B.C.A. Semester – 4

BCA-404

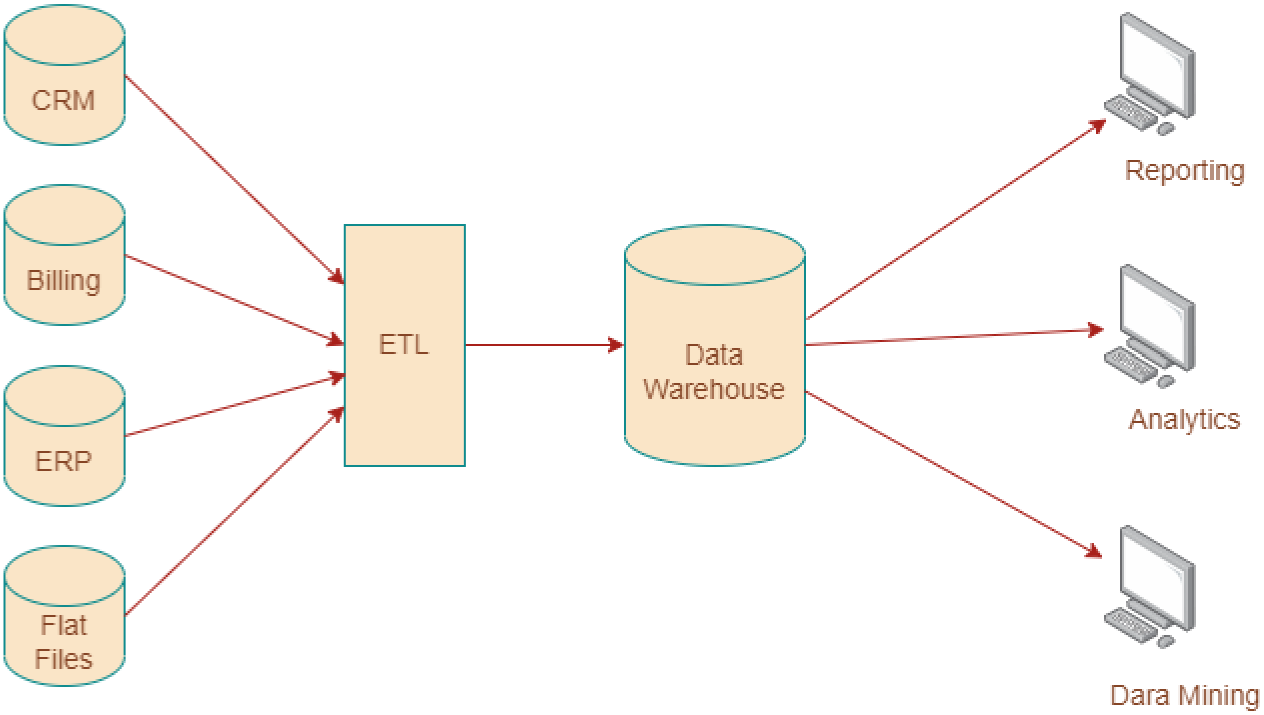
Data Mining & Data Ware Housing

UNIT - 2

Introduction to DMDH

Data Warehouse

A data warehouse is like a big library where we keep a lot of information from different places. It analyzes and understands the information easily. So you can make good decisions based on these facts. You have all the required information that you need in one place. We organize the information so it's easy to find and use. It takes information from different places and put it all together in one place, hence it is easier to understand.



Characteristics of Data Warehouse

Data Warehouse has the following characteristics.

Subject-oriented

A data warehouse focuses on a specific topic like sales, marketing, or distribution. It is designed to provide information about a particular theme rather than the day-to-day operations of an organization.

Integrated

A data warehouse combines data from different sources. These sources are mainframes and relational databases, into a single, reliable format. The data must be organized and structured in a way that allows for effective analysis.

Time-variant

Data in a data warehouse is maintained over time, in weekly/monthly/annual intervals. So you can do historical analysis and the ability to track changes over time.

Non-volatile

Data in a data warehouse is permanent. Data cannot be deleted or modified once it's stored. So you can do historical analysis and ensure that the data is always available in its original state.

By understanding these characteristics, organizations can use data warehouses to make better decisions by analyzing large amounts of data from different sources in a consistent and reliable way.

Data warehousing has some advantages and disadvantages.

Advantages

- Makes data easier to understand
- Continuous updating
- Accessibility

Disadvantages

- Accumulation of irrelevant data
- Data loss and erasure
- Data cleansing and transformation

Functions of Data warehouse

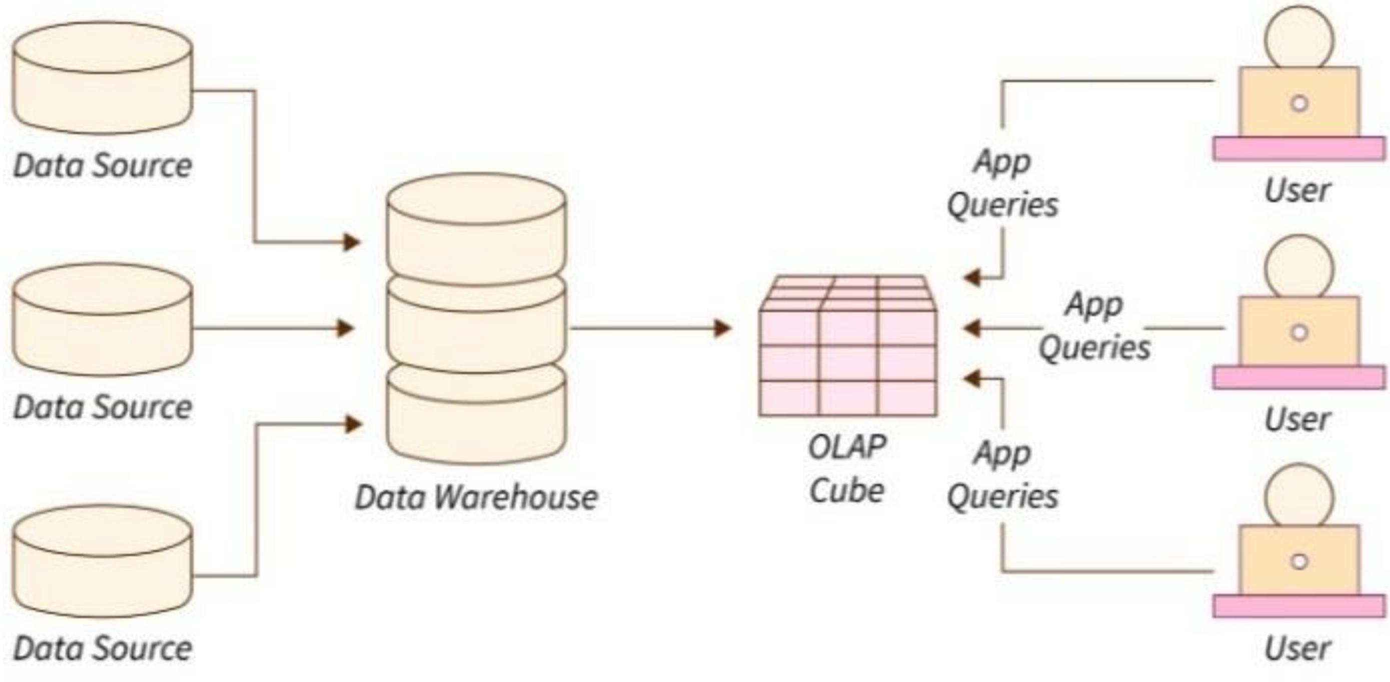
A data warehouse is a collection of data that is organized to provide various functions for managing and analyzing data. Some of the important functions of a data warehouse are −

- Data Consolidation
- Data Cleaning
- Data Integration
- Data Storage
- Data Transformation
- Data Analysis
- Data Reporting
- Data Mining
- Performance Optimization
-

These functions enable organizations to manage and analyze large amounts of data from different sources, and make informed decisions based on reliable and accurate information.

Online Analytical Processing Server (OLAP)

Online Analytical Processing Server (OLAP) is a software. Users can analyze information from many different databases all at once. It uses a multidimensional data model where users can ask questions based on multiple dimensions at the same time. For example, a user could ask for sales data from Delhi in the year 2018. OLAP databases are split up into cubes, which are also called hyper-cubes.



OLAP operations

These are used to analyze data in an OLAP cube. There are five basic operations:

Drill down

This makes the data more detailed by moving down the concept hierarchy or adding a new dimension. For example, in a cube showing sales data by Quarter, drilling down would show sales data by Month.

Roll up

This makes the data less detailed by climbing up the concept hierarchy or reducing dimensions. For example, in a cube showing sales data by City, rolling up would show sales data by Country.

Dice

This selects a sub-cube by choosing two or more dimensions and criteria. For example, in a cube showing sales data by Location, Time, and Item, dicing could select sales data for Delhi or Kolkata, in Q1 or Q2, for Cars or Buses.

Slice

This selects a single dimension and creates a new sub-cube. For example, in a cube showing sales data by Location, Time, and Item, slicing by Time would create a new sub-cube showing sales data for Q1.

Pivot

This rotates the current view to get a new representation. For example, after slicing by Time, pivoting could show the same data but with Location and Item as rows instead of columns

Comparison between Data Warehousing and OLAP

Feature	Data Warehousing	OLAP
Definition	A process of collecting, storing, and managing data from various sources to provide meaningful business insights	A technology that allows users to analyze information from multiple database systems at the same time based on the multi-dimensional data model
Purpose	To make data accessible and understandable for business users	To provide quick and interactive analysis of data from multiple sources
Data structure	Relational database	Multidimensional data model
Data source	Multiple data sources	Multiple data sources
Data type	Historical data	Current and historical data
Data processing	Batch processing	Real-time processing
Operations	Data cleaning, consolidation, integration, transformation, analysis, and reporting	Drill-down, roll-up, slice, dice, and pivot

Cube creation	Not applicable	Cubes are created to support fast and efficient analysis
Query performance	Slower query performance due to complex querying and data processing	Faster query performance due to pre-aggregation and indexing
User type	Business users and data analysts	Business users and data analysts
Use case	Decision-making and strategic planning	Real-time analysis and interactive reporting

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 −

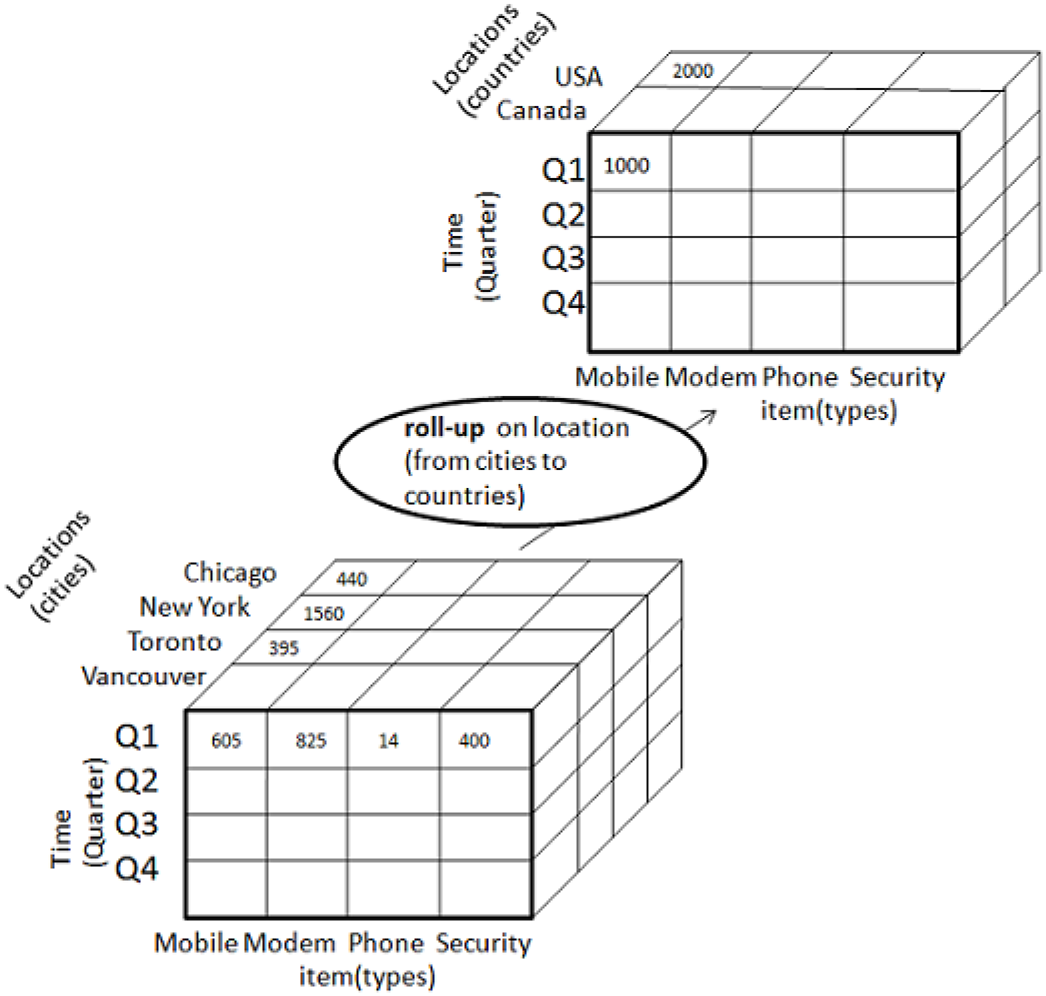
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

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.



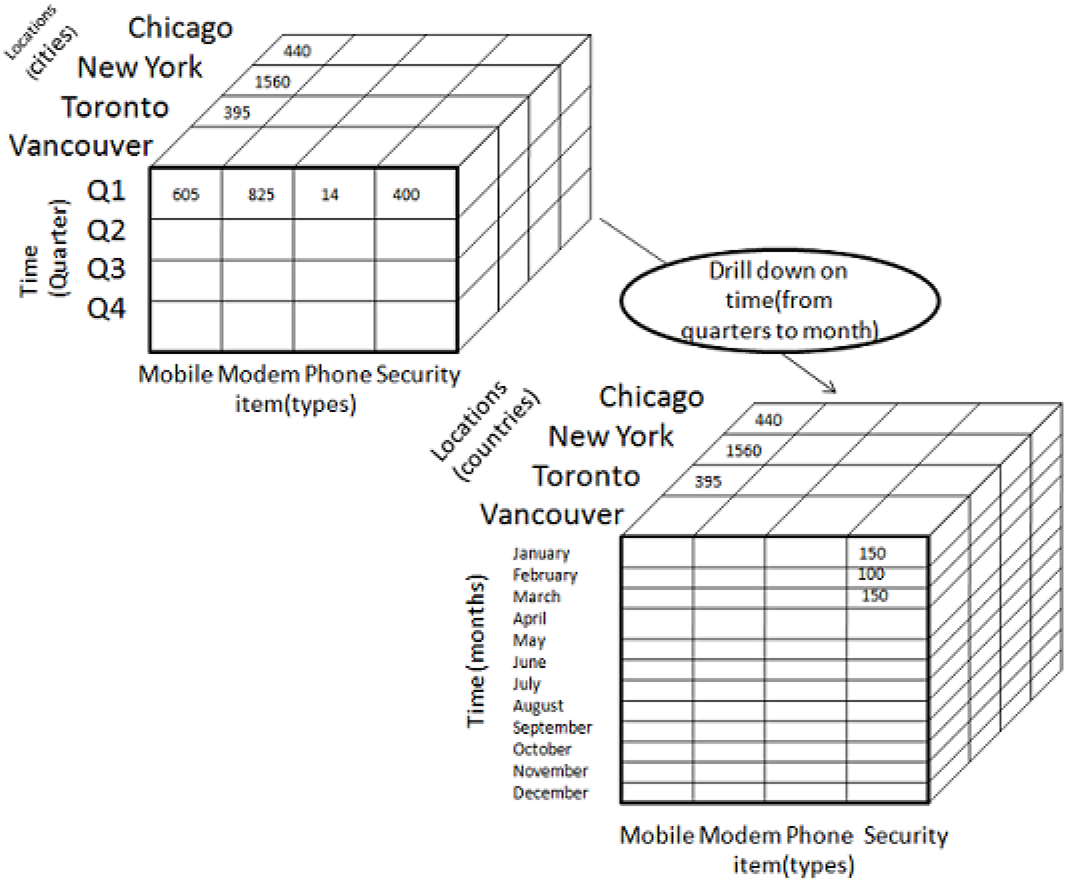
- Roll-up is performed by climbing up a concept hierarchy for the dimension location.
- Initially the concept hierarchy was "street < city < province < country".
- On rolling up, the data is aggregated by ascending the location hierarchy from the level of city to the level of country.
- The data is grouped 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.

