, 5.5.3,5.5.4, Chapter 6 – 6.1,6.3. 3. C.S.R.Prabhu, “Data Warehousing concepts, techniques, products & applications”, PHI, Second Edition.) (UNIT IV & V) REFERENCES: 1. Pieter Adriaans, Dolf Zantinge, “Data Mining” Pearson Education, 1998.

2. Arun K Pujari, “Data Mining Techniques”,Universities Press(India) Pvt, 2003.

3. S.Rajashekharan, G A Vijaylakshmi Bhai,”Neural Networks,Fuzzy Logic,and Genetic Algorithms synthesis and Application”, PHI 4. Margaret H.Dunham,” Data Mining Introductory and Advanced topics”,Pearson Eductaionn 2003.

*****

UNIT I

INTRODUCTION : WHAT IS DATA MINING?

Definition

- Data Mining refers to extracting or mining knowledge from large amount of data .
- In simple words ,data mining is defined a process used to extract usable data from a larger set of any raw data.
- Data mining is the practice of examining large pre-existing databases in order to generate new information

On defining data mining we can know the related terms of data mining , they are

Database

-Database is an organized collection of data, generally stored and accessed electronically from a computer system .

DBMS

-Database Management system is a software that interacts with the end users, applications, and the database itself to capture and analyze the data.

Data warehouse

- a large store of data accumulated from a wide range of sources within a company and used to guide management decisions.

OLTP

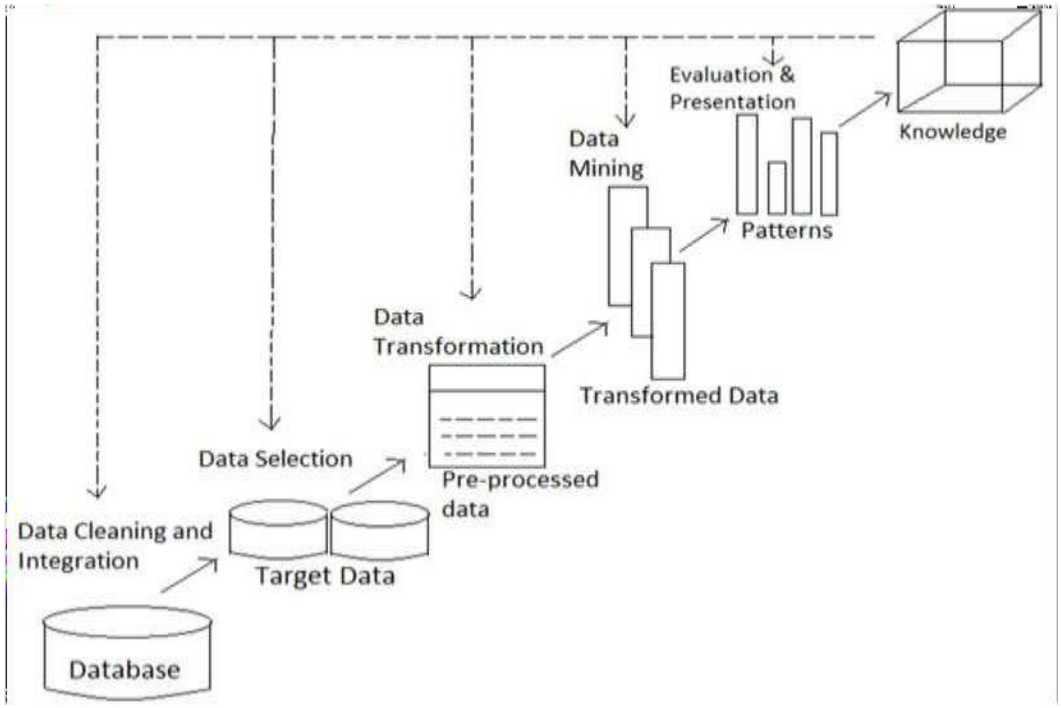
-Online Transaction processing is a class of software programs capable of supporting transactions oriented applications on the internet. (eg) log file, online banking .

KDD

- Many people treat data mining as a synonym for another popular used term Knowledge Discovery from Data or KDD.
- But Data Mining is an essential step in the process of knowledge discovery
- Data mining as a step in the process of Knowledge discovery

1.Data Cleaning
2. Data Integration
3.Data Selection
4.Data Transformation
5.Data mining
6.Pattern Evaluation
7.Knowledge Presentation

Steps in KDD



- Data Cleaning

  -To remove noise and inconsistent data.
- Data Integration

  -where multiple sources may be combined
- Data Integration

  -where data relevant to the analysis task are retrieved from the database
- Data Transformation

  -where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.
- Data Mining

  -an essential process where intelligent methods are applied in order to exact data pattern
- Pattern Evaluation

  -to identify the truly interesting patterns representing knowledge based on some interestingness measures

- Knowledge presentation

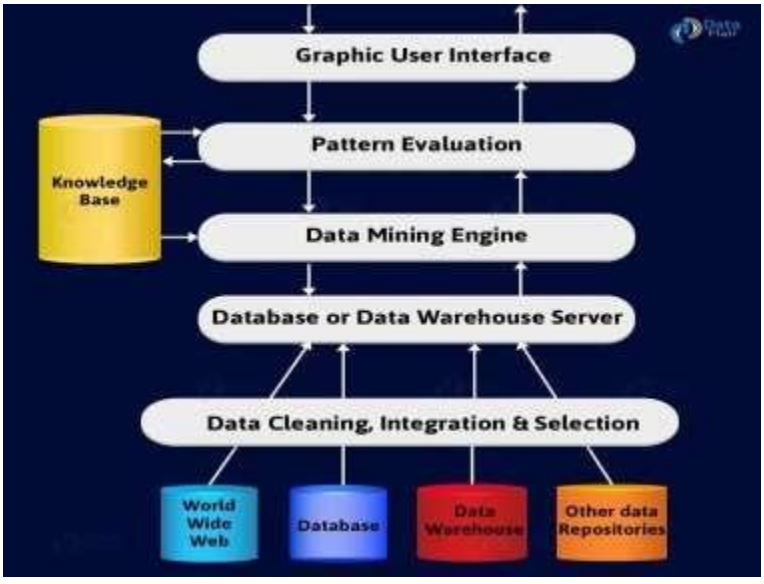
  - where visualization and knowledge representation technique are used to present the mined knowledge to the user

- But in database research ,Data mining is becoming more popular than KDD.
- Actually data mining is the process of discovering interesting knowledge fromlarge amounts of data stored in databases ,data warehouses or other information repositories.

ARCHITECTURE OF TYPICAL DATA MINING

The architecture optical data mining may have the following major components

- Database, Data warehouse, World Wide Web , or other information repository
- Database or Data warehouse server
- Knowledge base
- Data mining engine
- Pattern evaluation module
- Graphical user interface



- Database, Data warehouse, World Wide Web , or other information repository

  -one or a set of database, data warehouses, spread sheets, or other kinds information repositories.
  - Data cleaning and Data integration techniques may be performed on the data.

- Database or Data warehouse server

  -responsible for fetching the relevant data , based on the user's data mining request.

- Knowledge base

  -this is the domain knowledge that is used to guide the search or evaluate the interestingness of resulting patterns.
  -Knowledge can include used to concept hierarchies, used to attributes or attribute values into different levels of abstraction
  - Knowledge such as user beliefs, which can be used to assess a pattern 's interestingness based on its unexpectedness, may also be included

- Data mining engine

  -This is essential to the data mining system and ideally consists of a set of functional modules for tasks such as characterization, association and correlation analysis, classification, prediction, cluster analysis, outlier analysis and evolution analysis .

- Pattern evaluation module

  -This component typically employs interestingness measures and interacts with the data mining modules so as to focus the search toward interesting patterns .
  -For efficient data mining ,it is highly recommended to push the evaluation of pattern interestingness as deep as possible into the data mining process.

- Graphical User interface

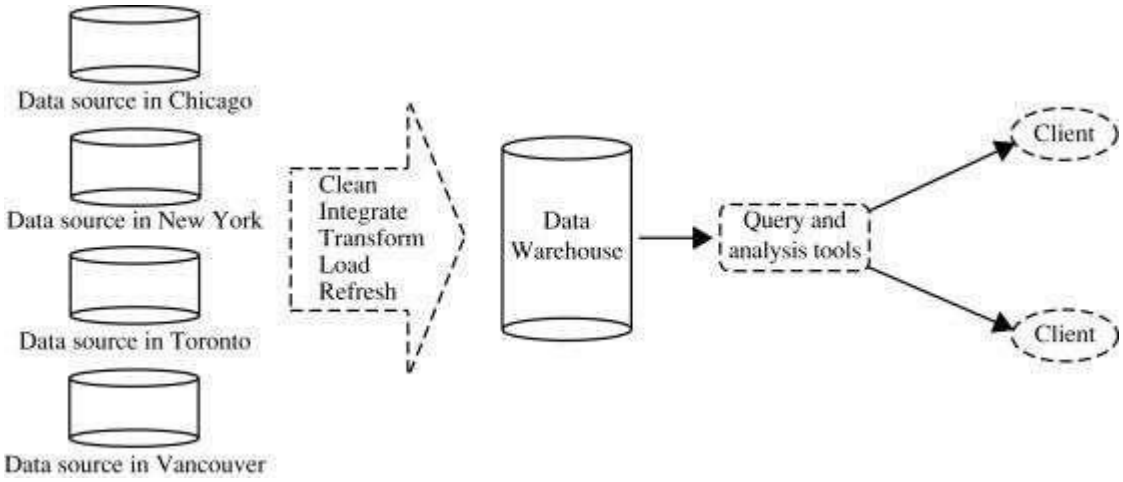
  -This module communicates between users and the data mining system, allowing the user to interact with the system by specifying a data mining query or task, providing information to help focus the search ,and performing exploratory data mining based on the intermediate data mining results.

DATA WAREHOUSES

A Data warehouse is a repository of information collected from multiple sources, stored under a unified schema , and that usually resides at a single site.

Data warehouse are constructed via a process of data cleaning ,data integration, data transformation, data loading, and periodic data refreshing.

Framework for construction and use of a data warehouse



Data warehouse provide information from a historical perspective and are typically summarized. For example ,rather than storing detailed information of each transaction in a super market just it stores the summarization of transaction based on item sales.

Data warehouse is usually periodically updated ,so it doesn't contain current information .

A data warehouse is usually modeled by a multidimensional database structure , where each dimensions corresponds to an attribute or a set of attributes in the schema.

The actual physical structure of a data warehouse may be relational data store or a multidimensional data cube.

DATA MINING FUNCTIONALITIES—WHAT KINDS OF PATTERNS CANBE MINED?

Data mining functionalities are used to specify the kind of patterns to be found in data mining tasks.

Data mining tasks can be classified into two categories:

1. Descriptive
2. Predictive.

Descriptive mining tasks characterize the general properties of the data in the database.

Predictive mining tasks perform inference on the current data in order to make

predictions.

Concept/Class Description: Characterization and Discrimination

Data can be associated with classes or concepts. For example, in the *AllElectronics* store, classes of items for sale include *computers* and *printers*, and concepts of customers include ***bigSpenders*** and ***budgetSpenders*****.** It can be useful to describe individual classes and concepts in summarized, concise, and yet precise terms. Such descriptions of a class or a concept are called class/concept descriptions. These descriptions can be derived via

Data characterization, by summarizing the data of the class under study (often called the target class) .

Data discrimination, by comparison of the target class with one or a set of comparative classes (often called the contrasting classes).

Data characterization is a summarization of the general characteristics or features of a target class of data. The data corresponding to the user-specified class are typically collected by a database query the output of data characterization can be presented in various forms. Examples include pie charts, bar charts, curves, multidimensional data cubes, and multidimensional tables, including crosstabs.

Data discrimination is a comparison of the general features of target class data objects with the general features of objects from one or a set of contrasting classes. The target and contrasting classes can be specified by the user, and the corresponding data objects retrieved through database queries. Discrimination descriptions expressed the output in rule form are referred to as discriminate rules.

Mining Frequent Patterns, Associations, and Correlations

Frequent patterns, as the name suggests, are patterns that occur frequently in data. There are many kinds of frequent patterns, including itemsets, subsequences, and substructures.

A ***frequent itemset*** typically refers to a set of items that frequently appear together in a transactional data set, such as Computer and Software.

A frequently occurring subsequence, such as the pattern that customers tend to purchase first a PC, followed by a digital camera, and then a memory card, is a (*frequent*) ***sequential pattern*****.**

A substructure can refer to different structural forms,such as graph,trees,or lattice ,which may be combined with itemsets or subsequences**.**If a substructure occurs

frequently it is called a **frequent structured pattern**.

Example: Association analysis. Suppose, as a marketing manager of *AllElectronics*, you would like to determine which items are frequently purchased together within the same transactions. An example of such a rule, mined from the *AllElectronics* transactional database, is

***buys*(*X*, "*computer*"‖) =>*buys*(*X*,"*software*")**

[*support* = 1%, *confidence* = 50%]

where *X* is a variable representing a customer. A confidence, or certainty, of 50% means that if a customer buys a computer, there is a 50% chance that she will buy software as well. A 1% support means that 1% of all of the transactions under analysis showed that computer and software were purchased together. This association rule involves a single attribute or predicate (i.e., *buys*) that repeats.

Association rules that contain a single predicate are referred to as

Single dimensional association rules.

Association rules that contain a multi predicate are referred to as

Multi association rules.

Classification and Prediction

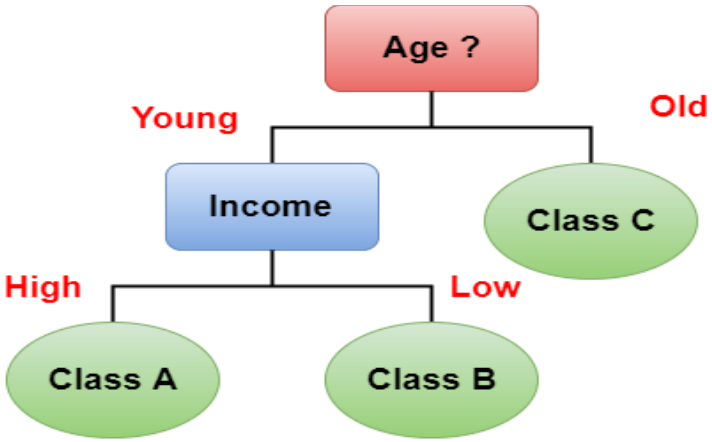
Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown. The derived model is based on the analysis of a set of training data (i.e., data objects whose class label is known).

The derived model may be represented in various forms, such as ***classification (IF-THEN) rules***, *decision trees*, *mathematical formulae*, or *neural networks.*

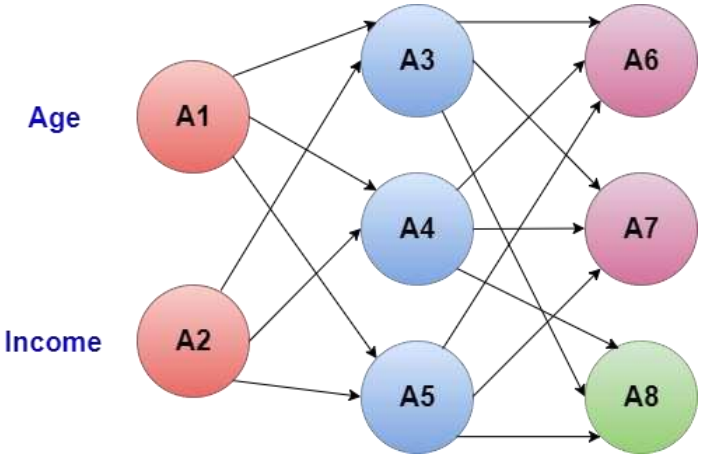
age (X,"youth")ANDincome(X,"high")->class (X,"A")

age (X,"youth")ANDincome(X,"low")->class (X,"B")

A decision tree is a flow-chart-like tree structure, where each node denotes a test on an attribute value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions. Decision trees can easily be converted to classification rules



A neural network, when used for classification, is typically a collection of neuron-like processing units with weighted connections between the units. There are many other methods for constructing classification models, such as naïve Bayesian classification, support vector machines, and *k*-nearest neighbor classification.



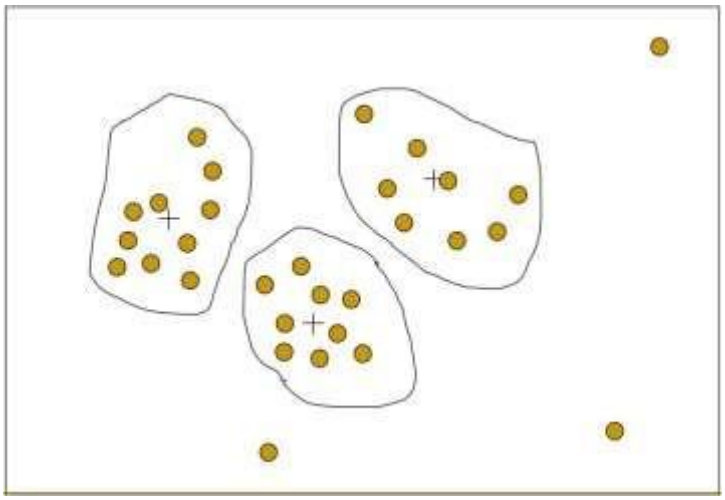
Whereas classification predicts categorical (discrete, unordered) labels, prediction models Continuous-valued functions. That is, it is used to predict missing or unavailable *numerical data values* rather than class labels. Although the term *prediction* may refer to both numeric prediction and class label prediction.

Cluster Analysis

Classification and prediction analyze class-labeled data objects, where as **clustering** analyzes data objects without consulting a known class label.

In general the class labels are not present in the training data simply because they are not known to begin with clustering can be used to create such labels .

The labels are clustered or grouped based on the principle of maximizing the intra class similarity and minimizing the inter class similarity.



Outlier Analysis

A database may contain data objects that do not comply with the general behavior or model of the data. These data objects are outliers. Most data mining methods discard outliers as noise or exceptions.

However, in some applications such as fraud detection, the rare events can be more interesting than the more regularly occurring ones. The analysis of outlier data is referred to as outlier mining.

