1.3 Data Warehouse Design Process:

1.3.1 A data warehouse can be built using a *top-down approach*, a *bottom-up approach*, or a *combination of both*.

- The top-down approach starts with the overall design and planning. It is useful in cases where the technology is mature and well known, and where the business problems that must be solved are clear and well understood.

- The bottom-up approach starts with experiments and prototypes. This is useful in the early stage of business modeling and technology development. It allows an organization to move forward at considerably less expense and to evaluate the benefits of the technology before making significant commitments.

- In the combined approach, an organization can exploit the planned and strategic nature of the top-down approach while retaining the rapid implementation and opportunistic application of the bottom-up approach.

The warehouse design process consists of the following steps:

- Choose a business process to model, for example, orders, invoices, shipments, inventory, account administration, sales, or the general ledger. If the business process is organizational and involves multiple complex object collections, a data warehouse model should be followed. However, if the process is departmental and focuses on the analysis of one kind of business process, a data mart model should be chosen.
- Choose the grain of the business process. The grain is the fundamental, atomic level of data to be represented in the fact table for this process, for example, individual transactions, individual daily snapshots, and so on.
- Choose the dimensions that will apply to each fact table record. Typical dimensions are time, item, customer, supplier, warehouse, transaction type, and status.
- Choose the measures that will populate each fact table record. Typical measures are numeric additive quantities like dollars sold and units sold.

1.3.2 Data Warehouse Usage for Information Processing

There are three kinds of dataware house applications: information processing, analytical processing, and data mining.

Information processing supports querying, basic statistical analysis, and reporting using crosstabs, tables, charts, or graphs. A current trend in data warehouse information processing is to construct low-cost web-based accessing tools that are then integrated with web browsers.

Analytical processing supports basic OLAP operations, including slice-and-dice, drill-down, roll-up, and pivoting. It generally operates on historic data in both summarized and detailed forms. The major strength of online analytical processing over information processing is the multidimensional data analysis of data warehouse data**.**

Datamining supports knowledge discovery by finding hidden patterns and associations, constructing analytical models, performing classification and prediction, and presenting the mining results using visualization tools.

1.3.3 From Online Analytical Processing to Multidimensional Data Mining

Multidimensional data mining is particularly important for the following reasons:

- **High quality of data in data warehouses**: Most data mining tools need to work on integrated, consistent, and cleaned data, which requires costly data cleaning, data integration, and data transformation as preprocessing steps. A data warehouse constructed by such preprocessing serves as a valuable source of high-quality data for OLAP as well as for data mining. Notice that data mining may serve as a valuable tool for data cleaning and data integration as well.

- **Available information processing infrastructure surrounding data warehouses:** Comprehensive information processing and data analysis infrastructures have been or will be systematically constructed surrounding data warehouses, which include accessing, integration, consolidation, and transformation of multiple heterogeneous databases, ODBC/OLEDB connections, Web accessing and service facilities, and reporting and OLAP analysis tools. It is prudent to make the best use of the available infrastructures rather than constructing everything from scratch.

- **OLAP-based exploration of multidimensional data:** Effective data mining needs exploratory data analysis. A user will often want to traverse through a database, select portions of relevant data, analyze them at different granularities, and present knowledge/results in different forms. Multidimensional data mining provides facilities for mining on different subsets of data and at varying levels of abstraction—by drilling, pivoting, filtering, dicing, and slicing on a data cube and/or intermediate data mining

results. This, together with data/knowledge visualization tools, greatly enhances the power and flexibility of data mining.

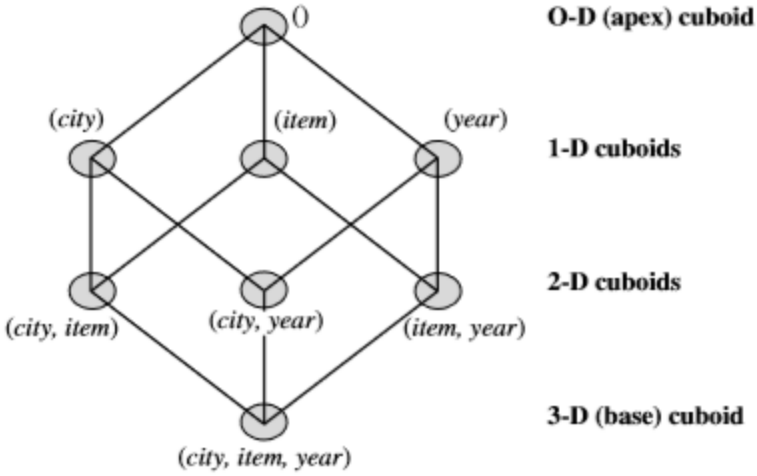
- **Online selection of data mining functions:** Users may not always know the specific kinds of knowledge they want to mine. By integrating OLAP with various data mining functions, multidimensional data mining provides users with the flexibility to select desired data mining functions and swap data mining tasks dynamically.

1.4 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 systems to support highly efficient cube computation techniques, access methods, and query processing techniques. In this section, we present an overview of methods for the efficient implementation of data warehouse systems.

1.4.1 Efficient Data Cube Computation: An Overview

At the core of multidimensional data analysis is the efficient computation of aggregations across many 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 cuboids defining a data cube.



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

The compute cube Operator and the Curse of Dimensionality

A data cube is a lattice of cuboids. Suppose that you want to create a data cube for AllElectronics sales that contains the following: city, item, year, and sales in dollars. You want 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."
- "Compute the sum of sales, grouping by item."

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 groupby's, that can be computed for this data cube is 23 =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 the figure above

An SQL query containing no group-by(e.g.,"compute the sum of total sales") is a zero dimensional operation. An SQL query containing one group-by (e.g., "compute the sum of sales, group-bycity") is a one-dimensional operation. A cube operator on n dimensions is equivalent to a collection of group-by statements, one for each subset of the n dimensions. Therefore, the cube operator is the n-dimensional generalization of the group-by operator. Similar to the SQL syntax, the data cube in Example 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 2n 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 eight subsets of the set {city, item, year}, including the empty subset.

Online analytical processing may need to access different cuboids for different queries. Therefore, it may seem like a good idea to compute in advance all or at least some of the cuboids in a data cube. Precomputation leads to fast response time and avoids some redundant computation

A major **challenge** related to this precomputation, however, is that the required storage space may explode if all 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**

- How many cuboids in an n-dimensional cube with L levels?

$$T = \prod_{i=1}^{n} (L_i + 1)$$

where Li is the number of levels associated with dimension i. One is added to Li to include the virtual top level, all. (Note that generalizing to all is equivalent to the removal of the dimension.)

If there are many cuboids, and these cuboids are large in size, a more reasonable option is partial materialization; that is, to materialize only some of the possible cuboids that can be generated.

- Materialize every (cuboid) (full materialization), none (no materialization), or some (partial materialization)
- Selection of which cuboids to materialize -Based on size, sharing, access frequency, etc

Partial Materialization: Selected Computation of Cuboids

There are three choices for data cube materialization given a base cuboid:

1. No materialization: Do not precompute any of the "nonbase" cuboids. This leads to computing expensive multidimensional aggregates on-the-fly, which can be extremely slow.

2. Full materialization: Precompute all of the cuboids. The resulting lattice of computed cuboids is referred to as the full cube. This choice typically requires huge amounts of memory space in order to store all of the precomputed cuboids.