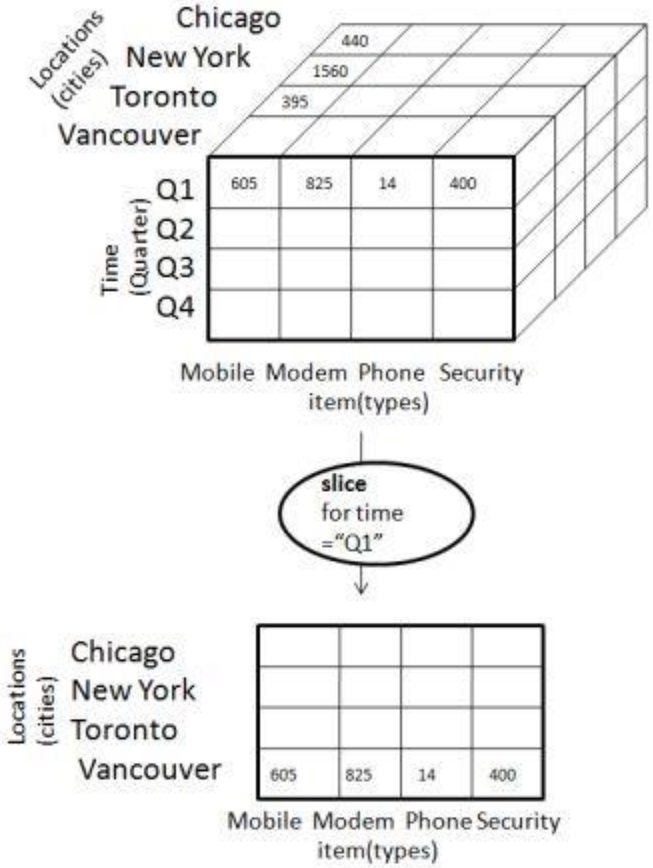
The following diagram illustrates how drill-down works –



- 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 level of 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

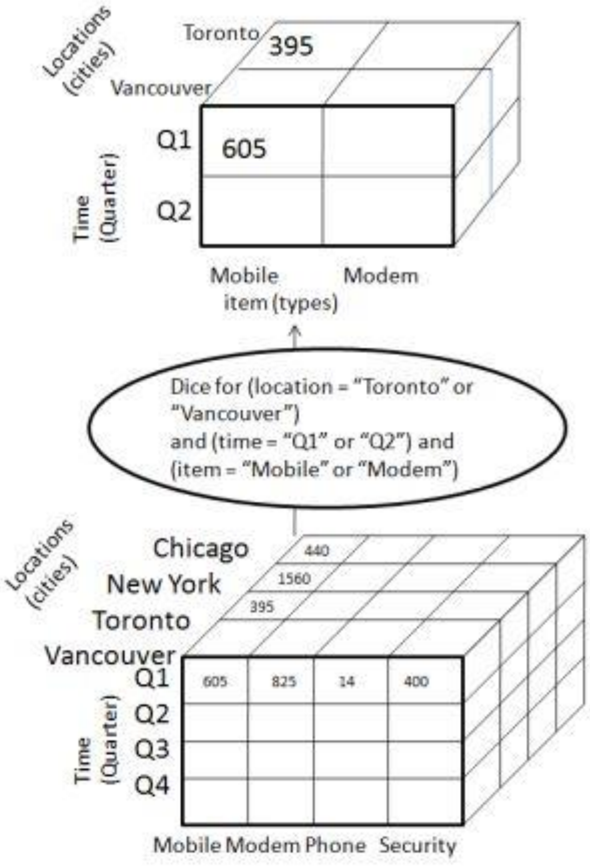
The slice operation selects one particular dimension from a given cube and provides a new sub-cube. Consider the following diagram that shows how slice works.



- 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.

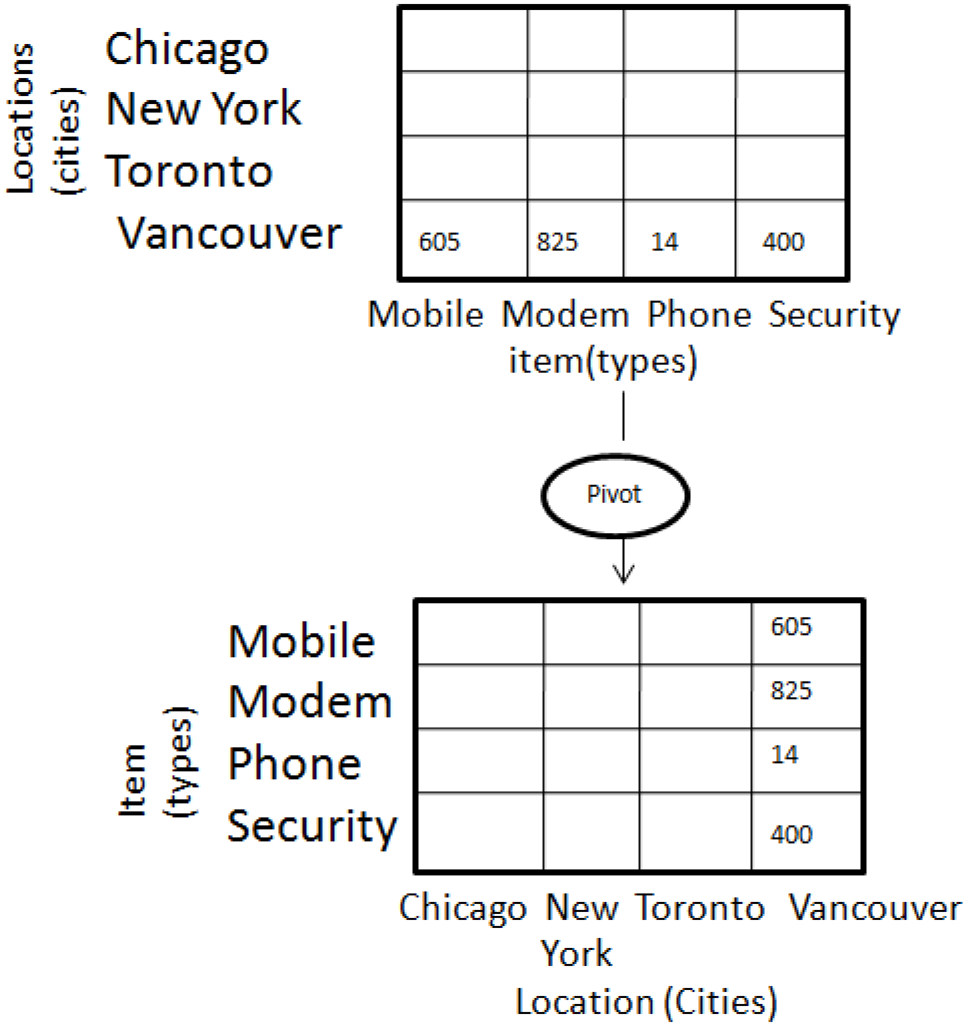


The dice operation on the cube based on the following selection criteria involves three dimensions.

- (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 vs OLTP

Sr.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.
11	Database size is from 100 GB to 1 TB	Database size is from 100 MB to 1 GB.
12	Highly flexible.	Provides high performance.

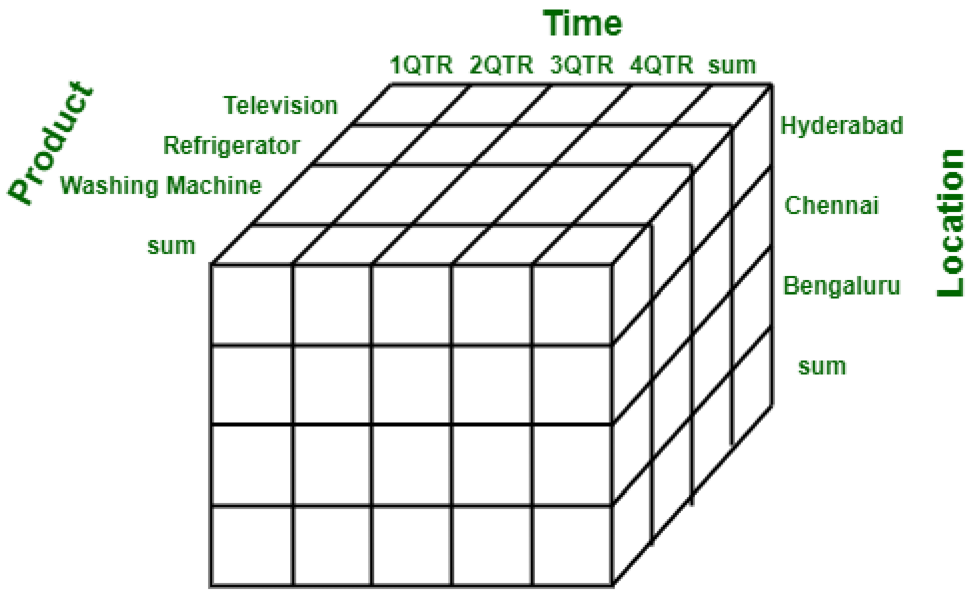
A multidimensional data model

The multi-Dimensional Data Model is a method which is used for ordering data in the database along with good arrangement and assembling of the contents in the database.

The Multi Dimensional Data Model allows customers to interrogate analytical questions associated with market or business trends, unlike relational databases which allow customers to access data in the form of queries. They allow users to rapidly receive answers to the requests which they made by creating and examining the data comparatively fast.

OLAP (online analytical processing) and data warehousing uses multi dimensional databases. It is used to show multiple dimensions of the data to users.

It represents data in the form of data cubes. Data cubes allow to model and view the data from many dimensions and perspectives. It is defined by dimensions and facts and is represented by a fact table. Facts are numerical measures and fact tables contain measures of the related dimensional tables or names of the facts.



Multidimensional Data Representation

Working on a Multidimensional Data Model

On the basis of the pre-decided steps, the Multidimensional Data Model works.

The following stages should be followed by every project for building a Multi Dimensional Data Model :

Stage 1 : Assembling data from the client : In first stage, a Multi Dimensional Data Model collects correct data from the client. Mostly, software professionals provide simplicity to the client about the range of data which can be gained with the selected technology and collect the complete data in detail.

Stage 2 : Grouping different segments of the system : In the second stage, the Multi Dimensional Data Model recognizes and classifies all the data to the respective section they belong to and also builds it problem-free to apply step by step.

Stage 3 : Noticing the different proportions : In the third stage, it is the basis on which the design of the system is based. In this stage, the main factors are recognized according to the user's point of view. These factors are also known as "Dimensions".

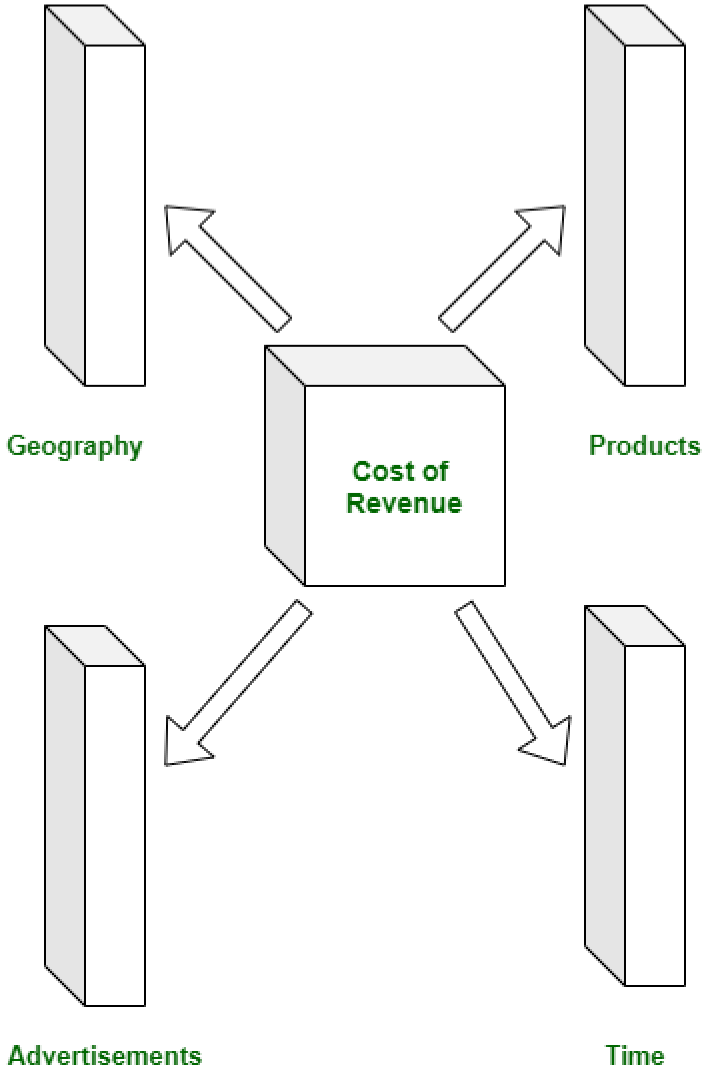
Stage 4 : Preparing the actual-time factors and their respective qualities : In the fourth stage, the factors which are recognized in the previous step are used further for identifying the related qualities. These qualities are also known as "attributes" in the database.

Stage 5 : Finding the actuality of factors which are listed previously and their qualities : In the fifth stage, A Multi Dimensional Data Model separates and differentiates the actuality from the factors which are collected by it. These actually play a significant role in the arrangement of a Multi Dimensional Data Model.

Stage 6 : Building the Schema to place the data, with respect to the information collected from the steps above : In the sixth stage, on the basis of the data which was collected previously, a Schema is built.

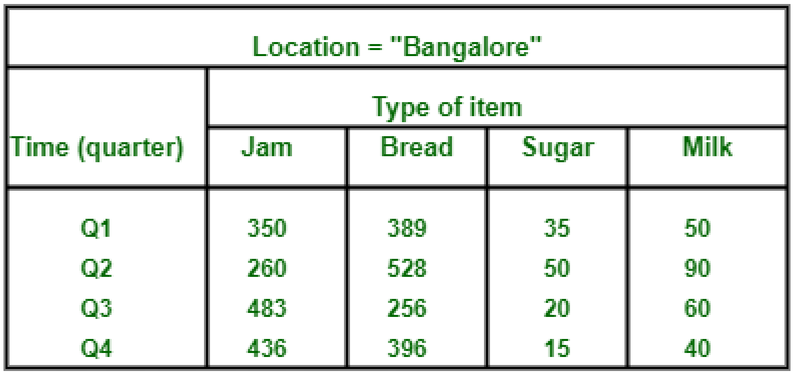
For Example :

1. Let us take the example of a firm. The revenue cost of a firm can be recognized on the basis of different factors such as geographical location of firm's workplace, products of the firm, advertisements done, time utilized to flourish a product, etc.



Example 1

2. Let us take the example of the data of a factory which sells products per quarter in Bangalore. The data is represented in the table given below :

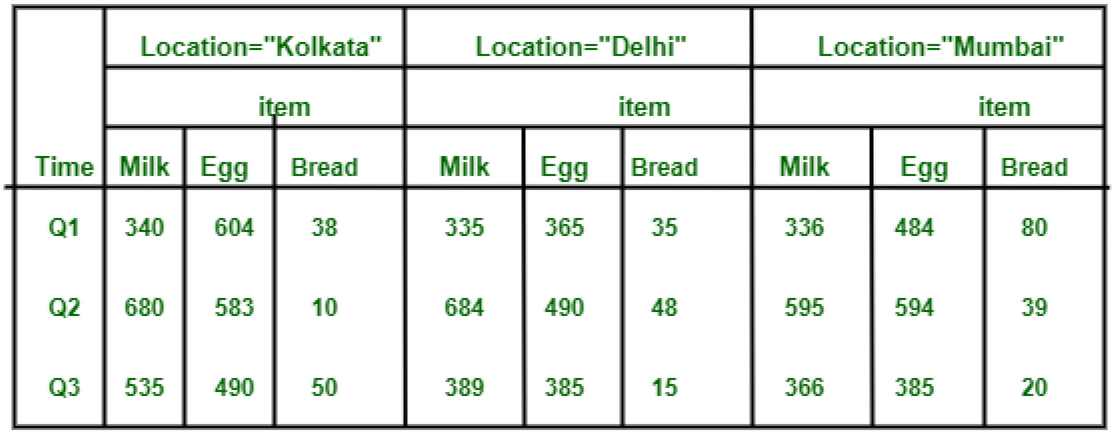


Location = "Bangalore"				
Time (quarter)	Type of item			
	Jam	Bread	Sugar	Milk
Q1	350	389	35	50
Q2	260	528	50	90
Q3	483	256	20	60
Q4	436	396	15	40

2D factory data

In the above given presentation, the factory's sales for Bangalore are, for the time dimension, which is organized into quarters and the dimension of items, which is sorted according to the kind of item which is sold. The facts here are represented in rupees (in thousands).

