Data Warehousing and Data Mining

ECAP446

Edited by
Sartaj Singh

LOVELY PROFESSIONAL UNIVERSITY

Data Warehousing and Data Mining

Edited By:
Sartaj Singh

CONTENT

Unit - 1: Data Warehousing and Online Analytical Processing

CONTENTS

Objectives

After studying this unit, you will be able to:

- Know data warehouse concept.
- Outlines the difference between the operational database system and data warehouse.
- Acquire how data cubes model n-dimensional data
- Examine the way to index OLAP information by bitmap indexing and join indexing.
- Explores different methods to compute data cubes efficiently.
- Study the design and usage of data warehousing for information processing, analytical processing, and data mining.

Introduction

Data warehouses simplify and combine data in multidimensional space. The building of data warehouses includes data cleaning, data integration, and data transformation, and can be seen as an significant preprocessing step for data mining. Furthermore, data warehouses offer online analytical processing (OLAP) tools for the collaborative analysis of multidimensional data of diverse granularities, which simplifies effective data generalization and data mining. Numerous other data

mining tasks, such as association, classification, prediction, and clustering, can be combined with OLAP operations to improve interactive mining of knowledge at several levels of abstraction. Henceforth, the data warehouse has convert an progressively important stage for data analysis and OLAP and will deliver an effective platform for data mining. So, data warehousing and OLAP form an important step in the knowledge discovery process. This chapter focus on the overview of data warehouse and OLAP technology.

1.1 What Is a Data Warehouse?

Data warehouses have been well-defined in many ways,making it hard to articulate a demanding definition. Lightly speaking, a data warehouse refers to a data repository that is maintained separately from an organization's operational databases. Data warehouse systems allow for the integration of a variety of application systems. They support information processing by providing a solid platform of consolidated historic data for analysis.

According to William H. Inmon, a leading architect in the construction of data warehouse systems, "A data warehouse is a subject-oriented, integrated, time-variant, and non-volatile collection of data in support of management's decision-making process". This short but comprehensive definition presents the major features of a data warehouse. The four keywords—subject-oriented, integrated, time-variant, and non-volatile—differentiate data warehouses from other data source systems, such as relational database systems, transaction processing systems, and file systems.

Let's take a closer look at each of these key features.

- **Subject-oriented**: A data warehouse is systematized around major subjects such as customer, supplier, product, and sales. Rather than focussed on the day-to-day operations and transaction processing of an organization, a data warehouse emphases on the modeling and analysis of data for decision-makers. Henceforth, data warehouses usually provide a simple and succinct view of particular subject issues by excluding data that are not useful in the decision support process.
- **Integrated**: A data warehouse is usually constructed by integrating multiple heterogeneous sources, such as relational databases, flat files, and online transaction records. Data cleaning and data integration techniques are applied to ensure consistency in naming conventions, encoding structures, attribute measures, and so on.
- **Time-variant**: Data is stored to provide information from a historic perspective (e.g., the past 5–10 years). Every key structure in the data warehouse contains, either implicitly or explicitly, a time element.
- **Non-volatile:** A data warehouse is always a physically separate store of data transformed from the application data found in the operational environment. Due to this separation, a data warehouse does not require transaction processing, recovery, and concurrency control mechanisms. It usually requires only two operations in data accessing: *initial loading of data* and *access of data.*

Example: A typical data warehouse is organized around major subjects, such as customer, vendor, product, and sales rather than concentrating on the day-to-day operations and transaction processing of an organization.

Features of a Data Warehouse

- ***It is separate from the Operational Database.***
- ***Integrates data from heterogeneous systems.***
- ***Stores HUGE amount of data, more historical than current data.***
- ***Does not require data to be highly accurate.***
- ***Queries are generally complex.***

- *The goal is to execute statistical queries and provide results that can influence decision-making in favor of the Enterprise.*
- *These systems are thus called Online Analytical Processing Systems (OLAP).*

1.2 The need for a Separate Data Warehouse

Because operational databases store huge amounts of data, you may wonder, "Why not perform online analytical processing directly on such databases instead of spending additional time and resources to construct a separate data warehouse?" A major reason for such a separation is to help promote the high performance of both systems. An operational database is designed and tuned from known tasks and workloads like indexing and hashing using primary keys, searching for particular records, and optimizing "canned" queries. On the other hand, data warehouse queries are often complex. They involve the computation of large data groups at summarized levels and may require the use of special data organization, access, and implementation methods based on multidimensional views. Processing OLAP queries in operational databases would substantially degrade the performance of operational tasks.

Moreover, an operational database supports the concurrent processing of multiple transactions. Concurrency control and recovery mechanisms (e.g., locking and logging) are required to ensure the consistency and robustness of transactions. An OLAP query often needs read-only access to data records for summarization and aggregation. Concurrency control and recovery mechanisms, if applied for such OLAP operations, may jeopardize the execution of concurrent transactions and thus substantially reduce the throughput of an OLTP system.

Finally, the separation of operational databases from data warehouses is based on the different structures, contents, and uses of the data in these two systems. Decision support requires historic data, whereas operational databases do not typically maintain historic data. In this context, the data in operational databases, though abundant, are usually far from complete for decision making. Decision support requires consolidation (e.g., aggregation and summarization) of data from heterogeneous sources, resulting in high-quality, clean, integrated data. In contrast, operational databases contain only detailed raw data, such as transactions, which need to be consolidated before analysis. Because the two systems provide quite different functionalities and require different kinds of data, it is presently necessary to maintain separate databases.

1.3 Data Warehouse Models: Enterprise Warehouse, Data Mart, and Virtual Warehouse

From the architecture point of view, there are three data warehouse models: *the enterprise warehouse, the data mart,* and the *virtual warehouse*.

Enterprise warehouse: An enterprise warehouse collects all of the information about subjects spanning the entire organization. It provides corporate-wide data integration, usually from one or more operational systems or external information providers, and is cross-functional in scope. It typically contains detailed data as well as summarized data and can range in size from a few gigabytes to hundreds of gigabytes, terabytes, or beyond. An enterprise data warehouse may be implemented on traditional mainframes, computer super servers, or parallel architecture platforms. It requires extensive business modeling and may take years to design and build.

Datamart: A data mart contains a subset of corporate-wide data that is of value to a specific group of users. The scope is confined to specific selected subjects. For example, a marketing data mart may confine its subjects to a customer, item, and sales. The data contained in data marts tend to be summarized.

Depending on the source of data, data marts can be categorized into the following two classes:

1. Independent data marts are sourced from data captured from one or more operational systems or external information providers, or data generated locally within a particular department or geographic area.

2. Dependent data marts are sourced directly from enterprise data warehouses.

Virtual warehouse: A virtual warehouse is a set of views over operational databases. For efficient query processing, only some of the possible summary views may be materialized. A virtual warehouse is easy to build but requires excess capacity on operational database servers.

A recommended method for the development of data warehouse systems is to implement the warehouse incrementally and evolutionarily, as shown in Figure 1.

First, a high-level corporate data model is defined within a reasonably short period (such as one or two months) that provides a corporate-wide, consistent, integrated view of data among different subjects and potential usages. This high-level model, although it will need to be refined in the further development of enterprise data warehouses and departmental data marts, will greatly reduce future integration problems. Second, independent data marts can be implemented in parallel with the enterprise warehouse based on the same corporate data model set noted before. Third, distributed data marts can be constructed to integrate different data marts via hub servers. Finally, a multitier data warehouse is constructed where the enterprise warehouse is the sole custodian of all warehouse data, which is then distributed to the various dependent data marts.

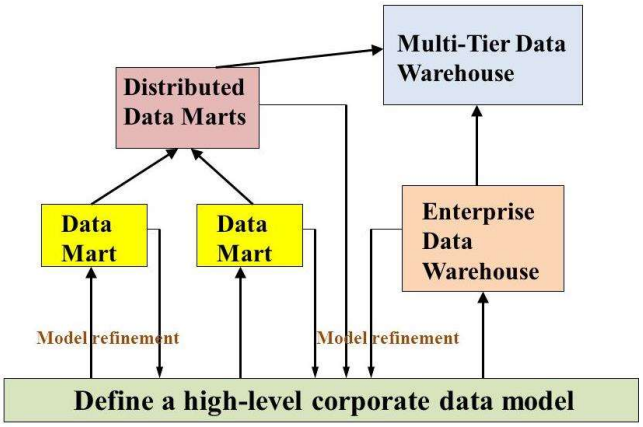


Figure 1 A recommended approach for data warehouse development

1.4 Differences between Operational Database Systems and Data Warehouses.

Because most people are familiar with commercial relational database systems, it is easy to understand what a data warehouse is by comparing these two kinds of systems.

The major task of online operational database systems is to perform online transaction and query processing. These systems are called **online transaction processing (OLTP) systems**. They cover most of the day-to-day operations of an organization such as purchasing, inventory, manufacturing, banking, payroll, registration, and accounting. Data warehouse systems, on the other hand, service users or knowledge workers in the role of data analysis and decision making. Such systems can organize and present data in various formats to accommodate the diverse needs of different users. These systems are known as **online analytical processing (OLAP) systems.**

The major distinguishing features between OLTP and OLAP are summarised in Table 1.

Table 1 Difference between OLTP and OLAP System

Feature	OLTP System	OLAP System
Characteristic	Operational Processing	Informational Processing
Users	Clerks, clients, and information technology professionals.	Knowledge workers, including managers, executives, and analysts.
System orientation	Customer-oriented and used for transaction and query processing Day to day operations	Market-oriented and used for data analysis long-term informational requirements, decision support.
Data contents	Manages current data that typically, are too detailed to be easily used for decision making.	Manages large amounts of historical data, provides facilities for summarization and

		aggregation, and stores and manages information at different levels of granularity.
Database design	Adopts an entity-relationship (ER) data model and an application-oriented database design	Adopts either a star or snowflake model and a subject-oriented database design.
View	Focuses mainly on the current data within an enterprise or department, without referring to historical data or data in different organizations.	In contrast, an OLAP system often spans multiple versions of a database schema, due to the evolutionary process of an organization. OLAP systems also deal with information that originates from different organizations, integrating information from many data stores.
Volume of data	Not very large	Because of their huge volume, OLAP data are stored on multiple storage media.
Access patterns	Consists mainly of short, atomic transactions. Such a system requires concurrency control and recovery mechanisms.	Accesses to OLAP systems are mostly read-only operations (since most data warehouses store historical rather than up-to-date information), although many could be complex queries.
Access mode	Read/write	Mostly write
Focus	Data in	Information out
Operations	Index/hash on a primary key	Lots of scans
Number of records accessed	Tens	Continue…. Millions
Number of users	Thousands	Hundreds
DB size	100 MB to GB	100 GB to TB
Priority	High performance, high availability	High flexibility, end-user autonomy
Metric	Transaction throughput	Query response time

Exactly what the difference between operational database and data warehouse? Explain with the suitable of suitable example.

1.5 Data Warehouse Modeling: Data Cube

The entity-relationship data model is commonly used in the design of relational databases, where a database schema consists of a set of entities and the relationships between them. Such a data model

is appropriate for on-line transaction processing. The data warehouse requires a concise, subject-oriented schema that facilitates OLAP. Data warehouses and OLAP tools are based on a multidimensional data model. This model views data in the form of a data cube. A data cube allows data to be modeled and viewed in multiple dimensions. It is defined by dimensions and facts. In general terms, dimensions are the perspectives or entities concerning which an organization wants to keep records. Each dimension may have a table associated with it, called a dimension table, which further describes the dimension. For example, a dimension table for an item may contain the attributes item name, brand, and type. Dimension tables can be specified by users or experts, or automatically generated and adjusted based on data distributions.

A multidimensional data model is typically organized around a central theme, such as sales. This theme is represented by a fact table. Facts are numeric measures. Think of them as the quantities by which we want to analyze relationships between dimensions. Examples of facts for a sales data warehouse include dollars sold (sales amount in dollars), units sold (number of units sold), and the amount budgeted. The fact table contains the names of the facts, or measures, as well as keys to each of the related dimension tables. You will soon get a clearer picture of how this works when we look at multidimensional schemas.

Table 2 2-D View of Sales Data for AllElectronics According to time and item

location = "Vancouver"				
	item (type)			
time (quarter)	home entertainment	computer	phone	security
Q1	605	825	14	400
Q2	680	952	31	512
Q3	812	1023	30	501
Q4	927	1038	38	580

Note: The sales are from branches located in the city of Vancouver. The measure displayed is dollars sold (in thousands).

Table 3 3-D View of Sales Data for AllElectronics According to time, item, and location

	location = "Chicago"				**location** = "New York"				**location** = "Toronto"				**location** = "Vancouver"			
	item				**item**				**item**				**item**			
	home				home				home				home			
time	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.
Q1	854	882	89	623	1087	968	38	872	818	746	43	591	605	825	14	400
Q2	943	890	64	698	1130	1024	41	925	894	769	52	682	680	952	31	512
Q3	1032	924	59	789	1034	1048	45	1002	940	795	58	728	812	1023	30	501
Q4	1129	992	63	870	1142	1091	54	984	978	864	59	784	927	1038	38	580

Table 2 and Table 3 show the data at different degrees of summarization. In the data warehousing research literature, a data cube like those shown in Figure 2 and Figure 3 is often referred to as a **cuboid.** Given a set of dimensions, we can generate a cuboid for each of the possible subsets of the given dimensions. The result would form a lattice of cuboids, each showing the data at a different level of summarization, or group-by. The lattice of cuboids is then referred to as a **data cube**. Figure 4 shows a lattice of cuboids forming a data cube for the dimensions time, item, location, and supplier.

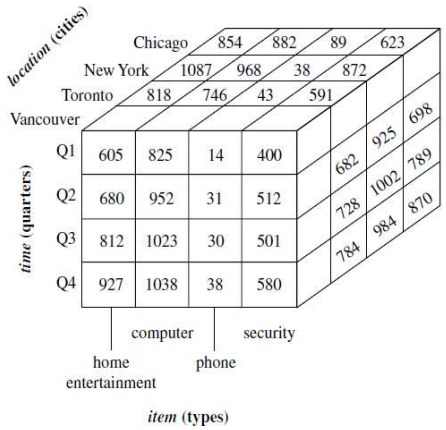


Figure 2 A 3-D data cube representation of the data in Table 3 according to time, item, and location.

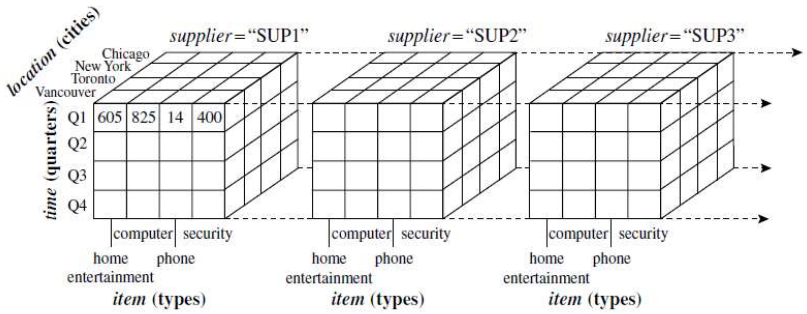


Figure 3 A 4-D data cube representation of sales data, according to time, item, location, and supplier.

The cuboid that holds the lowest level of summarization is called the base cuboid. For example, the 4-D cuboid in Figure 3 is the base cuboid for the given time, item, location, and supplier dimensions. Figure 2 is a 3-D (non-base) cuboid for time, item, and location summarized for all suppliers. The 0-D cuboid, which holds the highest level of summarization, is called the **apex cuboid**. In our example, this is the total sales, or dollars sold, summarized over all four dimensions. The apex cuboid is typically denoted by all.

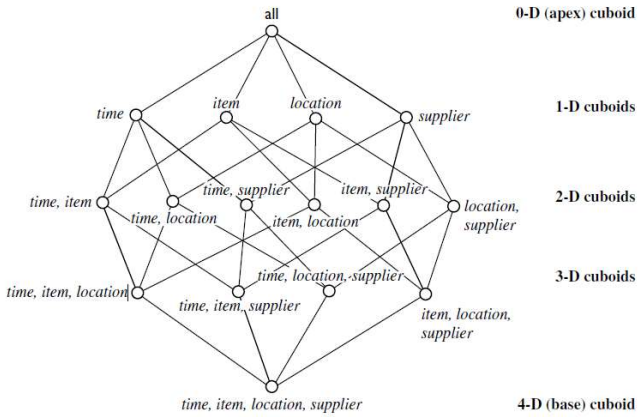


Figure 4 Lattice of cuboids, making up a 4-D data cube for time, item, location, and supplier.

Example: "Date" can be grouped into "day", "month", "quarter", "year" or "week", which forms a lattice structure.

1.6 Conceptual Modeling of Data Warehouse

The most popular data model for a data warehouse is a multidimensional model, which can exist in the form of a **star schema, a snowflake schema**, or a **fact constellation schema**. Let's look at each of these.

Star schema: The most common modeling paradigm is the star schema, in which the data warehouse contains (1) a large central table (fact table) containing the bulk of the data, with no redundancy, and (2) a set of smaller attendant tables (dimension tables), one for each dimension. The schema graph resembles a starburst, with the dimension tables displayed in a radial pattern around the central fact table.

A star schema for AllElectronics sales is shown in Figure 5. Sales are considered along four dimensions: time, item, branch, and location. The schema contains a central fact table for sales that contain keys to each of the four dimensions, along with two measures: dollars sold and units sold. To minimize the size of the fact table, dimension identifiers (e.g., time key and item key) are system-generated identifiers. Notice that in the star schema, each dimension is represented by only one table, and each table contains a set of attributes.

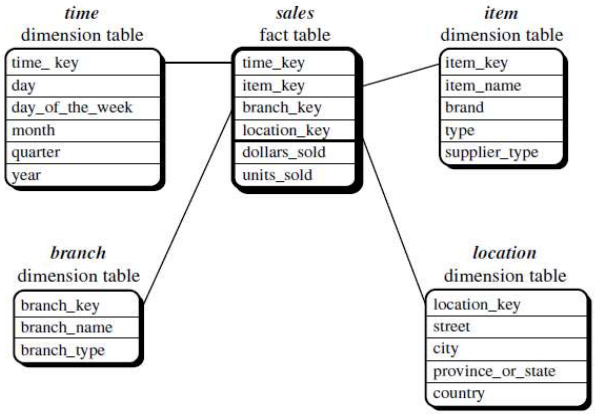


Figure 5 Star schema of a sales data warehouse.

Example: Let, an organization sells products throughout the world. The main four major dimensions are time, location, time, and branch.

Snowflake schema: The snowflake schema is a variant of the star schema model, where some dimension tables are normalized, thereby further splitting the data into additional tables. The resulting schema graph forms a shape similar to a snowflake.

The major difference between the snowflake and star schema models is that the dimension tables of the snowflake model may be kept in the normalized form to reduce redundancies. Such a table is easy to maintain and saves storage space. However, this space savings is negligible in comparison to the typical magnitude of the fact table. Furthermore, the snowflake structure can reduce the effectiveness of browsing, since more joins will be needed to execute a query. Consequently, the system performance may be adversely impacted. Hence, although the snowflake schema reduces redundancy, it is not as popular as the star schema in data warehouse design.

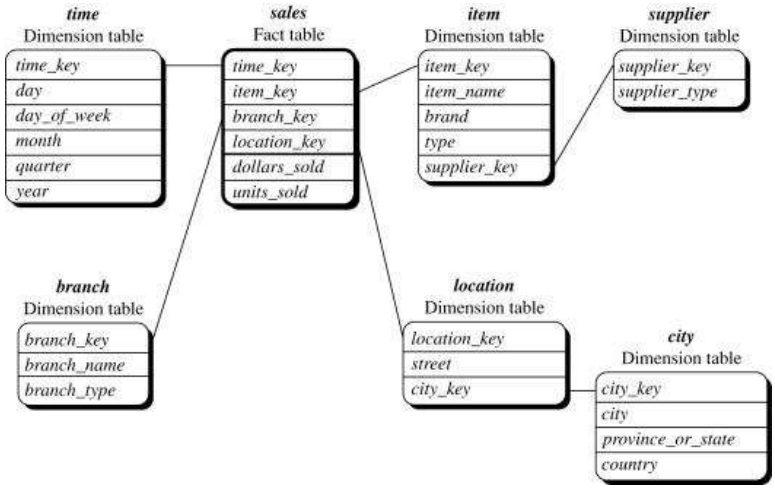


Figure 6 Snowflake schema of a sales data warehouse.

A snowflake schema for AllElectronics sales is given in Figure 6. Here, the sales fact table is identical to that of the star schema in Figure 5. The main difference between the two schemas is in the definition of dimension tables. The single dimension table for an item in the star schema is normalized in the snowflake schema, resulting in new item and supplier tables. For example, the item dimension table now contains the attributes *item key, item name, brand, type, and supplier key*, where supplier key is linked to the supplier dimension table, containing supplier key and supplier type information. Similarly, the single dimension table for location in the star schema can be normalized into two new tables: location and city. The city key in the new location table links to the city dimension. Notice that, when desirable, further normalization can be performed on province or state and country in the snowflake schema shown in Figure 6.

Fact constellation: Sophisticated applications may require multiple fact tables to share dimension tables. This kind of schema can be viewed as a collection of stars, and hence is called a galaxy schema or a fact constellation.

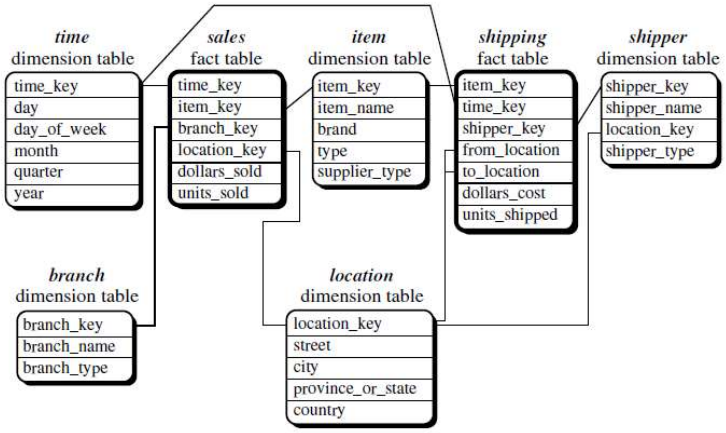


Figure 7 Fact constellation schema of a sales and shipping data warehouse.

A fact constellation schema is shown in Figure 7. This schema specifies two fact tables, sales, and shipping. The sales table definition is identical to that of the star schema (Figure 5). The shipping table has five dimensions, or keys—item key, time key, shipper key, from location, and to location—and two measures—*dollars cost and units shipped*. A fact constellation schema allows dimension tables to be shared between fact tables. For example, the dimensions tables for time, item, and location are shared between the sales and shipping fact tables.

1.7 Concept Hierarchies

*A **concept hierarchy** defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts. Consider a concept hierarchy for the dimension location. City values for location include*

Vancouver, Toronto, New York, and Chicago. Each city, however, can be mapped to the province or state to which it belongs. For example, Vancouver can be mapped to British Columbia, and Chicago to Illinois. The provinces and states can in turn be mapped to the country (e.g., Canada or the United States) to which they belong. These mappings form a concept hierarchy for the dimension location, mapping a set of low-level concepts (i.e., cities) to higher-level, more general concepts (i.e., countries). This concept hierarchy is illustrated in

Figure 8.

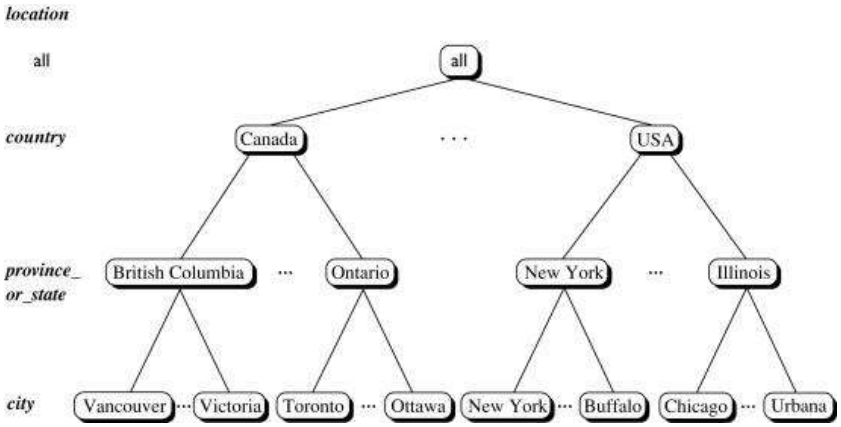


Figure 8 A concept hierarchy for location.

1.8 Measures: Their Categorization and Computation

"How are measures computed?" To answer this question, we first study how measures can be categorized. Note that a multidimensional point in the data cube space can be defined by a set dimension–value pairs; for example,(time ="Q1", location = "Vancouver", item ="computer"). A data cube measure is a numeric function that can be evaluated at each point in the data cube space. A measured value is computed for a given point by aggregating the data corresponding to the respective dimension-value pairs defining the given point. We will look at concrete examples of this shortly.

Distributive: An aggregate function is distributive if it can be computed in a distributed manner as follows. Suppose the data are partitioned into n sets. We apply the function to each partition, resulting in n aggregate values. If the result derived by applying the function to the n aggregate values is the same as that derived by applying the function to the entire data set (without partitioning), the function can be computed in a distributed manner. For example, sum() can be computed for a data cube by first partitioning the cube into a set of sub-cubes, computing sum() for each sub-cube, and then summing up the counts obtained for each sub-cube. Hence, sum() is a distributive aggregate function. For the same reason, count(), min(), and max() are distributive aggregate functions.

Algebraic: An aggregate function is algebraic if it can be computed by an algebraic function with M arguments (where M is a bounded positive integer), each of which is obtained by applying a distributive aggregate function. For example, avg() (average) can be computed by sum()/count(), where both sum() and count() are distributive aggregate functions. Similarly, it can be shown that min N() and max N() (which find the N minimum and N maximum values, respectively, in a given set) and standard deviation() are algebraic aggregate functions. A measure is algebraic if it is obtained by applying an algebraic aggregate function.

Holistic: An aggregate function is holistic if there is no constant bound on the storage size needed to describe a sub-aggregate. That is, there does not exist an algebraic function with M arguments (where M is a constant) that characterizes the computation. Common examples of holistic functions include median(), mode(), and rank(). A measure is holistic if it is obtained by applying a holistic aggregate function.

1.9 OLAP Operations

"A data cube allows data to be modeled and viewed in multiple dimensions. It is defined by dimensions and facts.

Dimensions are the entities concerning which an organization wants to keep records.

Facts are numerical measures. It is the quantities by which we want to analyze relationships between dimensions.

The data cube is used by the users of the decision support system to see their data. The cuboid that holds the lowest level of summarization is called the **base cuboid**. The 0-D cuboid, which holds the highest level of summarization, is called the **apex cuboid**.

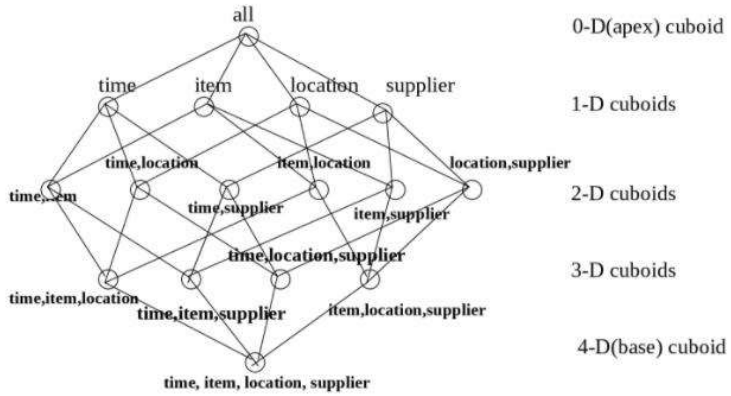


Figure 9 A Lattice of Cuboids

1.10 Operations in the Multidimensional Data Model (OLEP)

In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies. The organization provides users with the flexibility to view data from different perspectives. Some OLAP data cube operations exist to materialize these different views, allowing interactive querying and analysis of the data at hand.

Five basic OLAP commands are used to perform data retrieval from a Data warehouse.

1. Roll-up (drill-up):- The roll-up operation performs aggregation on a data cube either by climbing up the hierarchy or by dimension reduction.

Location	Medal
Delhi	5
New York	2
Patiala	3
Los Angles	5

→

Location	Medal
India	8
America	7

Delhi, New York, Patiala, and Los Angeles wins 5, 2, 3, and 5 medals respectively. So in this example, we roll upon Location from cities to countries.

2. Drill-down:- Drill-down is the reverse of roll-up. That means lower-level summary to higher-level summary.

Drill-down can be performed either by:-

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

Location	Medal
India	8
America	7

Location	Medal
Delhi	5
New York	2
Patiala	3
Los Angles	5

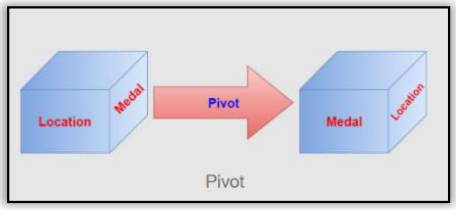
Drill-down on Location from countries to cities.

3. Slice and dice:- The slice operation perform a selection on one dimension of the given cube, resulting in a subcube. Reduces the dimensionality of the cubes. For example, if we want to make a selection where Medal = 5

Location	Medal
Delhi	5
Los Angles	5

The dice operation defines a sub-cube by performing a selection on two or more dimensions. For example, if we want to make a selection where Medal = 3 or Location = New York.

4. Pivot:- Pivot is also known as rotate. It rotates the data axis to view the data from different perspectives.



Discuss OLAP application identify its characteristics.

1.11 DataWarehouse Design and Usage

"What can business analysts gain from having a data warehouse?" First, having a data warehouse may provide a competitive advantage by presenting relevant information from which to measure performance and make critical adjustments to help win over competitors. Second, a data warehouse can enhance business productivity because it can quickly and efficiently gather the information that accurately describes the organization. Third, a data warehouse facilitates customer relationship management because it provides a consistent view of customers and items across all lines of business, all departments, and all markets. Finally, a data warehouse may bring about cost reduction by tracking trends, patterns, and exceptions over long periods consistently and reliably.

To design an effective data warehouse we need to understand and analyze business needs and construct a business analysis framework. The construction of a large and complex information system can be viewed as the construction of a large and complex building, for which the owner, architect, and builder have different views.

Four different views regarding a data warehouse design must be considered: *the top-down view, the data source view, the data warehouse view,* and *the business query view*.

- The **top-down view** allows the selection of the relevant information necessary for the data warehouse. This information matches current and future business needs.
- The **data source view** exposes the information being captured, stored, and managed by operational systems. This information may be documented at various levels of detail and accuracy, from individual data source tables to integrated data source tables. Data sources are often modeled by traditional data modeling techniques, such as the entity-relationship model or CASE (computer-aided software engineering) tools.

- The **data warehouse view** includes fact tables and dimension tables. It represents the information that is stored inside the data warehouse, including pre-calculated totals and counts, as well as information regarding the source, date, and time of origin, added to provide historical context.
- Finally, the **business query view** is the data perspective in the data warehouse from the end-user's viewpoint.

Building and using a data warehouse is a complex task because it requires *business skills, technology skills*, and *program management skills*. Regarding *business skills*, building a data warehouse involves understanding how systems store and manage their data, how to build **extractors** that transfer data from the operational system to the data warehouse, and how to build **warehouse refresh software** that keeps the data warehouse reasonably up-to-date with the operating system's data. Using a data warehouse involves understanding the significance of the data it contains, as well as understanding and translating the business requirements into queries that can be satisfied by the data warehouse.

Regarding *technology skills*, data analysts are required to understand how to make assessments from quantitative information and derive facts based on conclusions from historic information in the data warehouse. These skills include the ability to discover patterns and trends, extrapolate trends based on history and look for anomalies or paradigm shifts, and to present coherent managerial recommendations based on such analysis. Finally, *program management skills* involve the need to interface with many technologies, vendors, and end-users to deliver results in a timely and cost-effective manner.

Data Warehouse Design Process

Let's look at various approaches to the data warehouse design process and the steps involved. A data warehouse can be built using a **top-down approach, a bottom-up approach**, or a **combination of both.** The top-down approach starts with overall design and planning. It is useful in cases where the technology is mature and well known, and where the business problems that must be solved are clear and well understood. The *bottom-up approach* starts with experiments and prototypes. This is useful in the early stage of business modeling and technology development. It allows an organization to move forward at considerably less expense and to evaluate the technological benefits before making significant commitments. In the combined approach, an organization can exploit the planned and strategic nature of the top-down approach while retaining the rapid implementation and opportunistic application of the bottom-up approach.

In general, the warehouse design process consists of the following steps:

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

Because data warehouse construction is a difficult and long-term task, its implementation scope should be clearly defined. The goals of an initial data warehouse implementation should be *specific, achievable, and measurable*. This involves determining the time and budget allocations, the subset of the organization that is to be modeled, the number of data sources selected, and the number and types of departments to be served.
Once a data warehouse is designed and constructed, the initial deployment of the warehouse includes initial installation, roll-out planning, training, and orientation. Platform upgrades and

maintenance must also be considered. Data warehouse administration includes data refreshment, data source synchronization, planning for disaster recovery, managing access control and security, managing data growth, managing database performance, and data warehouse enhancement and extension. Scope management includes controlling the number and range of queries, dimensions, and reports; limiting the data warehouse's size; or limiting the schedule, budget, or resources. Various kinds of data warehouse design tools are available. **Data warehouse development tools** provide functions to define and edit metadata repository contents (e.g., schemas, scripts, or rules), answer queries, output reports, and ship metadata to and from relational database system catalogs. **Planning and analysis tools** study the impact of schema changes and refresh performance when changing refresh rates or time windows.

Data Warehouse Usage for Information Processing

Data warehouses and data marts are used in a wide range of applications. There are three kinds of data warehouse applications*: information processing, analytical processing, and data mining*.

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

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

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

"*How does data mining relate to information processing and online analytical processing*?" Information processing, based on queries, can find useful information. However, answers to such queries reflect the information directly stored in databases or computable by aggregate functions. They do not reflect sophisticated patterns or regularities buried in the database. Therefore, information processing is not data mining. Because data mining systems can also mine generalized class/concept descriptions, this raises some interesting questions: *"Do OLAP systems perform data mining? Are OLAP systems are data mining systems?"* The functionalities of OLAP and data mining can be viewed as disjoint: OLAP is a data summarization/aggregation tool that helps simplify data analysis, while data mining allows the automated discovery of implicit patterns and interesting knowledge hidden in large amounts of data. OLAP tools are targeted toward simplifying and supporting interactive data analysis, whereas the goal of data mining tools is to automate as much of the process as possible, while still allowing users to guide the process. In this sense, data mining goes one step beyond traditional online analytical processing. Data mining is not confined to the analysis of data stored in data warehouses. It may analyze data existing at more detailed granularities than the summarized data provided in a data warehouse. It may also analyze transactional, spatial, textual, and multimedia data that are difficult to model with current multidimensional database technology. In this context, data mining covers a broader spectrum than OLAP for data mining functionality and the complexity of the data handled.

1.12 From Online Analytical Processing to Multidimensional Data Mining

The data mining field has conducted substantial research regarding mining on various data types, including relational data, data from data warehouses, transaction data, time-series data, spatial data, text data, and flat files. **Multidimensional data mining** (also known as *exploratory multidimensional data mining*, **online analytical mining, or OLAM**) integrates OLAP with data mining to uncover knowledge in multidimensional databases. Among the many different paradigms and architectures of data mining systems, multidimensional data mining is particularly important for the following reasons:

High quality of data in data warehouses: Most data mining tools need to work on integrated, consistent, and cleaned data, which requires costly data cleaning, data integration, and data transformation as preprocessing steps. A data warehouse constructed by such preprocessing serves as a valuable source of high-quality data for OLAP as well as for data mining. Notice that data mining may serve as a valuable tool for data cleaning and data integration as well. Available information processing infrastructure surrounding data warehouses: Comprehensive information processing and data analysis infrastructures have been or will be systematically constructed surrounding data

warehouses, which include access, integration, consolidation, and transformation of multiple heterogeneous databases, ODBC/OLEDB connections, Web access, and service facilities, and reporting and OLAP analysis tools. It is prudent to make the best use of the available infrastructures rather than constructing everything from scratch.

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

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

1.13 DataWarehouse Implementation

Data warehouses contain huge volumes of data. OLAP servers demand that decision support queries be answered in the order of seconds. Therefore, it is crucial for data warehouse systems to support highly efficient cube computation techniques, access methods, and query processing techniques.

Efficient Data Cube Computation: An Overview

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

The compute cube Operator and the Curse of Dimensionality

One approach to cube computation extends SQL to include a compute cube operator. The compute cube operator computes aggregates over all subsets of the dimensions specified in the operation. This can require excessive storage space, especially for large numbers of dimensions. We start with an intuitive look at what is involved in the efficient computation of data cubes. Suppose that you want to create a data cube for all Electronics sales contain the following: city, item, year, and sales in dollars. You want to be able to analyze the data, with queries such as the following:

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

"Compute the sum of sales, grouping by city."

"Compute the sum of sales, grouping by item."

What is the total number of cuboids, or group-by's, that can be computed for this data cube? Taking the three attributes, city, item, and year, as the dimensions for the data cube, and sales in dollars as the measure, the total number of cuboids, or group by's, that can be computed for this data cube is 2^3=8. The possible group-by's are the following: {(city, item, year), (city, item), (city, year), (item, year), (city), (item), (year),( ),} where ( )means that the group-by is empty (i.e., the dimensions are not grouped). These group-by's form a lattice of cuboids for the data cube, as shown in Figure 10.

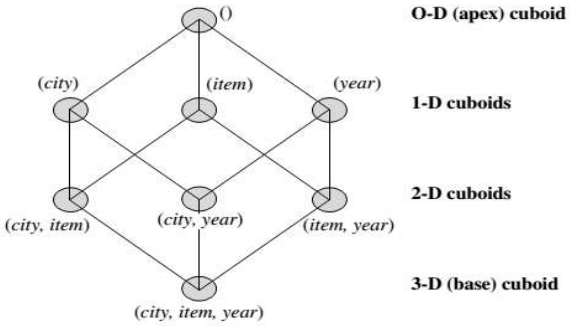


Figure 10 Lattice of cuboids, making up a 3-D data cube.

The **base cuboid** contains all three dimensions, city, item, and year. It can return the total sales for any combination of the three dimensions. The **apex cuboid**, or 0-D cuboid, refers to the case where the group-by is empty. It contains the total sum of all sales. The base cuboid is the least generalized (most specific) of the cuboids. The apex cuboid is the most generalized (least specific) of the cuboids and is often denoted as all. If we start at the apex cuboid and explore downward in the lattice, this is equivalent to drilling down within the data cube. If we start at the base cuboid and explore upward, this is akin to rolling up.

Online analytical processing may need to access different cuboids for different queries. Therefore, it may seem like a good idea to compute in advance all or at least some of the cuboids in a data cube. A major challenge related to this pre-computation, however, is that the required storage space may explode if all the cuboids in a data cube are pre-computed, especially when the cube has many dimensions. The storage requirements are even more excessive when many of the dimensions have associated concept hierarchies, each with multiple levels. This problem is referred to as the **curse of dimensionality**.

"How many cuboids are there in an n-dimensional data cube?" If there were no hierarchies associated with each dimension, then the total number of cuboids for an n-dimensional data cube, as we have seen, is 2n. However, in practice, many dimensions do have hierarchies. For example, time is usually explored not at only one conceptual level (e.g., year), but rather at multiple conceptual levels such as in the hierarchy "day < month < quarter < year." For an n-dimensional data cube, the total number of cuboids can be generated.

Total number of cuboids D, $= \prod_{i=1}^{n} (L_i + 1),$

.

where Li is the number of levels associated with dimension i. One is added to Li to include the virtual top level, all. This formula is based on the fact that, at most, one abstraction level in each dimension will appear in a cuboid. For example, the time dimension as specified before has four conceptual levels, or five if we include the virtual level all. If the cube has 10 dimensions and each dimension has five levels (including all), the total number of cuboids that can be generated is $5^{10} \approx 9.8 \times 10^6$. The size of each cuboid also depends on the cardinality (i.e., number of distinct values) of each dimension.

Partial Materialization: Selected Computation of Cuboids

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

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

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

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

The partial materialization of cuboids or subcubes should consider three factors: (1) identify the subset of cuboids or subcubes to materialize; (2) exploit the materialized cuboids or subcubes during query processing; and (3) efficiently update the materialized cuboids or subcubes during load and refresh. The selection of the subset of cuboids or subcubes to materialize should take into account the queries in the workload, their frequencies, and their accessing costs. Alternatively, we can compute an **iceberg cube**, which is a data cube that stores only those cube cells with an aggregate value (e.g., count) that is above some minimum support threshold. Another common strategy is to materialize a **shell cube**. This involves precomputing the cuboids for only a small number of dimensions (e.g., three to five) of a data cube.

1.14 Indexing OLAP Data: Bitmap Index and Join Index

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

In the AllElectronics data warehouse, suppose the dimension item at the top level has four values (representing item types): "home entertainment," "computer," "phone," and "security." Each value (e.g., "computer") is represented by a bit vector in the item bitmap index table. Suppose that the cube is stored as a relation table with 100,000 rows. Because the domain of the item consists of four values, the bitmap index table requires four-bit vectors (or lists), each with 100,000 bits. Figure 11 shows a base (data) table containing the dimensions item and city, and its mapping to bitmap index tables for each of the dimensions.

Base table

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

item bitmap index table

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

city bitmap index table

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

Figure 11 Indexing OLAP data using bitmap indices.

Bitmap indexing is advantageous compared to hash and Tree indices. It is especially useful for low-cardinality domains because comparison, join, and aggregation operations are then reduced to bit arithmetic, which substantially reduces the processing time. Bitmap indexing leads to significant reductions in space and input/output (I/O) since a string of characters can be represented by a single bit. For higher-cardinality domains, the method can be adapted using compression techniques.

The **join indexing** method gained popularity from its use in relational database query processing. Traditional indexing maps the value in a given column to a list of rows having that value. In contrast, join indexing registers the joinable rows of two relations from a relational database. For example, if two relations R.RID, A/ and S.B, SID/ join on the attributes A and B, then the join index record contains the pair.RID, SID/, where RID and SID are record identifiers from the R and S relations, respectively. Hence, the join index records can identify joinable tuples without performing costly join operations. Join indexing is especially useful for maintaining the relationship between a foreign key and its matching primary keys, from the joinable relation.

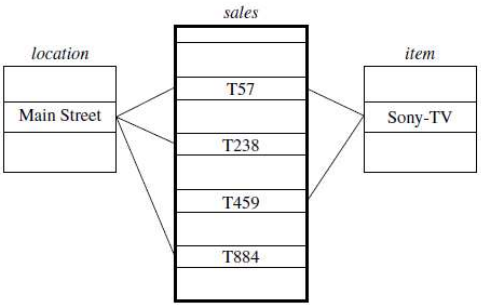


Figure 12 Linkages between a sales fact table and location and item dimension tables.

We defined a star schema for AllElectronics of the form "sales star [time, item, branch, location]: dollars_sold = sum (sales_in_dollars)." An example of a join index relationship between the sales fact table and the location and item dimension tables is shown in Figure 12. For example, the "Main Street" value in the location dimension table joins with tuples T57, T238, and T884 of the sales fact table. Similarly, the "Sony-TV" value in the item dimension table joins with tuples T57 and T459 of the sales fact table. The corresponding join index tables are shown in Figure 13.

Join index table for *location/sales*

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

Join index table for *item/sales*

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

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

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

Figure 13 Join index tables based on the linkages between the sales fact table and the location and item.

Efficient Processing of OLAP Queries

The purpose of materializing cuboids and constructing OLAP index structures is to speed up query processing in data cubes. Given materialized views, query processing should proceed as follows:

1. **Determine which operations should be performed on the available cuboids:** This involves transforming any selection, projection, roll-up (group-by), and drill-down operations specified in the query into the corresponding SQL and/or OLAP operations. For example, slicing and dicing a data cube may correspond to selection and/or projection operations on a materialized cuboid.

2. **Determine to which materialized cuboid(s) the relevant operations should be applied:** This involves identifying all of the materialized cuboids that may potentially be used to answer the query, pruning the set using knowledge of "dominance" relationships among the cuboids, estimating the costs of using the remaining materialized cuboids and selecting the cuboid with the least cost.

Suppose that we define a data cube for AllElectronics of the form"sales cube [time, item, location]: sum(sales_in_dollars)." The dimension hierarchies used are"day < month < quarter < year" for time; "item name < brand < type" for item; and "street < city < province or state < country" for location. Suppose that the query to be processed is on {brand, province, or state}, with the selection constant "year = 2010." Also, suppose that there are four materialized cuboids available, as follows:

cuboid 1: {year, item name, city}

cuboid 2: {year, brand, country}

cuboid 3: {year, brand, province or state}

cuboid 4: {item name, province or state}, where year = 2010

"Which of these four cuboids should be selected to process the query?" Finer-granularity data cannot be generated from coarser-granularity data. Therefore, cuboid 2 cannot be used because the country is a more general concept than province or state. Cuboids 1, 3, and 4 can be used to process the query because (1) they have the same set or a superset of the dimensions in the query, (2) the selection clause in the query can imply the selection in the cuboid, and (3) the abstraction levels for the item and location dimensions in these cuboids are at a finer level than brand and province or state, respectively.

"How would the costs of each cuboid compare if used to process the query?" Using cuboid 1 would likely cost the most because both item name and city are at a lower level than the brand and province or state concepts specified in the query. If there are not many year values associated with items in the cube, but there are several item names for each brand, then cuboid 3 will be smaller than cuboid 4, and thus cuboid 3 should be chosen to process the query. However, if efficient indices are available for cuboid 4, then cuboid 4 may be a better choice. Therefore, some cost-based estimation is required to decide which set of cuboids should be selected for query processing.

Summary

- A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile data collection organized in support of management decision-making.
- A data warehouse contains back-end tools and utilities for populating and refreshing the warehouse. These cover data extraction, data cleaning, data transformation, loading, refreshing, and warehouse management.
- A multidimensional data model is typically used for the design of corporate data warehouses and departmental data marts.
- A data cube consists of a lattice of cuboids, each corresponding to a different degree of summarization of the given multidimensional data.
- Concept hierarchies organize the values of attributes or dimensions into gradual abstraction levels. They are useful in mining at multiple abstraction levels.
- Full materialization refers to the computation of all of the cuboids in the lattice defining a data cube.
- OLAP query processing can be made more efficient with the use of indexing techniques.

Keywords

Data Sources: Data sources refer to any electronic repository of information that contains data of interest for management use or analytics.

Data Warehouse: It is a relational database that is designed for query and analysis rather than for transaction processing.

Data Mart: Data marts contain a subset of organization-wide data that is valuable to specific groups of people in an organization.

Dimensions: Dimensions contain a set of unique values that identify and categories data.

Hierarchy: A hierarchy is a way to organize data at different levels of aggregation.

Star Schema: A star schema is a convention for organizing the data into dimension tables, fact tables, and materialized views.

Self Assessment

1) OLTP stands for

(a) On-Line Transactional Processing

(b) On Link Transactional Processing

(c) On-Line Transnational Process

(d) On-Line Transactional Program

2) Data warehouse is

(a) The actual discovery phase of a knowledge discovery process

(b) The stage of selecting the right data for a KDD process

(c) A subject-oriented integrated time-variant non-volatile collection of data in support of management

(d) None of these

3) A data warehouse is which of the following?

(a) Can be updated by end-users.

(b) Contains numerous naming conventions and formats.

(c) Organized around important subject areas.

(d) Contains only current data. "

4) The data warehouse is normally a .....

5) A ..... is a physical database that receives all its information from the data warehouse.

Review Questions

1. Describe materialized views with the help of a suitable example.
2. What are the differences between the three main types of data warehouse usage: information processing, analytical processing, and data mining? Discuss the motivation behind OLAP mining (OLAM).
3. Describe OLAP operations in the multi-dimensional data model.
4. "Concept hierarchies that are common to many applications may be predefined in the data mining system". Explain.
5. "A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process." Discuss.
6. Differences between operational database systems and data warehouses.

Answers: Self-Assessment

1. a
2. c
3. c
4. relational database
5. dependent data mart

Further Readings

Jiawei Han, Micheline Kamber, Data Mining – Concepts and Techniques, Morgan Kaufmann Publishers, First Edition, 2003.

Matthias Jarke, Maurizio Lenzerini, Yannis Vassiliou, Panos Vassiliadis, *Fundamentals of Data Warehouses*, Publisher: Springer.

The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition.

Data Warehousing Fundamentals for IT Professionals.

Sam Anohory, Dennis Murray, *Data Warehousing in the Real World*, Addison Wesley, First Edition, 2000.

https://www.javatpoint.com/data-mining-cluster-vs-data-warehousing.

https://www.classcentral.com/subject/data-warehousing

https://www.tutorialspoint.com/dwh/dwh_data_warehousing.htm

https://www.oracle.com/in/database/what-is-a-data-warehouse

Unit 02: Introduction to Data Mining

CONTENTS

Objectives

After this unit you will be able to:

- Know the concept of data mining
- Understand the knowledge discovery process
- Analyze different types of data repositories
- Understand various data mining functionalities.
- Know the various tasks and current trends in data mining
- Learn various data mining issues

Introduction

Data mining states to the withdrawal of hidden analytical information from huge databases. Data mining techniques can produce the benefits of computerization on prevailing software and hardware stages. Tools of Data mining can respond to business queries that traditional methods were too time-consuming to determine.

2.1 Data Mining

Data mining is the act of consequently looking through huge stores of data to discover patterns and trends that go beyond simple analysis. It utilizes refined mathematical algorithms to segment the data and evaluate the probability of upcoming events. Some key terms to know before going into further detail in Data Mining.

Data

Data are the raw facts, figures, numbers, or text that can be processed by a computer. Today, organizations are gathering massive and growing amounts of data in different formats and different databases.

The operational or transactional data contains the day-to-day operation data (such inventory data, on-line shopping data), non-operational data, and metadata i.e. data about data.

Information

The arrangements, relations, or associations among all types of data can deliver information.

Which products are selling when are based upon the analysis of sales transactions by considering a retail idea.

Knowledge

Information can be converted into knowledge.

Supermarket sales information can be analyzed because of marketing efforts to deliver knowledge of consumer purchasing habits.

Data together in large data repositories develop "data tombs". Data tombs are converted into "golden nuggets" of knowledge with the use of data mining tools see in Figure 1. Golden nuggets mean "small but valuable facts". Data mining is also called as mining of knowledge from data, extraction of knowledge, data/arrangement analysis, data -archaeology, and data-dredging.

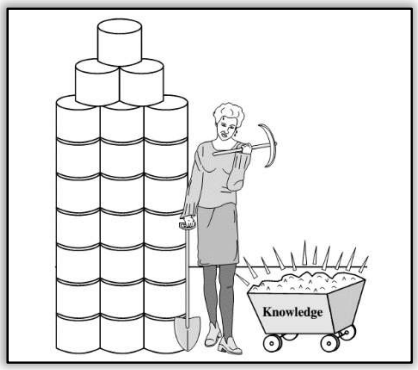


Figure 1: Mining of Data

Why Data Mining?

I am not able to find the data I need (Data is dispersed over the network).The data I have access to is poorly documented (Proper information is missing).I am not able to use the data I have (Unexpected results).Figure 2 shows why data mining is needed.

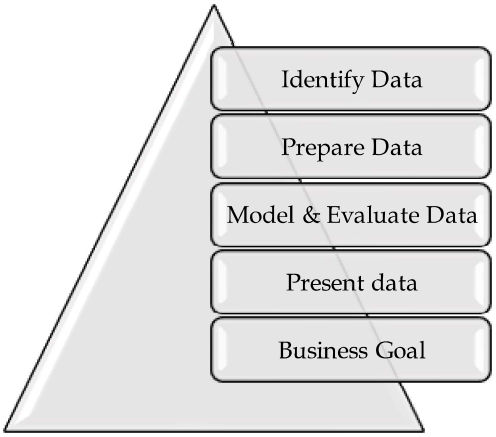


Figure 2: Need for Data Mining

2.2 Process of Knowledge Discovery

Let us have an overview of the steps one by one:
1. Data cleaning: It refers to the removal of inconsistent and noisy data.
2. Integration of data: Multiple information sources are joined during the integration process.

3. Data selection: During the selection process, data relevant to the examination are fetched from the data sets.
4. Data transformation: Data is converted or combined into forms suitable for mining by carrying out summarization or aggregate operations.
5. Data mining: This is a critical cycle where category-wise strategies are useful to extract information strategies.

The following figure displays the KDD process:

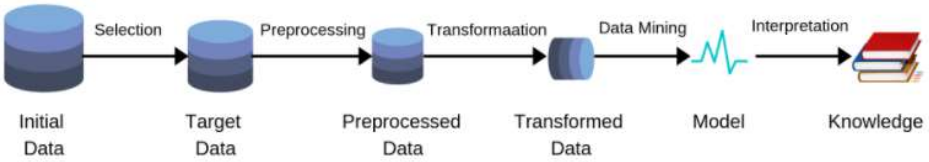


Figure 3:KDD Process

6. Pattern evaluation: This step is used to classify the essentially interesting patterns demonstrating knowledge based on some remarkable measures.
7. Knowledge presentation: Knowledge representation and visualization methods are used to present the mined knowledge to the user.

2.3 Types of Repositories

On a fundamental level, data mining isn't explicit to one sort of media or information. Data mining ought to be material to any sort of data Warehouse. Nonetheless, calculations and approaches may contrast when applied to various kinds of information. For sure, the difficulties introduced by various kinds of information differ altogether. Information mining is being placed into utilization and read for data sets, including social data sets, object-social data sets and article arranged data sets, information stockrooms, conditional data sets, unstructured and semi-organized archives like the World Wide Web, progressed data sets like spatial data sets, interactive media data sets, time-arrangement data sets and literary data sets, and surprisingly level records that are shown in Figure 4:

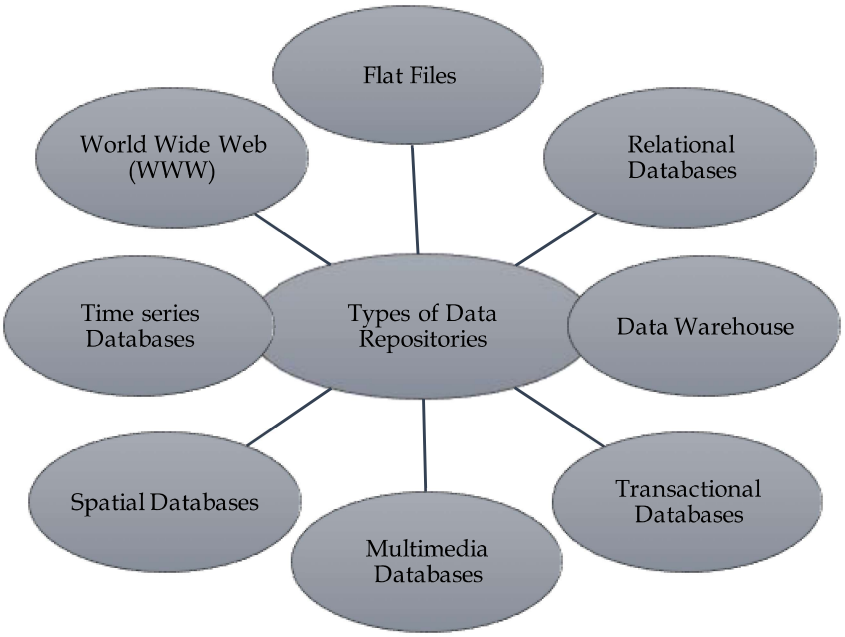


Figure 4: Types of Repositories

Here are some repositories in more detail:

Flat Files: Flat files are well-defined as data collections in text or binary form with an arrangement that can be easily fetched by data mining procedures. Data kept in flat files have no association or pathway between themselves, identically if a traditional database is stored on a flat file, we cannot create any relationship between the tables stored in that database. The data dictionary is used to represent flat files. They are used in a data warehouse to store and conveying information to and from the server, and so forth

Comma Separated Value file.

Relational Databases: Data mining algorithms using traditional databases can be more flexible than other data mining procedures precisely on paper for flat files. Data mining can benefit from SQL for data selection. Operational databases are quite possibly the most ordinarily accessible and most extravagant data archives. Application: Knowledge Extraction, Relational Online Analytical Processing model, etc.

Data Warehouse: A data warehouse center is characterized as the assortment of information incorporated from numerous sources that will inquiries and dynamic and useful for decision making. There are three types of data warehouses: Enterprise data warehouse, Data Mart and Virtual Warehouse. Two approaches can be used to update data in Data Warehouse: Query-driven Approach and Update-driven Approach. A data warehouse is generally demonstrated by a structure that uses multidimensional data that is called a data cube. Each attribute or collection of attributes is represented using dimension in the schema, and each fact table stores the aggregated measures like avg_ sales along with the key values of all the dimension tables. Applications of a data warehouse are commercial decision making, data extraction, etc.

Transactional Databases: A transactional database is a group of data that is prepared by considering different timestamps, dates, etc to characterize transactions in databases. This type of database always ensures consistent data even after updates because in case of any failure the transaction undoes all its updates if some failure occurs until the transaction finally commits its updates. It follows the ACID property of DBMS. Market Basket analysis is an example of typical data-mining analysis with the help of which the retailers they able to identify the frequent patterns by generating associations between the items that are purchased together. Transactional database Applications are banking, distributed systems, Object databases, etc.

Notes

Multimedia databases: This database consists of the data of audio, video, and other related files. Such multi-media information we are not able to store in traditional databases so object-oriented databases are used to store such type of information. Multimedia is characterized by high dimensionality, which makes data mining even more challenging. It may require computer vision, computer visuals, image elucidation, and natural language processing. Various applications of multi-media databases are digital libraries, video-on-demand, news-on-demand, musical databases, etc.

Spatial Databases: A spatial database is a database that is optimized for storing and querying data that represents objects defined in a geometric space. Most spatial databases allow the representation of simple geometric objects such as points, lines, and polygons. It stores data in the form of coordinates, topology, lines, polygons, etc. Applications of spatial databases are Maps, Global positioning, etc.

Time series data: A temporal database is a database that has certain features that support time-sensitive status for entries. Where some databases are considered current databases and only support factual data considered valid at the time of use, a temporal database can establish at what times certain entries are accurate. Data mining in time series databases encompasses the study of developments and associations between valuations of different variables as well as a prediction of trends. Various submissions of time series data are stock market data and logging activities.

World Wide Web: WWW states to World wide web is an assembly of different type of documents and resources which includeaudial, visuals, textual, etc which are recognized by Uniform Resource Locators (URLs) with the help of web browsers, connected by HTML pages, and reachable via the network. It is the most mixed and active data repository. Data in the WWW is prepared in interrelated documents. These documents can be audio, video, text, etc. Online shopping, Job search, Research, studying, etc are the various applications of WWW.

2.4 Data Mining Functionalities

Functionalities of data mining are used to identify the kind of patterns to be found in data mining tasks.

Data mining tasks can be categorized as follows:

Descriptive: It comprises certain knowledge to comprehend what is happening within the data without a prior idea. The collective data features are emphasized in the data set.

count, average, etc.

Predictive: It helps developers to offer unlabeled explanations of attributes. Based on earlier tests, the software assesses the absent features.

Judging from the findings of a patient's medical examinations that is he suffering from any particular disease.

Class/Concept Descriptions:

Classes or definitions can be correlated with results. In simplified, descriptive, and yet accurate ways, it can be helpful to define individual groups and concepts. These class or concept definitions are referred to as class/concept descriptions.

Data Characterization:

This refers to the summary of general characteristics or features of the class that is under the study. For example. To study the characteristics of a software product whose sales increased by 15% two years ago, anyone can collect these types of data related to such products by running SQL queries.

Data Discrimination:

It compares common features of class which is under study. The output of this process can be represented in many forms. Eg., bar charts, curves, and pie charts.

