SCS5623 - DATA MINING AND WAREHOUSING

UNIT 1

INTRODUCTION

Data Mining

Data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

Data Mining is defined as extracting information from huge sets of data. In other words, we can say that data mining is the procedure of mining knowledge from data. The information or knowledge extracted so can be used for any of the following applications –

- Market Analysis
- Fraud Detection
- Customer Retention
- Production Control
- Science Exploration

Data Mining Applications

Data mining is highly useful in the following domains –

- Market Analysis and Management
- Corporate Analysis & Risk Management
- Fraud Detection

Apart from these, data mining can also be used in the areas of production control, customer retention, science exploration, sports, astrology, and Internet Web Surf-Aid.

Market Analysis and Management

Listed below are the various fields of market where data mining is used –

- **Customer Profiling –** Data mining helps determine what kind of people buy what kind of products.
- **Identifying Customer Requirements –** Data mining helps in identifying the best products for different customers. It uses prediction to find the factors that may attract new customers.
- **Cross Market Analysis –** Data mining performs association/correlations between product sales.
- **Target Marketing –** Data mining helps to find clusters of model customers who share the same characteristics such as interests, spending habits, income, etc.
- **Determining Customer purchasing pattern –** Data mining helps in determining customer purchasing pattern.
- **Providing Summary Information –** Data mining provides us various multidimensional summary reports.

Corporate Analysis and Risk Management

Data mining is used in the following fields of the Corporate Sector –

- **Finance Planning and Asset Evaluation –** It involves cash flow analysis and prediction, contingent claim analysis to evaluate assets.
- **Resource Planning –** It involves summarizing and comparing the resources and spending.
- **Competition –** It involves monitoring competitors and market directions.

Fraud Detection

Data mining is also used in the fields of credit card services and telecommunication to detect frauds. In fraud telephone calls, it helps to find the destination of the call, duration of the call, time of the day or week, etc. It also analyzes the patterns that deviate from expected norms.

Knowledge discovery in databases (KDD)

Knowledge discovery in databases (KDD) is the process of discovering useful knowledge from a collection of data. This widely used data mining technique is a process that includes data preparation and selection, data cleansing, incorporating prior knowledge on data sets and interpreting accurate solutions from the observed results.

Here is the list of steps involved in the knowledge discovery process –

- **Data Cleaning –** In this step, the noise and inconsistent data is removed.
- **Data Integration –** In this step, multiple data sources are combined.
- **Data Selection –** In this step, data relevant to the analysis task are retrieved from the database.
- **Data Transformation –** In this step, data is transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.
- **Data Mining –** In this step, intelligent methods are applied in order to extract data patterns.
- **Pattern Evaluation –** In this step, data patterns are evaluated.
- **Knowledge Presentation –** In this step, knowledge is represented.