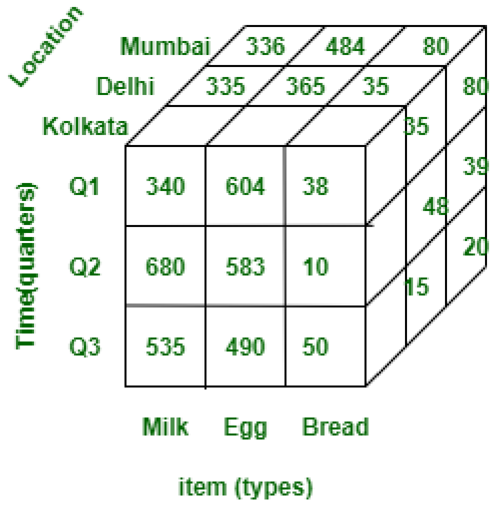
Now, if we desire to view the data of the sales in a three-dimensional table, then it is represented in the diagram given below. Here the data of the sales is represented as a two dimensional table. Let us consider the data according to item, time and location (like Kolkata, Delhi, Mumbai). Here is the table :



	Location="Kolkata"			Location="Delhi"			Location="Mumbai"		
	item			item			item		
Time	Milk	Egg	Bread	Milk	Egg	Bread	Milk	Egg	Bread
Q1	340	604	38	335	365	35	336	484	80
Q2	680	583	10	684	490	48	595	594	39
Q3	535	490	50	389	385	15	366	385	20

3D data representation as 2D

This data can be represented in the form of three dimensions conceptually, which is shown in the image below :



3D data representation

Features of multidimensional data models:

Measures: Measures are numerical data that can be analyzed and compared, such as sales or revenue. They are typically stored in fact tables in a multidimensional data model.

Dimensions: Dimensions are attributes that describe the measures, such as time, location, or product. They are typically stored in dimension tables in a multidimensional data model.

Cubes: Cubes are structures that represent the multidimensional relationships between measures and dimensions in a data model. They provide a fast and efficient way to retrieve and analyze data.

Aggregation: Aggregation is the process of summarizing data across dimensions and levels of detail. This is a key feature of multidimensional data models, as it enables users to quickly analyze data at different levels of granularity.

Drill-down and roll-up: Drill-down is the process of moving from a higher-level summary of data to a lower level of detail, while roll-up is the opposite process of moving from a lower-level detail to a higher-level summary. These features enable users to explore data in greater detail and gain insights into the underlying patterns.

Hierarchies: Hierarchies are a way of organizing dimensions into levels of detail. For example, a time dimension might be organized into years, quarters, months, and days. Hierarchies provide a way to navigate the data and perform drill-down and roll-up operations.

OLAP (Online Analytical Processing): OLAP is a type of multidimensional data model that supports fast and efficient querying of large datasets. OLAP systems are designed to handle complex queries and provide fast response times.

Advantages of Multi Dimensional Data Model

The following are the advantages of a multi-dimensional data model :

A multi-dimensional data model is easy to handle.

It is easy to maintain.

Its performance is better than that of normal databases (e.g. relational databases).

The representation of data is better than traditional databases. That is because the multi-dimensional databases are multi-viewed and carry different types of factors.

It is workable on complex systems and applications, contrary to the simple one-dimensional database systems.

The compatibility in this type of database is an upliftment for projects having lower bandwidth for maintenance staff.

Disadvantages of Multi Dimensional Data Model

The following are the disadvantages of a Multi Dimensional Data Model :

The multi-dimensional Data Model is slightly complicated in nature and it requires professionals to recognize and examine the data in the database.

During the work of a Multi-Dimensional Data Model, when the system caches, there is a great effect on the working of the system.

It is complicated in nature due to which the databases are generally dynamic in design.

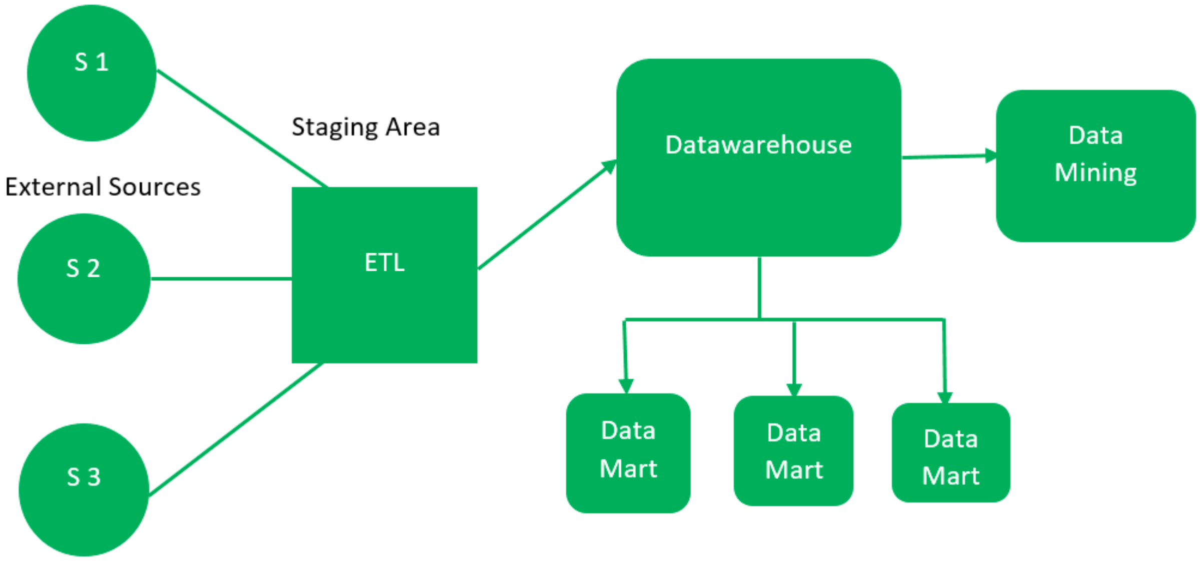
The path to achieving the end product is complicated most of the time.

As the Multi Dimensional Data Model has complicated systems, databases have a large number of databases due to which the system is very insecure when there is a security break.

Data Warehouse Architecture

A data-warehouse is a heterogeneous collection of different data sources organised under a unified schema. There are 2 approaches for constructing data-warehouse: Top-down approach and Bottom-up approach are explained as below.

1. Top-down approach:



The essential components are discussed below:

External Sources –
External source is a source from where data is collected irrespective of the type of data. Data can be structured, semi structured and unstructured as well.

Stage Area –
Since the data, extracted from the external sources does not follow a particular format, so there is a need to validate this data to load into datawarehouse. For this purpose, it is recommended to use ETL tool.

E(Extracted): Data is extracted from External data source.

T(Transform): Data is transformed into the standard format.

L(Load): Data is loaded into datawarehouse after transforming it into the standard format.

Data-warehouse –
After cleansing of data, it is stored in the datawarehouse as central repository. It actually stores the meta data and the actual data gets stored in the data marts. Note that datawarehouse stores the data in

its purest form in this top-down approach.

Data Marts –

Data mart is also a part of storage component. It stores the information of a particular function of an organisation which is handled by single authority. There can be as many number of data marts in an organisation depending upon the functions. We can also say that data mart contains subset of the data stored in datawarehouse.

Data Mining –

The practice of analysing the big data present in datawarehouse is data mining. It is used to find the hidden patterns that are present in the database or in datawarehouse with the help of algorithm of data mining.

This approach is defined by Inmon as – datawarehouse as a central repository for the complete organisation and data marts are created from it after the complete datawarehouse has been created.

Advantages of Top-Down Approach –

Since the data marts are created from the datawarehouse, provides consistent dimensional view of data marts.

Also, this model is considered as the strongest model for business changes. That's why, big organisations prefer to follow this approach.

Creating data mart from datawarehouse is easy.

Improved data consistency: The top-down approach promotes data consistency by ensuring that all data marts are sourced from a common data warehouse. This ensures that all data is standardized, reducing the risk of errors and inconsistencies in reporting.

Easier maintenance: Since all data marts are sourced from a central data warehouse, it is easier to maintain and update the data in a top-down approach. Changes can be made to the data warehouse, and those changes will automatically propagate to all the data marts that rely on it.

Better scalability: The top-down approach is highly scalable, allowing organizations to add new data marts as needed without disrupting the existing infrastructure. This is particularly important for organizations that are experiencing rapid growth or have evolving business needs.

Improved governance: The top-down approach facilitates better governance by enabling centralized control of data access, security, and quality. This ensures that all data is managed consistently and that it meets the organization's standards for quality and compliance.

Reduced duplication: The top-down approach reduces data duplication by ensuring that data is stored only once in the data warehouse. This saves storage space and reduces the risk of data inconsistencies.

Better reporting: The top-down approach enables better reporting by providing a consistent view of data across all data marts. This makes it easier to create accurate and timely reports, which can improve decision-making and drive better business outcomes.

Better data integration: The top-down approach enables better data integration by ensuring that all data marts are sourced from a common data warehouse. This makes it easier to integrate data from different sources and provides a more complete view of the organization's data.

Disadvantages of Top-Down Approach –

The cost, time taken in designing and its maintenance is very high.

Complexity: The top-down approach can be complex to implement and maintain, particularly for large organizations with complex data needs. The design and implementation of the data warehouse and data marts can be time-consuming and costly.

Lack of flexibility: The top-down approach may not be suitable for organizations that require a high degree of flexibility in their data reporting and analysis. Since the design of the data warehouse and data marts is pre-determined, it may not be possible to adapt to new or changing business requirements.

Limited user involvement: The top-down approach can be dominated by IT departments, which may lead to limited user involvement in the design and implementation process. This can result in data marts that do not meet the specific needs of business users.

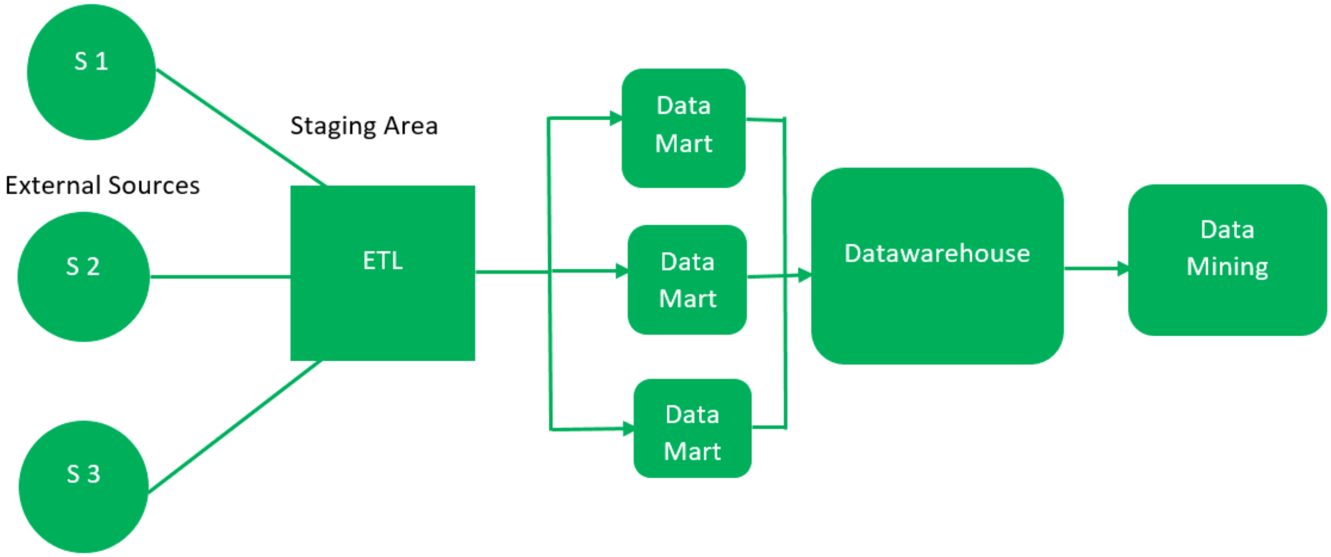
Data latency: The top-down approach may result in data latency, particularly when data is sourced from multiple systems. This can impact the accuracy and timeliness of reporting and analysis.

Data ownership: The top-down approach can create challenges around data ownership and control. Since data is centralized in the data warehouse, it may not be clear who is responsible for maintaining and updating the data.

Cost: The top-down approach can be expensive to implement and maintain, particularly for smaller organizations that may not have the resources to invest in a large-scale data warehouse and associated data marts.

Integration challenges: The top-down approach may face challenges in integrating data from different sources, particularly when data is stored in different formats or structures. This can lead to data inconsistencies and inaccuracies.

2. Bottom-up approach:



First, the data is extracted from external sources (same as happens in top-down approach).

Then, the data go through the staging area (as explained above) and loaded into data marts instead of datawarehouse. The data marts are created first and provide reporting capability. It addresses a single business area.

These data marts are then integrated into datawarehouse.

This approach is given by Kinball as – data marts are created first and provides a thin view for analyses and datawarehouse is created after complete data marts have been created.

Advantages of Bottom-Up Approach –

As the data marts are created first, so the reports are quickly generated.

We can accommodate more number of data marts here and in this way datawarehouse can be extended.

Also, the cost and time taken in designing this model is low comparatively.

Incremental development: The bottom-up approach supports incremental development, allowing for the creation of data marts one at a time. This allows for quick wins and incremental improvements in data reporting and analysis.

User involvement: The bottom-up approach encourages user involvement in the design and implementation process. Business users can provide feedback on the data marts and reports, helping to ensure that the data marts meet their specific needs.

Flexibility: The bottom-up approach is more flexible than the top-down approach, as it allows for the creation of data marts based on specific business needs. This approach can be particularly useful for organizations that require a high degree of flexibility in their reporting and analysis.

Faster time to value: The bottom-up approach can deliver faster time to value, as the data marts can be created more quickly than a centralized data warehouse. This can be particularly useful for smaller organizations with limited resources.

Reduced risk: The bottom-up approach reduces the risk of failure, as data marts can be tested and refined before being incorporated into a larger data warehouse. This approach can also help to identify and address potential data quality issues early in the process.

Scalability: The bottom-up approach can be scaled up over time, as new data marts can be added as needed. This approach can be particularly useful for organizations that are growing rapidly or undergoing significant change.

Data ownership: The bottom-up approach can help to clarify data ownership and control, as each data mart is typically owned and managed by a specific business unit. This can help to ensure that data is accurate and up-to-date, and that it is being used in a consistent and appropriate way across the organization.

Disadvantage of Bottom-Up Approach –

This model is not strong as top-down approach as dimensional view of data marts is not consistent as it is in above approach.

Data silos: The bottom-up approach can lead to the creation of data silos, where different business units create their own data marts without considering the needs of other parts of the organization. This can lead to inconsistencies and redundancies in the data, as well as difficulties in integrating data across the organization.

Integration challenges: Because the bottom-up approach relies on the integration of multiple data marts, it can be more difficult to integrate data from different sources and ensure consistency across the organization. This can lead to issues with data quality and accuracy.

Duplication of effort: In a bottom-up approach, different business units may duplicate effort by creating their own data marts with similar or overlapping data. This can lead to inefficiencies and higher costs in data management.