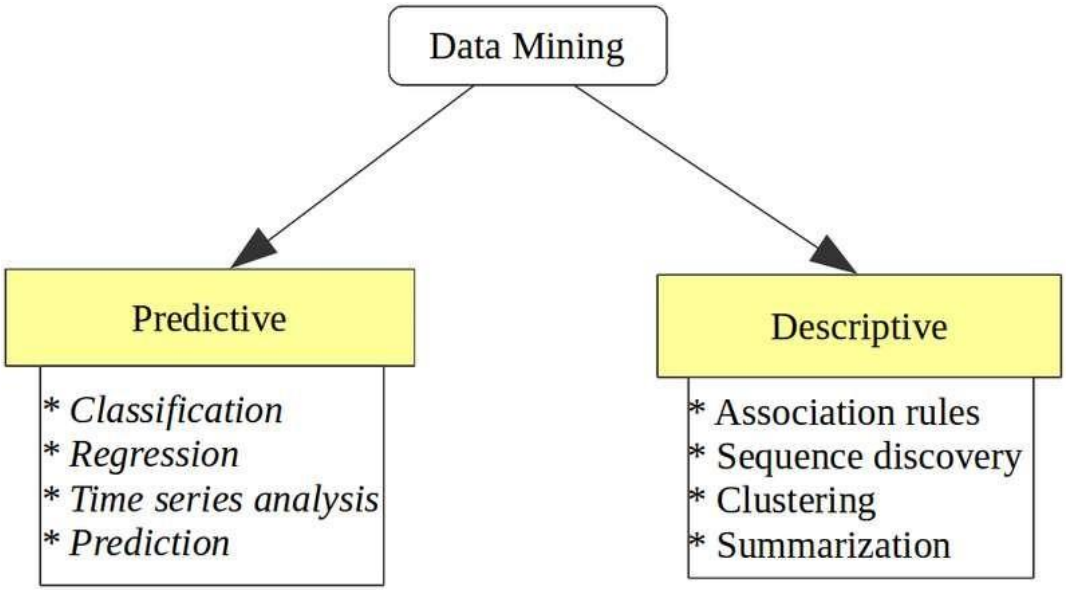
Evolution Analysis

Data evolution analysis describes and models regularities or trends for objects whose behavior changes over time. Although this may include characterization, discrimination, association and correlation analysis, classification, prediction, or clustering of *time related* data, distinct features of such an analysis include time-series data analysis,Sequence or periodicity pattern matching, and similarity-based dataanalysis.

BASIC DATA MINING TASKS

Data mining involve many different algorithms to accomplish different tasks. All these algorithms attempt to fit a model to the data. The algorithms to the characteristics of the data being examined.Data minng algorithms can be characterized as onsisiting of three parts:

- Model : The purpose of the algorithm is to fit model to the data.
- Preference: Some criteria must be used to fit one model over another.
- Search : All algorithms require some techniques to search the data.



Predictive data mining tasks come up with a model from the available data set that is helpful in predicting unknown or future values of another data set of interest. A medical practitioner trying to diagnose a disease based on the medical test results of a patient can be considered as a predictive data mining task.

Descriptive data mining tasks usually finds data describing patterns and comes up with new, significant information from the available data set. A retailer trying to identify products that are purchased together can be considered as a descriptive data mining task.

a) Classification

Classification derives a model to determine the class of an object based on its attributes. A collection of records will be available, each record with a set of attributes. One of the attributes will be class attribute and the goal of classification task is assigning a class attribute to new set of records as accurately as possible.

Example : An airport security screening station is used to determine if passengers are potential terrorists or criminals. To do this the face of each passenger is scanned and its basic pattern (distance between eyes ,size and shape of mouth , shape of head , etc.) is identified. This is pattern is compared to entries in a database o see if it matches any patterns that are associated with known offenders.

b) Regression

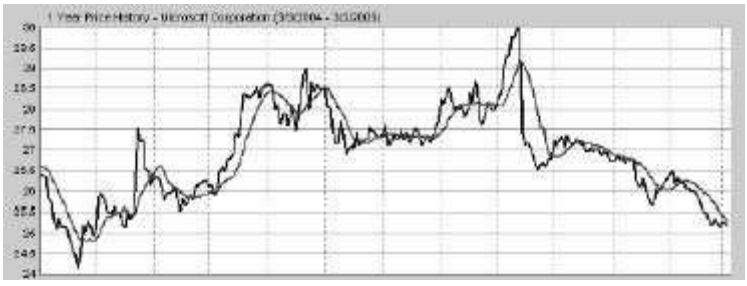
The regression task is similar to classification. The main difference is that the predictable attribute is a continuous number. Regression techniques have been widely studied for centuries in the field of statistics. Linear regression and logistic regression are the most popular regression methods. Other regression techniques include regression trees and neural networks. Regression tasks can solve many business problems.

Example : predict wind velocities based on past temperature, air pressure, and humidity.

c) Time - Series Analysis

Time series is a sequence of events where the next event is determined by one or more of the preceding events. Time series reflects the process being measured and there are certain components that affect the behavior of a process. Time series analysis includes methods to analyze time-series data in order to extract useful patterns, trends, rules and statistics.

Example : Stock market prediction is an important application of time- series analysis. A person is trying to determine whether to purchase stock from companies X,Y,Z. For period of one month he charts the daily stock price for each company .Based on this he take his decisions.



d) Prediction

Prediction task predicts the possible values of missing or future data. Prediction involves developing a model based on the available data and this model is used in predicting future values of a new data set of interest.

Example : A model can predict the income of an employee based on education, experience and other demographic factors like place of stay, gender etc. Also prediction analysis is used in different areas including medical diagnosis, fraud detection etc.

e) Association Rules

Association discovers the association or connection among a set of items. Association identifies the relationships between objects. Association analysis is used for commodity management, advertising, catalog design, direct marketing etc.

Example: A retailer can identify the products that normally customers purchase together or even find the customers who respond to the promotion of same kind of products. If a retailer finds that bread and jam are bought together mostly, he can put bread on sale to promote the sale of jam.

f) Sequence Discovery

Sequence analysis or Sequence discovery is used to determine sequential patterns in data. These patterns are based on a time sequence of actions. These patterns are similar to associations in that data are found to be related, but the relationship is based on time.

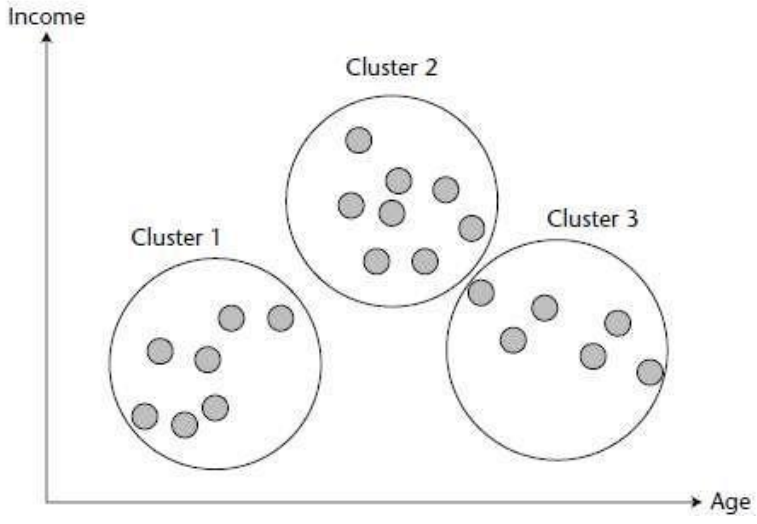
In market basket analysis the items are purchased at same time , but in the sequence discovery the items are purchased over time in some order.

Example :Most people who purchase CD players may be found to purchase CDs within one week and speaker ,and then home theater

g) Clustering

Clustering is used to identify data objects that are similar to one another. The similarity can be decided based on a number of factors like purchase behavior, responsiveness to certain actions, geographical locations and so on.

Example : An insurance company can cluster its customers based on age, residence, income etc. This group information will be helpful to understand the customers better and hence provide better customized services.



h) Summarization

Summarization is the generalization of data. A set of relevant data is summarized which result in a smaller set that gives aggregated information of the data.

Example: The shopping done by a customer can be summarized into total products, total spending, offers used, etc. Such high level summarized information can be useful for sales or customer relationship team for detailed customer and purchase behavior analysis. Data can be summarized in different abstraction levels and from different angles.

DATA MINING ISSUES

Data mining is not an easy task, as the algorithms used can get very complex and data is not always available at one place. It needs to be integrated from various heterogeneous data sources. These factors also create some issues.

1. Human interaction: In data mining ,interfaces may be needed with both domain and technical experts .Technical experts are used to formulate the queries and assist in interpreting the results.

2. Over fitting: When a model is generated that is associated with a given database state ,it is desirable that the model also fit future database states. Over fitting occurs when the model does not fit future states.

3. Outliers : There are often many data entries that do not fit nicely into the derived model. If a model is developed that includes these outliers, then the model may not behave well for data that are not outliers.

4. Interpretation of results: Data mining output may require experts to correctly interpret the results ,which might otherwise meaningless to the average database user.

5. Visualization of results: To easily view and understand the output of data mining algorithms, visualization of the results is helpful.

6. Large datasets : The massive datasets associated with data mining create problems when applying algorithms designed for small datasets. Many modeling applications grow exponentially on the dataset size and thus are too inefficient forlarger datasets.

7. **High dimensionality :** A conventio$_1$n$_5$al database scheme may be composed of many different attributes. The problem here is tat not all attributes may be needed to sole the problem, It is difficult to determine which ones should be needed . The use of all attributes simply increases the overall complexity. This problem called dimensionality curse. One solution is to reduce the number of attributes, which is known as dimensionality reduction.

8. **Multimedia** : Most previous data mining algorithms are targeted to traditional data types. The use of multimedia complicates or invalidates many proposedalgorithms.

9. **Missing data :** During the KDD phase, missing data may be placed with estimates, this may leads to invalid results.

10. **Irrelevant data :** some attributes in the database might not be of interest to the data mining task being developed.

11. **Noisy data :** some attribute values might be invalid or in correct .

12. **Changing data :** Databases cannot be assumed to be static, but most data mining algorithms require static database. this requires that the algorithm be completely rerun anytime the database changes.

13. **Integration :** The KDD process is not currently integrated into normal data processing activities. This may be treated as special ,unusual ,or onetime need.This makes them inefficient, ineffective, and not general to be used on an ongoing basis.

14. **Application :** Determining the intended use for the information obtained from the data mining function is a challenge. Indeed ,how business executives can effectively use the output is sometimes considered the more difficult part , not the running of the algorithms themselves.

SOCIAL IMPLICATION OF DATA MINING

The integration of data mining techniques into normal day to day activities has become common place .We confronted daily with targeted advertising and business have become more efficient through the use of data mining activities to reduce cost, at the same time the information is being obtained at the cost of reduced privacy.

Data mining applications on derive much demographic information concerning Customers that was previously not known or hidden in the data .The unauthorized use of such data could result in the disclosure of information that is deemed to be confidential.

We have recently seen an increase in data mining techniques targeted applications as fraud detection , identifying criminal suspects , and prediction of potential terrorists. The approach that is often used is " profiling" the typical behavior or characteristics involved. Some times these approaches to classification are imperfect. Mistakes can be made. Just because an individual makes a series of credit card purchases that are similar to those often made when a card is stolen does not mean that the card is stolen or that the individual is a criminal.

DATA MINING APPLICATIONS

Data mining is used in a vast array of areas ,and numerous commercial data mining systems are available. Still data mining is a relatively young discipline with wide and diverse applications there is a nontrivial gap between general principles of data mining and application specific , effective data mining tools.

List of areas where data mining is widely used −

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

Financial Data Analysis

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

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

Retail Industry

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

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

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

Telecommunication Industry

Today the telecommunication industry is one of the most emerging industries providing various services such as fax, cellular phone, internet messenger, images, e-mail, web data transmission, etc. Due to the development of new computer and communication technologies, the telecommunication industry is rapidly expanding. This is the reason why data mining is become very important to help and understand the business. Data mining in telecommunication industry helps in identifying the telecommunication patterns, catch fraudulent activities, make better use of resource, and improve quality of service. Here is the list of examples for which data mining improves telecommunication

services −

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

Biological Data Analysis

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

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

Other Scientific Applications

Huge amount of data have been collected from scientific domains such as geosciences, astronomy, etc. A large amount of data sets is being generated because of the fast numerical simulations in various fields such as climate and ecosystem modeling, chemical engineering, fluid dynamics, etc. Following are the applications of data mining in the field of Scientific Applications −

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

Intrusion Detection

Intrusion refers to any kind of action that threatens integrity, confidentiality, or the availability of network resources. In this w[1]o[9]rld of connectivity, security has become the

major issue. With increased usage of internet and availability of the tools and tricks for intruding and attacking network prompted intrusion detection to become a critical component of network administration.

Here is the list of areas in which data mining technology may be applied for intrusion detection −

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

TRENDS IN DATA MINING

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

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

UNIT-II
DATA PREPROCESSING

1. Preprocessing

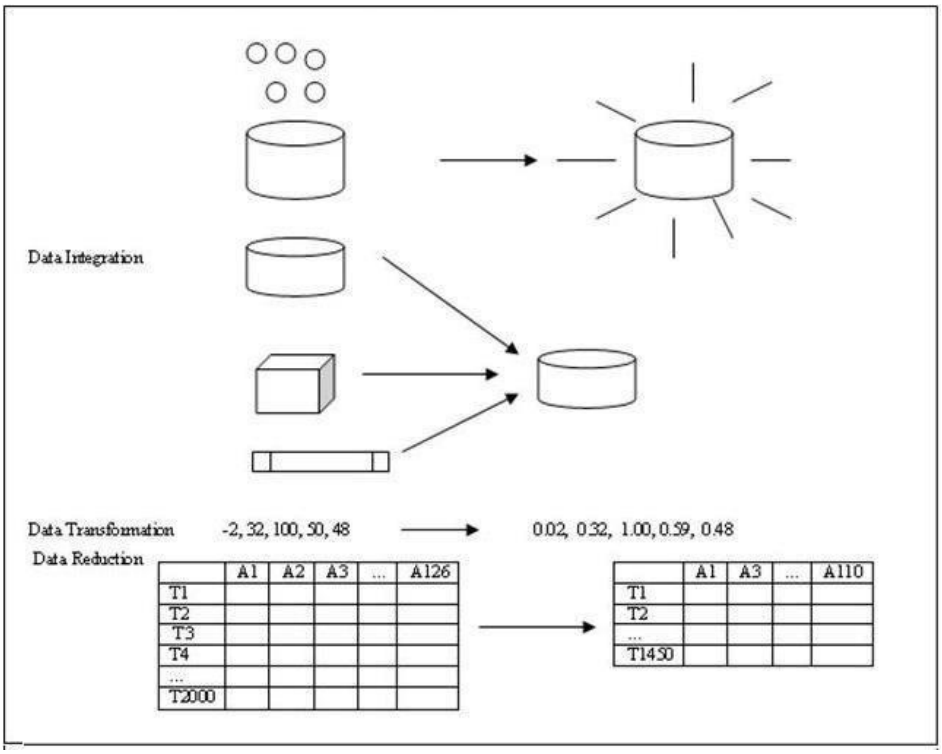
Real-world databases are highly susceptible to noisy, missing, and inconsistent data due to their typically huge size (often several gigabytes or more) and their likely origin from multiple, heterogeneous sources. Low-quality data will lead to low-quality mining results, so we prefer a preprocessing concepts.

Data Preprocessing Techniques

* **Data cleaning** can be applied to remove noise and correct inconsistencies in the data.
* **Data integration** merges data from multiple sources into coherent data store, such as a data warehouse.
* **Data reduction** can reduce the data size by aggregating, eliminating redundant features, or clustering, for instance. These techniques are not mutually exclusive; they may work together.
* **Data transformations**, such as normalization, may be applied.

Need for preprocessing

- Incomplete, noisy and inconsistent data are common place properties of large real world databases and data warehouses.
- Incomplete data can occur for a number of reasons:
  - Attributes of interest may not always be available
  - Relevant data may not be recorded due to misunderstanding, or because of equipment malfunctions.
  - Data that were inconsistent with other recorded data may have been deleted.
  - Missing data, particularly for tuples with missing values for some attributes, may need to be inferred.
  - The data collection instruments used may be faulty.
  - There may have been human or computer errors occurring at data entry.
  - Errors in data transmission can also occur.
  - There may be technology limitations, such as limited buffer size for coordinating synchronized data transfer and consumption.
  - Data cleaning routines work to ―clean‖ the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies.
  - Data integration is the process of integrating multiple databases cubes or files. Yet some attributes representing a given may have different names in different databases, causing inconsistencies and redundancies.
  - Data transformation is a kind of operations, such as normalization and aggregation, are additional data preprocessing procedures that would contribute toward the success of the mining process.
  - Data reduction obtains a reduced representation of data set that is much smaller in volume, yet produces the same(or almost the same) analytical results.



2. DATA CLEANING

Real-world data tend to be incomplete, noisy, and inconsistent. Data cleaning (or data cleansing) routines attempt to fill in missing values, smooth out noise while identifying outliers and correct inconsistencies in the data.

Missing Values

Many tuples have no recorded value for several attributes, such as customer income.so we can fillthe missing values for this attributes.

The following methods are useful for performing missing values over several attributes:

1. **Ignore the tuple**: This is usually done when the class label missing (assuming the mining task involves classification). This method is not very effective, unless the tuple contains several attributes with missing values. It is especially poor when the percentage of the missing values per attribute varies considerably.
2. **Fill in the missing values manually**: This approach is time –consuming and may not be feasible given a large data set with many missing values.
3. **Use a global constant to fill in the missing value**: Replace all missing attribute value by the same constant, such as a label like ‑unknown‖ or -∞.
4. **Use the attribute mean to fill in the missing value**: For example, suppose that the average income of customers is $56,000. Use this value to replace the missing value for income.
5. **Use the most probable value to fill in the missing value**: This may be determined with regression, inference-based tools using a Bayesian formalism or decision tree induction. For example, using the other customer attributes in the sets decision tree is constructed to predict the missing value for income.

Noisy Data

Noise is a random error or variance in a measured variable. Noise is removed using datasmoothing techniques.

Binning: Binning methods smooth a sorted data value by consulting its ―neighborhood,‖ that is the value around it. The sorted values are distributed into a number of ―buckets‖ or ―bins―. Because binning methods consult the neighborhood of values, they perform local smoothing. Sorted datafor price (in dollars): 3,7,14,19,23,24,31,33,38.

Example 1: Partition into (equal-frequency) bins:Bin 1: 3,7,14
Bin 2: 19,23,24
Bin 3: 31,33,38

In the above method the data for price are first sorted and then partitioned into equal- frequencybins of size 3.