The following diagram shows the process of knowledge discovery

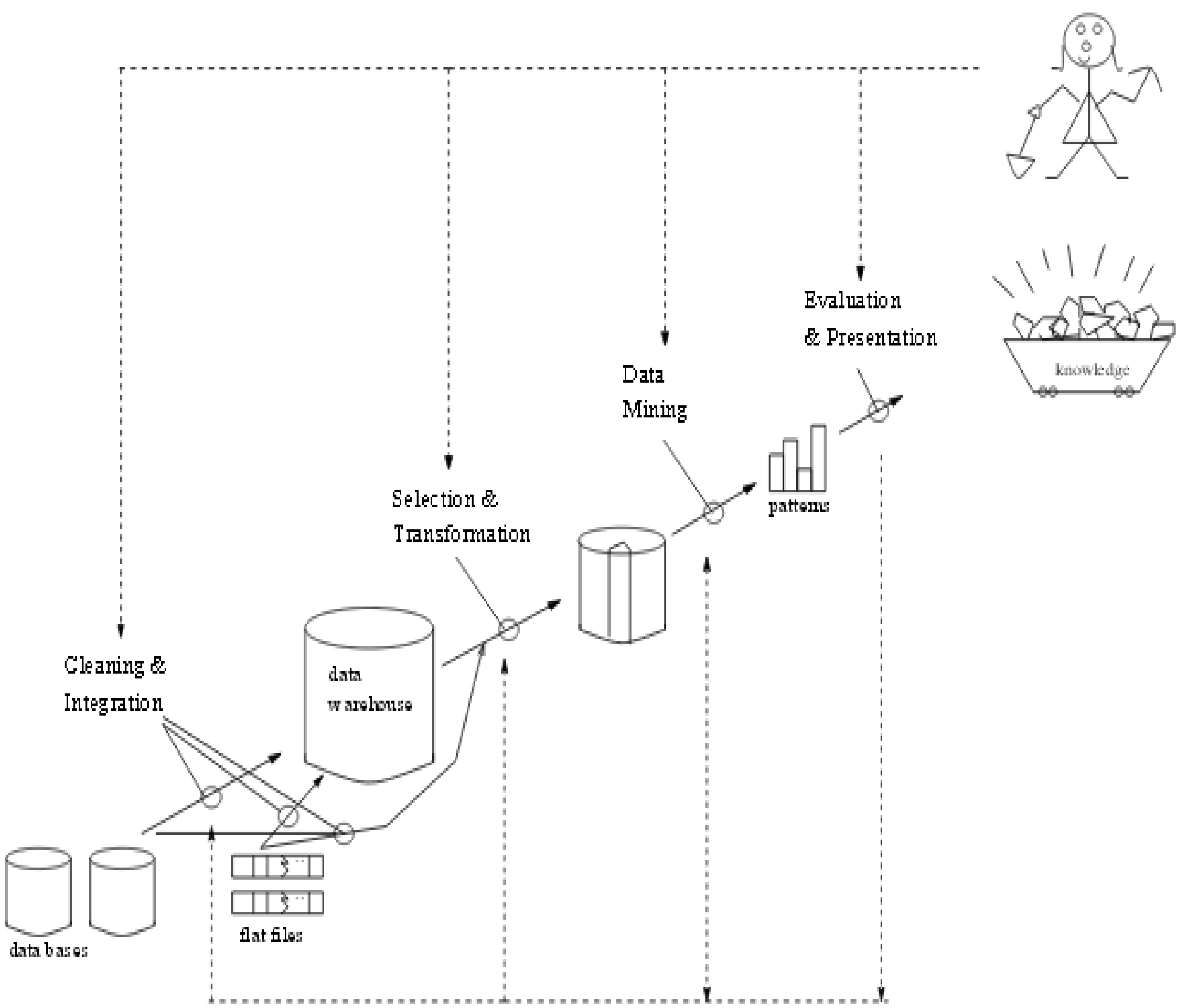


Fig 1: Data Mining as a process of knowledge discovery