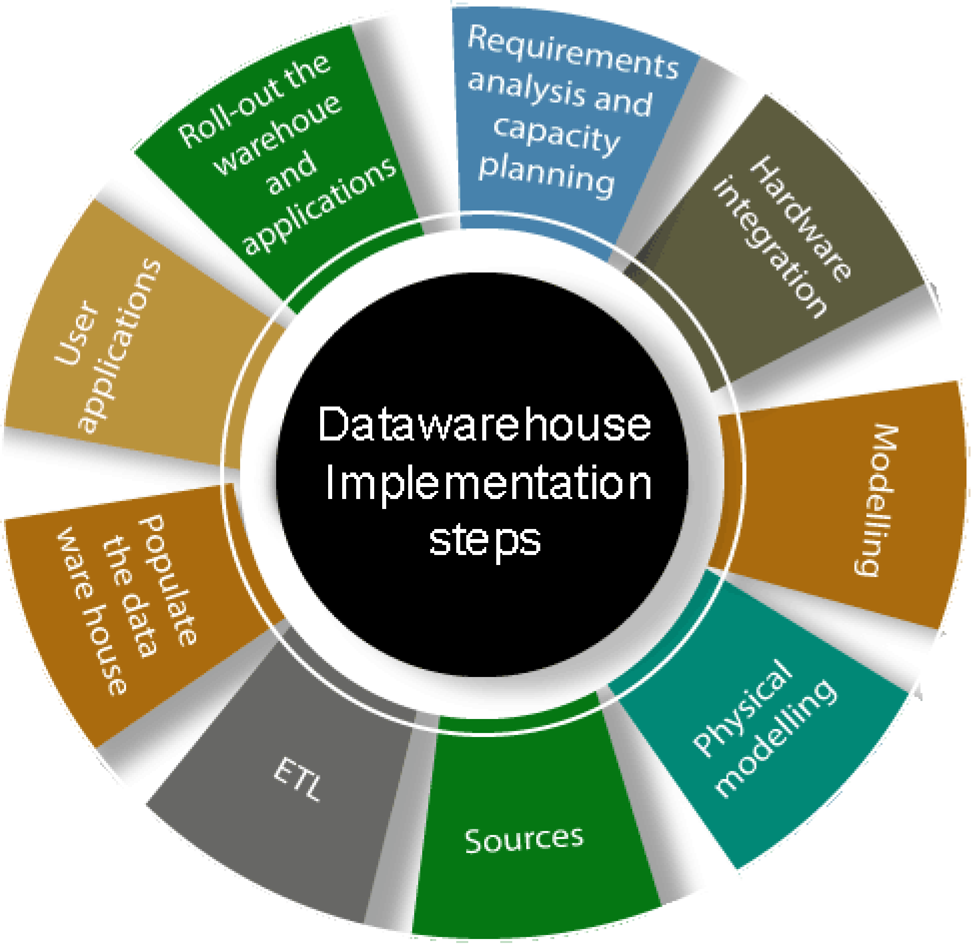
Lack of enterprise-wide view: The bottom-up approach can result in a lack of enterprise-wide view, as data marts are typically designed to meet the needs of specific business units rather than the organization as a whole. This can make it difficult to gain a comprehensive understanding of the organization’s data and business processes.

Complexity: The bottom-up approach can be more complex than the top-down approach, as it involves the integration of multiple data marts with varying levels of complexity and granularity. This can make it more difficult to manage and maintain the data warehouse over time.

Risk of inconsistency: Because the bottom-up approach allows for the creation of data marts with different structures and granularities, there is a risk of inconsistency in the data. This can make it difficult to compare data across different parts of the organization or to ensure that reports are accurate and reliable.

ata Warehouse Implementation

There are various implementation in data warehouses which are as follows


Datawarehouse Implementation steps
Requirements analysis and capacity planning
Hardware integration
Modelling
Physical modelling
Sources
ETL
Populate the data ware house
User applications
Roll-out the warehoue and applications

1. Requirements analysis and capacity planning: The first process in data warehousing involves defining enterprise needs, defining architectures, carrying out capacity planning, and selecting the hardware and software tools. This step will contain be consulting senior management as well as the different stakeholder.

2. Hardware integration: Once the hardware and software has been selected, they require to be put by integrating the servers, the storage methods, and the user software tools.

3. Modeling: Modelling is a significant stage that involves designing the warehouse schema and views. This may contain using a modeling tool if the data warehouses are sophisticated.

4. Physical modeling: For the data warehouses to perform efficiently, physical modeling is needed. This contains designing the physical data warehouse organization, data placement, data partitioning, deciding on access techniques, and indexing.

5. Sources: The information for the data warehouse is likely to come from several data sources. This step contains identifying and connecting the sources using the gateway, ODBC drives, or another wrapper.

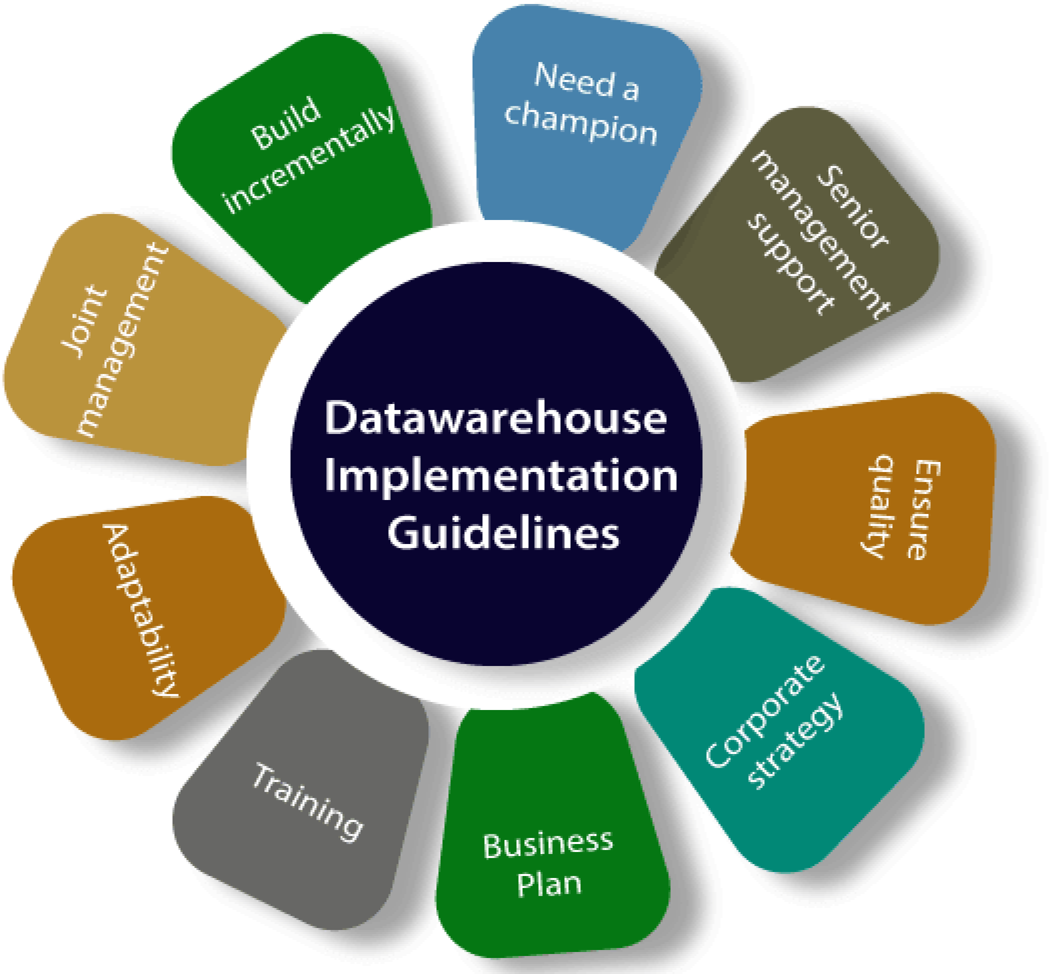
6. ETL: The data from the source system will require to go through an ETL phase. The process of designing and implementing the ETL phase may contain defining a suitable ETL tool vendors and purchasing and implementing the tools. This may contains customize the tool to suit the need of the enterprises.

7. Populate the data warehouses: Once the ETL tools have been agreed upon, testing the tools will be needed, perhaps using a staging area. Once everything is working adequately, the ETL tools may be used in populating the warehouses given the schema and view definition.

8. User applications: For the data warehouses to be helpful, there must be end-user applications. This step contains designing and implementing applications required by the end-users.

9. Roll-out the warehouses and applications: Once the data warehouse has been populated and the end-client applications tested, the warehouse system and the operations may be rolled out for the user's community to use.

Implementation Guidelines



1. Build incrementally: Data warehouses must be built incrementally. Generally, it is recommended that a data marts may be created with one particular project in mind, and once it is implemented, several other sections of the enterprise may also want to implement similar systems. An enterprise data warehouses can then be implemented in an iterative manner allowing all data marts to extract information from the data warehouse.

2. Need a champion: A data warehouses project must have a champion who is active to carry out considerable researches into expected price and benefit of the project. Data warehousing projects requires inputs from many units in an enterprise and therefore needs to be driven by someone who is needed for interacting with people in the enterprises and can actively persuade colleagues.

3. Senior management support: A data warehouses project must be fully supported by senior management. Given the resource-intensive feature of such project and the time they can take to implement, a warehouse project signal for a sustained commitment from senior management.

4. Ensure quality: The only record that has been cleaned and is of a quality that is implicit by the organizations should be loaded in the data warehouses.

5. Corporate strategy: A data warehouse project must be suitable for corporate strategies and business goals. The purpose of the project must be defined before the beginning of the projects.

6. Business plan: The financial costs (hardware, software, and peopleware), expected advantage, and a project plan for a data warehouses project must be clearly outlined and understood by all stakeholders. Without such understanding, rumors about expenditure and benefits can become the only sources of data, subversion the projects.

7. Training: Data warehouses projects must not overlook data warehouses training requirements. For a data warehouses project to be successful, the customers must be trained to use the warehouses and to understand its capabilities.

8. Adaptability: The project should build in flexibility so that changes may be made to the data warehouses if and when required. Like any system, a data warehouse will require to change, as the needs of an enterprise change.

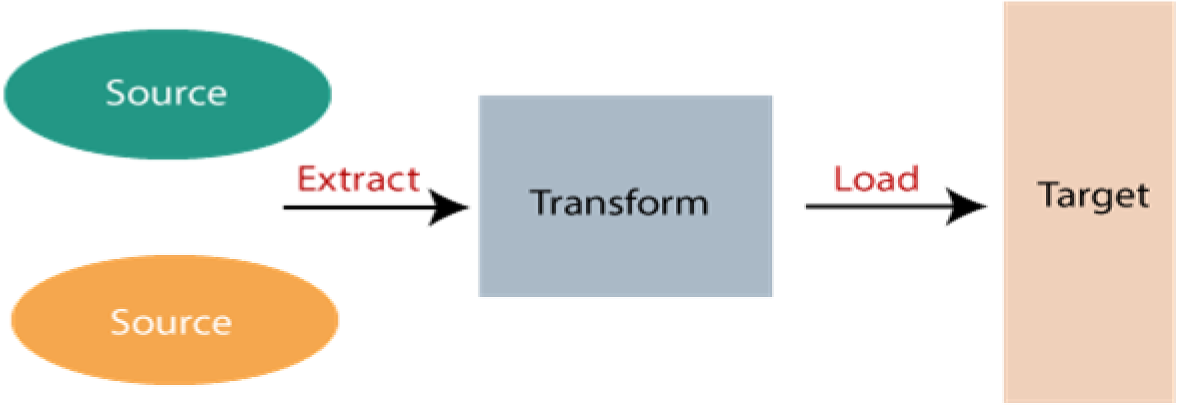
9. Joint management: The project must be handled by both IT and business professionals in the enterprise. To ensure that proper communication with the stakeholder and which the project is the target for assisting the enterprise's business, the business professional must be involved in the project along with technical professionals.

Data Mining Vs Data Warehousing

Data warehouse refers to the process of compiling and organizing data into one common database, whereas **data mining** refers to the process of extracting useful data from the databases. The data mining process depends on the data compiled in the data warehousing phase to recognize meaningful patterns. A data warehousing is created to support management systems.

Data Warehouse:

A **Data Warehouse** refers to a place where data can be stored for useful mining. It is like a quick computer system with exceptionally huge data storage capacity. Data from the various organization's systems are copied to the Warehouse, where it can be fetched and conformed to delete errors. Here, advanced requests can be made against the warehouse storage of data.



Data Warehousing Process

Data warehouse combines data from numerous sources which ensure the data quality, accuracy, and consistency. Data warehouse boosts system execution by separating analytics processing from transnational databases. Data flows into a data warehouse from different databases. A data warehouse works by sorting out data into a pattern that depicts the format and types of data. Query tools examine the data tables using patterns.

Data warehouses and **databases** both are relative data systems, but both are made to serve different purposes. A data warehouse is built to store a huge amount of historical data and empowers fast requests over all the data, typically using **Online Analytical Processing** (OLAP). A database is made to store current transactions and allow quick access to specific transactions for ongoing business processes, commonly known as **Online Transaction Processing** (OLTP).

Important Features of Data Warehouse

The Important features of Data Warehouse are given below:

1. Subject Oriented

A data warehouse is subject-oriented. It provides useful data about a subject instead of the company's ongoing operations, and these subjects can be customers, suppliers, marketing, product, promotion, etc. A data warehouse usually focuses on modeling and analysis of data that helps the business organization to make data-driven decisions.

2. Time-Variant:

The different data present in the data warehouse provides information for a specific period.

3. Integrated

A data warehouse is built by joining data from heterogeneous sources, such as social databases, level documents, etc.

4. Non- Volatile

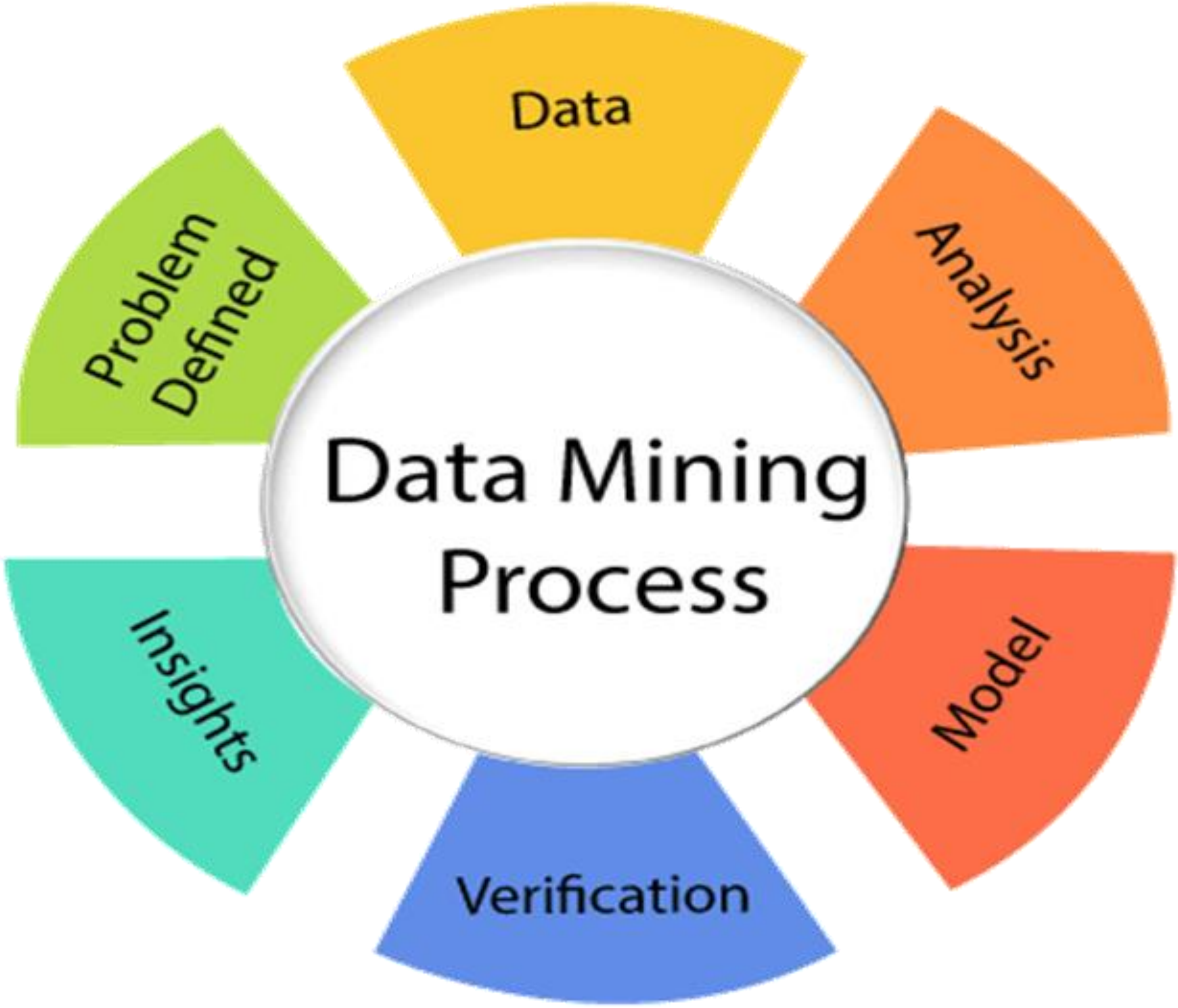
It means, once data entered into the warehouse cannot be change.