Smoothing by bin means:

Bin 1: 8,8,8
Bin 2: 22,22,22
Bin 3: 34,34,34

In smoothing by bin means method, each value in a bin is replaced by the mean value ofthe bin. Forexample, the mean of the values 3,7&14 in bin 1 is 8[(3+7+14)/3].

Smoothing by bin boundaries:

Bin 1: 3,3,14
Bin 2: 19,24,24
Bin 3: 31,31,38

In smoothing by bin boundaries, the maximum & minimum values in give bin or identify as the binboundaries. Each bin value is then replaced by the closest boundary value.

In general, the large the width, the greater the effect of the smoothing. Alternatively, bins may be equal-width, where the interval range of values in each bin is constant Example 2: Remove the noise in the following data using smoothing techniques:

8, 4,9,21,25,24,29,26,28,15

Sorted data for price (in dollars):4,8,9,15,21,21,24,25,26,28,29,34

Partition into equal-frequency (equi-depth) bins:

Bin 1: 4, 8,9,15
Bin 2: 21,21,24,25
Bin 3: 26,28,29,34

Smoothing by bin means:

Bin 1: 9,9,9,9

Bin 2: 23,23,23,23

Bin 3: 29,29,29,29

Smoothing by bin boundaries:

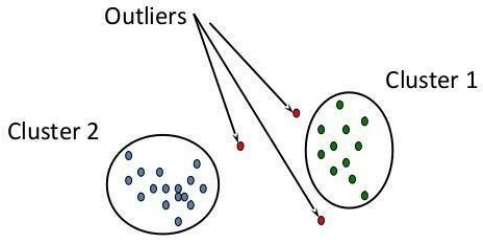
Bin 1: 4, 4,4,15

Bin 2: 21,21,25,25

Bin3: 26,26,26,34

Regression: Data can be smoothed by fitting the data to function, such as with regression. Linear regression involves finding the ―best‖ line to fit two attributes (or variables), so that one attribute can be used to predict the other. Multiple linear regressions is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface.

Clustering: Outliers may be detected by clustering, where similar values are organized into groups, or ―clusters.‖ Intuitively, values that fall outside of the



set of clusters may be considered outliers.

2.3 Inconsistent Data

Inconsistencies exist in the data stored in the transaction. Inconsistencies occur due to occur during data entry, functional dependencies between attributes and missing values. The inconsistencies can be detected and corrected either by manually or by knowledge engineering tools.

Data cleaning as a process

a) Discrepancy detection

b) Data transformations

a) Discrepancy detection

The first step in data cleaning is discrepancy detection. It considers the knowledge of meta data and examines the following rules for detecting the discrepancy.

Unique rules- each value of the given attribute must be different from all other values for that attribute.

Consecutive rules – Implies no missing values between the lowest and highest values for the attribute and that all values must also be unique.

Null rules - specifies the use of blanks, question marks, special characters, or other strings that may indicates the null condition

Discrepancy detection Tools:

- Data scrubbing tools - use simple domain knowledge (e.g., knowledge of postal addresses, and spell-checking) to detect errors and make corrections in the data

- ❖ Data auditing tools – analyzes the data to discover rules and relationship, and detecting data that violate such conditions.

b) Data transformations

This is the second step in data cleaning as a process. After detecting discrepancies, weneed to define and apply (a series of) transformations to correct them.

Data Transformations Tools:

- ❖ Data migration tools – allows simple transformation to be specified, such to replaced the string –gender‖ by –sex‖.
- ❖ ETL (Extraction/Transformation/Loading) tools – allows users to specific transforms through a graphical user interface(GUI)

3. Data Integration

Data mining often requires data integration - the merging of data from stores into a coherent data store, as in data warehousing. These sources may include multiple data bases, data cubes, or flatfiles.

Issues in Data Integration

a) Schema integration & object matching.
b) Redundancy.
c) Detection & Resolution of data value conflict

a) Schema Integration & Object Matching

Schema integration & object matching can be tricky because same entity can be represented in different forms in different tables. This is referred to as the entity identification problem. Metadata can be used to help avoid errors in schema integration. The meta data may also be used to help transform the data.

b) Redundancy:

Redundancy is another important issue an attribute (such as *annual revenue*, for instance) may be redundant if it can be –derived‖ from another attribute are set of attributes. Inconsistencies in attribute of dimension naming can also cause redundancies in the resulting data set. Some redundancies can be detected by correlation analysis and covariance analysis.

For Nominal data, we use the χ^2 (Chi-Square) test.

For Numeric attributes we can use the correlation coefficient and covariance.

χ^2 Correlation analysis for numerical data:

For nominal data, a correlation relationship between two attributes, A and B, can be discovered by a $\chi 2$ (Chi-Square) test. Suppose A has c distinct values, namely a_1, a_2, a_3, ……., a_c. B has r distinct values, namely b_1, b_2, b_3, …., b_r. The data tuples are described by table.

The $\chi 2$ value is computed as

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

Where o_{ij} is the observed frequency of the joint event (A_i,B_j) and e_{ij} is the expected frequency of (A_i,B_j), which can computed as

$$e_{ij} = \frac{count\ A=a_i\ X count(B=b_j)}{n}$$

For
Example,

	Male	Female	Total
Fiction	250	200	450
Non_Fiction	50	1000	1050
Total	300	1200	1500

$$e_{11} = \frac{count\ male\ X count\ (fiction\ )}{n} = \frac{300X450}{1500} = 90$$

$$e_{12} = \frac{count\ male\ X count\ (non\_fiction\ )}{n} = \frac{300X1050}{1500} = 210$$

$$e_{21} = \frac{count\ female\ X count\ (fiction\ )}{n} = \frac{1200\ X450}{1500} = 360$$

$$e = count\ female\ X count\ (non\_fiction\ ) = \frac{1200\ X1050}{} = 840$$

	Male	Female	Total
Fiction	250 (90)	200 (360)	450
Non_Fiction	50 (210)	1000 (840)	1050
Total	300	1200	1500

For $\chi 2$ computation, we get

$$\chi 2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{200-360^{\ 2}}{360} + \frac{(1000-840)^2}{840}$$

$$= 284.44 + 121.90 + 71.11 + 30.48 = 507.93$$

For this 2 X 2 table, the degrees of freedom are (2-1(2-1)=1. For 1 degree of freedom, the x2 value needed to reject the hypothesis at the 0.001 significance level is 10.828(from statistics table). Since our computed value is greater than this, we can conclude that two attributes are strongly correlated for the given group of people.

Correlation Coefficient for Numeric data:

For Numeric attributes, we can evaluate the correlation between two attributes, A and B,

by computing the correlation coefficient. This is

$$r_{A,B} = \frac{\sum_{i=1}^{n} (a_i - A)(b_i - B)}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^{n} a_i b_i - nAB}{n\sigma_A\sigma_B}$$

For Covariance between A and B defined as

$$Cov(A,B) = \frac{\sum_{i=1}^{n} (a_i - A)(b_i - B)}{n}$$

c) Detection and Resolution of Data Value Conflicts.

A third important issue in data integration is the *detection and resolution of data valueconflicts*. For example, for the same real–world entity, attribute value from differentsources may differ. This may be due to difference in representation, scaling, or encoding.

For instance, a weight attribute may be stored in metric units in one system and British imperial units in another. For a hotel chain, the *price* of rooms in different cities may involve not only different currencies but also different services (such as free breakfast) and taxes. An attribute in one system may be recorded at a lower level of abstraction than the

–same‖ attribute in another.

Careful integration of the data from multiple sources can help to reduce and avoid redundancies and inconsistencies in the resulting data set. This can help to improve the accuracyand speed of the subsequent of mining process.

4. Data Reduction:

Obtain a reduced representation of the data set that is much smaller in volume but yetproduces the same (or almost the same) analytical results.

Why data reduction? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.

Data reduction strategies

4.1.Data cube aggregation

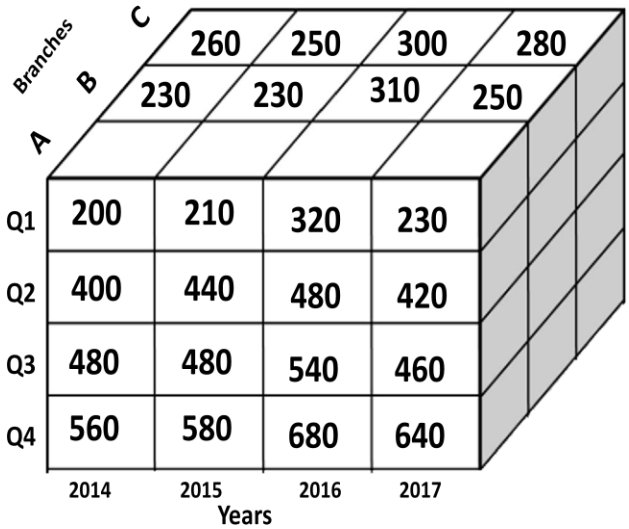
4.2.Attribute Subset Selection

4.3.Numerosity reduction — e.g., fit data into models

4.4.Dimensionality reduction - Data Compression

Data cube aggregation:

For example, the data consists of AllElectronics sales per quarter for the years 2014 to 2017.You are, however, interested in the annual sales, rather than the total per quarter. Thus, the data can be ***aggregated*** so that the resulting data summarize the total sales per year instead of per quarter.



Year/Quarter	2014	2015	2016	2017
Quarter 1	200	210	320	230

Year	Sales
2014	1640

Quarter 2	400	440	480	420
Quarter 3	480	480	540	460
Quarter 4	560	580	680	640

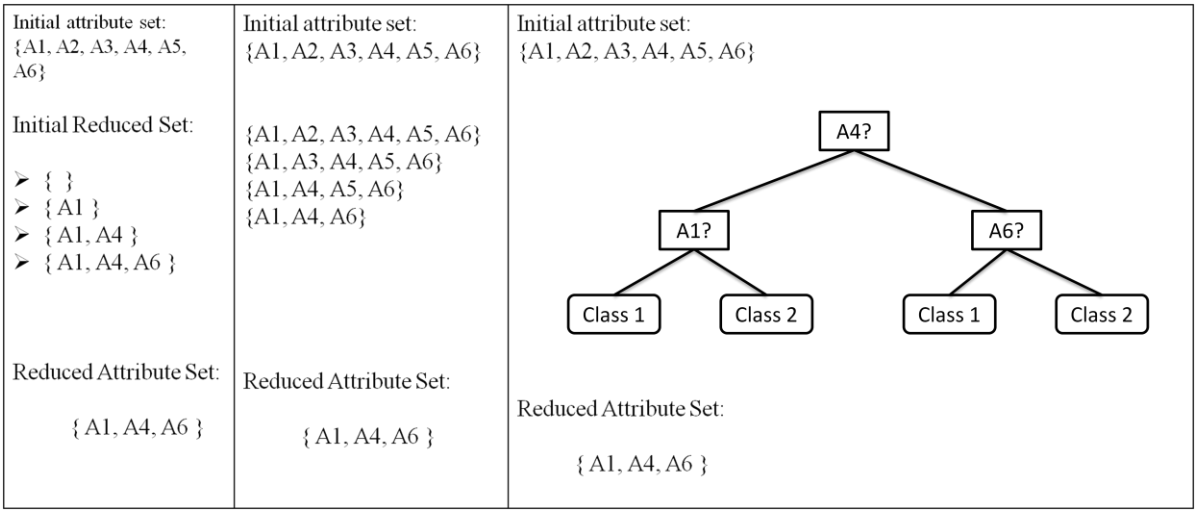
2015	1710
2016	2020
2017	1750

Attribute Subset Selection

Attribute subset selection reduces the data set size by removing irrelevant or redundant attributes (or dimensions). The goal of attribute subset selection is to find a minimum set of attributes. It reduces the number of attributes appearing in the discovered patterns, helping to make the patterns easier to understand.

For *n* attributes, there are 2*n* possible subsets. An exhaustive search for the optimal subset of attributes can be prohibitively expensive, especially as *n* and the number of data classes increase. Therefore, heuristic methods that explore a reduced search space are commonly used for attribute subset selection. These methods are typically greedy in that, while searching to attribute space, they always make what looks to be the best choice at that time. Their strategy to make a locally optimal choice in the hope that this will lead to a

globally optimal solution. Many other attributes evaluation



measure can be used, such as the information gain measure used in building decision trees for classification.

Techniques for heuristic methods of attribute sub set selection

- Stepwise forward selection

- Stepwise backward elimination
- Combination of forward selection and backward elimination
- Decision tree induction

1. ***Stepwise forward selection***: The procedure starts with an empty set of attributes as the reduced set. The best of original attributes is determined and added to the reduced set. At eachsubsequent iteration or step, the best of the remaining original attributes is added to the set.
2. ***Stepwise backward elimination***: The procedure starts with full set of attributes. At each step,it removes the worst attribute remaining in the set.
3. ***Combination of forward selection and backward elimination****:* The stepwise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.
4. ***Decision tree induction****:* Decision tree induction constructs a flowchart like structure where each internal node denotes a test on an attribute, each branch corresponds to an outcome of thetest, and each leaf node denotes a class prediction. At each node, the algorithm choices the

–best‖ attribute to partition the data into individual classes. A tree is constructed from the given data. All attributes that do not appear in the tree are assumed to be irrelevant. The set of attributes appearing in the tree from the reduced subset of attributes. Threshold measure is usedas stopping criteria.

Numerosity Reduction:

Numerosity reduction is used to reduce the data volume by choosing alternative, smaller forms ofthe data representation

Techniques for Numerosity reduction:

- Parametric - In this model only the data parameters need to be stored, instead of theactual data. (e.g.,) Log-linear models, Regression

- Nonparametric – This method stores reduced representations of data includehistograms, clustering, and sampling

Parametric model

1. Regression

- **Linear regression**
  - In linear regression, the data are model to fit a straight line. For

example, a randomvariable, Y called a response variable), can be modeled as a linear function of another random variable, X called a predictor variable), with the equation Y=αX+β

- Where the variance of Y is assumed to be constant. The coefficients, α and β (called regression coefficients), specify the slope of the line and the Y- intercept, respectively.

- **Multiple- linear regression**
  - Multiple linear regression is an extension of (simple) linear regression, allowing a response variable Y, to be modeled as a linear function of two or more predictor variables.

2. Log-Linear Models

- Log-Linear Models can be used to estimate the probability of each point in a multidimensional space for a set of discretized attributes, based on a smaller subset of dimensional combinations.

Nonparametric Model

1. Histograms

A histogram for an attribute A partitions the data distribution of A into disjoint subsets, orbuckets. If each bucket represents only a single attribute-value/frequency pair, the buckets are called singleton buckets.

Ex: The following data are bast of prices of commonly sold items at All Electronics. The numbershave been sorted:
1,1,5,5,5,5,5,8,8,10,10,10,10,12,14,14,14,15,15,15,15,15,18,18,18,18,18,18,18,18,20,20,20,20,20,21,21,21,21,21,25,25,25,25,25,28,28,30,30,30

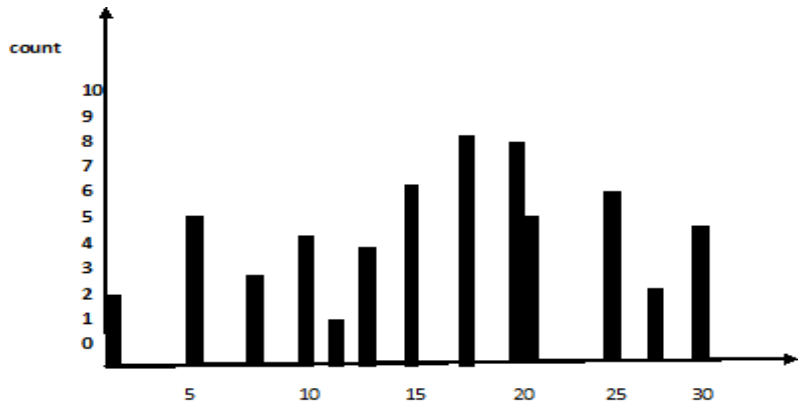


Figure 3.7 A Histogramfor price using Singleton Buckets

There are several partitioning rules including the following:Equal-width: The width of each bucket range is

uniform

- (Equal-frequency (or equi-depth): the frequency of each bucket is constant

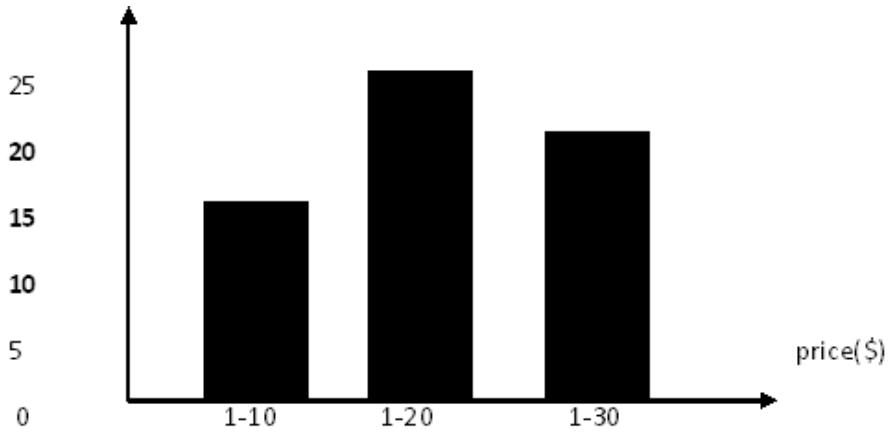


Figure.2.8 An equal-width histogram for price, where values are aggregated so ***that each bucket has a uniform width of $10.***

2. Clustering

Clustering technique consider data tuples as objects. They partition the objects into groups or clusters, so that objects within a cluster are similar to one another and dissimilar to objects in other clusters. Similarity is defined in terms of how close the objects are in space, based on a distance function. The quality of a cluster may be represented by its diameter, the maximum distance between any two objects in the cluster. Centroid distance is an alternative measure of cluster quality and is defined as the average distance of each cluster object from the cluster centroid.

3. Sampling:

Sampling can be used as a data reduction technique because it allows a large data set to be represented by a much smaller random sample (or subset) of the data. Suppose that a large data set D, contains N tuples, then the possible samples are Simple Random sample without Replacement (SRS WOR) of size n: This is created by drawing „n‟ of the „N‟ tuples from D (n<N), where the probability of drawing any tuple in D is 1/N, i.e., all tuples are equally likely to be sampled.