Mining Frequent Patterns, Associations, and Correlations:

Patterns that occur frequently in different transactions are termed as frequent patterns. There are several kinds of frequent patterns, including item-sets, and subsequences. A frequent item-set usually refers to a set of items that appear together frequently in a transactional data set, such as pencil and eraser, which are frequently bought together in stationery stores by many customers. The mining of frequent patterns leads to the detection of motivating relations and associations within the data.

Association Analysis

Assuming, as a marketing manager, you would like to define which items are frequently purchased collectively within the alike transactions.

Buys (X, "computer")=buys(X,"software") [support=1%,confidence=50%]

where X is a variable demonstrating a customer. Confidence=50% means that if a customer buys a computer, there is a 50% chance that he or she will purchase software as well. Support =1% means that 1% of all of the transactions under investigation displayed that computers and software were bought together.

Classification and Prediction

Classification is the method of finding a model that describes and differentiates data classes or concepts. The resultant model is based on the exploration of a set of training data for which the class labels are identified. The model is used to forecast the class label of objects for which the class label is unknown. **Regression analysis** is a statistical procedure that is most often used for numeric prediction, though other methods occur as well.

Example: male or female

2.5 Methods of presenting Derived Model

Different methods like classification (IF-THEN) rules, decision trees, mathematical formulae, or neural networks are available to present the derived model. The explanation of each model is as follows:

IF - THEN Rules

The IF part of the rule where we specify the condition is entitled to rule antecedent or precondition. The THEN fragment of the rule is called rule consequent. The antecedent part of the condition comprises of at least one trait test and these tests are consistently AND ed. The subsequent part comprises of the class forecast.

Example R1: (age = youth) ^ (student = yes))(buys computer = yes)

Decision Tree

A decision tree is a structure that represents the data in a hierarchal form that comprises a root node, branches, and leaf nodes. Every non-leaf node signifies a test condition on an attribute, each branch signifies the different outcomes of a test, and each leaf node represents a class label which the classifier is used for prediction. The top node in the tree is the root node. The following decision tree is for the idea buy_ computer that demonstrates whether a client at an organization is probably going to purchase a PC or not. Each interior hub addresses a test on a characteristic. Each leaf hub addresses a class.

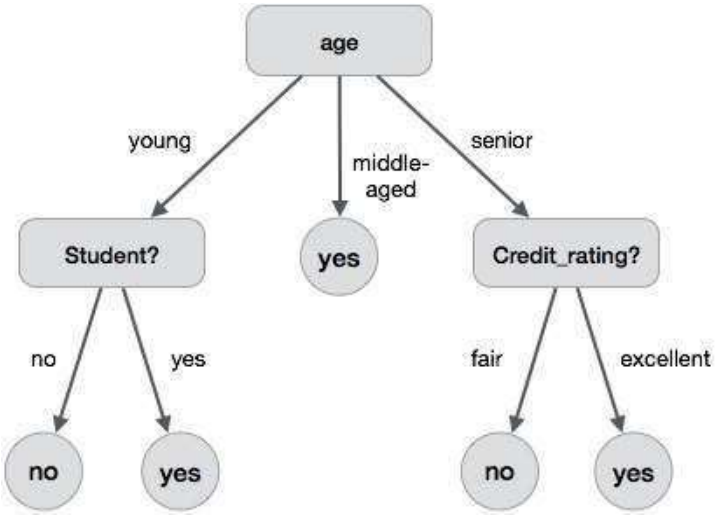


Figure 5: Decision Tree

Cluster Analysis

Unlike classification and prediction, which examine class-labeled data objects, clustering evaluates data objects deprived of referring to a known class label. In general, the class labels do not exist in the training data merely because they are not recognized, to begin with. Clustering can be used to create such labels. The items are clustered or gathered based on the principle of "maximizing the intra-class similarity and minimizing the interclass similarity"

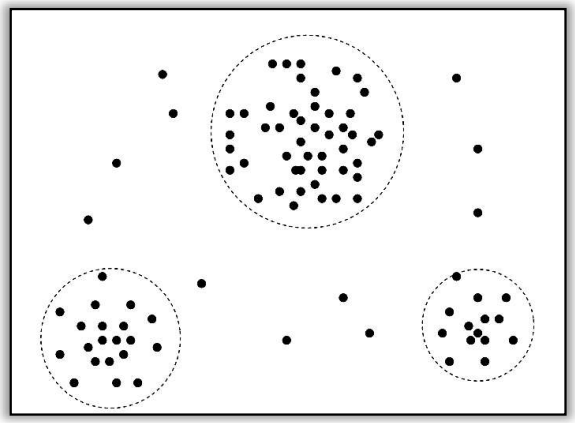


Figure 6: Cluster Analysis

Outlier Analysis

Outlier Analysis is a process that involves identifying the anomalous observation in the dataset. Let us first understand what outliers are. Outliers are nothing but an extreme value that deviates from the other observations in the dataset. Data objects that do not match with the general behavior or model of the data. Most analyses discard outliers as noise or exceptions. Outliers may be detected using statistical tests or using distance measures where objects that are a substantial distance from any other cluster are considered outliers. The examination of outlier data is referred to as outlier mining.

2.6 Data Mining Tasks

Data mining deals with the kind of patterns that can be mined. Based on the kind of data to be mined, there are two categories of tasks involved in Data Mining as shown in Figure 7 –

- Descriptive
- Classification and Prediction

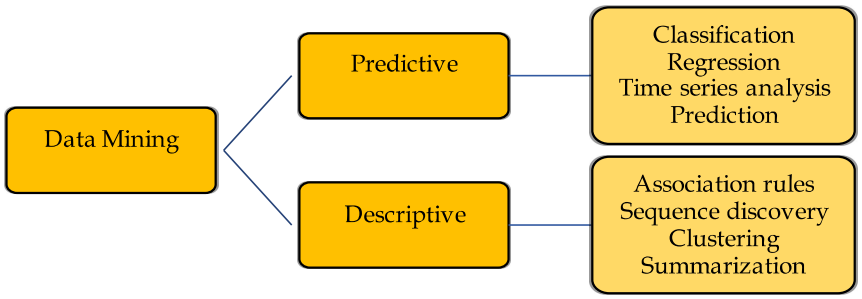


Figure 7: Types of data mining Tasks

Classification

During the learning phase, it constructs a model that classifies the data based on the training set and the values known as class labels and uses these labels for classifying new data.

Data classification is a two-step process:

Learning step (where a classification model is constructed)

Classification step (where the model is used to predict class labels for given data).

In the **learning step** (or training phase), a classification algorithm builds the classifier by analyzing or "learning from" a training set .A tuple, **X**, is represented by an N-dimensional attribute vector,

X ={x1, x2,……..xN}

Each tuple, X, is assumed to belong to a predefined class as determined by another database attribute called the **class label attribute.** The individual tuples making up the training set are referred to as **training tuples** and are randomly sampled from the database under analysis.

pattern recognition

Clustering

Clustering is the unsupervised learning process of building a group the different objects based upon their similarity index. While clustering one thing we need to remember that the cluster quality can be measured as well if and only if the intra-cluster similarity is high and inter-cluster similarity is low. During cluster analysis, we initially partition the data-set into groups based on the data similarity index and then assign the labels to the groups. It is the task of segmenting a dissimilar group into several similar sub-groups or clusters. Similar data items are grouped in one cluster while dissimilar in another cluster.

Bank customer

Time series analysis

Time series analysis comprises methods for analyzing time-series data to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values. The value of an attribute is examined as it varies over time. A time series plot is used to visualize time series.

Stock exchange.

Summarization

Data Summarization is a simple term for a short conclusion of a big theory or a paragraph. This is something where you write the code and, in the end, you declare the final result in the form of summarizing data. Data summarization has great importance in data mining. Abstraction or generalization of data resulting in a smaller set that gives a general overview of data. Alternatively, summary-type information can be derived from data.

Association

Association is a data mining function that discovers the probability of the co-occurrence of items in a collection. The relationships between co-occurring items are expressed as association rules. Association rules are often used to analyze sales transactions. Association is a further prevalent data mining task. Association is also termed as market basket analysis. The goal of the association task is as follows:

- Finding of frequent item-sets
- Finding of association rules.

A frequent item set may look like

{Product = "coke", Product = "Fries", Product = "Squash"}.

Regression

Regression is a data mining function that predicts a number. For example, a regression model could be used to predict the value of a house based on location, number of rooms, lot size, and other factors. A regression task begins with a data set in which the target values are known. The task of regression is related to classification. The key difference is that a probable feature is a continuous number. Regression methods have been extensively studied for centuries on the ground of statistics. Linear regression and logistic regression are the most prevalent regression methods.

2.7 Data Mining Trends

One of the best extensively used methods to fetch data from heterogeneous data sources and establish them for better usage is data mining. Complex algorithms form the source for data mining as they agree for data segmentation to recognize various trends and patterns, detect variations, and predict the chances of various events happening. Some of the key data mining trends are:

- **Application exploration**: The exploration of data mining for businesses continues to expand as e-commerce and e-marketing have become mainstream in the retail industry.
- **Interactive and Scalable data mining methods**: In appearance differently from conventional information examination strategies, data mining should have the option to deal with tremendous measures of information proficiently and, if conceivable, intelligently. Since the measure of information being gathered keeps on expanding quickly, versatile calculations for individual and incorporated data mining capacities become fundamental.
- **Multimedia Data Mining:** It involves the withdrawal of data from different kinds of multimedia devices such as audial, script, hypertext, video, pictures, etc. and the data is transformed into anarithmetic representation in different layouts. This further can be used in clustering and classifications, carrying out similarity checks, and also recognizing associations.
- **Ubiquitous Data Mining:** This technique includes the withdrawal of data from movable devices to get data about different entities. Regardless of having several challenges in this type such as complexity, confidentiality, cost, etc. this method has a lot of prospects to be vast in various trades especially in learning human-computer interactions.
- **Distributed Data Mining**: It involves the withdrawal of an enormous amount of info stored in different business locations or at diverse organizations. Highly refined algorithms are used to fetch data from diverse locations and offer suitable insights and reports based upon the available data.
- **Spatial and Geographic Data Mining:** Type of data mining which includes extracting information from environmental, astronomical, and geographical data which also includes images taken from outer space. This type of data mining can reveal various aspects such as distance and topology which is mainly used in geographic information systems and other navigation applications.
- **Time Series and Sequence Data Mining:** The primary application of this type of data mining is the study of cyclical and seasonal trends. This method is mainly being used by retail companies to access customer's buying patterns and their behaviors.

Data mining trends	Algorithms/ Techniques employed	Data formats	Computing Resources	Prime areas of application
Past	Statistical, machine learning techniques	Numerical data and structured data stored in a traditional database	Evolution of 4G PL and various related techniques	Business
Current	Statistical, machine learning, artificial intelligence, pattern recognition techniques	Heterogeneous data formats include structured, semi-structured, and unstructured data	High-speed networks, high-end storage devices, and parallel distributed computing, etc.	Business, web, medical diagnosis, etc.

Table 1: Past and Current trends in datamining

2.8 Data Mining Issues

Data mining is not an easy task, as the calculations utilized can get exceptionally unpredictable and information isn't generally accessible at one spot. It should be incorporated from different heterogeneous information sources. These variables likewise make a few issues. Here in this instructional exercise, we will examine the significant issues in regards to–

- Mining Methodology and User Interaction
- Performance Issues
- Diverse Data Types Issues

Figure 8 describes the major issues:

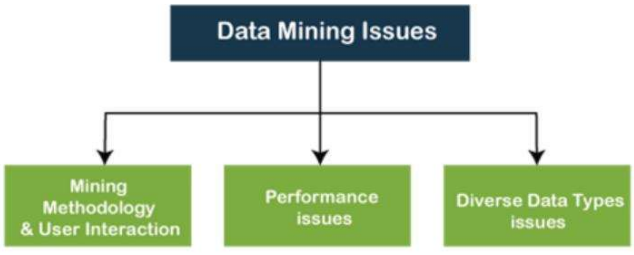


Figure 8: Mining Issues

Mining Methodology and User Interaction Issues

- Mining dissimilar kinds of knowledge in databases – Diverse users may be concerned with different types of knowledge. Consequently, it is essential for data mining to cover an extensive range of knowledge discovery tasks.
- Collaborating mining of knowledge at numerous levels of abstraction – The data mining process requests to be communicating because it allows users to emphasize the search for patterns, providing and cleansing data mining requests based on the reverted results.

- Integration of background knowledge – To monitor the discovery process and to prompt the discovered patterns, contextual knowledge can be used. Background knowledge may be used to rapidly the discovered patterns not only in concise terms but at multiple levels of abstraction.
- Data mining query languages and ad hoc data mining – Data Mining Query language that allows the user to describe ad hoc mining tasks, should be integrated with a data warehouse query language and optimized for efficient and flexible data mining.
- Presentation and visualization of data mining results – Once the patterns are discovered it needs to be expressed in high-level languages, and visual representations. These representations should be easily understandable.
- Handling noisy or incomplete data –Data cleaning methods are a prerequisite to handling the noise and incomplete objects whereas mining the data uniformities. In the absence of data cleaning methods, the accuracy of the discovered patterns will be poor.
- Pattern evaluation –Patterns discovered from the collected data should be motivating because either they signify common knowledge or lack novelty.

Performance Issues

- There can be performance-related issues such as follows –
- Efficiency and scalability of data mining algorithms – To effectively extract the information from a huge amount of data in databases, the data mining algorithm must be efficient and scalable.
- Parallel, distributed, and incremental mining algorithms – The factors such as the huge size of databases, wide distribution of data, and complexity of data mining methods motivate the development of parallel and distributed data mining algorithms. These algorithms divide the data into partitions which are further processed in a parallel fashion. Then the results from the partitions are merged. The incremental algorithms, update databases without mining the data again from scratch.

Diverse Data Types Issues

- Handling of relational and complex types of data – The database may contain complex data objects, multimedia data objects, spatial data, temporal data, etc. One system can't mine all this kind of data.
- Mining information from heterogeneous databases and global information systems – The data is available at different data sources on LAN or WAN. These data sources may be structured, semi-structured, or unstructured. Therefore mining the knowledge from them adds challenges to data mining.

2.9 Ethical Issues

As with many technologies, both positives and negatives lie in the power of data mining. There are, of course, valid arguments to both sides. Here is the positive as well as the negative things about data mining from different perspectives.

Consumers' Point of View

According to the consumers, data mining benefits businesses more than it benefits them. Consumers may benefit from data mining by having companies customized their products and service to fit the consumers' individual needs. However, the consumers' privacy may be lost as a result of data mining.

Data mining is a major way that companies can invade consumers' privacy. Consumers are surprised at how much companies know about their personal lives. For example, companies may know your name, address, birthday, and personal information about your family such as how many children you have. They may also know what medications you take, what kind of music you listen to, and what are your favourite books or movies. The lists go on and on. Consumers are afraid that these companies may misuse their information, or not having enough security to protect their personal information from unauthorized access. For example, the incident about the hackers

in the Ford Motor company case illustrated how insufficient companies are at protecting their customers' personal information. Companies are making profits from their customers' personal data, but they do not want to spend a lot amount of money to design a sophisticated security system to protect that data. At least half of Internet users interviewed by Statistical Research, Inc. claimed that they were very concerned about the misuse of credit card information given online, the selling or sharing of personal information by different websites, and cookies that track consumers' Internet activity.

Data mining can also be used to discriminate against a certain group of people in the population. For example, if through data mining, a certain group of people was determined to carry a high risk for a deathly disease (eg. HIV, cancer), then the insurance company may refuse to sell an insurance policy to them based on this information. The insurance company's action is not only unethical but may also have a severe impact on our health care system as well as the individuals involved. If these high-risk people cannot buy insurance, they may die sooner than expected because they cannot afford to go to the doctor as often as they should. Also, the government may have to step in and provide insurance coverage for those people, thus would drive up the health care costs.

Organizations' Point of View

Data mining is a dream that comes true to businesses because data mining helps enhance their overall operations and discover new patterns that may allow companies to better serve their customers. Through data mining, financial and insurance companies can detect patterns of fraudulent credit card usage, identify behaviour patterns of risk customers, and analyze claims. Data mining would help these companies minimize their risk and increase their profits. Since companies can minimize their risk, they may be able to charge the customers lower interest rates or a lower premium. Companies are saying that data mining is beneficial to everyone because some of the benefits that they obtained through data mining will be passed on to the consumers. When it comes to privacy issues, organizations are saying that they are doing everything they can to protect their customers' personal information. Besides, they only use consumer data for ethical purposes such as marketing, detecting credit card fraud, etc. To ensure that personal information is used ethically, the chief information officer (CIO) Magazine has put together a list of what they call the Six Commandments of Ethical Date Management. The six commandments include:

1. Data is a valuable corporate asset and should be managed as such, like cash, facilities, or any other corporate asset;

2. The CIO is a steward of corporate data and is responsible for managing it over its life cycle (from its generation to its appropriate destruction);

3. The CIO is responsible for controlling access to and use of data, as determined by governmental regulation and corporate policy;

4. The CIO is responsible for preventing inappropriate destruction of data;

5. The CIO is responsible for bringing technological knowledge to the development of data management practices and policies;

6. The CIO should partner with executive peers to develop and execute the organization's data management policies.

Government Point of View

The government is in a dilemma when it comes to data mining practices. On one hand, the government wants to have access to people's personal data so that it can tighten the security system and protect the public from terrorists, but on the other hand, the government wants to protect people's privacy rights. The government recognizes the value of data mining to society, thus wanting the businesses to use the consumers' personal information ethically. According to the government, it is against the law for companies and organizations to trade data they had collected for money or data collected by another organization. To protect people's privacy rights, the government wants to create laws to monitor data-mining practices. However, it is extremely difficult to monitor such disparate resources as servers, databases, and websites. Also, the Internet is global, thus creating tremendous difficulty for the government to enforce the laws.

Society's Point of View

Data mining can aid law enforcers in their process of identify criminal suspects and apprehend these criminals. Data mining can help reduce the amount of time and effort that these law enforcers

have to spend on any one particular case. Thus, allowing them to deal with more problems. Hopefully, this would make the country becomes a safer place. Also, data mining may also help reduce terrorist acts by allowing government officers to identify and locate potential terrorists early. Thus, preventing another incidence likes the World Trade Center tragedy from occurring on American soil.

Data mining can also benefit society by allowing researchers to collect and analyze data more efficiently. For example, it took researchers more than a decade to complete the Human Genome Project. But with data mining, similar projects could be completed in a shorter amount of time. Data mining may be an important tool that aids researchers in their search for new medications, biological agents, or gene therapy that would cure deadly diseases such as cancers or AIDS.

Summary

- The information and knowledge gained can be used for applications ranging from business management, production control, and market analysis, to engineering design and science exploration.
- Data mining can be viewed as a result of the natural evolution of information technology.
- An evolutionary path has been witnessed in the database industry in the development of data collection and database creation, data management, and data analysis functionalities.
- Data mining refers to extracting or "mining" knowledge from large amounts of data. Some other terms like knowledge mining from data, knowledge extraction, data/pattern analysis, data archaeology, and data dredging are also used for data mining.
- Knowledge discovery is a process and consists of an iterative sequence of data cleaning, data integration, data selection data transformation, data mining, pattern evaluation, and knowledge presentation.

Keywords

Data: Data are any facts, numbers, or text that can be processed by a computer.

Data mining: Data mining is the practice of automatically searching large stores of data to discover patterns and trends that go beyond simple analysis.

Data cleaning: To remove noisy and inconsistent data.

Data integration: Multiple data sources may be combined.

Data selection: Data relevant to the analysis task are retrieved from the database.

Data transformation: Where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.

KDD: Many people treat data mining as a synonym for another popularly used term.

Knowledge presentation: Visualisation and knowledge representation techniques are used to present the mined knowledge to the user.

Pattern evaluation: To identify the truly interesting patterns representing knowledge based on some interestingness measures.

Self Assessment

Choose the appropriate answers:

1. KDD stands for

(a) Knowledge Design Database

(b) Knowledge Discovery Database

(c) Knowledge Discovery Design

(d) Knowledge Design Development

2. What is true about data mining?

A. Data Mining is defined as the procedure of extracting information from huge sets of data
B. Data mining also involves other processes such as Data Cleaning, Data Integration, Data Transformation
C. Data mining is the procedure of mining knowledge from data.
D. All of the above.

3. What Data mining **is?**

a) Time variant non-volatile collection of data

b) The actual discovery phase of a knowledge

c) The stage of selecting the right data

d) None of these

4. ……..is not a data mining functionality?

A) Clustering and Analysis

B) Selection and interpretation

C) Classification and regression

D) Characterization and Discrimination

5. .....is the output of KDD

a) Query

b) Useful Information

c) Data

d) Information

6. Which of the following is not belong to data mining?

(A). Knowledge extraction

(B). Data transformation

(C). Data exploration

(D). Data archaeology

7..... databases include video, images, audio, and text media.

8. Time-series databases contain time-related data such as.....

9. A ..... is a set of heuristics and calculations that creates a data mining model from data.

10. ..... type of algorithm finds correlations between different attributes in a dataset.

Answer for Self Assessment

1.	B	2.	D	3.	B	4.	B	5.	B
6.	B	7.	Multimedia	8.	Stock market data	9.	Data mining algorithm	10.	Association algorithms

Notes

Review Questions

1. What is data mining? How does data mining differ from traditional database access?
2. Discuss, in brief, the characterization of data mining algorithms.
3. Briefly explain the various tasks in data mining.
4. Distinguish between the KDD process and data mining.
5. Define data, information, and knowledge.
6. Explain the process of knowledge discovery.
7. Elaborate on the issues of data mining.

Further Readings

A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, 1988.

Alex Berson, *Data Warehousing Data Mining and OLAP*, Tata McGraw Hill, 1997

Alex Berson, Stephen J. Smith, *Data warehousing, Data Mining & OLAP*, Tata McGraw Hill, Publications, 2004.

Alex Freitas and Simon Lavington, *Mining Very Large Databases with Parallel Processing*, Kluwer Academic Publishers, 1998.

J. Ross Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, 1993.

Jiawei Han, Micheline Kamber, *Data Mining – Concepts and Techniques*, Morgan Kaufmann Publishers, First Edition, 2003.

Carlo Vercellis (2011). "*Business Intelligence: Data Mining and Optimization for Decision Making*". John Wiley & Sons.

David Loshin (2012). "*Business Intelligence: The Savvy Manager's Guide*". Newnes.

www.cs.uiuc.edu/~hanj/pdf/ency99.pdf?
www.cs.uiuc.edu/homes/hanj/bk2/toc.pdf

www.stanford.edu/class/stats315b/Readings/DataMining.pdf

Unit03 : Data Warehousing Architecture

CONTENTS

Objectives

After this lecture, you will be able to

- Understand the concept of operational data.
- Learn the architecture and applications of operational data stores.
- Learn the functions and architecture of various process managers.
- Understand the functions and architecture of the query manager.
- Learn the different types of end-user access tools.
- Understand the concept of the architecture model.
- Know the difference in working of 2-tier,3-tier, and 4-tier architecture.

Introduction

Data warehouse provides architectures and tools for business executives to systematically organize, understand, and use their data to make strategic decisions. In the last several years, many firms have spent millions of dollars in building enterprise-wide data warehouses as it is assumed a way to keep customers by learning more about their needs.

In simple terms, a data warehouse refers to a database that is maintained separately from an organization's operational databases. Data warehouse systems allow for the integration of a variety of application systems. They support information processing by providing a solid platform of consolidated, historical data for analysis.

3.1 Operational Data and Datastore

Operational Data is exactly what it sounds like - Data that is produced by your organization's day-to-day operations. The Operational Database is the source of information for the data warehouse.It includes detailed information used to run the day-to-day operations of the business. Operational Database Management Systems also called as OLTP are used to manage dynamic data in real-time.

Operational data stores (ODS) are data repositories that store a snapshot of an organization's current data. This is a highly volatile data repository that is ideally suited for real-time analysis.

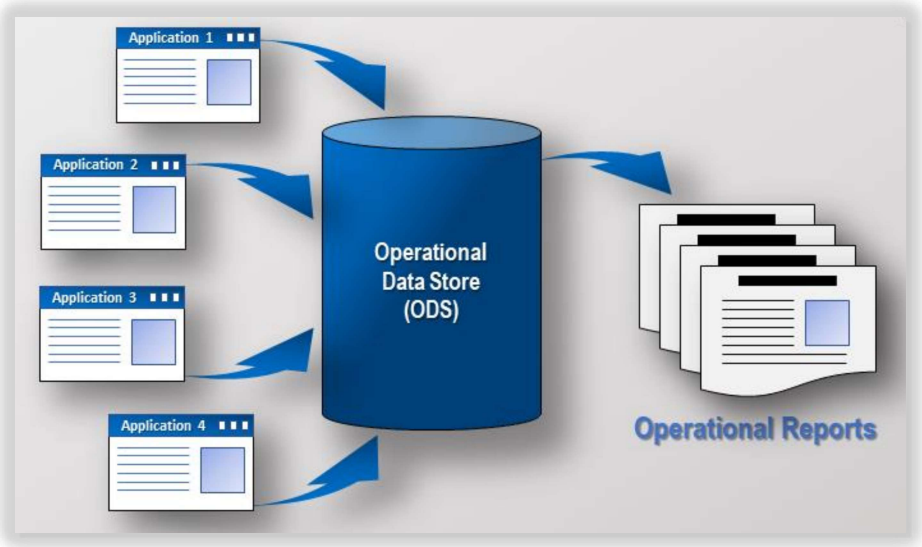


Figure 1: Operational Data Store

An ODS is an integrated database of operational data. Its sources include legacy systems, and it contains current or near-term data. An ODS may contain 30 to 60 days of information, while a data warehouse typically contains years of data. ODSs are used in some data warehouse architectures to provide near-real-time reporting capability if the Data Warehouse's loading time or architecture prevents it from being able to provide near-real-time reporting capability. The ODS then only provides access to the current, fine-grained, and non-aggregated data, which can be queried in an integrated manner without burdening the transactional systems. However, more complex analyses requiring high-volume historical and/or aggregated data are still conducted on the actual data warehouse.

Characteristics of Operational Data Store Systems

The following are the characteristics of Operational Data Stores(ODS):

- ODS systems are highly available and fault-tolerant.
- They occupy less space due to the compression of data and operations.
- ODS systems host configurable, easily accessible, and fast real-time comprehensive data.
- ODS systems are connected to one or more data sources.
- They do not host large amounts of historical data, and thus cannot handle huge data transactions.
- An ODS system makes the creation of back-ups and recovery processes effortlessly since the size of the data is small.

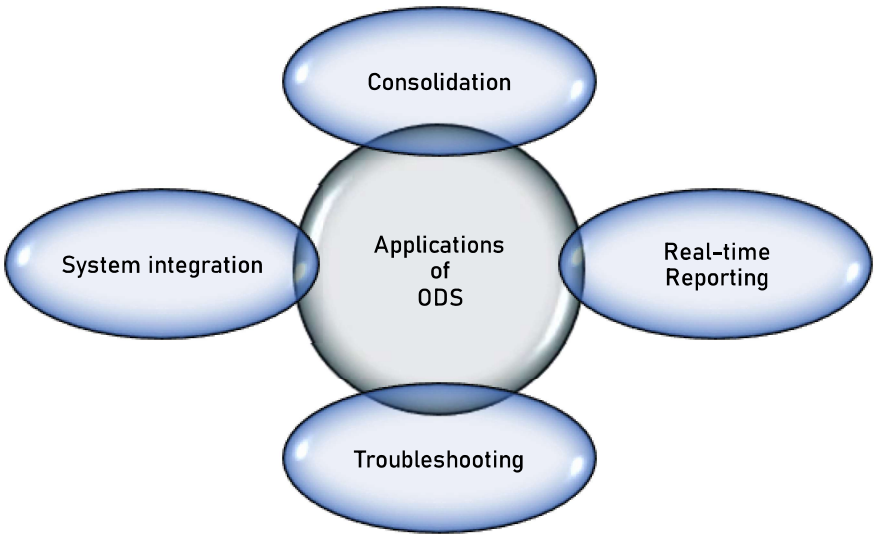


Figure 2: Applications of ODS

- **Consolidation**: The ODS approach can bring together disparate data sources into a single repository. It lacks the benefits of other repositories such as data lakes and data warehouses, but an operational data store has the advantage of being fast and light. ODS can consolidate data from different sources, different systems, or even different locations.
- **Real-time reporting:** An operational data store will generally hold very recent versions of business data. Combined with the right BI tools, businesses can perform real-time BI tasks, such as tracking orders, managing logistics, and monitoring customer activity.
- **Troubleshooting**: The current state view of ODS makes it easier to identify and diagnose issues when they occur. For The ODS will hold both versions of the data, allowing for easy comparison between the two systems. Automated processes can spot these problems and take action.
- **System integration:** Integration requires a continuous flow of data between systems, and ODS can provide the platform for this kind of exchange. It's possible to build business rules on an ODS so that data changes in one system triggers a corresponding action on another system.

A user might create an order on the e-commerce system, which should create a corresponding order on the logistics system. But this might have the wrong details due to an integration error.

Working of ODS

The extraction of data from source databases needs to be efficient, and the quality of records needs to be maintained. Since the data is refreshed generally and frequently, suitable checks are required to ensure the quality of data after each refresh. An ODS is a read-only database other than regular refreshing by the OLTP systems. Customers should not be allowed to update ODS information.

Populating an ODS contains an acquisition phase of extracting, transforming, and loading information from OLTP source systems. This procedure is ETL. Completing populating the database, analyze for anomalies, and testing for performance is essential before an ODS system can go online.

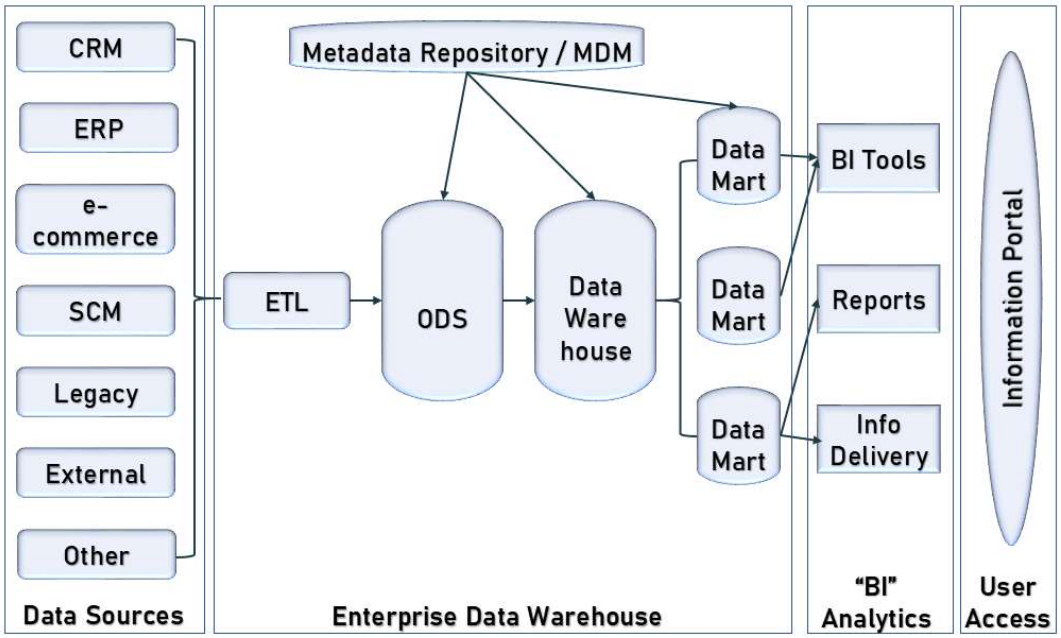


Figure 3: Working of Operational Datastore

Difference between Operational Data Source and Data Warehouse

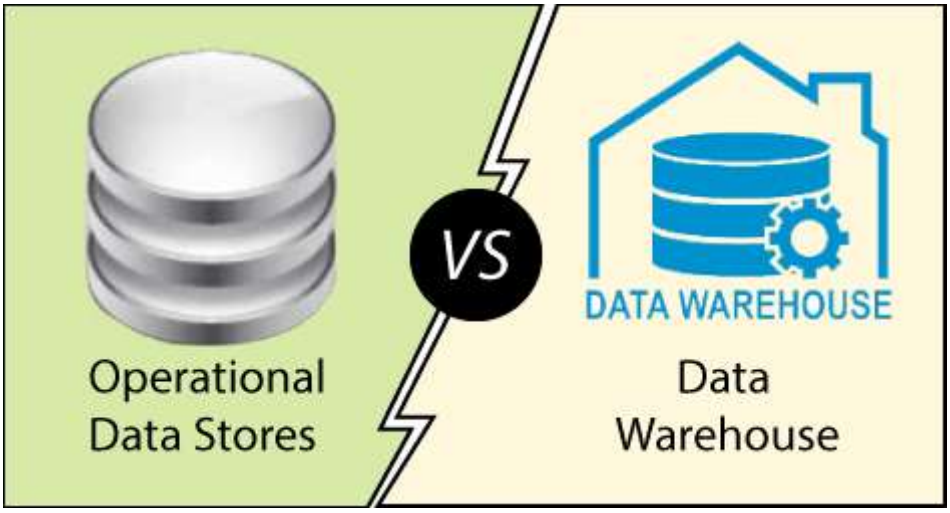


Figure 4: Difference in ODS and Data warehouse

Operational Data Stores	Data Warehouse
ODS means **operational**reporting and supports current or near real-time reporting requirements.	A **data warehouse** is intended for historical and trend analysis, usually reporting on a large volume of data.
An **ODS** consist of only a short window of **data**.	A **data warehouse** includes the entire history of **data**.
It is typically detailed data only.	It contains summarized and detailed data.

It is used for detailed decision-making and operational reporting.	It is used for long-term decision-making and management reporting.
It is used at the operational level.	It is used at the managerial level.
It serves as a conduit for data between operational and analytics systems.	It serves as a repository for cleansed and consolidated data sets.
It is updated often as the transactions system generates new data.	It is usually updated in batch processing mode on a set scheme
It is used at the operational level.	It is used at the managerial level.
It serves as a conduit for data between operational and analytics systems.	It serves as a repository for cleansed and consolidated data sets.
It is updated often as the transactions system generates new data.	It is usually updated in batch processing mode on a set scheme

You know about store room and warehouse. Exactly what the difference between ODS and data warehouse? Explain with the suitable of suitable example.

3.2 Process Manager

Process managers are responsible for maintaining the flow of data both into and out of the data warehouse. There are three different types of process managers:

a. Load manager

b. Warehouse manager

c. Query manager

Data Warehouse Load Manager

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

The architecture of load manager

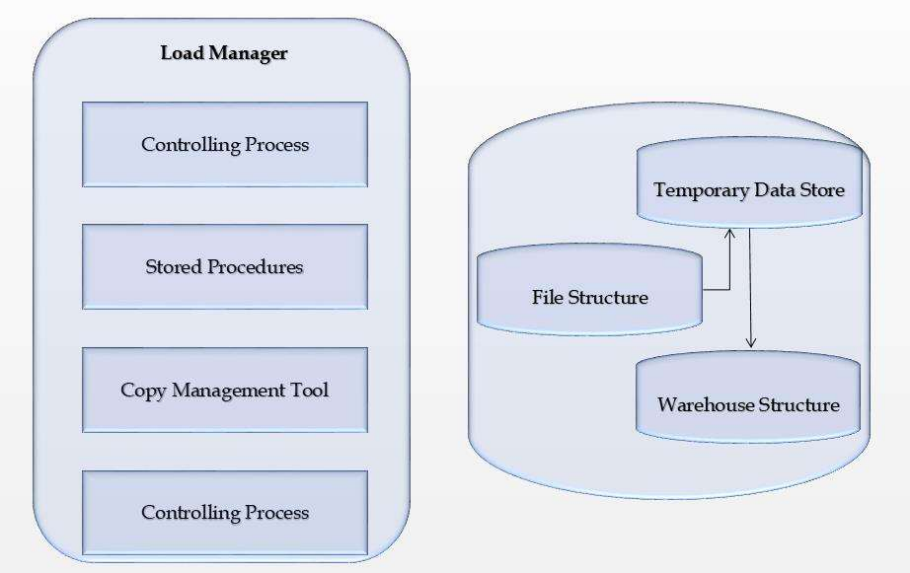


Figure 5: Load Manager Architecture

Functions of Load Manager

- The load manager does perform the following functions: -
- **Extract data from the source system**:The data is extracted from the operational databases or the external information providers.
- Gateways are the application programs that are used to extract data.
- **Fast load the extracted data into a temporary data store:**It is more effective to load the data into a relational database before applying transformations and checks.
- Perform simple transformations into a structure similar to the one in the data warehouse.
- Suppose we are loading the EPOS sales transaction,we need to perform the following checks:

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

➢ Convert all the values to required data types.

Warehouse Manager

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

Components

A warehouse manager includes the following: –

- The controlling process
- Stored procedures or C with SQL
- Backup/Recovery tool
- SQL scripts

The architecture of Warehouse Manager

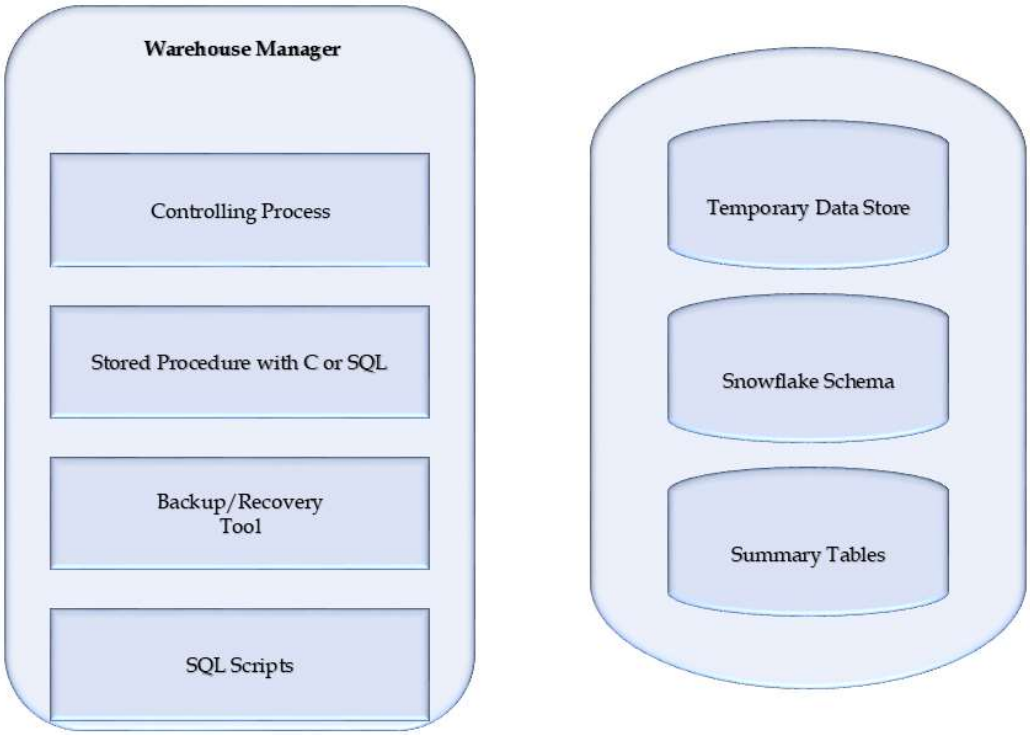


Figure 6: Warehouse Manager Architecture

Functions of Warehouse Manager

- Analyzes the data to perform consistency and referential integrity checks.
- Creates indexes, business views, partition views against the base data.
- Generates new aggregations and updates the existing aggregations.
- Generates normalizations.
- Transforms and merges the source data of the temporary store into the published data warehouse.
- Backs up the data in the data warehouse.
- Archives the data that has reached the end of its captured life.

3.3 Query Manager

The query manager is responsible for directing the queries to suitable tables. By directing the queries to appropriate tables, it speeds up the query request and response process. Besides, the query manager is responsible for scheduling the execution of the queries posted by the user. A query manager includes the following components :

- Query redirection via C tool or RDBMS
- Stored procedures
- Query management tool
- Query scheduling via C tool or RDBMS
- Query scheduling via third-party software.

Functions of Query Manager

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

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

- It stores query profiles to allow the warehouse manager to determine which indexes and aggregations are appropriate.
- Performs operations associated with management of user queries
- Constructed using vendor end-user data access tools, data warehouse monitoring tools, database facilities, and custom-built programs
- Complexity determined by facilities provided by end-user access tools and database
- Operations:
  - Directing queries to appropriate tables
  - Scheduling execution of queries

3.4 End-user Access Tools

There are five main groups of access tools:–

- Data reporting and query tools
- Application development tools
- Executive information system (EIS) tools
- Online analytical processing (OLAP) tools
- Data mining tools

Data Reporting and Query Tools

- The principal purpose of Data Warehousing is to provide information to business users for strategic decision-making. Different types of users engage in different types of decision

  support activities and therefore require different types of tools.When it comes time to start creating reports out of the data in your warehouse and to start making decisions with this data you are going to need to have a good query tool.Managed query tools shield end users from the Complexities of SQL and database structure by inserting a meta-layer between the user and the database.

Features for Query Tools

- **Cross-Browsing of Dimension Attributes**:A real dimension table, such as a list of all of your products or customers, takes the form of a large dimension table with many, many attributes (fields). Cross-browsing, on the other hand, refers to the capability of a query tool to present the valid values of the product brand, subject to a constraint elsewhere on that dimension table.
- **Open Aggregate Navigation:**Aggregate navigation is the ability to automatically choose pre-stored summaries, or aggregates, in the course of processing a user's SQL requests. Aggregate navigation must be performed silently and anonymously
- **Multipass SQL:**Breaking a single complex request into several small requests is called multipass SQL. It also allows drilling across several conformed data marts in different databases, in which the processing of a single galactic SQL statement would otherwise be impossible.
- **Semi-Additive Summations:**Semi Additive measures are values that you can summarize across any related dimension except time.Stock levels however are semi-additive; if you had 100 in stock yesterday, and 50 in stock today, you're total stock is 50, not 150. It doesn't make sense to add up the measures over time, you need to find the most recent value.
- **Show Me What Is Important:**Your query tools must help you automatically sift through the data to show you only what is important. At the low end, you simply need to show data rows in your reports that meet certain threshold criteria.

- **Behavioral Studies:**An interesting class of applications involves taking the results of a previous report or set of reports and then using these results over and over again at a later time.

Executive Information System (EIS) Tools

An Executive information system, also known as an Executive support system, is a type of management support system that facilitates and supports senior executive information and decision-making needs. It provides easy access to internal and external information relevant to organizational goals.

Reporting Tools

Reporting Tools can be divided into two categories:

Production Reporting Tools: These tools let companies generate regular operational reports or support high-volume batch jobs, such as calculating and printing paychecks. Production Reporting Tools include 3GLs such as COBOL, specialized 4GL, such as Information Builders, Inc's Focus, and high-end client/ server tools such as MITI's SQR.

Desktop Report Writers: Report writers are inexpensive desktop tools designed for end-users. A product such as Crystal Reports, lets users design and run reports without having to rely on the IS Department.

OLAP Tools

Online Analytical Processing Server (OLAP) is based on the multidimensional data model. It allows managers, and analysts to get an insight into the information through fast, consistent, and interactive access to information.

Classification of OLAP Tools

MOLAP: It stands for Multidimensional Online Analytical Processing. It stores data in multidimensional arrays and requires pre-computation and storage of information in the cube.Some of the tools for that are:

- **IBM Cognos:** It provides tools for reporting, analysis, monitoring of events and metrics.
- **SAP NetWeaver BW:** It is known as SAP NetWeaver Business Warehouse. Just like IBM Cognos, It also delivers reporting, analysis, and interpretation of business data. It runs on Industry-standard RDBMS and SAP's HANA in-memory DBMS.
- **Microsoft Analysis Services:** Microsoft Analysis Services is used by Organisations to make sense of data that is spread out in multiple databases or it is spread out in a discrete form.
- **MicroStrategy Intelligence Server**: MicroStrategy Intelligence Server helps the business to standardize themselves on a single open platform which in return will reduce their maintenance and operating cost.

MOLAP products are the commercial Hyperion Ebasse (www.hyperion.com) and the Applix TM1 (www.applix.com), as well as Palo (www.opensourceolap.org), which is an open-source product.

ROLAP: The 'R' in ROLAP stands for Relational. So, the full form of ROLAP becomes Relational Online Analytical Processing. The salient feature of ROLAP is that the data is stored in relational databases. Some of the top ROLAP is as follows:

- IBM Cognos
- SAP NetWeaver BW
- Microsoft Analysis Services
- Essbase
- Jedox OLAP Server

- SAS OLAP Server

ROLAP engines include the commercial IBM Informix Metacube (www.ibm.com) and the Micro-strategy DSS server (www.microstrategy.com), as well as the open-source product Mondrian (mondrian.sourceforge.net).

HOLAP: It stands for Hybrid Online Analytical Processing. So, HOLAP bridges the shortcomings of both MOLAP and ROLAP by combining their capabilities. Now how does it combine? It combines data by dividing data of the database between relational and specialized storage. Some of the top HOLAP is as follows:

Example IBM Cognos

SAP NetWeaver BW

Data Mining Tools

Data Mining tools have the objective of discovering patterns/trends/groupings among large sets of data and transforming data into more refined information. They Provide insights into corporate data that are not easily discerned with a managed query or OLAP tools. Tools usea variety of statistical and AI algorithm to analyze the correlation of variables in data. It investigates interesting patterns and their relationship

Figure 7: Data Mining Tools

Application Development Tools

Sometimes built-in graphical and analytical tools do not satisfy the analytical needs of an organization. In such cases, custom reports are developed using Application development tools.

3.5 Types of data in Data Warehouse

The data within the specific warehouse itself has a particular architecture with the emphasis on various levels of summarization, as shown in the figure:

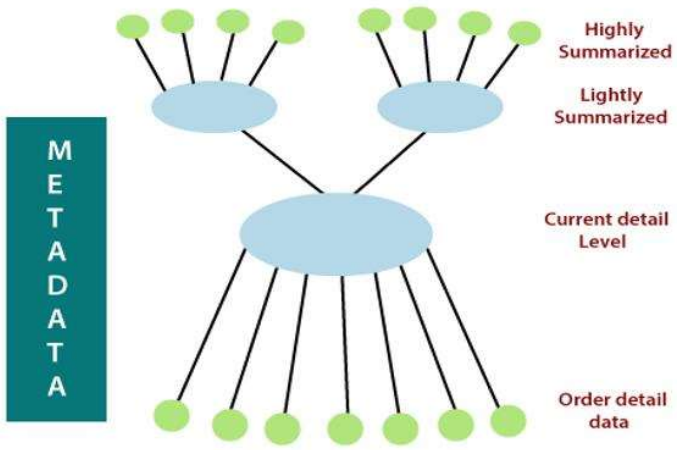


Figure 8:Structure of data inside Data Warehouse

Four types of data arestored in a data warehouse:

- older detail data,
- current detail data,
- lightly summarized data
- highly summarized data.

Older Detail Data:Older detail data represents data that is not very recent, maybe as old as ten years or longer.It is voluminous and most frequently stored on mass storage such as tape, although more expensive disk storage may be used. Its level of detail is consistent with current detail data

but due to its long time horizon is typically migrated to a "less-expensive" alternate storage medium.

Current Detail Data: This data represents data of a recent nature and always has a shorter time horizon than older detail data. Although it can be voluminous, it is almost always stored on a disk to permit faster access. The current detail record is central in importance as it:

- Reflects the most current happenings, which are commonly the most stimulating.
- It is numerous as it is saved at the lowest method of the Granularity.
- It is always (almost) saved on disk storage, which is fast to access but expensive and difficult to manage.

Lightly summarized data: Lightly summarized data represents data distilled from current detail data. It is summarized according to some unit of time and always resides on disk. This data extract from the low level of detail found at the current, detailed level and usually is stored on disk storage. When building the data warehouse have to remember what unit of time is the summarization done over and also the components or what attributes the summarized data will contain.

Highly summarized data is compact and directly available and can even be found outside the warehouse. Highly summarized data represents data distilled from lightly summarized data. It is always compact and easily accessible and resides on a disk.A final component of the data warehouse is that of metadata. Metadata is best described as data about data. It provides information about the structure of a data warehouse as well as the various algorithms used in data summarizations. It provides a descriptive view, or "blueprint", of how data is mapped from the operational level to the data warehouse.

3.6 Data Archiving

Data warehouse archiving is the process of moving data that is not likely needed to conduct business operations from the data warehouse to a medium where it can be stored for long-term retention and retrieval if needed.Archive data consists of older data that remains important to the organization or must be retained for future reference.Data archives are indexed and have search capabilities, so files can be located and retrieved.Some archive systems treat archive data as read-only to protect it from modification, while other data archiving products enable writes, as well as reads.

WORM (write once, read many) technologies use media that is not rewritable.

Backup data

A data warehouse is a complex system and it contains a huge volume of data. Therefore it is important to back up all the data so that it becomes available for recovery in the future as per requirement.

Backup Terminologies

- **Complete backup** − It backs up the entire database at the same time. This backup includes all the database files, control files, and journal files.
- **Partial backup** − As the name suggests, it does not create a complete backup of the database. Partial backup is very useful in large databases because they allow a strategy whereby various parts of the database are backed up in a round-robin fashion on a day-to-day basis so that the whole database is backed up effectively once a week.
- **Cold backup** − Cold backup is taken while the database is completely shut down. In a multi-instance environment, all the instances should be shut down.
- **Hot backup** − Hot backup is taken when the database engine is up and running. The requirements of hot backup vary from RDBMS to RDBMS.
- **Online backup** − It is quite similar to hot backup.

	Backup	Archiving
What is it?	Protection for mission critical systems and live data	Searchable records of inactive data in a "steady state"
Why use it?	Recovery – backup restores systems after data loss, interruption, or disaster	Searching – allows interrogation of data for regulatory inspections and data investigations
What does it contain?	Several snapshots of the live system(s) captured on a time basis	One single repository of historical data indexed and quickly searchable

3.7 Metadata

- Metadata is data about data that defines the data warehouse. It is used for building, maintaining, and managing the data warehouse.In the Data Warehouse Architecture, meta-data plays an important role as it specifies the source, usage, values, and features of data warehouse data. It also defines how data can be changed and processed. For example, a line in the sales database may contain:

4030 KJ732 299.90

This is meaningless data until we consult the Meta that tell us it was

- Model number: 4030

- Sales Agent ID: KJ732
- Total sales amount of $299.90
- Therefore, Meta Data are essential ingredients in the transformation of data into knowledge.

Metadata is the final element of the data warehouses and is really of various dimensions in which it is not the same as file drawn from the operational data, but it is used as:-

- A directory to help the DSS investigator locate the items of the data warehouse.
- A guide to the mapping of record as the data is changed from the operational data to the data warehouse environment.
- A guide to the method used for summarization between the current, accurate data and the lightly summarized information and the highly summarized data, etc.

Metadata helps to answer the following questions:

- What tables, attributes, and keys does the Data Warehouse contain?
- Where did the data come from?
- How many times do data get reloaded?
- What transformations were applied with cleansing?

Categories of metadata

- **Technical MetaData**: This kind of Metadata contains information about the warehouse which is used by Data warehouse designers and administrators.
- **Business MetaData:** This kind of Metadata contains detail that gives end-users a way easy to understand the information stored in the data warehouse.

Data warehouse metadata include table and column names, their detailed descriptions, their connection to business meaningful names, the most recent data load date, the business meaning of a data item and the number of users that are logged in currently

3.8 Architecture Model

Architecture model is a method of defining the overall architecture of data communication, processing, and presentation that exist for end-clients computing within the enterprise. Applications gather detailed data from day-to-day operations. A data warehouse architecture is a method of defining the overall architecture of data communication processing and presentation that exists for end-clients computing within the enterprise. Each data warehouse is different, but all are characterized by standard vital components.

Types of Data Warehouse Architectures

Two-Tier Architecture

The requirement for separation plays an essential role in defining the two-tier architecture for a data warehouse system, as shown in fig:

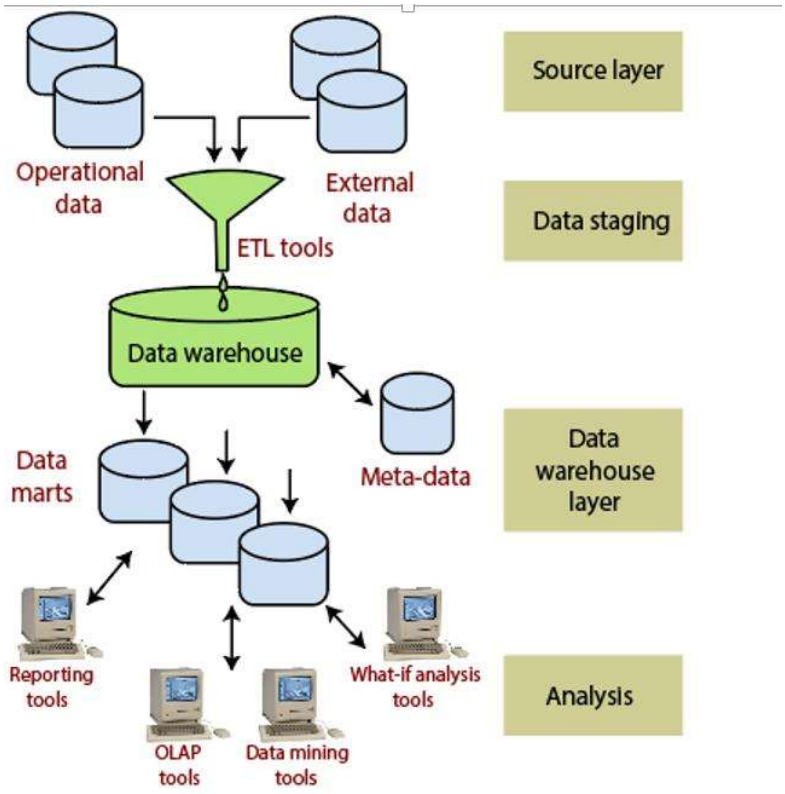


Figure 9:Two -Tier Data Warehouse Architecture

Although it is typically called two-layer architecture to highlight a separation between physically available sources and data warehouses, in fact, consists of four subsequent data flow stages:

Source layer: A data warehouse system uses a heterogeneous source of data. That data is stored initially to corporate relational databases or legacy databases, or it may come from an information system outside the corporate walls.

Data Staging: The data stored to the source should be extracted, cleansed to remove inconsistencies and fill gaps, and integrated to merge heterogeneous sources into one standard schema. The so-

named **Extraction**, **Transformation**, and **Loading Tools (ETL)** can combine heterogeneous schemata, extract, transform, cleanse, validate, filter, and load source data into a data warehouse.

Data Warehouse layer: Information is saved to one logically centralized individual repository: a data warehouse. The data warehouses can be directly accessed, but they can also be used as a source for creating data marts, which partially replicate data warehouse contents and are designed for specific enterprise departments. Meta-data repositories store information on sources, access procedures, data staging, users, data mart schema, and so on.

Analysis: In this layer, integrated data is efficiently, and flexible accessed to issue reports, dynamically analyze information, and simulate hypothetical business scenarios. It should feature aggregate information navigators, complex query optimizers, and customer-friendly GUIs.

Three-Tier Architecture

The three-tier architecture consists of the source layer (containing multiple source systems), the reconciled layer, and the data warehouse layer (containing both data warehouses and data marts). The reconciled layer sits between the source data and data warehouse.

The main advantage of the **reconciled layer** is that it creates a standard reference data model for a whole enterprise. At the same time, it separates the problems of source data extraction and integration from those of the data warehouse population. In some cases, the **reconciled layer** is also directly used to accomplish better some operational tasks, such as producing daily reports that

cannot be satisfactorily prepared using the corporate applications or generating data flows to feed external processes periodically to benefit from cleaning and integration.

This architecture is especially useful for extensive, enterprise-wide systems. A disadvantage of this structure is the extra file storage space used through the extra redundant reconciled layer. It also makes the analytical tools a little further away from being real-time.

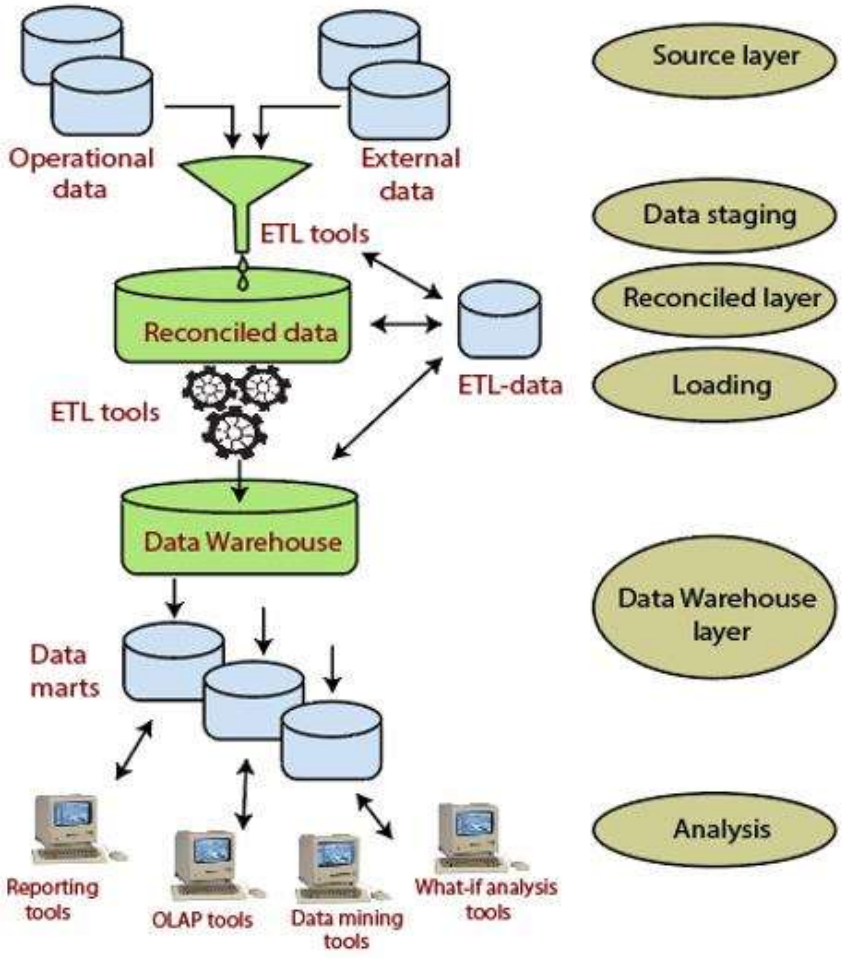


Figure 10: Three Tier Data Warehouse Architecture

4-Tier architecture

User: At the end-**user layer**, **data** in the ODS, **data warehouse**, and **data** marts can be accessed by using a variety of tools such as query and reporting tools, **data** visualization tools, and analytical applications.

Presentation layer: Its functions contain receiving data inputted, interpreting users' instructions, and sending requests to the data services layer, and displaying the data obtained from the data services layer to users by the way they can understand. It closest to users and provide an interactive operation interface.

Discuss various factors play vital role to design a good data warehouse.