Architecture of data mining system

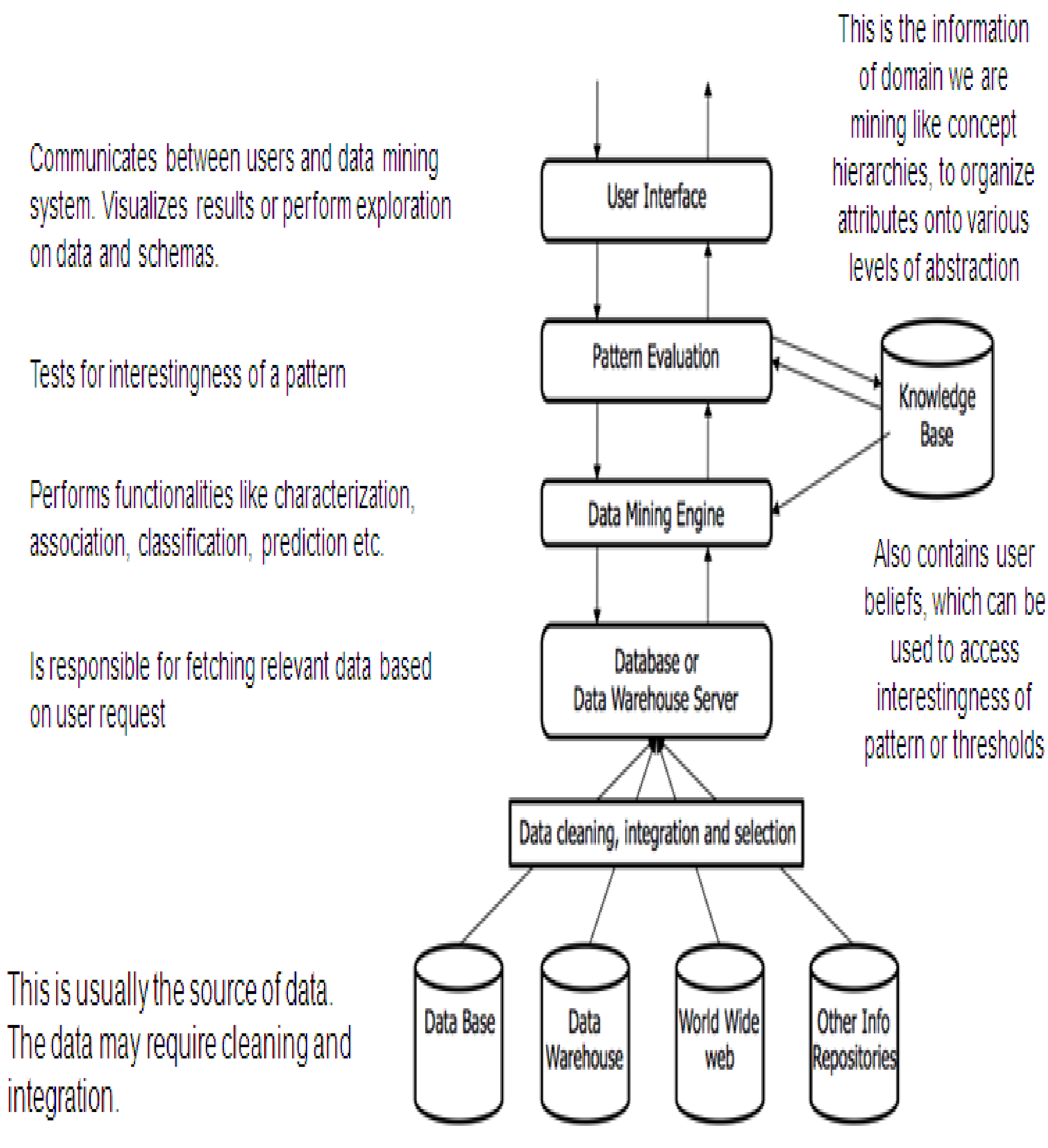


Fig 2: Architecture of data mining system

Data Mining Functionalities - What can be discovered?