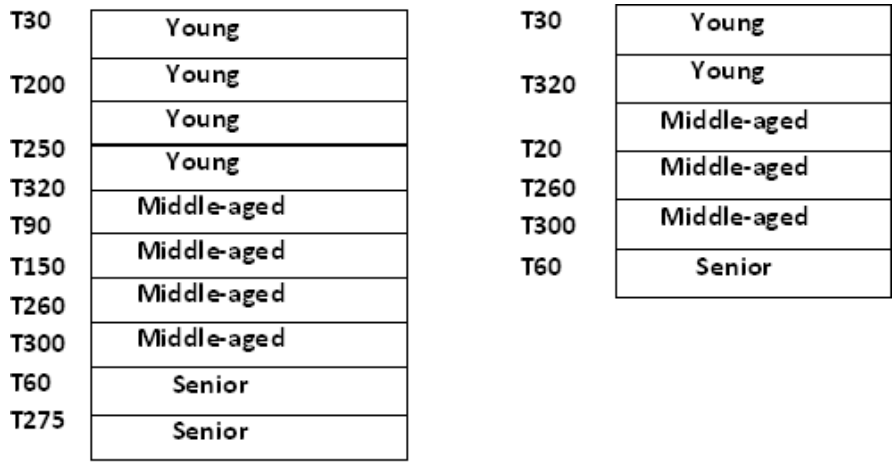


Figure 2.9. Sampling can be used for data reduction.

Dimensionality Reduction:

In dimensionality reduction, data encoding or transformations are applied so as to obtainedreduced or ‐compressed‖ representation of the oriental data.

Dimension Reduction Types

- Lossless - If the original data can be *reconstructed* from the compressed data without any loss of information
- Lossy - If the original data can be reconstructed from the compressed data with loss of information, then the data reduction is called lossy.

Effective methods in lossy dimensional reduction

a) Wavelet transforms

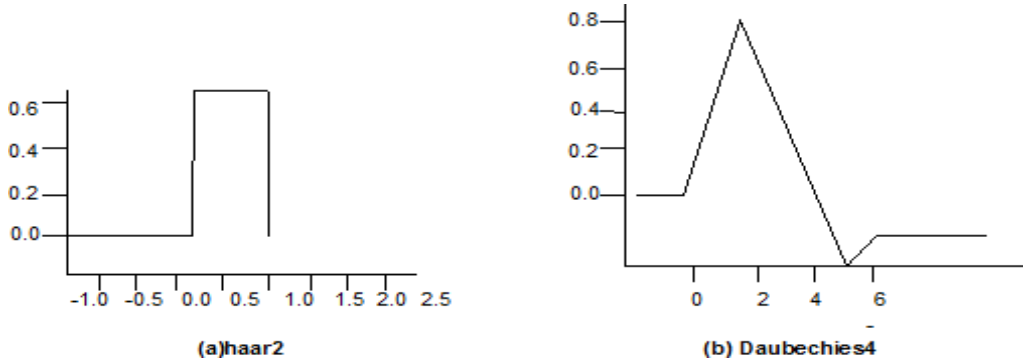
***b)* Principal components analysis.**

a) Wavelet transforms:

The discrete wavelet transform (DWT) is a linear signal processing technique that, whenapplied to a data vector, transforms it to a numerically different vector, of wavelet coefficients. The two vectors are of the same length. When applying this technique to data reduction, we consider each tuple as an n-dimensional data vector, that is, X=(x1,x2,.....,xn), depicting n measurements made on the tuple from n database attributes.

For example, all wavelet coefficients larger than some user-specified threshold can be retained. All other coefficients are set to 0. The resulting data representation is therefore very sparse, so that can take advantage of data sparsity are computationally very fast if performed in wavelet space.

The numbers next to a wave let name is the number of vanishing moment of the wavelet this is a set of mathematical relationships that the coefficient must satisfy and is related to numberof coefficients.



(a)haar2 (b) Daubechies4

1. The length, L, of the input data vector must be an integer power of 2. This conditioncan be met by padding the data vector with zeros as necessary (L >=n).
2. Each transform involves applying two functions
   - The first applies some data smoothing, such as a sum or weighted average.
   - The second performs a weighted difference, which acts to bring out the detailedfeatures of data.
3. The two functions are applied to pairs of data points in X, that is, to all pairs of measurements (X2i , X2i+1). This results in two sets of data of length *L/2*. In general,

these represent a smoothed or low-frequency version of the input data and high frequencycontent of it, respectively.

4. The two functions are recursively applied to the sets of data obtained in the previousloop, until the resulting data sets obtained are of length 2.

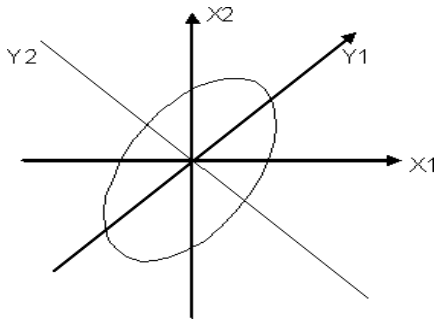
b) Principal components analysis

Suppose that the data to be reduced, which Karhunen-Loeve, K-L, method consists of tuples or data vectors describe by n attributes or dimensions. Principal components analysis, or PCA (also called the Karhunen-Loeve, or K-L, method), searches for k n-dimensional orthogonal vectors that can best be used to represent the data where *k*<=n. PCA combines the essence of attributes by creating an alternative, smaller set of variables. The initial data can then be projectedonto this smaller set.

The basic procedure is as follows:

- The input data are normalized.

- PCA computes *k* orthonormal vectors that provide a basis for the normalized input data. These are unit vectors that each point in a direction perpendicular to the others.
- The principal components are sorted in order of decreasing significance or strength.



In the above figure, Y1 and Y2, for the given set of data originally mapped to the axes X1 and X2. This information helps identify groups or patterns within the data. The sorted axes are such that the first axis shows the most variance among the data, the second axis shows the next highest variance, and so on.

- The size of the data can be reduced by eliminating the weaker components.

Advantage of PCA

- PCA is computationally inexpensive
- Multidimensional data of more than two dimensions can be handled by reducing theproblem to two dimensions.
- Principal components may be used as inputs to multiple regression and cluster analysis.

5. Data Transformation and Discretization

Data transformation is the process of converting data from one format or structure into another format or structure.

In *data transformation*, the data are transformed or consolidated into forms appropriate for mining.

Strategies for data transformation include the following:

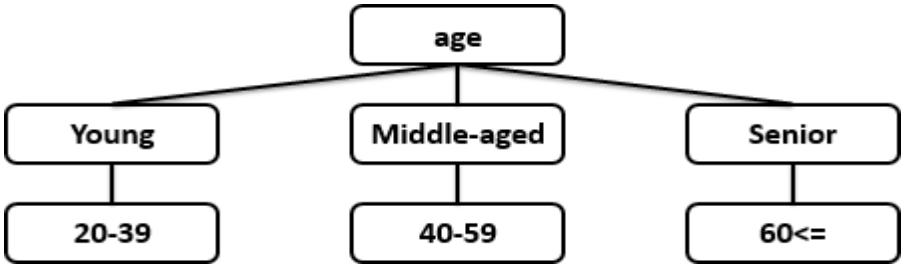
1. **Smoothing**, which works to remove noise from the data. Techniques include binning, regression, and clustering.
2. **Attribute construction** (or *feature construction*), where new attributes are constructed andadded from the given set of attributes to help the mining process.
3. **Aggregation**, where summary or aggregation operations are applied to

the data. For example, the daily sales data may be aggregated so as to compute monthly and annual total amounts. This step is typically used in constructing a data cube for data analysis at multiple abstraction levels.

4. Normalization, where the attribute data are scaled so as to fall within a smaller range, suchas 1.0 to 1.0, or 0.0 to 1.0.

5. Discretization, where the raw values of a numeric attribute (e.g., *age*) are replaced byinterval labels (e.g., 0–10, 11–20, etc.) or conceptual labels (e.g., *youth, adult, senior*).The labels, in turn, can be recursively organized into higher-level concepts, resulting in a *concept hierarchy* for the numeric attribute. Figure 3.12 shows a concept hierarchy for the attribute *price*. More than one concept hierarchy can be defined for the same attribute to accommodatethe needs of various users.

6. Concept hierarchy generation for nominal data, where attributes such as *street* can be generalized to higher-level concepts, like *city* or *country*. Many hierarchies for nominal attributes are implicit within the database schema and can be automatically defined at the schema definition level.



Data Transformation by Normalization:

The measurement unit used can affect the data analysis. For example, changing measurement units from meters to inches for *height*, or from kilograms to pounds for *weight*, may lead to very different results.

For distance-based methods, normalization helps prevent attributes with initially large ranges (e.g., *income*) from outweighing attributes with initially smaller ranges (e.g., binary attributes). It is also useful when given no prior knowledge of the data.

There are many methods for data normalization. We study *min-max normalization, z- score normalization,* and *normalization by decimal scaling.* For our discussion, let *A* be anumeric attribute with *n* observed values, v_1, v_2, …., v_n.

a) Min-max normalization performs a linear transformation on the original data. Suppose that min_A and max_A are the minimum and maximum values of

an attribute, *A*. Min- max normalization maps a value, v_i, of *A* to v_i'in the range [new_min_A, new_max_A]by computing

Min-max normalization preserves the relationships among the original data values.

It will encounter

$$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A.$$

an ―out-of-bounds‖ error if a future input case for normalization falls outside of the original data range for *A*.

Example:-Min-max normalization. Suppose that the minimum and maximum values for the attribute *income* are $12,000 and $98,000, respectively. We would like to map *income* to the range [0.0, 1.0]. By min-max normalization, a value of $73,600 for *income* is transformed to

$$\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716.$$

b) Z-Score Normalization

The values for an attribute, *A*, are normalized based on the mean (i.e.,

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A}$$

average) and standard deviation of *A*. A value, v_i, of *A* is normalized to v_i' by computing where *A* and σA are the mean and standard deviation, respectively, of attribute A. **Example** z-score normalization. Suppose that the mean and standard deviation of the values for the attribute income are $54,000 and $16,000, respectively. With z-score normalization, a value of $73,600 for income is transformed to

$$\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225.$$

c) Normalization by Decimal Scaling:

Normalization by decimal scaling normalizes by moving the decimal point of values of attribute A.The number of decimal points moved depends on the maximum absolute value of

A. A value, vi, of A is normalized to vi' by computing

$$v_i' = \frac{v_i}{10^j}$$

where j is the smallest integer such that max(|v_i'$)$< 1.

Example Decimal scaling. Suppose that the recorded values of A range from -986 to 917. The maximum absolute value of A is 986. To normalize by decimal scaling, we therefore divide each value by 1000 (i.e., j = 3) so that -986 normalizes to -0.986 and 917 normalizes to 0.917.

Data Discretization

a) *Discretization by binning:*

Binning is a top-down splitting technique based on a specified number of bins. For example, attribute values can be discretized by applying equal-width or equal-frequency binning, and then replacing each bin value by the bin mean or median, as in *smoothing by bin means* or *smoothing by bin medians*, respectively. These techniques can be applied recursively to the resulting partitions to generate concept hierarchies.

b) *Discretization by Histogram Analysis:*

Like binning, histogram analysis is an unsupervised discretization technique because it does not use class information. A histogram partitions the values of an attribute, A, into disjoint ranges called buckets or bins.

In an equal-width histogram, for example, the values are partitioned into equal-size partitions or ranges (e.g., for price, where each bucket has a width of $10). With an equal- frequency histogram, the values are partitioned so that, ideally, each partition contains the same number of data tuples.

c) *Discretization by Cluster, Decision Tree, and Correlation Analyses*

Cluster analysis is a popular data discretization method. A clustering algorithm can be applied to discretize a numeric attribute, A, by partitioning the values of A into clusters or groups. Techniques to generate decision trees for classification can be applied to discretization. Such techniques employ a top-down splitting approach. Unlike the other methods mentioned so far,

decision tree approaches to discretization are supervised, that is, they make use of class label

Concept Hierarchy Generation for Nominal Data

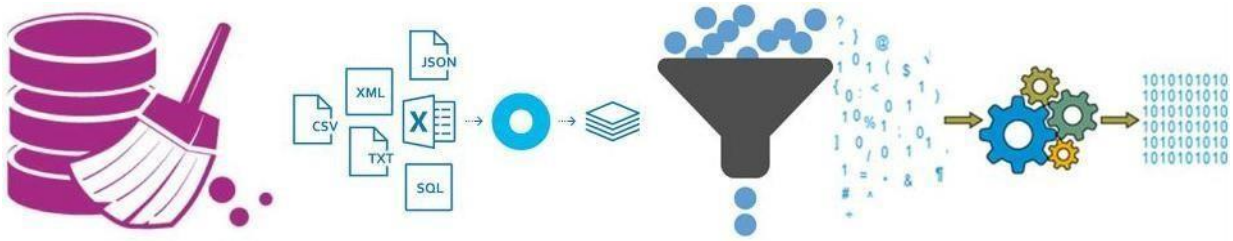
Nominal attributes have a finite (but possibly large) number of distinct values, with no ordering among the values. Examples include geographic location, job category, and item type.

Manual definition of concept hierarchies can be a tedious and time-consuming task for a user or a domain expert. Fortunately, many hierarchies are implicit within the database schema and can be automatically defined at the schema definition level. The concept hierarchies can be used to transform the data into multiple levels of granularity.

1. **Specification of a partial ordering of attributes explicitly at the schema level by users or experts:** A user or expert can easily define a concept hierarchy by specifyinga partial or total ordering of the attributes at the schema level.
2. **Specification of a portion of a hierarchy by explicit data grouping:** In a large database, it is unrealistic to define an entire concept hierarchy by explicit value enumeration. For example, after specifying that province and country form a hierarchyat the schema level, a user could define some intermediate levels manually.
3. **Specification of a set of attributes, but not of their partial ordering:** A user may specify a set of attributes forming a concept hierarchy, but omit to explicitly State their partial ordering. The system can then try to automatically generate the attribute ordering so as to construct a meaningful concept hierarchy.
4. **Specification of only a partial set of attributes:** Sometimes a user can be careless when defining a hierarchy, or have only a vague idea about what should be included in a hierarchy. Consequently, the user may have included only a small subset of there irrelevant attributes in the hierarchy specification.

- ✓ **Data cleaning** routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data.
- ✓ **Data integration** combines data from multiple sources to form a coherent data store. The resolution of semantic heterogeneity, metadata, correlation analysis ,tuple duplication detection, and data conflict detection contribute to smooth data integration.

- ✓ **Data reduction** techniques obtain a reduced representation of the data while minimizing the loss of information content. These include methods of *dimensionality reduction*, *numerosity reduction*, and *data compression*.
- ✓ **Data transformation** routines convert the data into appropriate forms for mining. For example, in **normalization**, attribute data are scaled so as to fall within a small range such as 0.0 to 1.0. Other examples are **data discretization** and **concept hierarchy generation**.
- ✓ **Data discretization** transforms numeric data by mapping values to interval or concept labels. Such methods can be used to automatically generate *concept hierarchies* for the data, which allows for mining at multiple levels of granularity.



UNIT - III

CLUSTERING

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 $k \leq n$. 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 oneor 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.

Advantages

- 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 clustersbased 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.

Data Mining Applications

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

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

Financial Data Analysis

The financial data in banking and financial industry is generally reliable and ofhigh quality which facilitates systematic data analysis and data mining. Some ofthe typical cases are as follows −

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

Retail Industry

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

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

- Design and Construction of data warehouses based on the benefits of data mining.
- Multidimensional analysis of sales, customers, products, time and region.
- Analysis of effectiveness of sales campaigns.

- Customer Retention.
- Product recommendation and cross-referencing of items.

Telecommunication Industry

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

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

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

Biological Data Analysis

In recent times, we have seen a tremendous growth in the field of biology such as genomics, proteomics, functional Genomics and biomedical research. Biological

data mining is a very important part of Bioinformatics. Following are the aspects in which data mining contributes for biological data analysis −

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

Other Scientific Applications

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

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

Intrusion Detection

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

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

Data Mining System Products

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

Choosing a Data Mining System

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

- **Data Types** − The data mining system may handle formatted text, record-based data, and relational data. The data could also be in ASCII text,

relational database data or data warehouse data. Therefore, we should check what exact format the data mining system can handle.

- **System Issues** − We must consider the compatibility of a data mining system with different operating systems. One data mining system may run on only one operating system or on several. There are also data mining systems that provide web-based user interfaces and allow XML data as input.
- **Data Sources** − Data sources refer to the data formats in which data mining system will operate. Some data mining system may work only on ASCII text files while others on multiple relational sources. Data mining system should also support ODBC connections or OLE DB for ODBC connections.
- **Data Mining functions and methodologies** − There are some data mining systems that provide only one data mining function such as classification while some provides multiple data mining functions such as concept description, discovery-driven OLAP analysis, association mining, linkage analysis, statistical analysis, classification, prediction, clustering, outlier analysis, similarity search, etc.
- **Coupling data mining with databases or data warehouse systems** − Data mining systems need to be coupled with a database or a data warehouse system. The coupled components are integrated into a uniform information processing environment. Here are the types of coupling listed below −
    - No coupling
    - Loose Coupling
    - Semi tight Coupling
    - Tight Coupling

- **Scalability** − There are two scalability issues in data mining −
  - **Row (Database size) Scalability** − A data mining system is considered as row scalable when the number or rows are enlarged 10 times. It takes no more than 10 times to execute a query.
  - **Column (Dimension) Salability** − A data mining system is considered as column scalable if the mining query execution time increases linearly with the number of columns.
- **Visualization Tools** − Visualization in data mining can be categorized as follows −
  - Data Visualization
  - Mining Results Visualization
  - Mining process visualization
  - Visual data mining
- **Data Mining query language and graphical user interface** − An easy-to-use graphical user interface is important to promote user-guided, interactive data mining. Unlike relational database systems, data mining systems do not share underlying data mining query language.

Trends in Data Mining

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

- Application Exploration.
- Scalable and interactive data mining methods.
- Integration of data mining with database systems, data warehouse systems and web database systems.
- Standardization of data mining query language.
- Visual data mining.

- New methods for mining complex types of data.
- Biological data mining.
- Data mining and software engineering.
- Web mining.
- Distributed data mining.
- Real time data mining.
- Multi database data mining.
- Privacy protection and information security in data mining.