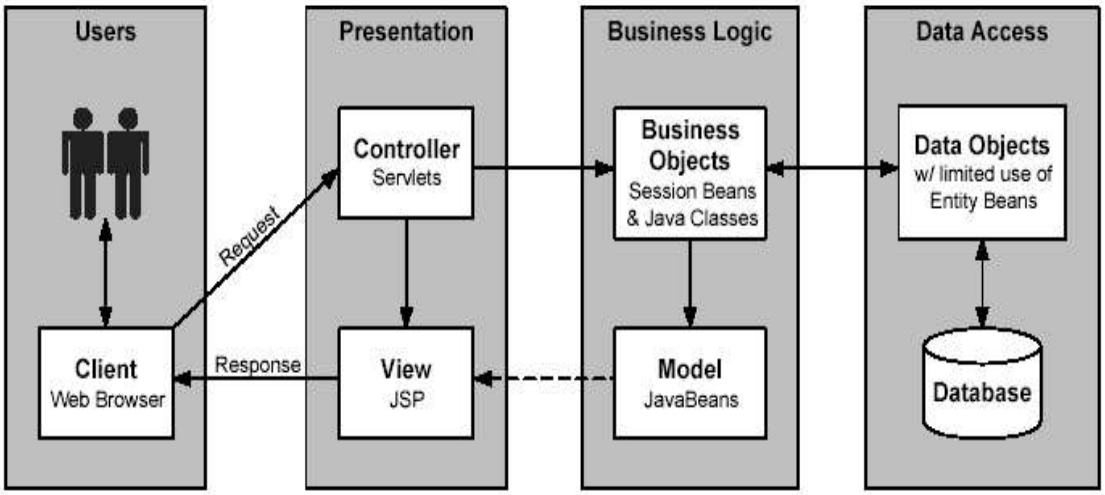


Figure 11:Four Tier Data Warehouse Architecture

Business logic:It is located between the PL and data access layer playing a connecting role in the data exchange.The layer's concerns are focused primarily on the development of business rules, business processes, and business needs related to the system.It's also known as the domain layer.

Data Access: It is located in the innermost layer that implements persistence logic and responsible for access to the database.Operations on the data contain finding, adding, deleting, modifying, etc. This level works independently, without relying on other layers.DAL extracts the appropriate data from the database and passes the data to the upper.

Summary

- OLAP servers may adopt a relational OLAP (ROLAP), a multidimensional OLAP (MOLAP), or a hybrid OLAP (HOLAP) implementation.
- Data warehousing is the consolidation of data from disparate data sources into a single target database to be utilized for analysis and reporting purposes.
- The primary goal of data warehousing is to analyze the data for business intelligence ⌘⌘purposes. For example, an insurance company might create a data warehouse to capture policy data for catastrophe exposure.
- The data is sourced from front-end systems that capture the policy information into the ⌘⌘data warehouse.
- The data might then be analyzed to identify windstorm exposures in coastal areas prone to ⌘⌘hurricanes and determine whether the insurance company is overexposed.
- The goal is to utilize the existing information to make accurate business decisions.

Keywords

Data Sources: Data sources refer to any electronic repository of information that contains data of interest for management use or analytics.

Data Warehouse Architecture: It is a description of the elements and services of the warehouse, with details showing how the components will fit together and how the system will grow over time.

Data Warehouse: It is a relational database that is designed for query and analysis rather than for transaction processing.

Job Control: This includes job definition, job scheduling (time and event), monitoring, logging, exception handling, error handling, and notification.

Metadata: Metadata, or "data about data", is used not only to inform operators and users of the data warehouse about its status and the information held within the data warehouse

Self Assesment

1) OLTP stands for

(a) On Line Transactional Processing

(b) On Link Transactional Processing

(c) On Line Transnations Processing

(d) On Line Transactional Program

2) ROLAP stands for

(a) Relational On Line Transition Processing

(b) Relative On Line Transactional Processing

(c) Relational On Line Transactional Processing

(d) Relational On Line Transactional Program

3) The data from the operational environment enter …….. of data warehouse.

A) Current detail data

B) Older detail data

C) Lightly Summarized data

D) Highly summarized data

4) ……..are designed to overcome any limitations placed on the warehouse by the nature of the relational data model.

A) Operational database

B) Relational database

C) Multidimensional database

D) Data repository

5) Data warehouse contains _____data that is seldom found in the operational environment Select one:

a)informational

b)normalized

c)denormalized

d)summary

6) _____ are numeric measurements or values that represent a specific business aspect or activity Select one:

a)Dimensions

b)Schemas

c)Facts

d)Tables

7) _____describes the data contained in the data warehouse Select one:

a)Relational data

b)Operational data

c)Informational data

d)Meta data

8) Dimensionality refers to Select one:

a)Cardinality of key values in a star schema

b)The data that describes the transactions in the fact table

c)The level of detail of data that is held in the fact table

d)The level of detail of data that is held in the dimension table

9) Business Intelligence and data warehousing is used for .....

A) Forecasting

B) Data Mining

C) Analysis of large volumes of product sales data

D) All of the above

10) Data warehouse is a kind of analytical tool used in .....

11) An operational system is which of the following?

a)A system that is used to run the business in real time and is based on historical data.

b)A system that is used to run the business in real time and is based on current data.

c)A system that is used to support decision making and is based on current data.

d)A system that is used to support decision making and is based on historical data.

12) Decision support systems (DSS) is

a)A family of relational database management systems marketed by IBM

b)Interactive systems that enable decision makers to use databases and models on a computer in order to solve ill-structured problems

c)It consists of nodes and branches starting from a single root node. Each node represents a test, or decision

d)None of these

13) ..... stores the data based on the already familiar relational DBMS technology.

14) Which of the following features usually applies to data in a data warehouse?

A.Data are often deleted

B.Most applications consist of transactions

C.Data are rarely deleted

D.Relatively few records are processed by applications

15) The following is true of three-tier data warehouses:

a)Once created, the data marts will keep on being updated from the data warehouse at periodic times

b)Once created, the data marts will directly receive their new data from the operational databases

c)The data marts are different groups of tables in the data warehouse

d)A data mart becomes a data warehouse when it reaches a critical size

Review Questions

1. What is a data warehouse? How is it better than traditional information-gathering techniques?

2. Describe the data warehouse environment.

3. List and explain the different layers in the data warehouse architecture.

4. Differentiate between ROLAP and MOLAP.

5.Describe three-tier data warehouse architecture in detail.

Answers: Self Assesment

1.	A	2.	C	3.	A	4.	C	5.	D
6.	C	7.	D	8.	B	9.	D	10.	Bussiness Area
11.	B	12.	B	13.	ROLAP	14.	C	15.	A

Further Readings

A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.

Alex Berson, *Data Warehousing Data Mining and OLAP*, Tata McGraw Hill, 1997

Alex Berson, Stephen J. Smith, *Data warehousing, Data Mining & OLAP*, Tata McGraw Hill, Publications, 2004.

Alex Freitas and Simon Lavington, *Mining Very Large Databases with Parallel Processing*, Kluwer Academic Publishers, 1998.

J. Ross Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, 1993.

Jiawei Han, MichelineKamber, *Data Mining – Concepts and Techniques*, Morgan Kaufmann Publishers, First Edition, 2003.

Matthias Jarke, Maurizio Lenzerini, YannisVassiliou, PanosVassiliadis, *Fundamentals of Data Warehouses*, Publisher: Springer

Michael Berry and Gordon Linoff, *Data Mining Techniques (For Marketing, Sales, and Customer Support)*, John Wiley & Sons, 1997.

Michael J. A. Berry, Gordon S Linoff, *Data Mining Techniques*, Wiley Publishing Inc, Second Edition, 2004.

www.en.wikipedia.org

www.web-source.net

www.webopedia.com

Unit04: Installation and development environment overview

CONTENT

Objectives

- Understand the use of Rapid Miner.
- Know the different Data Mining products.
- Learn the steps for installing Rapid Miner.
- Understand what WEKA is.
- Learn the installation process of WEKA.

Introduction

Rapid Miner provides an environment for machine learning and data mining processes. It follows a modular operator concept which allows the design of complex nested operator chains for a huge number of learning problems. Allows for the data handling to be transparent to the operators.

Weka is an open-source tool designed and developed by the scientists/researchers at the University of Waikato, New Zealand. WEKA stands for Waikato Environment for Knowledge Analysis. It is developed by the international scientific community and distributed under the free GNU GPL license. It provides a lot of tools for data preprocessing, classification, clustering, regression analysis, association rule creation, feature extraction, and data visualization. It is a powerful tool that supports the development of new algorithms in machine learning.

4.1 RapidMiner

The idea behind the Rapid Mining tool is to create one place for everything. RapidMiner is an integrated enterprise artificial intelligence framework that offers AI solutions to positively impact businesses. It is used as a data science software platform for data extraction, data mining, deep learning, machine learning, and predictive analytics. It is widely used in many business and commercial applications as well as in various other fields such as research, training, education, rapid prototyping, and application development. All major machine learning processes such as data preparation, model validation, results from visualization, and optimization can be carried out by using RapidMiner.

Facilities of RapidMiner

1. Rapid Miner provides its collection of datasets but it also provides options to set up a database in the cloud for storing large amounts of data. You can store and load the data from Hadoop, Cloud, RDBMS, NoSQL, etc. Apart from this, you can load your CSV data very easily and start using it as well.

2. The standard implementation of procedures like data cleaning, visualization, pre-processing can be done with drag and drop options without having to write even a single line of code.
3. Rapid Miner provides a wide range of machine learning algorithms in classification, clustering, and regression as well. You can also train optimal deep learning algorithms like Gradient Boost, XGBoost, etc. Not only this, but the tool also provides the ability to perform pruning and tuning.
4. Finally, to bind everything together, you can easily deploy your machine learning models to the web or mobiles through this platform. You just need to create user interfaces to collect real-time data and run it on the trained model to serve a task.

RapidMiner Products

The following are the various products of RapidMiner:

- RapidMiner Studio
- RapidMiner Auto Model
- RapidMiner Auto Model

RapidMiner Studio

- With RapidMiner Studio, one can access, load, and analyze both traditional structured data and unstructured data like text, images, and media.
- It can also extract information from these types of data and transform unstructured data into structured.

RapidMiner Auto Model

- Auto Model is an advanced version of RapidMiner Studio that increments the process of building and validating data models.
- Majorly three kinds of problems can be resolved with Auto Model namely prediction, clustering, and outliers.

RapidMiner Turbo Prep

- Data preparation is time-consuming and RapidMiner Turbo Prep is designed to make the preparation of data much easier.
- It provides a user interface where your data is always visible front and center, where you can make changes step-by-step and instantly see the results, with a wide range of supporting functions to prepare the data for model-building or presentation.

4.2 Downloading of Installation of RapidMiner

RapidMiner provides a data science platform to help you drive real business impact. Choose the solution based on your preferences.

Where to get it? To install RapidMiner Go to:

https://rapidminer.com/get-started/

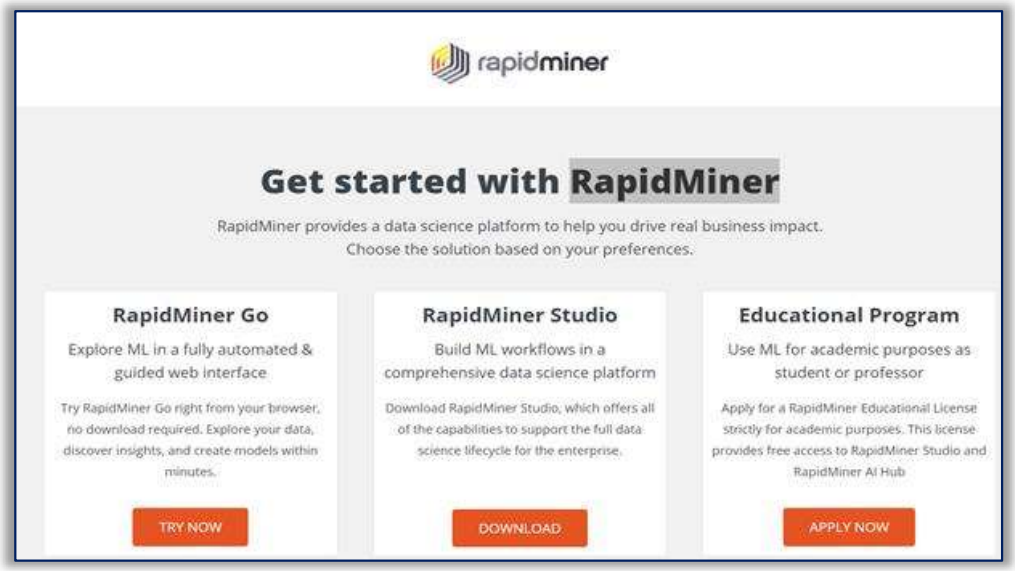


Figure 1: Get Started with Rapid Miner

RapidMiner Installation

And proceed to click on the version of RapidMiner Studio that fits your system:

- Windows 32bits
- Windows 64bits
- Mac OS 10.8+
- Linux

Figure 2: Operating System Selection

Installing RapidMiner Studio on Windows

Follow these simple instructions to run the launcher that installs RapidMiner Studio on Windows.

- Double-click the downloaded file (for example, rapidminer-studio-<version>-win64-install.exe).
- If prompted, allow the program to make changes to your computer. The RapidMiner Studio Setup Wizard appears. Click Next to continue.
- Read the terms of the license agreement and click I Agree to continue.

- Select a destination folder (or leave the default). Please ensure that the folder path does not contain + or % characters. By clicking Install, the wizard extracts and installs RapidMiner Studio. When the installation completes, click Next and then Finish to close the wizard and start RapidMiner Studio.
- Read the terms of the license agreement and click I Accept to continue.

  The download will start automatically. When it finishes, click "RUN" so that the installation can begin.

Agree with Terms and Select Destination

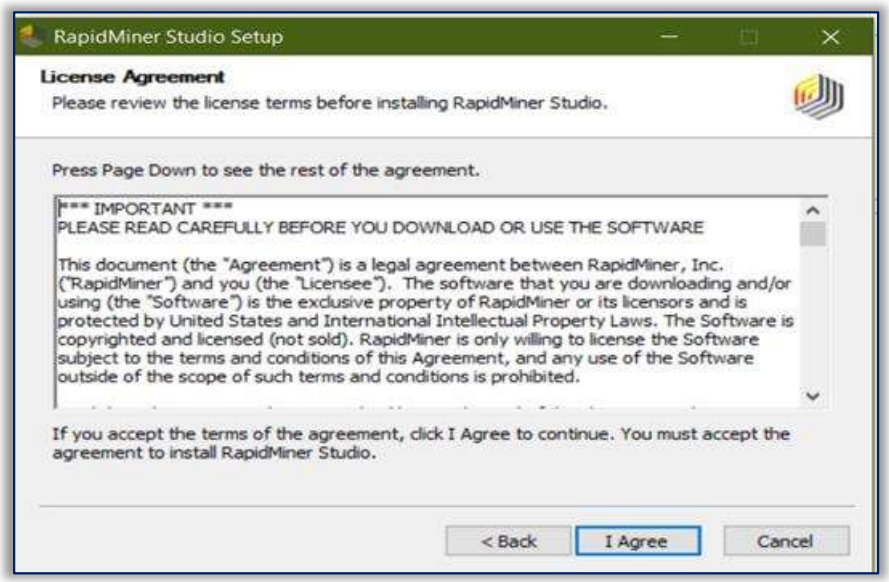


Figure 3: License Agreement

You'll see the welcome screen for the installation. Click "Next" to move to the terms of Use, and if you agree with them, click "I Agree". Finally, select the destination folder for the application (You'll need 224.5MB of free space to install) and click "Install".

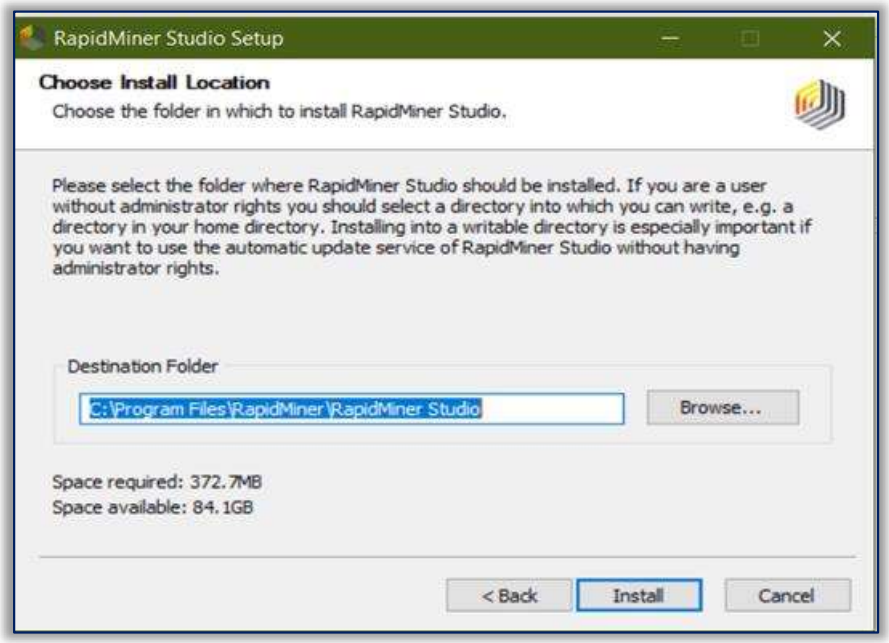


Figure 4: Choose install Location

A progress bar will show the progress of the installation, and when it finishes (takes less than 5 min) you'll see the "Completing the RapidMiner Studio Setup Wizard". Click the Finish button to finish.

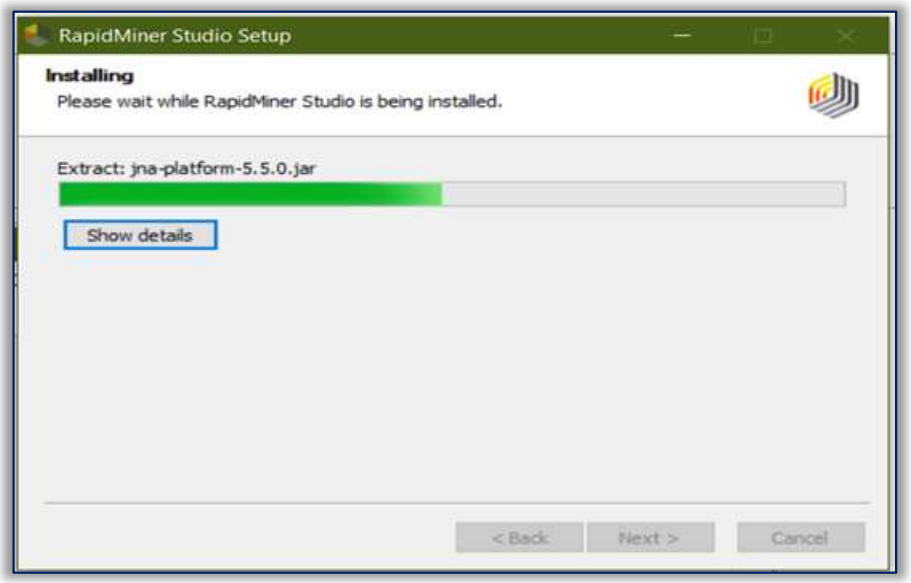


Figure 5: Installation Progress Bar

Click the Finish button to finish the installation process, and congratulations! You are ready to use the application! To open it, just look for it on the desktop, or search for "RapidMiner Studio" on the Windows Start Menu.

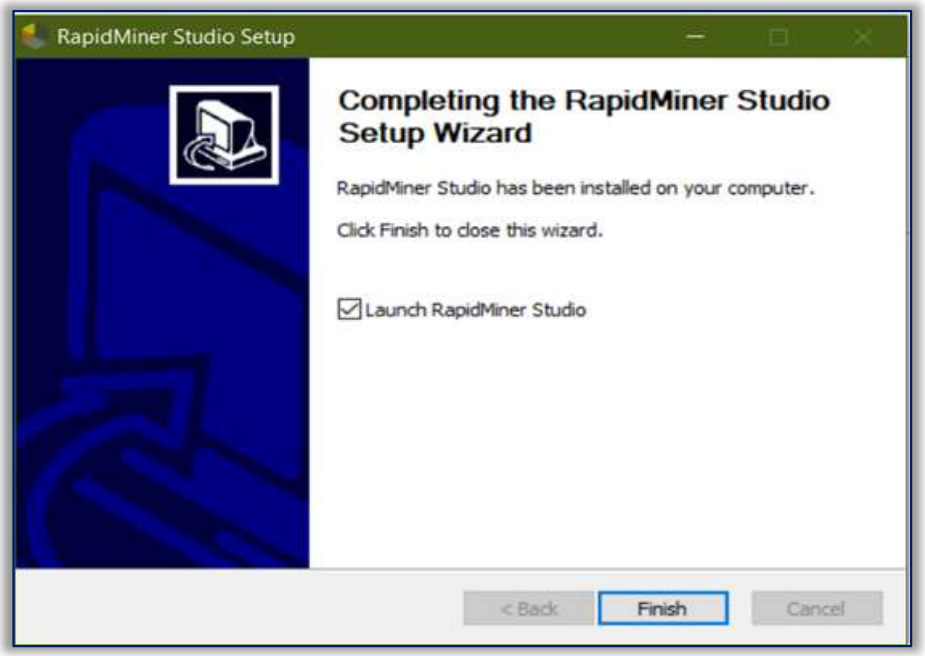


Figure 6: Set-Up Completion Wizard

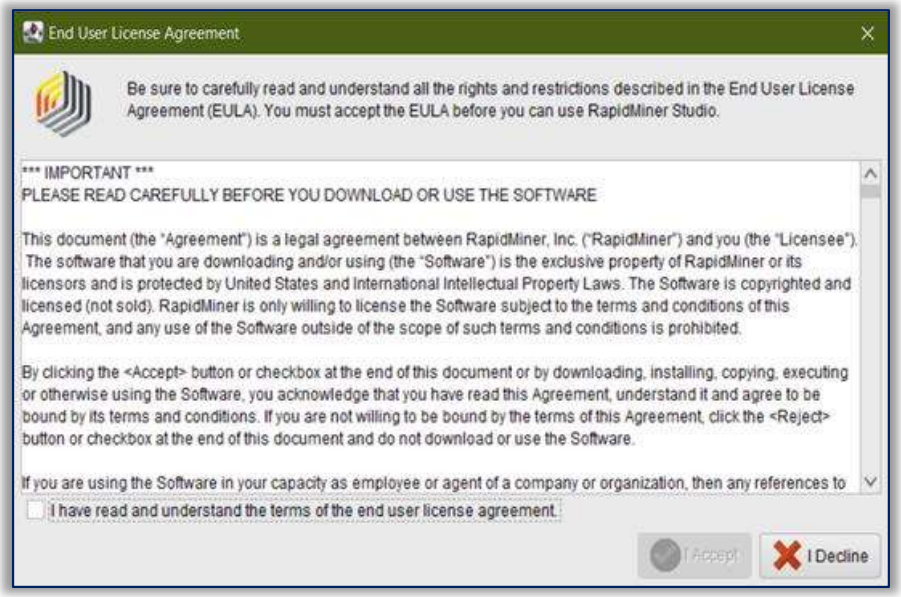


Figure 7: End-User License Agreement

Download and install RapidMiner on your system

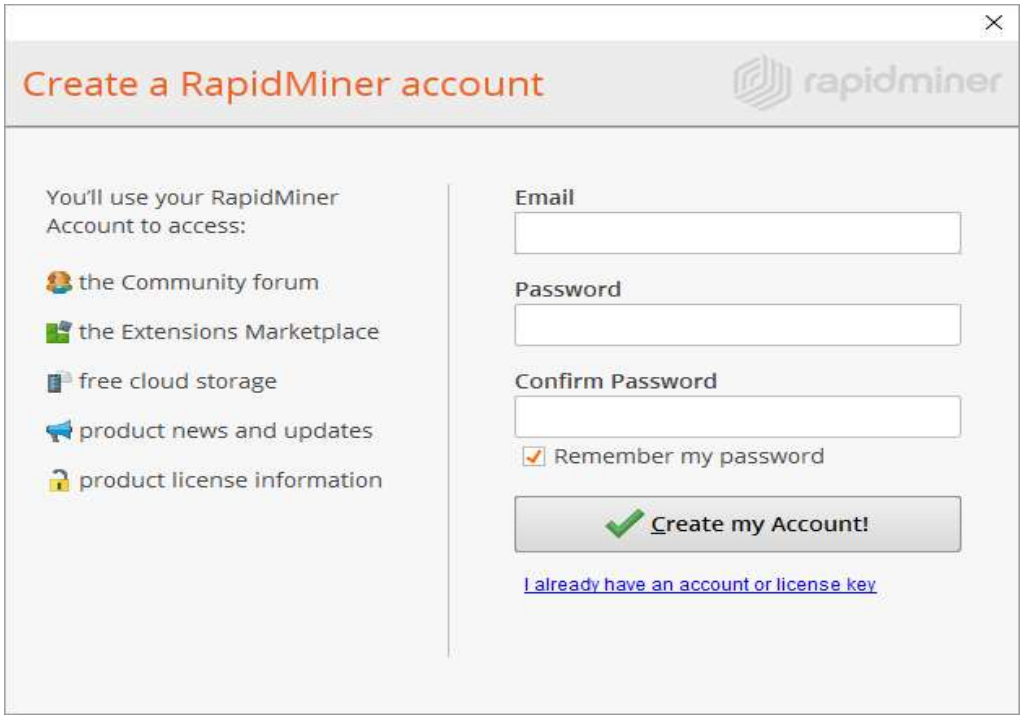


Figure 8: User account creation Interface

Getting started with RapidMiner Studio

Once you launch RapidMiner Studio, a Welcome screen appears, prompting with two options:

Table:

Option	Description
Create my Account	Creates an account with the given useful information.
I already have an account or license key	If you previously registered, enter your rapidminer.com credentials to log in so that you can download or install your license(s).

Create RapidMiner account

Creating an account requires that RapidMiner Studio can access the Internet. If you do not have an Internet connection for the application (for example, if prevented by a firewall), create an account through your browser and enter your license key manually.

Complete the steps to create an account by entering your email address and password. Note that the password must be a minimum of six characters.

Immediately, the system sends an activation email to the address that you registered. (Allow email from **RapidMiner** if you do not see an email titled "Verify Your Email" in your inbox.) Periodic checks verify whether your email has been validated.

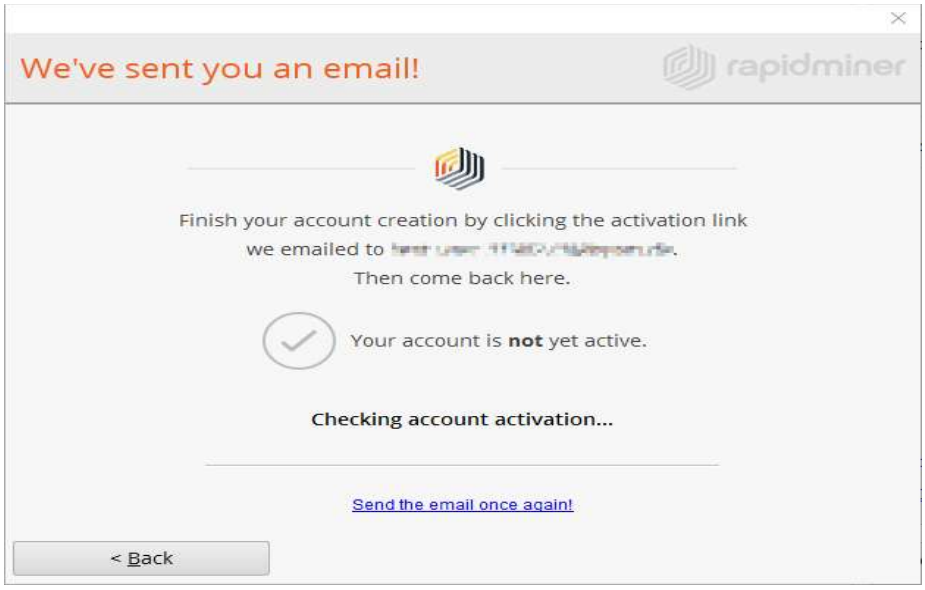


Figure 9:

Once you validate, you'll receive a success message in your browser:

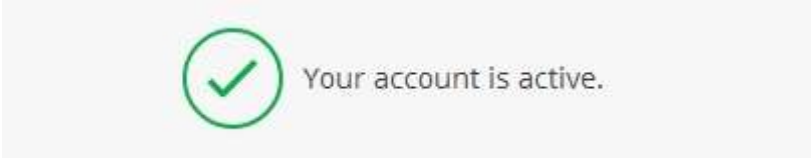


Figure10:

Return to the RapidMiner Studio application. Complete the installation by clicking I'm ready.

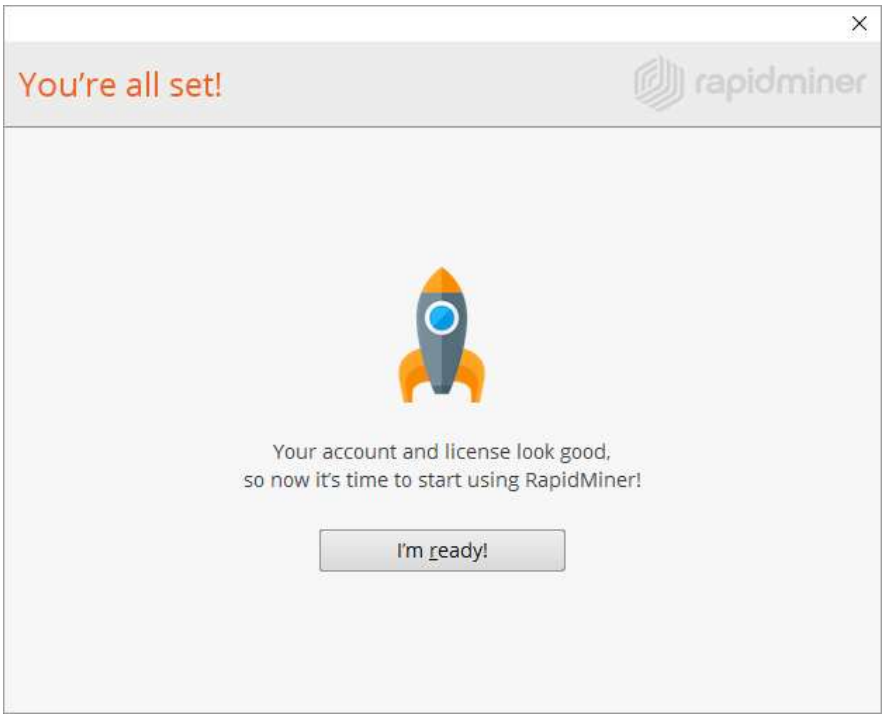


Figure 11

RapidMiner Studio login

If you have already created a RapidMiner account, clicking **I already have an account or license key** on the Welcome screen brings up the login screen:

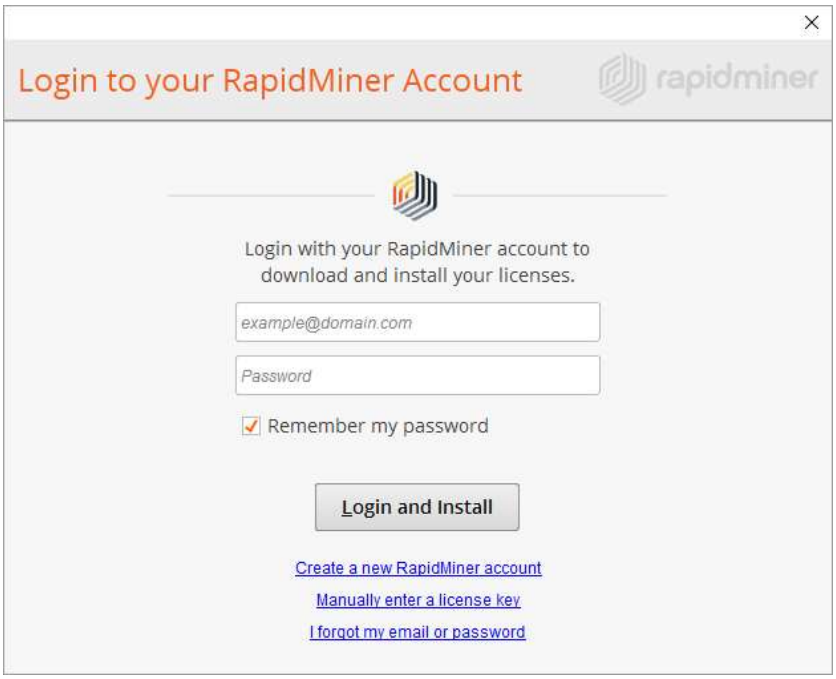


Figure 12: Login Screen

Enter your email address and password to log in with your RapidMiner.com account and then click **Login and Install**.

Create and verify your RapidMiner Account

4.3 Weka

WEKA - open-source software that provides tools for data preprocessing, implementation of several Machine Learning algorithms, and visualization tools so that you can develop machine learning techniques and apply them to real-world data mining problems. What WEKA offers is summarized in the following diagram:

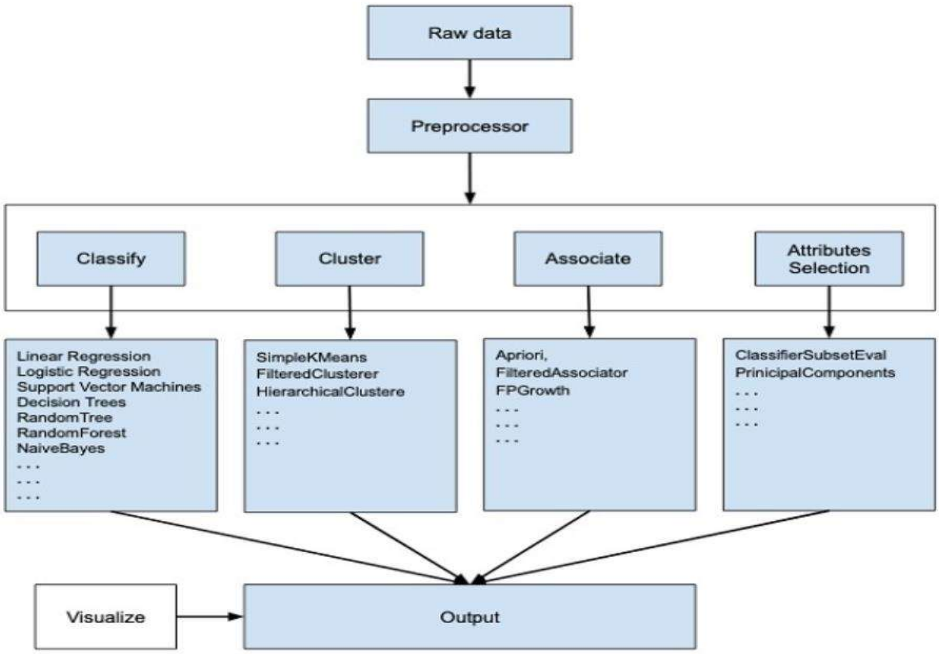


Figure 13: Weka Summary

If you observe the beginning of the flow of the image, you will understand that there are many stages in dealing with Big Data to make it suitable for machine learning –

First, you will start with the raw data collected from the field. This data may contain several null values and irrelevant fields. You use the data preprocessing tools provided in WEKA to cleanse the data. Then, you would save the preprocessed data in your local store for applying ML algorithms. Next, depending on the kind of ML model that you are trying to develop you would select one of the options such as Classify, Cluster, or Associate. The Attributes Selection allows the automatic selection of features to create a reduced dataset. Note that under each category, WEKA provides the implementation of several algorithms. You would select an algorithm of your choice, set the desired parameters, and run it on the dataset. Then, WEKA would give you the statistical output of the model processing. It provides you a visualization tool to inspect the data.

The various models can be applied to the same dataset. You can then compare the outputs of different models and select the best that meets your purpose. Thus, the use of WEKA results in quicker development of machine learning models on the whole.

Weka – Installation

To install WEKA on your machine, visit WEKA's official website and download the installation file. WEKA supports installation on Windows, Mac OS X, and Linux. You just need to follow the instructions on this page to install WEKA for your OS.

The steps for installing on Mac are as follows –

- Download the Mac installation file.
- Double click on the downloaded **weka-3-8-3-corretto-jvm.dmg file**.
- You will see the following screen on successful installation.

Figure 14: Weka Selection

Click on the **weak-3-8-3-corretto-JVM** icon to start Weka.

Optionally you may start it from the command line −

java -jar weka.jar

The WEKA GUI Chooser application will start and you would see the following screen −

Figure 15: Weka Home Screen

The GUI Chooser application allows you to run five different types of applications as listed here −

- Explorer
- Experimenter
- KnowledgeFlow
- Workbench
- Simple CLI

Install Weka and explore all the options of Weka Home Screen

WEKA Download And Installation

1) Download the software from https://waikato.github.io/weka-wiki/downloading_weka/

Check the configuration of the computer system and download the stable version of WEKA.

2) After a successful download, open the file location and double click on the downloaded file. The Step Up wizard will appear. Click on Next.

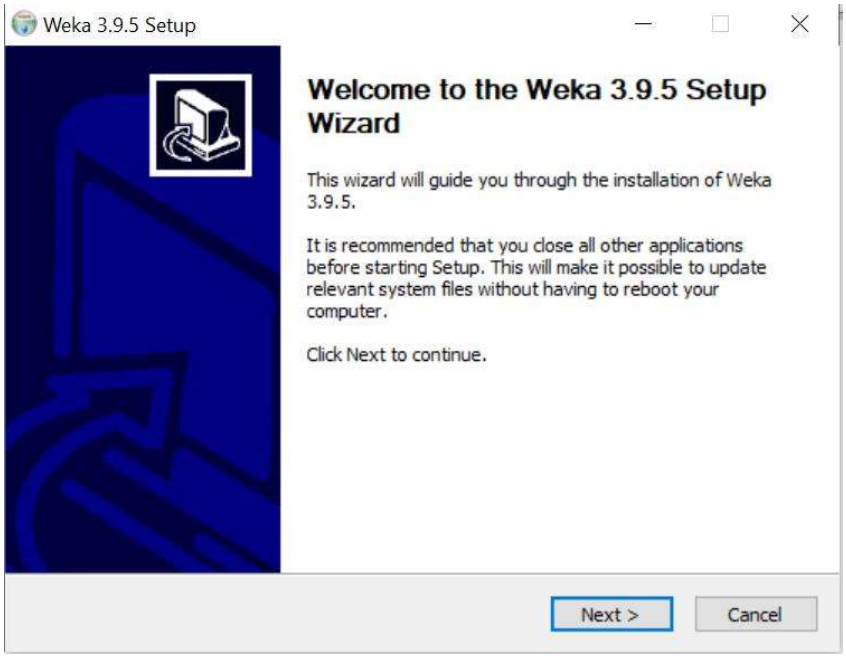


Figure 16: Set-up Wizard

3) The License Agreement terms will open. Read it thoroughly and click on "I Agree".

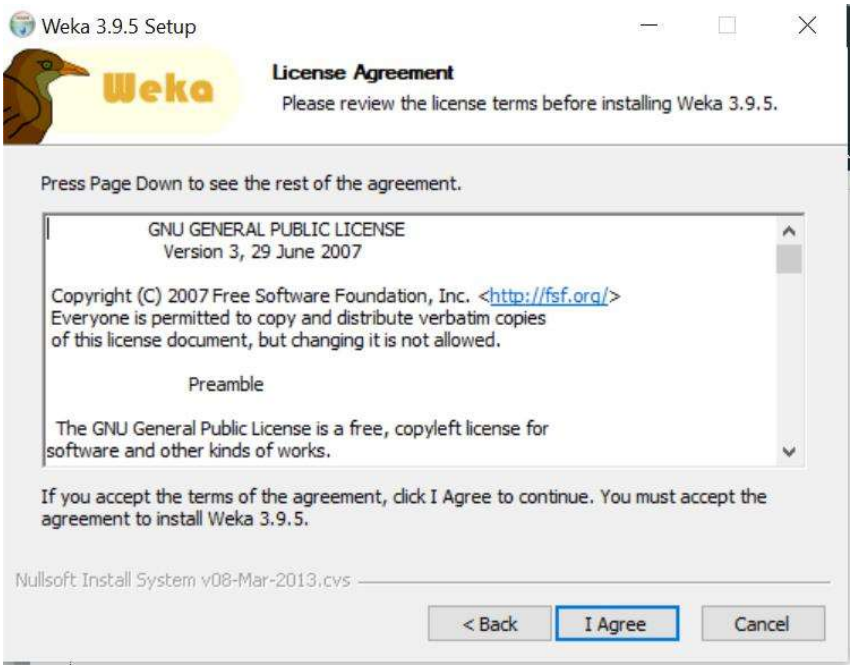


Figure17:

4) According to your requirements, select the components to be installed. Full component installation is recommended. Click on Next.

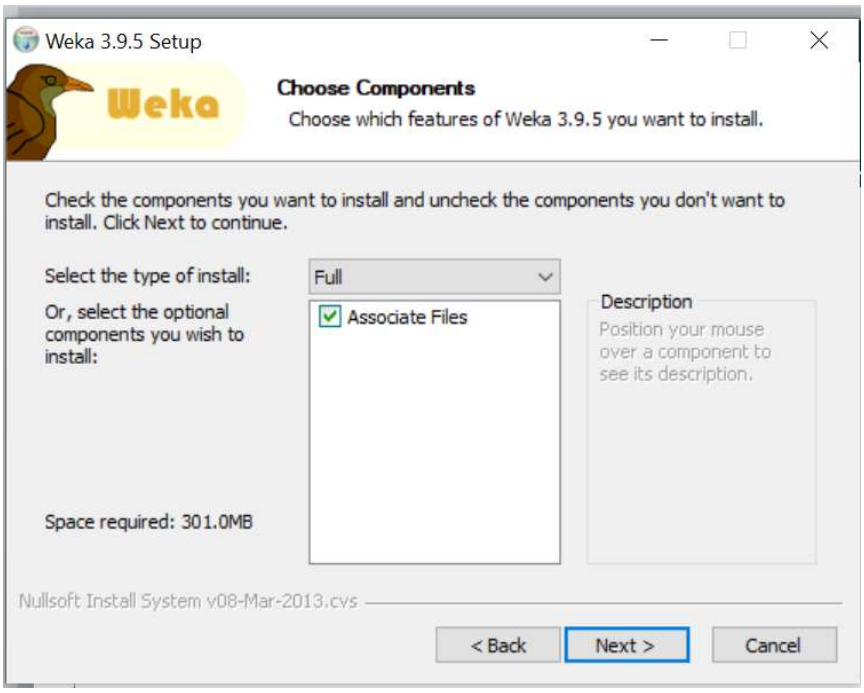


Figure18:

5) Select the destination folder and Click on Next.

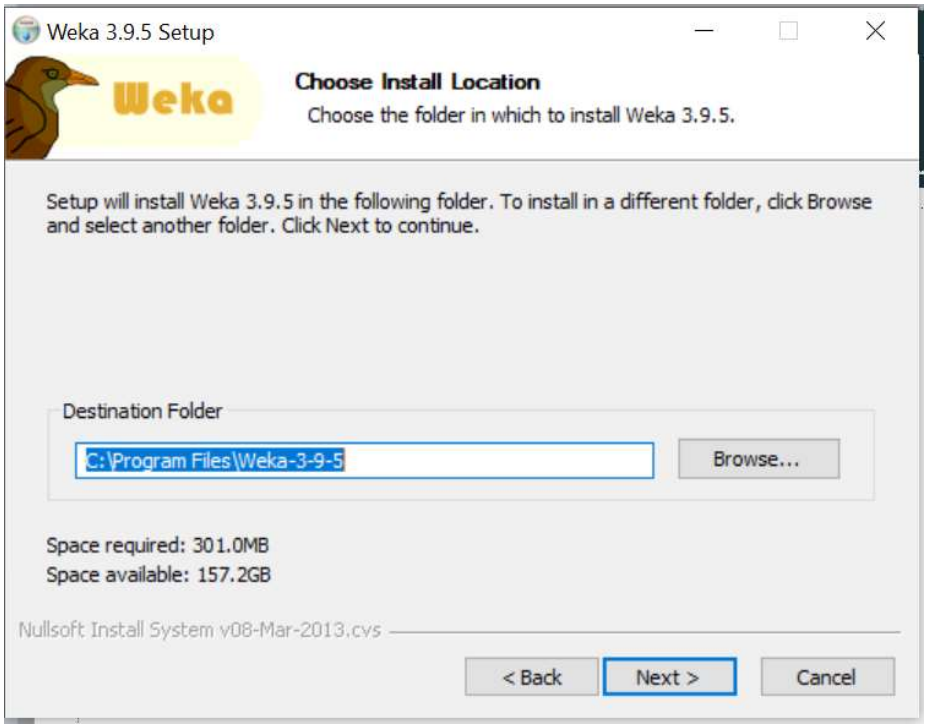


Figure19:

6) Then, Installation will start.

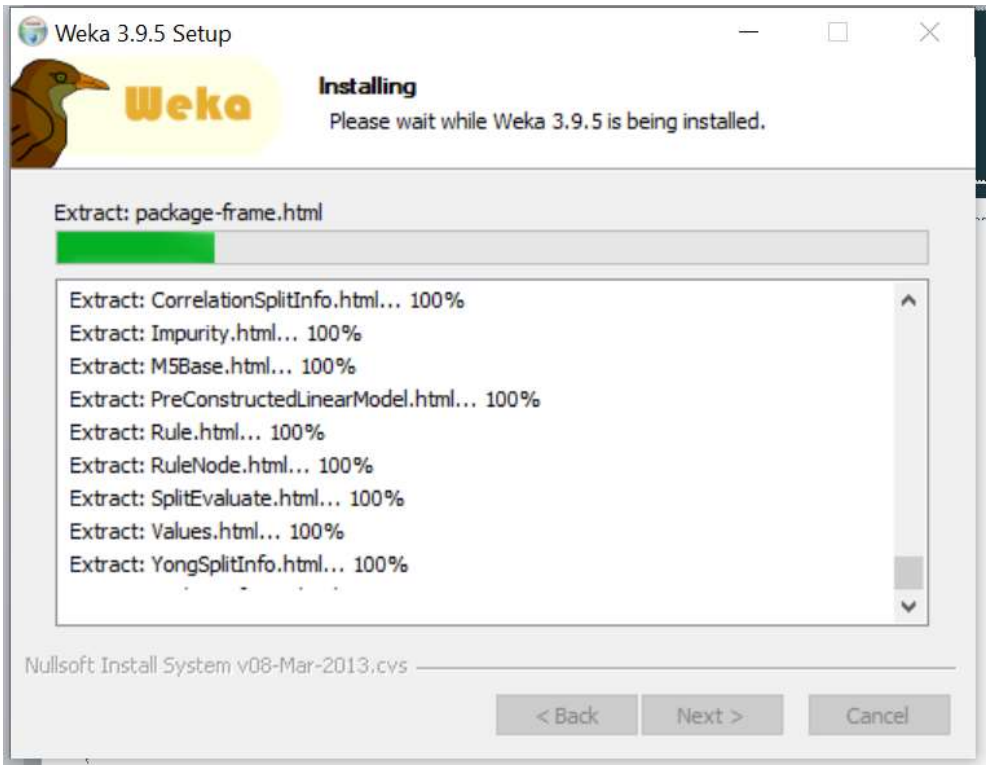


Figure20:

7) After the installation is complete, the following window will appear. Click on Next.

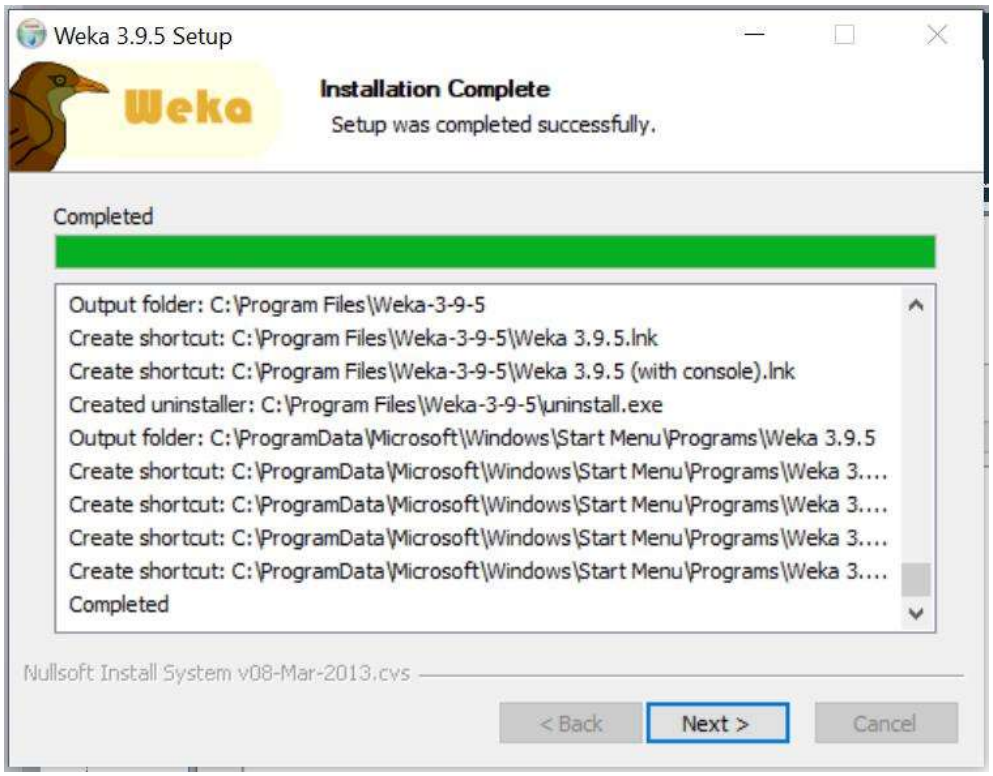


Figure21:

8) Select the Start Weka checkbox. Click on Finish.

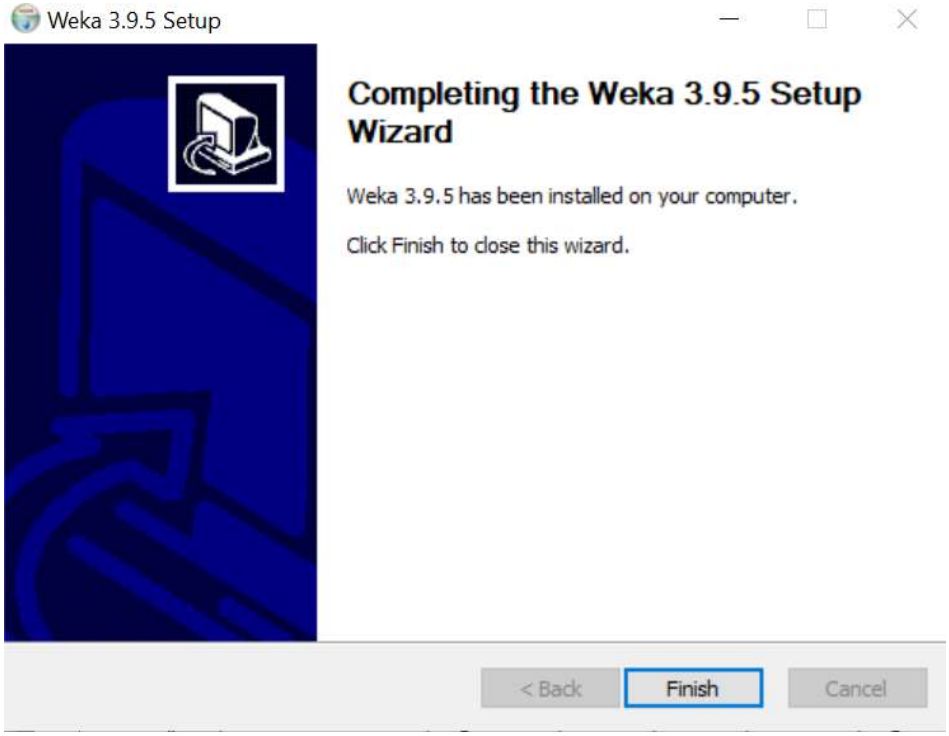


Figure22:

9) WEKA Tool and Explorer window opens.

Figure23:

Install the current version of Weka on your system

Summary

- RapidMiner is a free-of-charge, open-source software tool for data and text mining.
- In addition to Windows operating systems, RapidMiner also supports Macintosh, Linux, and Unix systems.
- RapidMiner Studio is a visual data science workflow designer accelerating the prototyping & validation of models.
- With RapidMiner Studio, you can access, load, and analyze any type of data – both traditional structured data and unstructured data.
- Weka is a collection of machine learning algorithms for data mining tasks.
- Weka algorithms can either be applied directly to a dataset or called from your own Java code.
- Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization.

Keywords

Numeric: All kinds of number values; include the date, time, integer, and real numbers.

Text: Nominal data type that allows for more granular distinction (to differentiate from polynomial).

Delete Repository Entry: An operator to delete a repository entry within a process.

Specificity: Specificity relates to the classifier's ability to identify negative results.

Arff Format: An Arff file contains two sections - header and data.

Self Assessment

1. A database where all of the values for a particular column are stored contiguously is called?

a. Column-oriented storage
b. In-memory database
c. Partitioning
d. Data Compression

2. Data mining can also be applied to other forms such as .....

i) Data streams
ii) Sequence data
iii) Networked data
iv) Text data
v) Spatial data

Select the correct one:

a. i, ii, iii, and v only
b. ii, iii, iv, and v only
c. I, iii, iv, and v only
d. All i, ii, iii, iv and v

3. The _____ is a symbolic representation of facts or ideas from which information can potentially be extracted.

a. knowledge.
b. data.
c. algorithm.
d. program.

4. Data mining is used to refer to the _____ stage in knowledge discovery in the database.
   a. selection.
   b. retrieving.
   c. discovery.
   d. coding.
5. Which of the following can be considered as the correct process of Data Mining?
   a. Infrastructure, Exploration, Analysis, Interpretation, Exploitation
   b. Exploration, Infrastructure, Analysis, Interpretation, Exploitation
   c. Exploration, Infrastructure, Interpretation, Analysis, Exploitation
   d. Exploration, Infrastructure, Analysis, Exploitation, Interpretation
6. Which of the following is an essential process in which the intelligent methods are applied to extract data patterns?
   a. Warehousing
   b. Data Mining
   c. Text Mining
   d. Data Selection
7. In data mining, how many categories of functions are included?
   a. 5
   b. 4
   c. 2
   d. 3
8. The issues like efficiency, scalability of data mining algorithms comes under_____
   a. Performance issues
   b. Diverse data type issues
   c. Mining methodology and user interaction
   d. All of the above
9. Which of the following statements about the query tools is correct?
   a. Tools developed to query the database
   b. Attributes of a database table that can take only numerical values
   c. Both and B
   d. None of the above
10. Which one of the following refers to the binary attribute?
    a. This takes only two values. In general, these values will be 0 and 1, and they can be coded as one bit
    b. The natural environment of a certain species
    c. Systems that can be used without knowledge of internal operations
    d. All of the above
11. Which of the following is the data mining tool?
    a. Borland C.
    b. Weka.
    c. Borland C++.
    d. Visual C.

12. Which one of the following issues must be considered before investing in data mining?
    a. Compatibility
    b. Functionality
    c. Vendor consideration
    d. All of the above

Review Questions

1. What is Rapid Miner? explain the various facilities provided by Rapid Miner.
2. Explain the process of creating a user account in Rapid Miner.
3. Elaborate on various Rapid Miner products.
4. Write down the installation steps of Rapid Miner.
5. How to install Weka in a windows environment. Write all the steps required for the installation.

Answers:

1	a	2	d
3	b	4	c
5	a	6	b
7	c	8	a
9	a	10	a
11	b	12	d

Further Readings

Hofmann, M., & Klinkenberg, R. (Eds.). (2016). *RapidMiner: Data mining use cases and business analytics applications*. CRC Press.

Kotu, V., & Deshpande, B. (2014). *Predictive analytics and data mining: concepts and practice with rapidminer*. Morgan Kaufmann.

Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

Garner, S. R. (1995, April). Weka: The Waikato environment for knowledge analysis. In *Proceedings of the New Zealand computer science research students conference* (Vol. 1995, pp. 57-64).

https://rapidminer.com/get-started/

https://rapidminer.software.informer.com/5.3/

https://waikato.github.io/weka-wiki/downloading_weka/

https://www.tutorialspoint.com/weka/weka_installation.htm

Unit 05: Introduction to Mining Tools

CONTENTS

Objectives

After this lecture, you will be able to

- Understand the basics of rapidminer and its products.
- Learn the working of the Weka tool.
- understand various features of a rapid miner and Weka Tool.
- Know the Interface of RapidMiner and Weka.

Introduction

Several open-source and proprietary-based data mining and data visualization tools exist which are used for information extraction from large data repositories and for data analysis. Some of the data mining tools which exist in the market are Weka, Rapid Miner, Orange, R, KNIME, ELKI, GNU Octave, Apache Mahout, SCaViS, Natural Language Toolkit, Tableau, etc.

5.1 RapidMiner

RapidMiner is an integrated enterprise artificial intelligence framework that offers AI solutions to positively impact businesses. RapidMiner is widely used in many business and commercial applications as well as in various other fields such as research, training, education, rapid prototyping, and application developmentAll major machine learning processes such as data preparation, model validation, results in visualization, and optimization can be carried out by using RapidMiner.

Rapidminer comes with :

- **Over 125 mining algorithms**
- Over 100 data cleaning and preparation functions.
- Over 30 charts for data visualization,and selection of metrics to evaluate model performance.

5.2 Installing Rapidminer on your machine

- The latest version of Rapidminer Studio is V7, it can be downloaded from https://rapidminer.com/products/comparison/
- For Windows: download the rapidminer-install.exe and install.
- Defaults install it to C:\program files and add it to the start>programs menu.
- For mac: download the .dmg and add it to your applications folder.

RapidMiner Products

There are many products of RapidMiner that are used to perform multiple operations. Some of the products are:

- **RapidMiner Studio:**With RapidMiner Studio, one can access, load, and analyze both traditional structured data and unstructured data like text, images, and media. It can also extract information from these types of data and transform unstructured data into structured.
- **RapidMiner Auto Model:**Auto Model is an advanced version of RapidMiner Studio that increments the process of building and validating data models. You can customize the processes and can put them in production based on your needs. Majorly three kinds of problems can be resolved with Auto Model namely prediction, clustering, and outliers.
- **RapidMiner Turbo Prep:**Data preparation is time-consuming and RapidMiner Turbo Prep is designed to make the preparation of data much easier. It provides a user interface where your data is always visible front and center, where you can make changes step-by-step and instantly see the results, with a wide range of supporting functions to prepare the data for model-building or presentation.

TOOL CHARACTERISTICS

- **Usability: E***asy to use*
- **Tool orientation:**The **t**ool is designed for general-purpose analysis
- **Data mining type:**This tool is made for *Structured data mining, Text mining, Image mining, Audio mining, Video mining, Data gathering, Social network analysis*.
- **Manipulation type:**This tool is designed for *Data extraction, Data transformation, Data analysis, Data visualization, Data conversion, Data cleaning*

5.3 Features of RapidMiner

- **Application & Interface:**RapidMiner Studio is a visual data science workflow designer accelerating the prototyping & validation of models.
- **Data Access**: With RapidMiner Studio, you can access, load, and analyze any type of data – both traditional structured data and unstructured data like text, images, and media. It can also extract information from these types of data and transform unstructured data into structured.
- **Data Exploration:** Immediately understand and create a plan to prepare the data automatically extract statistics and key information.
- **Data Prep:**The richness of the data preparation capabilities in RapidMiner Studio can handle any real-life data transformation challenges, so you can format and create the optimal data set for predictive analytics. RapidMiner Studio can blend structured with unstructured data and then leverage all the data for predictive analysis. Any data preparation process can be saved for reuse.
- **Modeling**: RapidMiner Studio comes equipped with an un-paralleled set of modeling capabilities and machine learning algorithms for supervised and unsupervised learning. They are flexible, robust and allow you to focus on building the best possible models for any use case.

- **Validation:**RapidMiner Studio provides the means to accurately and appropriately estimate model performance. Where other tools tend to too closely tie modeling and model validation, RapidMiner Studio follows a stringent modular approach which prevents information used in pre-processing steps from leaking from model training into the application of the model. This unique approach is the only guarantee that no overfitting is introduced and no overestimation of prediction performances can occur.
- **Scoring:**RapidMiner Studio makes the application of models easy, whether you are scoring them in the RapidMiner platform or using the resulting models in other applications.
- **Code Control & Management:**Unlike many other predictive analytics tools, RapidMiner Studio covers even the trickiest data science use cases without the need to program. Beyond all the great functionality for preparing data and building models, RapidMiner Studio has a set of utility-like process control operations that lets you build processes that behave like a program to repeat and loop over tasks, branch flows and call on system resources. RapidMiner Studio also supports a variety of scripting languages.