3. Partial materialization: Selectively compute a proper subset of the whole set of possible cuboids. Alternatively, we may compute a subset of the cube, which contains only those cells that satisfy some user-specified criterion, such as where the tuple count of each cell is above some threshold. We will use the term sub cube to refer to the lattercase,where only some of the cells may be precomputed for various cuboids. Partial materialization represents an interesting trade-off between storage space and response time.

The partial materialization of cuboids or subcubes should consider three factors:
(1) identify the subset of cuboids or subcubes to materialize;
(2) exploit the materialized cuboids or subcubes during query processing; and
(3)efficiently update the materialized cuboids or subcubes during load and refresh.

1.4.2 Indexing OLAP Data: Bitmap Index and Join Index

The bitmap indexing method is popular in OLAP products because it allows quick searching in data cubes. The bitmap index is an alternative representation of the record ID (RID) list. In the bitmap index for a given attribute, there is a distinct bit vector, Bv, for each value v in the attribute's domain. If a given attribute's domain consists of n values, then n bits are needed for each entry in the bitmap index (i.e., there are n bit vectors). If the attribute has the value v for a given row in the data table, then the bit representing that value is set to 1 in the corresponding row of the bitmap index. All other bits for that row are set to 0.

- **Index on a particular column**

- **Each value in the column has a bit vector: bit-op is fast**
- **The length of the bit vector: # of records in the base table**
- **The *i*-th bit is set if the *i*-th row of the base table has the value for the indexed column**
- **not suitable for high cardinality domains**

Bitmap indexing. In the *AllElectronics* data warehouse, suppose the dimension *item* at the top level has four values (representing item types): *"home entertainment," "computer," "phone,"* and *"security."* Each value (e.g., *"computer"*) is represented by a bit vector in the *item* bitmap index table. Suppose that the cube is stored as a relation table with 100,000 rows. Because the domain of *item* consists of four values, the bitmap index table requires four bit vectors (or lists), each with 100,000 bits. Figure 4.15 shows a base (data) table containing the dimensions *item* and *city*, and its mapping to bitmap index tables for each of the dimensions. ■

Base table

RID	*item*	*city*
R1	H	V
R2	C	V
R3	P	V
R4	S	V
R5	H	T
R6	C	T
R7	P	T
R8	S	T

item bitmap index table

RID	H	C	P	S
R1	1	0	0	0
R2	0	1	0	0
R3	0	0	1	0
R4	0	0	0	1
R5	1	0	0	0
R6	0	1	0	0
R7	0	0	1	0
R8	0	0	0	1

city bitmap index table

RID	V	T
R1	1	0
R2	1	0
R3	1	0
R4	1	0
R5	0	1
R6	0	1
R7	0	1
R8	0	1

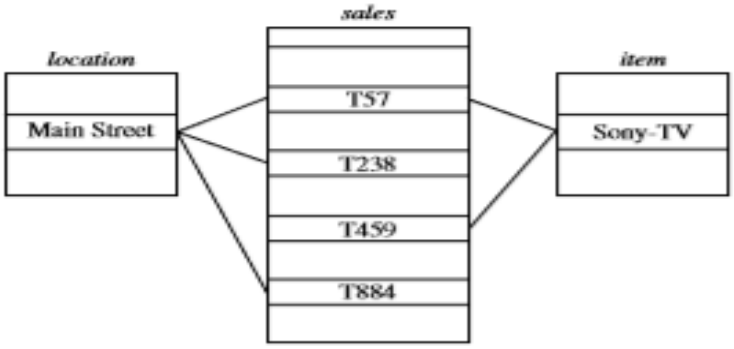
Note: H for "home entertainment," C for "computer," P for "phone," S for "security," V for "Vancouver," T for "Toronto."

Indexing OLAP data using bitmap indices.

The join indexing method gained popularity from its use in relational database query processing. Traditional indexing maps the value in a given column to a list of rows having that value. In contrast, join indexing registers the joinable rows of two relations from a relational database.

- In data warehouses, join index relates the values of the dimensions of a start schema to rows in the fact table.
  - E.g. fact table: *Sales* and two dimensions *city* and *product*
    - A join index on *city* maintains for each distinct city a list of R-IDs of the tuples recording the Sales in the city
  - Join indices can span multiple dimensions

Join indexing. In Example 3.4, we defined a star schema for *AllElectronics* of the form "*sales_star* [*time, item, branch, location*]: *dollars_sold* = sum (*sales_in_dollars*)." An example of a join index relationship between the *sales* fact table and the *location* and *item* dimension tables is shown in Figure 4.16. For example, the "Main Street" value in the *location* dimension table joins with tuples T57, T238, and T884 of the *sales* fact table. Similarly, the "Sony-TV" value in the *item* dimension table joins with tuples T57 and T459 of the *sales* fact table. The corresponding join index tables are shown in Figure 4.17.



Linkages between a *sales* fact table and *location* and *item* dimension tables.

Join index table for *location/sales*

location	*sales_key*
Main Street	T57
Main Street	T238
Main Street	T884

Join index table for *item/sales*

item	*sales_key*
Sony-TV	T57
Sony-TV	T459

Join index table linking *location* and *item* to *sales*

location	*item*	*sales_key*
Main Street	Sony-TV	T57

Join index tables based on the linkages between the *sales* fact table and the *location* and *item* dimension tables shown in Figure 4.16.

1.4.3 Efficient Processing of OLAP Queries

Given materialized views, query processing should proceed as follows:

- Determine which operations should be performed on the available cuboids
  - Transform drill, roll, etc. into corresponding SQL and/or OLAP operations, e.g., dice = selection + projection
- Determine which materialized cuboid(s) should be selected for OLAP op.
  - Let the query to be processed be on {brand, province_or_state} with the condition "year = 2004", and there are 4 materialized cuboids available:

1) {year, item_name, city}
2) {year, brand, country}
3) {year, brand, province_or_state}
4) {item_name, province_or_state} where year = 2004

Which should be selected to process the query?

- Explore indexing structures and compressed vs. dense array structs in MOLAP

1.4.4 OLAP Server Architectures: ROLAP versus MOLAP versus HOLAP

1. *Relational OLAP (ROLAP):*

- ROLAP works directly with relational databases. The base data and the dimension tables are stored as relational tables and new tables are created to hold the aggregated information. It depends on a specialized schema design.
- This methodology relies on manipulating the data stored in the relational database to give the appearance of traditional OLAP's slicing and dicing functionality. In essence, each action of slicing and dicing is equivalent to adding a "WHERE" clause in the SQL statement.
- ROLAP tools do not use pre-calculated data cubes but instead pose the query to the standard relational database and its tables in order to bring back the data required to answer the question.
- ROLAP tools feature the ability to ask any question because the methodology does not limit to the contents of a cube. ROLAP also has the ability to drill down to the lowest level of detail in the database.

2. *Multidimensional OLAP (MOLAP):*

- MOLAP is the 'classic' form of OLAP and is sometimes referred to as just OLAP.
- MOLAP stores this data in an optimized multi-dimensional array storage, rather than in a relational database. Therefore it requires the pre-computation and storage of information in the cube - the operation known as processing.
- MOLAP tools generally utilize a pre-calculated data set referred to as a data cube.