UNIT - IV
DATA WAREHOUSING

INTRODUCTION

A data warehouse is constructed by integrating data from multiple heterogeneous sources. It supports analytical reporting, structured and/or ad hoc queries and decision making. This tutorial adopts a step-by-step approach to explain all the necessary concepts of data warehousing.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'soperational database.
- There is no frequent updating done in a data warehouse.
- It possesses consolidated historical data, which helps the organization to analyze itsbusiness.
- A data warehouse helps executives to organize, understand, and use their data to takestrategic 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 to read and modify 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.

Note: A data warehouse does not require transaction processing, recovery, and concurrency controls, because it is physically stored and separate from the operational database.

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

Integrating Heterogeneous Databases

To integrate heterogeneous databases, we have two approaches:

- Query-driven Approach
- Update-driven Approach

Query-Driven Approach

This is the traditional approach to integrate heterogeneous databases. This approach was used to build wrappers and integrators on top of multiple heterogeneous databases. These integrators are also known as mediators.

Process of Query-Driven Approach

- When a query is issued to a client side, a metadata dictionary translates the query into anappropriate form for individual heterogeneous sites involved.
- Now these queries are mapped and sent to the local query processor.
- The results from heterogeneous sites are integrated into a global answer set.

Disadvantages

- Query-driven approach needs complex integration and filtering processes.
- This approach is very inefficient.
- It is very expensive for frequent queries.
- This approach is also very expensive for queries that require aggregations.

Update-Driven Approach

This is an alternative to the traditional approach. Today's data warehouse systems follow update- driven approach rather than the traditional approach discussed earlier. In update-driven approach, the information from multiple heterogeneous sources are integrated in advance and are stored in a warehouse. This information is available for direct querying and analysis.

Advantages

This approach has the following advantages:

- This approach provide high performance.
- The data is copied, processed, integrated, annotated, summarized and restructured insemantic data store in advance.
- Query processing does not require an interface to process data at local sources.

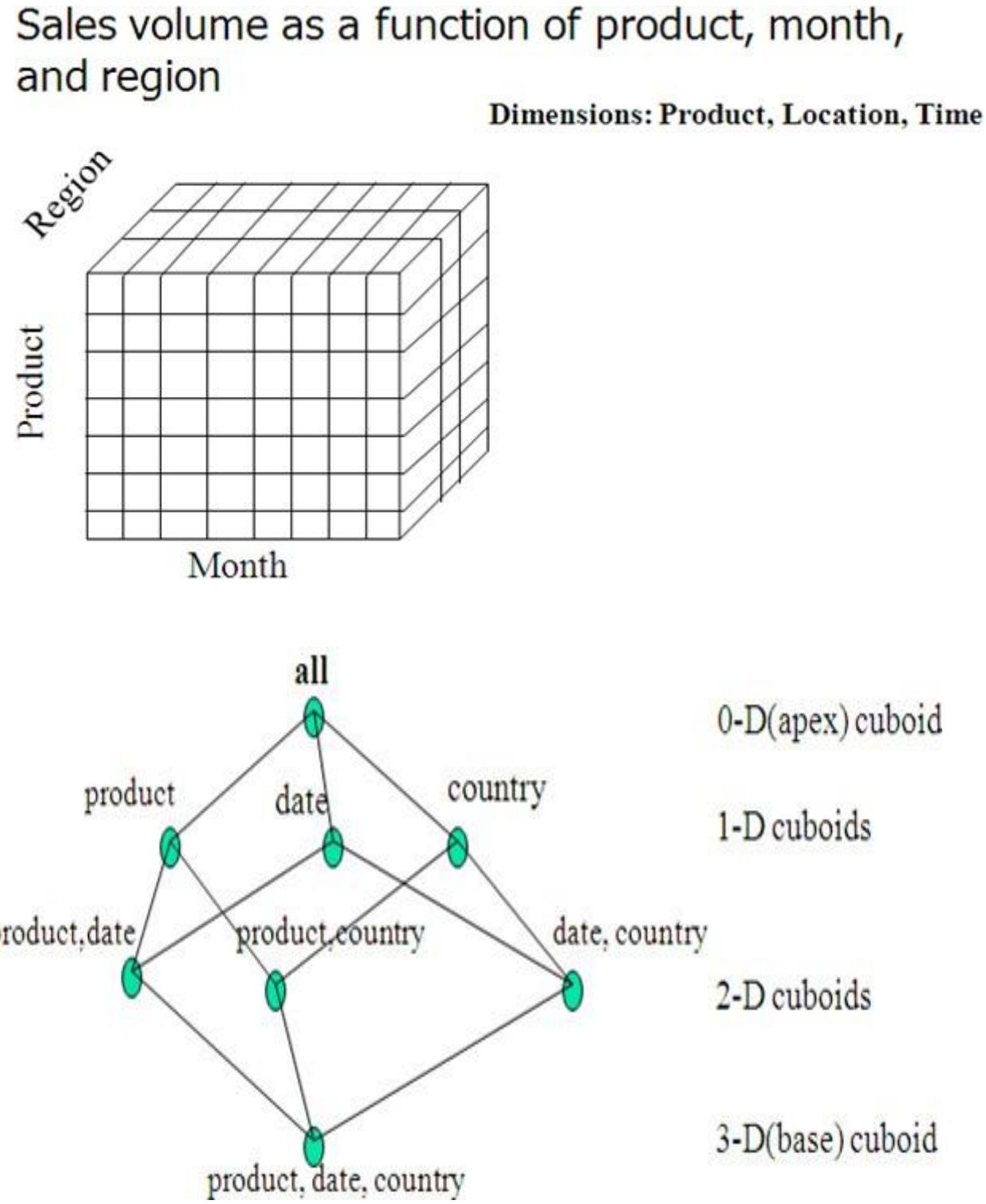
Functions of Data Warehouse Tools and Utilities

The following are the functions of data warehouse tools and utilities:

- **Data Extraction** - Involves gathering data from multiple heterogeneous sources.
- **Data Cleaning** - Involves finding and correcting the errors in data.
- **Data Transformation** - Involves converting the data from legacy format to warehouseformat.
- **Data Loading** - Involves sorting, summarizing, consolidating, checking integrity, andbuilding indices and partitions.
- **Refreshing** - Involves updating from data sources to warehouse.

Note: Data cleaning and data transformation are important steps in improving the quality of dataand data mining results.

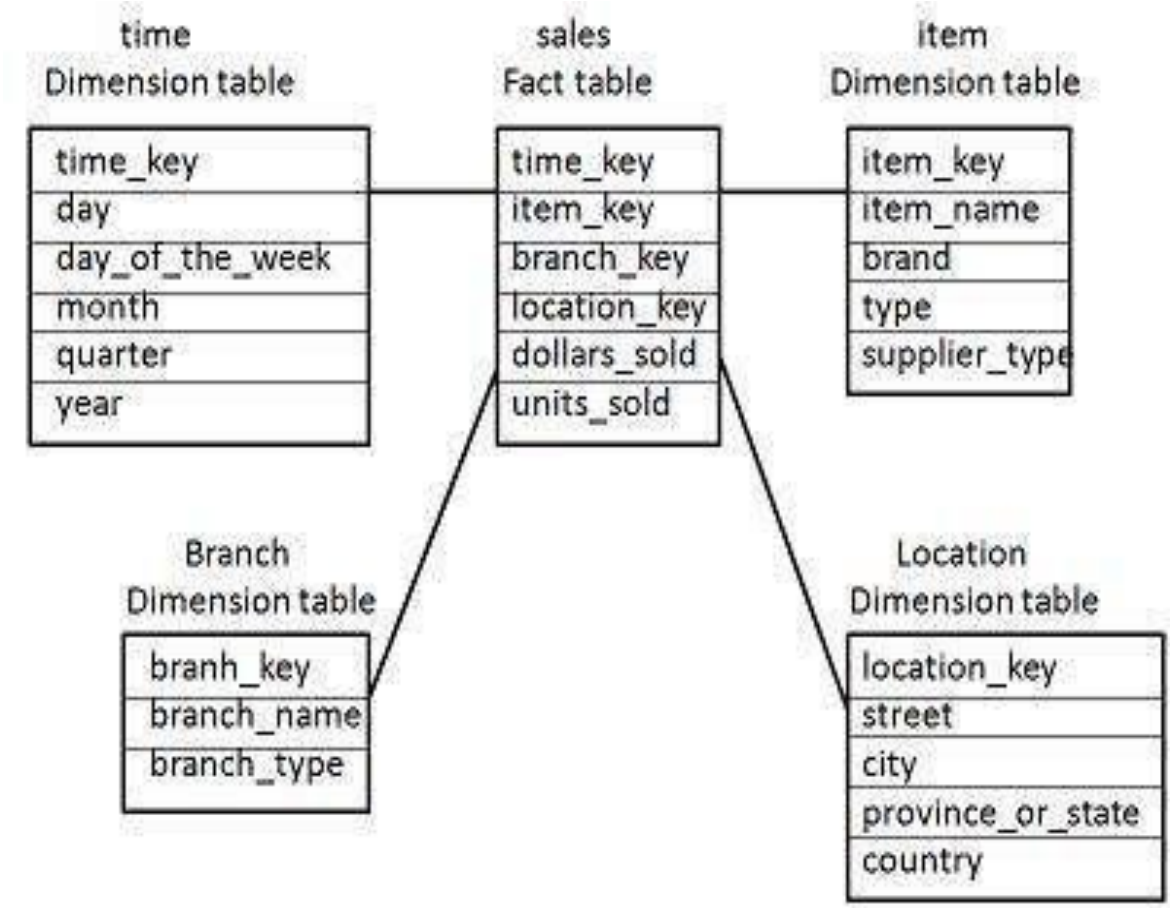
Multidimensional data model



Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema. In this chapter, we will discuss the schemas used in a data warehouse.

Star Schema

- Each dimension in a star schema is represented with only one-dimension table.
- This dimension table contains the set of attributes.
- The following diagram shows the sales data of a company with respect to the fourdimensions, namely time, item, branch, and location.

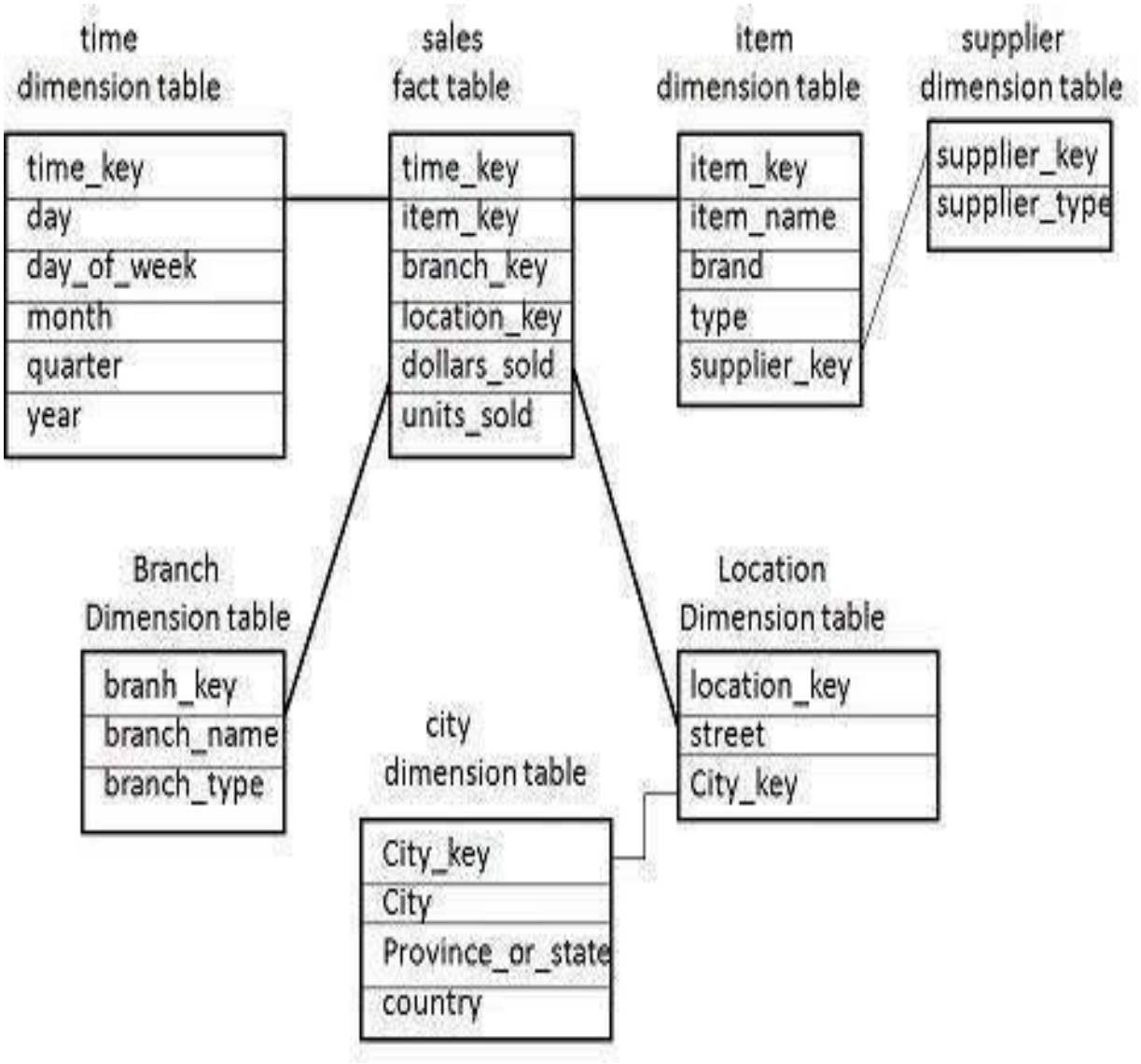


- There is a fact table at the center. It contains the keys to each of four dimensions.
- The fact table also contains the attributes, namely dollars sold and units sold.

Note: Each dimension has only one dimension table and each table holds a set of attributes. For example, the location dimension table contains the attribute set {location_key, street, city, province_or_state,country}. This constraint may cause data redundancy. For example, "Vancouver" and "Victoria" both the cities are in the Canadian province of British Columbia. The entries for such cities may cause data redundancy along the attributes province_or_state and country.

Snowflake Schema

- Some dimension tables in the Snowflake schema are normalized.
- The normalization splits up the data into additional tables.
- Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.

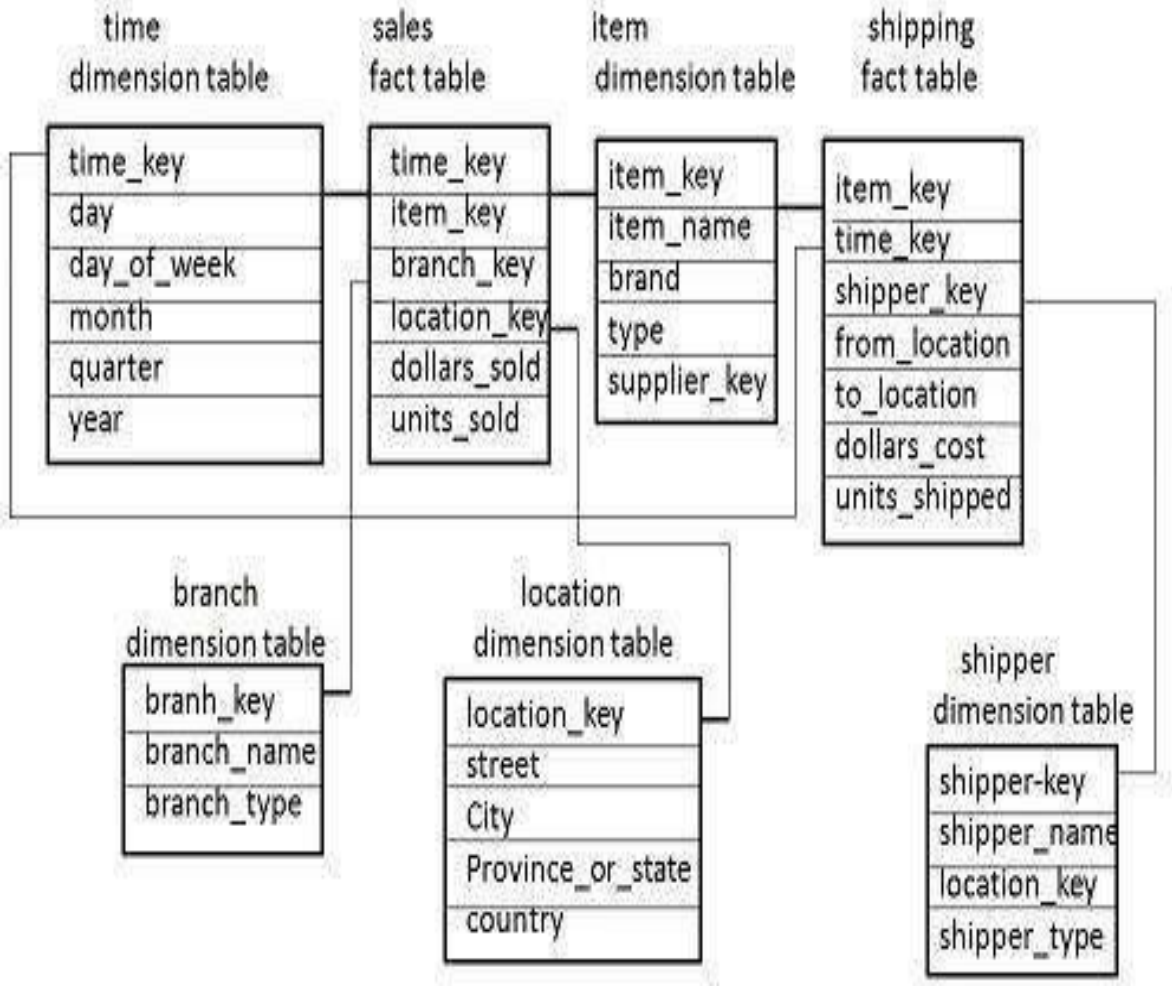


- Now the item dimension table contains the attributes item_key, item_name, type, brand,and supplier-key.
- The supplier key is linked to the supplier dimension table. The supplier dimension tablecontains the attributes supplier_key and supplier_type.

<b<>Note: Due to normalization in the Snowflake schema, the redundancy is reduced andtherefore, it becomes easy to maintain and the save storage space.</b<>

Fact Constellation Schema

- A fact constellation has multiple fact tables. It is also known as galaxy schema.
- The following diagram shows two fact tables, namely sales and shipping.



- The sales fact table is same as that in the star schema.
- The shipping fact table has the five dimensions, namely item_key, time_key, shipper_key, from_location, to_location.
- The shipping fact table also contains two measures, namely dollars sold and units sold.
- It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.