Advantages of Data Warehouse:

- More accurate data access
- Improved productivity and performance
- Cost-efficient
- Consistent and quality data

Data Mining:

Data mining refers to the analysis of data. It is the computer-supported process of analyzing huge sets of data that have either been compiled by computer systems or have been downloaded into the computer. In the data mining process, the computer analyzes the data and extract useful information from it. It looks for hidden patterns within the data set and try to predict future behavior. Data mining is primarily used to discover and indicate relationships among the data sets.



Data mining aims to enable business organizations to view business behaviors, trends relationships that allow the business to make data-driven decisions. It is also known as knowledge Discover in Database (KDD). Data mining tools utilize AI, statistics, databases, and machine learning systems to discover the relationship between the data. Data mining tools can support business-related questions that traditionally time-consuming to resolve any issue.

Important features of Data Mining:

The important features of Data Mining are given below:

- It utilizes the Automated discovery of patterns.
- It predicts the expected results.

- It focuses on large data sets and databases
- It creates actionable information.

Advantages of Data Mining:

i. Market Analysis:

Data Mining can predict the market that helps the business to make the decision. For example, it predicts who is keen to purchase what type of products.

ii. Fraud detection:

Data Mining methods can help to find which cellular phone calls, insurance claims, credit, or debit card purchases are going to be fraudulent.

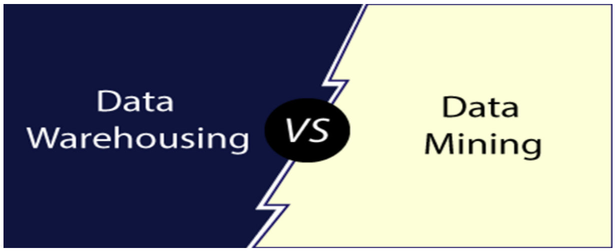
iii. Financial Market Analysis:

Data Mining techniques are widely used to help **Model Financial Market**

iv. Trend Analysis:

Analyzing the current existing trend in the marketplace is a strategic benefit because it helps in cost reduction and manufacturing process as per market demand.

Differences between Data Mining and Data Warehousing:



S. No.	Basis of Comparison	Data Warehousing	Data Mining
1.	Definition	A data warehouse is a database system that is designed for analytical analysis instead of transactional work.	Data mining is the process of analyzing data patterns.
2.	Process	Data is stored periodically.	Data is analyzed regularly.
3.	Purpose	Data warehousing is the process of extracting and storing data to allow easier reporting.	Data mining is the use of pattern recognition logic to identify patterns.
4.	Managing Authorities	Data warehousing is solely carried out by engineers.	Data mining is carried out by business users with the help of engineers.
5.	Data Handling	Data warehousing is the process of pooling all relevant data together.	Data mining is considered as a process of extracting data from large data sets.
6.	Functionality	Subject-oriented, integrated, time-varying and non-volatile constitute data warehouses.	AI, statistics, databases, and machine learning systems are all used in data mining technologies.
7.	Task	Data warehousing is the process of extracting and	Pattern recognition logic is used in data mining to find patterns.

S. No.	Basis of Comparison	Data Warehousing	Data Mining
		storing data in order to make reporting more efficient.	
8.	Uses	It extracts data and stores it in an orderly format, making reporting easier and faster.	This procedure employs pattern recognition tools to aid in the identification of access patterns.
9.	Examples	When a data warehouse is connected with operational business systems like CRM (Customer Relationship Management) systems, it adds value.	Data mining aids in the creation of suggestive patterns of key parameters. Customer purchasing behavior, items, and sales are examples. As a result, businesses will be able to make the required adjustments to their operations and productio

Data Generalization

Data Generalization is the process of summarizing data by replacing relatively low level values with higher level concepts. It is a form of descriptive data mining.

There are two basic approaches of data generalization :

1. Data cube approach :

It is also known as OLAP approach.

It is an efficient approach as it is helpful to make the past selling graph.

In this approach, computation and results are stored in the Data cube.

It uses Roll-up and Drill-down operations on a data cube.

These operations typically involve aggregate functions, such as count(), sum(), average(), and max().

These materialized views can then be used for decision support, knowledge discovery, and many other applications.

2. Attribute oriented induction :

It is an online data analysis, query oriented and generalization based approach.

In this approach, we perform generalization on basis of different values of each attributes within the relevant data set. after that same tuple are merged and their respective counts are accumulated in order to perform aggregation.

It performs off-line aggregation before an OLAP or data mining query is submitted for processing.

On the other hand, the attribute oriented induction approach, at least in its initial proposal, a relational database query – oriented, generalized based (on-line data analysis technique).

It is not limited to particular measures nor categorical data.

Attribute oriented induction approach uses two method :

(i). Attribute removal.
(ii). Attribute generalization.

Frequent Pattern Mining in Data Mining

Frequent pattern mining in data mining is the process of identifying patterns or associations within a dataset that occur frequently. This is typically done by analyzing large datasets to find items or sets of items that appear together frequently.

Frequent pattern extraction is an essential mission in data mining that intends to uncover repetitive patterns or itemsets in a granted dataset. It encompasses recognizing collections of components that occur together frequently in a transactional or relational database. This procedure can offer valuable perceptions into the connections and affiliations among diverse components or features within the data.

Here’s an elaborate explanation of repeating arrangement prospecting:

Transactional and Relational Databases:

Repeating arrangement prospecting can be applied to transactional databases, where each transaction consists of a collection of objects. For instance, in a retail dataset, each transaction may represent a customer’s purchase with objects like loaf, dairy, and ovals. It can also be used with relational databases, where data is organized into multiple related tables. In this case, repeating arrangements can represent connections among different attributes or columns.

Support and Repeating Groupings:

The support of a grouping is defined as the proportion of transactions in the database that contain that particular grouping. It represents the frequency or occurrence of the grouping in the dataset. Repeating groupings are collections of objects whose support is above a specified minimum support threshold. These groupings are considered interesting and are the primary focus of repeating arrangement prospecting.

Apriori Algorithm:

The Apriori algorithm is one of the most well-known and widely used algorithms for repeating arrangement prospecting. It uses a breadth-first search strategy to discover repeating groupings efficiently. The algorithm works in multiple iterations. It starts by finding repeating individual objects by scanning the database once and counting the occurrence of each object. It then generates candidate groupings of size 2 by combining the repeating groupings of size 1. The support of these candidate groupings is calculated by scanning the database again. The process continues iteratively, generating candidate groupings of size k and calculating their support until no more repeating groupings can be found.

Support-based Pruning:

During the Apriori algorithm’s execution, aid-based pruning is used to reduce the search space and enhance efficiency. If an itemset is found to be rare (i.e., its aid is below the minimum aid threshold), then all its supersets are also assured to be rare. Therefore, these supersets are trimmed from further consideration. This trimming step significantly decreases the number of potential item sets that need to be evaluated in subsequent iterations.

Association Rule Mining:

Frequent item sets can be further examined to discover association rules, which represent connections between different items. An association rule consists of an antecedent and a consequent (right-hand side), both of which are item sets. For instance, {milk, bread} => {eggs} is an association rule. Association rules are produced from frequent itemsets by considering different combinations of items and calculating measures such as aid, confidence, and lift. Aid measures the frequency of both the antecedent and the consequent appearing together, while confidence measures the conditional probability of the consequent given the antecedent. Lift indicates the strength of the association between the antecedent and the consequent, considering their individual aid.

Applications:

Frequent pattern mining has various practical uses in different domains. Some examples include market basket analysis, customer behavior analysis, web mining, bioinformatics, and network traffic analysis. Market basket analysis involves analyzing customer purchase patterns to identify connections between items and enhance sales strategies. In bioinformatics, frequent pattern mining can be used to identify common patterns in DNA sequences, protein structures, or gene expressions, leading to insights in genetics and drug design. Web mining can employ frequent pattern mining to discover navigational patterns, user preferences, or collaborative filtering recommendations on the web.

Regular pattern extraction is a data extraction approach employed to spot repeating forms or itemsets in transactional or relational databases. It entails locating collections of objects that occur collectively often and possesses numerous uses in different fields. The Apriori algorithm is a well-liked technique utilized to effectively detect consistent itemsets, and association rule extraction can be carried out to obtain significant connections between objects.

There are several different algorithms used for frequent pattern mining, including:

Apriori algorithm: This is one of the most commonly used algorithms for frequent pattern mining. It uses a “bottom-up” approach to identify frequent itemsets and then generates association rules from those itemsets.

ECLAT algorithm: This algorithm uses a “depth-first search” approach to identify frequent itemsets. It is particularly efficient for datasets with a large number of items.

FP-growth algorithm: This algorithm uses a “compression” technique to find frequent patterns efficiently. It is particularly efficient for datasets with a large number of transactions.

Frequent pattern mining has many applications, such as Market Basket Analysis, Recommender Systems, Fraud Detection, and many more.

Advantages:

It can find useful information which is not visible in simple data browsing

It can find interesting association and correlation among data items

Disadvantages:

It can generate a large number of patterns

With high dimensionality, the number of patterns can be very large, making it difficult to interpret the results.

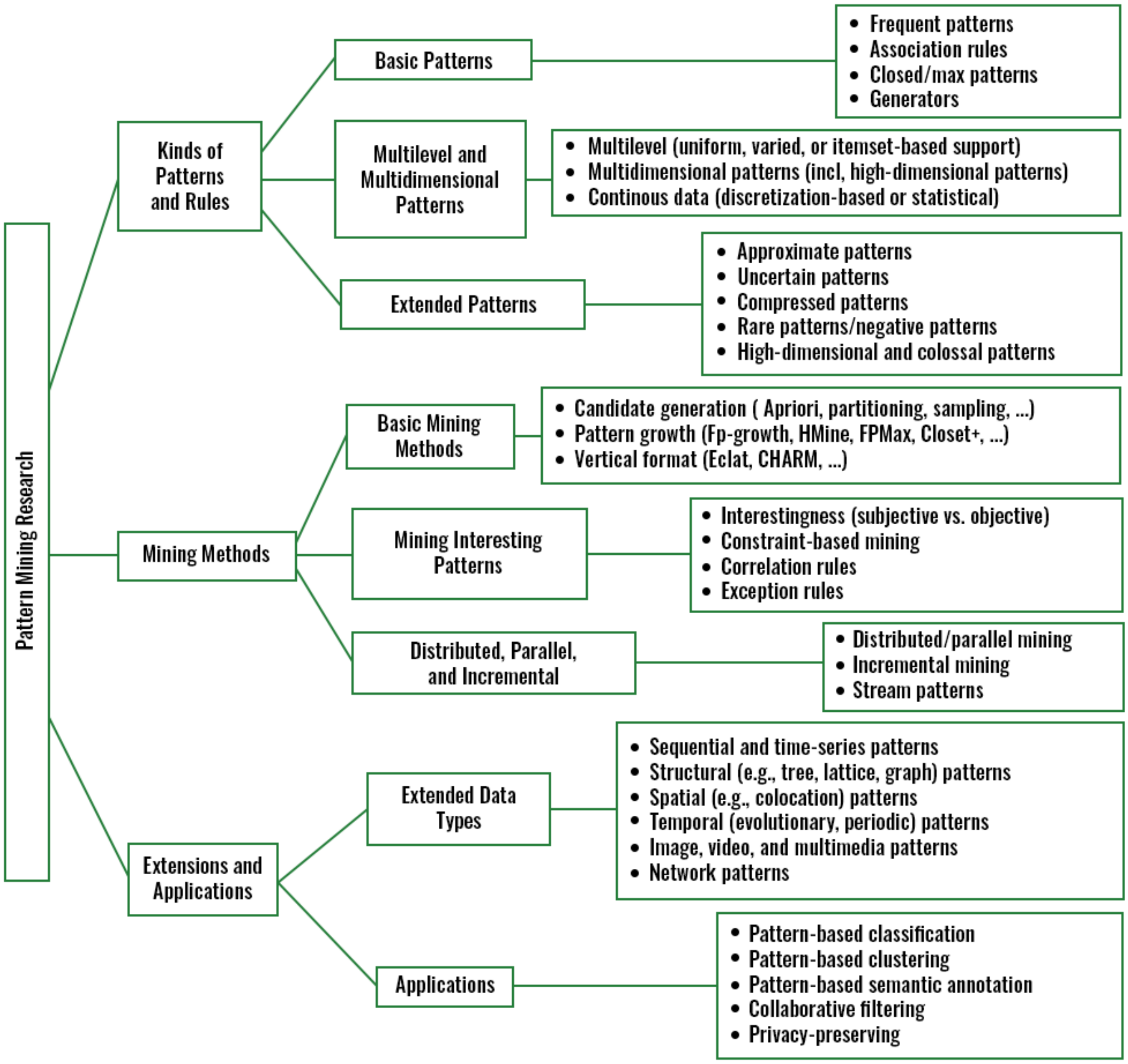
The increasing power of computer technology creates a large amount of data and storage. Databases are increasing rapidly and in this computerized world everything is shifting online and data is increasing as a new currency. Data comes in different shapes and sizes and is collected in different ways. By using data mining there are many benefits it helps us to improve the particular process and in some cases, it costs saving or revenue generation. Data mining is commonly used to search a large amount of data for patterns and trends, and not only for searching it uses the data for further processes and develops actionable processes.

Data mining is the process of converting raw data into suitable patterns based on trends.

Data mining has different types of patterns and frequent pattern mining is one of them. This concept was introduced for mining transaction databases. Frequent patterns are patterns(such as items, subsequences, or substructures) that appear frequently in the database. It is an analytical process that finds frequent patterns, associations, or causal structures from databases in various databases. This process aims to find the frequently occurring item in a transaction. By frequent patterns, we can identify strongly correlated items together and we can identify similar characteristics and associations among them. By doing frequent data mining we can go further for clustering and association.

Frequent pattern mining is a major concern it plays a major role in associations and correlations and disclose an intrinsic and important property of dataset.

Frequent data mining can be done by using association rules with particular algorithms eclat and apriori algorithms. Frequent pattern mining searches for recurring relationships in a data set. It also helps to find the inheritance regularities. to make fast processing software with a user interface and used for a long time without any error.



Pattern Mining Research
Kinds of Patterns and Rules
Basic Patterns
Frequent patterns
Association rules
Closed/max patterns
Generators
Multilevel and Multidimensional Patterns
Multilevel (uniform, varied, or itemset-based support)
Multidimensional patterns (incl, high-dimensional patterns)
Continous data (discretization-based or statistical)
Extended Patterns
Approximate patterns
Uncertain patterns
Compressed patterns
Rare patterns/negative patterns
High-dimensional and colossal patterns
Mining Methods
Basic Mining Methods
Candidate generation (Apriori, partitioning, sampling, ...)
Pattern growth (Fp-growth, HMine, FPMax, Closet+, ...)
Vertical format (Eclat, CHARM, ...)
Mining Interesting Patterns
Interestingness (subjective vs. objective)
Constraint-based mining
Correlation rules
Exception rules
Distributed, Parallel, and Incremental
Distributed/parallel mining
Incremental mining
Stream patterns
Extensions and Applications
Extended Data Types
Sequential and time-series patterns
Structural (e.g., tree, lattice, graph) patterns
Spatial (e.g., colocation) patterns
Temporal (evolutionary, periodic) patterns
Image, video, and multimedia patterns
Network patterns
Applications
Pattern-based classification
Pattern-based clustering
Pattern-based semantic annotation
Collaborative filtering
Privacy-preserving

Association Rule Mining:

It is easy to find associations in frequent patterns:

for each frequent pattern x for each subset y c x.

calculate the support of y-> x – y.

if it is greater than the threshold, keep the rule. There are two algorithms that support this lattice

Apriori algorithm

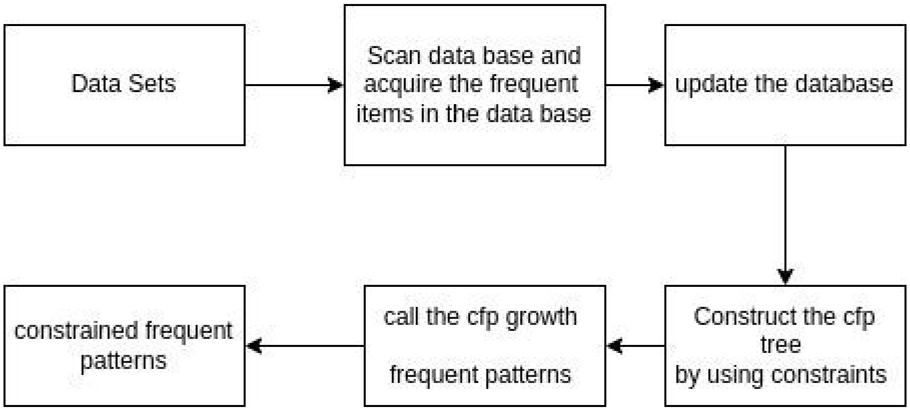
eclat algorithm

Apriori	Eclat
It performs “perfect” pruning of infrequent item sets.	It reduces memory requirements and is faster.
It requires a lot of memory(all frequent item sets are represented) and support counting takes very long for large transactions. But this is not efficient in practice.	Its storage of transaction list.