5.4 How does it work?

You visually design a data mining process. A process is like a flow chart for mining operators.

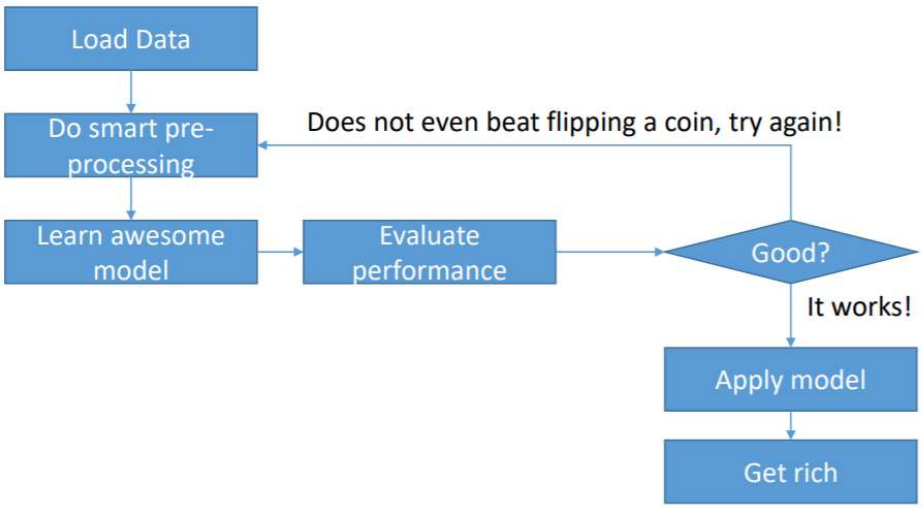


Figure 1: Working of RapidMiner

All data that you load will be contained in an example set. Each example is described by Attributes (a.k.a. features) and all attributes have Value Types and their specific Roles. The Value types define how data is treated

- Numeric data has an order (2 is closer to 1 than to 5)
- Nominal data has no order (red is as different from green as from blue)

Table 1: Different Value Type

Value Type	Description
binominal	Only two different values are permitted
polynomial	More than two different values are permitted
integer	Whole numbers, positive and negative
real	Real numbers, positive and negative
date_time	Date as well as time date Only date-time Only time

Roles define how the attribute is treated by the Operators.

Table 2: Types of Roles

Role	Description
Id	A unique identifier, no two examples in an example set can have the same value
Regular (default)	Regular attribute that contains data
Label	The target attribute for classification tasks
Weight	The weight of the Examples concerning the label
Cluster	Created by RapidMiner as the result of a clustering task
Prediction	Created by RapidMiner as the result of a classification task

The Repository

This is where you store your data and processes.Only if you load data from the repository, RapidMiner can show you which attributes exist. Add data via the "Add Data" button or the "Store" operator. You can load data via drag 'n' drop or the "Retrieve" operator. If you have a question starting with "Why does RapidMiner not show me …?" Then the answer most likely is "Because you did not load your data into the Repository!"

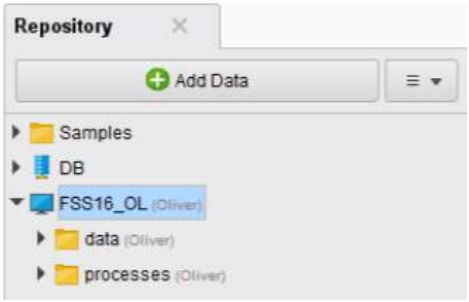


Figure2:Reository

A repository is simply a folder that holds all of your RapidMiner data sets (we call them "ExampleSets), processes, and other file objects that you will create using RapidMiner Studio. This folder can be stored locally on your computer, or on a RapidMiner Server.

Figure3: Store Operator

This operator stores an IO Object at a location in the data repository. The location of the object to be stored is specified through the *repository entry* parameter. The stored object can be used by other processes by using the Retrieve operator.

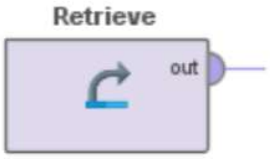


Figure 4: Retrieve Operator

Example: Themost simple process which can be created with RapidMiner: It just retrieves data from a repository and delivers it as a result to the user to allow for inspection.

Benefits of RapidMiner

The main benefits of RapidMiner are its robust features, user-friendly interface, and maximization of data usage. Learn more of its benefits below:

Robust features and user-friendly interface
RapidMiner's tools and features offer powerful capabilities for the users while at the same time are presented through a user-friendly interface that allows users to perform productively in their works from the start. Thus, each of the tools' robust components is easy to be operated. One feature of the system is the visual workflow designer which is a tool that provides a visual environment to the users. This environment is where analytics processes can be designed, created, and then deployed. Visual presentation and models can also be made and processed here. All of these can easily be done by the users because of the friendly environment.

Maximize the usage of data
The system provides the users with the right set of tools which makes relevant uses of even the most disorganized, uncluttered, and seemingly useless data. This can be accomplished by enabling the users and their team to structure data in an easy way for them to comprehend. To do this, RapidMiner offers capabilities that simplify data access and management that will empower users to load, access, and evaluate all types of data such as images and texts.

Not only does the system allows the usage of any data but it also allows them to create models and plans out of them, which can then be used as a basis for decision making and formulation of strategies. RapidMiner has data exploration features, such as descriptive statistics and graphs and visualization, which allows users to get valuable insights out of the information they gained.RapidMiner is also powerful enough to provide analytics that is based on real-life data transformation settings. This means that users can manipulate their data any way they want since they have control of its formatting and the system. Because of this, they can create the optimal data set when performing predictive analytics.

How to use RapidMiner

Use the "Design Perspective" to create your Process

- See your current Process – "Process"
- Access your data and processes – "Repository"
- Add operators to the process – "Operators"
- Configure the operators – "Parameters"
- Learn about operators – "Help"

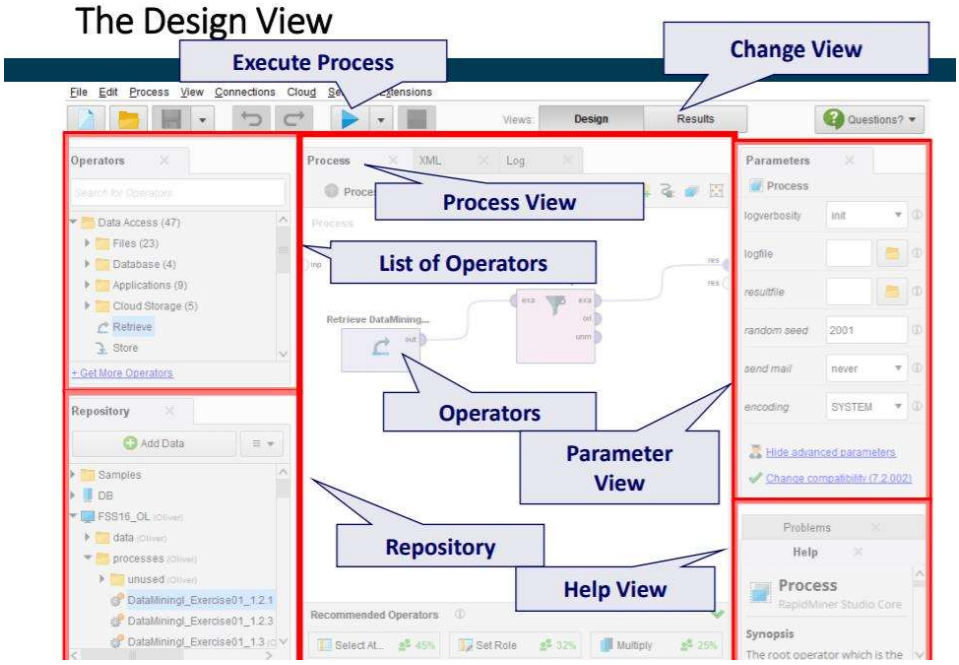


Figure 5:Design View

Use the "Results Perspective" to inspect the output

- The "Data View" shows your example set
- The "Statistics View" contains metadata and statistics
- The "Visualizations View" allows you to visualize the data

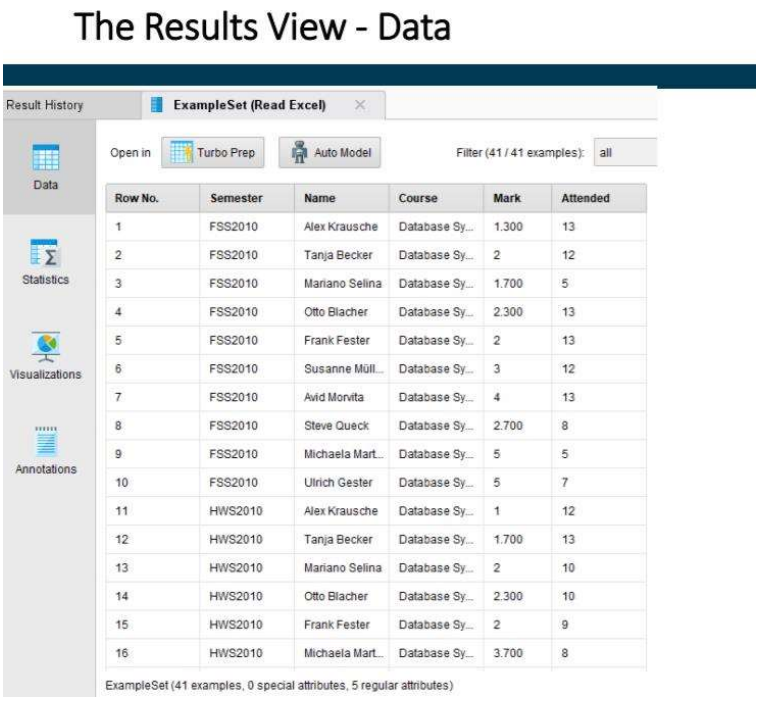


Figure 6:The Result View

Finding an operator

Once you get familiar with operator names, you can find them more easily using the filter at the top of the operator window.

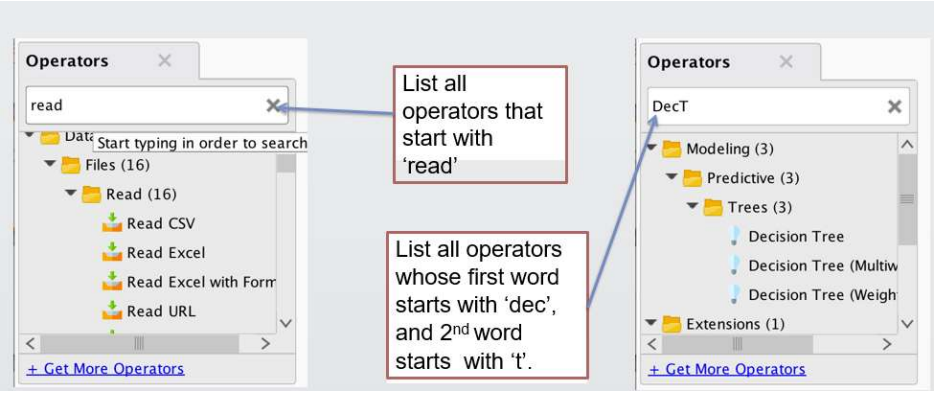


Figure 7: Ways to find an Operator

5.5 Weka

Weka is data mining software that uses a collection of machine learning algorithms. These algorithms can be applied directly to the data or called from the Java code. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

WEKA an open-source software that provides tools for data preprocessing, implementation of several Machine Learning algorithms, and visualization tools so that you can develop machine learning techniques and apply them to real-world data mining problems. What WEKA offers is summarized in the following diagram –

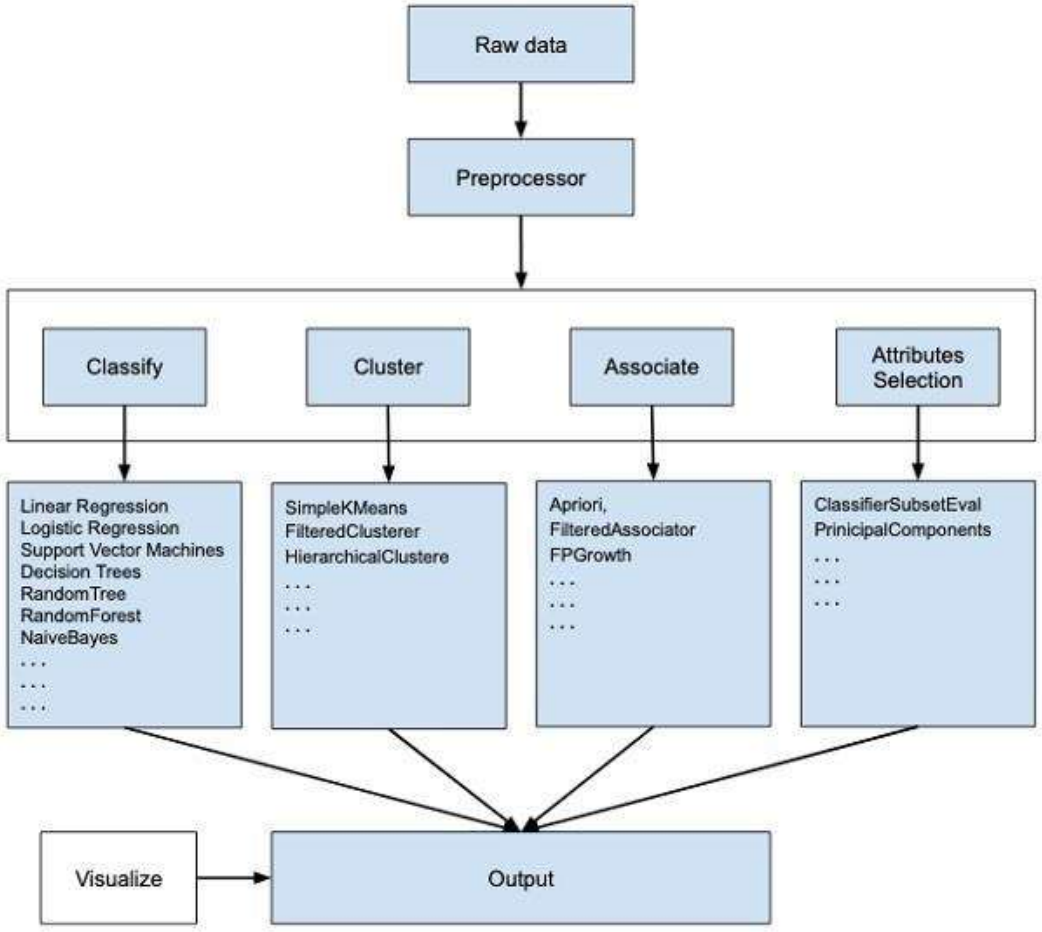


Figure 8: Working of WEKA

If you observe the beginning of the flow of the image, you will understand that there are many stages in dealing with Big Data to make it suitable for machine learning –

First, you will start with the raw data collected from the field. This data may contain several null values and irrelevant fields. You use the data preprocessing tools provided in WEKA to cleanse the data.Then, you would save the preprocessed data in your local store for applying ML algorithms.

Next, depending on the kind of ML model that you are trying to develop you would select one of the options such as Classify, Cluster, or Associate. The Attributes Selection allows the automatic selection of features to create a reduced dataset.Note that under each category, WEKA provides the implementation of several algorithms. You would select an algorithm of your choice, set the desired parameters, and run it on the dataset.

Then, WEKA would give you the statistical output of the model processing. It provides you a visualization tool to inspect the data.The various models can be applied to the same dataset. You can then compare the outputs of different models and select the best that meets your purpose.Thus, the use of WEKA results in quicker development of machine learning models on the whole.

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. The WEKA GUI Chooser application will start and you would see the following screen –

Figure 9:Weka GUI Chooser

The GUI Chooser application allows you to run five different types of applications as listed here –

- Explorer
- Experimenter
- KnowledgeFlow
- Workbench
- Simple CLI

When you click on the Explorer button in the Applications selector, it opens the following screen –

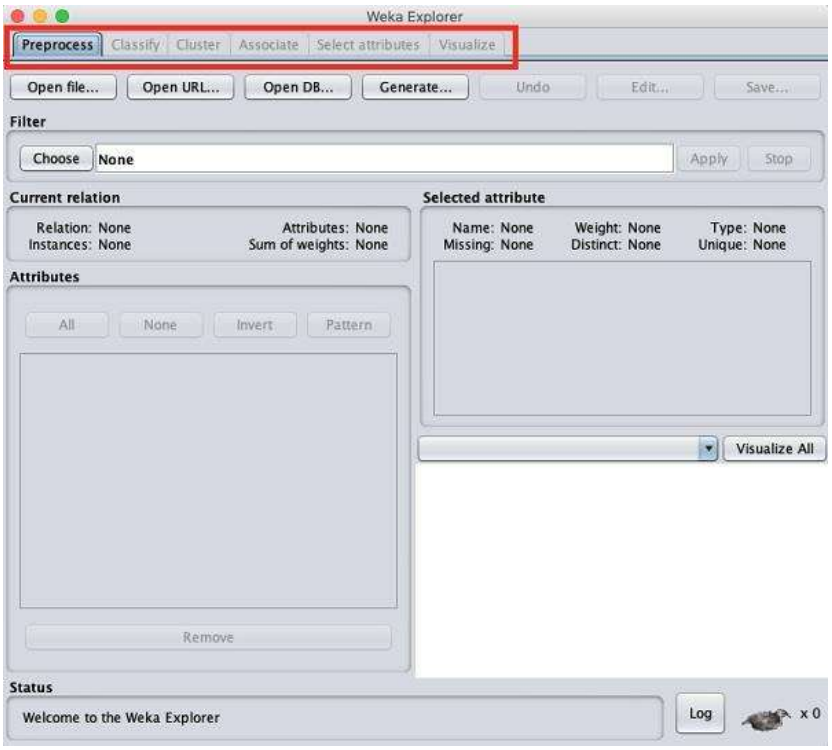


Figure 10: Weka Explorer

- On the top, you will see several tabs as listed here –
- Preprocess
- Classify
- Cluster
- Associate
- Select Attributes
- Visualize
- Under these tabs, there are several pre-implemented machine learning algorithms. Let us look into each of them in detail now.

Preprocess Tab

Initially, as you open the explorer, only the **Preprocess** tab is enabled. The first step in machine learning is to preprocess the data. Thus, in the **Preprocess** option, you will select the data file, process it, and make it fit for applying the various machine learning algorithms.

Classify Tab

The **Classify** tab provides you several machine learning algorithms for the classification of your data. To list a few, you may apply algorithms such as Linear Regression, Logistic Regression, Support Vector Machines, Decision Trees, RandomTree, RandomForest, NaiveBayes, and so on. The list is very exhaustive and provides both supervised and unsupervised machine learning algorithms.

Cluster Tab

Under the **Cluster** tab, there are several clustering algorithms provided - such as SimpleKMeans, FilteredClusterer, HierarchicalClusterer, and so on.

Associate Tab

Under the **Associate** tab, you would find Apriori, FilteredAssociator, and FPGrowth.

Select Attributes Tab

Select Attributes allows you to feature selections based on several algorithms such as ClassifierSubsetEval, PrinicipalComponents, etc.

 Example:They can be used, for example, to store an identifier with each instance in a dataset.

Visualize Tab

The **Visualize** option allows you to visualize your processed data for analysis. WEKA provides several ready-to-use algorithms for testing and building your machine learning applications. To use WEKA effectively, you must have a sound knowledge of these algorithms, how they work, which one to choose under what circumstances, what to look for in their processed output, and so on. In short, you must have a solid foundation in machine learning to use WEKA effectively in building your apps.

we start with the first tab that you use to preprocess the data. This is common to all algorithms that you would apply to your data for building the model and is a common step for all subsequent operations in WEKA.

For a machine-learning algorithm to give acceptable accuracy, you must cleanse your data first. This is because the raw data collected from the field may contain null values, irrelevant columns, and so on.

First, you will learn to load the data file into the WEKA Explorer. The data can be loaded from the following sources −

- Local file system
- Web
- Database

 Analyze your data with WEKA Explorer using various learning schemes and interpret received results.

We will see all three options of loading data in detail.

Loading Data from Local File System

Just under the Machine Learning tabs that you studied in the previous lesson, you would find the following three buttons −

- Open file ...
- Open URL ...
- Open DB ...

Click on the **Open file** ... button. A directory navigator window opens as shown in the following screen −

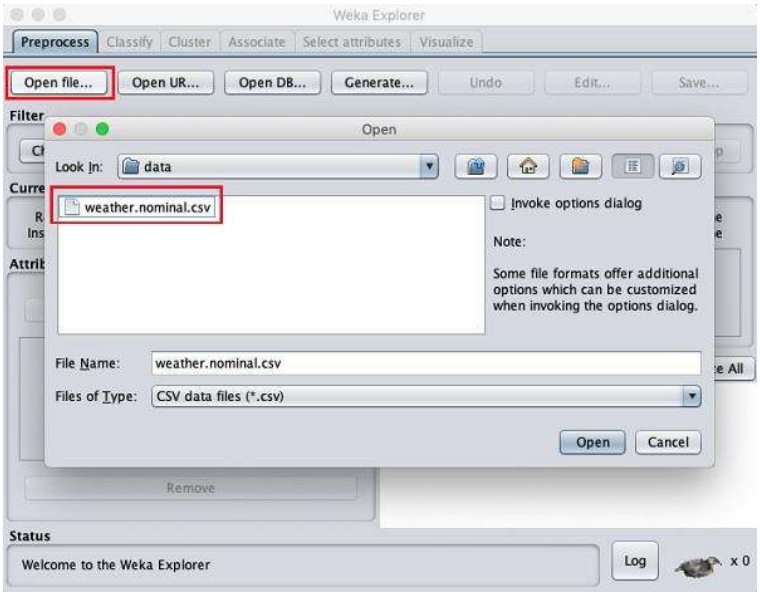


Figure 11: Opening a File using Explorer

Now, navigate to the folder where your data files are stored. WEKA installation comes up with many sample databases for you to experiment with. These are available in the **data** folder of the WEKA installation.

For learning purposes, select any data file from this folder. The contents of the file would be loaded in the WEKA environment. We will very soon learn how to inspect and process this loaded data. Before that, let us look at how to load the data file from the Web.

Developed by the University of Waikato, New Zealand, Weka stands for Waikato Environment for Knowledge Analysis.

Loading Data from Web

Once you click on the **Open URL ...** button, you can see a window as follows –

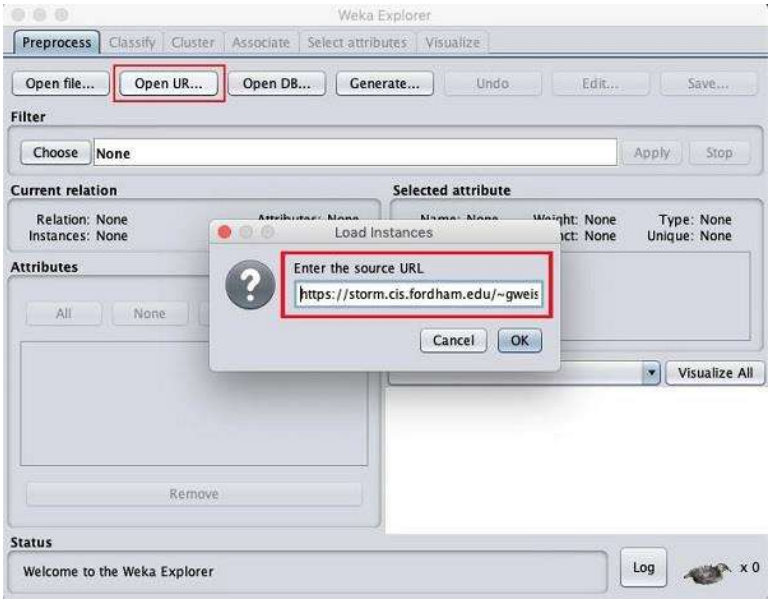


Figure 12:Loading Data from Web

We will open the file from a public URL Type the following URL in the popup box –

https://storm.cis.fordham.edu/~gweiss/data-mining/weka-data/weather.nominal.arff

You may specify any other URL where your data is stored. The **Explorer** will load the data from the remote site into its environment.

Loading Data from DB

Once you click on the **Open DB** ... button, you can see a window as follows –

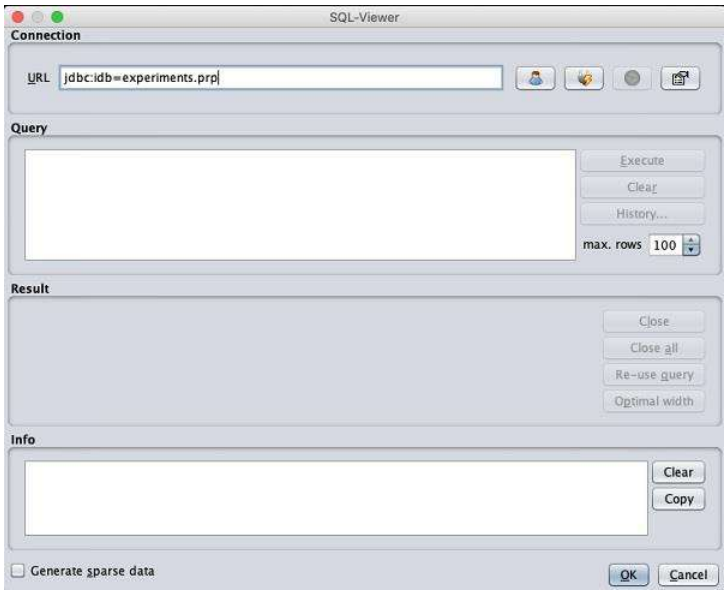


Figure 13: Loading of Data Using DB

Set the connection string to your database, set up the query for data selection, process the query, and load the selected records in WEKA.WEKA supports a large number of file formats for the data. The types of files that it supports are listed in the drop-down list box at the bottom of the screen. This is shown in the screenshot given below.

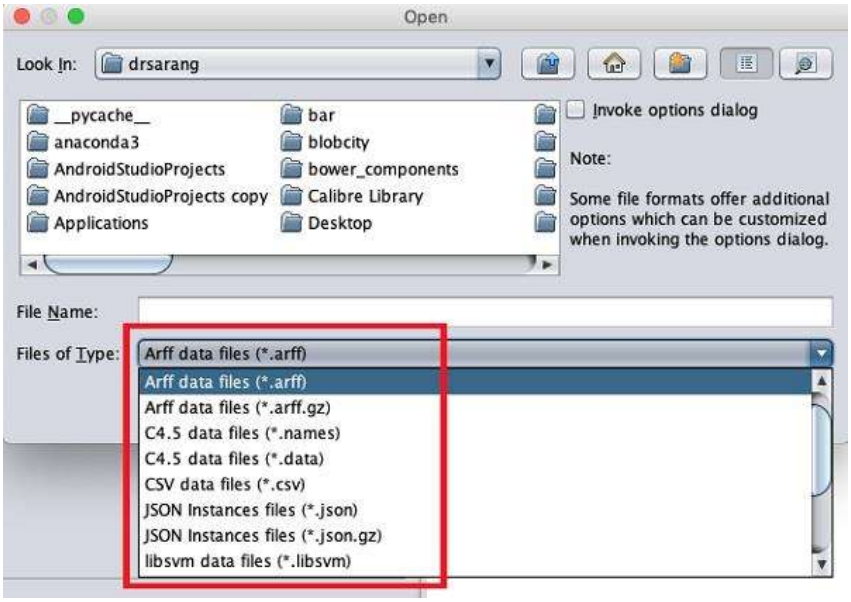


Figure 14: Files Supported by Weka

As you would notice it supports several formats including CSV and JSON. The default file type is Arff.

Arff Format

An **Arff** file contains two sections - header and data.

- The header describes the attribute types.
- The data section contains a comma-separated list of data.

As an example for Arff format, the **Weather** data file loaded from the WEKA sample databases is shown below −

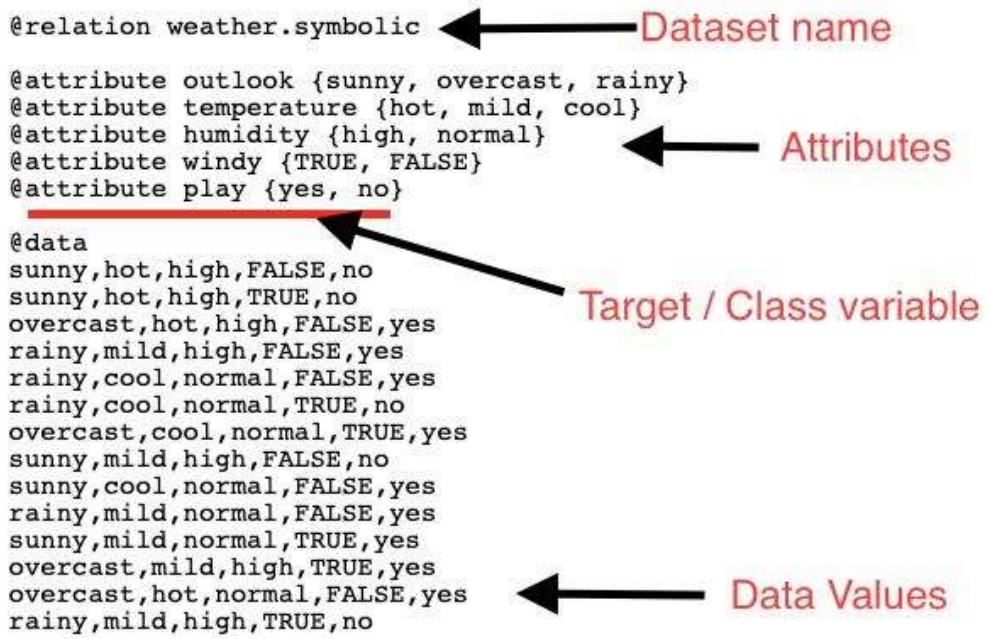


Figure 15:ARFF File

From the screenshot, you can infer the following points −

- The @relation tag defines the name of the database.
- The @attribute tag defines the attributes.
- The @data tag starts the list of data rows each containing the comma-separated fields.

The attributes can take nominal values as in the case of outlook shown here −

@attribute outlook (sunny, overcast, rainy)

The attributes can take real values as in this case −

@attribute temperature real

You can also set a Target or a Class variable called to play as shown here −

@attribute play (yes, no)

The Target assumes two nominal values yes or no.

Create your ARFF file and load it using Weka.

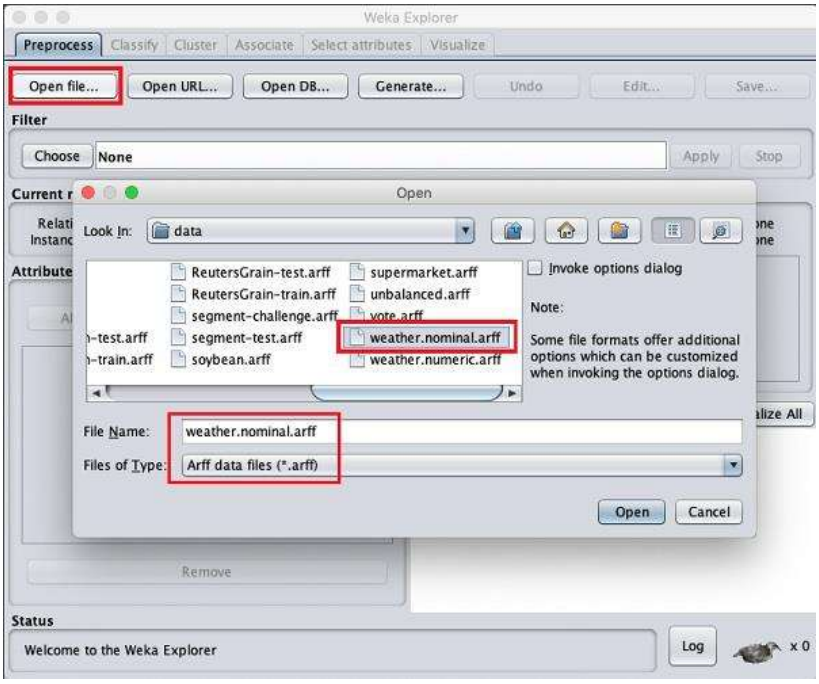


Figure 16:Opening a file using Weka

When you open the file, your screen looks like as shown here –

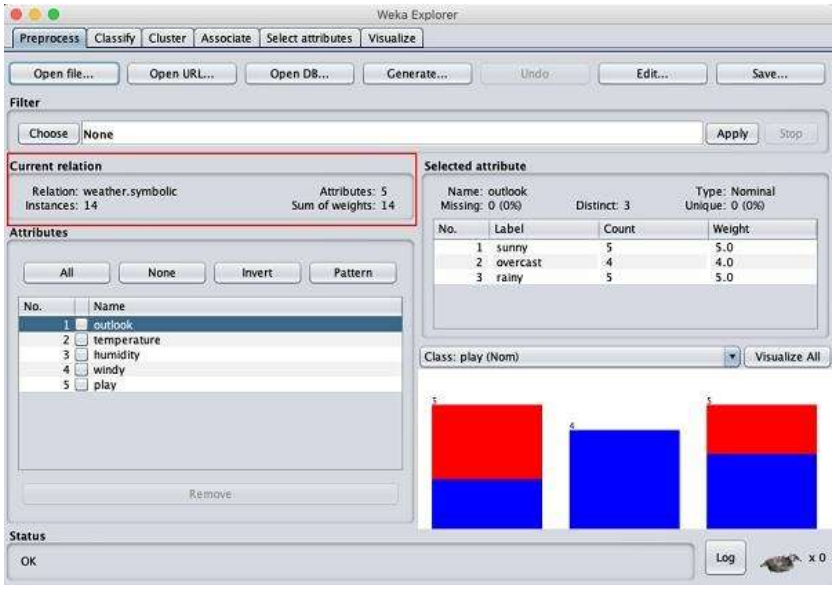


Figure 17: Pre-processing of Data

Understanding Data

Let us first look at the highlighted **Current relation** sub-window. It shows the name of the database that is currently loaded. You can infer two points from this sub window –

- There are 14 instances - the number of rows in the table.
- The table contains 5 attributes - the fields, which are discussed in the upcoming sections.

On the left side, notice the **Attributes** sub-window that displays the various fields in the database.

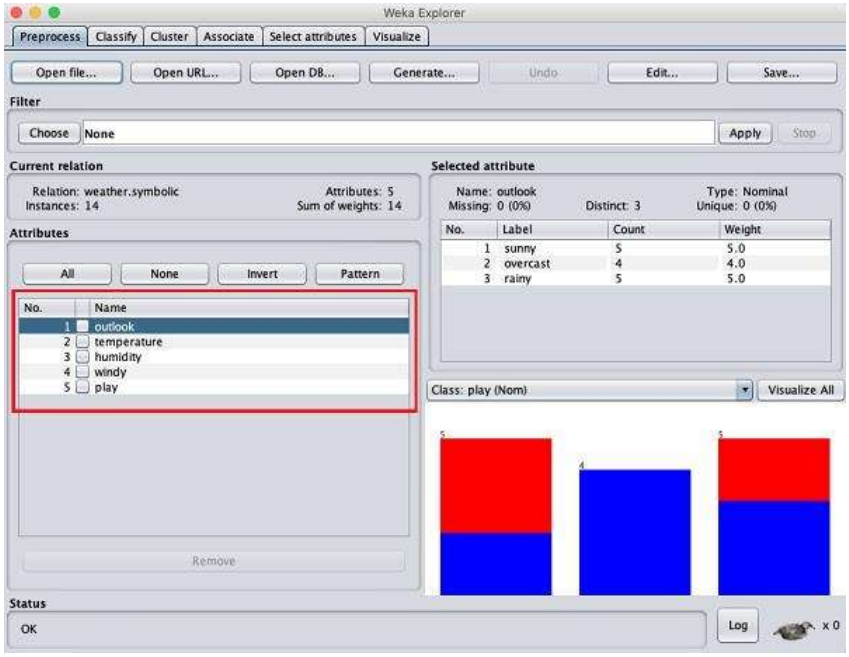


Figure 18: Selected Relation Attributes

The **weather** database contains five fields - outlook, temperature, humidity, windy, and play. When you select an attribute from this list by clicking on it, further details on the attribute itself are displayed on the right-hand side.

Let us select the temperature attribute first. When you click on it, you would see the following screen

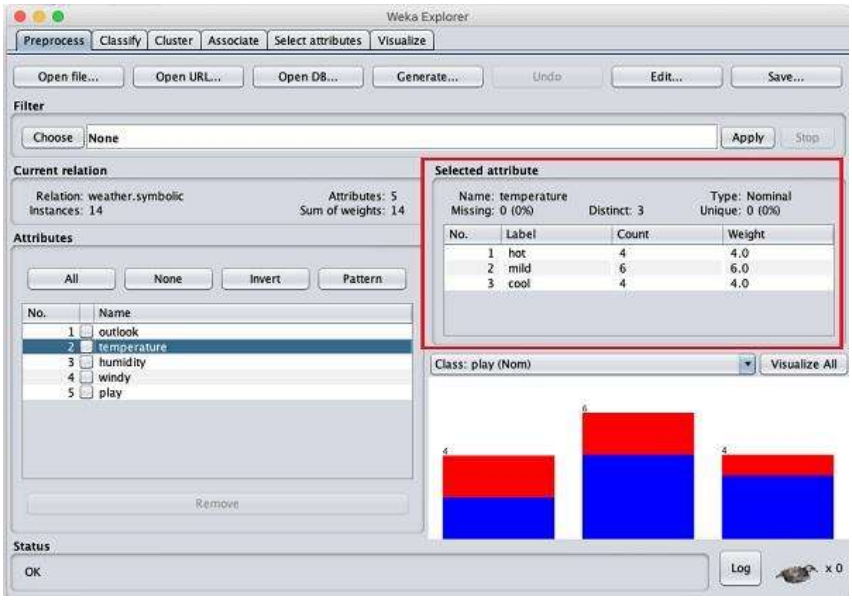


Figure 19: Statistics of Selected Attributes

In the **Selected Attribute** subwindow, you can observe the following −

- The name and the type of attribute are displayed.
- The type for the **temperature** attribute is **Nominal**.
- The number of **Missing** values is zero.
- There are three distinct values with no unique value.

- The table underneath this information shows the nominal values for this field as hot, mild, and cold.
- It also shows the count and weight in terms of a percentage for each nominal value.

At the bottom of the window, you see the visual representation of the **class** values.

If you click on the **Visualize All** button, you will be able to see all features in one single window as shown here –

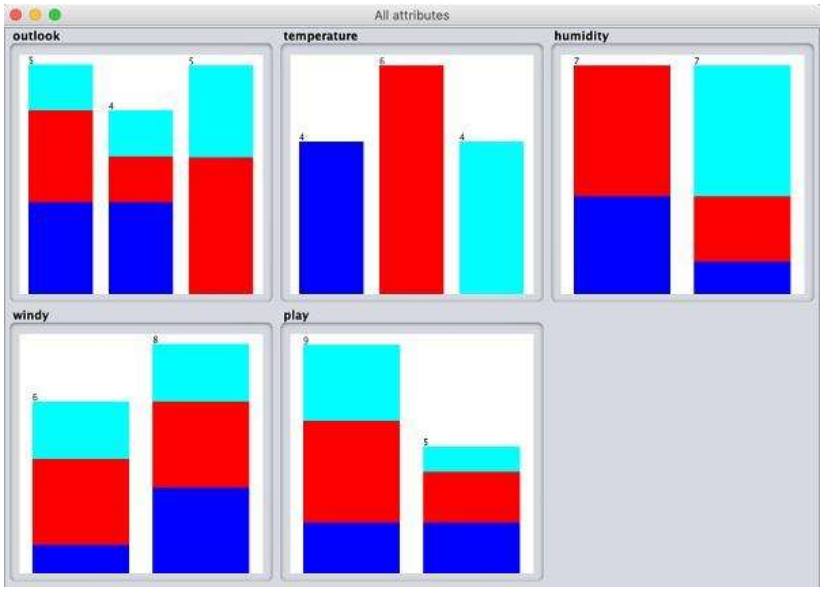


Figure 20: Visualization of Selected Data

Removing Attributes

Many a time, the data that you want to use for model building comes with many irrelevant fields. For example, the customer database may contain his mobile number which is relevant in analyzing his credit rating.

Figure 21:Attribute Removal

To remove Attribute/s select them and click on the **Remove** button at the bottom.

The selected attributes would be removed from the database. After you fully preprocess the data, you can save it for model building.

Next, you will learn to preprocess the data by applying filters to this data.

Applying Filters

Some of the machine learning techniques such as association rule mining requires categorical data. To illustrate the use of filters, we will use **weather-numeric.arff** database that contains two **numeric** attributes - **temperature** and **humidity**.

We will convert these to **nominal** by applying a filter to our raw data. Click on the **Choose** button in the **Filter** subwindow and select the following filter –

weka→filters→supervised→attribute→Discretize

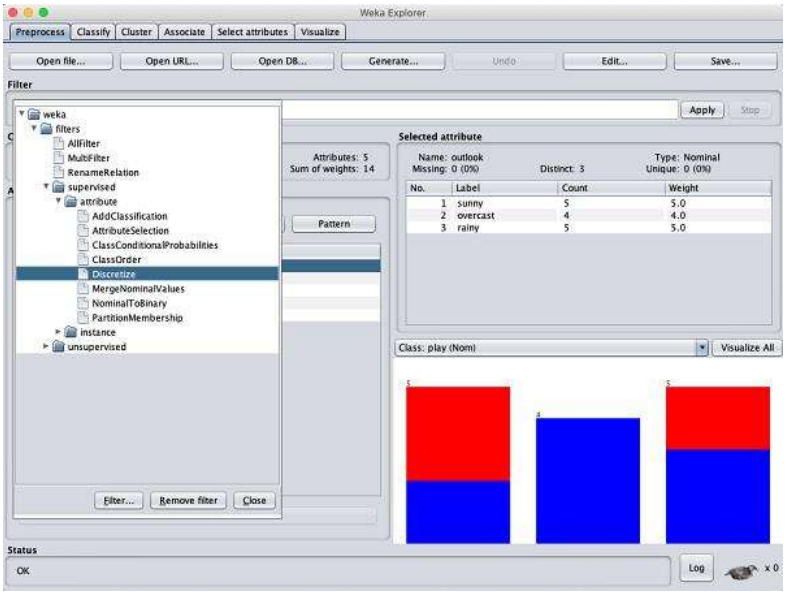


Figure 22:Application of Filters

After you are satisfied with the preprocessing of your data, save the data by clicking the **Save** ... button. You will use this saved file for model building.

5.6 Comparison between Rapid Miner and Weka

In the comparative study, I have concentrated on 2 of the commonly used tools:

- Rapid Miner
- Weka.

Features for Comparative Study

A comparative study is done based upon the following features:

- Usability
- Speed
- Visualization
- Algorithms supported
- Data Set Size
- Memory Usage
- Primary Usage
- Interface Type Supported

Table 3:Comparison

Features	Rapid Miner	Weka
Usability	Easy to use	Easiest to use.
Speed	Require more memory to operate.	Works faster on any machine.
Visualization	More options	Fewer options.
Algorithms supported	Classification and clustering	Classification and clustering
Data Set Size	Support small and large data sets.	Supports only small data sets.
Memory Usage	Requires more memory	Requires less memory and hence works faster.
Primary Usage	Data mining, predictive analysis	Machine learning
Interface type supported	GUI	GUI/CLI

Summary

- A perspective consists of a freely configurable selection of individual user interface elements, the so-called views.
- RapidMiner will eventually also ask you automatically if switching to another perspective.
- All work steps or building blocks for different data transformation or analysis tasks are called operators in RapidMiner. Those operators are presented in groups in the Operator View on the left side.
- One of the first steps in a process for data analysis is usually to load some data into the system. RapidMiner supports multiple methods for accessing datasets.
- It is always recommended to use the repository whenever this is possible instead of files.
- Open Recent Process opens the process which is selected in the list below the actions. Alternatively, you can open this process by double-clicking inside the list.
- WEKA supports many different standard data mining tasks such as data pre-processing, classification, clustering, regression, visualization and feature selection.
- The WEKA application allows novice users a tool to identify hidden information from database and file systems with simple to use options and visual interfaces.

Keywords

Process: A connected set of Operators that help you to transform and analyze your data.

Port: To build a process, you must connect the output from each Operator to the input of the next via a *port*.

Repository: your central data storage entity. It holds connections, data, processes, and results, either locally or remotely.

Operators: The elements of a Process, each Operator takes input and creates output, depending on the choice of parameters.

Filters. Processes that transform instances and sets of instances are called filters.

New Process: Starts a new analysis process.

Self Assessment Questions

1) _____ is the central RapidMiner perspective where all analysis processes are created and managed.

a) Design Perspective

b) Result Perspective

c) Welcome Perspective

d) All of the Above

2) _____Opens the repository browser and allows you to select a process to be opened within the process Design Perspective.

a) Open Template

b) Online Tutorial

c) Open Process

d) Open Recent Process

3) Which of the following contains a large number of operators for writing data and objects into external formats such as files, databases, etc.

a) Data Transformation

b) Export

c) Evaluation

d) Import

4) Rapidminer comes with :

a) Over 125 mining algorithms

b) Over 100 data cleaning and preparation functions.

c) Over 30 charts for data visualization

d) All of the above

5) _____s designed to make the preparation of data much easier

a) RapidMiner Auto Model

b) RapidMiner Turbo Prep

c) RapidMiner Studio

d) None

6) Which of the following kinds of problems can be resolved with Auto Model.

a) Prediction.

b) Clustering.

c) outliers.

d) All of the above

7) The _____ allows the automatic selection of features to create a reduced dataset.

a) Attributes Selection

b) Cluster, or Associate.

c) Classification

d) None

8) Which of the following is/are the features of RapidMiner.

a) Application & Interface

b) Data Access

c) Data Exploration

d) All

9) Which of the following is/are the features of RapidMiner.

a) attribute selection

b) Experiments

c) workflow and visualization.

d) All

10) Which of the following is the data mining tool?

a) RapidMiner

b) Weka.

c) Both a and b.

d) None

11) Two fundamental goals of Data Mining are _____.

a) Analysis and Description

b) Data cleaning and organizing the data

c) Prediction and Description

d) Data cleaning and organizing the data

12) What is WEKA?

a) Waikato Environment for Knowledge Learning

b) Waikato Environmental for Knowledge Learning

c) Waikato Environment for Knowledge Learn

d) None.

13) Which of the following statements about the query tools is correct?

a) Tools developed to query the database

b) Attributes of a database table that can take only numerical values

c) Both and B

d) None of the above

14) Which one of the following refers to the binary attribute?

a) The natural environment of a certain species

b) Systems that can be used without knowledge of internal operations

c) This takes only two values. In general, these values will be 0 and 1, and they can be coded as one bit

d) All of the above

15) Data mining tools that exist in the market are

a) Weka.

b) Rapid Miner.

c) Orange

d) All of the above

Answers: Self Assessment

1.	A	2.	C	3.	B	4.	D	5.	B
6.	D	7.	A	8.	D	9.	D	10.	C
11.	C	12.	A	13.	A	14.	C	15.	D

Review Questions

Q1) Explain different perspectives available in RapidMiner Studio?

Q2) Differentiate between RapidMiner and Weka by considering different features.

Q3) Explain the various functions available under explorer in Weka?

Q4) Elaborate on the RapidMiner GUI in detail?

Q5) Write down the different methods of creating a repository in rapidMiner?

Further Readings

Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

Markov, Z., & Russell, I. (2006). An introduction to the WEKA data mining system. ACM SIGCSE Bulletin, 38(3), 367-368.

Kotu, V., & Deshpande, B. (2014). Predictive analytics and data mining: concepts and practice with rapidminer. Morgan Kaufmann.

Hofmann, M., &Klinkenberg, R. (Eds.). (2016). RapidMiner: Data mining use cases and business analytics applications. CRC Press.

Web Links

https://www.cs.waikato.ac.nz/ml/weka/

https://www.analyticsvidhya.com/learning-paths-data-science-business-analytics-business-intelligence-big-data/weka-gui-learn-machine-learning/

https://storm.cis.fordham.edu/~gweiss/data-mining/weka.html

https://docs.rapidminer.com/9.5/server/configure/connections/creating-other-conns.html

https://docs.rapidminer.com/latest/studio/connect/

Unit 06: Extracting Data Sets

CONTENTS

Objectives

After this lecture, you will be able to

- Know the process of accessing and loading information from the Repository into the Process using retrieve Operator.
- Implementation of storage operator to store data and model.
- Various methods of visualizing the data.
- Creation of a new repository and usage of an existing repository.

Introduction

Following the directed dialogue or using the drag and drop feature to import data to your repository is easy. Simply drag the file from your file browser onto the canvas and follow the on-screen instructions. Check that the data types are right and that the goal or mark is correctly flagged. The fact that this "import then open" method is not like other methods of data opening is a significant difference.

6.1 Read Excel

Read excel operator reads an ExampleSet from the specified Excel file.

Figure 1: Read Excel operator

Data from Microsoft Excel spreadsheets can be loaded using this operator. Excel 95, 97, 2000, XP, and 2003 data can be read by this operator. The user must specify which of the workbook's spreadsheets will be used as a data table. Each row must represent an example, and each column must represent an attribute in the table.Please keep in mind that the first row of the Excel sheet can be used for attribute names that are defined by a parameter. The data table can be put anywhere on the sheet and can include any formatting instructions, as well as empty rows and columns. Empty cells or cells with only "?" can be used to show missing data values in Excel.

The Design perspective is your creative canvas and the location where you will spend the majority of your time. You will merge operators into data mining processes here.

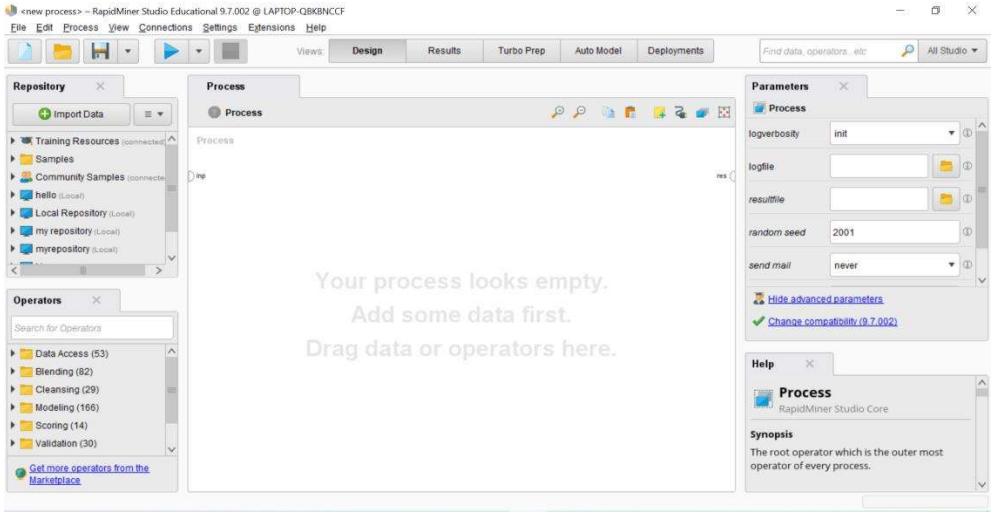


Figure 2: Design view

In the Repository view, tomake a new folder:

1. Right-click on Local Repository and select Properties.

2. Select Create Folder from the drop-down menu.

3. Give the new folder a name, such as Getting Started, and then click OK.

Connect a data folder and a processes folder to the procedure from the previous step.Your Repositories view should look similar to this:

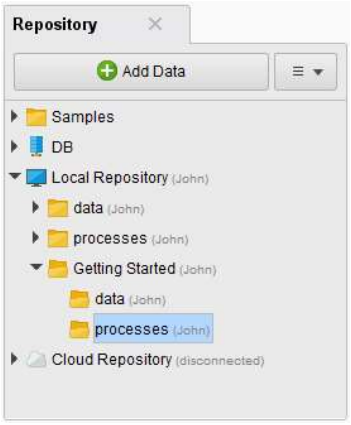


Figure 3: Repository View

The easiest and shortest way to import an Excel file is to use the *import configuration wizard* from the Parameters panel. The easiest approach, which could take a little more time, is to set all of the parameters in the Parameters panel first, then use the wizard. Before creating a method that uses the Excel file, please double-check that it can be read correctly.

To get started with RapidMiner Studio, build a local repository on your computer.

From the Repositories view, select Import Excel Sheet from the pull-down to import the training data set.

The Import Wizard launches. Browse to the location where you saved customer-churn-data.xlsx, select the file, and click Next.

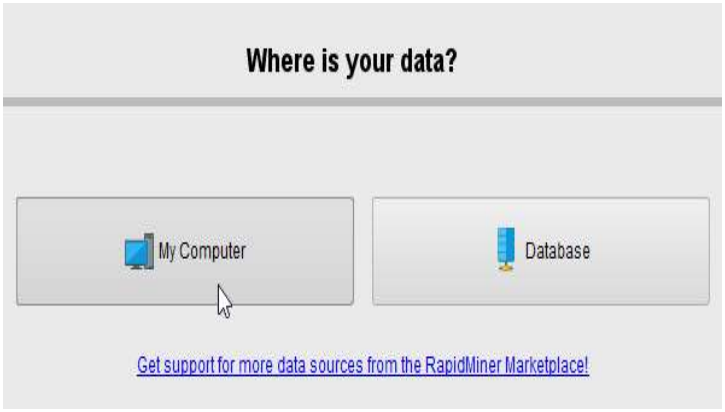
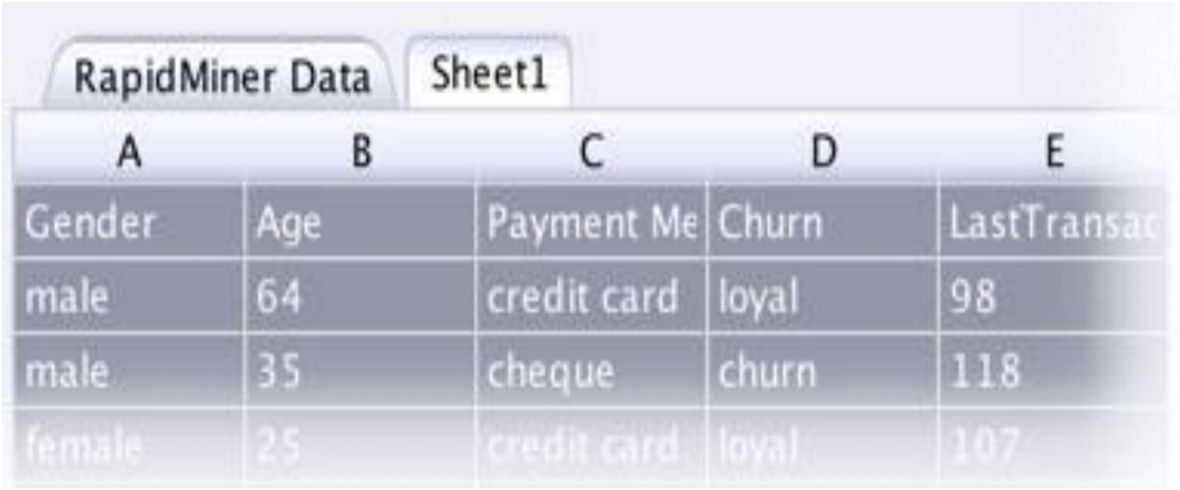
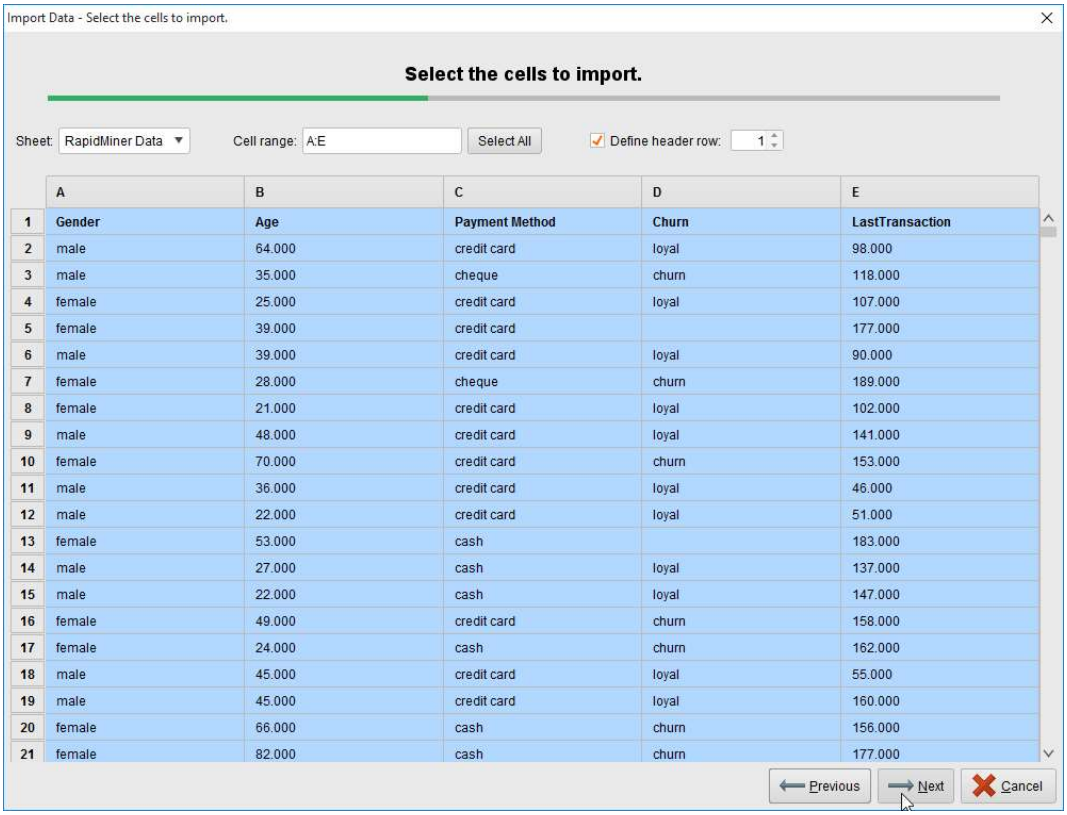


Figure 4:Data Source Selection

The wizard will walk you through the process of importing data into the repository. Verify that you're importing the right Excel sheet by looking at the tabs at the end. Although there is only one sheet in this file, RapidMiner Data, it is always a good idea to double-check.If there were more sheets, it may look like this:



Step 2 also allows you to select a range of cells for import. For this tutorial, you want all cells (the default). Click Next.

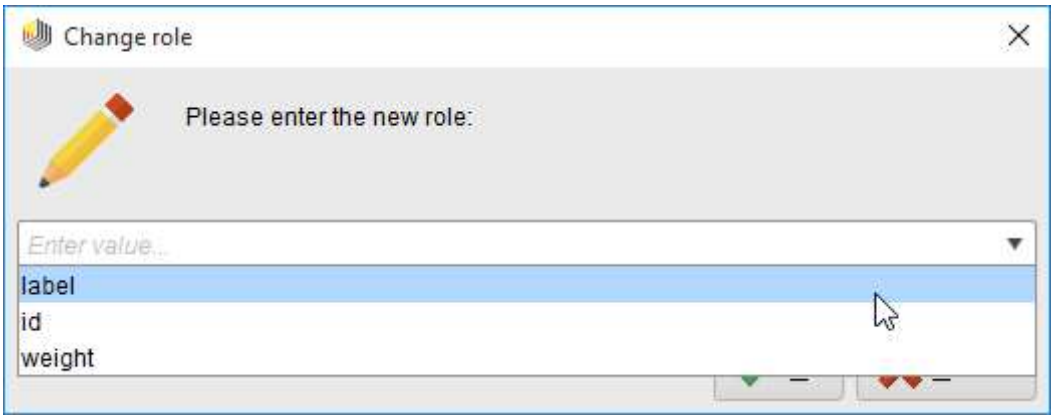


	A	B	C	D	E
1	Gender	Age	Payment Method	Churn	LastTransaction
2	male	64.000	credit card	loyal	98.000
3	male	35.000	cheque	churn	118.000
4	female	25.000	credit card	loyal	107.000
5	female	39.000	credit card		177.000
6	male	39.000	credit card	loyal	90.000
7	female	28.000	cheque	churn	189.000
8	female	21.000	credit card	loyal	102.000
9	male	48.000	credit card	loyal	141.000
10	female	70.000	credit card	churn	153.000
11	male	36.000	credit card	loyal	46.000
12	male	22.000	credit card	loyal	51.000
13	female	53.000	cash		183.000
14	male	27.000	cash	loyal	137.000
15	male	22.000	cash	loyal	147.000
16	female	49.000	credit card	churn	158.000
17	female	24.000	cash	churn	162.000
18	male	45.000	credit card	loyal	55.000
19	male	45.000	credit card	loyal	160.000
20	female	66.000	cash	churn	156.000
21	female	82.000	cash	churn	177.000

RapidMiner has preselected the first row as the row that contains column names in this process. You could fix it here if it was inaccurate, but this isn't important for your data collection. Accept the default and move on to the next step. Define the data that will be imported. The example set consists of the entire spreadsheet, with each row or example representing one customer.