Schema Definition

Multidimensional schema is defined using Data Mining Query Language (DMQL). The two primitives, cube definition and dimension definition, can be used for defining the data warehouses and data marts.

Syntax for Cube Definition

define cube < cube_name > [< dimension-list > }: < measure_list >

Syntax for Dimension Definition

define dimension < dimension_name > as (< attribute_or_dimension_list >)

Star Schema Definition

The star schema that we have discussed can be defined using Data Mining Query Language(DMQL) as follows:

define cube sales star [time, item, branch, location]:
dollars sold = sum(sales in dollars), units sold = count(*)
define dimension time as (time key, day, day of week, month,
quarter, year)define dimension item as (item key, item name,
brand, type, supplier type) define dimension branch as (branch
key, branch name, branch type)
define dimension location as (location key, street, city, province or state, country)

Snowflake Schema Definition

Snowflake schema can be defined using DMQL as follows:

define cube sales snowflake [time, item, branch, location]:
dollars sold = sum(sales in dollars), units sold = count(*)
define dimension time as (time key, day, day of week, month, quarter, year)
define dimension item as (item key, item name, brand, type, supplier (supplier key, suppliertype))
define dimension branch as (branch key, branch name, branch type)
define dimension location as (location key, street, city (city key, city, province or state, country))

Fact Constellation Schema Definition

Fact constellation schema can be defined using DMQL as follows:

define cube sales [time, item, branch, location]:
dollars sold = sum(sales in dollars), units sold = count(*)
define dimension time as (time key, day, day of week, month,
quarter, year)define dimension item as (item key, item name,
brand, type, supplier type) define dimension branch as (branch
key, branch name, branch type)
define dimension location as (location key, street, city, province or
state,country)define cube shipping [time, item, shipper, from
location, to location]:
dollars cost = sum(cost in dollars), units shipped =
count(*)define dimension time as time in cube
sales
define dimension item as item in cube sales
define dimension shipper as (shipper key, shipper name, location as location in cube sales,shipper type)
define dimension from location as location in
cube salesdefine dimension to location as
location in cube sales

Data Warehouse Architecture

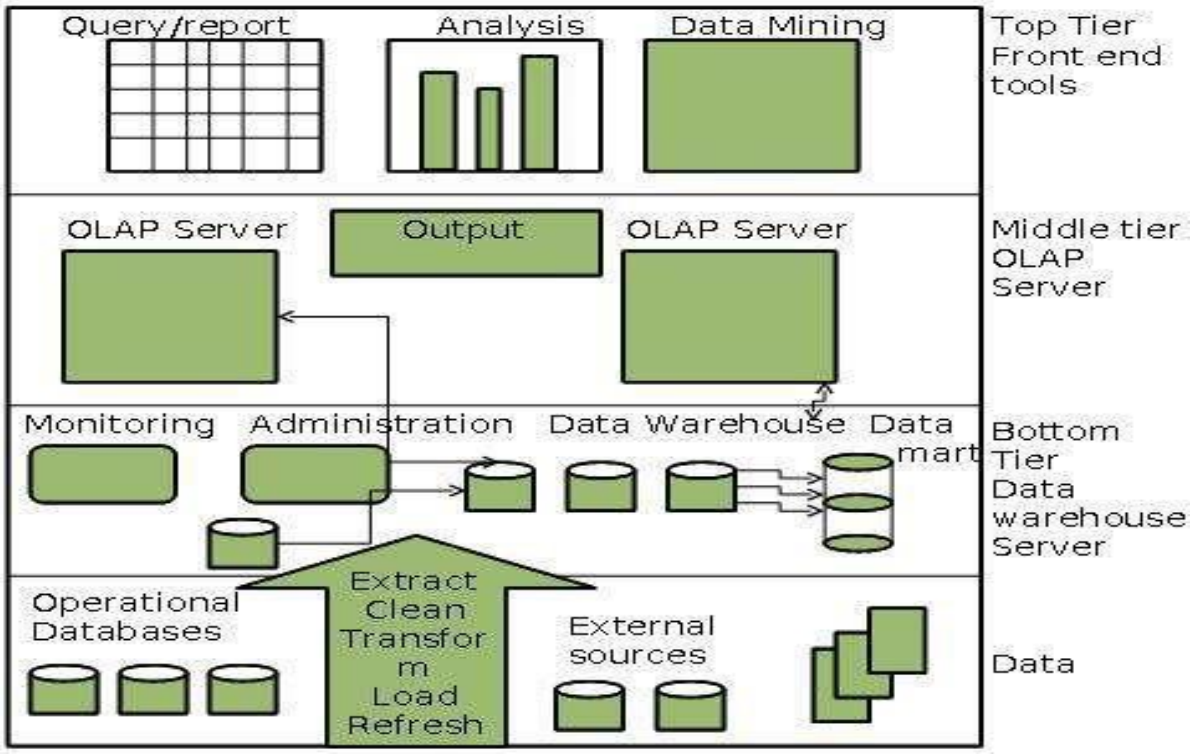
Three-Tier Data Warehouse Architecture

Generally a data warehouses adopts a three-tier architecture. Following are the three tiers of thedata warehouse architecture.

- **Bottom Tier** - The bottom tier of the architecture is the data warehouse database server. It is the relational database system. We use the back end tools and utilities to feed data into the bottom tier. These back end tools and utilities perform the Extract, Clean, Load, and refresh functions.
- **Middle Tier** - In the middle tier, we have the OLAP Server that can be implemented in either of the following ways.
  - By Relational OLAP (ROLAP), which is an extended relational database management system. The ROLAP maps the operations on multidimensional data to standard relational operations.
  - By Multidimensional OLAP (MOLAP) model, which directly implements the multidimensional data and operations.
- **Top-Tier** - This tier is the front-end client layer. This layer holds the query

tools and reporting tools, analysis tools and data mining tools.

The following diagram depicts the three-tier architecture of data warehouse:



Data Warehouse Models

From the perspective of data warehouse architecture, we have the following data warehousemodels:

- Virtual Warehouse
- Data mart
- Enterprise Warehouse

Virtual Warehouse

The view over an operational data warehouse is known as a virtual warehouse. It is easy to build a virtual warehouse. Building a virtual warehouse requires excess capacity on operational database servers.

Data Mart

Data mart contains a subset of organization-wide data. This subset of data is valuable to specific groups of an organization.

In other words, we can claim that data marts contain data specific to a particular group. For example, the marketing data mart may contain data related to items, customers, and sales. Data marts are confined to subjects.

Points to remember about data marts:

- Window-based or Unix/Linux-based servers are used to implement data marts. They areimplemented on low-cost servers.
- The implementation data mart cycles is measured in short periods of time, i.e., in weeksrather than months or years.
- The life cycle of a data mart may be complex in long run, if its planning and design arenot organization-wide.
- Data marts are small in size.
- Data marts are customized by department.
- The source of a data mart is departmentally structured data warehouse.
- Data mart are flexible.

Enterprise Warehouse

- An enterprise warehouse collects all the information and the subjects spanning an entireorganization
- It provides us enterprise-wide data integration.
- The data is integrated from operational systems and external information providers.
- This information can vary from a few gigabytes to hundreds of gigabytes, terabytes orbeyond.

Data Warehouse Implementation

Data warehouses contain huge volumes of data. OLAP servers demand that decision support queries be answered in the order of seconds. Therefore, it is crucial for data warehouse systemsto support highly efficient cube computation techniques, access methods, and query processingtechniques. In this section, we present an overview of methods for the efficient implementationof data warehouse systems.

Efficient Computation of Data Cubes

At the core ofmultidimensional data analysis is the efficient computation of aggregations acrossmany sets of dimensions. In SQL terms, these aggregations are referred to as group-by's. Each group-by can be represented by a *cuboid*, where the set of group-by's forms a lattice of cuboidsdefining a data cube. In this section, we explore issues relating to the efficient computation of data cubes.

The compute cube Operator and the Curse of Dimensionality

One approach to cube computation extends SQL so as to include a compute cube operator. Thecompute cube operator computes aggregates over all subsets of the dimensions specified in theoperation. This can require excessive storage space, especially for large numbers of dimensions.We start with an intuitive look at what is involved in the efficient computation of data cubes.

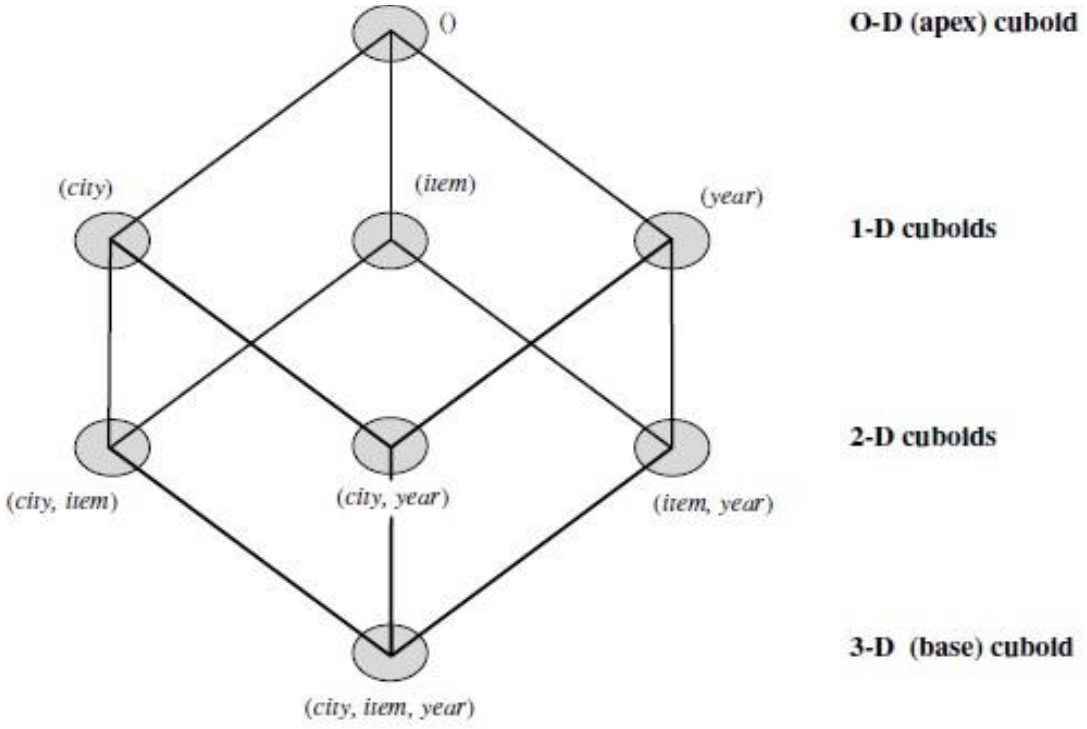
Example 3.11 A data cube is a lattice of cuboids. Suppose that you would like to create a datacube for

AllElectronics sales that contains the following: *city*, *item*, *year*, and *sales in dollars*. You wouldlike to be able to analyze the data, with queries such as the following:

"*Compute the sum of sales, grouping by city and item.*""*Compute the sum of sales, grouping by city*."

What is the total number of cuboids, or group-by's, that can be computed for this data cube? Taking the three attributes, *city, item*, and *year*, as the dimensions for the data cube, and *sales_in_dollars* as the measure, the total number of cuboids, or group-by's, that can be computed for this data cube is $2^3 = 8$. The possible group-by's are the following: {(*city, item, year*), (*city, item*), (*city, year*), (*item, year*), (*city*), (*item*), (*year*), ()}, where () means that the group-by is empty (i.e., the dimensions are not grouped). These group-by's form a lattice of cuboids for the data cube, as shown in Figure 3.14. The **base cuboid** contains all three dimensions, *city, item*, and *year*. It can return the total sales for any combination of the three dimensions. The **apex cuboid**, or 0-D cuboid, refers to the case where the group-by is empty. It contains the total sum of all sales. The base cuboid is the least generalized (most specific) of the cuboids. The apex cuboid is the most generalized (least specific) of the cuboids, and is often denoted as **all**. If we start at the apex cuboid and explore downward in the lattice, this is equivalent to drilling down within the data cube. If we start at the base cuboid and explore upward, this is akin to rolling up.

An SQL query containing no group-by, such as "compute the sum of total sales," is a *zero-dimensional operation*. An SQL query containing one group-by, such as "compute the sum of sales, group by city," is a *one-dimensional operation*. A cube operator on *n* dimensions is equivalent to a collection of **group by** statements, one for each subset



Lattice of cuboids, making up a 3-D data cube. Each cuboid represents a different group-by. The base cuboid contains the three dimensions *city, item*, and *year*.

of the n dimensions. Therefore, the cube operator is the n-dimensional generalization of the **group by** operator.

Based on the syntax of DMQL introduced in Section 3.2.3, the data cube in Example 3.11 could be defined as

define cube sales_cube [city, item, year]: **sum**(sales_in_dollars)

For a cube with n dimensions, there are a total of 2^n cuboids, including the base cuboid. A statement such as

compute cube sales_cube

would explicitly instruct the system to compute the sales aggregate cuboids for all of the eight subsets of the set {*city, item, year*}, including the empty subset. A cube computation operator was first proposed and studied by Gray et al. [GCB+97].

On-line analytical processing may need to access different cuboids for different queries Therefore, it may seem like a good idea to compute all or at least some of the cuboids in a data cube in advance. Precomputation leads to fast response time and avoids some redundant computation. Most, if not all, OLAP products resort to some degree of pre-computation of multidimensional aggregates.

A major challenge related to this precomputation, however, is that the required storage space may explode if all of the cuboids in a data cube are precomputed, especially when the cube has many dimensions. The storage requirements are even more excessive when many of the dimensions have associated concept hierarchies, each with multiple levels. This problem is referred to as the **curse of dimensionality.**

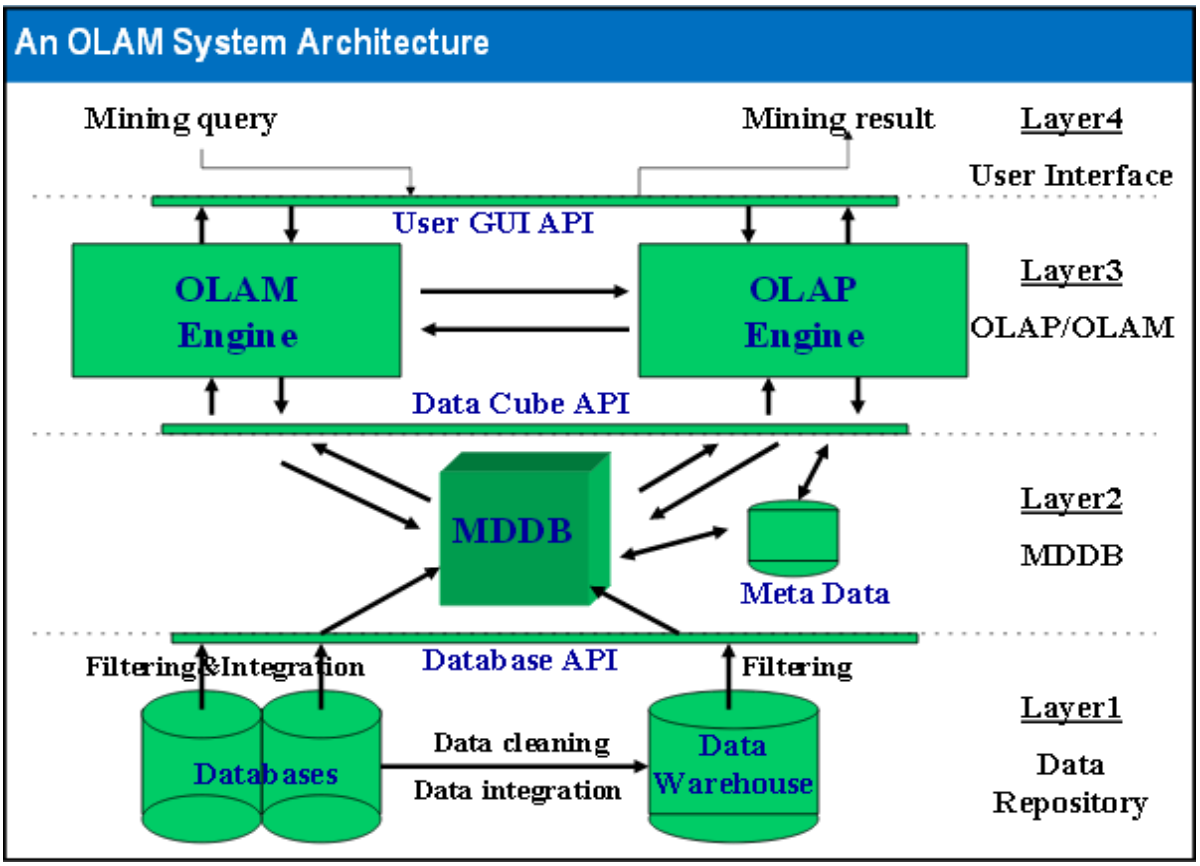
From Data warehousing to Data Mining

From OLAP to On Line Analytical Mining (OLAM)

Why online analytical mining?

- High quality of data in data warehouses
  - DW contains integrated, consistent, cleaned data
- Available information processing structure surrounding data warehouses
  - ODBC, OLEDB, Web accessing, service facilities, reporting and OLAP tools
- OLAP-based exploratory data analysis
  - Mining with drilling, dicing, pivoting, etc.
- On-line selection of data mining functions
  - Integration and swapping of multiple mining functions, algorithms, and tasks

An OLAM System Architecture



UNIT - V

OLAP- Need - Categorization of OLAP Operations

Online Analytical Processing Server (OLAP) is based on the multidimensional data model. It allows managers, and analysts to get an insight of the information through fast, consistent, and interactive access to information. This chapter cover the types of OLAP, operations on OLAP, difference between OLAP, and statistical databases and OLTP.

Types of OLAP Servers

We have four types of OLAP servers:

- Relational OLAP (ROLAP)
- Multidimensional OLAP (MOLAP)
- Hybrid OLAP (HOLAP)
- Specialized SQL Servers

Relational OLAP

ROLAP servers are placed between relational back-end server and client front-end tools. To store and manage warehouse data, ROLAP uses relational or extended-relational DBMS.

ROLAP includes the following:

- Implementation of aggregation navigation logic.
- Optimization for each DBMS back end.
- Additional tools and services.

Multidimensional OLAP

MOLAP uses array-based multidimensional storage engines for multidimensional views of data. With multidimensional data stores, the storage utilization may be low if the data set is sparse. Therefore, many MOLAP server use two levels of data storage representation to handle dense and sparse data sets.