The words support and confidence support the association rule.

Support: how often a given rule in a database is mined? support the transaction contains x U y

Confidence: the number of times the given rule in a practice is true. The conditional probability is a transaction having x as well as y.


Data Sets
Scan data base and acquire the frequent items in the data base
update the database
Construct the cfp tree by using constraints
call the cfp growth frequent patterns
constrained frequent patterns

working principle (it is a simple point of scale application for any supermarket which has a good off-product scale)

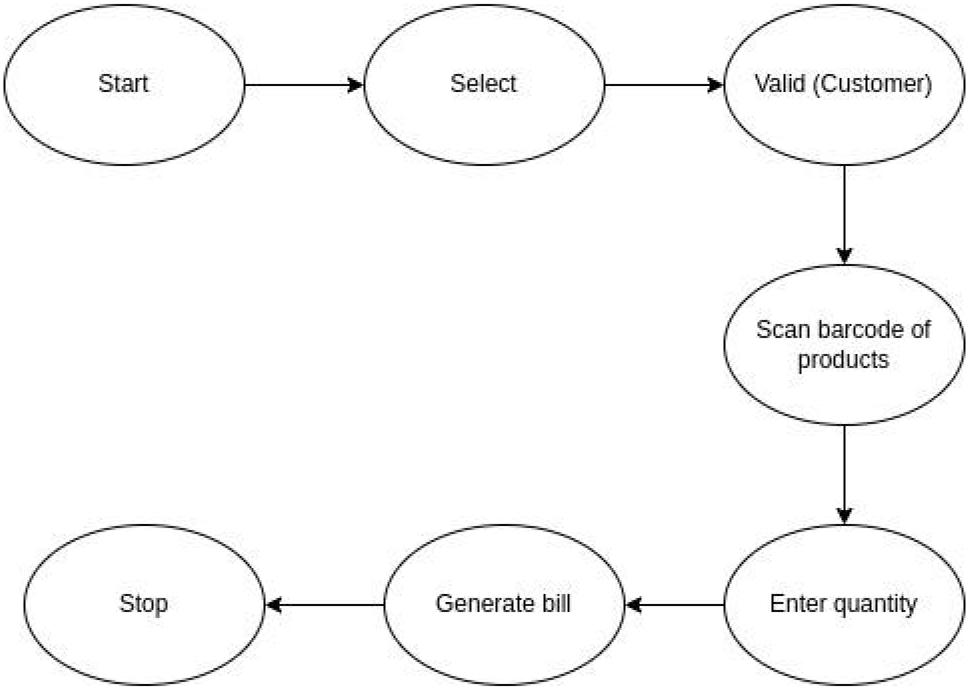
the product data will be entered into the database.

the taxes and commissions are entered.

the product will be purchased and it will be sent to the bill counter.

the bill calculating operator will check the product with the bar code machine it will check and match the product in the database and then it will show the information of the product.

the bill will be paid by the customer and he will receive the products.



Tasks in the frequent pattern mining:

Association

Cluster analysis: frequent pattern-based clustering is well suited for high-dimensional data. by the extension of dimension the sub-space clustering occurs.

Data warehouse: iceberg cube and cube gradient

Broad applications

There are some to improve the efficiency of the tasks.

Closed Pattern:

A frequent pattern, it meets the minimum support criteria. All super patterns of a closed pattern are less frequent than the closed pattern.

Max Pattern:

It also meets the minimum support criteria(like a closed pattern). All super patterns of a max pattern are not frequent patterns. both patterns generate fewer numbers of patterns so therefore they increase the efficiency of the task.

Applications of Frequent Pattern Mining:

basket data analysis, cross-marketing, catalog design, sale campaign analysis, web log analysis, and DNA sequence analysis.

Issues of frequent pattern mining

flexibility and reusability for creating frequent patterns

most of the algorithms used for mining frequent item sets do not offer flexibility for reusing

much research is needed to reduce the size of the derived patterns

Frequent pattern mining has several applications in different areas, including:

Market Basket Analysis: This is the process of analyzing customer purchasing patterns in order to identify items that are frequently bought together. This information can be used to optimize product placement, create targeted marketing campaigns, and make other business decisions.

Recommender Systems: Frequent pattern mining can be used to identify patterns in user behavior and preferences in order to make personalized recommendations.

Fraud Detection: Frequent pattern mining can be used to identify abnormal patterns of behavior that may indicate fraudulent activity.

Network Intrusion Detection: Network administrators can use frequent pattern mining to detect patterns of network activity that may indicate a security threat.

Medical Analysis: Frequent pattern mining can be used to identify patterns in medical data that may indicate a particular disease or condition.

Text Mining: Frequent pattern mining can be used to identify patterns in text data, such as keywords or phrases that appear frequently together in a document.

Web usage mining: Frequent pattern mining can be used to analyze patterns of user behavior on a website, such as which pages are visited most frequently or which links are clicked on most often.

Gene Expression: Frequent pattern mining can be used to analyze patterns of gene expression in order to identify potential biomarkers for different diseases.

These are a few examples of the application of frequent pattern mining. The list is not exhaustive and the technique can be applied in many other areas, as well.

Conclusion:

It is impossible to give complete coverage of this topic with the limited space and our limited knowledge. Frequent pattern mining has achieved tremendous progress and claimed a good set of applications. However in-depth research is required that the field may have a long-lasting and deep impact on data mining applications.

Correlation Analysis

Correlation analysis is a statistical technique for determining the strength of a link between two variables. It is used to detect patterns and trends in data and to forecast future occurrences.

Consider a problem with different factors to be considered for making optimal conclusions

Correlation explains how these variables are dependent on each other.

Correlation quantifies how strong the relationship between two variables is. A higher value of the correlation coefficient implies a stronger association.

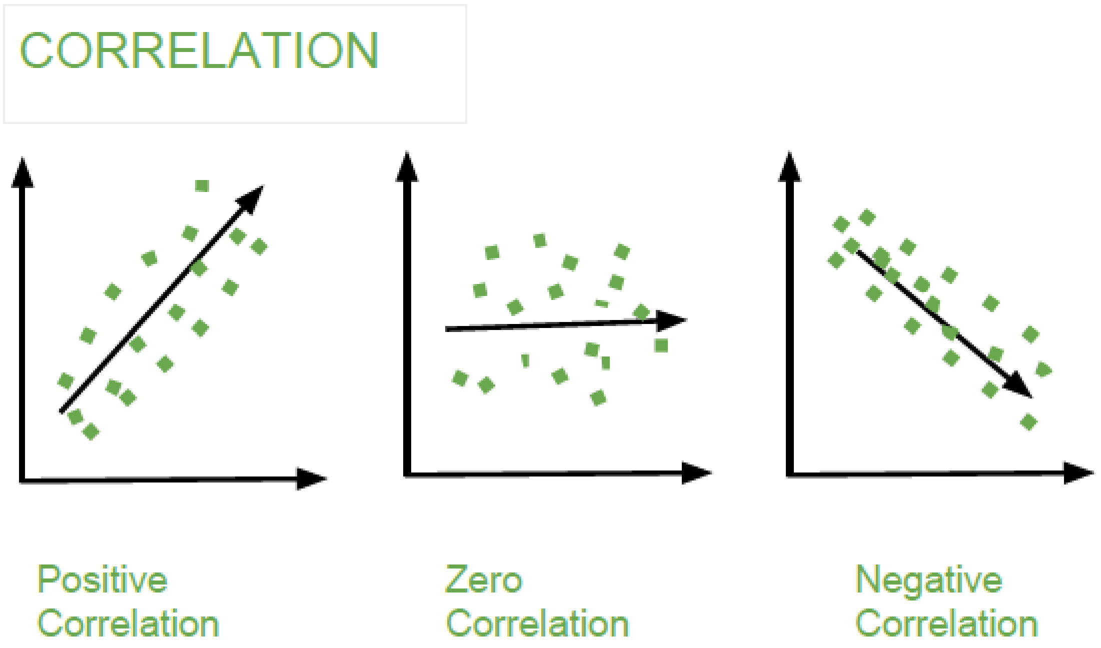
The sign of the correlation coefficient indicates the direction of the relationship between variables. It can be either positive, negative, or zero.

What is Correlation?

The Pearson correlation coefficient is the most often used metric of correlation. It expresses the linear relationship between two variables in numerical terms. The Pearson correlation coefficient, written as "r," is as follows:

Types of Correlation

There are three types of correlation:



Correlation

Positive Correlation: Positive correlation indicates that two variables have a direct relationship. As one variable increases, the other variable also increases. For example, there is a positive correlation between height and weight. As people get taller, they also tend to weigh more.

Negative Correlation: Negative correlation indicates that two variables have an inverse relationship. As one variable increases, the other variable decreases. For example, there is a negative correlation between price and demand. As the price of a product increases, the demand for that product decreases.

Zero Correlation: Zero correlation indicates that there is no relationship between two variables. The changes in one variable do not affect the other variable. For example, there is zero correlation between shoe size and intelligence.

A positive correlation indicates that the two variables move in the same direction, while a negative correlation indicates that the two variables move in opposite directions.

The strength of the correlation is measured by a correlation coefficient, which can range from -1 to 1. A correlation coefficient of 0 indicates no correlation, while a correlation coefficient of 1 or -1 indicates a perfect correlation.

Frequent Item set in Data set (Association Rule Mining)

Read

Courses

Jobs

INTRODUCTION:

Frequent item sets, also known as association rules, are a fundamental concept in association rule mining, which is a technique used in data mining to discover relationships between items in a dataset. The goal of association rule mining is to identify relationships between items in a dataset that occur frequently together.

A frequent item set is a set of items that occur together frequently in a dataset. The frequency of an item set is measured by the support count, which is the number of transactions or records in the dataset that contain the item set. For example, if a dataset contains 100 transactions and the item set {milk, bread} appears in 20 of those transactions, the support count for {milk, bread} is 20.

Association rule mining algorithms, such as Apriori or FP-Growth, are used to find frequent item sets and generate association rules. These algorithms work by iteratively generating candidate item sets and pruning those that do not meet the minimum support threshold. Once the frequent item sets are found, association rules can be generated by using the concept of confidence, which is the ratio of the number of transactions that contain the item set and the number of transactions that contain the antecedent (left-hand side) of the rule.

Frequent item sets and association rules can be used for a variety of tasks such as market basket analysis, cross-selling and recommendation systems. However, it should be noted that association rule mining can generate a large number of rules, many of which may be irrelevant or uninteresting. Therefore, it is important to use appropriate measures such as lift and conviction to evaluate the interestingness of the generated rules.

Association Mining searches for frequent items in the data set. In frequent mining usually, interesting associations and correlations between item sets in transactional and relational databases are found. In short, Frequent Mining shows which items appear together in a transaction or relationship.

Need of Association Mining: Frequent mining is the generation of association rules from a Transactional Dataset. If there are 2 items X and Y purchased frequently then it's good to put them together in stores or provide some discount offer on one item on purchase of another item. This can really increase sales. For example, it is likely to find that if a customer buys Milk and bread he/she also buys Butter. So the association rule is ['milk]^['bread']=>['butter']. So the seller can suggest the customer buy butter if he/she buys Milk and Bread.

Important Definitions :

Support : It is one of the measures of interestingness. This tells about the usefulness and certainty of rules. 5% Support means total 5% of transactions in the database follow the rule.

Support(A -> B) = Support_count(A ∪ B)

Confidence: A confidence of 60% means that 60% of the customers who purchased a milk and bread also bought butter.

Confidence(A -> B) = Support_count(A ∪ B) / Support_count(A)

If a rule satisfies both minimum support and minimum confidence, it is a strong rule.

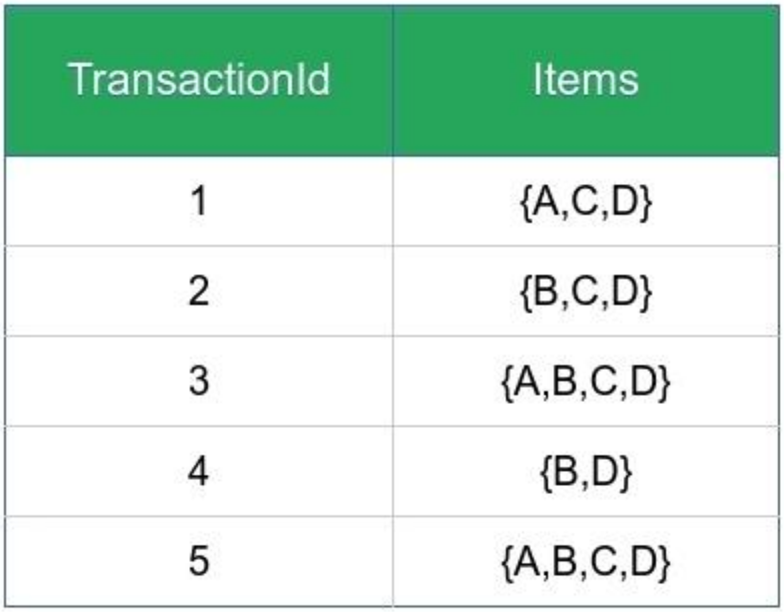
Support_count(X): Number of transactions in which X appears. If X is A union B then it is the number of transactions in which A and B both are present.

Maximal Itemset: An itemset is maximal frequent if none of its supersets are frequent.

Closed Itemset: An itemset is closed if none of its immediate supersets have same support count same as Itemset.

K- Itemset: Itemset which contains K items is a K-itemset. So it can be said that an itemset is frequent if the corresponding support count is greater than the minimum support count.

Example On finding Frequent Itemsets – Consider the given dataset with given transactions.



TransactionId	Items
1	{A,C,D}
2	{B,C,D}
3	{A,B,C,D}
4	{B,D}
5	{A,B,C,D}

Lets say minimum support count is 3

Relation hold is maximal frequent => closed => frequent

1-frequent: {A} = 3; // not closed due to {A, C} and not maximal {B} = 4; // not closed due to {B, D} and no maximal {C} = 4; // not closed due to {C, D} not maximal {D} = 5; // closed item-set since not immediate super-set has same count. Not maximal

2-frequent: {A, B} = 2 // not frequent because support count < minimum support count so ignore {A, C} = 3 // not closed due to {A, C, D} {A, D} = 3 // not closed due to {A, C, D} {B, C} = 3 // not closed due to {B, C, D} {B, D} = 4 // closed but not maximal due to {B, C, D} {C, D} = 4 // closed but not maximal due to {B, C, D}

3-frequent: {A, B, C} = 2 // ignore not frequent because support count < minimum support count {A, B, D} = 2 // ignore not frequent because support count < minimum support count {A, C, D} = 3 // maximal frequent {B, C, D} = 3 // maximal frequent

4-frequent: {A, B, C, D} = 2 //ignore not frequent </

ADVANTAGES OR DISADVANTAGES:

Advantages of using frequent item sets and association rule mining include:

Efficient discovery of patterns: Association rule mining algorithms are efficient at discovering patterns in large datasets, making them useful for tasks such as market basket analysis and recommendation systems.

Easy to interpret: The results of association rule mining are easy to understand and interpret, making it possible to explain the patterns found in the data.

Can be used in a wide range of applications: Association rule mining can be used in a wide range of applications such as retail, finance, and healthcare, which can help to improve decision-making and increase revenue.

Handling large datasets: These algorithms can handle large datasets with many items and transactions, which makes them suitable for big-data scenarios.

Disadvantages of using frequent item sets and association rule mining include:

Large number of generated rules: Association rule mining can generate a large number of rules, many of which may be irrelevant or uninteresting, which can make it difficult to identify the most important patterns.

Limited in detecting complex relationships: Association rule mining is limited in its ability to detect complex relationships between items, and it only considers the co-occurrence of items in the same transaction.

Can be computationally expensive: As the number of items and transactions increases, the number of candidate item sets also increases, which can make the algorithm computationally expensive.

Need to define the minimum support and confidence threshold: The minimum support and confidence threshold must be set before the association rule mining process, which can be difficult and requires a good understanding of the data.