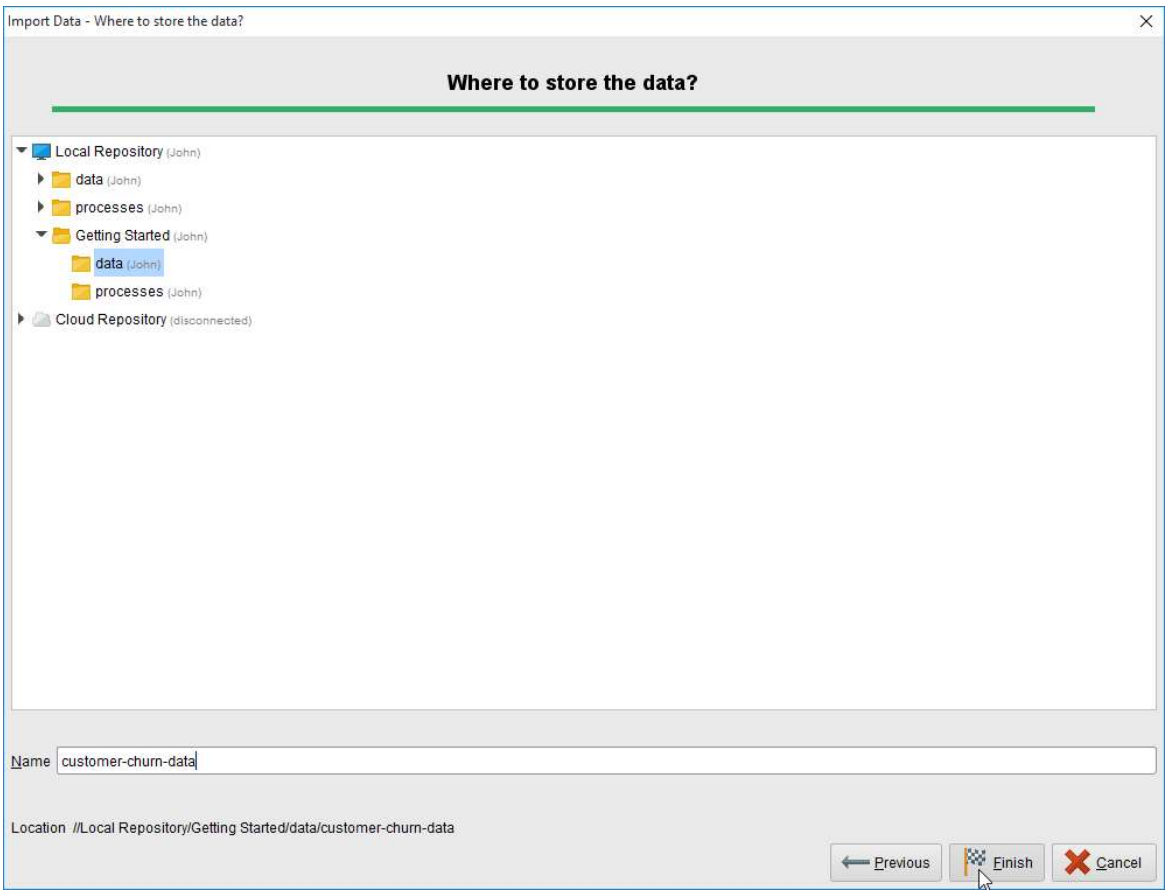
This phase has four essential components: The specifies the columns should be imported. The column names (or attributes, as RapidMiner refers to them) are those that were defined in the previous phase by the Name annotation. These are Gender, Age, Payment Method, Churn, and LastTransaction.The data types for each attribute are described by the drop-down boxes in the third row. The data type determines the values are permitted for an attribute (polynomial, numeric, integer, etc.).

The pull-down showing attribute in the fourth row is the most important task for this import. This pull-down allows you to tell RapidMiner which attribute is the main focus of your model. You set the function of Churn to mark because it is the column you want to forecast. RapidMiner can identify the example by predicting a value for each label based on what it has learned.



To finish the import, navigate to the data folder in Getting Started and give the file a name. It's worth noting that while it's in the RapidMiner archive, it'll be saved in RapidMiner's special format, which means it won't have a file name extension by convention. Click on Finish.

When you import data in RapidMiner, you need to select the attribute type "label" for the column you wish to classify.



When you click Finish, the data set loads into your repository and RapidMiner switches to the Resultsperspective, where your data displays. The following parameters you need to set to import your data successfully.

Person research tasks should be organized into new folders in the repository, which should be named appropriately.

Input

An Excel file is expected to be a file object that can be generated by using other operators that have file output ports, such as the Read File operator.

Output

This port provides a tabular version of the Excel file as well as metadata. This contribution is comparable to that of the Retrieve operator.

Parameters

- **import_configuration_wizard:** This choice allows you to use a wizard to configure this operator. This operator is simple to use thanks to this user-friendly wizard.
- **excel_file:**The path of the Excel file is specified here. It can be selected using the choose a file button.

- **sheet_selection:** This option allows you to change the sheet selection between sheet number and sheet name.
- **sheet_number:** The number of the sheet which you want to import should be specified here.Range: integer
- **sheet_name:** The name of the sheet which you want to import should be specified here.Range: string
- **imported_cell_range:** This is a mandatory parameter. The range of cells to be imported from the specified sheet is given here. It is specified in 'xm:yn' format where 'x' is the column of the first cell of the range, 'm' is the row of the first cell of the range, 'y' is the column of the last cell of the range, 'n' is the row of the last cell of the range. 'A1:E10' will select all cells of the first five columns from rows 1 to 10.
- **first_row_as_names:** If this option is set to true, it is assumed that the first line of the Excel file has the names of attributes. Then the attributes are automatically named and the first line of the Excel file is not treated as a data line.Range: boolean
- **annotations:** If the first row as names parameter is not set to true, annotations can be added using the 'Edit List' button of this parameter which opens a new menu. This menu allows you to select any row and assign an annotation to it. Name, Comment, and Unit annotations can be assigned. If row 0 is assigned Name annotation, it is equivalent to setting the first row as names parameter to true. If you want to ignore any rows you can annotate them as Comments.
- **date_format:** The date and time format are specified here. Many predefined options exist; users can also specify a new format. If text in an Excel file column matches this date format, that column is automatically converted to date type. Some corrections are automatically made in the date type values. For example, a value '32-March' will automatically be converted to '1-April'. Columns containing values that can't be interpreted as numbers will be interpreted as nominal, as long as they don't match the date and time pattern of the date format parameter. If they do, this column of the Excel file will be automatically parsed as the date and the according attribute will be of date type.
- **time_zone:** This is an expert parameter. A long list of time zones is provided; users can select any of them.
- **Locale:** This is an expert parameter. A long list of locales is provided; users can select any of them.
- **read_all_values_as_polynominal:** This option allows you to disable the type handling for this operator. Every column will be read as a polynomial attribute. To parse an excel date afterwards use 'date_parse(86400000 * (parse(date_attribute) - 25569))' (- 24107 for Mac Excel 2007) in the Generate Attributes operator.Range: boolean

- **data_set_meta_data_information:** This option is an important one. It allows you to adjust the metadata of the ExampleSet created from the specified Excel file. Column index, name, type, and role can be specified here. The Read Excel operator tries to determine an appropriate type of attribute by reading the first few lines and checking the occurring values. If all values are integers, the attribute will become an integer. Similarly, if all values are real numbers, the attribute will become of type real. Columns containing values that can't be interpreted as numbers will be interpreted as nominal, as long as they don't match the date and time pattern of the date format parameter. If they do, this column of the Excel file will be automatically parsed as the date and the according attribute will be of type date. Automatically determined types can be overridden using this parameter.
- **read_not_matching_values_as_missings:** If this value is set to true, values that do not match with the expected value type are considered as missing values and are replaced by '?'. For example, if 'back is written in an integer column, it will be treated as a missing value. A question mark (?) or an empty cell in the Excel file is also read as a missing value.Range: boolean

6.2 Store Operator

Store operator a place in the data repository where an IO Object is stored The repository entry parameter specifies the location of the object to be stored. Using the Retrieve operator, other processes may access the stored object. Please see the attached Example Processes for a basic understanding of how this operator works. In the Example, the Store operator is used to store an ExampleSet and a model.

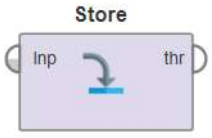


Figure 5:Store Operator

Storing an ExampleSet using the Store operator

The following procedure demonstrates how to store an ExampleSet using the Store operator. The Retrieve operator is used to load the 'Golf' and 'Golf-Testset' data sets. The Append operator is used to combine these ExampleSets. The name of the resulting ExampleSet is 'Golf-Complete,' and it is saved using the Store operator. In the third Example Method, the stored ExampleSet is used.

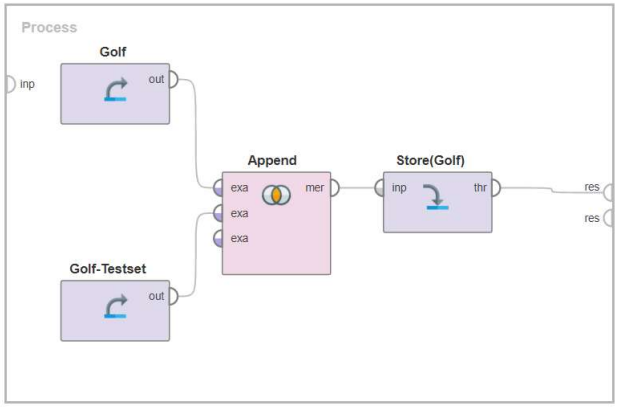


Figure 6: Storing Golf dataset using Store Operator

Create two datasets and merge the results of both datasets and store the merged results using store operator.

Storing a model using the Store operator

The following procedure demonstrates how to use the Store operator to save a model. The Retrieve operator is used to load the 'Golf' data set. It is subjected to the Naive Bayes operator, and the resulting model is saved in the repository using the Store operator. 'Golf-Naive-Style' is the name given to the model. In the third Example Method, the stored model is used.

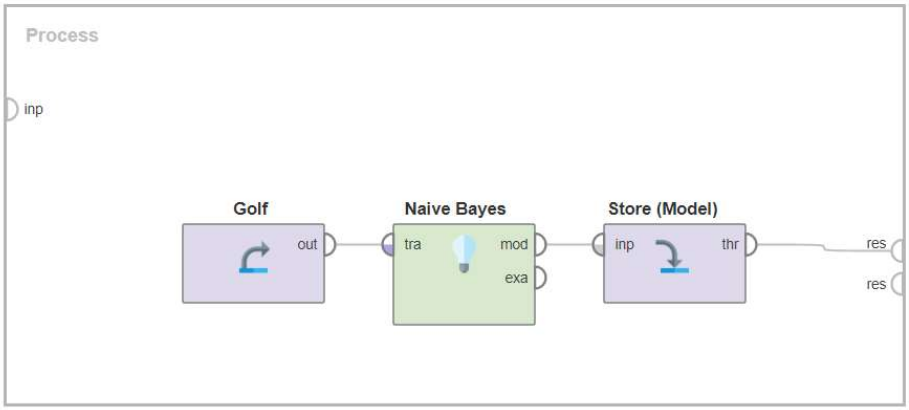


Figure 7: Storage of Naive Bayes using Store Operator

There are two quick and easy ways to store a RapidMiner model in a repository

1. Right-click on the tab in the **Results panel** and you should see an option to store the model in the repository:

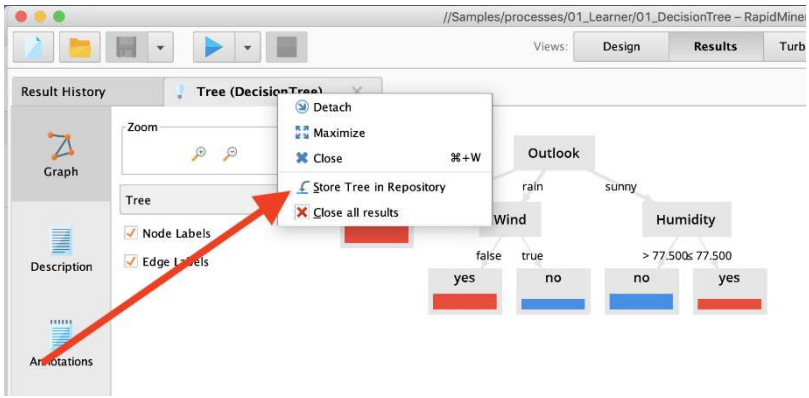


Figure 8: Storage using result panel

2.Use the Store operator on any green model "wire".

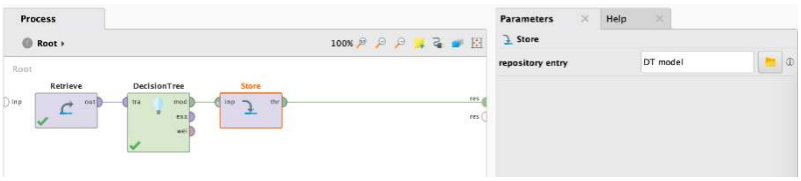


Figure 9: Store operator using Design View

Either way will get you a model object in the repository which you can use/view any time you like:

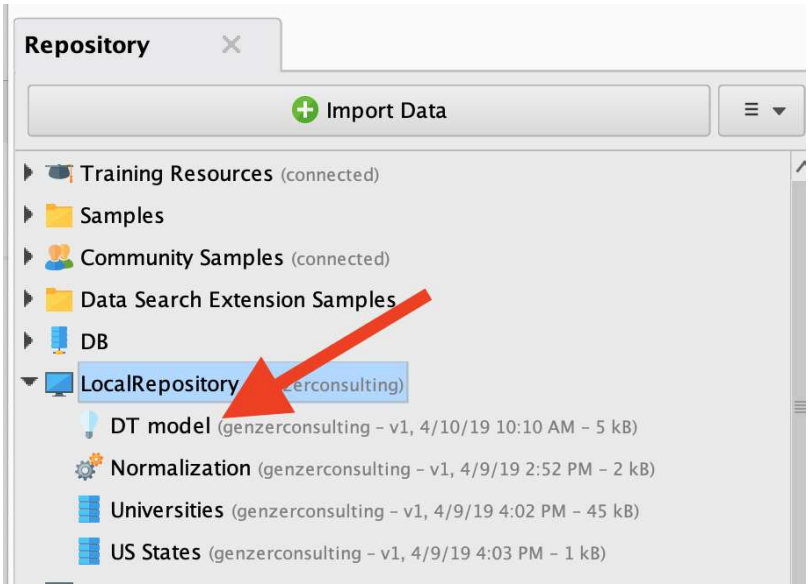


Figure 10: Storage in Repository

6.3 Retrieve Operator

The Retrieve operator a RapidMiner Object is loaded into the Process. This Object is usually an ExampleSet, but it may also be a Collection or a Model. This method of data retrieval also returns the RapidMiner Object's metadata.

Figure 11:Retrive Operator

This Operator is similar to the Data Access group's various Read source> Operators. The benefit of storing data in a repository is that metadata properties are also preserved. Additional information about the RapidMiner Object you retrieve is provided by metadata.

Output

The RapidMiner Object with the path defined in the repository entry parameter is returned.

Parameters

repository_entry

The location of the RapidMiner Object to be loaded. This parameter points to a registry entry, which will be returned as the Operator's output.

Repository locations are resolved relative to the Repository folder containing the current Process. Folders in the Repository are separated by a forward slash ('/'). A '..' references the parent folder. A leading forward slash references the root folder of the Repository containing the current Process. A leading double forward slash ('//') is interpreted as an absolute path starting with the name of a Repository. The list below shows the different methods:

- 'MyData' looks up an entry 'MyData' in the same folder as the current Process
- '../Input/MyData' looks up an entry 'MyData' located in a folder 'Input' next to the folder containing the current Process
- '/data/Model' looks up an entry 'Model' in a top-level folder 'data' in the Repository holding the current Process

- '//Samples/data/Golf' looks up the Iris data set in the 'Samples' Repository.

When using the "Select the repository location" button, it is possible to check if the path should be resolved, relative. This is useful when sharing Processes with others.

Loading of Data using the Retrieve Operator

The Golf data set is loaded from the repository using the following procedure. The repository entry parameter is '/Samples/data/Golf', which is an absolute direction. As a result, the Golf data collection and sub-folder data are returned from the Samples repository.

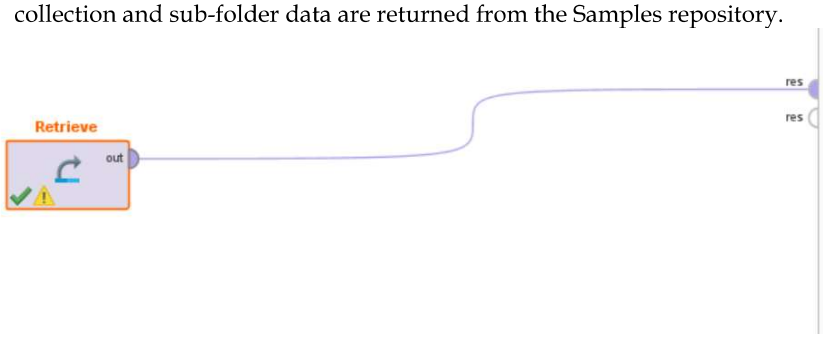


Figure 12:Data Retrieval

Create a student data set and load it in the local repository of rapidminer studio and generate the results.

6.4 Graphical Representation of data in Rapidminer

Objects located at the result ports on the right-hand side of a process are automatically reflected in the Results Perspective after the process is completed, as we've already seen. The wide area on the top left-hand side, where the Results Overview is already shown, is used for this. Iris Dataset is being considered for graphical representation. Figure 13 shows the attributes and corresponding values of the Iris dataset.

Result History | ExampleSet (Retrieve)

Data | Statistics | Visualizations | Annotations

Open in Turbo Prep | Auto Model | Filter (150 / 150 examples): all

Row No.	sepal_length	sepal_width	petal_length	petal_width	species
1	5.100	3.500	1.400	0.200	setosa
2	4.900	3	1.400	0.200	setosa
3	4.700	3.200	1.300	0.200	setosa
4	4.600	3.100	1.500	0.200	setosa
5	5	3.600	1.400	0.200	setosa
6	5.400	3.900	1.700	0.400	setosa
7	4.600	3.400	1.400	0.300	setosa
8	5	3.400	1.500	0.200	setosa
9	4.400	2.900	1.400	0.200	setosa
10	4.900	3.100	1.500	0.100	setosa
11	5.400	3.700	1.500	0.200	setosa
12	4.800	3.400	1.600	0.200	setosa
13	4.800	3	1.400	0.100	setosa

ExampleSet (150 examples, 0 special attributes, 5 regular attributes)

Figure 13:Iris Dataset

2D Scatter Plot

The resulting interface demonstrates a wide range of visualization options; in this case, we'll use RapidMiner's advanced plotting capabilities. Set the domain dimension of the iris dataset to a1, the axis to a2, and the color dimension to the mark in the advanced charts tab. Adding a dimension to the axis of a 2D scatter plot, as shown in Figure 14.

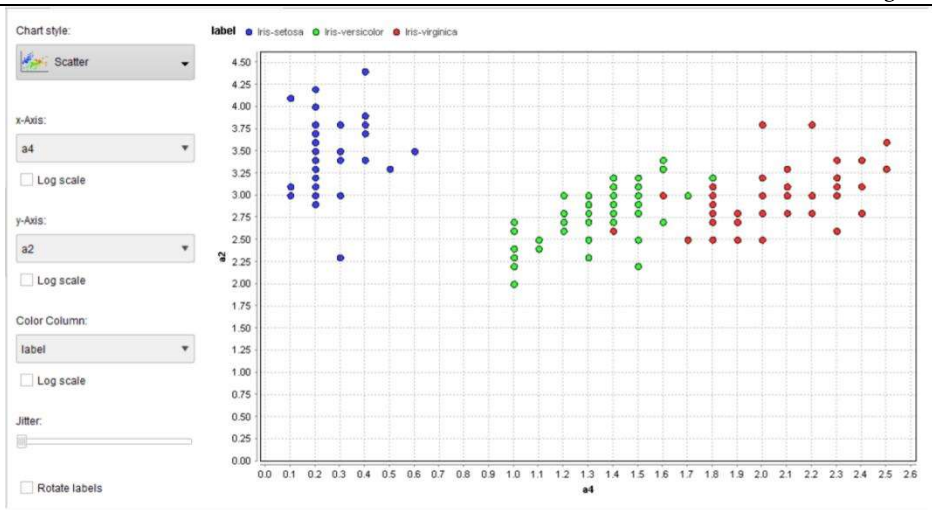


Figure 14:Scatter Plot

Line Chart

If you'd rather make a line chart, the process is the same, but you'll need to change the format to lines as shown in Figure 15.

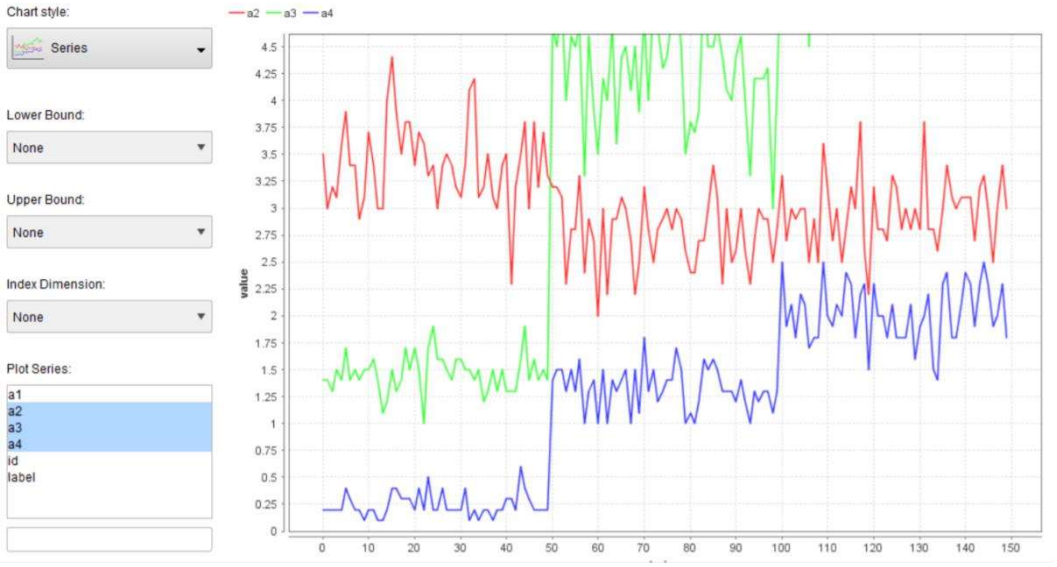


Figure 15: Line Chart

Scatter plots are most useful when the objects in the dataset are static points in time. When plotting time series, line charts are extremely useful.

Histogram

A histogram is a chart that charts the frequency of occurrence of a particular value, similar to a scatter plot. In more detail, this implies that a collection of bins is generated for one of the axes' range values. Consider the iris dataset and the a2, a3, and a4 axis once more. In RapidMiner, go to the charts tab and pick a2, a3, and a4 to plot a histogram. The number of bins can now be set to 40 as illustrated in Figure 16.

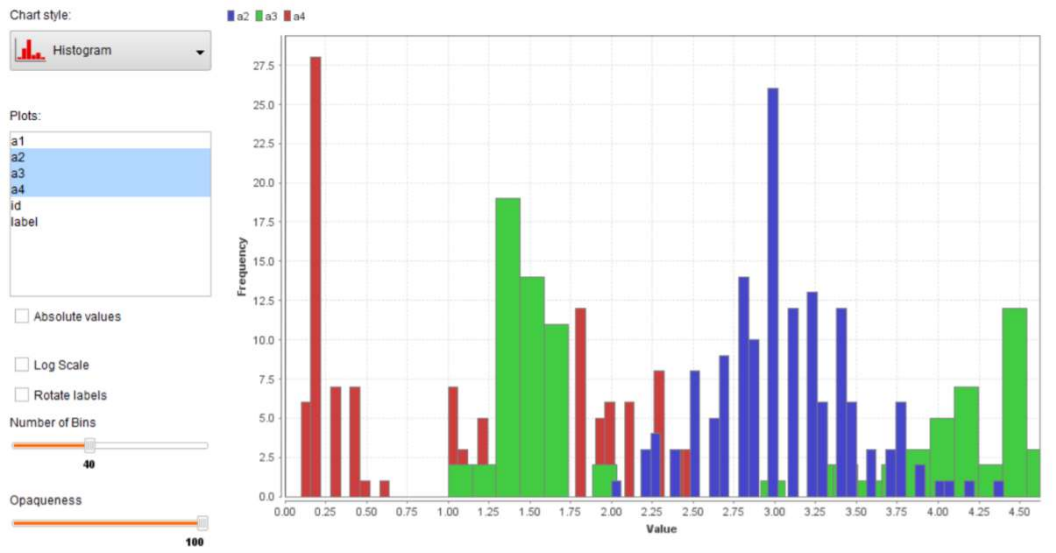


Figure 16:Histogram

Histograms are important to understand how an attribute or a set of attributes is distributed in terms of value.

Pie Chart

A pie chart is typically depicted as a circle divided into sectors, with each sector representing a percentage of a given quantity. Each of the sectors, or slices, is often annotated with a percentage indicating how much of the sum falls into each of the categories. In data analytics, pie charts can be useful for observing the proportion of data points that belong to each of the categories.

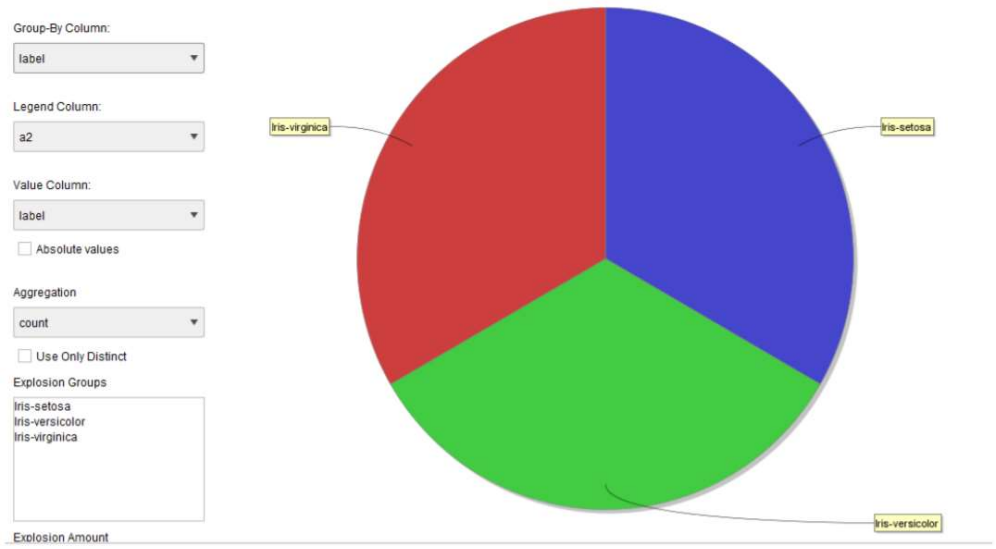


Figure 17:Pie-Chart

To make this map in RapidMiner, go to the Design tab, then to the Charts tab on the top, then to the Pie chart. Pie charts only accept one value as input, and in this case, the labeled column should be considered the input. An example of such an interface is shown in Figure 14. Pie charts, in particular, are a simple way to display how well a dataset is balanced.The Iris dataset is perfectly balanced, as you can see in Figure *17*, with the area equally split between the three classes in the dataset.

Did you Know?To display numerical data, pie graphs are circular graphs divided into sectors or slices.

Box Plots

A box plot is a convenient way to visualize data in terms of its means and quartiles. RapidMiner's box plot is shown in Figure 15. Five main statistics are shown in a box plot: minimum, first quartile, median, third quartile, and maximum. The region of the box is defined by the first and third quartiles, the minimum and maximum are indicated by whiskers, and the median is located within the box plot.White dots are commonly used to represent outliers. Box plots are a useful tool for determining if the dataset's features contain outliers. It's the first step in determining a dataset's quality and whether outlier removal methods are needed to clean it up.

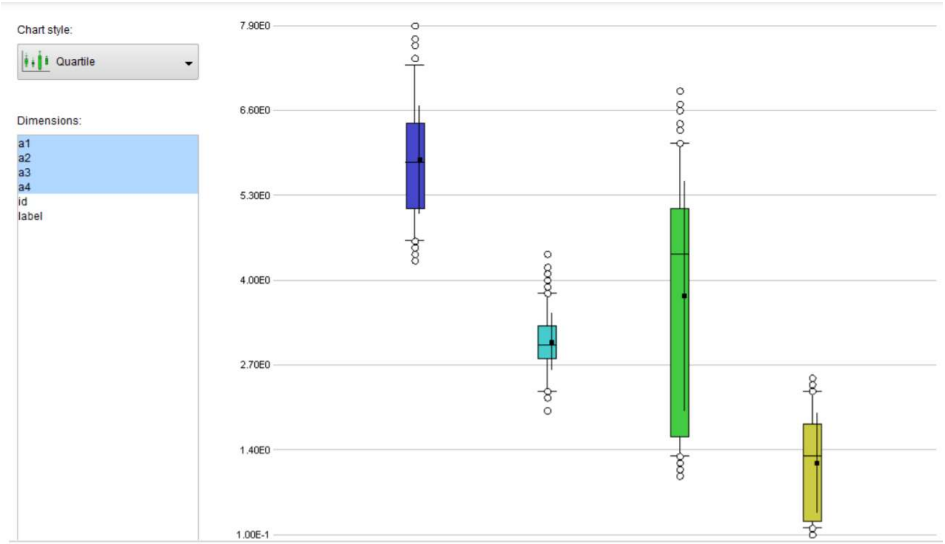
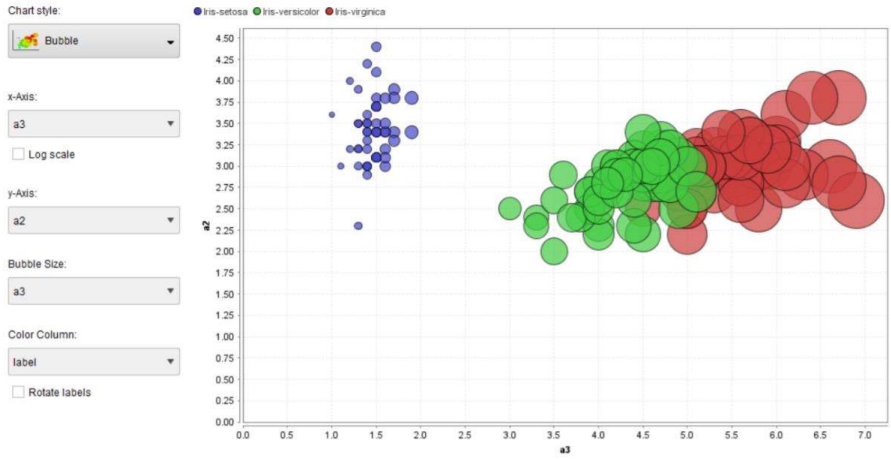


Figure 18:Box Plots

Bubble Charts

Bubble charts can also be modeled with RapidMiner. Bubble charts are a type of two-dimensional diagram that represents three-dimensional data. The ray of a circle represents the third dimension. Using the Iris dataset, Figure 16 shows an example of this diagram in RapidMiner. Bubble charts are important because they provide a two-dimensional representation of the dataset, allowing viewers to maintain a sense of the third dimension based on the size of the circle surrounding the



object.

Figure 19:Bubble Chart of Iris Dataset

Summary

- Store operator stores an IO Object in the data repository.
- The IO Object provided at the input port is delivered through this output port without any modifications.
- The stored object can be used by other processes by using the Retrieve operator.
- The behavior of an Operator can be modified by changing its *parameters*.
- There are two quick and easy ways to store a RapidMiner model in a repository

- There are other different ways of displaying a large number of results, which are also referred to as views within RapidMiner Studio.
- The Retrieve operator loads a RapidMiner Object into the Process.
- Retrieving data this way also provides the metadata of the RapidMiner Object.
- The resulting interface shows you a lot of possibilities in terms of visualization

Keywords

Operators: The elements of a Process, each Operator takes input and creates output, depending on the choice of parameters.

Parameters: Options for configuring the behavior of an Operator.

Help: Displays a help text for the current Operator.

Repository_entry: This parameter is used to specify the location where the input IO Object is to be stored.

Bubble charts: These charts are used for representing three-dimensional data in the form of a two-dimensional diagram.

Self Assessment Questions

1. The Operator that can access stored information in the Repository and load them into the Process.

a) Retrieve Operator
b) Store Operator
c) Rename Operator
d) Filter Examples Operator

2. _____Contains a large number of operators in order to read data and objects from external formats such as files, databases etc.

a) Export
b) Repository Access
c) Import
d) Evaluation

3.Data, processes, and results are stored in the_____

a) Process.
b) Repository.
c) Program
d) RapidMiner Server

4. The *Repository* can be used to store:

a) Data
b) Processes
c) Results
d) All of the above

5. Which of the following ways of getting operator into the Process Panel.

a) Drag-and-drop the Operator
b) Double-click the Operator
c) Right-click the Operator, and choose insert operator from the context menu.
d) All of the above

6. Which of the following format is used for importing data.

a) CSV File

b) Excel Sheet

c) Binary File

d) All

7. You can also load your dataset either from your local system or from a database by clicking on the _____option.

a) Import Data

b) Export Data

c) Merge Data

d) None

8. RapidMiner Studio can blend structured data with unstructured data and then leverage all the data for _____analysis.

a) Descriptive

b) Diagnostic

c) Predictive

d) Prescriptive

9. _____ is an advanced version of RapidMiner Studio that increments the process of building and validating data models.

a) RapidMiner Turbo Prep

b) RapidMiner Auto Model

c) RapidMiner Studio

d) None

10. _____ RapidMiner Turbo Prep is designed to make the preparation of data much easier.

a) RapidMiner Turbo Prep

b) RapidMiner Auto Model

c) RapidMiner Studio

d) None

11. The _____shows the individual steps within the analysis process as well as their interconnections.

a) Process View

b) Operator View

c) Repository View

d) None

12. Which of the following is/are used to define an operator?

a) The description of the expected inputs

b) The description of the supplied outputs

c) The action performed by the operator on the inputs, which ultimately leads to the supply of the outputs

d) All of the above

13. Using which of the following option we can delete the selected operator.

a) Pressing the DELETE key

b) Selecting the action "Delete" in the context menu of one of the selected operators

c) Using the menu entry "Edit" – "Delete"

d) All of the Above

14. _____rearranges all operators of the current process according to the connections and the current execution order.

a) Show and alter execution order
b) Automatic arrangement
c) Automatic size
d) Update projected metadata

15. _____is not dedicated to pre-defined descriptions but rather to your own comments on individual steps of the process.

a) Meta-Data View
b) Help View
c) Comment View
d) Problems View

Answer for Self Assessment

1.	A	2.	C	3.	B	4.	D	5.	D
6.	D	7.	A	8.	C	9.	B	10.	A
11.	A	12.	D	13.	D	14.	B	15.	C

Review Questions

Q1) How to connect with the data using Rapidminer Studio. What are the different file formats supported by Rapidminer?

Q2) Write down the different ways to store a RapidMiner model in a repository.

Q3) Create your won dataset and write down the steps on how you can import and store the data created by you using the store and import operator.

Q4) With an appropriate example explain the different ways of importing the data in rapidminer studio.

Q4) When and how to how to add a 'label' attribute to a dataset? Explain with example?

Q5) Create your repository and load data into it. Process the data and represent the results using three different charts available in rapidminer.

Q6) Write down the step-by-step procedure to display the data using a line chart.

Further Readings

Kotu, V., & Deshpande, B. (2014). Predictive analytics and data mining: concepts and practice with rapidminer. Morgan Kaufmann.

Hofmann, M., &Klinkenberg, R. (Eds.). (2016). RapidMiner: Data mining use cases and business analytics applications. CRC Press.

Ertek, G., Tapucu, D., &Arin, I. (2013). Text mining with rapidminer. RapidMiner: Data mining use cases and business analytics applications, 241.

Ryan, J. (2016). Rapidminer for text analytic fundamentals. Text Mining and Visualization: Case Studies Using Open-Source Tools, 40, 1.

Siregar, A. M., Kom, S., Puspabhuana, M. K. D. A., Kom, S., &Kom, M. (2017). Data Mining: Pengolahan Data MenjadiInformasidenganRapidMiner. CV Kekata Group.

Unit 07: Data Preprocessing

CONTENTS

Objectives

After this lecture, you will be able to

- Learn the need for preprocessing of data.
- Know the major Tasks in Data Preprocessing
- Understand the concept of missing data and various methods for handling missing data.
- Learn the concept of data discretization and data reduction.
- Understand the various strategies of data reduction.

Introduction

Preprocessing data is a data mining technique for transforming raw data into a usable and efficient format. The measures taken to make data more suitable for data mining are referred to as data preprocessing. The steps involved in Data Preprocessing are normally divided into two categories:

- Selecting data objects and attributes for analysis, and selecting data objects and attributes for analysis.
- Adding/removing attributes is a two-step process.

In data mining, data preprocessing is one of the most important aspects of the well-known information innovation from the data processor. Data taken directly from the source will contain errors, contradictions, and, most importantly, will not be willing to be used for a data mining tool.

7.1 Why Data Preprocessing?

We need to preprocess the data because of the following reasons:

1. Data in the real world is dirty which means the data is :
    - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - noisy: containing errors or outliers
    - inconsistent: containing discrepancies in codes or names

2. No quality data, no quality mining results!
    - Quality decisions must be based on quality data
    - Data warehouse needs consistent integration of quality data

Multi-Dimensional Measure of Data Quality

A well-accepted multidimensional view must have the following qualities:

- Accuracy
- Completeness
- Consistency
- Timeliness
- Believability
- Value-added
- Interpretability
- Accessibility

Accuracy: The degree to which knowledge accurately represents an event or object mentioned is referred to as "accuracy."

Example, If a customer's age is 32 but the system says she's 34, the system is inaccurate.

Completeness: When data meets comprehensiveness expectations, it is considered "complete." Assume you ask the customer to include his or her name. You can make a customer's middle name optional, but the data is full as long as you have their first and last names.

Consistency: The same information can be held in many locations at several businesses. It's called "consistent" if the information matches.

Example: ,It's inconsistent if your human resources information systems claim an employee no longer works there, but your payroll system claims he's still getting paid.

Timeliness: Is your data readily accessible when you need it? The "timeliness" factor of data quality is what it is called. Let's say you need financial data every quarter; the data is timely if it arrives when it's expected to.

7.2 Major Tasks in Data Preprocessing

The following diagram shows the various task associated with data preprocessing:

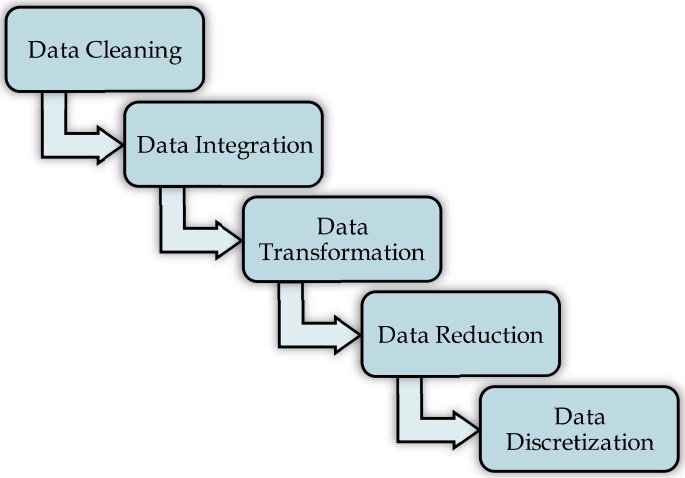


Figure 1:Data Preprocessing Tasks

Data Cleaning: Data cleaning is the method of cleaning data to make it easier to integrate.

Data Integration: Data integration is the method of bringing all of the data together.

Transformation of data: The process of transforming data into a reliable format is known as data transformation.

Reduction of data: Data reduction is the method of breaking down large amounts of data into smaller chunks so that they can be easily converted.

Discretization of data: Data discretization reduces a large number of data values to a limited number of them, making data assessment and management much easier.

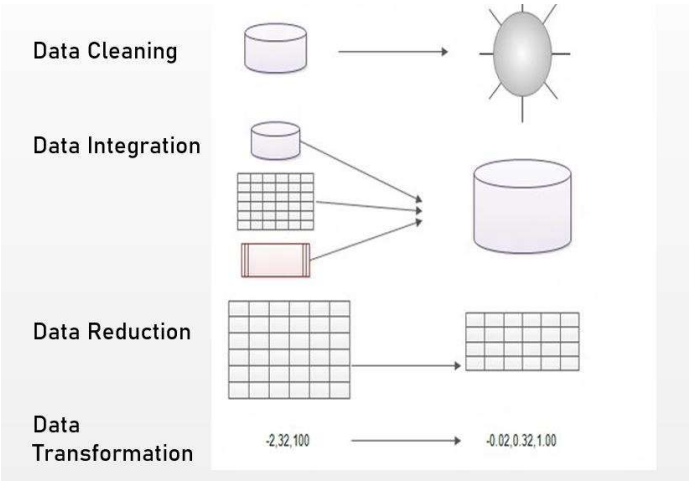


Figure 2: Forms of Data Preprocessing

Data Cleaning Task

When it comes to the final review, the quality of the data is crucial. Any data that is incomplete, noisy, or inconsistent may have an impact on your final result. The process of detecting and deleting corrupt or incorrect records from a record collection, table, or database is known as data cleaning in data mining. The following are the data cleaning tasks:

- Fill in missing values
- Identify outliers and smooth out noisy data
- Correct inconsistent data

Missing data is not always available.Missing data may be due to

- equipment malfunction
- inconsistent with other recorded data and thus deleted
- data not entered due to misunderstanding
- certain data may not be considered important at the time of entry

Example: many tuples have no recorded value for several attributes, such as customer income in sales data

Some data cleansing techniques:

1 Forget the tuple. If the classmark is absent, this is finished. Unless the tuple includes multiple attributes with missing values, this approach is ineffective.

2 You can manually fill in the missing value. This method works well with small data sets that have some missing values.

3. You may use a global constant like "Unknown" or minus infinity to replace all missing attribute values.

4 To fill in the missing value, use the attribute mean. If a customer's average income is $25,000, you can use this amount to fill in the lost income value.

5 Fill in the missing value with the most likely value.

Noisy Data

A random error or deviation in a calculated variable is referred to as noise. Noisy data may occur as a result of defective data collection instruments, data entry issues, or technological limitations.

How Do You Deal With Noisy Data?

Binning: Binning approaches sorted data values by consulting their "neighborhood," or the values in their immediate vicinity. The sorted values are divided into several "buckets" or bins.

Equal-depth (frequency) partitioning:

- It divides the range into N intervals, each containing the approximately same number of samples
- Good data scaling
- Managing categorical attributes can be tricky.

Equal-width (distance) partitioning:

- It divides the range into N intervals of equal size: uniform grid
- if A and B are the lowest and highest values of the attribute, the width of intervals will be $W = \frac{B-A}{N}$.
- The most straightforward
- But outliers may dominate the presentation
- Skewed data is not handled well.

As an example,

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

- Partition into (equi-depth) bins:

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

- Smoothing by bin means:

- Bin 1: 9, 9, 9, 9
- Bin 2: 23, 23, 23, 23
- Bin 3: 29, 29, 29, 29

- Smoothing by bin boundaries:

- Bin 1: 4, 4, 4, 15
- Bin 2: 21, 21, 25, 25
- Bin 3: 26, 26, 26, 34

Data Integration Task

Data integration is a data preprocessing technique that involves integrating data from several heterogeneous data sources into a single data store. Multiple data cubes, databases, and flat files are examples of these sources.

Data mining combines techniques from a variety of fields, including database and data warehouse technology, statistics, machine learning, high-performance computing, pattern recognition, neural networks, data visualisation, information retrieval, image and signal processing, and spatial and temporal data analysis

For data integration, there are primarily two approaches:

A. Tight Coupling

The method of ETL - Extraction, Transformation, and Loading - is used to integrate data from various sources into a single physical location in tight coupling.

B. Loose Coupling

Loose Coupling is a term that refers to a coupling that is only the data from the source databases is held in loose coupling. An interface is given in this method that takes a user's query and converts it into a format that the source database can understand, then sends the query directly to the source databases to get the response.

Data Integration Issues

We must deal with many issues while integrating the data, which are discussed below:

1. Entity Identification Problem

How do we 'match the real-world entities from the data, given that the data is unified from heterogeneous sources? We have consumer data from two separate data sources, for example. The customer id is assigned to one data source's entity, while the customer number is assigned to the other data source's entity. How does the data analyst or the machine know that these two entities lead to the same thing?

2. Redundancy and Correlation Analysis

One of the major concerns during data integration is redundancy. Unimportant data or data that is no longer needed is referred to as redundant data. It can also happen if there are attributes in the data set that can be extracted from another attribute.

Example: If one data set contains the customer's age and another data set contains the customer's date of birth, age will be a redundant attribute since the date of birth could be used to derive it.

The extent of redundancy is also increased by attribute inconsistencies. Correlation analysis can be used to find the redundancy. The attributes are evaluated to see whether they are interdependent on one another and if so, to see if there is a connection between them.

3. Data Conflict Detection and Resolution

Data conflict occurs when data from various sources is combined and does not fit. The attribute values, for example, can vary between data sets. The disparity maybe because they are depicted differently in different data sets. Assume that the price of a hotel room in different cities is expressed in different currencies. This type of problem is detected and fixed during the data integration process.

Data Transformation Task

Data is converted from one format to another which is more suitable for data mining during the data transformation process.

Listed below are a few data transformation strategies:-

1. **Smoothing:** The term "smoothing" refers to the process of removing noise from data.
2. **Aggregation:**Aggregation is the application of description or aggregation operations to data.
3. **Generalization:** Using definition hierarchies climbing, low-level data is replaced with high-level data in generalization.
4. **Normalization:** Using normalization, attribute data was scaled to fall within a narrow range, such as 0.0 to 1.0.
5. **Attribute Construction:** New attributes are created from a specified collection of attributes in attribute construction.

Data Smoothing

Smoothing data entails eliminating noise from the data collection under consideration. We've seen how techniques like binning, regression, and clustering are used to eliminate noise from results.

Binning: This approach divides the sorted data into several bins and smooths the data values in each bin based on the values in the surrounding neighborhood.

Regression: This approach determines the relationship between two dependent attributes so that we can use one attribute to predict the other.

Clustering: In this process, related data values are grouped together to create a cluster. Outliers are values that are located outside of a cluster.

Aggregation of Data

By performing an aggregation process on a large collection of data, data aggregation reduces the volume of the data set.

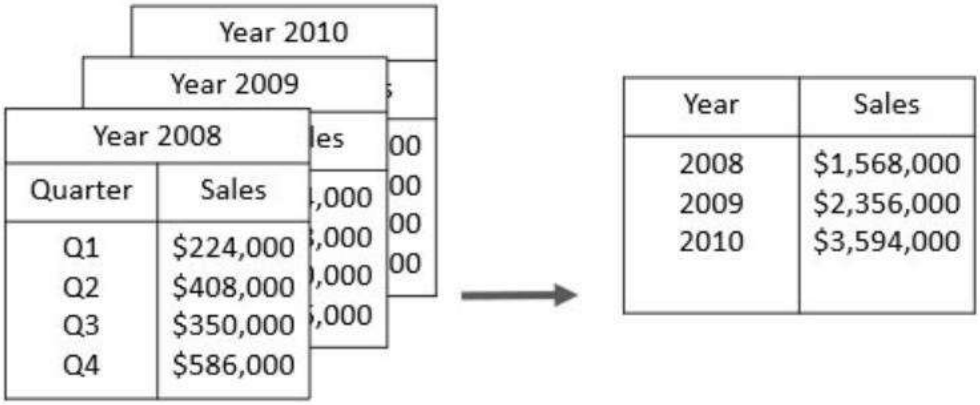


Figure 3: Aggregated Data

Example: we have a data set of sales reports of an enterprise that has quarterly sales of each year. We can aggregate the data to get the annual sales report of the enterprise.

Generalization

The nominal data or nominal attribute is a set of values with a finite number of unique values and no ordering between them. Job-category, age-category, geographicregions, items-category, and other nominal attributes are examples. By adding a group of attributes, the nominal attributes form the definition hierarchy. Concept hierarchy can be created by combining words like street, city, state, and nation.

The data is divided into several levels using a concept hierarchy. At the schema level, the definition hierarchy can be created by adding partial or complete ordering between the attributes. Alternatively, a concept hierarchy can be created by specifically grouping data on a portion of the intermediate level.

Normalization

The process of data normalization entails translating all data variables into a specific set. Data normalization is the process of reducing the number of data values to a smaller range, such as [-1, 1] or [0.0, 1.0].

The following are some examples of normalization techniques:

A. Min-Max Normalization:

Minimum-Maximum Normalization is a linear transformation of the original results. Assume that the minima and maxima of an attribute are min A and max A, respectively.

We Have the Formula:

$$v' = \frac{v - min_P}{max_P - min_P}(new\_max_P - new\_min_P) + new\_min_P$$

Where v is the value in the new range that you want to plot. After normalizing the old value, you get v', which is the new value. For example, the minimum and maximum values for the attribute 'income' are $1200 and $9800, respectively, and the range in which we would map a value of $73,600 is [0.0, 1.0]. The value of $73,600 will be normalized using the min-max method:

$$\frac{73600-1200}{9800-1200}(1.0-0.0)+0.0=0.716$$

B. Z-score Normalization

Using the mean and standard deviation, this approach normalizes the value for attribute A. The formula is as follows:

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

Here Ā and σAare the mean and standard deviation for attribute A are respectively.For instance, attribute A now has a mean and standard deviation of $54,000 and $16,000, respectively. We must also use z-score normalization to normalize the value of $73,600.

$$\frac{73600-5400}{1600}=1.225$$

C. Decimal Scaling

By shifting the decimal point in the value, this method normalizes the value of attribute A. The maximum absolute value of A determines the movement of a decimal point.

The decimal scaling formula is as follows:

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

Here j is the smallest integer such that max(|v'i|)<1.

For example, the observed values for attribute A range from -986 to 917, with 986 as the maximum absolute value. To use decimal scaling to normalize each value of attribute A, we must divide each value of attribute A by 1000, i.e. j=3. As a result, the value -986 is normalized to -0.986, and the value 917 is normalized to 0.917.

To uniformly normalize future data, normalization parameters such as mean, standard deviation, and maximum absolute value must be maintained.

Attribute Construction

In the attribute construction process, new attributes are created by consulting an existing collection of attributes, resulting in a new data set that is easier to mine. Consider the following scenario: we have a data set containing measurements of various plots, such as the height and width of each plot. So, using the attributes 'height' and 'weight,' we can create a new attribute called 'area.' This often aids in the comprehension of the relationships between the attributes in a data set.

7.3 Data Discretization

Data discretization encourages data transformation by replacing integer data values with interval marks. The interval labels (0-10, 11-20...) or the interval labels (0-10, 11-20...) may be used to substitute the values for the attribute "age" (kid, youth, adult, senior).Data discretization can be classified into two types **supervised discretization** where the class information is used and the other is **unsupervised discretization** which is based on which direction does the process proceed i.e. 'top-down splitting strategy' or 'bottom-up merging strategy'. For Discretization different attributes needs consideration which is as follows:

Nominal Attribute: Nominal Attributes only provide enough information to distinguish one object from another. For example, the person's sex and the student's roll number.

Ordinal Attribute: The value of the ordinal attribute is sufficient to order the objects. Rankings, grades, and height are only a few examples.

Numeric Attributes: It is quantitative, in the sense that quantity can be calculated and expressed in integer or real numbers.

Ratio Scaled attribute: A proportion Ratio is a scaled parameter that is important for both inequalities and ratios. For example, age, height, and weight.

Types of Data Discretization

Top-down Discretization

Top-down discretization or splitting is when a process begins by finding one or a few points (called breakpoints or cut points) to split the entire attribute set, and then repeats the process recursively on the resulting intervals.

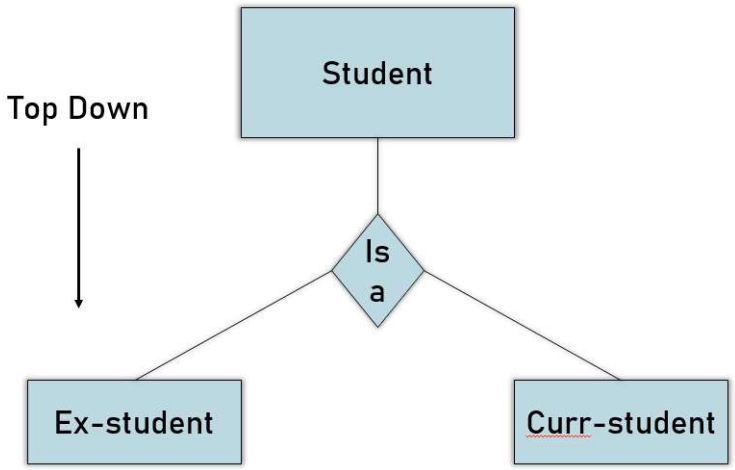


Figure 4: Top-Down Discretization

Bottom-up Discretization

Bottom-up discretization or merging is described as a process that begins by considering all continuous values as potential split-points and then removes some by merging neighborhood values to form intervals.

Discretization on an attribute can be done quickly to create a definition hierarchy, which is a hierarchical partitioning of the attribute values.

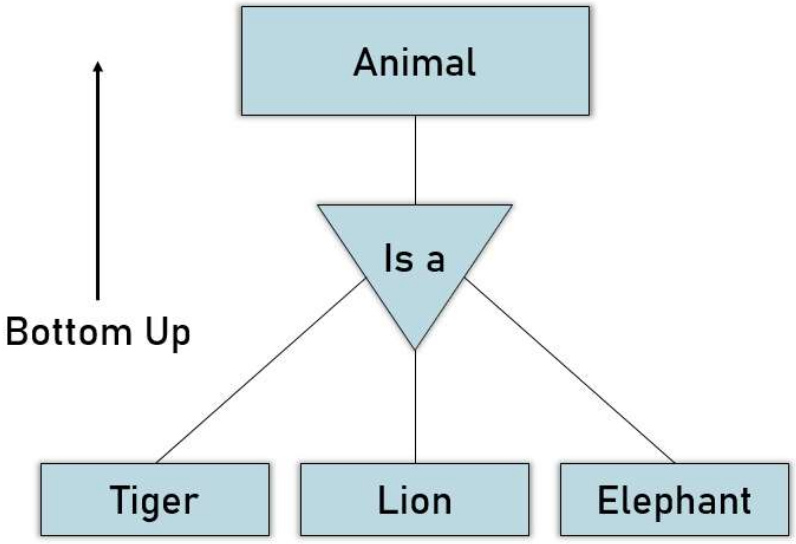


Figure 5: Bottom-Up Discretization

Supervised Discretization

Supervised discretization is when you take the class into account when making discretization boundaries.The discretization must be determined solely by the training set and not the test set.

Unsupervised Discretization

Unsupervised discretization algorithms are the simplest algorithms to make use of, because the only parameter you would specify is the number of intervals to use; or else, how many values should be included in each interval.

Discuss the significance of supervised and un-supervised discretization.

7.4 Need of Data Reduction

Terabytes of data may be stored in a database or data center. As a result, data processing and mining on such massive datasets can take a long time.

Data reduction methods may be used to create a smaller-volume representation of the data set that still contains important information. The data reduction process reduces the size of data and makes it suitable and feasible for analysis. In the reduction process, the integrity of the data must be preserved, and data volume is reduced. Many strategies can be used for data reduction.

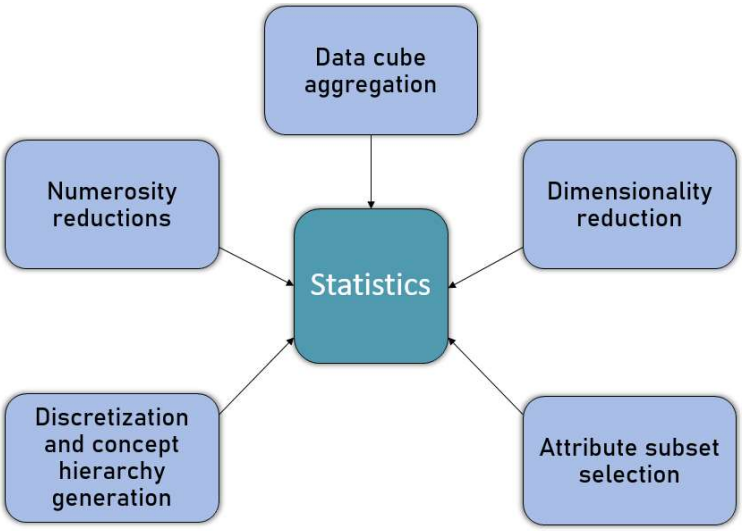


Figure 6: Data Reduction Strategies

1. **Data Cube Aggregation:** In the construction of a data cube, aggregation operations are applied to the data. This method is used to condense data into a more manageable format. As an example, consider the data you collected for your study from 2012 to 2014, which includes your company's revenue every three months. Rather than the quarterly average, they include you in the annual revenue.

2. **Dimension Reduction:**We use the attribute needed for our analysis whenever we come across data that is weakly significant. It shrinks data by removing obsolete or redundant functions. The following are the methods used for Dimension reduction:

- Step-wise Forward Selection
- Step-wise Backward Selection
- Combination of forwarding and Backward Selection

Step-wise Forward Selection

The selection begins with an empty set of attributes later on we decide the best of the original attributes on the set based on their relevance to other attributes.

Initial attribute Set: {X1, X2, X3, X4, X5, X6}

Initial reduced attribute set: { }

- Step-1: {X1}

- Step-2: {X1, X2}
- Step-3: {X1, X2, X5}

Final reduced attribute set: {X1, X2, X5}

Step-wise Backward Selection

This selection starts with a set of complete attributes in the original data and at each point, it eliminates the worst remaining attribute in the set.

Initial attribute Set: {X1, X2, X3, X4, X5, X6}

Initial reduced attribute set: {X1, X2, X3, X4, X5, X6 }

- Step-1: {X1, X2, X3, X4, X5}
- Step-2: {X1, X2, X3, X5}
- Step-3: {X1, X2, X5}

Final reduced attribute set: {X1, X2, X5}

Combination of forwarding and backward selection

It helps us to eliminate the worst attributes and pick the better ones, saving time and speeding up the process.

Data Compression

Using various encoding methods, the data compression technique reduces the size of files (Huffman Encoding & run-length Encoding). Based on the compression techniques used, we can split it into two forms.