  The data cube contains all the possible answers to a given range of questions.
- MOLAP tools have a very fast response time and the ability to quickly write back data into the data set.

3. *Hybrid OLAP (HOLAP):*

- There is no clear agreement across the industry as to what constitutes Hybrid OLAP, except that a database will divide data between relational and specialized storage.
- For example, for some vendors, a HOLAP database will use relational tables to hold the larger quantities of detailed data, and use specialized storage for at least some aspects of the smaller quantities of more-aggregate or less-detailed data.
- HOLAP addresses the shortcomings of MOLAP and ROLAP by combining the capabilities of both approaches.

- HOLAP tools can utilize both pre-calculated cubes and relational data sources.

Property	MOLAP	ROLAP
Data structure	Multidimensional database using sparse arrays	Relational tables (each cell is a row)
Disk space	Separate database for data cube; large for large data cubes	May not require any space other than that available in the data warehouse
Retrieval	Fast(pre-computed)	Slow(computes on-the-fly)
Scalability	Limited (cubes can be very large)	Excellent
Best suited for	Inexperienced users, limited set of queries	Experienced users, queries change frequently
DBMS facilities	Usually weak	Usually very strong

Question Bank

1What is data warehouse? Discuss key features
2. Differentiate between Operational Database Systems and Data Warehouses.
3. Differentiate between OLAP and OLTP
4. Why multidimensional views of data and data-cubes are used? With a neat diagram, explain data-cube implementations.
5. Describe the Multitiered Architecture of data warehousing.
6. Explain the data warehouse models
7. What is ETL? Explain the steps in ETL
8. Write a note on Metadata Repository
9. Explain in detail data warehouse design and implementation. Explain star schema and snow flake schema in detail?
10. Explain the schemas for multidimensional data models
11. Discuss the role of concept hierarchies
12. Explain OLAP operations with an example.
13. Describe the steps in data warehouse design process.
14. Explain Bitmap indexing and join indexing
15. Describe the difference between ROLAP & MOLAP

Module-2

Why Mine Data?

Commercial Viewpoint:

Lots of data is being collected and warehoused

- Web data, e-commerce
- purchases at department/grocery stores
- Bank/Credit Card transactions

Data mining techniques can be used to support a wide range of business intelligence applications such as customer profiling, targeted marketing, workflow management, store layout, and fraud detection.

It can also help retailers answer important business questions such as

- "Who are the most profitable customers?"
- "What products can be cross-sold or up-sold?"
- "What is the revenue outlook of the company for next year?

Scientific Viewpoint

Data is collected and stored at enormous speeds (GB/hour). E.g.

- remote sensors on a satellite
- telescopes scanning the skies
- scientific simulations

As an important step toward improving our understanding of the Earth's climate system, NASA has deployed a series of Earth orbiting satellites that continuously generate global observations of the Land surface, oceans, and atmosphere.

Techniques developed in data mining can aid Earth scientists in answering questions such as

- "What is the relationship between the frequency and intensity of ecosystem disturbances such as droughts and hurricanes to global warming?"
- "How is land surface precipitation and temperature affected by ocean surface temperature?"
- "How well can we predict the beginning and end of the growing season for a region?"

What Is Data Mining?(Definition)

Data mining is the process of automatically discovering useful information in large data repositories.

- Finding hidden information in a database
- Data mining techniques are deployed to scour large databases in order to find novel and useful patterns that might otherwise remain unknown. They also provide capabilities to predict e outcome of a future observation.

Applications of data mining

- Banking: loan/credit card approval
  - Given a database of 100,000 names, which persons are the least likely to default on their credit cards?
- Customer relationship management:
  - Which of my customers are likely to be the most loyal, and which are most likely to leave for a competitor?
- Targeted marketing:
  - identify likely responders to promotions
- Fraud detection: telecommunications, financial transactions
  - from an online stream of event identify fraudulent events
- Manufacturing and production:
  - automatically adjust knobs when process parameter changes
- Medicine: disease outcome, effectiveness of treatments
  - analyze patient disease history: find relationship between diseases
- Molecular/Pharmaceutical: identify new drugs
- Scientific data analysis:
  - identify new galaxies by searching for sub clusters
- Web site/store design and promotion:
  - find affinity of visitor to pages and modify layout

What is not Data Mining?

- Find all credit applicants with last name of Smith.
- Identify customers who have purchased more than $10,000 in the last month.
- Find all customers who have purchased milk
- Iooking up individual records using a database managemenr sysrem Look up phone number in phone directory.
- finding particular Web pages via a query to an Internet search engine.

What is Data Mining?

- Find all credit applicants who are poor credit risks. (classification)
- Identify customers with similar buying habits. (Clustering)
- Find all items which are frequently purchased with milk. (association rules)
- Discover groups of similar documents on the Web
- Certain names are more popular in certain locations

Data Mining and Knowledge Discovery

Data mining is an integral part of knowledge discovery in databases (KDD), which is the overall process of converting raw data into useful information,

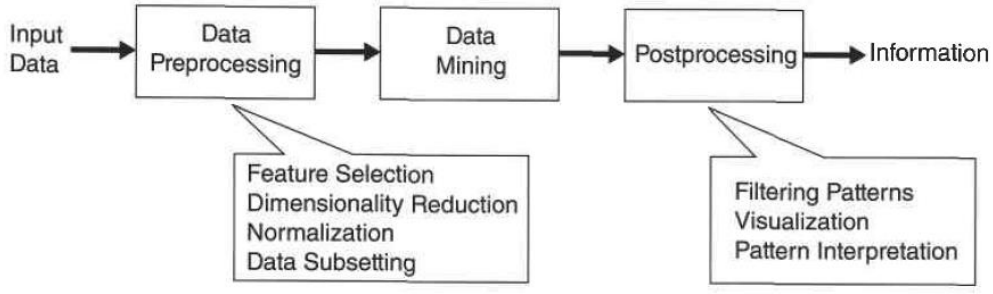


Figure 1.1. The process of knowledge discovery in databases (KDD).

Preprocessing

- The input data can be stored in a variety of formats (flat files, spreadsheets, or relational tables) and may reside in a centralized data repository or be distributed across multiple sites.

- The purpose of preprocessing is to transform the raw input data into an appropriate format for subsequent analysis.
- Data preprocessing include fusing data from multiple sources, cleaning data to remove noise and duplicate observations, and selecting records and features that are relevant to the data mining task at hand.

Post processing:

- Ensures that only valid and useful results are incorporated into the system.
- Which allows analysts to explore the data and the data mining results from a variety of viewpoints.
- Testing methods can also be applied during post processing to eliminate spurious data mining results.

Motivating Challenges

The following are some of the specific challenges that motivated the development of data mining.

Scalability :

Because of advances in data generation and collection, data sets with sizes of gigabytes, terabytes, or even petabytes are becoming common.

If data mining algorithms are to handle these massive data sets, then they must be scalable.

Scalability may also require the implementation of novel data structures to access individual records in an efficient manner.

High Dimensionality:

It is now common to encounter data sets with hundreds or thousands of attributes.

Traditional data analysis techniques that were developed for low-dimensional data often do not work well for such high dimensional data.

Heterogeneous and Complex Data:

Recent years have also seen the emergence of more complex data objects, such as

- Collections of Web pages containing semi-structured text and hyperlinks;
- DNA data with sequential and three-dimensional structure;
- climate data that consists of time series measurements (temperature, pressure, etc.) various locations on the Earth's surface .