Mining Various Kinds of Association Rules

1. Mining Multilevel Association Rules

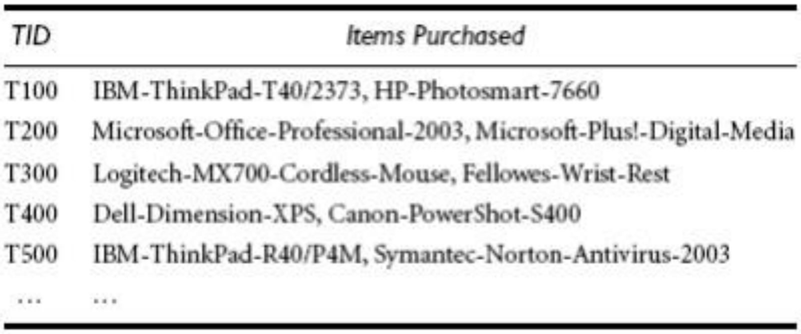
For many applications, it is difficult to find strong associations among data items at low or primitive levels of abstraction due to the sparsity of data at those levels. Strong associations discovered at high levels of abstraction may represent commonsense knowledge.

. Therefore, data mining systems should provide capabilities for mining association rules at multiple levels of abstraction, with sufficient flexibility for easy traversal among different abstraction spaces.

Let’s examine the following example.

Mining multilevel association rules. Suppose we are given the task-relevant set of transactional data in Table for sales in an *AllElectronics* store, showing the items purchased for each transaction.

The concept hierarchy for the items is shown in Figure . A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher level, more general concepts. Data can be generalized by replacing low-level concepts within the data by their higher-level concepts, or *ancestors*, from a concept hierarchy.



TID	*Items Purchased*
T100	IBM-ThinkPad-T40/2373, HP-Photosmart-7660
T200	Microsoft-Office-Professional-2003, Microsoft-Plus!-Digital-Media
T300	Logitech-MX700-Cordless-Mouse, Fellowes-Wrist-Rest
T400	Dell-Dimension-XPS, Canon-PowerShot-S400
T500	IBM-ThinkPad-R40/P4M, Symantec-Norton-Antivirus-2003
...	...

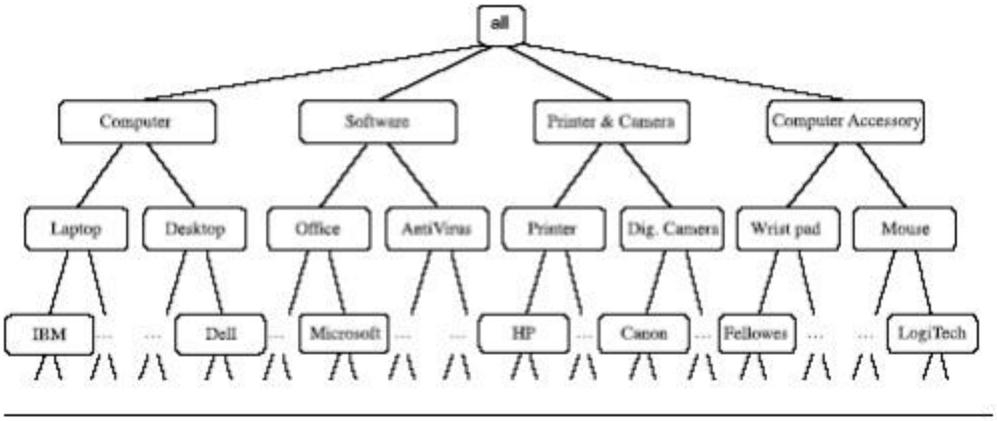


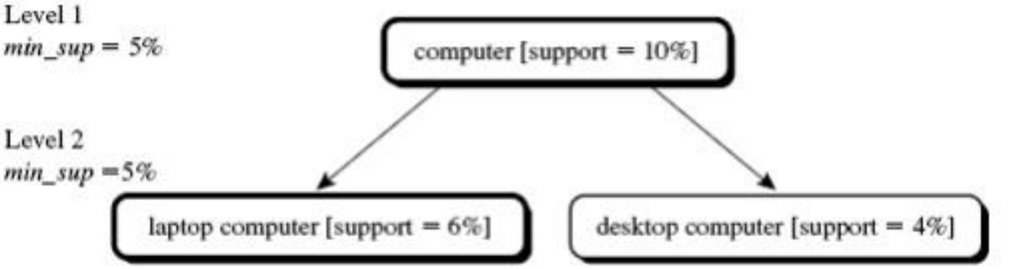
Figure A concept hierarchy for *AllElectronics* computer items.

Association rules generated from mining data at multiple levels of abstraction are called multiple-level or multilevel association rules. Multilevel association rules can be mined efficiently using concept hierarchies under a support-confidence framework. In general, a top-down strategy is employed, For each level, any algorithm for discovering frequent itemsets may be used, such as Apriori or its variations.

Using uniform minimum support for all levels (referred to as uniform support): The same minimum support threshold is used when mining at each level of abstraction. For example, in Figure 5.11, a minimum support threshold of 5% is used throughout (e.g., for mining from

"computer" down to *"laptop computer"*). Both *"computer"* and *"laptop computer"* are found to be frequent, while *"desktop computer"* is not.

When a uniform minimum support threshold is used, the search procedure is simplified. The method is also simple in that users are required to specify only one minimum support threshold. An Apriori-like optimization technique can be adopted, based on the knowledge that an ancestor is a superset of its descendants: The search avoids examining itemsets containing any item whose ancestors do not have minimum support.



Multilevel mining with uniform support.

Using reduced minimum support at lower levels (referred to as reduced support): Each level of abstraction has its own minimum support threshold. The deeper the level of abstraction, the smaller the corresponding threshold is. For example, in Figure, the minimum support thresholds for levels 1 and 2 are 5% and 3%, respectively. In this way, *"computer," "laptop computer,"* and *"desktop computer"* are all considered frequent.

□

Using item or group-based minimum support (referred to as group-based support):

□

Because users or experts often have insight as to which groups are more important than others, it is sometimes more desirable to set up user-specific, item, or group based minimal support thresholds when mining multilevel rules. For example, a user could set up the minimum support thresholds based on product price, or on items of interest, such as by setting particularly low support thresholds for *laptop computers* and *flash drives* in order to pay particular attention to the association patterns containing items in these categories.

2. Mining Multidimensional Association Rules from Relational Databases and Data Warehouses

We have studied association rules that imply a single predicate, that is, the predicate *buys*. For instance, in mining our *AllElectronics* database, we may discover the Boolean association rule

$$buys(X, \text{“}digital\ camera\text{”}) \Rightarrow buys(X, \text{“}HP\ printer\text{”}).$$

Following the terminology used in multidimensional databases, we refer to each distinct predicate in a rule as a dimension. Hence, we can refer to Rule above as a single dimensional or intra dimensional association rule because it contains a single distinct predicate (e.g., *buys*)with multiple occurrences (i.e., the predicate occurs more than once within the rule). As we have seen in the previous sections of this chapter, such rules are commonly mined from transactional data.

Considering each database attribute or warehouse dimension as a predicate, we can therefore mine association rules containing *multiple* predicates, such as

$$age(X, \text{“20...29”}) \wedge occupation(X, \text{“student”}) \Rightarrow buys(X, \text{“laptop”}).$$

Association rules that involve two or more dimensions or predicates can be referred to as multidimensional association rules. Rule above contains three predicates (*age, occupation,* and *buys*), each of which occurs *only once* in the rule. Hence, we say that it has no repeated predicates. Multidimensional association rules with no repeated predicates are called inter dimensional association rules. We can also mine multidimensional association rules with repeated predicates, which contain multiple occurrences of some predicates. These rules are called hybrid-dimensional association rules. An example of such a rule is the following, where the predicate *buys* is repeated:

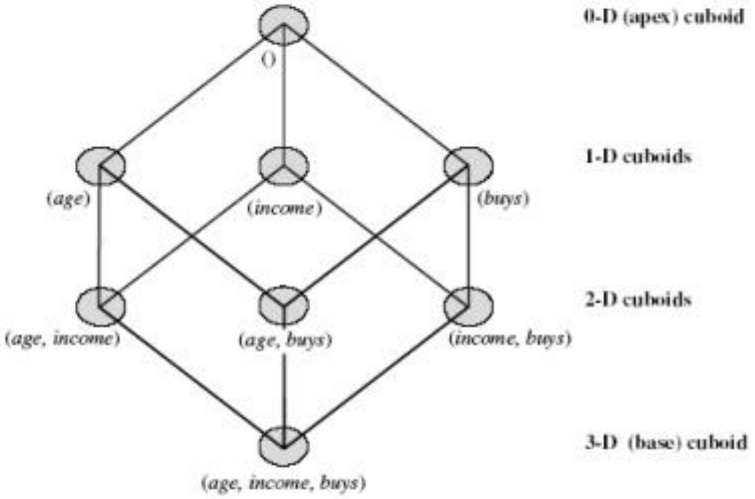
$$age(X, \text{“20...29”}) \wedge buys(X, \text{“laptop”}) \Rightarrow buys(X, \text{“HP printer”})$$

Note that database attributes can be categorical or quantitative. Categorical attributes have a finite number of possible values, with no ordering among the values (e.g., *occupation, brand, color*). Categorical attributes are also called nominal attributes, because their values are ―names of things.‖ Quantitative attributes are numeric and have an implicit ordering among values (e.g., *age, income, price*). Techniques for mining multidimensional association rules can be categorized into two basic approaches regarding the treatment of quantitative attributes.

Mining Multidimensional Association Rules Using Static Discretization of Quantitative

Attributes

Quantitative attributes, in this case, are discretized before mining using predefined concept hierarchies or data discretization techniques, where numeric values are replaced by interval labels. Categorical attributes may also be generalized to higher conceptual levels if desired. If the resulting task-relevant data are stored in a relational table, then any of the frequent itemset mining algorithms we have discussed can be modified easily so as to find all frequent predicate sets rather than frequent itemsets. In particular, instead of searching on only one attribute like *buys*, we need to search through all of the relevant attributes, treating each attribute-value pair as an itemset.



Lattice of cuboids, making up a 3-D data cube. Each cuboid represents a different group-by. The base cuboid contains the three predicates *age*, *income*, and *buys*.

Mining Quantitative Association Rules

Mining Quantitative Association Rules

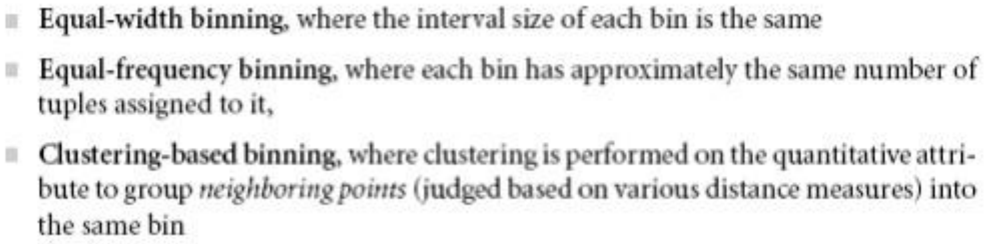
Quantitative association rules are multidimensional association rules in which the numeric attributes are *dynamically* discretized during the mining process so as to satisfy some mining criteria, such as maximizing the confidence or compactness of the rules mined. In this section, we focus specifically on how to mine quantitative association rules having two quantitative attributes on the left-hand side of the rule and one categorical attribute on the right-hand side of the rule. That is,

$$A_{quan1} \wedge A_{quan2} \Rightarrow A_{cat}$$

where *Aquan*1 and *Aquan*2 are tests on quantitative attribute intervals (where the intervals are dynamically determined), and *Acat* tests a categorical attribute from the task-relevant data. Such rules have been referred to as two-dimensional quantitative association rules, because they contain two quantitative dimensions. For instance, suppose you are curious about the association relationship between pairs of quantitative attributes, like customer age and income, and the type of television (such as *high-definition TV,* i.e., *HDTV*) that customers like to buy. An example of such a 2-D quantitative association rule is

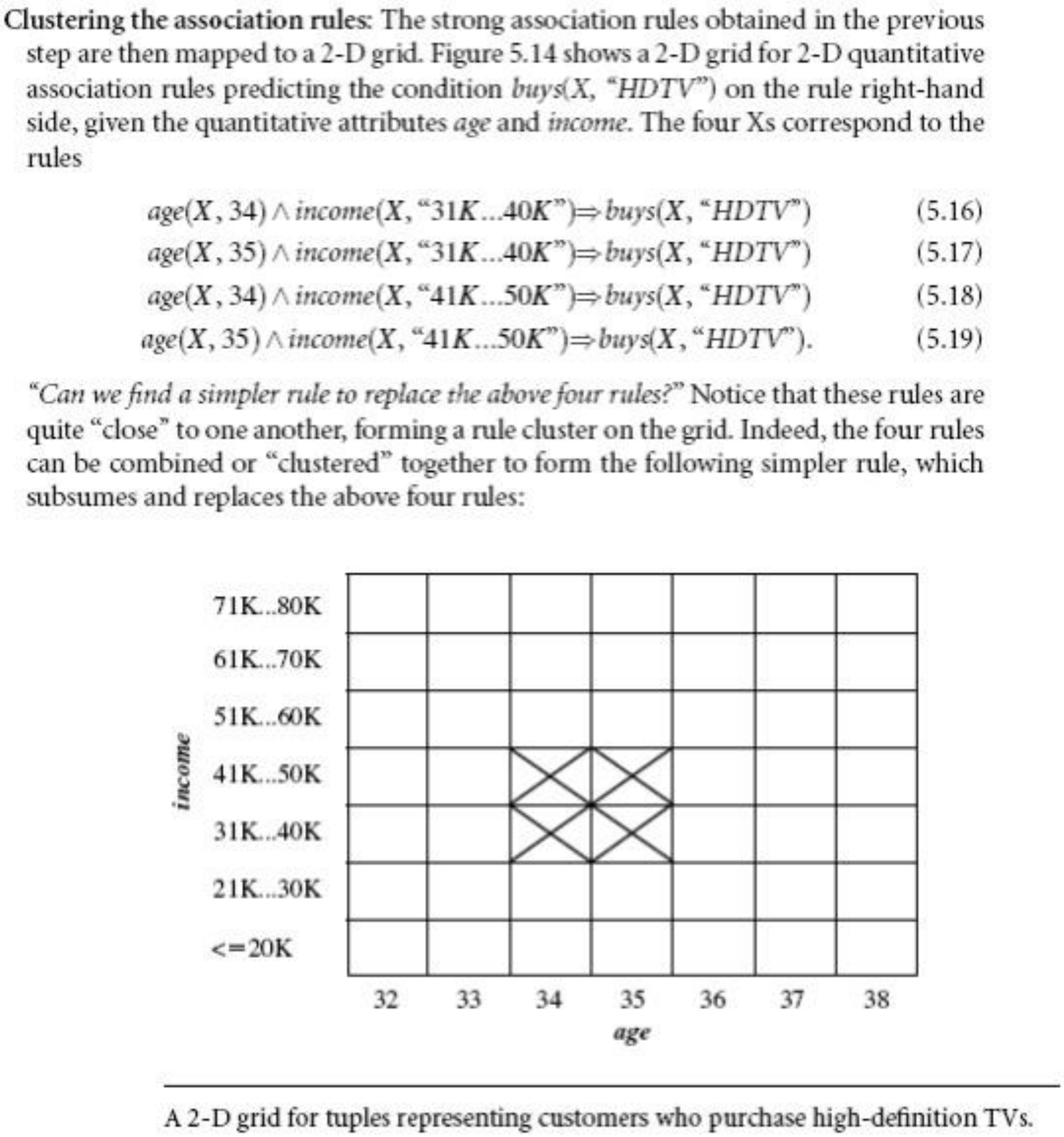
$$age(X, \text{“30...39”}) \wedge income(X, \text{“42}K\text{...48}K\text{”}) \Rightarrow buys(X, \text{“}HDTV\text{”})$$

Binning: Quantitative attributes can have a very wide range of values defining their domain. Just think about how big a 2-D grid would be if we plotted *age* and *income* as axes, where each possible value of *age* was assigned a unique position on one axis, and similarly, each possible value of *income* was assigned a unique position on the other axis! To keep grids down to a manageable size, we instead partition the ranges of quantitative attributes into intervals. These intervals are dynamic in that they may later be further combined during the mining process. The partitioning process is referred to as binning, that is, where the intervals are considered ―bins.‖ Three common binning strategies area as follows:



- **Equal-width binning,** where the interval size of each bin is the same
- **Equal-frequency binning,** where each bin has approximately the same number of tuples assigned to it,
- **Clustering-based binning,** where clustering is performed on the quantitative attribute to group *neighboring points* (judged based on various distance measures) into the same bin

Finding frequent predicate sets: Once the 2-D array containing the count distribution for each category is set up, it can be scanned to find the frequent predicate sets (those satisfying minimum support) that also satisfy minimum confidence. Strong association rules can then be generated from these predicate sets, using a rule generation algorithm.