- Lossless Compression
- Lossy Compression

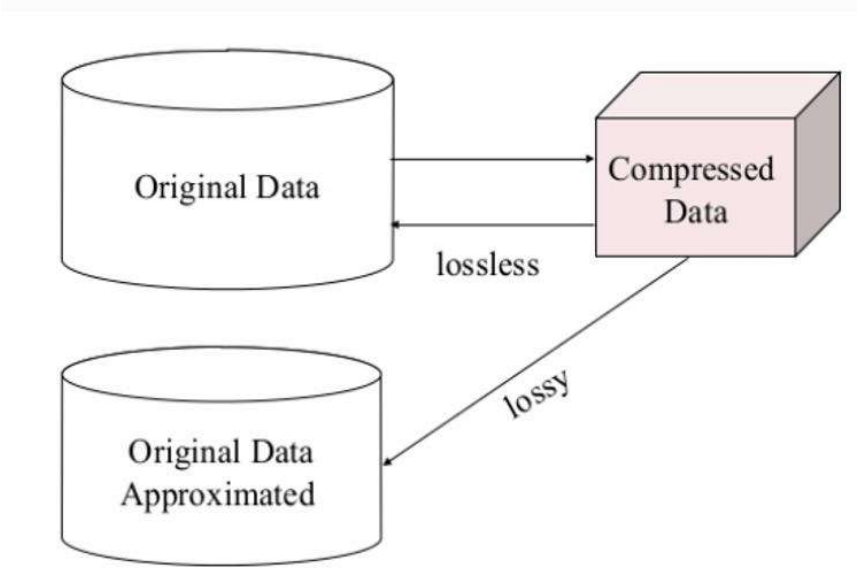


Figure 7: Data Compression Techniques

Lossless Compression – Encoding techniques (Run Length Encoding) allow for a quick and painless data reduction. Algorithms are used in lossless data compression to recover the exact original data from the compressed data.

Lossy Compression – Examples of this compression include the Discrete Wavelet Transform Technique and PCA (principal component analysis). JPEG image format, for example, is a lossy compression, but we can find a sense equivalent to the original image. The decompressed data in lossy data compression vary from the original data, but they are still useful for retrieving information.

Numerosity Reduction

Numerosity Reduction is a data reduction strategy that uses a smaller type of data representation to replace the original data. There are two approaches for reducing numerosity: parametric and non-parametric.

Parametric Methods: Parametric methods use a model to represent data. The model is used to estimate data, requiring only data parameters to be processed rather than real data. These models are built using regression and log-linear methods.

- **Regression**

Easy linear regression and multiple linear regression are two types of regression. While there is only one independent attribute, the regression model is referred to as simple linear regression, and when there are multiple independent attributes, it is referred to as multiple linear regression.

- **Log-Linear Model**

  Based on a smaller subset of dimensional combinations, a log-linear model may be used to estimate the likelihood of each data point in a multidimensional space for a collection of discretized attributes. This enables the development of a higher-dimensional data space from lower-dimensional attributes.


Did you Know? On sparse data, both regression and the log-linear model can be used, but their implementation is limited.

Non-Parametric Methods: Histograms, clustering, sampling, and data cube aggregation are examples of non-parametric methods for storing reduced representations of data.

- **Histograms:** A histogram is a frequency representation of data. It is a common method of data reduction that employs binning to approximate data distribution.
- **Clustering:** Clustering is the division of data into classes or clusters. This method divides all of the data into distinct clusters. The cluster representation of the data is used to replace the actual data in data reduction. It also aids in the detection of data outliers.
- **Sampling:** Sampling is a data reduction technique that allows a large data set to be represented by a much smaller random data set.
- **Aggregation of Data Cubes**: Aggregation of data cubes entails transferring data from a complex level to a smaller number of dimensions. The resulting data set is smaller in size while retaining all of the details required for the analysis mission.

Summary

- Data transformation is the process of converting data into a format that allows for effective data mining.
- Normalization, discretization, and concept hierarchy are the most powerful ways of transforming data.
- Data normalization is the process of growing the collection of data.
- The data values of a numeric variable are replaced with interval labels when data discretization is used.
- The data is transformed into multiple l by using concept hierarchy.
- Accuracy, completeness, continuity, timeliness, believability, and interpretability are all w ords used to describe data quality. These qualities are evaluated based on the data's intend ed use.
- Data cleaning routines aim to fix errors in the data by filling in missing values, smoothing out noise when finding outliers, and filling in missing values. Cleaning data is normally done in two steps, one after the other.

Keywords

Data cleaning: To remove noise and inconsistent data.

Data integration: Multiple data sources may be combined.

Data transformation: Where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.

Data discretization: Ittransforms numeric data by mapping values to interval or concept

labels.

Self Assessment Questions

1. What is the use of data cleaning?

A. To remove the noisy data

B. Correct the inconsistencies in data

C. Transformations to correct the wrong data.

D. All of the above

2. Data cleaning is

A. Large collection of data mostly stored in a computer system

B. The removal of noise errors and incorrect input from a database

C. The systematic description of the syntactic structure of a specific database. It describes the structure of the attributes of the tables and foreign key relationships.

D. None of these

3. Which of the following is not a data pre-processing methods

A. Data Visualization

B. Data Discretization

C. Data Cleaning

D. Data Reduction

4. Dimensionality reduction reduces the data set size by removing _____

A. composite attributes

B. derived attributes

C. relevant attributes

D. irrelevant attributes

5. Which of the following activities is a data mining task?

A. Monitoring the heart rate of a patient for abnormalities

B. Extracting the frequencies of a sound wave

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

D. Dividing the customers of a company according to their profitability

6. Which of the following is NOT an example of ordinal attributes?

A. Ordered numbers

B. Movie ratings

C. Zipcodes

D. Military ranks

7. Which data mining task can be used for predicting wind velocities as a function of temperature, humidity, air pressure, etc.?

A. Cluster Analysis

B. Regression

C. Classification

D. Sequential pattern discovery

8. Which of the following is not a data mining task? Select one:

A. Feature Subset Detection

B. Association Rule Discovery

C. Regression

D. Sequential Pattern Discovery

9. _____Combines data from multiple sources into a coherent store.

A. Data integration

B. Data reduction

C. Data Transformation

D. Data cleaning

10. Careful integration can help reduce and avoid _____and inconsistencies in resulting data set.

A. Noise

B. Redundancies

C. Error

D. None

11. _____is the process of changing the format, structure, or values of data.

A. Data integration

B. Data reduction

C. Data Transformation

D. Data cleaning

12. In Binning, we first sort data and partition into (equal-frequency) bins and then which of the following is not a valid step

A. smooth by bin boundaries

B. smooth by bin median

C. smooth by bin means

D. smooth by bin values

13. Which of the following is NOT a data quality-related issue?

A. Missing values

B. Outlier records

C. Duplicate records

D. Attribute value range

14. Which of the following is used to remove noise from data.

A. Generalization

B. Normalization

C. Aggregation

D. Smoothing

15. IN _____ Encoding mechanisms are used to reduce the data set size.

A. Numerosity Reduction

B. Data Compression

C. Dimensionality Reduction

D. Data Cube Aggregation

Answers for Self Assessment

1.	D	2.	B	3.	A	4.	D	5.	A
6.	C	7.	B	8.	A	9.	A	10.	B
11.	C	12.	D	13.	C	14.	D	15.	B

Review Questions

Q1) Data quality can be assessed in terms of several issues, including accuracy, completeness, and consistency. For each of the above three issues, discuss how data quality assessment can depend on the intended use of the data, giving examples. Propose two other dimensions of data quality.

Q2) In real-world data, tuples with missing values for some attributes are a common occurrence. Describe various methods for handling this problem.

Q3) Discuss issues to consider during data integration.

Q4) For example explain the various methods of Normalization.

Q5) Elaborate on various data reduction strategies by giving the example of each strategy.

Q6) Explain the concept of data discretization along with its various methods.

Q7) What is the need for data preprocessing? Explain the different data preprocessing tasks in detail.

Further Readings

Kotu, V., & Deshpande, B. (2014). Predictive analytics and data mining: concepts and practice with rapidminer. Morgan Kaufmann.

Hofmann, M., &Klinkenberg, R. (Eds.). (2016). RapidMiner: Data mining use cases and business analytics applications. CRC Press.

Ertek, G., Tapucu, D., &Arin, I. (2013). Text mining with rapidminer. RapidMiner: Data mining use cases and business analytics applications, 241.

Ryan, J. (2016). Rapidminer for text analytic fundamentals. Text Mining and Visualization: Case Studies Using Open-Source Tools, 40, 1.

Siregar, A. M., Kom, S., Puspabhuana, M. K. D. A., Kom, S., &Kom, M. (2017). Data Mining: Pengolahan Data MenjadiInformasidenganRapidMiner. CV Kekata Group

https://towardsdatascience.com/data-preprocessing-in-data-mining-machine-learning-79a9662e2eb

https://www.geeksforgeeks.org/numerosity-reduction-in-data-mining/

https://cs.ccsu.edu/~markov/ccsu_courses/datamining-3.html

https://www.cse.wustl.edu/~zhang/teaching/cs514/Spring11/Data-prep.pdf

Unit 08: Data Preprocessing Using Rapid Miner

CONTENTS

Objectives

After this lecture, you will be able to

- Learn the need for preprocessing of data.
- Need and implementation of Remove Duplicate Operator.
- Learn the implementation of Different methods of Handing Missing Values.
- Understand the various methods of renaming an attribute.
- Learn the implementation of the Apriori Algorithm in Weka.

Introduction

Data pre-processing is a data mining technique that entails converting raw data into a format that can be understood. Real-world data is often incomplete, unreliable, and/or deficient in specific habits or patterns, as well as containing numerous errors. The most important information at the pre-processing phase is about the missing values.Preprocessing data is a data mining technique for transforming raw data into a usable and efficient format. The measures taken to make data more suitable for data mining are referred to as data preprocessing. The steps involved in Data Preprocessing are normally divided into two categories:

- Selecting data objects and attributes for analysis, and selecting data objects and attributes for analysis.
- Adding/removing attributes is a two-step process.

8.1 Remove Duplicate Operator

The Remove Duplicates operator compares all examples in an Example Set against each other using the listed attributes to remove duplicates. This operator eliminates duplicate examples one by one, leaving only one duplicate example. If the selected attributes have the same values, two instances are called duplicates.

Figure 1: Remove Duplicate Operator

The attribute filter type parameter and other related parameters can be used to select attributes. Assume that two attributes, 'att1' and 'att2,' are chosen, with three and two possible values, respectively. As a result, there are a total of six (three times two) special combinations of these two attributes. As a consequence, the resulting Example Set can only have 6 examples. This operator applies to all attribute types. The following Iris dataset contains 150 examples before the use of remove duplicate operator. The following are the various parameters applicable to remove duplicate operators.

Consider any dataset from sample dataset in Rapid miner and implement Remove Duplicate operator on different attributes.

Input

An Example Set is expected at this input port. In the attached Example Method, it's the result of the Retrieve operator. Other operators' output can also be used as data.

Output

- Example set Output(Data Table)

The duplicate examples in the provided Example Set are removed, and the resulting Example Set is delivered via this port.

- Original (Data Table)

This port passes the Example Set that was provided as input to the output without any changes. This is commonly used to reuse the same Example Set through several operators or to display the Example Set in the Results Workspace.

- Duplicates (Data Table)

This port is used to deliver duplicated examples from the given Example Set.

Parameters

attribute_filter_type

This parameter lets you choose the attribute selection filter, which is the tool you'll use to pick the appropriate attributes. It comes with the following features:

- **All**: This choice simply selects all of the Example Set's attributes. This is the standard-setting.
- **single:** Selecting a single attribute is possible with this choice. When you select this choice, a new parameter (attribute) appears in the Parameters panel.
- **subset:** This choice allows you to choose from a list of multiple attributes. The list contains all of the Example Set's attributes; appropriate attributes can be easily selected. If the metadata is unknown, this option will not work.
- **regular_expression:** This feature allows you to pick attributes using a standard expression. Other parameters (regular expression, use except for expression) become available in the Parameters panel when this choice is selected.

- **value_type:** This option allows you to pick all of a type's attributes. It's worth noting that styles are arranged in a hierarchy. Real and integer types, for example, are also numeric types. When selecting attributes via this option, users should have a basic understanding of type hierarchy. Other parameters in the Parameters panel become available when this choice is selected.
- **block_type:** This option works similarly to the value type option. This choice allows you to choose all of the attributes for a specific block form. Other parameters (block type, use block type exception) become available in the Parameters panel when this choice is selected.
- **no_missing_values**: This choice simply selects all of the Example Set's attributes that do not have a missing value in any of the examples. All attributes with a single missing value are deleted.
- **numeric value filter:** When the numeric value filter choice is selected, a new parameter (numeric condition) appears in the Parameters panel. The examples of all numeric attributes that satisfy the numeric condition are chosen. Please note that regardless of the numerical situation, all nominal attributes are chosen.

attribute

This choice allows you to choose the desired attribute. If the metadata is identified, the attribute name can be selected from the attribute parameter's drop-down box.

attributes

This choice allows you to pick the appropriate attributes. This will bring up a new window with two lists. The left list contains all attributes, which can be transferred to the right list, which contains the list of selected attributes for which the conversion from nominal to numeric will be performed; all other attributes will remain unchanged.

Regular_expression

This expression will be used to pick the attributes whose names match this expression. Regular expressions are a powerful tool, but they require a thorough introduction for beginners. It's always a good idea to use the edit and display regular expression menu to define the regular expression.

value_type

A drop-down menu allows you to pick the type of attribute you want to use. You may choose one of the following types: nominal, text, binominal, polynomial, or file path.

The following figure shows the design view of the Iris dataset.

block_type

A drop-down list may be used to choose the block type of attributes to be used. 'single value' is the only possible value.

numeric_condition

This is where you specify the numeric condition for checking examples of numeric attributes. The numeric condition '> 6', for example, will hold all nominal and numeric attributes with a value greater than 6 in any example. '> 6 && 11' or '= 5 || 0' are examples of potential situations. However, you can't use && and || in the same numeric situation.

Did you know? Conditions like '(> 0 && 2) || (>10 && 12)' are not authorized, since they use both && and ||. After '>', '=', and '', use a blank space; for example, '5' will not work; instead, use '5'.

include_special_attributes

The examples are identified by unique attributes, which are attributes with specific functions. Regular attributes, on the other hand, simply define the instances. Id, label, prediction, cluster, weight, and batch are special attributes.

invert_selection

When set to real, this parameter acts as a NOT gate, reversing the selection. In that case, all of the previously selected attributes are unselected, while previously unselected attributes are selected. Before checking this parameter, for example, if attribute 'att1' is selected and attribute 'att2' is unselected. 'att1' will be unselected after testing this parameter, while 'att2' will be selected.

treat_missing_values_as_duplicates

This parameter determines whether or not missing values should be considered duplicates. If valid, missing values are treated the same as duplicate values.

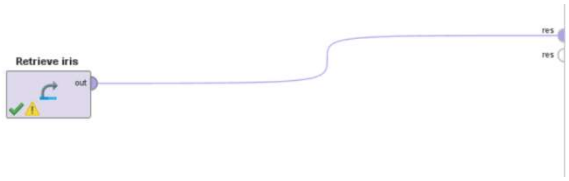


Figure 2:Design View

Figure 3 shows the result view of the Iris dataset.

Open in Turbo Prep Auto Model Filter (150 / 150 examples): all

Row No.	sepal_length	sepal_width	petal_length	petal_width	species
1	5.100	3.500	1.400	0.200	setosa
2	4.900	3	1.400	0.200	setosa
3	4.700	3.200	1.300	0.200	setosa
4	4.600	3.100	1.500	0.200	setosa
5	5	3.600	1.400	0.200	setosa
6	5.400	3.900	1.700	0.400	setosa
7	4.600	3.400	1.400	0.300	setosa
8	5	3.400	1.500	0.200	setosa
9	4.400	2.900	1.400	0.200	setosa
10	4.900	3.100	1.500	0.100	setosa
11	5.400	3.700	1.500	0.200	setosa
12	4.800	3.400	1.600	0.200	setosa
13	4.800	3	1.400	0.100	setosa

ExampleSet (150 examples, 0 special attributes, 5 regular attributes)

Figure 3: Iris Dataset before removal of Duplicates

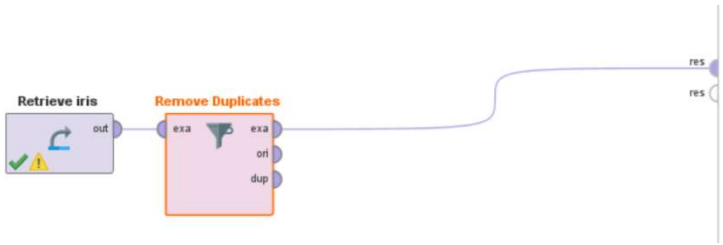
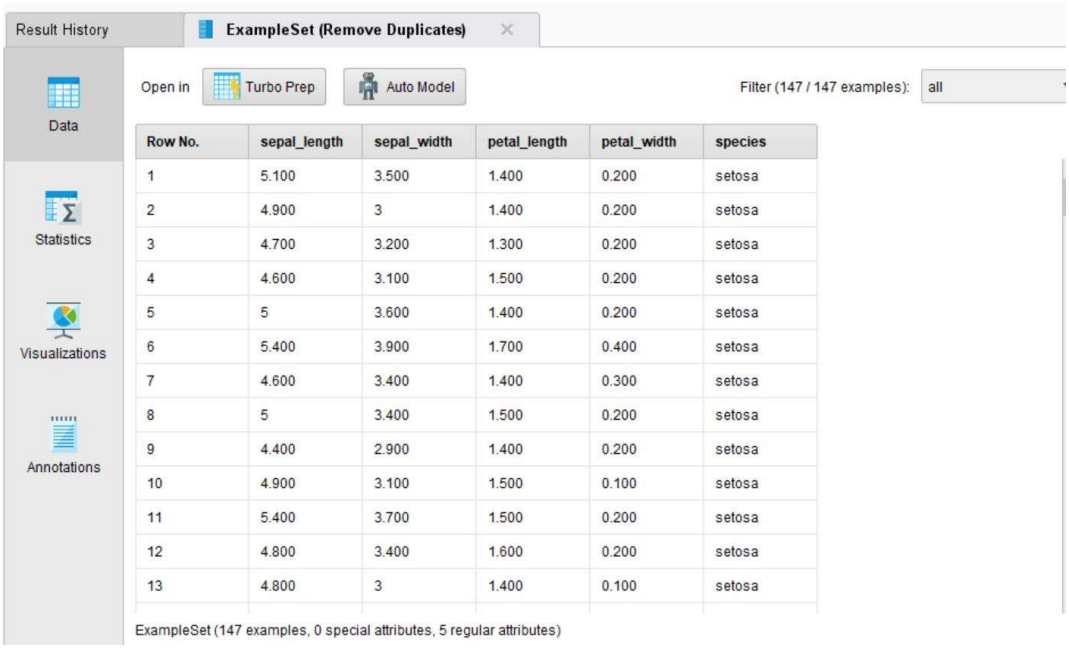


Figure 4: Design View using Remove Duplicate Operator



Row No.	sepal_length	sepal_width	petal_length	petal_width	species
1	5.100	3.500	1.400	0.200	setosa
2	4.900	3	1.400	0.200	setosa
3	4.700	3.200	1.300	0.200	setosa
4	4.600	3.100	1.500	0.200	setosa
5	5	3.600	1.400	0.200	setosa
6	5.400	3.900	1.700	0.400	setosa
7	4.600	3.400	1.400	0.300	setosa
8	5	3.400	1.500	0.200	setosa
9	4.400	2.900	1.400	0.200	setosa
10	4.900	3.100	1.500	0.100	setosa
11	5.400	3.700	1.500	0.200	setosa
12	4.800	3.400	1.600	0.200	setosa
13	4.800	3	1.400	0.100	setosa

Figure 5: Iris dataset Removing Duplicates

Originally there is a total of 150 examples in the Iris dataset after the implementation of the remove duplicate operator we have left with 147 examples rest 3 examples were duplicates which are removed with the remove duplicate operator.

If the selected attributes have the same values, two instances are called duplicate. The attribute filter type parameter and other related parameters can be used to select attributes. Assume that two attributes, 'att1' and 'att2,' are chosen, with three and two possible values, respectively.

8.2 Rename an Attribute

The following are the various operators used for the renaming task.

Rename

The Rename operator is used to rename one or more of the input ExampleSet's attributes. It's important to remember that attribute names must be special. The Rename operator does not affect an attribute's form or position.

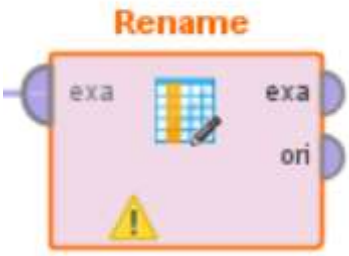


Figure 6: Rename Operator

For example, suppose you have an integer form and standard role attribute called 'alpha.' The attribute's name would be changed if it is renamed to 'beta.' It will keep its integer form and daily function. Use the Set Function operator to adjust an operator's role. At 'Data Transformation/Class Conversion,' a variety of type conversion operators are available for modifying the type of an attribute.

The Rename operator takes two parameters; one is the existing attribute name and the other is the new name you want the attribute to be renamed to.

The Rename operator has the following ports:

Input

An Example Set is expected at this input port. In the attached Example Method, it's the result of the Retrieve operator. Other operators' output can also be used as data. Since attributes are defined in its metadata, metadata must be attached to data for input. The Retrieve operator returns metadata in addition to data.

Output

This port produces an Example Set with renamed attributes.

Original

This port passes the Example Set that was provided as input to the output without any changes. This is commonly used to reuse the same Example Set through several operators or to display the Example Set in the Results Workspace.

Parameters

old name

This parameter is used to specify which attribute's name should be modified.

new name

This parameter is used to specify the attribute's new name. Special characters may also be used in a name.

rename additional attributes

Click the Edit List button to rename several attributes. You can choose attributes and give them new names here.

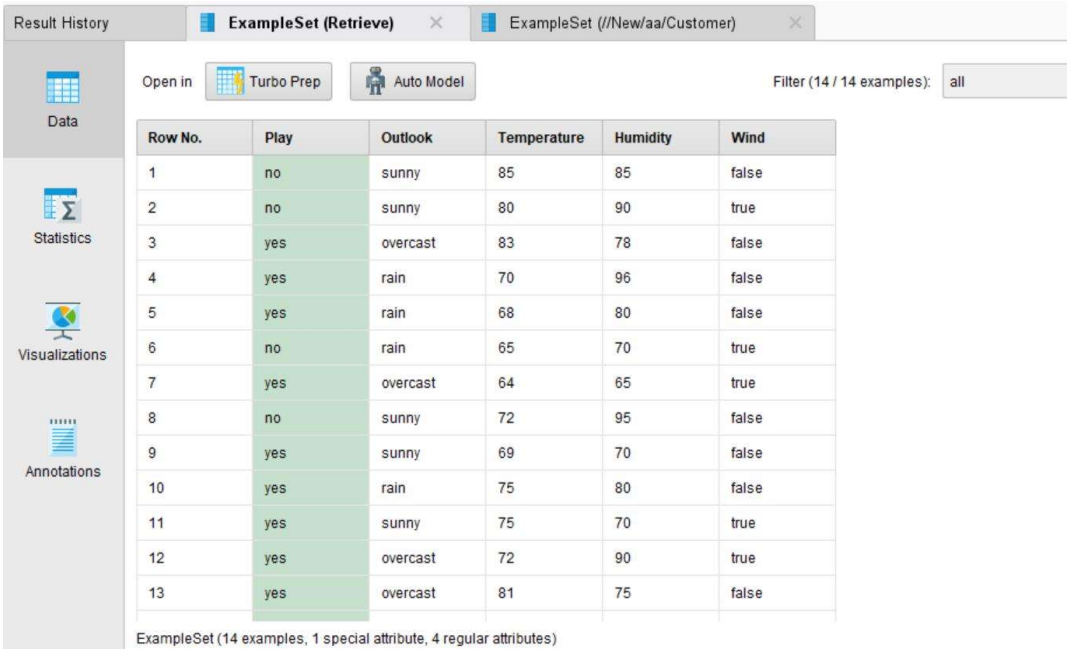
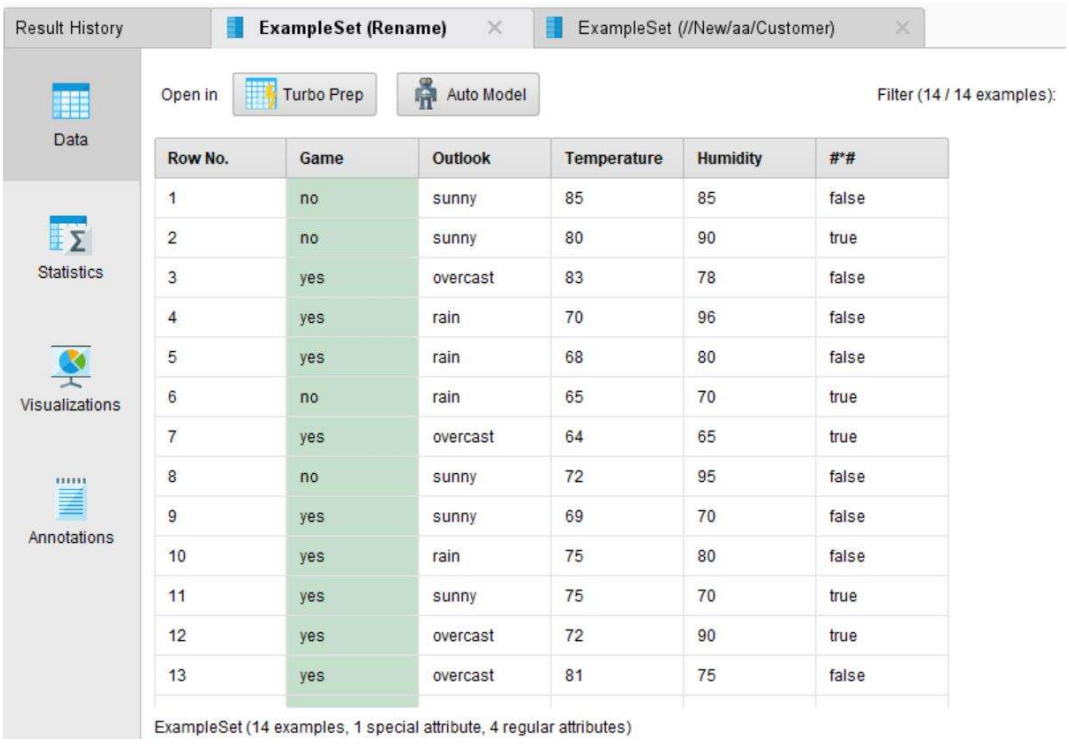


Figure 7: Exampleset before Rename Operator



Row No.	Game	Outlook	Temperature	Humidity	#*#
1	no	sunny	85	85	false
2	no	sunny	80	90	true
3	yes	overcast	83	78	false
4	yes	rain	70	96	false
5	yes	rain	68	80	false
6	no	rain	65	70	true
7	yes	overcast	64	65	true
8	no	sunny	72	95	false
9	yes	sunny	69	70	false
10	yes	rain	75	80	false
11	yes	sunny	75	70	true
12	yes	overcast	72	90	true
13	yes	overcast	81	75	false

Figure 8:Resulset After Implementing Rename Operator

This Example Process makes use of the 'Golf' data collection. The 'Wind' attribute has been renamed to '#*#', and the 'Play' attribute has been renamed to 'Game'. To demonstrate that special characters can be used to rename attributes, the 'Wind' attribute is renamed to '#*#'. Attribute names, on the other hand, should always be meaningful and specific to the type of data stored in them.

Rename By replacing

This operator can be used to rename a collection of attributes by substituting a substitute for parts of the attribute names.

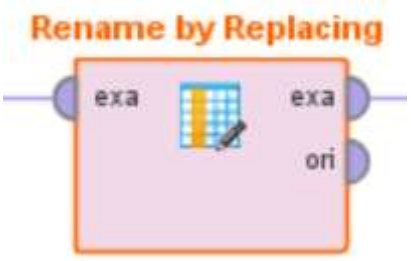
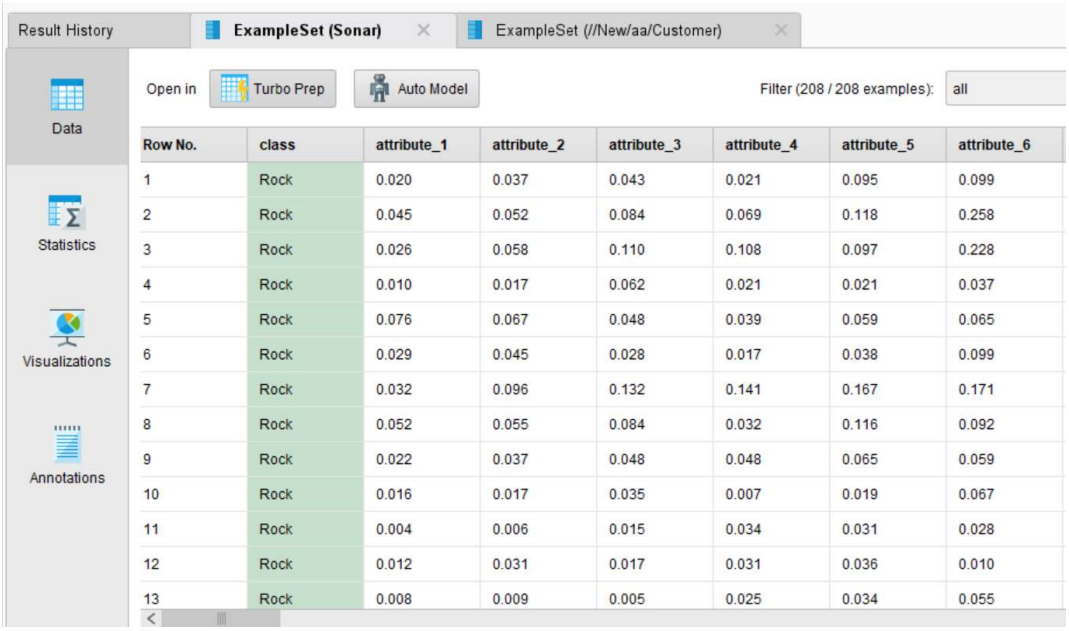


Figure 9: Rename by Replacing Operator

The Rename by Replacing operator substitutes the required replacement for parts of the attribute names. This operator is often used to exclude unnecessary characters from attribute names, such as whitespaces, parentheses, and other characters. The substitute parameter specifies which part of the attribute name should be changed. It's a normal term, and it's a very strong one at that.An arbitrary string may be used as the replace by parameter. Empty strings are also permitted. Using $1, $2, $3, and so on to capture groups of the regular expression of the replace what parameter can be accessed. Please take a look at the attached Example Process for more details.



Row No.	class	attribute_1	attribute_2	attribute_3	attribute_4	attribute_5	attribute_6
1	Rock	0.020	0.037	0.043	0.021	0.095	0.099
2	Rock	0.045	0.052	0.084	0.069	0.118	0.258
3	Rock	0.026	0.058	0.110	0.108	0.097	0.228
4	Rock	0.010	0.017	0.062	0.021	0.021	0.037
5	Rock	0.076	0.067	0.048	0.039	0.059	0.065
6	Rock	0.029	0.045	0.028	0.017	0.038	0.099
7	Rock	0.032	0.096	0.132	0.141	0.167	0.171
8	Rock	0.052	0.055	0.084	0.032	0.116	0.092
9	Rock	0.022	0.037	0.048	0.048	0.065	0.059
10	Rock	0.016	0.017	0.035	0.007	0.019	0.067
11	Rock	0.004	0.006	0.015	0.034	0.031	0.028
12	Rock	0.012	0.031	0.017	0.031	0.036	0.010
13	Rock	0.008	0.009	0.005	0.025	0.034	0.055

Figure 10: Result set of Rename by Replacing

The Retrieve operator is used to load the 'Sonar' data set. A breakpoint has been placed here to allow you to see the Example Set. The Example Set has 60 standard attributes with names such as attribute 1, attribute 2, and so on. On it, the Rename by Replacing operator is used. Since the attribute filter form parameter is set to 'all,' this operator can rename any attribute. The first capturing category and a slash, i.e. 'att-', are used in place of 'attribute_' in the names of the 'Sonar' attributes. As a result, attributes are renamed to att-1, att-2, and so on.

Create your dataset and rename the attributes of your dataset by considering different rename operators.

Rename by Generic Names

The Rename by Generic Names operator renames the attributes of a given ExampleSet to a collection of generic names such as att1, att2, att3, and so on. The generic name stem parameter determines the name stem to be used when constructing generic names. Using the stem 'att' as an example, attribute names will be 'att1', 'att2', and so on.

Row No.	class	att1	att2	att3	att4	att5	att6
1	Rock	0.020	0.037	0.043	0.021	0.095	0.099
2	Rock	0.045	0.052	0.084	0.069	0.118	0.258
3	Rock	0.026	0.058	0.110	0.108	0.097	0.228
4	Rock	0.010	0.017	0.062	0.021	0.021	0.037
5	Rock	0.076	0.067	0.048	0.039	0.059	0.065
6	Rock	0.029	0.045	0.028	0.017	0.038	0.099
7	Rock	0.032	0.096	0.132	0.141	0.167	0.171
8	Rock	0.052	0.055	0.084	0.032	0.116	0.092
9	Rock	0.022	0.037	0.048	0.048	0.065	0.059
10	Rock	0.016	0.017	0.035	0.007	0.019	0.067
11	Rock	0.004	0.006	0.015	0.034	0.031	0.028
12	Rock	0.012	0.031	0.017	0.031	0.036	0.010
13	Rock	0.008	0.009	0.005	0.025	0.034	0.055

Figure 11: Result set of Rename by Generic names

Rename By Example Values

The Rename by Example Values operator creates new attribute names based on the values of the listed Example Set example. Please note that all attributes, both standard and unique, have been renamed. The example is also removed from the Example Set. When an example contains the names of the attributes, this operator may be useful.

Row No.	new_label	new_name1	new_name2
1	negative	value2	value2
2	negative	value1	value4
3	negative	value4	value3
4	negative	value2	value4
5	positive	value3	value0
6	negative	value1	value2
7	negative	value1	value1
8	negative	value1	value1
9	negative	value1	value3
10	negative	value3	value4
11	positive	value0	value0
12	negative	value3	value4
13	positive	value0	value4

Figure 12: Rename By Example Values

The Sub process operator is the first step in this Example Process. The Example Set is returned by the Sub process operator. The names of the attributes are actually 'label,"att1', and'att2', as you can see. The values 'new label," new name1', and new name2' is used in the first example. On this Example Set, the Rename by Example Values operator is used to rename the first example's values to attribute names. The row number parameter is set to 1 in this example. You'll notice that the attributes have been renamed appropriately after the process has been completed. Furthermore, the first example from the Example Set has been deleted.

Describe different renaming operators.

8.3 Identification and Removal of Missing values in Data set

Missing values can be replaced with the Attribute's minimum, maximum, or average value. Zero may also be used to fill in gaps in data. As a substitute for missing values, any replenishment value can be listed.

Figure 13: Dataset with Missing values

The following operators are used to handle missing values:

Replace Missing Values

This Operator replaces missing values in Examples of selected Attributes by a specified replacement.

Figure 14: Replace Missing Value Operator

Missing values can be replaced with the Attribute's minimum, maximum, or average value. Zero may also be used to fill in gaps in data. As a substitute for missing values, any replenishment value can be listed.

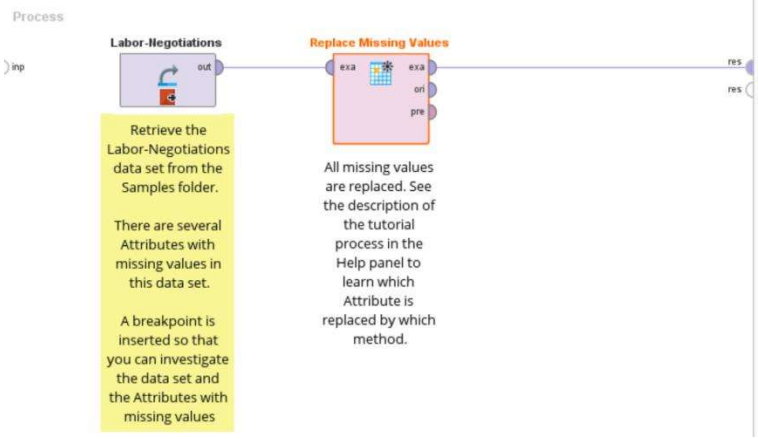
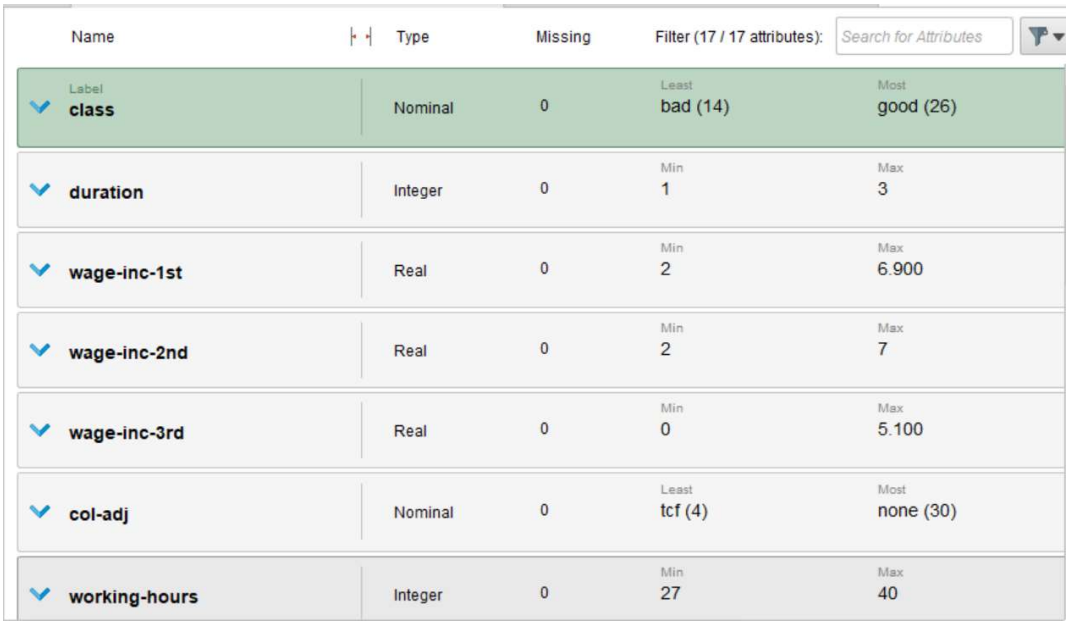


Figure 15: Design view of Replace Missing Values

This Process shows the usage of the Replace Missing Values Operator on the Labor-Negotiations data set from the Samples folder. The Operator is configured that it applies the replacement on all Attributes which have at least one missing value (*attribute filter type* is *no_missing_values* and *invert selection* is true). In the columns parameter several Attributes are set to different replacement methods:

- wage-inc-1st: minimum
- wage-inc-2nd: maximum
- wage-inc-3rd: zero
- working-hours: value

The parameter replenishment value is set to 35 so that all missing values of the Attribute working-hours are replaced by 35. The missing values of the remaining Attributes are replaced by the average of the Attribute (parameter *default*).



Name	Type	Missing	Filter (17 / 17 attributes): Search for Attributes	
Label **class**	Nominal	0	Least bad (14)	Most good (26)
duration	Integer	0	Min 1	Max 3
wage-inc-1st	Real	0	Min 2	Max 6.900
wage-inc-2nd	Real	0	Min 2	Max 7
wage-inc-3rd	Real	0	Min 0	Max 5.100
col-adj	Nominal	0	Least tcf (4)	Most none (30)
working-hours	Integer	0	Min 27	Max 40

Figure 16: Result after Implementing Replace Missing Values

Declare Missing Value

Declare Missing Value replaces the listed values of the selected attributes with Double.NaN, resulting in these values being marked as missing. The subsequent operators will treat these values as missing values. Nominal, numeric, and regular expression modes may be used to choose the desired values. The mode parameter can be used to manage this action.

Input

An Example Set is expected at this input port. In the attached Example Method, it's the result of the Retrieve operator. Other operators' output can also be used as data.

Output

The missing values are substituted for the required values of the selected attributes, and the resultant Example Set is provided via this port.

Original

This port passes the Example Set that was provided as input to the output without any changes. This is commonly used to reuse the same Example Set through several operators or to display the Example Set in the Results Workspace.

Parameters

attribute filter type

This parameter lets you choose the attribute selection filter, which is the tool you'll use to pick the appropriate attributes. It comes with the following features:

- **All:** This choice simply selects all of the Example Set's attributes. This is the standard-setting.
- **single**: Selecting a single attribute is possible with this choice. When you select this choice, a new parameter (attribute) appears in the Parameters panel.
- **subset:** This choice allows you to choose from a list of multiple attributes. The list contains all of the Example Set's attributes; appropriate attributes can be easily selected. If the metadata is unknown, this option will not work. When you select this choice, a new parameter appears in the Parameters panel.
- **numeric value filter**: When the numeric value filter choice is selected, a new parameter (numeric condition) appears in the Parameters panel. The examples of all numeric attributes that satisfy the numeric condition are chosen. Please note that regardless of the numerical situation, all nominal attributes are chosen.
- **no_missing_values:** This choice simply selects all of the Example Set's attributes that do not have a missing value in any of the examples. All attributes with a single missing value are deleted.
- **value_type:** This option allows you to pick all of a type's attributes. It's worth noting that styles are arranged in a hierarchy. The numeric form, for example, includes both real and integer types. When selecting attributes via this option, users should have a basic understanding of type hierarchy. Other parameters in the Parameters panel become visible when it is picked.
- **block_type:** The block type option works similarly to the value type option. This choice allows you to choose all of the attributes for a specific block form. Other parameters (block type, use block type exception) become available in the Parameters panel when this choice is selected.

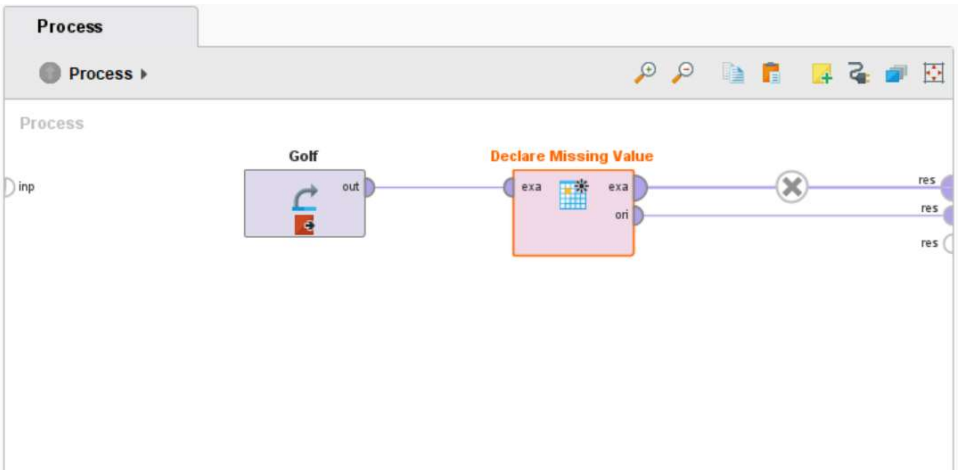


Figure 17: Implementation of Declare Missing value

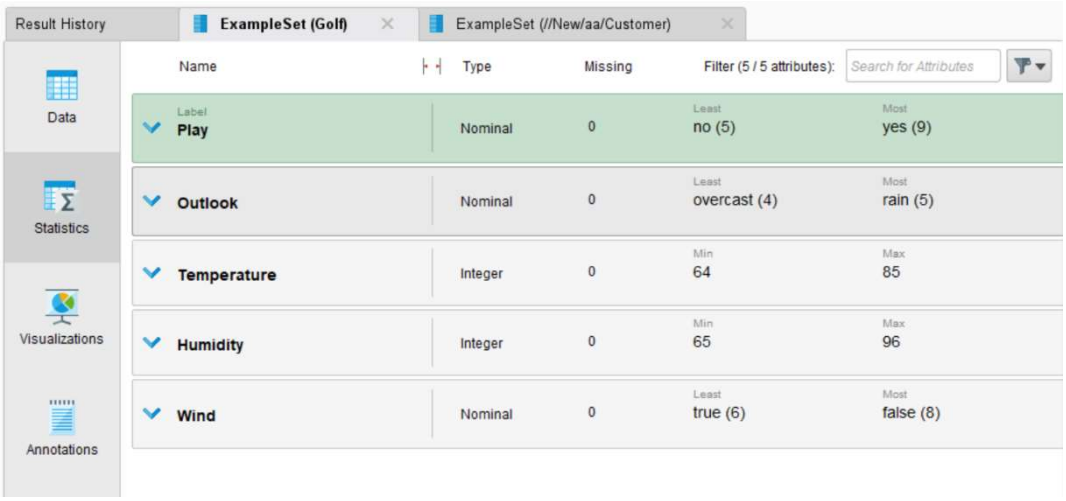


Figure 18: Result view of Declare Missing value

Replace all Missing

A universal missing value handler that replaces missing values with new ones when dealing with nominal values. MISSING uses the average of the non-missings to substitute missings and infinite in numerical columns. If a column is absent entirely, all values will be set to zero. If all dates are missing, the first known date is used, or a zero date (1 January 1970) is used again.

Figure 19: Replace All Missings Operator

This operator handles all missing values in a data set automatically.

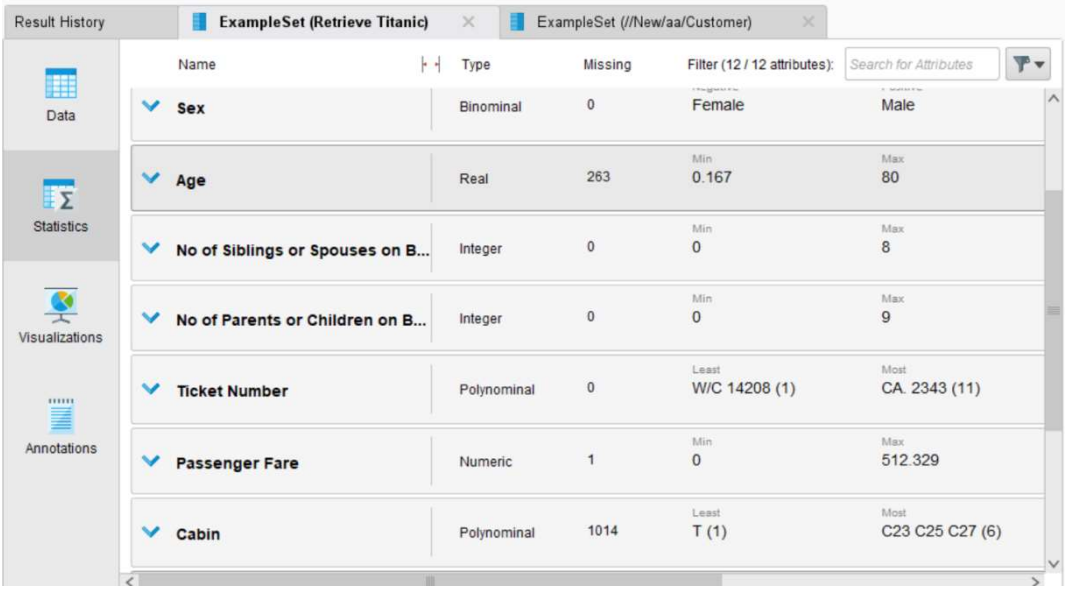


Figure 20: Before All Replace Missing

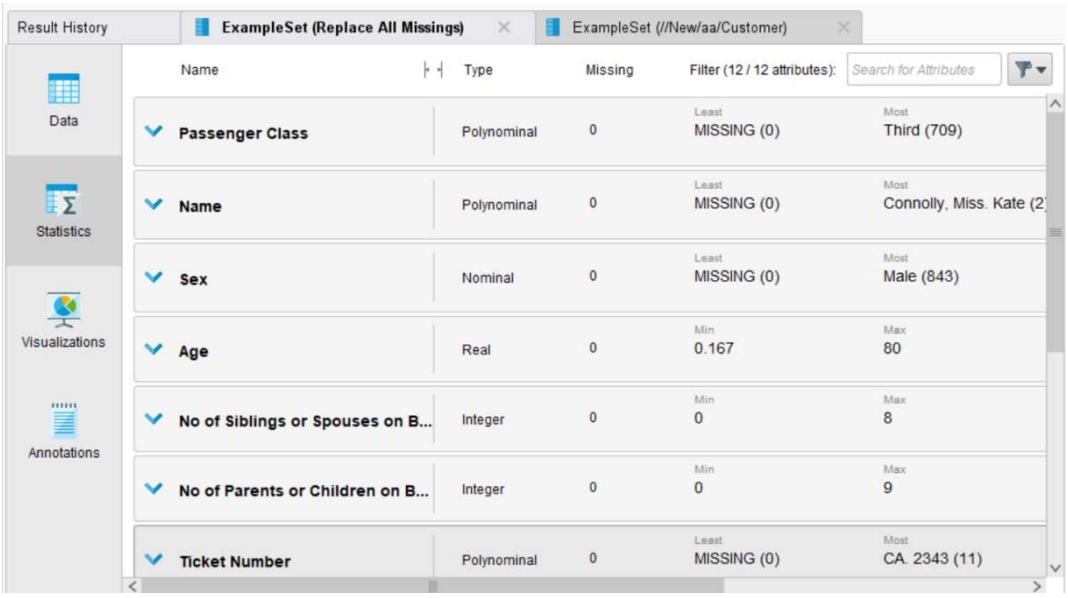


Figure 21: Result view of Relace All Missing

In the Titanic data collection, this method removes all missing values. For the remaining rows, all numerical values are replaced by average values, such as using the age 29.881 for missing ages. The term MISSING has been used to replace all nominal missings. The Titanic data set lacks dates, but they would have been replaced by the set's first date.

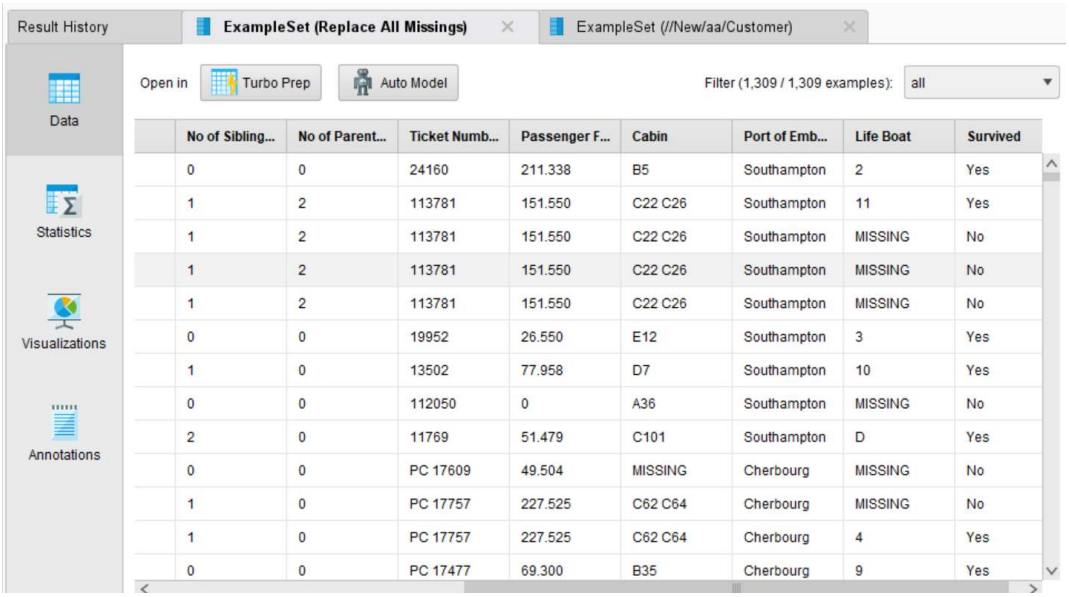


Figure 22: Replacement of Nominal and Numerical Values

8.4 Apriori method for finding frequent item set Weka

The Apriori algorithm is a machine learning algorithm that finds likely associations and generates association rules. WEKA is a tool that implements the Apriori algorithm. While computing these guidelines, you should specify a minimum level of support and an appropriate level of confidence. The Apriori algorithm will be applied to the supermarket data given in the WEKA installation.

Loading Data

Open the Preprocess tab in the WEKA explorer, pick the supermarket.arff database from the installation folder, and press the Open file... icon. After the data has been loaded, you can see the screen below.

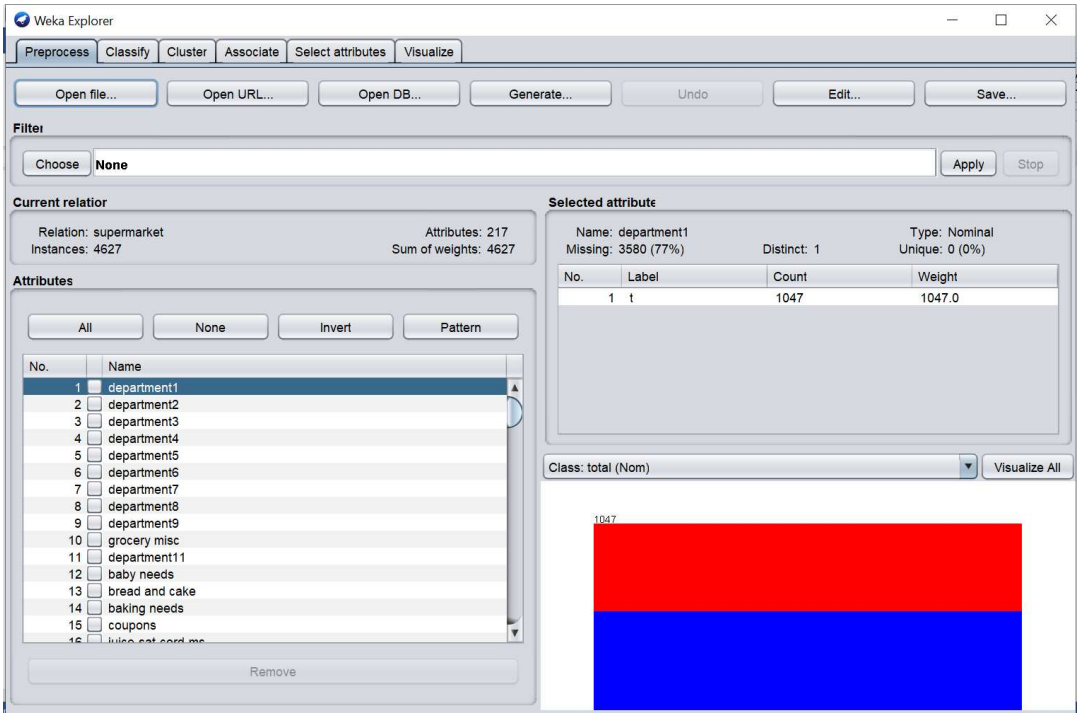


Figure 23: Loading of data

A total of 4627 instances and 217 attributes are stored in the database. It's easy to see how complicated it will be to find a connection between so many variables. The Apriori algorithm, fortunately, automates this task.

Associate

Then, on the Associate TAB, select the Choose option. As shown in the screenshot, choose the Apriori association.

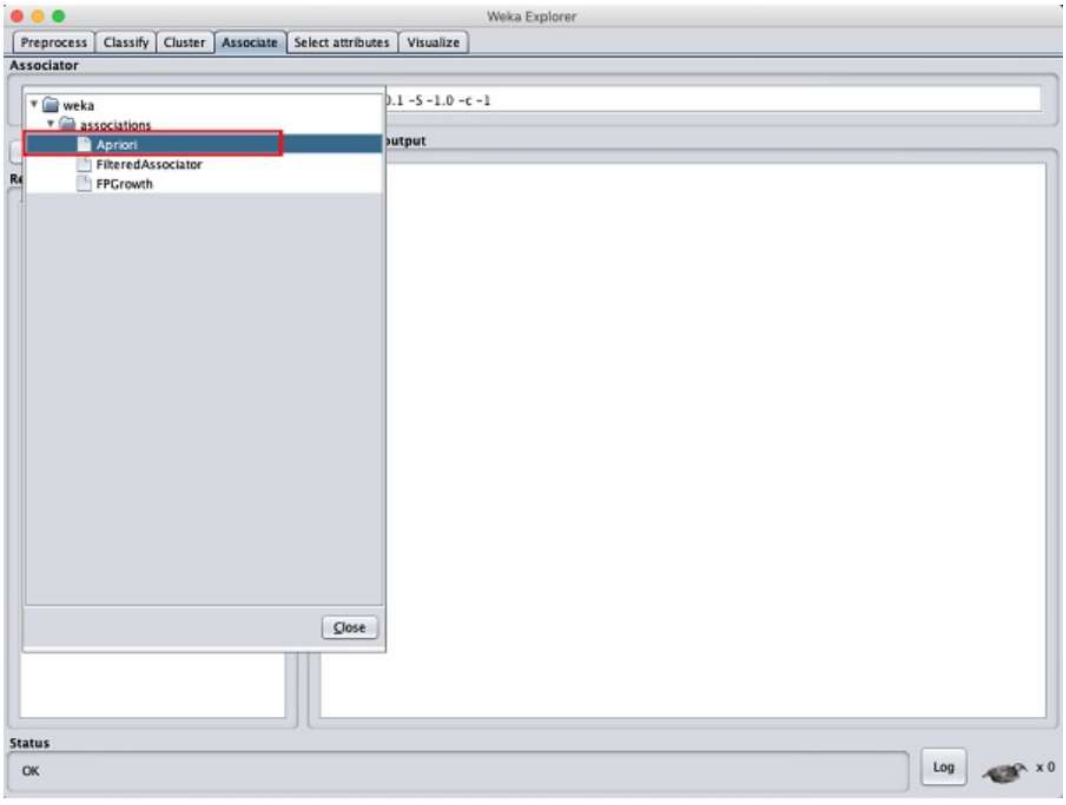


Figure 24: Selection of Apriori

To set the parameters for the Apriori algorithm, click on its name, and a window will appear, as shown below, allowing you to do so.

On the weather info, run Apriori. What is the support for this item set based on the output? outlook = rainy humidity = normal windy = FALSE play = yes.

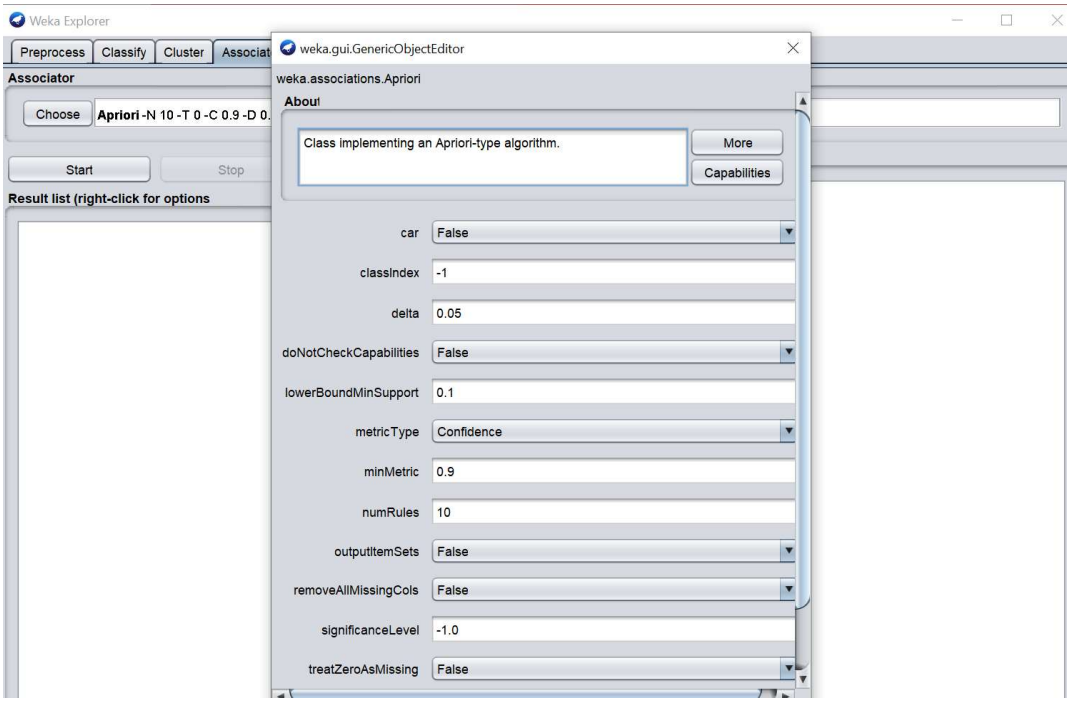


Figure 25: Parameter Setting

Click the Start button after you've set the parameters. After a while, the results will appear as shown in the screenshot below.