Techniques developed for mining such complex objects should take into consideration relationships in the data, such as temporal and spatial autocorrelation, graph connectivity, and parent-child relationships between the elements in semi-structured text.

Data ownership and Distribution:

Sometimes, the data needed for an analysis is not stored in one location or owned by one organization. Instead, the data is geographically distributed among resources belonging to multiple entities. This requires the development of distributed data mining techniques.

Among the key challenges faced by distributed data mining algorithms include

(1) How to reduce the amount of communication needed to perform the distributed computation.
(2)How to effectively consolidate the data mining results obtained from multiple sources.

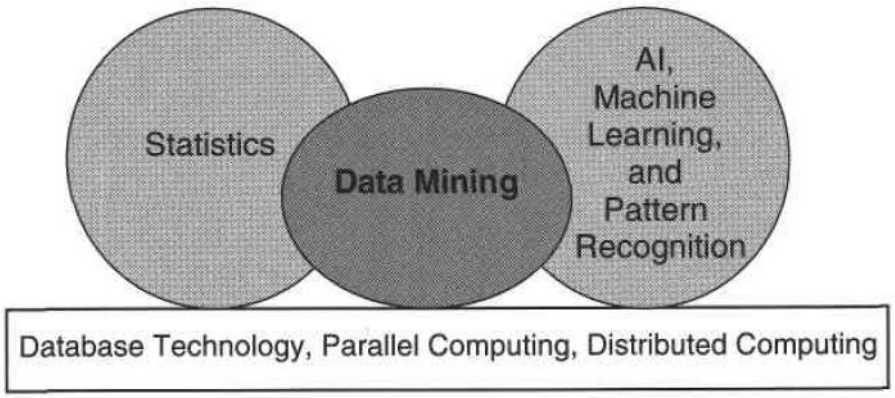
(3) How to address data security issues.

Origins of Data Mining

Draws ideas from machine learning/AI, pattern recognition, statistics, and database systems

Traditional Techniques may be unsuitable due to Enormity of data,High dimensionality of data, Heterogeneous, distributed nature of data.

Figure 1.2 shows the relationship of data mining to other areas.



Data Mining Tasks

Two broad categories

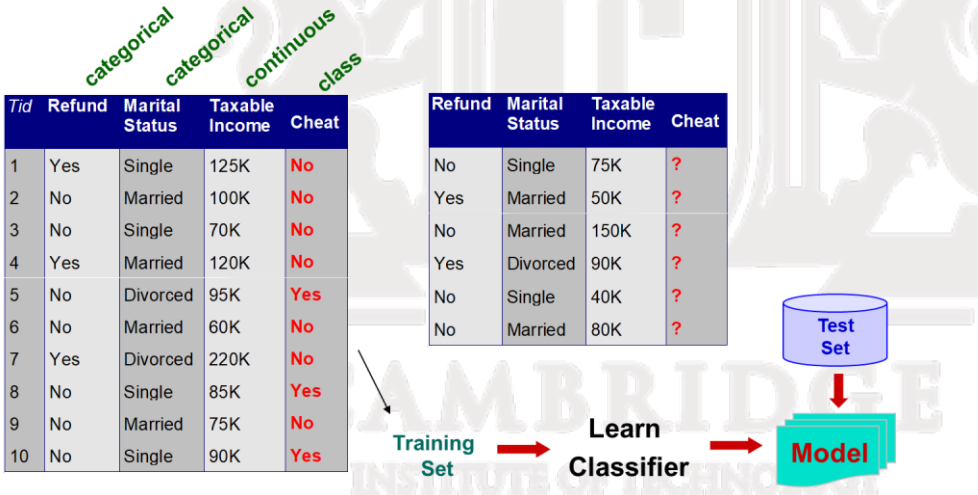
1) Prediction Methods*:* Use some variables to predict unknown or future values of other variables.

2) Description Methods *:*Find human-interpretable patterns that describe the data

Classification [Predictive]
Clustering [Descriptive]
Association Rule Discovery [Descriptive]
Sequential Pattern Discovery [Descriptive]
Regression [Predictive]
Deviation/Anamoly Detection [Predictive]

Classification: Definition

- ✓ Given a collection of records (training set)
- ✓ Each record contains a set of attributes, one of the attributes is the class.
- ✓ Find a model for class attribute as a function of the values of other attributes.
- ✓ **Goal**: previously unseen records should be assigned a class as accurately as possible.
- ✓ A test set is used to determine the accuracy of the model. Usually, the given data set is divided into training and test sets, with training set used to build the model and test set used to validate it.



Classification: Application 1

Direct Marketing

Goal: Reduce cost of mailing by targeting a set of consumers likely to buy a new cell-phone product.

Approach:

- ✓ Use the data for a similar product introduced before.
- ✓ We know which customers decided to buy and which decided otherwise. This **{buy, don't buy}** decision forms the class attribute.

- ✓ Collect various demographic, lifestyle, and company-interaction related information about all such customers, such as type of business, where they stay, how much they earn, etc.
- ✓ Use this information as input attributes to learn a classifier model.

Classification: Application 2

Fraud Detection

Goal: Predict fraudulent cases in credit card transactions.

Approach:

- ✓ Use credit card transactions and the information on its account-holder as attributes.
- ✓ When does a customer buy, what does he buy, how often he pays on time, etc
- ✓ Label past transactions as fraud or fair transactions. This forms the class attribute.
- ✓ Learn a model for the class of the transactions.
- ✓ Use this model to detect fraud by observing credit card transactions on an account

Classification: Application 3

Customer Attrition/Churn:

Goal: To predict whether a customer is likely to be lost to a competitor.

Approach:

- ✓ Use detailed record of transactions with each of the past and present customers, to find attributes.
- ✓ How often the customer calls, where he calls, what time-of-the day he calls most, his financial status, marital status, etc.
- ✓ Label the customers as loyal or disloyal.
- ✓ Find a model for loyalty.

Clustering: Definition

Given a set of data points, each having a set of attributes, and a similarity measure among them, find clusters such that

- ✓ Data points in one cluster are more similar to one another.
- ✓ Data points in separate clusters are less similar to one another.
- ✓ Similarity Measures:
  - Euclidean Distance if attributes are continuous.

- Other Problem-specific Measures.

Clustering: Application 1

Market Segmentation:

Goal: Subdivide a market into distinct subsets of customers where any subset may conceivably be selected as a market target to be reached with a distinct marketing mix.

Approach:

- ✓ Collect different attributes of customers based on their geographical and lifestyle related information.
- ✓ Find clusters of similar customers.
- ✓ Measure the clustering quality by observing buying patterns of customers in same cluster vs. those from different clusters.

Clustering: Application 2

Document Clustering

Goal: To find groups of documents that are similar to each other based on the important terms appearing in them.

Approach:

- ✓ To identify frequently occurring terms in each document. Form a similarity measure based on the frequencies of different terms. Use it to cluster.