Hybrid OLAP (HOLAP)

Hybrid OLAP is a combination of both ROLAP and MOLAP. It offers higher scalability of ROLAP and faster computation of MOLAP. HOLAP servers allows to store the large data volumes of detailed information. The aggregations are stored separately in MOLAP store.

Specialized SQL Servers

Specialized SQL servers provide advanced query language and query processing support for SQL queries over star and snowflake schemas in a read-only environment.

OLAP Operations

Since OLAP servers are based on multidimensional view of data, we will discuss OLAPoperations in multidimensional data.

Here is the list of OLAP operations:

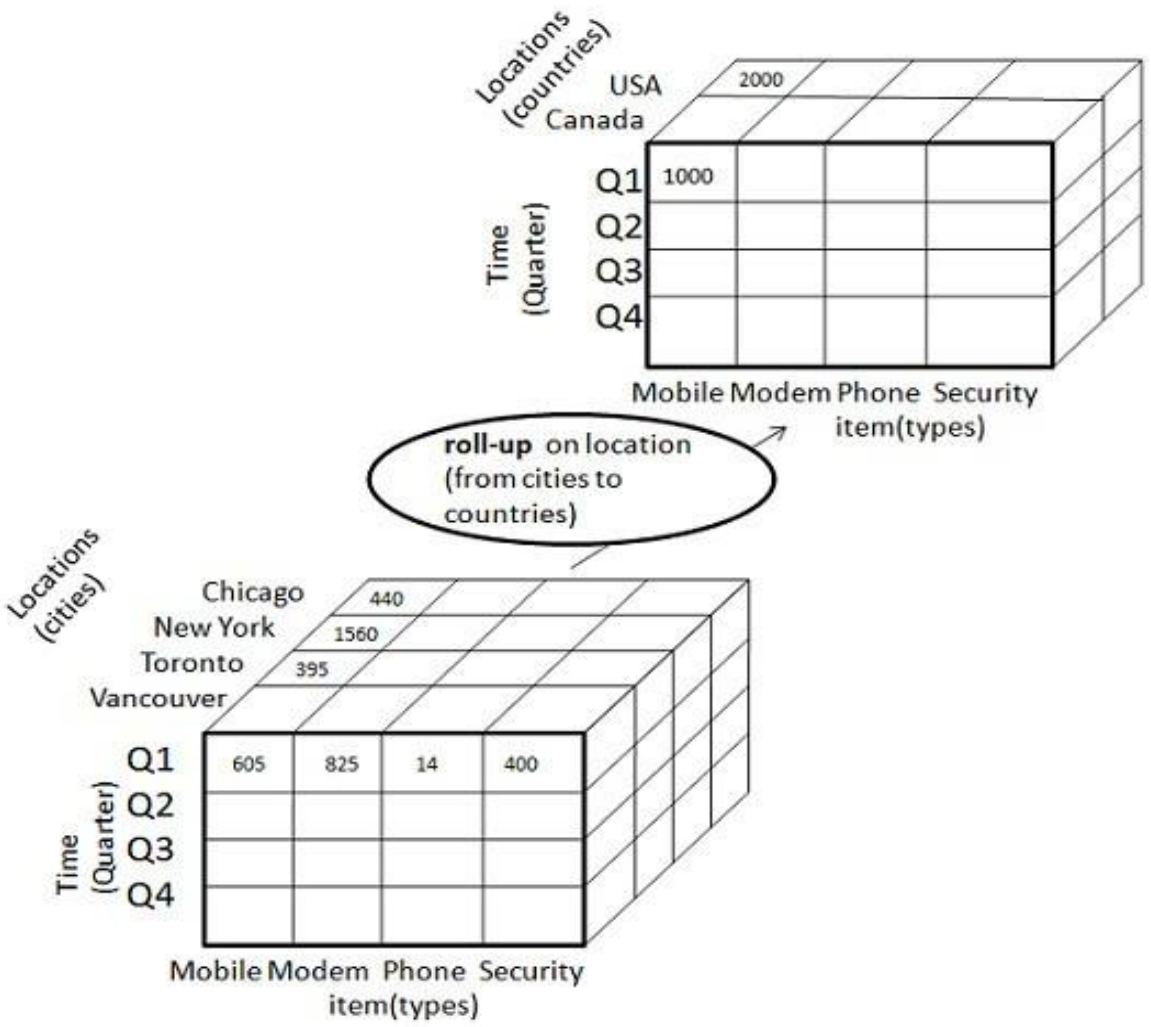
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

Roll-up

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

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

The following diagram illustrates how roll-up works.



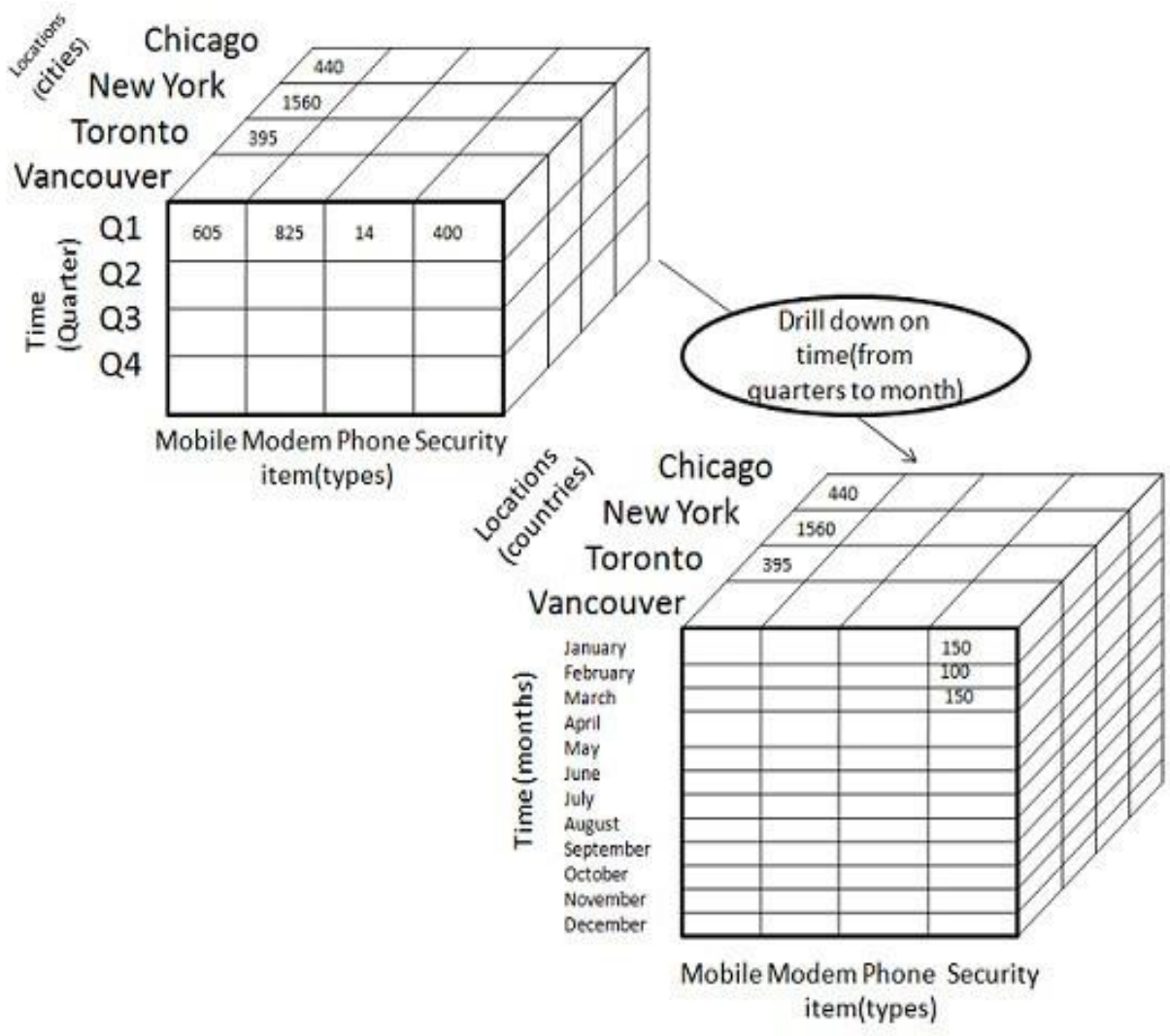
- Roll-up is performed by climbing up a concept hierarchy for the dimension location.
- Initially the concept hierarchy was "street < city < province < country".
- On rolling up, the data is aggregated by ascending the location hierarchy from the level ofcity to the level of country.
- The data is grouped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

Drill-down

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

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

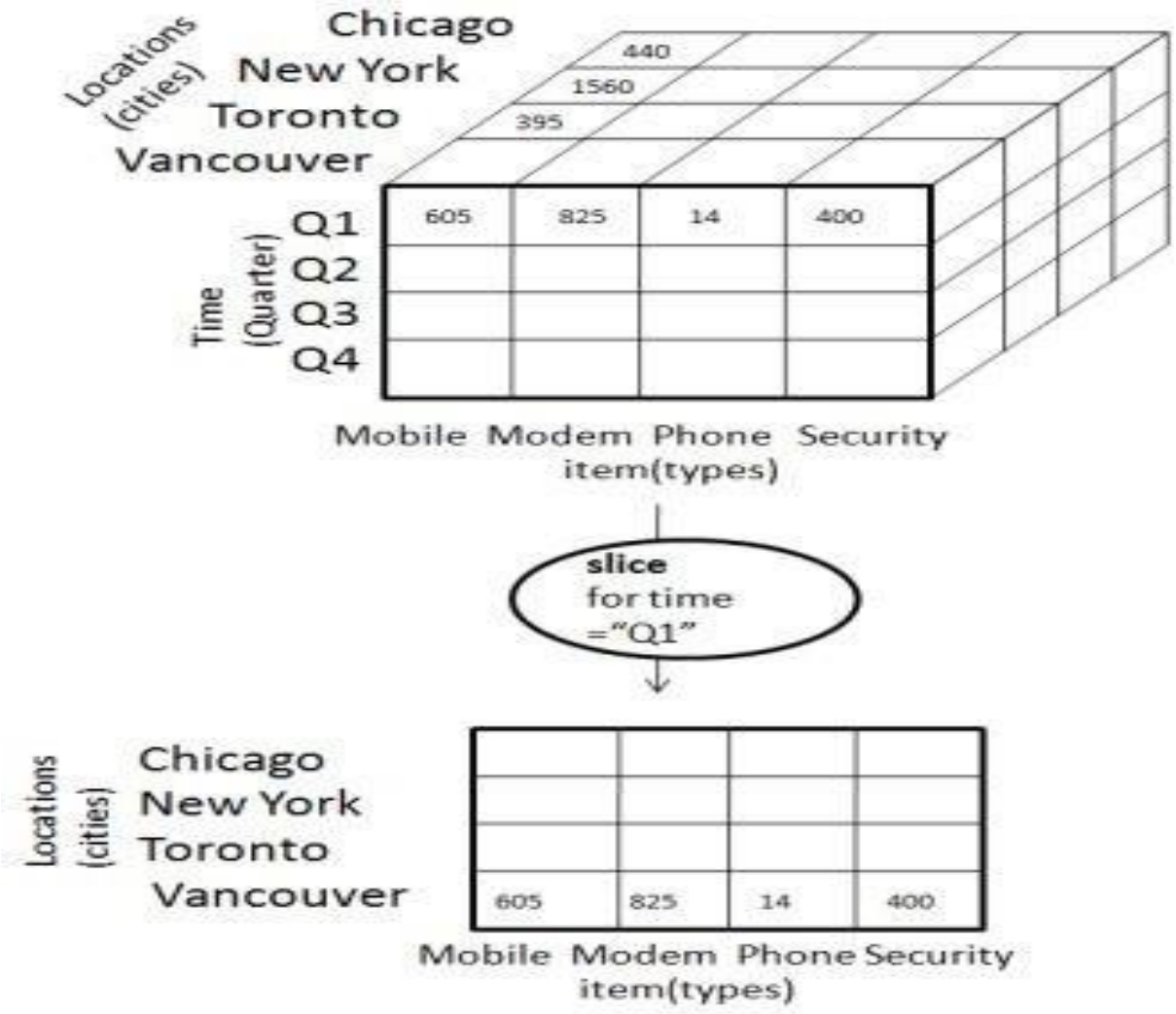
The following diagram illustrates how drill-down works:



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

Slice

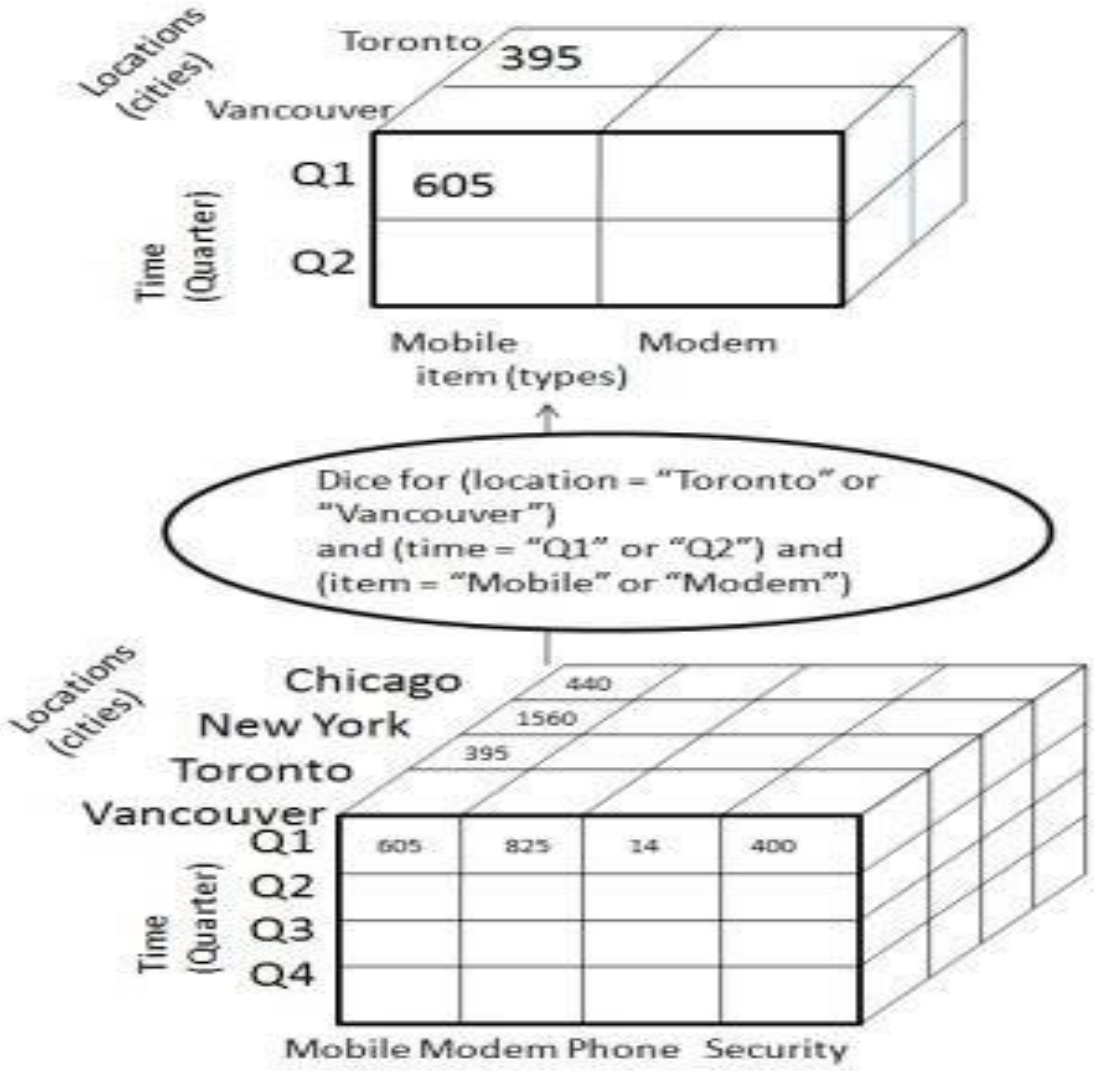
The slice operation selects one particular dimension from a given cube and provides a new sub-cube. Consider the following diagram that shows how slice works.



- Here Slice is performed for the dimension "time" using the criterion time = "Q1".
- It will form a new sub-cube by selecting one or more dimensions.

Dice

Dice selects two or more dimensions from a given cube and provides a new sub-cube. Consider the following diagram that shows the dice operation.

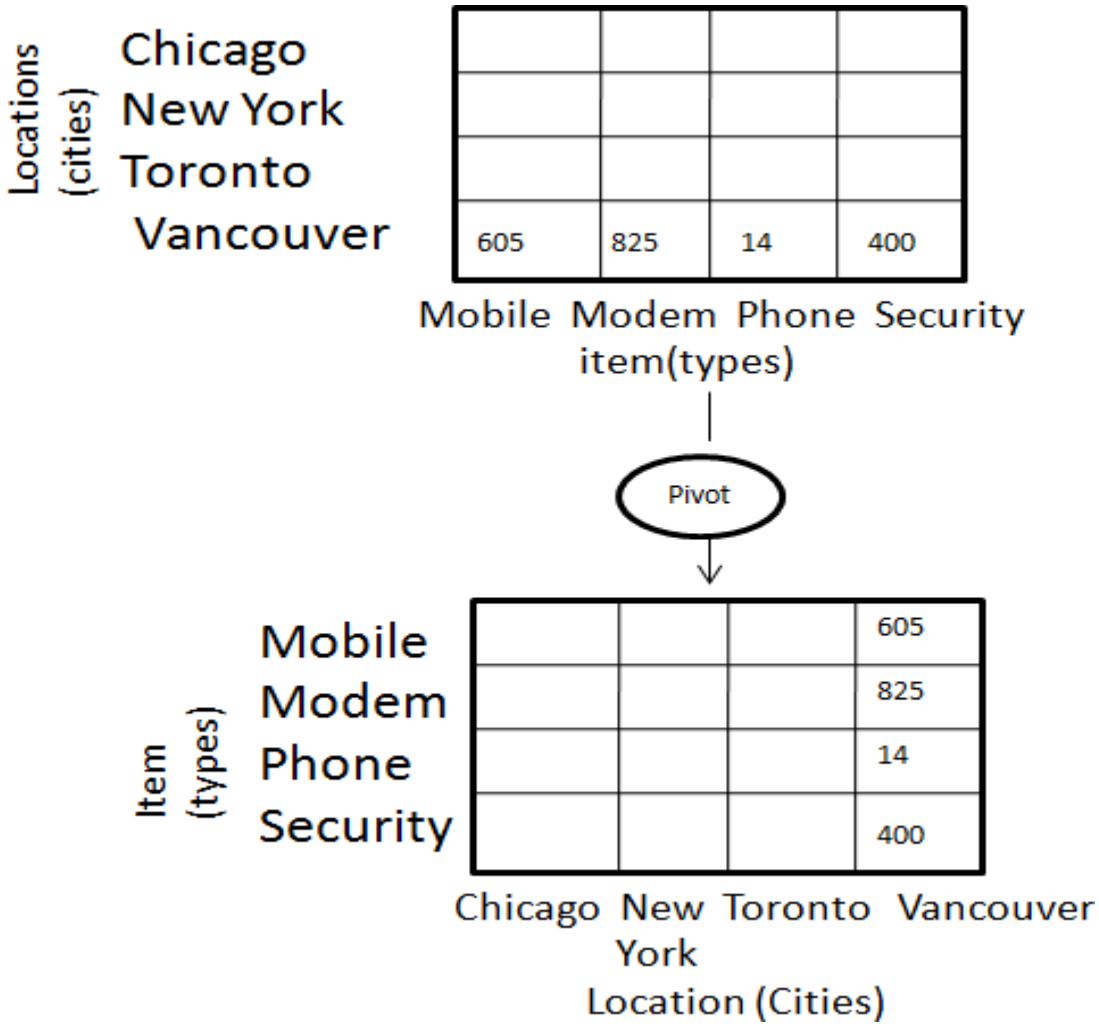


The dice operation on the cube based on the following selection criteria involves three dimensions.