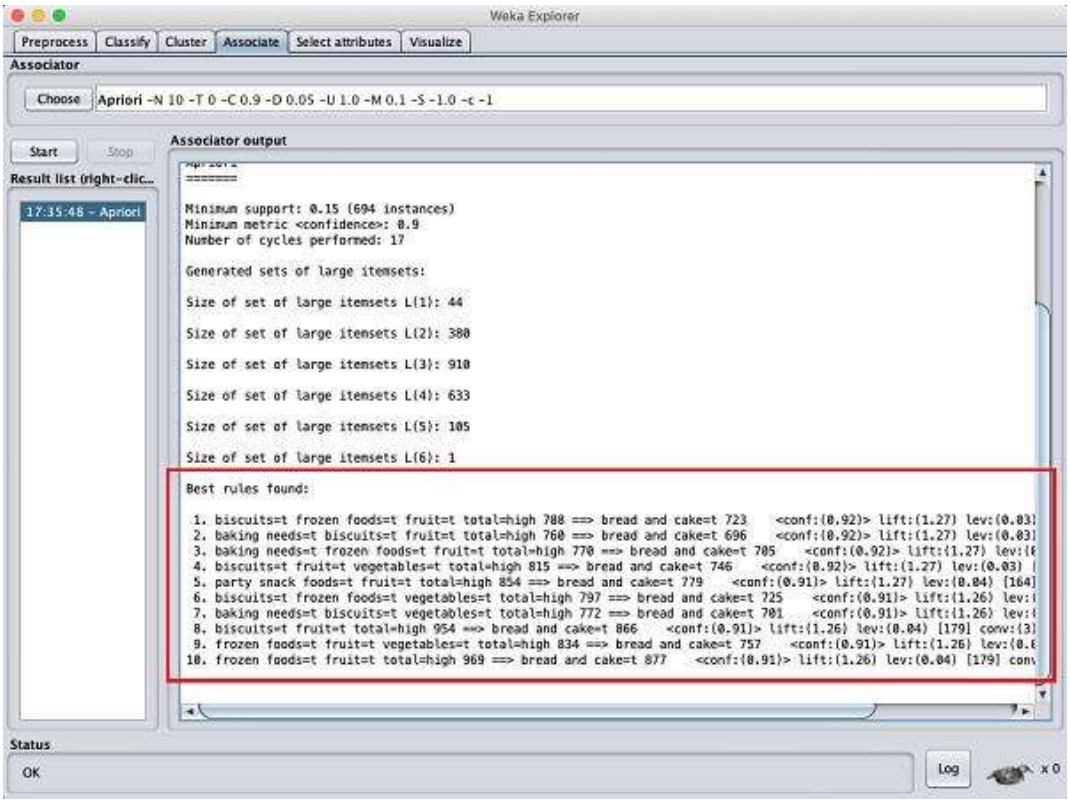


Figure 26: Apriori Results

The detected best rules of associations can be found at the bottom of the page. This will assist the supermarket in stocking the necessary shelves with their items.

 Run Apriori on this data with default settings. Comment on the rules that are generated. Several of them are quite similar. How are their support and confidence values related?

Summary

- The Remove Duplicates operator compares all examples in an Example Set against each other using the listed attributes to remove duplicates.
- The special attributes are attributes with special roles which identify the examples. Special attributes are id, label, prediction, cluster, weight, and batch.
- The Rename operator is used for renaming one or more attributes of the input Example Set.
- To change the role of an operator, use the Set Role operator.
- Whitespaces, brackets, and other unnecessary characters are commonly removed from attribute names using rename by replacing.
- Missing values can be replaced with the attribute's minimum, maximum, or average value.
- Using invert selection all of the previously selected attributes have been unselected, and previously unselected attributes have been selected.

Keywords

Output Ports: The duplicate examples are removed from the given Example Set and the resultant Example Set is delivered through this port.

old name: This parameter is used to specify which attribute's name should be modified.

Rename by replacing: The Rename by Replacing operator substitutes the required replacement for parts of the attribute names.

replace what: The replace what parameter specifies which part of the attribute name should be changed.

no missing values : This choice simply selects all of the Example Set's attributes that do not have a missing value in any of the examples.

Create view : Instead of modifying the underlying data, you can build a View. To allow this choice, simply select this parameter.

Invert selection : When set to true, this parameter acts as a NOT gate, reversing the selection.

Self Assessment Questions

1. The…….operator removes duplicate examples from an Example Set by comparing all examples with each other based on the specified attributes.

A. Remove Duplicates

B. Duplicate Removal

C. Delete Duplicate

D. Erase Duplicate

2. _____ is the default option used to remove duplicate values.

A. Subset

B. Single

C. All

D. Block_type

3. _____ allows selection of multiple attributes through a list for the removal of duplicate values.

A. Subset

B. Single

C. All

D. Block_type

4. The _____ operator has no impact on the type or role of an attribute.

A. Retrieve

B. Store

C. Rename

D. Filter

5. To change the role of an operator, use the _____operator.

A. Get Role

B. Set Role

C. Change Role

D. Update Role

6. The _____operator replaces missing values in Examples of selected Attributes by a specified replacement.

A. Replace Missing Values

B. Impute Missing Values

C. Remove Missing values

D. Hide Missing Values

7. The _____operator estimates values for the missing values by applying a model learned for missing values.

A. Replace Missing Values

B. Impute Missing Values

C. Remove Missing values

D. Hide Missing Values

8. Which of the following attribute_filter_type allows the selection of multiple attributes through a list

A. value_type

B. subset

C. All

D. block_type

9. Which of the following attribute_filter_type option selects all Attributes of the ExampleSet which do not contain a missing value in any Example.

A. numeric_value_filter

B. regular_expression

C. no_missing_values

D. None

10. Missing data can_____ the effectiveness of classification models in terms of accuracy and bias.

A. Reduce

B. Increase

C. Maintain

D. Eliminate

11. _____is a rule-based machine learning method for discovering interesting relations between variables in large databases.

A. Association rule learning

B. Market Based Analysis

C. Rule Learning

D. None

12. Let X and Y are the data items where _____specifies the probability that a transaction contains $X \cup Y$.

A. Support

B. Confidence

C. Both

D. None

13. Let X and Y are the data items where _____ specifies the conditional probability that a transaction having X also contains *Y*.

A. Support

B. Confidence

C. Both

D. None

14. The……algorithm is one such algorithm in ML that finds out the probable associations and creates association rules.

A. Decision Tree

B. KNN

C. Apriori

D. All of the above

15. You can select Apriori in Weka by clicking which of the following tab:

A. Classify

B. Cluster

C. Associate

D. Select Attribute

Answer for Self Assessment

1.	A	2.	C	3.	A	4.	C	5.	B
6.	A	7.	B	8.	B	9.	C	10.	A
11.	A	12.	A	13.	B	14.	C	15.	C

Review Questions

Q1) Why is it necessary to remove duplicate values from the dataset. Explain the step-by-step process of removing duplicate values from the dataset.

Q2) With example elucidate the different renaming operators available in rapidminer.

Q3) Explain the step-by-step process of generating association rules in weka using the apriori algorithm.

Q4) "Missing values lead to incorrect analysis of the data" Justify the statement with an appropriate example.

Q5) Elucidate the different methods of handling missing data in rapid miner.

Further Readings

Hofmann, M., & Klinkenberg, R. (Eds.). (2016). RapidMiner: Data mining use cases and business analytics applications. CRC Press.

Kotu, V., & Deshpande, B. (2014). Predictive analytics and data mining: concepts and practice with rapidminer. Morgan Kaufmann.

Chisholm, A. (2013). Exploring data with Rapidminer. Packt Publishing Ltd.

Nisbet, R., Elder, J., & Miner, G. (2009). Handbook of statistical analysis and data mining applications. Academic Press.

Smith, T. C., & Frank, E. (2016). Introducing machine learning concepts with WEKA. In Statistical genomics (pp. 353-378). Humana Press, New York, NY.

https://rapidminer.com/wp-content/uploads/2014/10/RapidMiner-5-Operator-Reference.pdf

https://www.myexperiment.org/workflows/1345.html

https://docs.rapidminer.com/latest/studio/operators/blending/attributes/names_and_roles/rename_by_replacing.html

https://docs.rapidminer.com/latest/studio/operators/blending/attributes/names_and_roles/rename_by_example_values.html

https://eeisti.fr/grug/ATrier/GSI/MachineLearningOptimisation/Algorithmes_ML/Apriori/weka_a_priori.pdf

Unit 09: Association and Correlation Analysis

CONTENTS

Objectives

After this lecture, you will be able to

- Understand the concept of frequent patterns and association rules.
- Learn how to calculate support and confidence.
- Understand the basic concepts of the Apriori algorithm.
- Learn the working of the Apriori algorithm.
- Understand the process of finding frequent patterns using FP-tree.
- Know the applications of association rule mining.

Introduction

Imagine that you are a sales manager, and you are talking to a customer who recently bought a PC and a digital camera from the store. What should you recommend to her next?Frequent patterns and association rules are the knowledge that you want to mine in such a scenario.The term "association mining" refers to the process of looking for frequent items in a data set. Typically, interesting connections and similarities between item sets in transactional and relational databases are discovered through regular mining. Frequent Mining, in a nutshell, shows which objects appear together in a transaction or relationship.

9.1 Basic Concepts

Frequent patterns are patterns (e.g., itemsets, or subsequences) that appear frequently in a data set.Frequent Pattern Mining is a Data Mining topic that aims to extract frequently occurring itemsets from a database. Frequent itemsets are linked to interesting trends in data, such as Association Rules, and play an important role in many Data Mining tasks.

For example, a set of items, such as milk and bread, that appear frequently together in a transaction data set is a frequent item-set.

A subsequence, such as buying first a PC, then a digital camera, and then a memory card, if it occurs frequently in a shopping history database, is a (frequent) sequential pattern.

< {Homepage} {Electronics} {Digital Cameras} {Canon Digital Camera} {Shopping Cart} {Order Confirmation} {Return to Shopping} >

Subgraphs, subtrees, and sublattices are examples of structural structures that can be combined with itemsets or subsequences to form a substructure. A (frequent) structural pattern is a substructure that appears regularly in a graph database. In mining associations, correlations, and many other interesting relationships among data, finding frequent patterns is critical.

Market Basket Analysis: A Motivating Example

Market basket analysis is a common example of frequent itemset mining. This method examines consumer purchasing patterns by identifying links between the various products that customers put in their "shopping baskets."The discovery of these associations can help retailers develop marketing strategies by gaining insight into which items are frequently purchased together by customers.Customers buying milk, for example, are they more likely to buy bread (and what kind of bread) on the same trip to the supermarket? This data will help retailers manage their shelf space and do selective promotions, which can lead to increased sales.

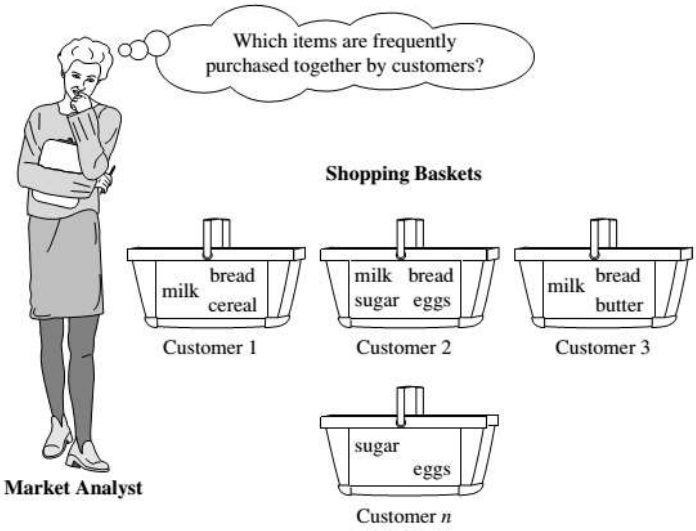


Figure 1: Market Basket Analysis

9.2 How does Association Rule Learning work?

Association rules are "if-then" statements that display the likelihood of relationships between data items in large data sets in a variety of databases.

Rule form

Antecedent → Consequent [support, confidence]

(support and confidence are user-defined measures of interestingness)

If A then B, for example, association rule learning is based on the principle of If and Else Statements.

Figure 2: Association Rule

The If aspect is referred to as the antecedent, and the Then statement is referred to as the Consequent. Single cardinality refers to relationships in which we may discover an interaction or relationship between two objects. It's all about making laws, and as the number of things grows, so does cardinality.

age(x, "30..39") ^ income(x, "42..48K") →buys(x, "car") [1%,75%]

Frequent Patterns and Association Rules

If we think of the universe as the set of items available at the store, then each item has a Boolean variable representing the presence or absence of that item. Each basket can then be represented by a Boolean vector of values assigned to these variables. The Boolean vectors can be analyzed for buying patterns that reflect items that are frequently associated or purchased together. These patterns can be represented in the form of ASSOCIATION RULES.

We used association rules to find mistakes often occurring together while solving exercises.The purpose of looking for these associations is for the teacher to ponder and, may be, toreview the course material or emphasize subtleties while explaining concepts to students. Thus, it makes sense to have a support that is not too low.

For example, the information that customers who purchase computers also tend to buy antivirus software at the same time is represented in the following association rule:

Computer=>antivirus_Software [support=2%, confidence=60%].

Support of 2% for Rule means that 2% of all the transactions under analysis show that computer and antivirus software are purchased together.A confidence of 60% means that 60% of the customers who purchased a computer also bought the software. There are several metrics for measuring the relationships between thousands of data objects. These figures are as follows:

Support

Confidence

Support

The frequency of A, or how often an object appears in the dataset, is called "support." It's the percentage of the transaction T that has the itemset X in it. If there are X datasets, the following can be written for transaction T:

Support(X)=Freq(X)/T

Confidence

The term "confidence" refers to how much the rule has been proven correct. Or, when the frequency of X is already known, how often the items X and Y appear together in the dataset. It's the ratio of the number of records that contain X to the number of transactions that contain X.

Confidence=Freq(X,Y)/Freq(X)

Table 1: DataItems along with Transaction ids

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

From the above table, {Milk, Diaper}=>{Beer}

s= σ({Milk, Diaper, Beer}) $\div$ |T|

= 2/5

= 0.4

c= σ (Milk, Diaper, Beer) $\div$ (Milk, Diaper)

= 2/3

= 0.67

Rules that satisfy both a minimum support threshold (min_sup) and a minimum confidence threshold (min_conf) are called strong. A set of items is referred to as an itemset. An itemset that contains k items is a k-itemset. The set **{computer, antivirus software}** is a 2-itemset. The occurrence frequency of an itemset is the number of transactions that contain the itemset. This is also known, simply, as the **frequency, support count, or count** of the itemset.

In general, association rule mining can be viewed as a two-step process:

- ***Find all frequent itemsets:*** By definition, each of these itemsets will occur at least as frequently as a predetermined minimum support count, min sup.
- ***Generate strong association rules from the frequent itemsets:*** By definition, these rules must satisfy minimum support and minimum confidence.

It could be useful for the OurVideoStore manager to know what movies are often rented together or if there is a relationship between renting a certain type of movie and buying popcorn or pop. The discovered association rules are of the form: P→Q [s, c], where P and Q are conjunctions of attribute value-pairs, and s (for support) is the probability that P and Q appear together in a transaction and c (for confidence) is the conditional probability that Q appears in a transaction when P is present. For example, the hypothetic association rule

Rent Type (X, "game") ^ Age(X,"13-19") → Buys(X, "pop") [s=2% ,c=55%]

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

Why Frequent Itemset Mining?

Because of its many applications in mining association rules, correlations, and graph pattern constraints that are focused on frequent patterns, sequential patterns, and many other data mining tasks, frequent itemset or pattern mining is widely used.

9.3 The Apriori Algorithm: Basics

The following are the key concepts used in context to apriori algorithm :

- **Frequent Itemsets**: The sets of an item that has minimum support
- **Apriori Property**: Any subset of frequent item-set must be frequent.
- **Join Operation**: To find Lk, a set of candidate k-item-sets is generated by joining Lk-1 with itself.

Apriori Algorithm

The Apriori algorithm was the first algorithm for frequent itemset mining to be proposed. R Agarwal and R Srikant improved it later, and it became known as Apriori. To reduce the search space, this algorithm uses two steps: "join" and "prune." It is an iterative method for identifying the most frequent itemsets.

Apriori says:

The probability that item I is not frequent is if:

- P(I) < minimum support threshold, then I is not frequent.
- P (I+A) < minimum support threshold, then I+A is not frequent, where A also belongs to itemset.
- If an itemset set has a value less than minimum support then all of its supersets will also fall below min support, and thus can be ignored. This property is called the Antimonotone property.

The Apriori Algorithm for data mining includes the following steps:

1. **Join Step:** By joining each item with itself, this move generates (K+1) itemset from K-itemsets.
2. **Prune Step:** This step counts all of the items in the database. If a candidate item fails to fulfill the minimum support requirements, it is classified as infrequent and hence withdrawn. This step aims to reduce the size of the candidate itemsets.

Steps In Apriori

The apriori algorithm is a series of steps that must be followed to find the most frequent itemset in a database. This data mining technique repeats the join and prune steps until the most frequently occurring itemset is found. The issue specifies a minimum assistance threshold, or the user assumes it.

1) Each object is treated as a 1-itemsets candidate in the first iteration of the algorithm. Each item's occurrences will be counted by the algorithm.

2) Set a minimum level of support, min sup. The set of 1 – itemsets whose occurrence meets the minimum sup requirement is determined. Only those candidates with a score greater than or equal to min sup are advanced to the next iteration, while the rest are pruned.

3) Next, min sup is used to find 2-itemset frequent itemsets. The 2-itemset is formed in the join phase by forming a group of 2 by combining items with itself.

4) The min-sup threshold value is used to prune the 2-itemset candidates. The table will now have two –itemsets, one with min-sup and the other with just min-sup.

5) Using the join and prune step, the next iteration will create three –itemsets. This iteration will use the antimonotone property, which means that the subsets of 3-itemsets, i.e. the two –itemset subsets of each category, will fall into min sup. The superset will be frequent if all 2-itemset subsets are frequent, otherwise, it will be pruned.

6) Making 4-itemset by joining a 3-itemset with itself and pruning if its subset does not meet the min sup criteria would be the next move. When the most frequent itemset is reached, the algorithm is terminated.

Table 2: Itemsets

Transaction	List of items
T1	I1,I2,I3
T2	I2,I3,I4
T3	I4,I5
T4	I1,I2,I4
T5	I1,I2,I3,I5
T6	I1,I2,I3,I4

Example of Apriori: Support threshold=50%, Confidence= 60%

Solution:

Support threshold=50% => 0.5*6= 3 => min_sup=3

1. Frequency of each item.

Table 3: Items With Frequency Count

Item	Count
I1	4
I2	5
I3	4
I4	4
I5	2

2. Prune Step: Table 3 shows that the I5 item does not meet min_sup=3, thus it is deleted, only I1, I2, I3, I4 meet min_sup count.
3. Step 2: Form a 2-itemset. Find the occurrences of 2-itemset in Table 2.

Table 4: Two Itemset Data

Item	Count
I1,I2	4
I1,I3	3
I1,I4	2
I2,I3	4
I2,I4	3
I3,I4	2

4. Prune Step: Table 4reveals that item sets I1, I4, and I3, I4 do not follow min sup, so they are deleted.

Table 5: Frequent Two Itemset data

Item	Count
I1,I2	4
I1,I3	3
I2,I3	4
I2,I4	3

5. Join and Prune Step: join and prune Form a three-itemset. Find the occurrences of the 3-itemset in Table 2. Find the 2-itemset subsets that endorse min sup from Table 5.

We can see that for itemset{ I1, I2, I3} subsets, TABLE-5 contains {I1, I2}, {I3, I2},{I1, I3} indicating that {I1, I2, I3 } is frequent.

Table 6: Three Itemset Data

Item
I1,I2,I3
I1,I2,I4
I1,I3,I4

Only {I1, I2, I3} is frequent.

6. **Generate Association Rules:** From the frequent itemset discovered above the association could be:

{I1, I2} => {I3}

Confidence = support {I1, I2, I3} / support {I1, I2} = (3/ 4)* 100 = 75%

{I1, I3} => {I2}

Confidence = support {I1, I2, I3} / support {I1, I3} = (3/ 3)* 100 = 100%

{I2, I3} => {I1}

Confidence = support {I1, I2, I3} / support {I2, I3} = (3/ 4)* 100 = 75%

{I1} => {I2, I3}

Confidence = support {I1, I2, I3} / support {I1} = (3/ 4)* 100 = 75%

{I2} => {I1, I3}

Confidence = support {I1, I2, I3} / support {I2 = (3/ 5)* 100 = 60%

{I3} => {I1, I2}

Confidence = support {I1, I2, I3} / support {I3} = (3/ 4)* 100 = 75%

This shows that all the above association rules are strong if minimum confidence threshold is 60%.

Advantages

- The algorithm is simple to comprehend.
- On large itemets in large databases, the join and prune steps are simple to implement.

Disadvantages

- If the itemsets are wide and the minimum support is held low, it necessitates a lot of computation.
- The database as a whole must be scanned.

9.4 FP Growth Algorithm

The FP growth algorithm is a step forward from the apriori algorithm. Without candidate generation, the FP growth algorithm is used to find frequent itemsets in a transaction database. In frequent pattern trees or FPtrees, FP growth represents frequent items.

The relationship between the itemsets will be maintained by this tree structure. Using one frequent item, the database is fragmented. The broken portion is referred to as a "pattern fragment." These fragmented patterns' itemsets are examined. As a result, the search for frequently occurring itemsets is significantly reduced using this approach.

FP Tree

The Frequent Pattern Tree is a tree-like structure created from the database's initial itemsets. The FP tree aims to find the most frequent pattern. Each object in the itemset is represented by a node in the FP tree.Null is represented by the root node, while itemsets are represented by the lower nodes. When forming the tree, the connection of the nodes with the lower nodes, that is, the itemsets with the other itemsets is retained.

Steps In frequent pattern algorithm

We will find the frequent pattern using the frequent pattern growth method without having to generate candidates.

1) The first step is to search the database for instances of the itemsets. This move is identical to Apriori's first step. Help count or frequency of 1-itemset refers to the number of 1-itemsets in the database.
2) The FP tree is built in the second stage. To do so, start by making the tree's root. Null is used to represent the root.
3) The following move is to re-scan the database and look over the transactions. Examine the first transaction to determine the itemset contained therein. The highest-counting itemset is at the top, followed by the next-lowest-counting itemset, and so on. It means that the tree's branch is made up of transaction itemsets arranged in descending order of count.
4) The database's next transaction is analyzed. The itemsets are sorted by count in ascending order. This transaction branch will share a common prefix to the root if any itemset of this transaction is already present in another branch (for example, in the first transaction).This implies that in this transaction, the common itemset is connected to the new node of another itemset.
5) In addition, as transactions occur, the count of the itemset is increased. As nodes are generated and connected according to transactions, the count of both the common node and new node increases by one.
6) The next move is to mine the FP Tree that has been developed. The lowest node, as well as the relations between the lowest nodes, are analyzed first. The frequency pattern length 1 is represented by the lowest node. The conditional pattern base is a sub-database that contains prefix paths in the FP tree that start at the lowest node (suffix).
7) Create a Conditional FP Tree based on the number of itemsets in the route. The Conditional FP Tree considers the itemsets that meet the threshold support.
8) The Conditional FP Tree generates Frequent Patterns.

Example Of FP-Growth Algorithm

Support threshold=50%, Confidence= 60%

Table 7: Purchased Itemset

Transaction	List of items
T1	I1,I2,I3
T2	I2,I3,I4
T3	I4,I5
T4	I1,I2,I4
T5	I1,I2,I3,I5

T6	I1,I2,I3,I4

Support threshold=50% => 0.5*6= 3 => min_sup=3

1. Count of each item.

Table 8: Items with frequency Count

Item	Count
I1	4
I2	5
I3	4
I4	4
I5	2

2. Sort the itemset in descending order.

Table 9: Itemset in descending order of their frequency

Item	Count
I2	5
I1	4
I3	4
I4	4

3. Build FP tree.

a) Considering the root node null.

b) The first scan of Transaction T1: I1, I2, I3 contains three items {I1:1}, {I2:1}, {I3:1}, where I2 is linked as a child to root, I1 is linked to I2, and I3 is linked to I1.

c) T2: I2, I3, I4 contains I2, I3, and I4, where I2 is linked to root, I3 is linked to I2, and I4 is linked to I3. But this branch would share the I2 node as common as it is already used in T1.

d) Increment the count of I2 by 1 and I3 is linked as a child to I2, I4 is linked as a child to I3. The count is {I2:2}, {I3:1}, {I4:1}.

e) T3: I4, I5. Similarly, a new branch with I5 is linked to I4 as a child is created.

f) T4: I1, I2, I4. The sequence will be I2, I1, and I4. I2 is already linked to the root node, hence it will be incremented by 1. Similarly I1 will be incremented by 1 as it is already linked with I2 in T1, thus {I2:3}, {I1:2}, {I4:1}.

g) T5:I1, I2, I3, I5. The sequence will be I2, I1, I3, and I5. Thus {I2:4}, {I1:3}, {I3:2}, {I5:1}.

h) T6: I1, I2, I3, I4. The sequence will be I2, I1, I3, and I4. Thus {I2:5}, {I1:4}, {I3:3}, {I4 1}.

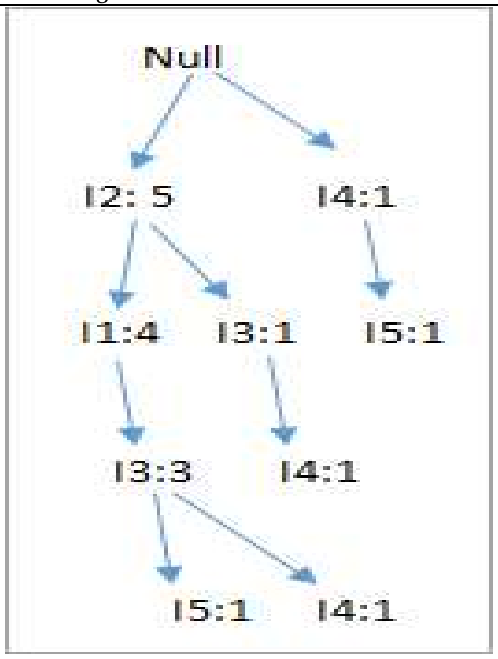


Figure 3: FP tree generation

Mining of FP-tree is summarized below:

1. The lowest node item I5 is not considered as it does not have a min support count, hence it is deleted.
2. The next lower node is I4. I4 occurs in 2 branches , {I2,I1,I3:,I41},{I2,I3,I4:1}. Therefore considering I4 as suffix the prefix paths will be {I2, I1, I3:1}, {I2, I3: 1}. This forms the conditional pattern base.
3. The conditional pattern base is considered a transaction database, an FP-tree is constructed. This will contain {I2:2, I3:2}, I1 is not considered as it does not meet the min support count.
4. This path will generate all combinations of frequent patterns : {I2,I4:2},{I3,I4:2},{I2,I3,I4:2}
5. For I3, the prefix path would be: {I2,I1:3},{I2:1}, this will generate a 2 node FP-tree : {I2:4, I1:3} and frequent patterns are generated: {I2,I3:4}, {I1:I3:3}, {I2,I1,I3:3}.
6. For I1, the prefix path would be: {I2:4} this will generate a single node FP-tree: {I2:4} and frequent patterns are generated: {I2, I1:4}.

Table 10: Process of Frequent Pattern Generation

Item	Conditional Pattern Base	Conditional FP-tree	Frequent Patterns Generated
I4	{I2,I1,I3:1},{I2,I3:1}	{I2:2, I3:2}	{I2,I4:2},{I3,I4:2},{I2,I3,I4:2}
I3	{I2,I1:3},{I2:1}	{I2:4, I1:3}	{I2,I3:4}, {I1:I3:3}, {I2,I1,I3:3}
I1	{I2:4}	{I2:4}	{I2,I1:4}

The Benefits of the FP Growth Algorithm

1. As compared to Apriori, which scans the transactions for each iteration, this algorithm only needs to scan the database twice.
2. This algorithm does not perform item pairing, which makes it faster.
3. The database is kept in memory in a compressed form.

4. It can be used to mine both long and short regular patterns and is both productive and scalable.

The FP-Growth Algorithm's Drawbacks

1. The FP Tree is more time-consuming and difficult to build than the Apriori.
2. It could be costly.
3. The algorithm can not fit in shared memory if the database is massive.

Discuss how the discovery of different patterns through different data miningalgorithms and visualisation techniques suggest you a simple pedagogical policy?

9.5 Applications of Association Rule Learning

It can be used in a variety of machine learning and data mining applications. The following are some of the most common uses of association rule learning:

Analysis of the Market Basket: One of the most well-known examples and implementations of association rule mining is this. Big retailers often employ this technique to evaluate the relationship between items.

Medical Diagnosis: Patients may be cured quickly using association guidelines, as they assist in determining the likelihood of infection with a specific condition.

Protein Sequence: The rules of association aid in the synthesis of artificial proteins.

It's also used for **Catalog Design, Loss-leader Analysis**, and a variety of other tasks.

Summary

- In selective marketing, decision analysis, and business management, the discovery of frequent trends, correlations, and correlation relationships among massive amounts of data is useful.
- A popular area of application is **market basket analysis**, which studies customers' buying habits by searching for itemsets that are frequently purchased together.
- The process of "association rule mining" entails first identifying frequent itemsets (groups of items, such as A and B, that satisfy a minimum support threshold, or percentage of task-related tuples), and then generating strong association rules in the form of A=>B.
- For frequent itemset mining, several powerful and scalable algorithms have been created, from which association and correlation rules can be extracted. These algorithms can be classified into three categories: (1) Apriori-like algorithms, (2) frequent pattern growth–based algorithms such as FP-growth.

Keywords

Antecedent: An antecedent is something that's found in data, and a consequent is an item that is found in combination with the antecedent.

Support: Support indicates how frequently the if/then relationship appears in the database.

Confidence: Confidence tells about the number of times these relationships are true.

Frequent Itemsets: The sets of an item that has minimum support

Apriori Property: Any subset of frequent item-set must be frequent.

Join Operation: To find Lk, a set of candidate k-item-sets is generated by joining Lk-1 with itself.

Sequence analysis algorithms: This type summarizes frequent sequences or episodes in data.

Association algorithms: This type of algorithm finds correlations between differentattributes in a dataset. The most common application of this kind of algorithm is for creating association rules, which can be used in a market analysis.

Self Assessment

1.A collection of one or more items is called as _____

A. Itemset
B. Support
C. Confidence
D. Support Count

2. Frequency of occurrence of an itemset is called as _____

A. Support
B. Confidence
C. Support Count
D. Rules

3. An itemset whose support is greater than or equal to a minimum support threshold is _____

A. (a)Itemset
B. (b)Frequent Itemset
C. (c)Infrequent items
D. (d)Threshold values

4. What does FP growth algorithm do?

A. It mines all frequent patterns through pruning rules with lesser support
B. It mines all frequent patterns through pruning rules with higher support
C. It mines all frequent patterns by constructing a FP tree
D. It mines all frequent patterns by constructing an itemsets

5. What do you mean by support(A)?

A. Total number of transactions containing A
B. Total Number of transactions not containing A
C. Number of transactions containing A / Total number of transactions
D. Number of transactions not containing A / Total number of transactions

6. Which of the following is the direct application of frequent itemset mining?

A. Social Network Analysis
B. Market Basket Analysis
C. Outlier Detection
D. Intrusion Detection

7. What is not true about FP growth algorithms?

A. It mines frequent itemsets without candidate generation
B. There are chances that FP trees may not fit in the memory
C. FP trees are very expensive to build
D. It expands the original database to build FP trees

8. When do you consider an association rule interesting?

A. If it only satisfies min_support
B. If it only satisfies min_confidence
C. If it satisfies both min_support and min_confidence
D. There are other measures to check so

9. What is the relation between a candidate and frequent itemsets?

A. A candidate itemset is always a frequent itemset
B. A frequent itemset must be a candidate itemset
C. No relation between these two
D. Strong relation with transactions

10. Which algorithm requires fewer scans of data?

A. Apriori
B. FP Growth
C. Naive Bayes
D. Decision Trees

11. For the question given below consider the data Transactions :

A. I1, I2, I3, I4, I5, I6
B. I7, I2, I3, I4, I5, I6
C. I1, I8, I4, I5
D. I1, I9, I10, I4, I6
E. I10, I2, I4, I11, I5

With support as 0.6 find all frequent itemsets?

A. <I1>, <I2>, <I4>, <I5>, <I6>, <I1, I4>, <I2, I4>, <I2, I5>, <I4, I5>, <I4, I6>, <I2, I4, I5>
B. <I2>, <I4>, <I5>, <I2, I4>, <I2, I5>, <I4, I5>, <I2, I4, I5>
C. <I11>, <I4>, <I5>, <I6>, <I1, I4>, <I5, I4>, <I11, I5>, <I4, I6>, <I2, I4, I5>
D. <I1>, <I4>, <I5>, <I6>

12. What is association rule mining?

A. Same as frequent itemset mining
B. Finding of strong association rules using frequent itemsets
C. Using association to analyze correlation rules
D. Finding Itemsets for future trends

13. The basic idea of the apriori algorithm is to generate_____ item sets of a particular size & scans the database.

A. Primary.
B. Candidate.
C. Secondary.
D. Superkey.

14. This approach is best when we are interested in finding all possible interactions among a set of attributes.

A. Decision tree
B. Association rules
C. K-Means algorithm
D. Genetic learning

15. Which of the following is not a frequent pattern mining algorithm?

A. Apriori
B. FP growth
C. Decision trees
D. Eclat

Answer for Self Assessment

1.	A	2.	C	3.	B	4.	C	5.	C
6.	B	7.	D	8.	C	9.	B	10.	B
11.	A	12.	B	13.	B	14.	B	15.	C

Review Questions

Q1) The 'database' below has four transactions. What association rules can be found in this set, if the

minimum support (i.e coverage) is 60% and the minimum confidence (i.e. accuracy) is 80% ?

Trans_id Itemlist

T1 {K, A, D, B}

T2 {D, A C, E, B}

T3 {C, A, B, E}

T4 {B, A, D}

Q2) With appropriate example explain the concept of support and confidence in Association rule mining.

Q3) Discuss the various applications of Association rule learning.

Q4) Step-by-step exaplin the working of Apriori algorithm.

Q5) Differentiate between apriori and FP tree algorithm.

Q6) Elucidate the process how frequent items can be mined using FPtree.

Further Readings

Gkoulalas-Divanis, A., & Verykios, V. S. (2010). *Association rule hiding for data mining* (Vol. 41). Springer Science & Business Media.

Ventura, S., & Luna, J. M. (2016). *Pattern mining with evolutionary algorithms* (pp. 1-190). Berlin: Springer.

Fournier-Viger, P., Lin, J. C. W., Nkambou, R., Vo, B., & Tseng, V. S. (2019). *High-utility pattern mining*. Springer.

Elizabeth Vitt, Michael Luckevich, Stacia Misner (2010). "*Business Intelligence*".O'Reilly Media, Inc.

Rajiv Sabhrwal, Irma Becerra-Fernandez (2010). "*Business Intelligence*". John Wiley & Sons

https://www.javatpoint.com/apriori-algorithm-in-machine-learning

https://www.softwaretestinghelp.com/apriori-algorithm/

https://towardsdatascience.com/fp-growth-frequent-pattern-generation-in-data-mining-with-python-implementation-244e561ab1c3

https://www.mygreatlearning.com/blog/understanding-fp-growth-algorithm/

https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html

.

Unit 10: Clustering Algorithms and Cluster Analysis

CONTENTS

Objectives

After this lecture, you will be able to

- Learn the various measures of similarity.
- Know the concept of unsupervised learning.
- Understand the working of the K-Means Algorithm.
- Understand the working of K-medoids and Clarans clustering algorithms.
- Understand the working of various hierarchal clustering algorithms.
- Learn the concept of cluster analysis.
- Understand the concept of outliers and the different types of outliers.

Introduction

Clustering is the process of grouping objects that are identical to one another. It can be used to determine if two things have similar or dissimilar properties. Clustering aids in the division of data into subsets. The data in each of these subsets is comparable, and these subsets are referred to as clusters. We can make an informed conclusion on who we believe is most suited for this product now that the data from our consumer base has been separated into clusters.

Outliers are cases that are out of the ordinary because they fall outside of the data's normal distribution. The distance from a normal distribution center reflects how typical a given point is

concerning the data's distribution. Each situation can be categorized as normal or atypical based on the likelihood of being typical or atypical.

10.1 Measures of Similarity

The similarity metric is a distance between two objects with dimensions that describe their characteristics. That is, if the distance between two data points is small, the objects have a high degree of similarity, and vice versa. The similarity is a subjective concept that is highly influenced by context and application.

The taste, size, color, and other characteristics of vegetables, for example, can be used to determine their similarity.

To evaluate the similarities or differences between two objects, most clustering methods use distance measures. The most commonly used distance measures are:

Euclidean Distance

The standard metric for solving geometry problems is Euclidean distance. It is the ordinary distance between two points, to put it simply. It is one of the most widely used cluster analysis algorithms. K-mean is one of the algorithms that use this formula.Mathematically it computes the root of squared differences between the coordinates between two objects.

$$d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

Figure 1:Formula for Euclidean Distance

Manhattan Distance

This defines the absolute difference between two coordinate pairs. To evaluate the distance between two points P and Q, simply measure the perpendicular distance between the points from the X-Axis and the Y-Axis.In a plane with P at coordinate (x1, y1) and Q at (x2, y2).

Manhattan distance between P and Q = |x1 – x2| + |y1 – y2|

Minkowski Distance

It's a combination of the Euclidean and Manhattan Distance Measurements. A point in an N-dimensional space is defined as (x1, x2, ..., xN)

Consider the following two points, P1 and P2:

P1: (X1, X2, ..., XN)

P2: (Y1, Y2, ..., YN)

The Minkowski distance between P1 and P2 is then calculated as follows:

$$\sqrt[p]{(x1 - y1)^p + (x2 - y2)^p + \ldots + (xN - yN)^p}$$

Figure 2:Minkowski Distance Formula

- When **p = 2**, Minkowski distance is the same as the **Euclidean** distance.
- When **p = 1**, Minkowski distance is the same as the **Manhattan** distance.

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

Figure 3: Similarity/Dissimilarity for Simple Attributes

Where p and q are the attribute values for two data objects.

Common Properties of a Distance

Distances, such as the Euclidean distance, have some well-known properties.

1. d(p,q)≥0 for all p and q and d(p, q) = 0 only if p = q. (Positive definiteness)
2. d(p, q) = d(q, p) for all p and q. (Symmetry)
3. d(p,r)≤d(p, q)+d(q,r) for all points p, q, and r. (Triangle Inequality)

where d(p, q) is the distance (dissimilarity) between points (data objects), p and q.A distance that satisfies these properties is a metric.

Common Properties of a Similarity

Similarities, also have some well-known properties.

1. s(p, q) = 1 (or maximum similarity) only if p = q.
2. s(p, q) = s(q, p) for all p and q. (Symmetry)

where *s(p, q)* is the similarity between points (data objects), *p* and *q*.

10.2 Unsupervised Learning

Unsupervised learning is a type of machine learning in which models are trained using an unlabeled dataset and are allowed to act on that data without any supervision.Unsupervised learning is the process of training a computer using data that hasn't been categorized or named, and then allowing the algorithm to operate on that data without supervision. The machine's job here is to sort unsorted data into groups based on similarities, patterns, and differences without any previous data training.

Unlike supervised learning, there is no instructor present, which means the computer will not be trained. As a result, the computer is limited in its ability to discover hidden structures in unlabeled data on its own.

The goal of unsupervised learning is to:

- find the underlying structure of the dataset
- the group that data according to similarities
- represent that dataset in a compressed format.

There are two types of algorithms for unsupervised learning:

- **Clustering:** A clustering problem is one in which you want to find the data's underlying groupings, such as grouping customers based on their shopping habits.
- **Association:**People who buy X also want to buy Y, for example, is an example of an association rule learning problem in which you want to discover rules that describe large portions of your data.

Working of Unsupervised Learning

Working of unsupervised learning can be understood by the below diagram:

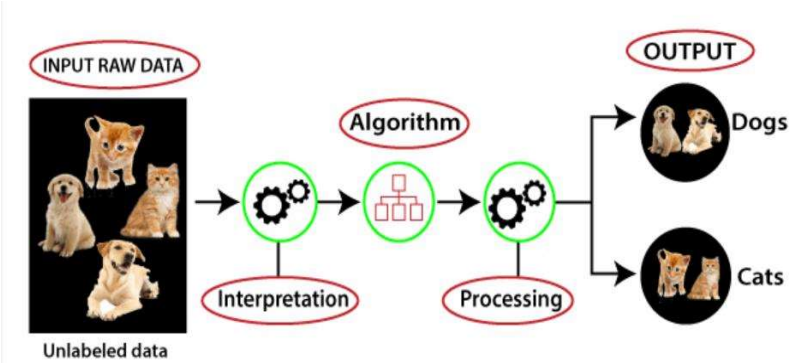


Figure 4: Unsupervised Learning

Assume it is shown a picture of both dogs and cats that it has never seen before.As a result, the machine has no understanding of the characteristics of dogs and cats, and we are unable to classify it as such. However, it can classify them based on their similarities, patterns, and differences, allowing us to easily divide the above image into two parts. The first part may contain all photos of dogs, while the second part may contain all photos of cats. It allows the model to work on its own to discover patterns and information that was previously undetected. It mainly deals with unlabelled data.

Recommender systems, which involve grouping together users with similar viewing patterns to recommend similar content.

10.3 K-Means Clustering

K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning. It is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.The k-means clustering algorithm mainly performs two tasks:

- Determines the best value for K center points or centroids by an iterative process.
- Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

Working of K-Means Algorithm

Step-1: Select the number K to decide the number of clusters.Suppose we have two variables M1 and M2. The x-y axis scatter plot

of these two variables is given below:

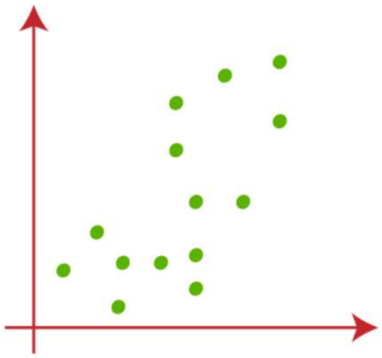


Figure 5: Scatter plot of M1 and M2

Step-2:Select random K points or centroids. It can be other from the input dataset. We need to choose some random k points or centroid to form the cluster. These points can be either the points from the dataset or any other point.

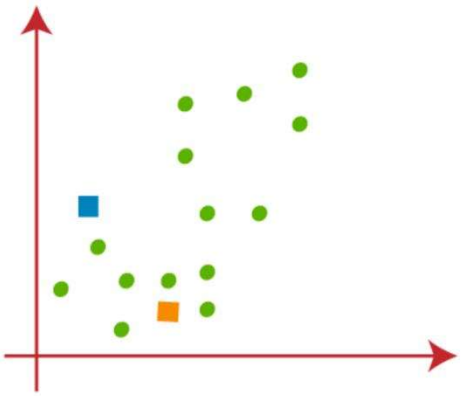


Figure 6: Selection of Centroid

Step-3:Assign each data point to their closest centroid, which will form the predefined K clusters.Now we will assign each data point of the scatter plot to its closest K-point or centroid. So, we will draw a median between both the centroids

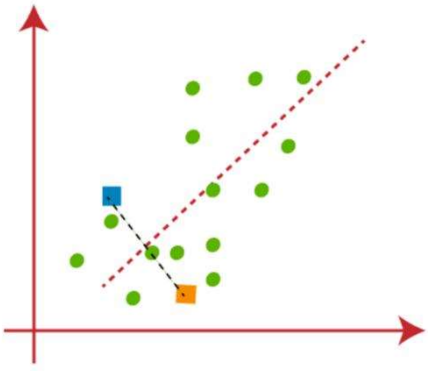


Figure 7: Assigning of data points to their closest centroid

Step-4:Calculate the variance and place a new centroid of each cluster.From the previous image, it is clear that points left side of the line are near to the blue centroid, and points to the right of the line are close to the yellow centroid. Let's color them as blue and yellow for clear visualization.

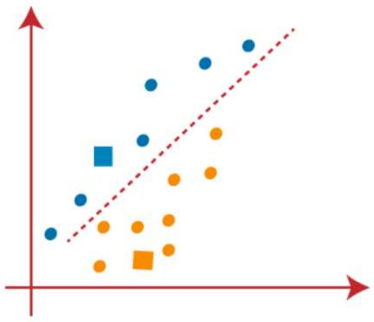


Figure 8: New centroid based upon new variance

Step-5:Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.As we need to find the closest cluster, so we will repeat the process by choosing **a new centroid**. To choose the new centroids, we will compute the center of gravity of these centroids and will find new centroids

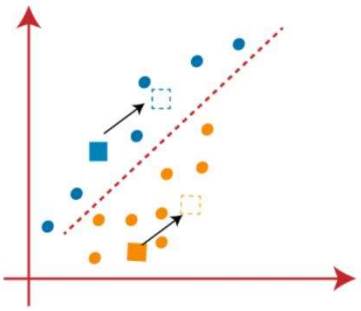


Figure 9: Reassign each point to the new closest centroid

Step-6:If any reassignment occurs, then go to step-4 else go to FINISH.From the above image, we can see, one yellow point is on the left side of the line, and two blue points are right to the line. So, these three points will be assigned to new.

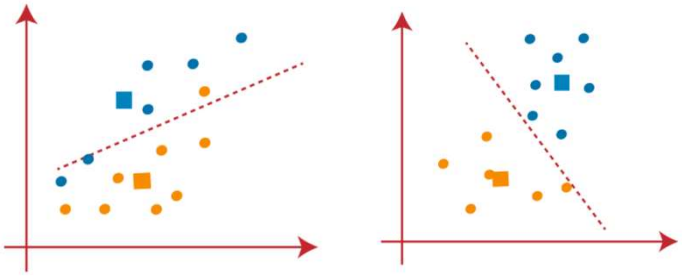


Figure 10: Points after reassignment

Step-7: The model is ready.As we got the new centroids so again will draw the median line and reassign the data points. As our model is ready, so we can now remove the assumed centroids and the two final clusters

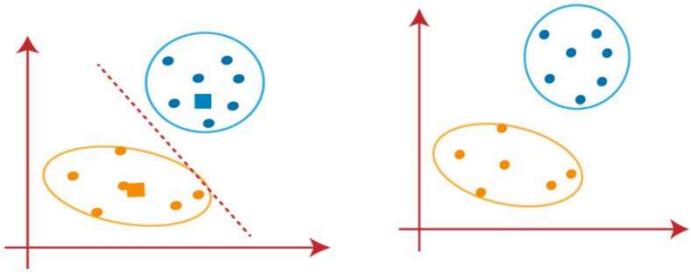


Figure 11: Final clusters

10.4 K-Medoids clustering(PAM)

A medoid is a point in a cluster whose dissimilarities with all other points in the cluster is its minimum.

E = |Pi - Ci| is used to measure the dissimilarity of the medoid(Ci) and object(Pi).

In the K-Medoids algorithm, the cost is given as:

Algorithm

1. Initialize select k random points out of the n data points as the medoids.
2. Associate each data point to the closest medoid by using any common distance metric methods.
3. While the cost decreases:

For each medoid m, for each data o point which is not a medoid:

1. Swap m and o, associate each data point to the closest medoid, recompute the cost.
2. If the total cost is more than that in the previous step, undo the swap.

Let's consider the following example:

	X	Y
0	8	7
1	3	7
2	4	9
3	9	6
4	8	5
5	5	8
6	7	3
7	8	4
8	7	5
9	4	5

Figure 12: Random Selection of Mediods

Step 1: Let the randomly chosen two medoids be C1 -(4, 5) and C2 -(8, 5) respectively.

Step 2: Calculating cost. Each non-medoid point's dissimilarity to the medoids is measured and tabulated:

	X	Y	Dissimilarity from C1	Dissimilarity from C2
0	8	7	6	2
1	3	7	3	7
2	4	9	4	8
3	9	6	6	2
4	8	5	-	-
5	5	8	4	6
6	7	3	5	3
7	8	4	5	1
8	7	5	3	1
9	4	5	-	-

Figure 13: Similarity and dissimilarity index based upon selected centroids

Each point is assigned to the medoid cluster with less dissimilarity. Cluster C1 is represented by points 1, 2, 5; cluster C2 is represented by the points 0, 3, 6, 7, 8.

The cost= (3 + 4 + 4)+ (3 + 1 + 1 + 2 + 2 2 + 2 + 2) = 20

Step 3: Pick one non-medoid point at random and recalculate the cost. Allow the point to be chosen at random (8, 4). Each non-medoid point's dissimilarity to the medoids – C1 (4, 5) and C2 (8, 4) – is calculated and tabulated.

	X	Y	Dissimilarity from C1	Dissimilarity from C2
0	8	7	6	3
1	3	7	3	8
2	4	9	4	9
3	9	6	6	3
4	8	5	4	1
5	5	8	4	7
6	7	3	5	2
7	8	4	-	-
8	7	5	3	2
9	4	5	-	-

Figure 14: Randomly selected new centroids

Each point is assigned to that cluster whose dissimilarity is less. So, points 1, 2, 5 go to cluster C1 and 0, 3, 6, 7, 8 go to cluster C2.
The New cost = (3 + 4 + 4) + (2 + 2 + 1 + 3 + 3) = 22
Swap Cost = New Cost – Previous Cost = 22 – 20 and 2 >0

As the swap cost is not less than zero, we undo the swap. Hence (3, 4) and (7, 4) are the final medoids. The clustering would be in the following way

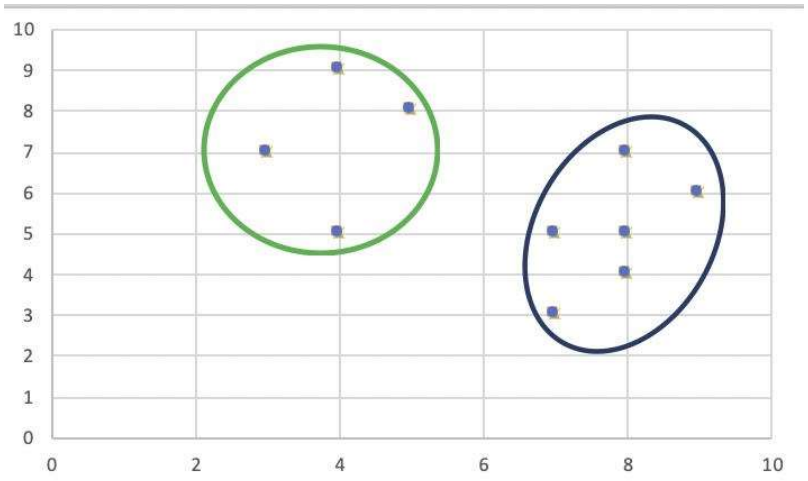


Figure 15: Cluster creation using K-Medoid

The time complexity is $O(k*(n-k)^2)$.

Advantages

1. The K-Medoid Algorithm is a fast algorithm that converges in a fixed number of steps.
2. It is simple to understand and easy to implement.
3. PAM is less sensitive to outliers than other partitioning algorithms.

Disadvantages

1. The key drawback of K-Medoid algorithms is that they cannot cluster non-spherical (arbitrary shaped) groups of points. This is because it relies on minimizing the distances between non-medoid objects and the medoid (cluster center) – in other words, it clusters based on compactness rather than connectivity.
2. Since the first k medoids are chosen at random, it can produce different results for different runs on the same dataset.

Which method is more robust—k-means or k-medoids?

The k-medoids method is more robust than k-means.Because in the presence of noise and outliers medoid is less influenced by outliers or other extreme values than a mean.

10.5 The CLARANS algorithm

Clustering Large Applications based upon RANdomized Search. It is an efficient medoid-based clustering algorithm. The k-medoids algorithm is an adaptation of the k-means algorithm.

CLARANS is a clustering partitioning method that is particularly useful in spatial data mining. By spatial data mining, we mean recognizing patterns and relationships in spatial data (such as distance-related, direction-related, or topological data, such as data plotted on a road map).CLARANS has two parameters:

max neighbor

- The maximum number of neighbors examined

numerical

- The number of local minima obtained.
- The higher the value of max neighbor, the closer is CLARANS to PAM

Steps of CLARANS algorithm

1. For the time being, select 'k' random data points and mark them as medoids.
2. Choose a random point, say 'a,' from the points chosen in step 1, as well as a point, say 'b,' that isn't included in those points.
3. Since that computation is needed for selecting the points in step 2, we would already have the number of distances of point 'a' from all other points (1). Carry out a similar calculation for point 'b.'
4. Replace 'a' with 'b' if the number of distances from all other points for point 'b' is less than that for point 'a'.
5. The algorithm runs a randomized search of medoids 'x' times, where 'x' is the number of local minima computed, i.e. the number of iterations to run, which we define as a parameter. The set of medoids obtained after such an 'x' number of measures is referred to as the **Local optimum.**
6. Any time substitution of points is made, a counter is incremented. The method of looking for potential replacement points is repeated until the counter reaches the maximum number of neighbors to be tested (specified as a parameter).
7. When the algorithm ends, the set of medoids obtained is the best local optimum option of medoids.

CLARANS –Basic Idea

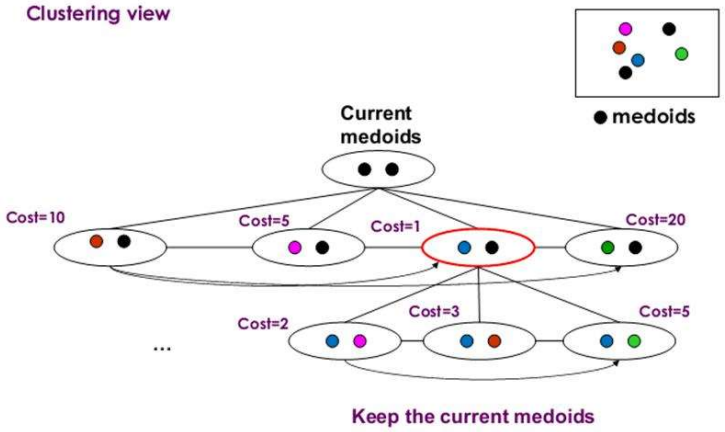


Figure 16: Selection of medoids

Draws a sample of nodes at the beginning of the search.Neighbors are from the chosen sample.Restricts the search to a specific area of the original data

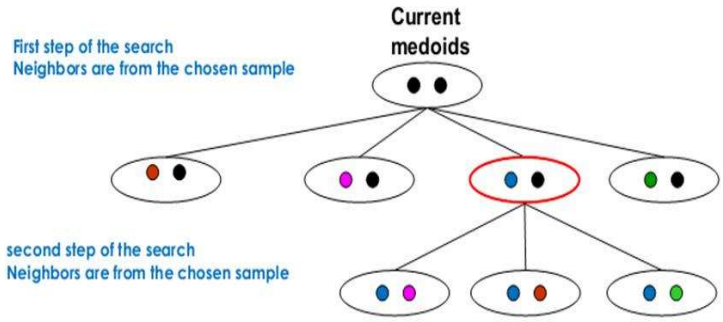


Figure 17: Bes local optimum.

Do not confine the search to a localized area. Stops the search when a local minimum is found. Finds several local optimums and output the clustering with the best local optimum.

Advantages

- Experiments show that CLARANS is more effective than both PAM and CLARA.
- Handles outliers

Disadvantages

- The computational complexity of CLARANS is O(n2), where n is the number of objects.
- The clustering quality depends on the sampling method.

10.6 Hierarchical Clustering

Hierarchical clustering begins by treating every data point as a separate cluster. Then, it repeatedly executes the subsequent steps:

a. Identify the 2 clusters which can be closest together

b. Merge the 2 maximum comparable clusters.

c. We need to continue these steps until all the clusters are merged.

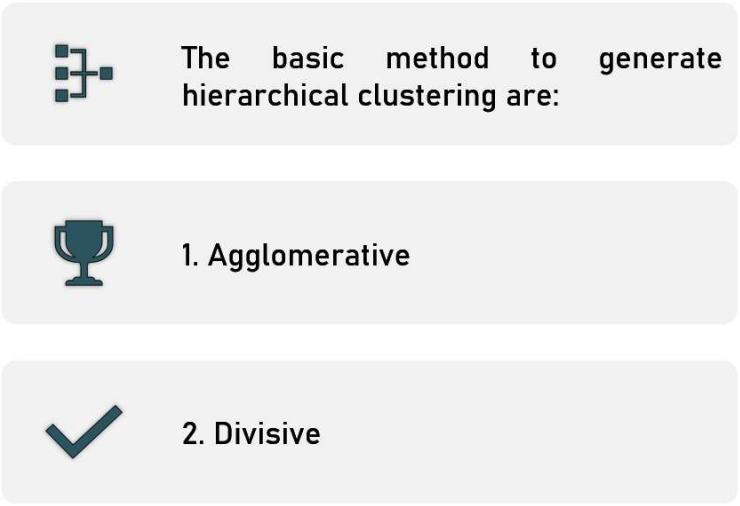


Figure 18: Types of Clustering algorithm

The basic method to generate hierarchical clustering are:

1.Agglomerative:

Consider and data point as a separate Cluster at first, then combine the cluster's nearest pairs at each stage. (It's a bottom-up approach.) At first, each data set is regarded as a separate entity or cluster. The clusters combine with other clusters in each iteration until only one cluster is formed.

Agglomerative Hierarchical Clustering Algorithm:

1. Calculate the degree of similarity between one cluster and all others (calculate proximity matrix)
2. Consider each data point as an individual cluster.
3. Combine clusters that are very similar or identical to one another.
4. For each cluster, recalculate the proximity matrix.
5. Steps 3 and 4 should be repeated until only one cluster remains.

Let's say we have six data points **A, B, C, D, E, F**.

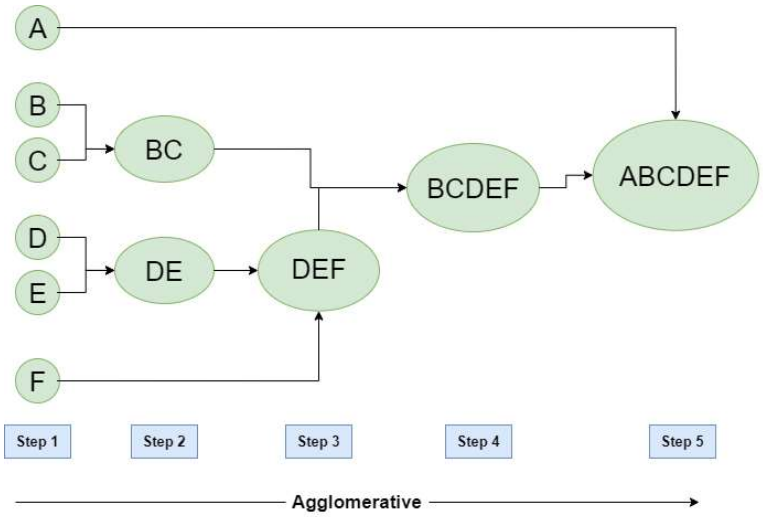


Figure 19: Agglomerative clustering

Step 1: Treat each alphabet as a separate cluster and measure the distance between each cluster and the others.

Step 2: Comparable clusters are combined to create a single cluster in the second step. Let's assume clusters (B) and (C) are very close, so we merge them in the second stage, just as we did with clusters (D) and (E), and we end up with clusters [(A), (BC), (DE), (F)].