Gain: Information Retrieval can utilize the clusters to relate a new document or search term to clustered documents

Association Rule Discovery: Definition

Given a set of records each of which contain some number of items from a given collection;

Produce dependency rules which will predict occurrence of an item based on occurrences of other items

TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

Rules Discovered:

{Milk} --> {Coke}

{Diaper, Milk} --> {Beer}

Association Rule Discovery: Application 1

***Marketing and Sales Promotion*:**

Let the rule discovered be

{Bagels, … } --> {Potato Chips}

(antecedent) (consequent)

- ✓ Potato Chips as consequent => Can be used to determine what should be done to boost its sales.
- ✓ Bagels in the antecedent => Can be used to see which products would be affected if the store discontinues selling bagels.

Bagels in antecedent and Potato chips in consequent

=> Can be used to see what products should be sold with Bagels to promote sale of Potato chips!

Association Rule Discovery: Application 2

Supermarket shelf management

Goal: To identify items that are bought together by sufficiently many customers.

Approach:

- ✓ Process the point-of-sale data collected with barcode scanners to find dependencies among items.
- ✓ A classic rule --
  - If a customer buys diaper and milk, then he is very likely to buy beer.
  - So, don't be surprised if you find six-packs stacked next to diapers!

Regression: Definition

Predict a value of a given continuous valued variable based on the values of other variables, assuming a linear or nonlinear model of dependency.

Greatly studied in statistics and neural network fields.

Examples:

- ✓ Predicting sales amounts of new product based on advertising expenditure.
- ✓ Predicting wind velocities as a function of temperature, humidity, air pressure, etc.
- ✓ Time series prediction of stock market indices.

Deviation/Anomaly Detection:Definition

Detect significant deviations from normal behavior

Applications: Credit Card Fraud Detection

Network Intrusion Detection

Problems: Discuss whether or not each of the following activities is a data mining task.

(a) Dividing the customers of a company according to their gender.

No. This is a simple database query.

(b) Dividing the customers of a company according to their profitability.

No. This is an accounting calculation, followed by the application of a threshold. However, predicting the profitability of a new customer would be data mining.

(c) Computing the total sales of a company.

No. Again, this is simple accounting.

(d) Sorting a student database based on student identification numbers.

No. Again, this is a simple database query.

(e) Predicting the outcomes of tossing a (fair) pair of dice.

No. Since the die is fair, this is a probability calculation. If the die were not fair, and we needed to estimate the probabilities of each outcome from the data, then this is more like the problems considered by data mining. However, in this specific case, solutions to this problem were developed by mathematicians a long time ago, and thus, we wouldn't consider it to be data mining.

(f) Predicting the future stock price of a company using historical records.

Yes. We would attempt to create a model that can predict the continuous value of the stock price. This is an example of the area of data mining known as predictive modeling. We could use regression for this modeling, although researchers in many fields have developed a wide variety of techniques for predicting time series.

(g) Monitoring the heart rate of a patient for abnormalities.

Yes. We would build a model of the normal behavior of heart rate and raise an alarm when an unusual heart behavior occurred. This would involve the area of data mining known as anomaly detection. This could also be considered as a classification problem if we had examples of both normal and abnormal heart behavior.

(h) Monitoring seismic waves for earthquake activities.

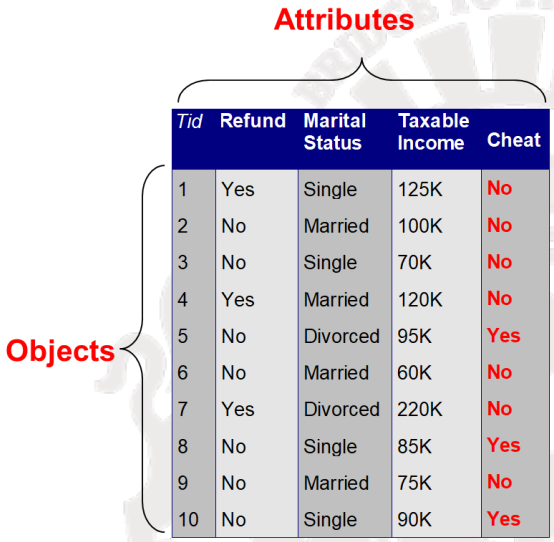
Yes. In this case, we would build a model of different types of seismic wave behavior associated with earthquake activities and raise an alarm when one of these different types of seismic activity was observed. This is an example of the area of data mining known as classification.

(i) Extracting the frequencies of a sound wave.

No. This is signal processing.

What is Data?

- ✓ Collection of data objects and their attributes
- ✓ An attribute is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.
  - Attribute is also known as variable, field, characteristic, or feature
- ✓ A collection of attributes describe an object
  - Object is also known as record, point, case, sample, entity, or instance



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Attribute Values

- ✓ Attribute values are numbers or symbols assigned to an attribute
- ✓ Distinction between attributes and attribute values
  - Example: height can be measured in feet or meters
- ✓ Different attributes can be mapped to the same set of values
  - Example: Attribute values for ID and age are integers
- ✓ But properties of attribute values can be different
  - ID has no limit but age has a maximum and minimum value

Types of Attributes

There are different types of attributes

- ✓ Nominal
  - Examples: ID numbers, eye color, zip codes

- ✓ Ordinal
  - Examples: rankings (e.g., taste of potato chips on a scale from 1-10), grades, height in {tall, medium, short}
- ✓ Interval
  - Examples: calendar dates, temperatures in Celsius or Fahrenheit.
- ✓ Ratio
  - Examples: temperature in Kelvin, length, time, counts

Properties of Attribute Values

The type of an attribute depends on which of the following properties it possesses:

- ✓ Distinctness: = ≠
- ✓ Order: < >
- ✓ Addition: + -
- ✓ Multiplication: * /

- ✓ Nominal attribute: distinctness
- ✓ Ordinal attribute: distinctness & order
- ✓ Interval attribute: distinctness, order & addition
- ✓ Ratio attribute: all 4 properties

Attribute Type	Description	Examples	Operations
Nominal	The values of a nominal attribute are just different names, i.e., nominal attributes provide only enough information to distinguish one object from another. (=, ≠)	zip codes, employee ID numbers, eye color, sex: {*male, female*}	mode, entropy, contingency correlation, χ^2 test
Ordinal	The values of an ordinal attribute provide enough information to order objects. (<, >)	hardness of minerals, {*good, better, best*}, grades, street numbers	median, percentiles, rank correlation, run tests, sign tests
Interval	For interval attributes, the differences between values are meaningful, i.e., a unit of measurement exists. (+, -)	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, Pearson's correlation, *t* and *F* tests
Ratio	For ratio variables, both differences and ratios are meaningful. (*, /)	temperature in Kelvin, monetary quantities, counts, age, mass, length, electrical current	geometric mean, harmonic mean, percent variation

Describing Attribute by the number of values

An independent way of distinguishing between attributes is by the number of values they can take

Discrete Attribute

Has only a finite or countable infinite set of values

- ✓ Examples: zip codes, counts, or the set of words in a collection of documents .
- ✓ Often represented as integer variables.
- ✓ Note: binary attributes are a special case of discrete attributes

Continuous Attribute

Has real numbers as attribute values

- ✓ Examples: temperature, height, or weight.
- ✓ Practically, real values can only be measured and represented using a finite number of digits.
- ✓ Continuous attributes are typically represented as floating-point variables.