Clustering the association rules: The strong association rules obtained in the previous step are then mapped to a 2-D grid. Figure 5.14 shows a 2-D grid for 2-D quantitative association rules predicting the condition *buys*(*X*, "*HDTV*") on the rule right-hand side, given the quantitative attributes *age* and *income*. The four Xs correspond to the rules

$$age(X, 34) \wedge income(X, \text{“}31K...40K\text{”}) \Rightarrow buys(X, \text{“}HDTV\text{”}) \quad (5.16)$$

$$age(X, 35) \wedge income(X, \text{“}31K...40K\text{”}) \Rightarrow buys(X, \text{“}HDTV\text{”}) \quad (5.17)$$

$$age(X, 34) \wedge income(X, \text{“}41K...50K\text{”}) \Rightarrow buys(X, \text{“}HDTV\text{”}) \quad (5.18)$$

$$age(X, 35) \wedge income(X, \text{“}41K...50K\text{”}) \Rightarrow buys(X, \text{“}HDTV\text{”}). \quad (5.19)$$

"Can we find a simpler rule to replace the above four rules?" Notice that these rules are quite "close" to one another, forming a rule cluster on the grid. Indeed, the four rules can be combined or "clustered" together to form the following simpler rule, which subsumes and replaces the above four rules:

A 2-D grid for tuples representing customers who purchase high-definition TVs.

Correlation Analysis in Data Mining

Correlation analysis is a statistical method used to measure the strength of the linear relationship between two variables and compute their association. Correlation analysis calculates the level of change in one variable due to the change in the other. A high correlation points to a strong relationship between the two variables, while a low correlation means that the variables are weakly related.

Researchers use correlation analysis to analyze quantitative data collected through research methods like surveys and live polls for market research. They try to identify relationships, patterns, significant connections, and trends between two variables or datasets. There is a positive correlation between two variables when an increase in one variable leads to an increase in the other. On the other hand, a negative correlation means that when one variable increases, the other decreases and vice-versa.

Correlation is a bivariate analysis that measures the strength of association between two variables and the direction of the relationship. In terms of the strength of the relationship, the correlation coefficient's value varies between +1 and -1. A value of ± 1 indicates a perfect degree of association between the two variables.

As the correlation coefficient value goes towards 0, the relationship between the two variables will be weaker. The coefficient sign indicates the direction of the relationship; a + sign indicates a positive relationship, and a - sign indicates a negative relationship.

PauseNext

Unmute

Current TimeÂ 0:30

/

DurationÂ 18:10

Loaded: 8.44%

Â

Fullscreen

Why Correlation Analysis is Important

Correlation analysis can reveal meaningful relationships between different metrics or groups of metrics. Information about those connections can provide new insights and reveal interdependencies, even if the metrics come from different parts of the business.

Suppose there is a strong correlation between two variables or metrics, and one of them is being observed acting in a particular way. In that case, you can conclude that the other one is also being affected similarly. This helps group related metrics together to reduce the need for individual data processing.

Types of Correlation Analysis in Data Mining

Usually, in statistics, we measure four types of correlations: Pearson correlation, Kendall rank correlation, Spearman correlation, and the Point-Biserial correlation.

1. Pearson r correlation

Pearson r correlation is the most widely used correlation statistic to measure the degree of the relationship between linearly related variables. For example, in the stock market, if we want to measure how two stocks are related to each other, Pearson r correlation is used to measure the degree of relationship between the two. The point-biserial correlation is conducted with the Pearson correlation formula, except that one of the variables is dichotomous. The following formula is used to calculate the Pearson r correlation:

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\ \sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

r_{xy}= Pearson r correlation coefficient between x and y

n= number of observations

x_i = value of x (for ith observation)

y_i= value of y (for ith observation)

2. Kendall rank correlation

Kendall rank correlation is a non-parametric test that measures the strength of dependence between two variables. Considering two samples, a and b, where each sample size is n, we know that the total number of pairings with a b is n(n-1)/2. The following formula is used to calculate the value of Kendall rank correlation:

$$\tau = \frac{n_C - n_d}{\frac{1}{2}n(n-1)}$$

Nc= number of concordant

Nd= Number of discordant

3. Spearman rank correlation

Spearman rank correlation is a non-parametric test that is used to measure the degree of association between two variables. The Spearman rank correlation test does not carry any assumptions about the data distribution. It is the appropriate correlation analysis when the variables are measured on an at least ordinal scale.

This coefficient requires a table of data that displays the raw data, its ranks, and the difference between the two ranks. This squared difference between the two ranks will be shown on a scatter graph, which will indicate whether there is a positive, negative, or no correlation between the two variables. The constraint that this coefficient works under is $-1 \leq r \leq +1$, where a result of 0 would mean that there was no relation between the data whatsoever. The following formula is used to calculate the Spearman rank correlation:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2 - 1)}$$

ρ= Spearman rank correlation

di= the difference between the ranks of corresponding variables

n= number of observations

When to Use These Methods

The two methods outlined above will be used according to whether there are parameters associated with the data gathered. The two terms to watch out for are:

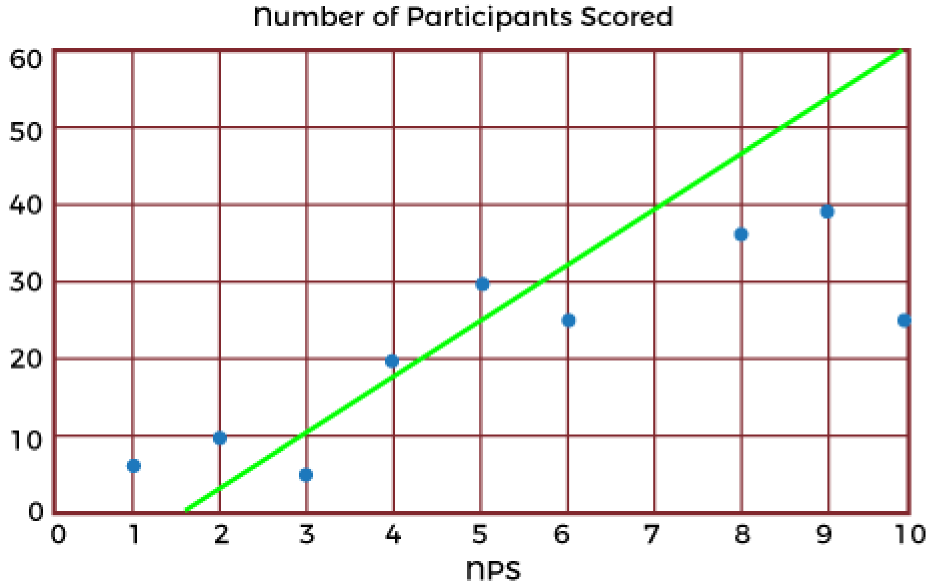
- **Parametric:***(Pearson's Coefficient)* The data must be handled with the parameters of populations or probability distributions. Typically used with quantitative data already set out within said parameters.
- **Non-parametric:***(Spearman's Rank)* Where no assumptions can be made about the probability distribution. Typically used with qualitative data, but can be used with quantitative data if Spearman's Rank proves inadequate.

In cases when both are applicable, statisticians recommend using the parametric methods such as Pearson's Coefficient because they tend to be more precise. But that doesn't mean discounting the non-parametric methods if there isn't enough data or a more specified accurate result is needed.

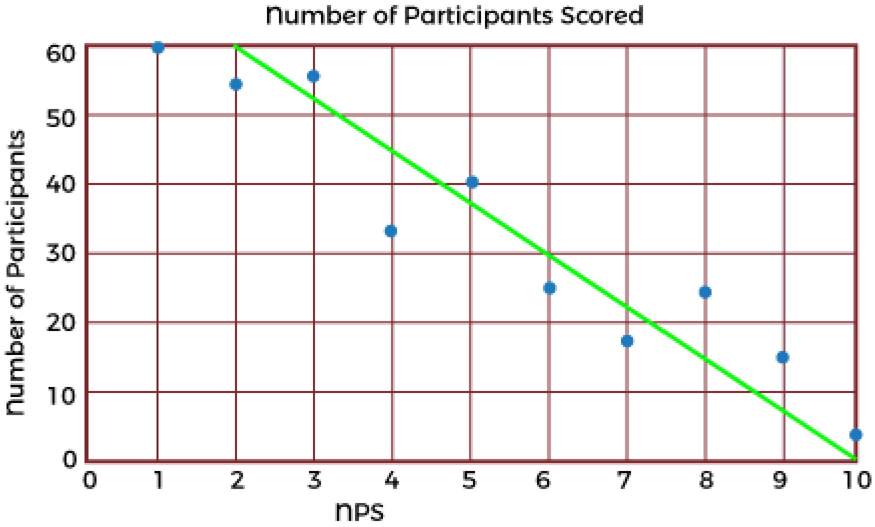
Interpreting Results

Typically, the best way to gain a generalized but more immediate interpretation of the results of a set of data is to visualize it on a scatter graph such as these:

1. **Positive Correlation:** Any score from +0.5 to +1 indicates a very strong positive correlation, which means that they both increase simultaneously. This case follows the data points upwards to indicate the positive correlation. The line of best fit, or the trend line, places to best represent the graph's data.

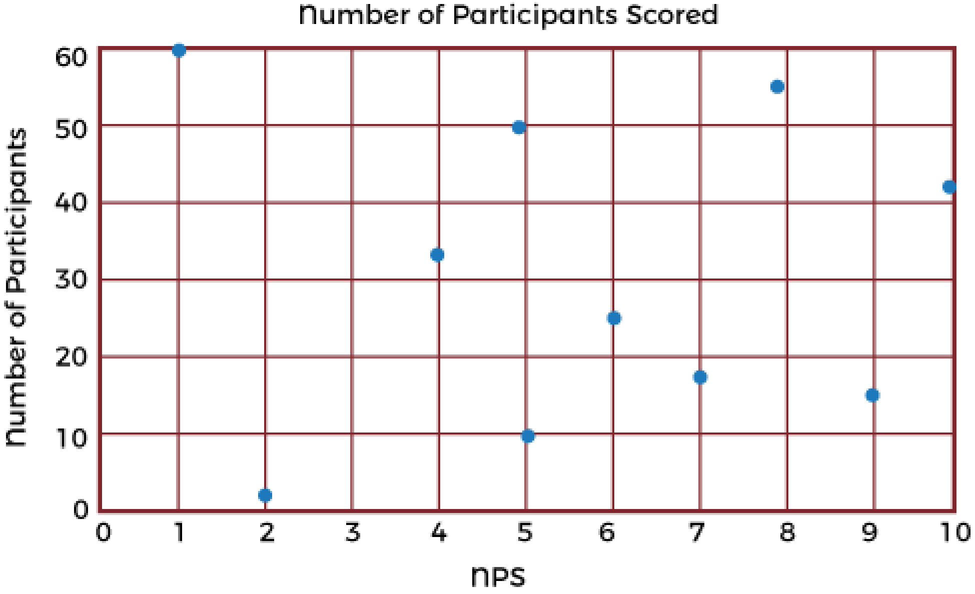


2. **Negative Correlation:** Any score from -0.5 to -1 indicates a strong negative correlation, which means that as one variable increases, the other decreases proportionally. The line of best fit can be seen here to indicate the negative correlation. In these cases, it will slope downwards from the point of origin.



3. **No Correlation:** Very simply, a score of 0 indicates no correlation, or relationship, between the two variables. This fact will stand true for all, no matter which formula is used. The more data inputted into the formula, the more accurate the result will be. The larger the sample size, the

more accurate the result.



Outliers or anomalies must be accounted for in both correlation coefficients. Using a scatter graph is the easiest way of identifying any anomalies that may have occurred. Running the correlation analysis twice (with and without anomalies) is a great way to assess the strength of the influence of the anomalies on the analysis. Spearman's Rank coefficient may be used if anomalies are present instead of Pearson's Coefficient, as this formula is extremely robust against anomalies due to the ranking system used.

Benefits of Correlation Analysis

Here are the following benefits of correlation analysis, such as:

1. Reduce Time to Detection

In anomaly detection, working with many metrics and surfacing correlated anomalous metrics helps draw relationships that reduce time to detection (TTD) and support shortened time to remediation (TTR). As data-driven decision-making has become the norm, early and robust detection of anomalies is critical in every industry domain, as delayed detection adversely impacts customer experience and revenue.

2. Reduce Alert Fatigue

Another important benefit of correlation analysis in anomaly detection is reducing alert fatigue by filtering irrelevant anomalies (based on the correlation) and grouping correlated anomalies into a single alert. Alert storms and false positives are significant challenges organizations face - getting hundreds, even thousands of separate alerts from multiple systems when many of them stem from the same incident.

3. Reduce Costs

Correlation analysis helps significantly reduce the costs associated with the time spent investigating meaningless or duplicative alerts. In addition, the time saved can be spent on more strategic initiatives that add value to the organization.

Example Use Cases for Correlation Analysis

Marketing professionals use correlation analysis to evaluate the efficiency of a campaign by monitoring and testing customers' reactions to different marketing tactics. In this way, they can better understand and serve their customers.

Financial planners assess the correlation of an individual stock to an index such as the S&P 500 to determine if adding the stock to an investment portfolio might increase the portfolio's systematic risk.

For data scientists and those tasked with monitoring data, correlation analysis is incredibly valuable for root cause analysis and reduces time to detection (TTD) and remediation (TTR). Two unusual events or anomalies happening simultaneously/rate can help pinpoint an underlying cause of a problem. The organization will incur a lower cost of experiencing a problem if it can be understood and fixed sooner.

Technical support teams can reduce the number of alerts they must respond to by filtering irrelevant anomalies and grouping correlated anomalies into a single alert. Tools such as Security Information and Event Management (SIEM) systems automatically facilitate incident response.

Does Correlation Imply Causation?

While correlation analysis techniques may identify a significant relationship, correlation does not imply causation. The analysis cannot determine the cause, nor should this conclusion be attempted. The significant relationship implies more understanding and extraneous or underlying factors that should be explored further to search for a cause. While a causal relationship may exist, any researcher would be remiss in using the correlation results to prove this existence.

The cause of any relationship discovered through the correlation analysis is for the researcher to determine through other means of statistical analysis, such as the coefficient of determination analysis. However, correlation analysis can provide a great amount of value; for example, the value of the dependency or the variables can be estimated, which can help firms estimate the cost and sale of a product or service.

In essence, the uses for and applications of correlation-based statistical analyses allow researchers to identify which aspects and variables are dependent on each other, which can generate actionable insights as they are or starting points for further investigations and deeper insights.

What is Constraint-Based Association Mining?

Data MiningDatabaseData Structure

A data mining procedure can uncover thousands of rules from a given set of information, most of which end up being independent or tedious to the users. Users have a best sense of which “direction” of mining can lead to interesting patterns and the “form” of the patterns or rules they can like to discover.

Therefore, a good heuristic is to have the users defines such intuition or expectations as constraints to constraint the search space. This strategy is called constraint-based mining.

Constraint-based algorithms need constraints to decrease the search area in the frequent itemset generation step (the association rule generating step is exact to that of exhaustive algorithms).

The general constraint is the support minimum threshold. If a constraint is uncontrolled, its inclusion in the mining phase can support significant reduction of the exploration space because of the definition of a boundary inside the search space lattice, following which exploration is not needed.

The important of constraints is well-defined − they create only association rules that are appealing to users. The method is quite trivial and the rules space is decreased whereby remaining methods satisfy the constraints.

Constraint-based clustering discover clusters that satisfy user-defined preferences or constraints. It depends on the characteristics of the constraints, constraint-based clustering can adopt rather than different approaches.

The constraints can include the following which are as follows −

Knowledge type constraints − These define the type of knowledge to be mined, including association or correlation.

Data constraints − These define the set of task-relevant information such as Dimension/level constraints − These defines the desired dimensions (or attributes) of the information, or methods of the concept hierarchies, to be utilized in mining.

Interestingness constraints – These defines thresholds on numerical measures of rule interestingness, including support, confidence, and correlation.

Rule constraints – These defines the form of rules to be mined. Such constraints can be defined as metarules (rule templates), as the maximum or minimum number of predicates that can appear in the rule antecedent or consequent, or as relationships between attributes, attribute values, and/or aggregates.

The following constraints can be described using a high-level declarative data mining query language and user interface. This form of constraint-based mining enables users to define the rules that they can like to uncover, thus by creating the data mining process more efficient.

Furthermore, a sophisticated mining query optimizer can be used to deed the constraints defined by the user, thereby creating the mining process more effective. Constraint-based mining boost interactive exploratory mining and analysis.