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

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

- **Characterization**: Data characterization is a summarization of general features of objects in a target class, and produces what is called *characteristic rules*. The data relevant to a user-specified class are normally retrieved by a database query and run through a summarization module to extract the essence of the data at different levels of abstractions. For example, one may want to characterize the OurVideoStore customers who regularly rent more than 30 movies a year. With concept hierarchies on the attributes describing the target class, the *attribute-oriented induction* method can be used, for example, to carry out data summarization. Note that with a data cube containing summarization of data, simple OLAP operations fit the purpose of data characterization.
- **Discrimination**: Data discrimination produces what are called *discriminant rules* and is basically the comparison of the general features of objects between two classes referred to as the *target class* and the *contrasting class*. For example, one may want to compare the general characteristics of the customers who rented more than 30 movies in the last year with those whose rental account is lower than 5. The techniques used for data discrimination are very similar to the techniques used for data characterization with the exception that data discrimination results include comparative measures.
- **Association analysis**: Association analysis is the discovery of what are commonly called *association rules*. It studies the frequency of items occurring together in transactional databases, and based on a threshold called *support*, identifies the frequent item sets. Another threshold, *confidence*, which is the conditional probability than an item appears in a transaction when another item appears, is used to pinpoint association rules. Association analysis is commonly used for market basket analysis. For example, it could be useful for the OurVideoStore manager to know what movies are often rented together or if there is a relationship between renting a certain type of movies and buying popcorn or pop. The discovered association rules are of the form: P -> Q [s,c], where P and Q are conjunctions of attribute value-pairs, and s (for support) is the probability that P and Q appear together in a transaction and c (for confidence) is the conditional probability that Q appears in a transaction when P is present. For example, the hypothetic association rule:
  RentType(X, "game") AND Age(X, "13-19") -> Buys(X, "pop") [s=2% ,c=55%]
  would indicate that 2% of the transactions considered are of customers aged between 13 and 19 who are renting a game and buying a pop, and that there is a certainty of 55% that teenage customers who rent a game also buy pop.
- **Classification**: Classification analysis is the organization of data in given classes. Also known as *supervised classification*, the classification uses given class labels to order the objects in the data collection. Classification approaches normally use a *training set* where

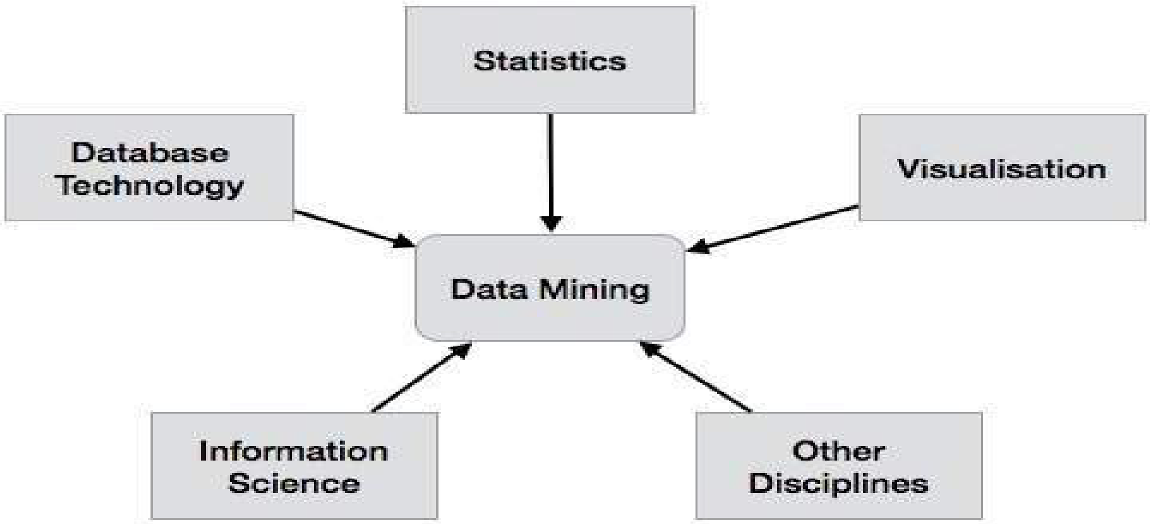
all objects are already associated with known class labels. The classification algorithm learns from the training set and builds a model. The model is used to classify new objects. For example, after starting a credit policy, the OurVideoStore managers could analyze the customers behaviours vis-à-vis their credit, and label accordingly the customers who received credits with three possible labels "safe", "risky" and "very risky". The classification analysis would generate a model that could be used to either accept or reject credit requests in the future.