Types of data sets

Record

- ✓ Collection of records , each of which consists of a fixed set of data fields (attributes).
- ✓ For the most basic form of record data, there is no explicit relationship among records every record (object) has the same set of attributes.
- ✓ Record data is usually stored either in flat files or in relational databases
- ✓ Example:
    - o Data Matrix
    - o Document Data
    - o Transaction Data

Graph-Based Data

- ✓ frequently convey important information. In such cases, the data is often represented as a graph.
- ✓ In particular, the data objects are mapped to nodes of the graph, while the relationships among objects are captured by the links between objects and link properties, such as direction and weight.

Example: World Wide Web, Molecular Structures etc.,

Ordered Data

For some types of data, the attributes have relationships that involve order in time or space Different types of ordered data are

Sequential Data: Also referred to as temporal data, can be thought of as an extension of record data, where each record has a time associated with it.

Example:A retail transaction data set that also stores the time at which the transaction took place

Sequence Data : Sequence data consists of a data set that is a sequence of individual entities, such as a sequence of words or letters. It is quite similar to sequential data, except that there are no time stamps; instead, there are positions in an ordered sequence.

Example: the genetic information of plants and animals can be represented in the form of sequences of nucleotides that are known as genes.

Time Series Data : Time series data is a special type of sequential data in which each record is a time series, i.e., a series of measurements taken over time.

Example: A financial data set might contain objects that are time series of the daily prices of various stocks.

Example,: consider a time series of the average monthly temperature for a city during the years 1982 to 1994

Spatial Data : Some objects have spatial attributes, such as positions or areas, as well as other types of attributes.

Example: Weather data (precipitation, temperature, pressure) that is collected for a variety of geographical locations.

General Characteristics of Data Sets

Dimensionality :

- ✓ It is the number of attributes that the objects in the data set possess.
- ✓ Data with a small number of dimensions tends to be qualitatively different than moderate or high-dimensional data.

Sparsity:

- ✓ For some data sets, most attributes of an object have values of 0; in many cases fewer than 1% of the entries are non-zero.
- ✓ In practical terms, sparsity is an advantage because usually only the non-zero values need to be stored and manipulated.
- ✓ This results in significant savings with respect to computation time and storage.

Resolution:

- ✓ It is frequently possible to obtain data at different levels of resolution, and often the properties of the data are different at different resolutions.
- ✓ For instance, the surface of the Earth seems very uneven at a resolution of meters, but is relatively smooth at a resolution of tens of kilometers.
- ✓ The patterns in the data also depend on the level of resolution.
- ✓ If the resolution is too fine, a pattern may not be visible or may be buried in noise; if the resolution is too coarse, the pattern may disappear.

Data Quality:

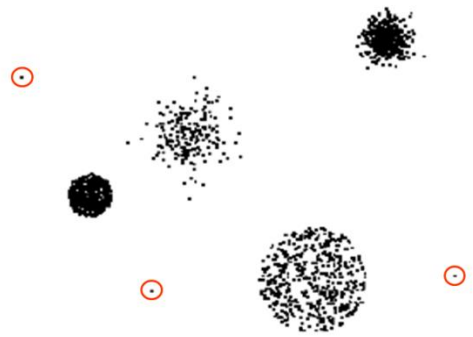
- ✓ Data mining applications are often applied to data that was collected for another purpose, or for future, but unspecified applications.
- ✓ For that reason ,data mining cannot usually take advantage of the significant benefits of "addressing quality issues at the source."
- ✓ Data mining focuses on (1) the detection and correction of data quality problems (called data cleaning.)
- ✓ (2) the use of algorithms that can tolerate poor data quality.
- ✓ Examples of data quality problems:

- Noise and outliers
- missing values
- duplicate data

Noise: Noise refers to modification of original values

Examples: distortion of a person's voice when talking on a poor phone and "snow" on television screen

Outliers :Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set



Missing Values:

- ✓ Reasons for missing values
  - Information is not collected (e.g., people decline to give their age and weight)
  - Attributes may not be applicable to all cases (e.g., annual income is not applicable to children)

- ✓ Handling missing values
  - Eliminate Data Objects
  - Estimate Missing Values
  - Ignore the Missing Value During Analysis
  - Replace with all possible values (weighted by their probabilities)

Duplicate Data:

- ✓ Data set may include data objects that are duplicates, or almost duplicates of one another
- ✓ Major issue when merging data from heterogeneous sources
- ✓ Examples:

  Same person with multiple email addresses
- ✓ Data cleaning

  Process of dealing with duplicate data issues

Precision, Bias, and Accuracy:

(Precision). The closeness of repeated measurements(of the same quantity) to one another

(Bias). A systematic quantity being measured

(Accuracy). The closeness of measurements o the true value of the quantity being measured

Data Preprocessing

- ✓ Preprocessing steps should be applied to make the data more suitable for data mining
- ✓ The most important ideas and approaches are
  - Aggregation
  - Sampling
  - Dimensionality Reduction
  - Feature subset selection
  - Feature creation
  - Discretization and Binarization
  - Attribute Transformation

Aggregation

- ✓ Combining two or more attributes (or objects) into a single attribute (or object)

Purpose:

- o Data reduction
- o Reduce the number of attributes or objects
- o Change of scale
  - Cities aggregated into regions, states, countries, etc
- o More “stable” data

- Aggregated data tends to have less variability

Sampling

- ✓ Sampling is the main technique employed for data selection.
- ✓ It is often used for both the preliminary investigation of the data and the final data analysis.

- ✓ Statisticians sample because obtaining the entire set of data of interest is too expensive or time consuming.
- ✓ Sampling is used in data mining because processing the entire set of data of interest is too expensive or time consuming.
- ✓ The key principle for effective sampling is the following:
    - using a sample will work almost as well as using the entire data sets, if the sample is representative
    - A sample is representative if it has approximately the same property (of interest) as the original set of data

Types of Sampling

- ✓ Simple Random Sampling

  There is an equal probability of selecting any particular item
- ✓ Sampling without replacement

  As each item is selected, it is removed from the population
- ✓ Sampling with replacement

  Objects are not removed from the population as they are selected for the sample.

  In sampling with replacement, the same object can be picked up more than once
- ✓ Stratified sampling

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

Dimensionality Reduction:

Purpose:

- Avoid curse of dimensionality
- Reduce amount of time and memory required by data mining algorithms

- Allow data to be more easily visualized
- May help to eliminate irrelevant features or reduce noise

✓ **The Curse of Dimensionality**: the curse of dimensionality refers to the phenomenon that many types of data analysis become significantly harder as the dimensionality of the data increases. Specifically, as dimensionality increases, the data becomes increasingly sparse in the space that it occupies.

Feature Subset Selection:

✓ Another way to reduce dimensionality of data

✓ Redundant features

- duplicate much or all of the information contained in one or more other attributes
- Example: purchase price of a product and the amount of sales tax paid

✓ Irrelevant features

- contain no information that is useful for the data mining task at hand
- Example: students' ID is often irrelevant to the task of predicting students' GPA

✓ Techniques of feature subset selection:

Brute-force approach:

Try all possible feature subsets as input to data mining algorithm

Embedded approaches:

Feature selection occurs naturally as part of the data mining algorithm

Filter approaches:

Features are selected before data mining algorithm is run

Wrapper approaches:

Use the data mining algorithm as a black box to find best subset of attributes

Feature Creation

- ✓ Create new attributes that can capture the important information in a data set much more efficiently than the original attributes
- ✓ Three general methodologies:
    - Feature Extraction
    - domain-specific
    - Mapping Data to New Space
    - Feature Construction
    - combining features

Discretization and Binarization:

- ✓ Some data mining algorithms require that the data be in the form of categorical attributes. Algorithms that find association patterns require that the data be in the form of binary attributes.
- ✓ Thus, it is often necessary to transform a continuous attribute into a categorical attribute (discretization).
- ✓ Both continuous and discrete attributes may need to be transformed into one or more binary attributes (binarization).