- (location = "Toronto" or "Vancouver")
- (time = "Q1" or "Q2")
- (item =" Mobile" or "Modem")

Pivot

The pivot operation is also known as rotation. It rotates the data axes in view in order to provide an alternative presentation of data. Consider the following diagram that shows the pivot operation.



OLAP

Online Analytical Processing Server (OLAP) is based on the multidimensional data model. It allows managers, and analysts to get an insight of the information through fast, consistent, and interactive access to information. This chapter cover the types of OLAP, operations on OLAP, difference between OLAP, and statistical databases and OLTP.

Types of OLAP Servers

We have four types of OLAP servers −

- Relational OLAP (ROLAP)
- Multidimensional OLAP (MOLAP)
- Hybrid OLAP (HOLAP)
- Specialized SQL Servers

Relational OLAP

ROLAP servers are placed between relational back-end server and client front-end tools. To store and manage warehouse data, ROLAP uses relational or extended-relational DBMS.

ROLAP includes the following −

- Implementation of aggregation navigation logic.
- Optimization for each DBMS back end.
- Additional tools and services.

Multidimensional OLAP

MOLAP uses array-based multidimensional storage engines for multidimensional views of data. With multidimensional data stores, the storage utilization may be low if the data set is sparse. Therefore, many MOLAP server use two levels of data storage representation to handle dense and sparse data sets.

Hybrid OLAP

Hybrid OLAP is a combination of both ROLAP and MOLAP. It offers higher scalability of ROLAP and faster computation of MOLAP. HOLAP servers allows to store the large data volumes of detailed information. The aggregations are stored separately in MOLAP store.

Specialized SQL Servers

Specialized SQL servers provide advanced query language and query processing support for SQL queries over star and snowflake schemas in a read-only environment.

OLAP Operations

Since OLAP servers are based on multidimensional view of data, we will discuss OLAP operations in multidimensional data.

Here is the list of OLAP operations −

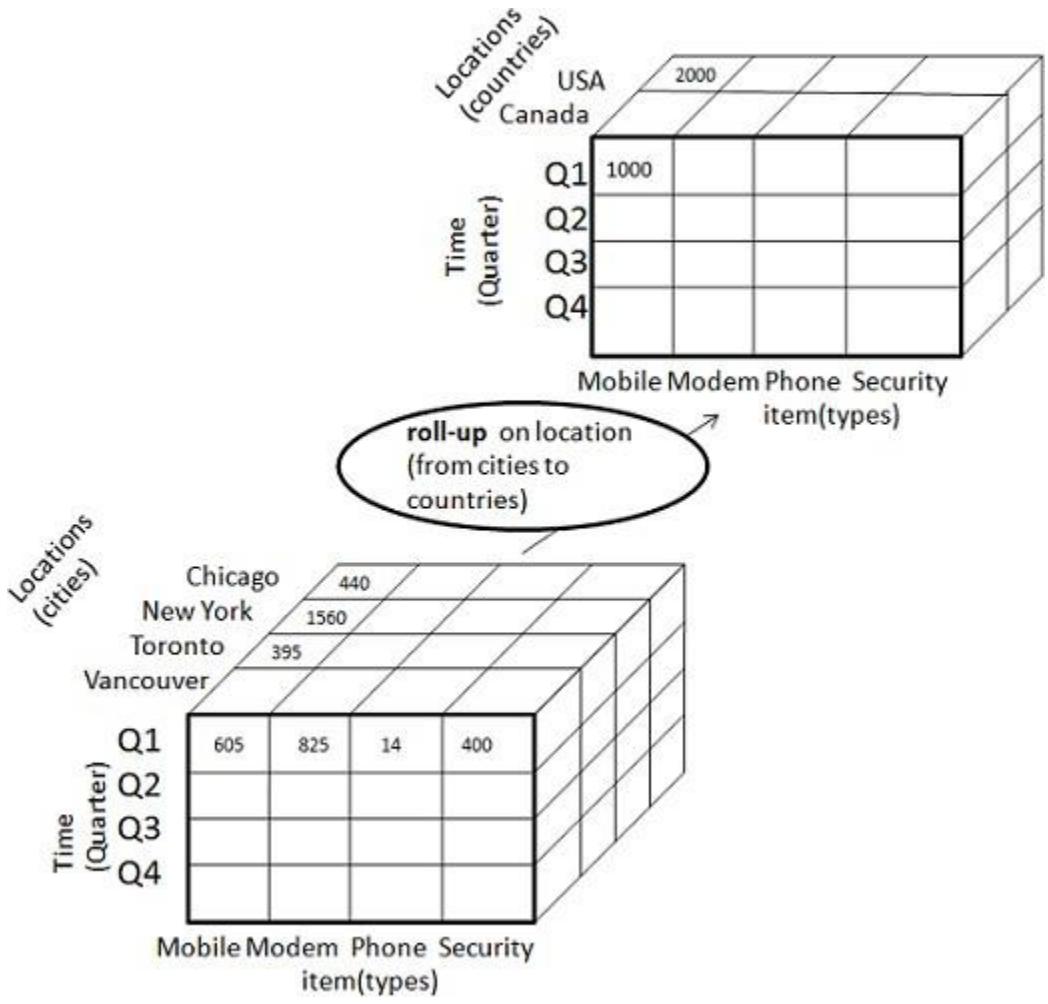
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

Roll-up

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

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

The following diagram illustrates how roll-up works.

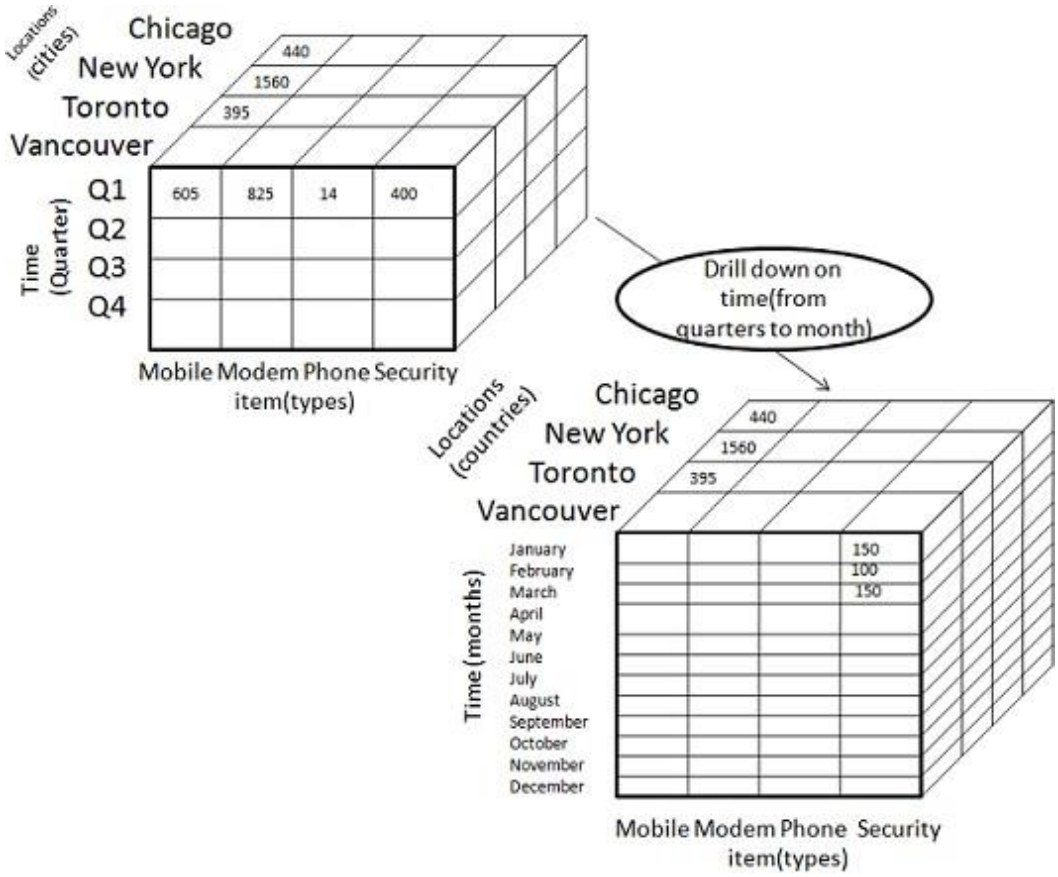


Drill-down

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

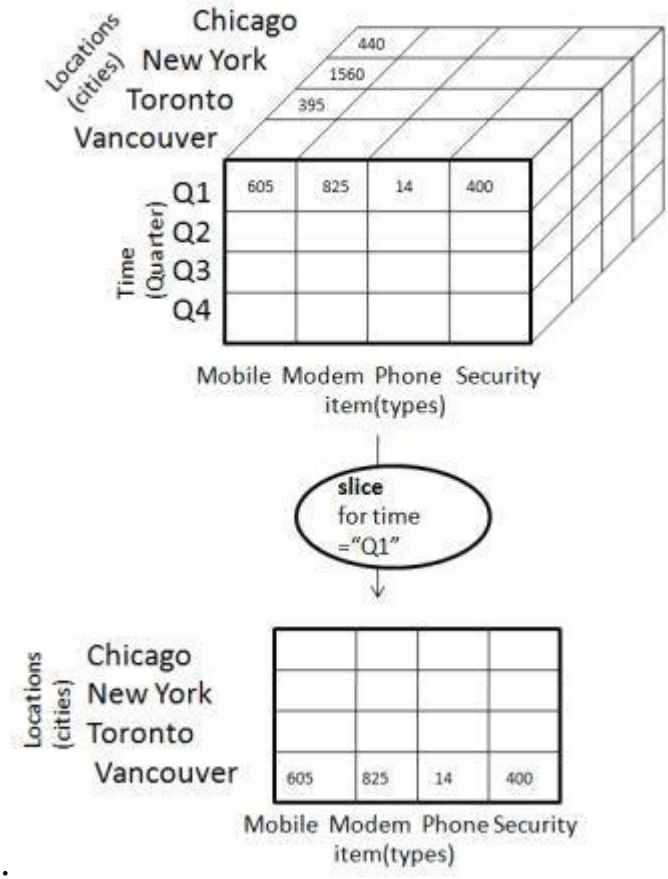
- By stepping down a concept hierarchy for a dimension
- By introducing a new dimension.

The following diagram illustrates how drill-down works −



Slice

The slice operation selects one particular dimension from a given cube and provides a new sub-cube. Consider the following diagram that shows how slice works.



-

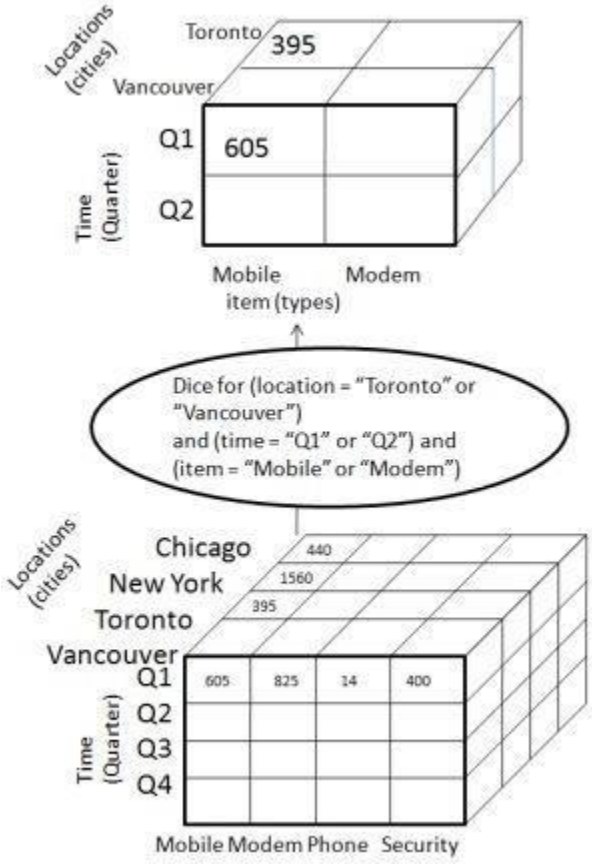
Here Slice is performed for the dimension "time" using the criterion time = "Q1".

-
-

It will form a new sub-cube by selecting one or more dimensions.

Dice

Dice selects two or more dimensions from a given cube and provides a new sub-cube. Consider the following diagram that shows the dice operation.

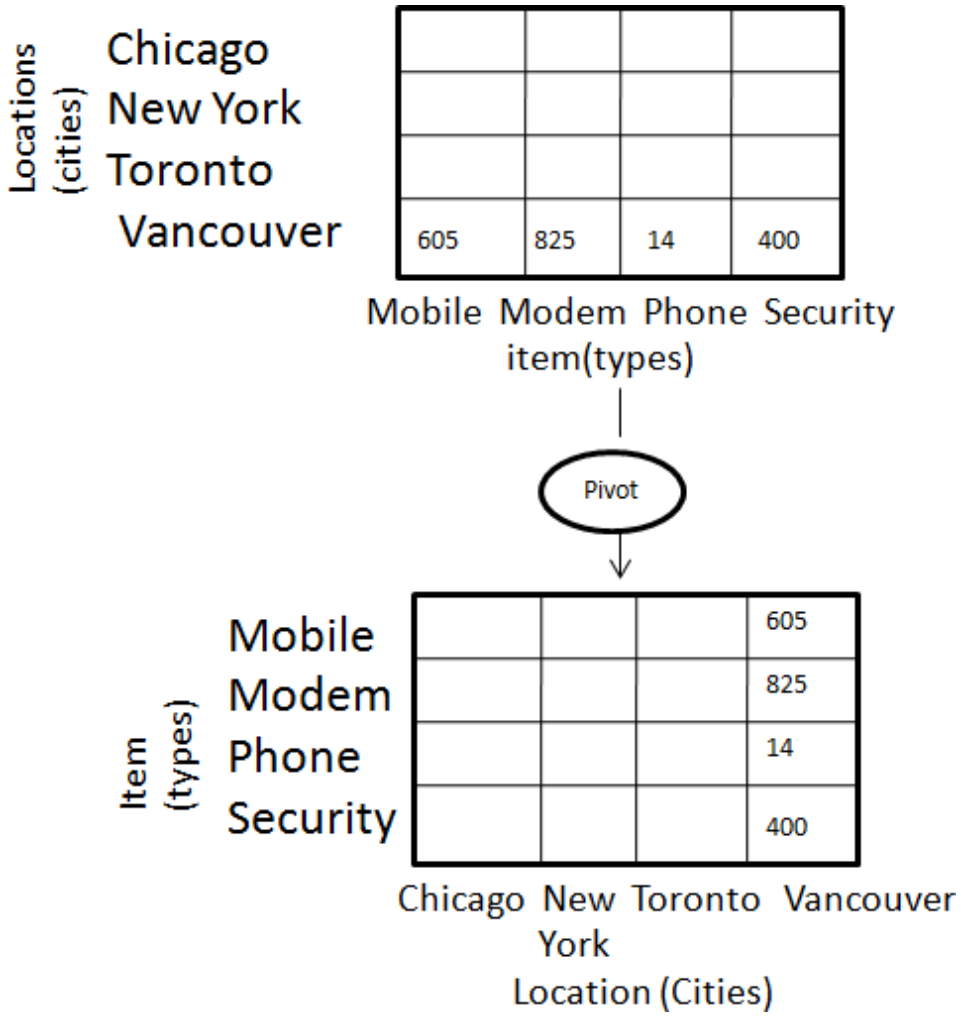


The dice operation on the cube based on the following selection criteria involves three dimensions.

- (location = "Toronto" or "Vancouver")
- (time = "Q1" or "Q2")
- (item =" Mobile" or "Modem")

Pivot

The pivot operation is also known as rotation. It rotates the data axes in view in order to provide an alternative presentation of data. Consider the following diagram that shows the pivot operation.



OLAP Vs OLTP

S. No	Data Warehouse (OLAP)	Operational Database (OLTP)
1	Involves historical processing of information.	Involves day-to-day processing.
2	OLAP systems are used by knowledge workers such as executives, managers and analysts.	OLTP systems are used by clerks, DBAs, or database professionals.
3	Useful in analyzing the business.	Useful in running the business.
4	It focuses on Information out.	It focuses on Data in.
5	Based on Star Schema, Snowflake, Schema and Fact Constellation Schema.	Based on Entity Relationship Model.
6	Contains historical data.	Contains current data.
7	Provides summarized and consolidated data.	Provides primitive and highly detailed data.
8	Provides summarized and multidimensional view of data.	Provides detailed and flat relational view of data.
9	Number or users is in hundreds.	Number of users is in thousands.
10	Number of records accessed is in millions.	Number of records accessed is in tens.