- **Prediction**: Prediction has attracted considerable attention given the potential implications of successful forecasting in a business context. There are two major types of predictions: one can either try to predict some unavailable data values or pending trends, or predict a class label for some data. The latter is tied to classification. Once a classification model is built based on a training set, the class label of an object can be foreseen based on the attribute values of the object and the attribute values of the classes. Prediction is however more often referred to the forecast of missing numerical values, or increase/ decrease trends in time related data. The major idea is to use a large number of past values to consider probable future values.
- **Clustering**: Similar to classification, clustering is the organization of data in classes. However, unlike classification, in clustering, class labels are unknown and it is up to the clustering algorithm to discover acceptable classes. Clustering is also called *unsupervised classification*, because the classification is not dictated by given class labels. There are many clustering approaches all based on the principle of maximizing the similarity between objects in a same class (*intra-class similarity*) and minimizing the similarity between objects of different classes (*inter-class similarity*).
- **Outlier analysis**: Outliers are data elements that cannot be grouped in a given class or cluster. Also known as *exceptions* or *surprises*, they are often very important to identify. While outliers can be considered noise and discarded in some applications, they can reveal important knowledge in other domains, and thus can be very significant and their analysis valuable.
- **Evolution and deviation analysis**: Evolution and deviation analysis pertain to the study of time related data that changes in time. Evolution analysis models evolutionary trends in data, which consent to characterizing, comparing, classifying or clustering of time related data. Deviation analysis, on the other hand, considers differences between measured values and expected values, and attempts to find the cause of the deviations from the anticipated values.

Data Mining System Classification

A data mining system can be classified according to the following criteria –

- Database Technology
- Statistics
- Machine Learning
- Information Science
- Visualization
- Other Disciplines



Apart from these, a data mining system can also be classified based on the kind of (a) databases mined, (b) knowledge mined, (c) techniques utilized, and (d) applications adapted.

Classification Based on the Databases Mined

We can classify a data mining system according to the kind of databases mined. Database system can be classified according to different criteria such as data models, types of data, etc. And the data mining system can be classified accordingly.For example, if we classify a database according to the data model, then we may have a relational, transactional, object-relational, or data warehouse mining system.

Classification Based on the kind of Knowledge Mined

We can classify a data mining system according to the kind of knowledge mined. It means the data mining system is classified on the basis of functionalities such as –

- Characterization
- Discrimination
- Association and Correlation Analysis
- Classification
- Prediction
- Prediction
- Outlier Analysis
- Evolution Analysis

Classification Based on the Techniques Utilized

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

Classification Based on the Applications Adapted

We can classify a data mining system according to the applications adapted. These applications are as follows –

- Finance
- Telecommunications
- DNA
- Stock Markets
- E-mail

Integrating a Data Mining System with a DB/DW System

If a data mining system is not integrated with a database or a data warehouse system, then there will be no system to communicate with. This scheme is known as the non-coupling scheme. In this scheme, the main focus is on data mining design and on developing efficient and effective algorithms for mining the available data sets.

The list of Integration Schemes is as follows –

- **No Coupling –** In this scheme, the data mining system does not utilize any of the database or data warehouse functions. It fetches the data from a particular source and processes that data using some data mining algorithms. The data mining result is stored in another file.
- **Loose Coupling –** In this scheme, the data mining system may use some of the functions of database and data warehouse system. It fetches the data from the data respiratory managed by these systems and performs data mining on that data. It then stores the mining result either in a file or in a designated place in a database or in a data warehouse.
- **Semi–tight Coupling** - In this scheme, the data mining system is linked with a database or a data warehouse system and in addition to that, efficient implementations of a few data mining primitives can be provided in the database.
- **Tight coupling –** In this coupling scheme, the data mining system is smoothly integrated into the database or data warehouse system. The data mining subsystem is treated as one functional component of an information system.