Table 2.5. Conversion of a categorical attribute to three binary attributes.

Categorical Value	Integer Value	x_1	x_2	x_3
awful	0	0	0	0
poor	1	0	0	1
OK	2	0	1	0
good	3	0	1	1
great	4	1	0	0

Attribute Transformation

- ✓ A function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values

Similarity and Dissimilarity: Definition

- ✓ **Similarity** between two objects is a numerical measure of how alike two data objects are.

- ✓ Similarities are higher for pairs of objects that are more alike.
- ✓ Similarities are usually non-negative and are often between 0 (no similarity) and 1 (complete similarity).

- ✓ **Dissimilarity** between two objects is a Numerical measure of how different are two data objects.
- ✓ Dissimilarities are lower for more similar pairs of objects.
- ✓ Minimum dissimilarity is often 0, Upper limit varies

Similarity/Dissimilarity for Simple Attributes

p and q are the attribute values for two data objects

Attribute Type	Dissimilarity	Similarity
Nominal	$d = \begin{cases} 0 & \text{if } p = q \\ 1 & \text{if } p \neq q \end{cases}$	$s = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$
Ordinal	$d = \frac{\lvert p-q \rvert}{n-1}$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - \frac{\lvert p-q \rvert}{n-1}$
Interval or Ratio	$d = \lvert p - q \rvert$	$s = -d$, $s = \frac{1}{1+d}$ or $s = 1 - \frac{d-min\_d}{max\_d-min\_d}$

Table 5.1. Similarity and dissimilarity for simple attributes

Dissimilarities between Data Objects

Euclidean Distance

The Euclidean distance, d, between two points, x and y, in one-, two-, three-, or higher dimensional space, is given by the following familiar formula**:**

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2},$$

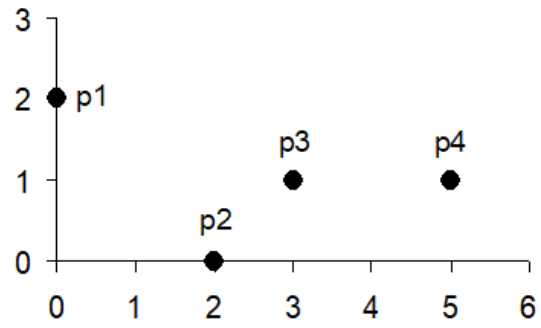
Where n is the number of dimensions (attributes) and xk and yk are, respectively, the kth attributes (components) or data objects x and y.

Example:

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix



Minkowski Distance

Minkowski Distance is a generalization of Euclidean Distance

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{k=1}^{n} |x_k - y_k|^r \right)^{1/r},$$

Where r is a parameter. Where n is the number of dimensions (attributes) and xk and yk are, respectively, the kth attributes (components) or data objects x and y.

The following are the three most common examples of Minkowski distances.

1) r = 1. City block (Manhattan, taxicab, L1 norm) distance.

   A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors

2) r = 2. Euclidean distance
3) $r \to \infty$. "supremum" (Lmax norm, L∞ norm) distance.

   This is the maximum difference between any component of the vectors

Minkowski Distance: Example

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

Common Properties of a Distance

- ✓ Distances, such as the Euclidean distance, have some well-known properties.
- ✓ If d(x, y) is the distance between two points, x and y, then the following properties hold.

1**. Positivity**

(a) d(x, y) >= 0 for all x and y,

(b) d(x, y) = 0 only if x = y

2. Symmetry

d(x,y)= d(y,x) for all x and y.

3. Triangle Inequality

d(x,z) <= d(x, y) + d(y, z) for all points x, y, and z.

Measures that satisfy all three properties are known as metrics.

Common Properties of a Similarity

If s(x, y) is the similarity between points x and y, then the typical properties of similarities are the following:

1. s(x,y) = 1 only if x= y. (0 <= s <= 1)
2. s(x,y)= s(y, x) for all x and y. (Symmetry)

Similarity Measures for Binary Data

Let x and y be two objects that consist of n binary attributes. The comparison of two such objects, i.e., two binary vectors, Leads to the following four quantities (frequencies:)

f_{00} = the number of attributes where **x** is 0 and **y** is 0
f_{01} = the number of attributes where **x** is 0 and **y** is 1
f_{10} = the number of attributes where **x** is 1 and **y** is 0
f_{11} = the number of attributes where **x** is 1 and **y** is 1

Simple Matching Coefficient (SMC):

$$SMC = \frac{\text{number of matching attribute values}}{\text{number of attributes}} = \frac{f_{11} + f_{00}}{f_{01} + f_{10} + f_{11} + f_{00}}.$$

Jaccard Coefficient (J):

$$J = \frac{\text{number of matching presences}}{\text{number of attributes not involved in 00 matches}} = \frac{f_{11}}{f_{01} + f_{10} + f_{11}}.$$

Example :The SMC and Jaccard Similarity Coefficients

Calculate SMC and J for the following two binary vectors.

$\mathbf{x} = (1, 0, 0, 0, 0, 0, 0, 0, 0, 0)$
$\mathbf{y} = (0, 0, 0, 0, 0, 0, 1, 0, 0, 1)$

$f_{01} = 2$ the number of attributes where **x** was 0 and **y** was 1
$f_{10} = 1$ the number of attributes where **x** was 1 and **y** was 0
$f_{00} = 7$ the number of attributes where **x** was 0 and **y** was 0
$f_{11} = 0$ the number of attributes where **x** was 1 and **y** was 1

$$SMC = \frac{f_{11}+f_{00}}{f_{01}+f_{10}+f_{11}+f_{00}} = \frac{0+7}{2+1+0+7} = 0.7$$

$$J = \frac{f_{11}}{f_{01}+f_{10}+f_{11}} = \frac{0}{2+1+0} = 0$$

Cosine Similarity

x and y are two document vectors, then

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \, \|\mathbf{y}\|}, \tag{2.7}$$

where $\cdot$ indicates the vector dot product, $\mathbf{x} \cdot \mathbf{y} = \sum_{k=1}^{n} x_k y_k$, and $\|\mathbf{x}\|$ is the length of vector $\mathbf{x}$, $\|\mathbf{x}\| = \sqrt{\sum_{k=1}^{n} x_k^2} = \sqrt{\mathbf{x} \cdot \mathbf{x}}$.