Major Issues in Data Mining

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

Security and social issues: Security is an important issue with any data collection that is shared and/or is intended to be used for strategic decision-making. In addition, when data is collected for customer profiling, user behaviour understanding, correlating personal data with other information, etc., large amounts of sensitive and private information about individuals or companies is gathered and stored. This becomes controversial given the confidential nature of some of this data and the potential illegal access to the information. Moreover, data mining could disclose new implicit knowledge about individuals or groups that could be against privacy policies, especially if there is potential dissemination of discovered information. Another issue that arises from this concern is the appropriate use of data mining. Due to the value of data, databases of all sorts of content are regularly sold, and because of the competitive advantage that can be attained from implicit knowledge discovered, some important information could be withheld, while other information could be widely distributed and used without control.

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

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

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

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

Performance issues: Many artificial intelligence and statistical methods exist for data analysis and interpretation. However, these methods were often not designed for the very large data sets data mining is dealing with today. Terabyte sizes are common. This raises the issues of scalability and efficiency of the data mining methods when processing considerably large data. Algorithms with exponential and even medium-order polynomial complexity cannot be of practical use for data mining. Linear algorithms are usually the norm. In same theme, sampling can be used for mining instead of the whole dataset. However, concerns such as completeness and choice of samples may arise. Other topics in the issue of performance are *incremental updating*, and parallel programming. There is no doubt that parallelism can help solve the size problem if the dataset can be subdivided and the results can be merged later. Incremental updating is important for merging results from parallel mining, or updating data mining results when new data becomes available without having to re-analyze the complete dataset.

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

Data preprocessing

Why preprocessing ?

1. Real world data are generally
   - Incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
   - Noisy: containing errors or outliers
   - Inconsistent: containing discrepancies in codes or names
2. Tasks in data preprocessing
   - Data cleaning: fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.
   - Data integration: using multiple databases, data cubes, or files.
   - Data transformation: normalization and aggregation.
   - Data reduction: reducing the volume but producing the same or similar analytical results.
   - Data discretization: part of data reduction, replacing numerical attributes with nominal ones.

Data cleaning

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

Data transformation

1. Normalization:
   - Scaling attribute values to fall within a specified range.

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

Data reduction

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

Discretization and generating concept hierarchies

1. Unsupervised discretization - class variable is not used.
    - Equal-interval (equiwidth) binning: split the whole range of numbers in intervals with equal size.
    - Equal-frequency (equidepth) binning: use intervals containing equal number of values.
2. Supervised discretization - uses the values of the class variable.
    - Using class boundaries. Three steps:
        - Sort values.
        - Place breakpoints between values belonging to different classes.
        - If too many intervals, merge intervals with equal or similar class distributions.
    - Entropy (information)-based discretization. Example:
        - Information in a class distribution:
            - Denote a set of five values occurring in tuples belonging to two classes (+ and -) as [+,+,+,-,-]
            - That is, the first 3 belong to "+" tuples and the last 2 - to "-" tuples
            - Then, Info([+,+,+,-,-]) = -(3/5)*log(3/5)-(2/5)*log(2/5) (logs are base 2)

    - 3/5 and 2/5 are relative frequencies (probabilities)
    - Ignoring the order of the values, we can use the following notation: [3,2] meaning 3 values from one class and 2 - from the other.
    - Then, Info([3,2]) = -(3/5)*log(3/5)-(2/5)*log(2/5)
  - Information in a split (2/5 and 3/5 are weight coefficients):
    - Info([+,+],[+,-,-]) = (2/5)*Info([+,+]) + (3/5)*Info([+,-,-])
    - Or, Info([2,0],[1,2]) = (2/5)*Info([2,0]) + (3/5)*Info([1,2])
  - Method:
    - Sort the values;
    - Calculate information in all possible splits;
    - Choose the split that minimizes information;
    - Do not include breakpoints between values belonging to the same class (this will increase information);
    - Apply the same to the resulting intervals until some stopping criterion is satisfied.

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