Example (Cosine Similarity of Two Document Vectors)

$\mathbf{x} = (3, 2, 0, 5, 0, 0, 0, 2, 0, 0)$
$\mathbf{y} = (1, 0, 0, 0, 0, 0, 0, 1, 0, 2)$

$\mathbf{x} \cdot \mathbf{y} = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5$
$\|\mathbf{x}\| = \sqrt{3*3 + 2*2 + 0*0 + 5*5 + 0*0 + 0*0 + 0*0 + 2*2 + 0*0 + 0*0} = 6.48$
$\|\mathbf{y}\| = \sqrt{1*1 + 0*0 + 0*0 + 0*0 + 0*0 + 0*0 + 0*0 + 1*1 + 0*0 + 2*2} = 2.24$
$\cos(\mathbf{x}, \mathbf{y}) = \mathbf{0.31}$

Extended Jaccard Coefficient (Tanimoto Coefficient):

$$EJ(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - \mathbf{x} \cdot \mathbf{y}}.$$

Correlation

The correlation between two data objects that have binary or continuous variables is a measure of the linear relationship between the attributes of the objects**.**

$$\text{corr}(\mathbf{x}, \mathbf{y}) = \frac{\text{covariance}(\mathbf{x}, \mathbf{y})}{\text{standard\_deviation}(\mathbf{x}) * \text{standard\_deviation}(\mathbf{y})} = \frac{s_{xy}}{s_x \; s_y}, \quad (2.10)$$

where we are using the following standard statistical notation and definitions:

$$\text{covariance}(\mathbf{x}, \mathbf{y}) = s_{xy} = \frac{1}{n-1} \sum_{k=1}^{n} (x_k - \overline{x})(y_k - \overline{y}) \quad (2.11)$$

$$\text{standard\_deviation}(\mathbf{x}) = s_x = \sqrt{\frac{1}{n-1} \sum_{k=1}^{n} (x_k - \overline{x})^2}$$

$$\text{standard\_deviation}(\mathbf{y}) = s_y = \sqrt{\frac{1}{n-1} \sum_{k=1}^{n} (y_k - \overline{y})^2}$$

$$\overline{x} = \frac{1}{n} \sum_{k=1}^{n} x_k \text{ is the mean of } \mathbf{x}$$

$$\overline{y} = \frac{1}{n} \sum_{k=1}^{n} y_k \text{ is the mean of } \mathbf{y}$$

Problems:

1)Classify the following attributes as binary, discrete, or continuous. Also classify them as qualitative (nominal or ordinal) or quantitative (interval or ratio). Some cases may have more than one interpretation, so briefly indicate your reasoning if you think there may be some ambiguity.

Example: Age in years. Answer: Discrete, quantitative, ratio

(a) Time in terms of AM or PM. Binary, qualitative, ordinal

(b) Brightness as measured by a light meter. Continuous, quantitative, ratio

(c) Brightness as measured by people's judgments. Discrete, qualitative, ordinal

(d) Angles as measured in degrees between 0◦ and 360◦. Continuous, quantitative, ratio

(e) Bronze, Silver, and Gold medals as awarded at the Olympics. Discrete, qualitative, ordinal

(f) Height above sea level. Continuous, quantitative, interval/ratio (depends on whether sea level is regarded as an arbitrary origin)

(g) Number of patients in a hospital. Discrete, quantitative, ratio

(h) ISBN numbers for books. (Look up the format on the Web.) Discrete, qualitative, nominal (ISBN numbers do have order information, though)

(i) Ability to pass light in terms of the following values: opaque, translucent, transparent. Discrete, qualitative, ordinal

(j) Military rank. Discrete, qualitative, ordinal

(k) Distance from the center of campus. Continuous, quantitative, interval/ratio (depends)

(l) Density of a substance in grams per cubic centimeter. Discrete, quantitative,ratio

(m) Coat check number. (When you attend an event, you can often give your coat to someone who, in turn, gives you a number that you can use to claim your coat when you leave.) Discrete, qualitative, nominal

2) Compute the Hamming distance and the Jaccard similarity between the following two binary vectors

x = 0101010001

y = 0100011000

Solution: Hamming distance = number of different bits = 3

Jaccard Similarity = number of 1-1 matches /(number of bits – number matches) = 2 / 5 = 0.4

4)For the following vectors, x and y, calculate the indicated similarity or distance

measures.

(a) x=: (1, 1, 1, 1), y : (2,2,2,2) cosine, correlation, Euclidean

(b) x=: (0, 1,0, 1), y : (1,0, 1,0) cosine, correlation, Euclidean, Jaccard

(c) x= (0,- 1,0, 1) , y: (1,0,- 1,0)) cosine, correlation Euclidean

(d) x = (1,1 ,0,1 ,0,1) , y : (1,1 ,1 ,0,0,1)) cosine, correlation ,Jaccard

(e) x = (2, -7,0,2,0, -3) , y : (-1, 1,- 1,0,0, -1) cosine, correlation

1.What is data mining ? what are the applications of data mining.

2.Explain Knowledge data discovery KDD with a neat diagram. .

3.Discuss the challenges that motivate the development of Data Mining. .

4.Explain the origin of data mining .

5.What is data mining? Explain various data mining task with examples .

6.What are data and data attributes ? Explain the types and properties of attributes. .

7.Differentiate between discrete and continuous attributes.

8.Distinguish between categorical and numerical attributes. .

9.Explain the types of data sets.

10.List and explain general characteristics of data sets.

11. What is data quality? What are the dimension that asses the data quality.

12.Describe any five data preprocessing approaches. .

13.What is sampling? Explain simple random sampling v/s stratified sampling v/s progressive sampling.

14.Describe the various approaches for feature selection. .

15.What is curse of dimensionality? Explain .

16.What is similarity and dissimilarity? Explain similarity and dissimilarity measures between simple attributes based on different types of attributes. .

17.Discuss the measures of proximity between objects that involve multiple attribute.

18.Explain the cosine similarity for calculating the similarity of two documents with an example. .

19.Consider the following vectors. Find a) Simple Matching Co-efficient b) Jaccard Co-efficient c) Hamming Distance .

i)X: 0101010001 Y: 0100011000

ii)X: 1000000000 Y: 0000001001

20.Distinguish between:

a)Noise and Outlier

b)Jaccard Co-efficient and SMC

c)Discretization and Binarization.

21.For the following vectors find: a) Cosine Similarity b) Correlation c) Jaccard Similarity

i)X: 0101 Y: 1010

ii)X: 110101 Y: 111001

22.For the following vectors find: a) Cosine Similarity b) Correlation

X: 3205000200 Y: 1000000102

23,Discuss whether or not each of the following activities is a data mining task.

(a) Dividing the customers of a company according to their gender.

(b) Dividing the customers of a company according to their profitability.

(c) Computing the total sales of a company.

(d) Sorting a student database based on student identification numbers.

(e) Predicting the outcomes of tossing a (fair) pair of dice.

(f) Predicting the future stock price of a company using historical records.

(g) Monitoring the heart rate of a patient for abnormalities.

(h) Monitoring seismic waves for earthquake activities.

(i) Extracting the frequencies of a sound wave

24. Classify the following attributes as binary, discrete, or continuous. Also classify them as qualitative (nominal or ordinal) or quantitative (interval or ratio). Some cases may have more than one interpretation, so briefly indicate your reasoning if you think there may be some ambiguity.

Example: Age in years. Answer: Discrete, quantitative, ratio

(a) Time in terms of AM or PM.

(b) Brightness as measured by a light meter.

(c) Brightness as measured by people's judgments.

(d) Angles as measured in degrees between 0◦ and 360◦.

(e) Bronze, Silver, and Gold medals as awarded at the Olympics.

(f) Height above sea level.

(g) Number of patients in a hospital.

(h) ISBN numbers for books. (Look up the format on the Web.)

(i) Ability to pass light in terms of the following values: opaque, translucent, transparent.

(j) Military rank.

(k) Distance from the center of campus.