Step 3: Using the algorithm, we recalculate the proximity and combine the two closest clusters ([(DE), (F)]) to form new clusters as [(A), (BC), (DEF)].

Step 4: Repeat the process; the DEF and BC clusters are comparable and are combined to create a new cluster. Clusters [(A), (BCDEF)] are now all that's left.

Step 5: Finally, the remaining two clusters are combined to create a single cluster [(ABCDEF)].

2. Divisive

Divisive Hierarchical Clustering is the exact opposite of Agglomerative Hierarchical Clustering. We consider all of the data points as a single cluster in Divisive Hierarchical clustering, and we distinguish the data points from the clusters that aren't equivalent in each iteration. We're left with N clusters in the end.

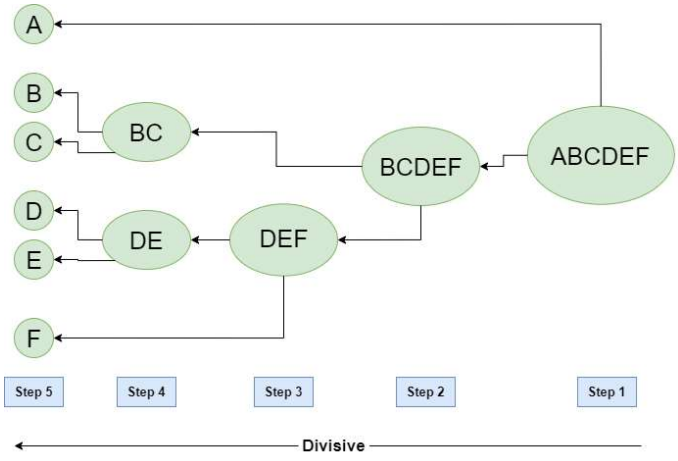


Figure 20: Divisive Clustering

10.7 BIRCH

BIRCH (Balanced Iterative Reducing and Clustering Using Hierarchies) is a clustering algorithm that clusters large datasets by first producing a small and compact summary of the large dataset that preserves as much information as possible. Instead of clustering the larger dataset, this smaller overview is clustered.

Why BIRCH?

Clustering algorithms like K-means clustering do not perform clustering very efficiently and it is difficult to process large datasets with a limited amount of resources (like memory or a slower CPU). Regular clustering algorithms do not scale well in terms of running time and quality as the size of the dataset increases. This is where BIRCH clustering comes in.

Before we implement BIRCH, we must understand two important terms:

- **Clustering Feature (CF)**
- **CF – TreeClustering Feature (CF)**

Clustering Feature: BIRCH summarizes large datasets into smaller, dense regions called Clustering Feature (CF) entries.Formally, a Clustering Feature entry is defined as an ordered triple, (N, LS, SS).A CF entry can be composed of other CF entries.

CF Tree: The CF tree is the compact representation we've been discussing up to this stage. A CF tree has a sub-cluster at each leaf node. A pointer to a child node and a CF entry made up of the number of CF entries in the child nodes are included in each entry in a CF tree. Each leaf node has a maximum number of entries. The threshold is the highest possible number.

Parameters of BIRCH Algorithm

The following are the various parameters of the BIRCH Algorithm

Threshold:the threshold is the maximum number of data points a sub-cluster in the leaf node of the CF tree can hold.

branching_factor: This parameter specifies the maximum number of CF sub-clusters in each node (internal node).

n_clusters: The number of clusters to be returned after the entire BIRCH algorithm is complete. If set to None, the final clustering step is not performed and intermediate clusters are returned.

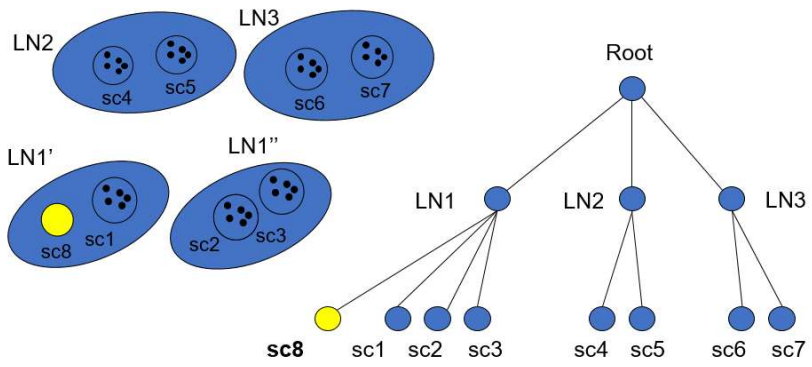


Figure 21: CF Tree Insertion

If the branching factor of a leaf node can not exceed 3 then the following tree results.

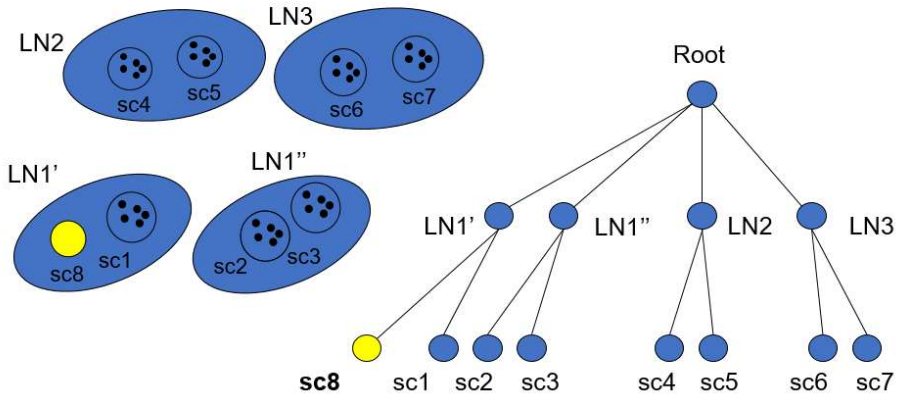


Figure 22: Leaf Branching Factor does not exceed 3

If the branching factor of a non-leaf node can not exceed 3, then the root is split, and the height of the CF Tree increases by one.

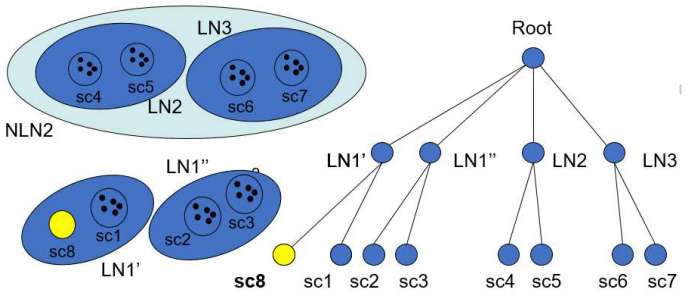


Figure 23: branching factor of a non-leaf node can not exceed 3

Advantages

- Birchperforms faster than existing algorithms (CLARANS and KMEANS) on large datasets.
- Scans whole data only once.
- Handles outliers better.

- Superior to other algorithms in stability and scalability.

Disadvantages

- Since each node in a CF tree can hold only a limited number of entries due to the size, a CF tree node doesn't always correspond to what a user may consider a nature cluster.
- Moreover, if the clusters are not spherical, it doesn't perform well because it uses the notion of radius or diameter to control the boundary of a cluster.

10.8 What is Cluster Analysis

The process of dividing a set of input data into possibly overlapping, subsets, where elements in each subset are considered related by some similarity measure.

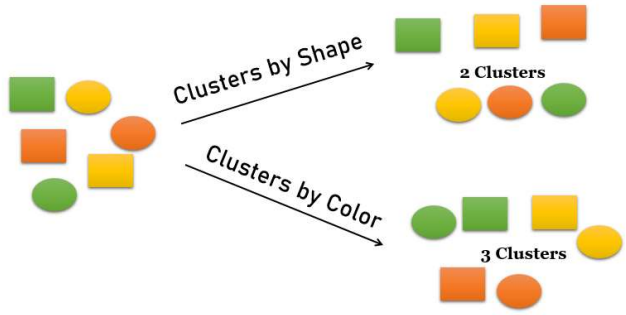


Figure 24: Clusters based upon shape and color

Clustering Approaches

The following are the two clustering approaches that we are going to discuss:

1) DBSCAN
2) Graph-Based Clustering

DBCSAN

The density-based clustering algorithm has been extremely useful in identifying non-linear shape structures. The most commonly used density-based algorithm is DBSCAN (Density-Based Spatial Clustering of Applications with Noise). It makes use of the density reachability and density connectivity concepts.

Density Reachability- A point "p" is said to be density reachable from a point "q" if point "p" is within ε distance from point "q" and "q" has a sufficient number of points in its neighbors which are within distance ε.

Density Connectivity - A point "p" and "q" are said to be density connected if there exists a point "r" which has a sufficient number of points in its neighbors and both the points "p" and "q" is within the ε distance. This is the chaining process. So, if "q" is neighbor of "r", "r" is neighbor of "s", "s" is a neighbor of "t" which in turn is neighbor of "p" implies that "q" is neighbor of "p".

Algorithm

The set of data points is X = x1, x2, x3,..., xn. Two parameters are needed by DBSCAN: (eps) and the minimum number of points required to form a cluster (minPts).

1) Begin at a random starting point that has never been visited before.

2) Using (All points within the distance are neighborhood), extract the neighborhood of this point.

3) If there are enough neighbors in the region, the clustering process begins, and the point is labeled as visited; otherwise, it is labeled as noise (Later this point can become part of the cluster).

4) If a point is found to be part of a cluster, its neighbors are also part of the cluster, and the process from step 2 is repeated for all neighborhood points. This process is repeated until all of the cluster's points have been calculated.

5) A new, previously unexplored point is retrieved and processed, resulting in the detection of a new cluster or noise.

6) Repeat this step until all of the points have been marked as visited.

DBSCAN algorithm requires two parameters

- **Eps:** Maximum radius of the neighborhood.
- **MinPts:** Minimum number of points in an Eps-neighbourhood of that point.

In this algorithm, we have 3 types of data points.

- **core point** - which has at least min pts points in its neighborhood
- **border point** - one which has a core point in its neighborhood
- **noise point** - one which is neither a core nor a border point and is considered an outlier in the dataset

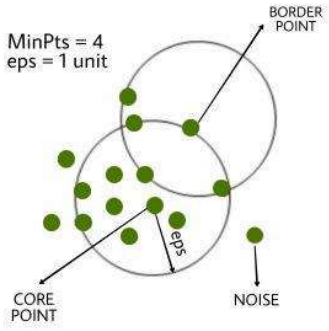


Figure 25: Points of DBSCAN

Advantages

1) No need to specify the number of clusters ahead of time.

2) Capable of detecting noise data during clustering.

3) Clusters of any size and form can be found using the DBSCAN algorithm.

Disadvantages

1) In the case of varying density clusters, the DBSCAN algorithm fails.

2) Fails in the case of datasets with a neck.

3) Doesn't work well for data with a lot of dimensions.

10.9 Graph-Based Clustering

Graph clustering refers to the clustering of data in the form of graphs. The following are the types of Graph Clustering:

- **Between-graph** - Clustering a set of graphs
- **Within-graph** - Clustering the nodes/edges of a single graph

Between-Graph: Between-graph clustering methods divide a set of graphs into different clusters.

A set of graphs representing chemical compounds can be grouped into clusters based on their structural similarity

Within Graph: Within-graph clustering methods divide the nodes of a graph into clusters.

In a social networking graph, these clusters could represent people with the same/similar hobbies

K-Spanning Tree

Following are the steps to obtain a K-Spanning Tree:

- Obtains the Minimum Spanning Tree (MST) of input graph G

- Removes k-1 edges from the MST
- Results in k clusters

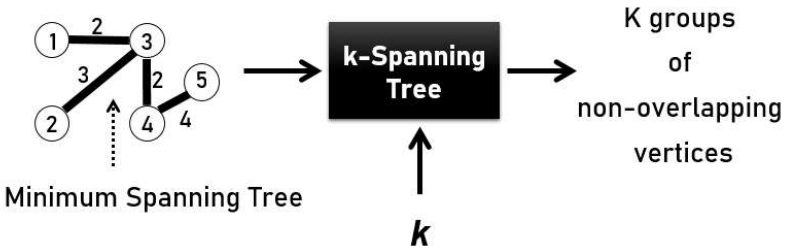


Figure 26: Working of K-Spanning tree

What is the Minimum Spanning tree

A spanning tree of a connected and undirected graph is a subgraph that is a tree that links all of the vertices together. Many different spanning trees may exist in a single graph.For a weighted, connected, undirected graph, a minimum spanning tree (MST) or minimum weight spanning tree is a spanning tree with a weightless than or equal to the weight of any other spanning tree. The weight of a spanning tree is the sum of the weights assigned to each of its edges.

How many edges does a minimum spanning tree has? A minimum spanning tree has (V – 1) edges where V is the number of vertices in the given graph.

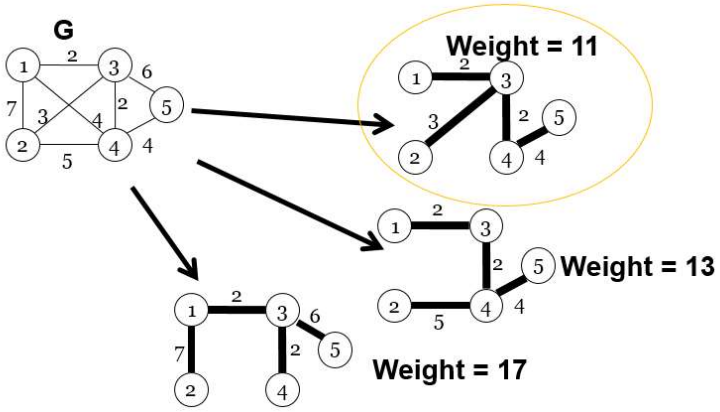


Figure 27:MST

maximum possible sum of edge weights, if the edge weights represent similarity

Prim's Algorithmto Obtain MST

Given Input Graph G

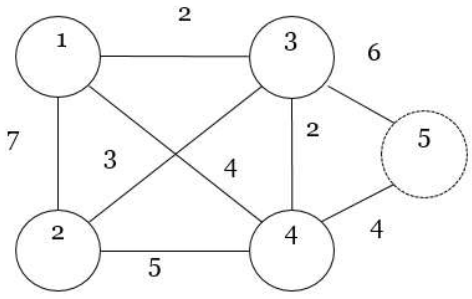


Figure 28: Graph

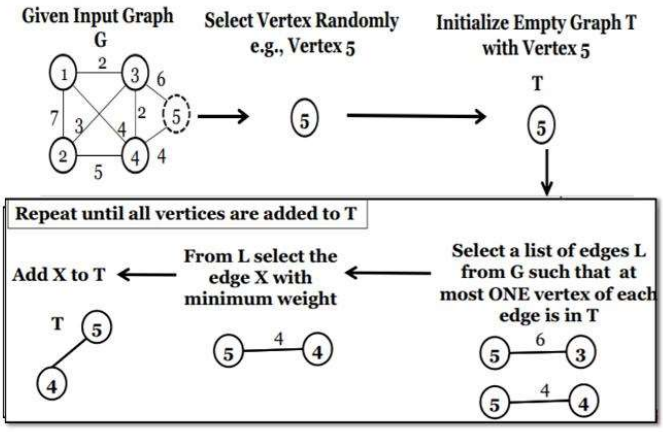


Figure 29: Steps for MST

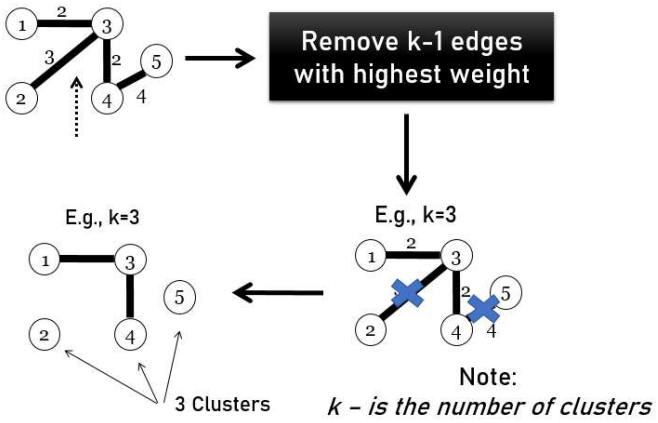


Figure 30: Formation of Cluster

10.10 Cluster Evaluation

Cluster evaluation assesses the feasibility of clustering analysis on a data set and the quality of the results generated by a clustering method.

Clustering evaluation includes the following tasks:

a) Assessing clustering tendency.

b) Determining the number of clusters in a data set.

c) Measuring clustering quality

Assessing Clustering Tendency

Before applying any clustering method to your data, it's important to evaluate whether the data sets contain meaningful clusters (i.e.: non-random structures) or not. If yes, then how many clusters are there. This process is defined as the assessment of **clustering tendency.** The **Hopkins statistic**is used to assess the clustering tendency of a data set by measuring the probability that a given data set is generated by uniform data distribution. In other words, it tests the spatial randomness of the data.

The Hopkins statistic can be calculated as follow:

$$H = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i}$$

A value for H higher than 0.75 indicates a clustering tendency at the 90% confidence level.

The null and the alternative hypotheses are defined as follow:

- **Null hypothesis**: the data set D is uniformly distributed (i.e., no meaningful clusters)
- **Alternative hypothesis**: the data set D is not uniformly distributed (i.e., contains meaningful clusters)

We can conduct the Hopkins Statistic test iteratively, using 0.5 as the threshold to reject the alternative hypothesis. That is, if H < 0.5, then it is unlikely that D has statistically significant clusters.

Determining the Number of Clusters

Determining the "right" number of clusters in a data set is important, not only because some clustering algorithms like k-means require such a parameter, but also because the appropriate number of clusters controls the proper granularity of cluster analysis.It can be regarded as finding a good balance between compressibility and accuracy in cluster analysis.

What if you were to treat the entire data set as a cluster? This would maximize the compression of the data, but such a cluster analysis has no value.

On the other hand, treating each object in a data set as a cluster gives the finest clustering resolution. (i.e., most accurate due to the zero distance between an object and the corresponding cluster center).Figuring out the right number of clusters should often depend on the distribution's shape and scale in the data set, as well as the clustering resolution required by the user.

 This approach of segmenting the database via clustering analysis is often used as anexploratory technique because it is not necessary for the end-user/analyst to identify ahead of time how records should be associated simultaneously.

Measuring Clustering Quality

After clustering is complete, a variety of metrics can be used to assess how well the clustering worked. Minimal intra-cluster distance and maximum inter-cluster distance characterize ideal clustering.In general, these methods can be categorized into two groups according to whether ground truth is available or not. Ground Truth is factual data that has been observed or measured and can be analyzed objectively. It has not been inferred. If the data is based on an assumption, subject to opinion, or up for discussion, then, by definition, that is not Ground Truth data.

If ground truth is available, it can be used by **extrinsic methods,** which compare the clustering against the group truth and measure.

If the ground truth is unavailable, we can use **intrinsic methods,** which evaluate the goodness of clustering by considering how well the clusters are separated. Ground truth can be considered as supervision in the form of "**cluster labels**." Hence, extrinsic methods are also known as supervised methods, while intrinsic methods are unsupervised methods.In general, a measure Q on clustering quality is effective if it satisfies the following four essential criteria:

Cluster Homogeneity:A clustering result satisfies homogeneity if all of its clusters contain only data points that are members of a single class.

Cluster completeness: A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster.

Rag-Bag: Elements with low relevance to the categories (e.g., noise) should be preferably assigned to the less homogeneous clusters (macro-scale, low-resolution, coarse-grained, or top-level clusters in a hierarchy).

 How can I evaluate whether the clustering results are good or not when I try out a clustering method on the data set?

10.11 Outlier Detection and Analysis

An outlier is a data object that deviates significantly from the normal objects as if it were generated by a different mechanism. An outlier is a unique entity that stands out from the rest of the group.

They may be the result of a calculation or execution mistake. Outlier analysis, also known as outlier mining, is the study of outlier data.

Unusual credit card purchase, sports: Michael Jordon, Wayne Gretzky

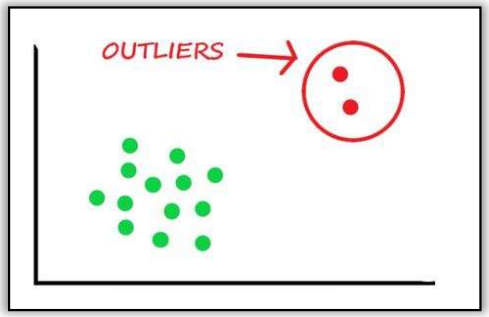


Figure 31:Outlier

Why outlier analysis?

Most data mining methods ignore outliers, noise, or exceptions; however, in some applications, such as fraud detection, unusual events may be more interesting than those that occur more often, and thus outlier analysis becomes essential.

Outlier Detection

To find the outlier, we must first set the threshold value such that every data point with a distance greater than it from its nearest cluster is considered an outlier for our purposes. The distance between the test data and each cluster means must then be determined.If the distance between the test data and the nearest cluster is greater than the threshold value, the test data will be classified as an outlier.

Algorithm for outlier detection

1. Calculate each cluster's average.
2. Initialize the Threshold value
3. Calculate the test data's distance from each cluster means.
4. Find the cluster that is closest to the test results.
5. Outlier exists if (Distance > Threshold).

Challenges of Outlier Detection

1. Modeling normal objects and outliers properly

- Hard to enumerate all possible normal behaviors in an application
- The border between normal and outlier objects is often a gray area

2. Application-specific outlier detection

- Choice of distance measure among objects and the model of the relationship among objects are often application-dependent

E.g., clinic data: a small deviation could be an outlier; while in marketing analysis, larger fluctuations

3. Handling noise in outlier detection

- Noise may distort the normal objects and blur the distinction between normal objects and outliers. It may help hide outliers and reduce the effectiveness of outlier detection

4. Understandability

- Understand why these are outliers: Justification of the detection

- Specify the degree of an outlier: the unlikelihood of the object being generated by a normal mechanism

Types of Outliers

The following are the different types of outliers:

- Global Outliers
- Contextual Outliers
- Collective Outliers.

Global Outliers

A data point is considered a global outlier if its value is far outside the **entirety** of the data set in which it is found.

A **global outlier** is a measured sample point that has a very high or a very low value relative to all the values in a dataset. For example, if 9 out of 10 points have values between 20 and 30, but the 10th point has a value of 85, the 10th point may be a **global outlier**.

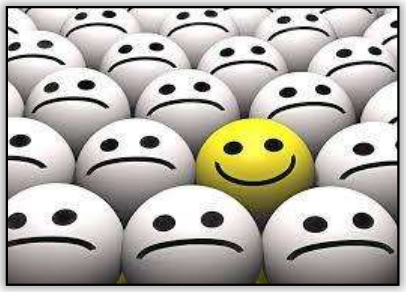


Figure 32: Global Outlier

Contextual Outliers

If an individual data point is different in a specific context or condition (but not otherwise), then it is termed as a contextual outlier.

Attributes of data objects should be divided into two groups:

• Contextual attributes: defines the context, e.g., time & location

• Behavioral attributes: characteristics of the object, used in outlier evaluation, e.g., temperature

Contextual outliers are hard to spot if there was no background information. If you had no idea that the values were temperatures in summer, it may be considered a valid data point.

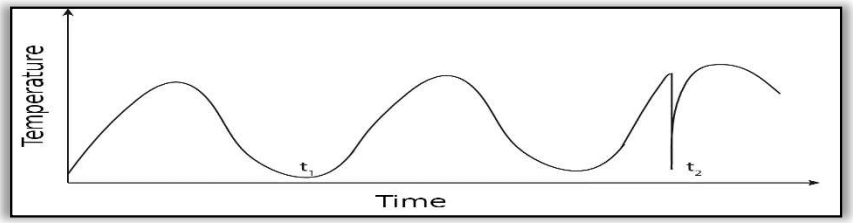


Figure 33: Contextual Outlier

Collective Outliers

A subset of data objects *collectively* deviates significantly from the whole data set, even if the individual data objects may not be outliers. When several computers keep sending denial-of-service packages to each other.

Applications: E.g., *intrusion detection*:

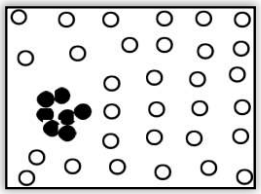


Figure 34:Collective Outliers

Give an example of a situation in which global outliers, contextual outliers, and collective outliers are all relevant. What are the characteristics, as well as the environmental and behavioral characteristics? In collective outlier detection, how is the link between items modeled?

Summary

- **Types of outliers** include global outliers, contextual outliers, and collective outliers.
- An object may be more than one type of outlier.
- A **density-based method** cluster objects based on the notion of density. It grows clusters either according to the density of neighborhood objects (e.g., in DBSCAN)or according to a density function (e.g., in DENCLUE). OPTICS is a density-basedmethod that generates an augmented ordering of the data's clustering structure.
- **Clustering evaluation** assesses the feasibility of clustering analysis on a data set andthe quality of the results generated by a clustering method.
- A cluster of data objects can be treated as one group.
- Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
- **Outliers** are the **data** points that cannot be fitted in any type of cluster.
- The analysis of outlier data is referred to as outlier analysis or outlier mining.

Keywords

Clustering:Clustering is a method of **data** analysis which groups **data** points to "maximizing the intraclass similarity and minimizing the interclass similarity.

Outliers: outliers are data items that did not (or are thought not to have) come from the assumed population of data.

Unsupervised learning: This term refers to the collection of techniques where groupings of the data are defined without the use of a dependent variable. Cluster analysis is an example.

Cluster Homogeneity:A clustering result satisfies homogeneity if all of its clusters contain only data points that are members of a single class.

Cluster Evaluation: Cluster evaluation assesses the feasibility of clustering analysis on a data set and the quality of the results generated by a clustering method.

Hopkins static: Hopkins statisticis used to assess the clustering tendency of a data set by measuring the probability that a given data set is generated by uniform data distribution.

Contextual attributes: Attributes that define the context, e.g., time & location

Self Assessment Questions

1. Which is needed by K-means clustering?

A.Defined distance metric

B. Number of clusters

C. Initial guess as to cluster centroids

D. All of these

2. K means and K-medoids are an example of which type of clustering method?

A. Hierarchical

B. Partition

C. Probabilistic

D. None of the above.

3.CLARANS stands for :

A. Clustering Large Applications based upon RANdomized Search

B. Clustering Long Applications based upon RANdomized Search

C. Clustering Large Accounts based upon RANdomized Search

D. Customize Large Applications based upon RANdomized Search

4.CLARANS combines sampling techniques with _____.

A. K-means

B. CLARA

C. PAM

D. All of the above

5.The k-medoids method is more robust than

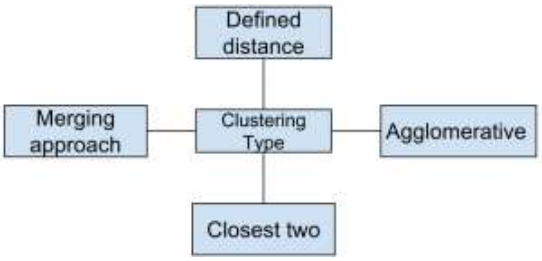
A. K-means

B. CLARA

C. PAM

D. CLARANS

6. Which of the following clustering type has characteristics shown in the below figure?



A. Partitional

B. Hierarchical

C. Naive Bayes

D. None of the mentioned

7. Which of the following is finally produced by Hierarchical Clustering?

A. Final estimate of cluster centroids

B. Tree showing how close things are to each other

C. Assignment of each point to clusters

D. All of the mentioned

8. Which of the following clustering requires a merging approach?

A. Partitional

B. Hierarchical

C. Naive Bayes

D. None of the mentioned

9. _____specifies the maximum number of data points a sub-cluster in the leaf node of the CF tree can hold.

A. n_clusters

B. Branching_factor

C. Threshold

D. All Of the above

10. Find the outliers in the given data set below.

68, 6, 74, 70, 82

A. 6

B. 82

C. 68

D. 70

11. Which statement about outliers is true?

A. Outliers should be part of the training dataset but should not be present in the test data.

B. Outliers should be identified and removed from a dataset.

C. The nature of the problem determines how outliers are used

D. Outliers should be part of the test dataset but should not be present in the training data.

12. What does the term 'outlier' mean?

A. A score that is left out of the analysis because of missing data

B. The arithmetic mean

C. A type of variable that cannot be quantified

D. An extreme value at either end of a distribution

13. Extreme values that occur infrequently are called as _____.

A. Outliers.

B. Rare values.

C. Dimensionality reduction.

D. All of the above.

14. Which of the following is required by K-means clustering?

A. Defined distance metric

B. Number of clusters

C. Initial guess as to cluster centroids

D. All of the mentioned

15. K-means is not deterministic and it also consists of several iterations.

A. True

B. False

C. Can't Say

D. Maybe

Answers for Self Assessment

1.	D	2.	B	3.	A	4.	C	5.	A
6.	B	7.	B	8.	B	9.	B	10.	A
11.	C	12.	D	13.	A	14.	D	15.	A

Review Questions

Q1) Give an application example where global outliers, contextual outliers, and collective outliers are all interesting. What are the attributes, and what are the contextual and behavioral attributes?

Q2) Briefly describe and give examples of each of the following approaches to clustering: partitioning methods, hierarchical methods, and density-based methods.

Q3) For example explain the different types of outliers.

Q4) Define hierarchal clustering along with its various types.

Q5) Explain the K-mean algorithm.

Q6) Differentiate between K-means and K-Mediod.

Q7) Elucidate the step-by-step working of K-Mediod with example.

Further Reading

Gan, G., Ma, C., & Wu, J. (2020). *Data clustering: theory, algorithms, and applications*. Society for Industrial and Applied Mathematics.

Long, B., Zhang, Z., & Philip, S. Y. (2010). *Relational data clustering: models, algorithms, and applications*. CRC Press.

Celebi, M. E. (Ed.). (2014). *Partitional clustering algorithms*. Springer.

Cabena, P., Hadjinian, P., Stadler, R., Verhees, J., & Zanasi, A. (1998). *Discovering data mining: from concept to implementation*. Prentice-Hall, Inc.

Cios, K. J., Pedrycz, W., Swiniarski, R. W., & Kurgan, L. A. (2007). *Data mining: a knowledge discovery approach*. Springer Science & Business Media.

Funatsu, K. (Ed.). (2011). *New fundamental technologies in data mining*. BoD–Books on Demand.

https://www.tutorialspoint.com/data_mining/dm_cluster_analysis.htm

https://towardsdatascience.com/17-clustering-algorithms-used-in-data-science-mining-49dbfa5bf69a

https://www.analytixlabs.co.in/blog/types-of-clustering-algorithms/

https://www.anblicks.com/blog/an-introduction-to-outliers/

Unit 11: Classification

CONTENTS

Objectives

After this Unit, you will be able to

- Understand the concept of supervised learning.
- Learn the concept of classification and various methods used for classification.
- Know the basic concept of binary classification.
- Understand the working of naïve Bayes classifier.
- Analyze the use and working of Association based and Rule-based classification.
- Know the working of the KNN algorithm.
- Understand the working of the Decision tree and Random forest algorithm.
- Learn the concept of Cross-Validation.

Introduction

In the process of data mining, large data sets are first sorted, then patterns are identified and relationships are established to perform data analysis and solve problems. Attributes represent different features of an object. Different types of attributes are:

- **Binary**: Possesses only two values i.e. True or False

- **Nominal**: When more than two outcomes are possible. It is in Alphabet form rather than being in Integer form.
- **Ordinal:** Values that must have some meaningful order.
- **Continuous:** May have an infinite number of values, it is in float type
- **Discrete:** Finite number of values.

11.1 Supervised Learning

As the name implies, supervised learning involves the involvement of a supervisor who often serves as an instructor. In a nutshell, supervised learning is when we teach or train a computer using well-labeled data. This means that certain information has already been labeled with the correct answer.

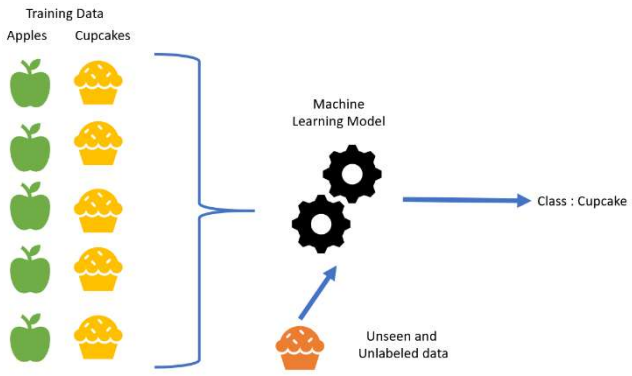


Figure 1: Supervised Learning

Steps Involved in Supervised Learning

- First Determine the type of training dataset
- Collect/Gather the labeled training data.
- Split the training dataset into a training dataset, test dataset, and validation dataset.
- Determine the input features of the training dataset, which should have enough knowledge so that the model can accurately predict the output.
- Determine the suitable algorithm for the model, such as support vector machine, decision tree, etc.
- Execute the algorithm on the training dataset. Sometimes we need validation sets as the control parameters, which are the subset of training datasets.
- Evaluate the accuracy of the model by providing the test set. If the model predicts the correct output, which means our model is accurate.

11.2 Classification

Classification is a data mining function that assigns items in a collection to target categories or classes.The goal of classification is to accurately predict the target class for each case in the data.

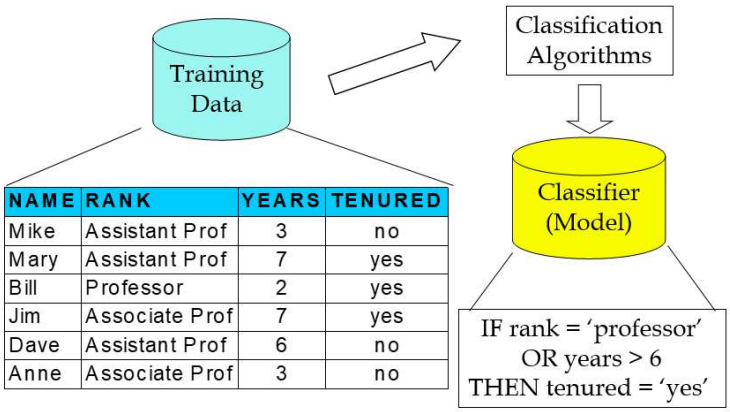
You may wish to use classification to predict whether the weather on a particular day will be "sunny", "rainy" or "cloudy". Popular classification techniques include decision trees and neural networks.

Classification is a two-step process:

(a) ***Learning:*** Training data are analyzed by a classification algorithm. Here, the class label attribute is a credit rating, and the learned model or classifier is represented in the form of a classification rule. In the learning step (or training phase), a classification algorithm builds the classifier by analyzing or "learning from" a training set.A tuple, X, is represented by an N-dimensional attribute vector,

$$X = \{x1, x2, \ldots\ldots xN\}$$

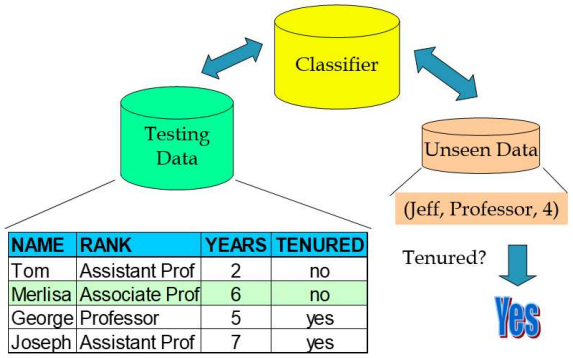
Each tuple, X, is assumed to belong to a predefined class as determined by another database attribute called the class label attribute.The individual tuples making up the training set are referred to as training tuples and are randomly sampled from the database under analysis.



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Figure 2: Learning Step

(b) ***Classification:*** Test data are used to estimate the accuracy of the classification rules. If the accuracy is considered acceptable, the rules can be applied to the classification of new data tuples.The model is used to predict class labels and testing the constructed model on test data and hence estimate the accuracy of the classification rules.



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	6	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Figure 3: Classification Step

11.3 Issues regarding Classification

To prepare the data for classification and prediction, the following preprocessing steps may be applied to the data to help improve the accuracy, efficiency, and scalability of the classification or prediction process.

1. ***Data cleaning:*** This refers to the preprocessing of data to remove or reduce noise (by applying smoothing techniques, for example), and the treatment of missing values (e.g., by replacing a missing value with the most commonly occurring value for that attribute, or with the most probable value based on statistics). Although most classification algorithms have some mechanisms for handling noisy or missing data, this step can help reduce confusion during learning.

2. ***Relevance analysis:*** Many of the attributes in the data may be irrelevant to the classification or prediction task. For example, data recording the day of the week on which a bank loan application was filled is unlikely to be relevant to the success of the application. Furthermore, other attributes may be redundant. Hence, relevance analysis may be performed on the data to remove any irrelevant or redundant attributes from the learning process. In machine learning, this step is known as feature selection. Including such attributes may otherwise slow down, and possibly mislead the learning step. Ideally,

the time spent on relevance analysis, when added to the time spent on learning from the resulting "reduced" feature subset, should be less than the time that would have been spent on learning from the original set of features. Hence, such analysis can help improve classification efficiency and scalability.

3. ***Data Transformation and reduction***: **Normalization** involves scaling all values for a given attribute to make them fall within a small specified range. It involves scaling all values for a given attribute so that they fall within a small specified range, such as -1.0 to 1.0, or 0 to 1.0. In methods that use distance measurements, for example, this would prevent attributes with initially large ranges (like, say income) from outweighing attributes with initially smaller ranges (such as binary attributes).
4. ***Generalization***: The data can also be transformed by generalizing it to the higher concept. For this purpose, we can use the concept hierarchies. This is particularly useful for continuous-valued attributes. For example, numeric values for the attributeincome may be generalized to discrete ranges such as low, medium, and high. Similarly, nominal-valued attributes, like streets, can be generalized to higher-level concepts, like a city. Since generalization compresses the original training data, fewer input/output operations may be involved during learning.

11.4 Types of Classifiers

Classifiers can be categorized into two major types:

- Discriminative
- Generative

Discriminative

It is a very basic classifier and determines just one class for each row of data. It tries to model just by depending on the observed data, depends heavily on the quality of data rather than on distributions.

Logistic Regression

Suppose there are few students and the Result of them are as follows :

Student 1 : Test Score: 9/10, Grades: 8/10 Result: Accepted

Student 2 : Test Score: 3/10, Grades: 4/10, Result: Rejected

Student 3: Test Score: 7/10, Grades: 6/10, Result: to be tested

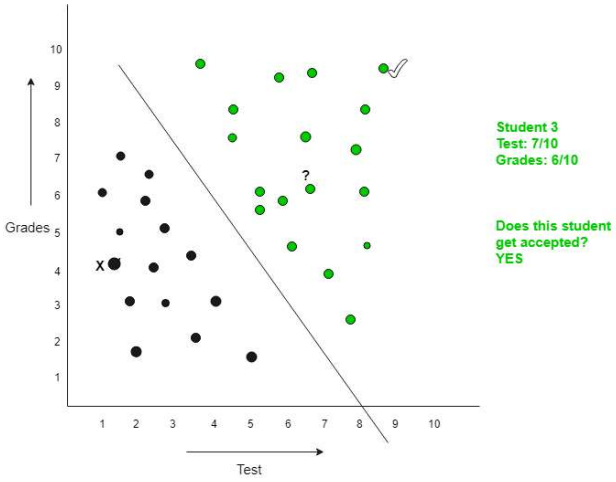


Figure 4: Discriminative Classifier

Generative

It models the distribution of individual classes and tries to learn the model that generates the data behind the scenes by estimating assumptions and distributions of the model. Used to predict the unseen data.

Naive Bayes Classifier

Detecting Spam emails by looking at the previous data. Suppose 100 emails and that too divided in 1:4 i.e. Class A: 25%(Spam emails) and Class B: 75%(Non-Spam emails). Now if a user wants to check that if an email contains the word cheap, then that may be termed as Spam. So, if the email contains the word cheap, what is the probability of it being spam ?? (= 80%).

11.5 Binary Classification

Binary classification is the task of classifying the elements of a set into two groups based on a classification rule. Typical binary classification problems include:

- Medical testing to determine if a patient has a certain disease or not;
- Quality control in industry, deciding whether a specification has been met;
- In information retrieval, deciding whether a page should be in the result set of a search or not.

Typically, binary classification tasks involve one class that is the normal state, and another class that is the abnormal state.

"cancer not detected" is the normal state of a task that involves a medical test and "cancer detected" is the abnormal state.

The class for the normal state is assigned the class label 0 and the class with the abnormal state is assigned the class label 1.

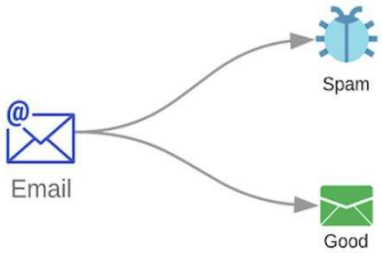


Figure 5: Binary Classification.

Popular algorithms that can be used for binary classification include:

- Logistic Regression
- k-Nearest Neighbors
- Decision Trees
- Support Vector Machine
- Naive Bayes

11.6 Why Bayesian Classification?

It is a statistical classifier, it can predict class membership probabilities such as the probability that a given tuple belongs to a particular class.It is based on Bayes' Theorem. A simple Bayesian classifier, naïve Bayesian classifier, has comparable performance with decision tree and selected neural network classifiers

Bayesian Theorem: Basics

Let X be a data sample ("*evidence*"): class label is unknown.Let H be a *hypothesis* that X belongs to class C.Classification is to determine P(H|X) is the posteriori probability, of H, conditioned on X.

- X is a 35-year-old customer with an income of $40,000.

Then P(H/X) reflects the probability that customer X will buy a computer given that we know the customer's age and income. In contrast, P(H) is the prior probability, of H. For our example, this is the probability that any given customer will buy a computer, regardless of age, income, or any other information.P(H), which is independent of X.P(X|H) (*posteriori probability*), the probability of X conditioned on H. That is, it is the probability that a customer, X, is 35 years old and earns $40,000, given that we know the customer will buy a computer.

P(X) is the prior probability of X. Using our example, it is the probability that a person from our set of customers is 35 years old and earns $40,000.

Bayesian Theorem

Given training data X, posteriori probability of a hypothesis H, P(H|X), follows the Bayes theorem

$$P(H|\mathbf{X}) = \frac{P(\mathbf{X}|H)P(H)}{P(\mathbf{X})}$$

Derivation of Naïve Bayesian Classifier

Let D be a training set of tuples and their associated class labels, and each tuple is represented by an n-D attribute vector X = (x_1, x_2, …,x_n). Suppose there are *m* classes C_1, C_2, …, C_m.Classification is to derive the maximum posteriori, i.e., the maximal P(C_i|X). This can be derived from Bayes' theorem

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

$$P(C_i|\mathbf{X}) = P(\mathbf{X}|C_i)P(C_i)$$

Since P(X) is constant for all classes, only needs to be maximized. A simplified assumption: attributes are conditionally independent (i.e., no dependence relation between attributes):

$$P(\mathbf{X}|C_i) = \prod_{k=1}^{n} P(x_k|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times ... \times P(x_n|C_i)$$

Example

For instance, to compute P(X/Ci), we consider the following:

age	income	student	credit_rating	buys_compu
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Figure 6: Data Set

Class:

C1:buys_computer = 'yes'

C2:buys_computer = 'no'

Data sample

X = (age <=30,Income = medium,Student = yes, Credit_rating = Fair).

$P(C_i)$: P(buys_computer = "yes") = 9/14 = 0.643

P(buys_computer = "no") = 5/14= 0.357

Compute $P(X|C_i)$ for each class

P(age = "<=30" | buys_computer = "yes") = 2/9 = 0.222

P(age = "<= 30" | buys_computer = "no") = 3/5 = 0.6

P(income = "medium" | buys_computer = "yes") = 4/9 =0.444

P(income = "medium" | buys_computer = "no") = 2/5 = 0.4

P(student = "yes" | buys_computer = "yes) = 6/9 = 0.667

P(student = "yes" | buys_computer = "no") = 1/5 = 0.2

P(credit_rating = "fair" | buys_computer = "yes") = 6/9 = 0.667

P(credit_rating = "fair" | buys_computer = "no") = 2/5 = 0.4

X = (age <= 30 , income = medium, student = yes, credit_rating = fair)

$P(X|C_i)$: $P(X|$buys_computer = "yes") = 0.222 x 0.444 x 0.667 x 0.667 = 0.044

$P(X|$buys_computer = "no") = 0.6 x 0.4 x 0.2 x 0.4 = 0.019

$P(X|C_i)*P(C_i)$:$P(X|$buys_computer = "yes") * P(buys_computer = "yes") = 0.028

$P(X|$buys_computer = "no") * P(buys_computer = "no") = 0.007

Therefore, X belongs to class ("buys_computer = yes")

Advantages

- It is simple and easy to implement
- It doesn't require as much training data
- It handles both continuous and discrete data

- It is highly scalable with the number of predictors and data points
- It is fast and can be used to make real-time predictions
- It is not sensitive to irrelevant features

Disadvantages

- Assumption: class conditional independence, therefore loss of accuracy
- Practically, dependencies exist among variables

  E.g., hospitals: patients: Profile: age, family history, etc.

  Symptoms: fever, cough, etc., Disease: lung cancer, diabetes, etc.
- Dependencies among these cannot be modeled by Naïve Bayesian Classifier.

How effective are Bayesian Classifiers?

In theory, Bayesian classifiers have the minimum error rate in comparison to all other classifiers. However, in practice, this is not always the case owing to inaccuracies in the assumptions made for its use, such as class conditional independence, and the lack of available probability data. However, various empirical studies of this classifier in comparison to the decision tree and neural network classifiers have found it to be comparable in some domains.

Bayesian classifiers are also useful in that they provide a theoretical justification for other classifiers which do not explicitly use Bayes theorem. For example, under certain assumptions, it can be shown that many neural network and curve fitting algorithms output the maximum posteriori hypothesis, as does the naive Bayesian classifier.

"Classification is a data mining technique used to predict group membership for data instances". Discuss.

11.7 Association based classification

Association rule mining finds interesting associations and relationships among large sets of data items. It shows how frequently anitemset occurs in a transaction. A typical example is a Market Based Analysis.An associative classifier (AC) is a type of supervised learning model that assigns a target value using association rules.

Market-Based Analysis is one of the key techniques used by large relations to show associations between items.It allows retailers to identify relationships between the items that people buy together frequently.

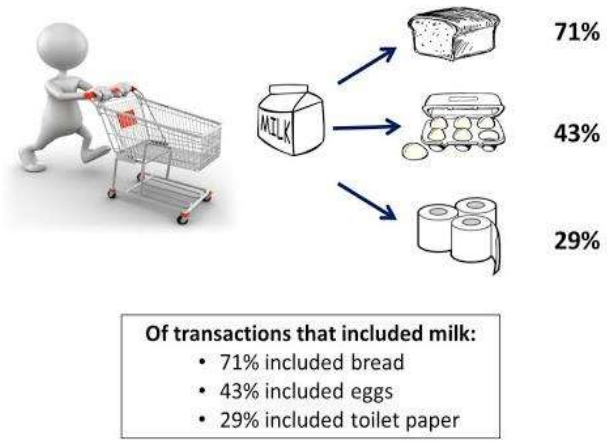


Figure 7: Association based classification

Given a set of transactions, we can find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction.

Table 1:Set of transactions

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

Before we start defining the rule, let us first see the basic definitions.

- **Support Count( )- σ** Frequency of occurrence of an itemset.Here σ ({Milk, Bread, Diaper})=2
- **Frequent Itemset –** An itemset whose support is greater than or equal to the minsup threshold.
- **Association Rule–** An implication expression of form X -> Y, where X and Y are any 2 itemsets.

{Milk, Diaper}->{Beer}

Support(s) –
The number of transactions that include items in the {X} and {Y} parts of the rule as a percentage of the total number of transactions.It is a measure of how frequently the collection of items occur together as a percentage of all transactions.

Support=σ(X+Y)/total

It is interpreted as a fraction of transactions that contain both X and Y.

Confidence(c) –
It is the ratio of the no of transactions that includes all items in {B} as well as the no of transactions that includes all items in {A} to the no of transactions that includes all items in {A}.

$$Conf(X=>Y) = Supp(X \cup Y) \div Supp(X)$$

It measures how often each item in Y appears in transactions that contain items in X also.

{Milk, Diaper}=>{Beer}

$$s = \sigma(\{Milk, Diaper, Beer\}) \div |T|$$
$$= 2/5$$
$$= 0.4$$

$$c = \sigma(Milk, Diaper, Beer) \div \sigma(Milk, Diaper)$$
$$= 2/3$$
$$= 0.67$$

11.8 Rule-Based Classifier

For classification, a rule-based classifier employs a collection of IF-THEN rules. From the following, we can express a rule:

IF condition THEN conclusion

Points to remember:

- The IF part of the rule is referred to as the rule antecedent or precondition.
- The rule's THEN part is referred to as rule consequent.
- The condition's antecedent part consists of one or more attribute checks that are logically ANDed.
- The consequent part consists ofclass prediction.

Application of Rule-Based Classifier

A rule *r* covers an instance **x** if the attributes of the instance satisfy the condition of the rule

R1: (Give Birth = no) ^ (Can Fly = yes) -> Birds

R2: (Give Birth = no) ^ (Live in Water = yes) -> Fishes

R3: (Give Birth = yes) ^ (Blood Type = warm) ->Mammals

R4: (Give Birth = no) ^ (Can Fly = no) -> Reptiles

R5: (Live in Water = sometimes) -> Amphibians

Table 2: Test Dataset

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
Hawk	warm	no	yes	no	?
grizzly bear	warm	yes	no	no	?

The rule R1 covers a hawk => Bird

The rule R3 covers the grizzly bear => Mammal

How does Rule-based Classifier Work?

R1: (Give Birth = no) ^Can Fly = yes) ->Birds

R2: (Give Birth = no) ^ (Live in Water = yes) -> Fishes

R3: (Give Birth = yes)^ (Blood Type = warm)->Mammals

R4: (Give Birth = no) ^ (Can Fly = no) ->Reptiles

R5: (Live in Water = sometimes) ->Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
lemur	warm	yes	no	no	?
turtle	cold	no	no	sometimes	?
dogfish shark	cold	yes	no	yes	?

A lemur triggers rule R3, so it is classified as a mammal A turtle triggers both R4 and R5

A dogfish shark triggers none of the rules

From Decision Trees To Rules

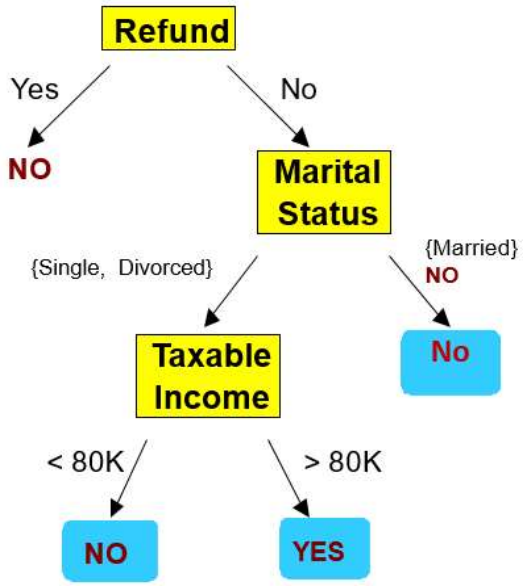


Figure 8: Decision Tree for Rule Generation

The Following are the classification rules generated from the given decision tree.

Classification Rules

(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced},Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single,Divorced},Taxable Income>80K) ==> Yes

(Refund=No, Marital Status={Married}) ==> No.

Explain how will you remove the training data covered by rule R.

11.9 K-Nearest Neighbour(KNN)

The KNN algorithm is a type of supervised machine learning algorithm that can be used to solve both classification and regression predictive problems. However, in industry, it is primarily used to solve classification and prediction problems. The following two characteristics would be a good way to describe KNN:

1. **Lazy learning algorithm**: KNN is a lazy learning algorithm since it doesn't have a dedicated training process and instead uses all of the data for training and classification.
2. **Non-parametric learning algorithm**: KNN is also a non-parametric learning algorithm since it makes no assumptions about the underlying data.

Working of KNN Algorithm

The KNN algorithm predicts the values of new datapoints using 'feature similarity,' which means that the new data point will be assigned a value based on how closely it matches the points in the training set. With the aid of the steps below, we can understand how it works.

Step1: We must load both training and test data in the first step of KNN.

Step2: The value of K, i.e. the closest data points, must then be chosen. Any integer can be used as K.

Step3: For each point in the test data do the following :

- Calculate the distance between each row of training data and the test data using one of the following methods: Euclidean, Manhattan, or Hamming distance. The Euclidean method is the most widely used method for calculating distance.
- Sort them in ascending order based on the distance value.
- The top K rows of the sorted array will then be chosen.
- The test point will now be assigned a class based on the most common class of these rows.

Example

Customer	Age	Income	No. credit cards	Response
John	**35**	**35K**	**3**	**No**
Rachel	**22**	**50K**	**2**	**Yes**
Hannah	**63**	**200K**	**1**	**No**
Tom	**59**	**170K**	**1**	**No**
Nellie	**25**	**40K**	**4**	**Yes**
David	**37**	**50K**	**2**	**?**

Figure 9: dataset for Findingthenearest neighbor

Table 3: Distance from each customer

Distance from David
John= sqrt $[(35-37)^2+(35-50)^2+(3-2)^2]=15.16$
Rachel= sqrt $[(22-37)^2+(50-50)^2+(2-2)^2]=15$
Hannah= sqrt $[(63-37)^2+(200-50)^2+(1-2)^2]=152.23$
Tom= sqrt $[(59-37)^2+(170-50)^2+(1-2)^2]=122$
Nellie= sqrt $[(25-37)^2+(40-50)^2+(4-2)^2]=15.74$

As per the distance calculation, nearest to David is Rachel so the predicted class for David is the same as Rachel.

Table 4: Predicted class for David

Customer	Age	Income (K)	No. cards	Response

John	35	35	3	No
Rachel	22	50	2	Yes
Hannah	63	200	1	No
Tom	59	170	1	No
Nellie	25	40	4	Yes
David	37	50	2	Yes

11.10 Decision Tree Classifier

A decision tree is a flow-chart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf nodes represent classes or class distributions. The topmost node in a tree is the root node. The following decision tree is for the concept of buy computer that indicates whether a customer at a company is likely to buy a computer or not.

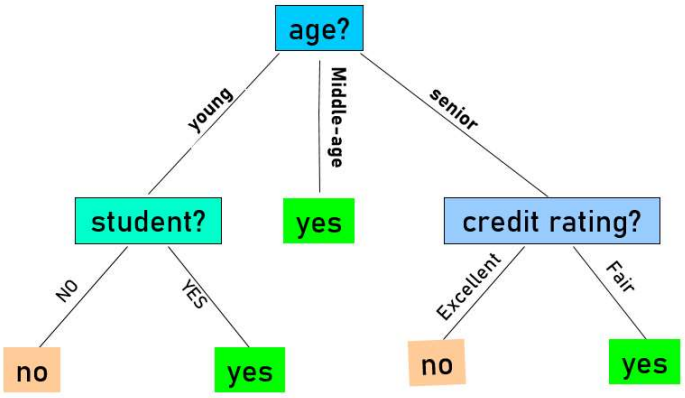


Figure 10: Decision Tree for the concept of buy computer

Decision Tree Induction

The automatic generation of decision rules from examples is known as rule induction or automatic rule induction.

Generating decision rules in the implicit form of a decision tree is also often called rule induction, but the terms tree induction or decision tree inductions are sometimes preferred.

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Figure 11: Dataset for buys_computer

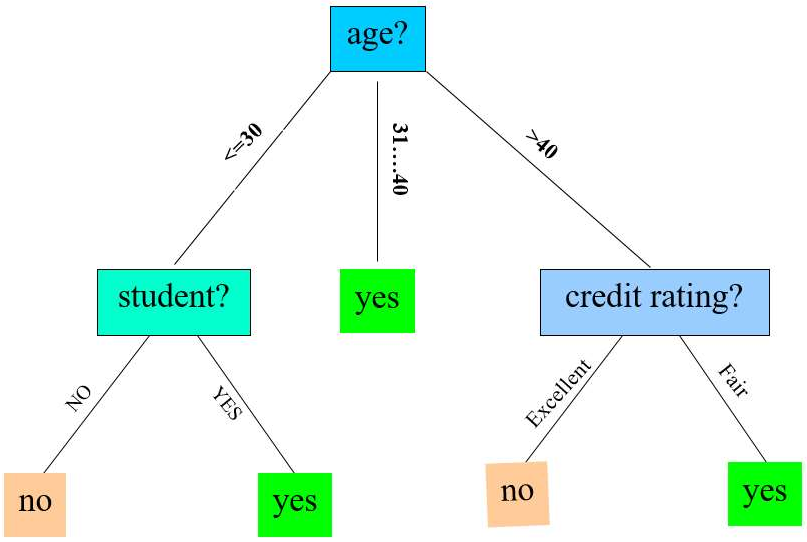


Figure 12: A Decision Tree for "buys_computer"

How are decision trees used for classification?

Given a tuple, X, for which the associated class label is unknown.The attribute values of the tuple are tested against the decision tree. A path is traced from the root to a leaf node, which holds the class prediction for that tuple. Decision trees can easily be converted to classification rules.

Why are decision tree classifiers so popular?

The construction of decision tree classifiers does not require any domain knowledge or parameter setting.Decision trees can handle multidimensional data. Their representation of acquired knowledge in tree form is intuitive and generally easy to understand by humans. The learning and classification steps of decision tree induction are simple and fast.

Attribute Selection Measure

The key problem that emerges when implementing a Decision tree is how to choose the best attribute for the root node and sub-nodes. So, there is a technique called Attribute Selection Measure, or ASM, that can be used to solve such problems. We can easily pick the best attribute for the tree's nodes using this measurement. There are two commonly used ASM techniques are:

- Information Gain

- Gini Index

Information Gain

The calculation of changes in entropy after segmenting a dataset based on an attribute is known as information gain. It determines how much data a function provides about a class. We split the node and built the decision tree based on the importance of information gain.The highest information gain node/attribute is split first in a decision tree algorithm, which always seeks to maximize the amount of information gain. The formula below can be used to measure it.

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

Where

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

In simple terms it can be represented as :

Information Gain= Entropy(S)- [(Weighted Avg) *Entropy(each feature)

Entropy: Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as:

```
Entropy(s)= -P(yes)log2 P(yes)- P(no) log2 P(no)
```

Where,

S= Total number of samples

P(yes)= probability of yes

P(no)= probability of no

The information gained by branching on attribute A is represented as:

$$Gain(A) = Info(D) - Info_A(D)$$

Gini Index

The Gini index is a metric of impurity or purity used in the CART (Classification and Regression Tree) algorithm to build a decision tree. In comparison to a high Gini index, an attribute with a low Gini index should be favored. It only produces binary splits, and the CART algorithm creates binary splits using the Gini index.The following formula can be used to measure the Gini index:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2,$$

Example

Table 5: Data for calculating attribute selection measures

RID	age	income	student	Credit_rating	Class:buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle-aged	high	no	fair	yes

4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle-aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle-aged	medium	no	excellent	yes
13	middle-aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

The class label attribute, buys_computer, has two distinct values (namely, {yes, no}); therefore, there are two distinct classes (i.e., m = 2).There are nine tuples of class, yes, and five tuples of class no.

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ bits.}$$

We need to look at the distribution of yes and no tuples for each category of age. For the age category "youth," there are two yes tuples and three no tuples. For the category "middle-aged," there are four yes tuples and zero no tuples. For the category "senior," there are three yes tuples and two no tuples.

$$Info_{age}(D) = \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) + \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4}\right) + \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right) = 0.694 \text{ bits.}$$

Hence, the gain in information from such partitioning would be:

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.694 = 0.246 \text{ bits.}$$

Similarly, we can compute Gain(income) = 0.029 bits, Gain(student) = 0.151 bits, and Gain(credit_rating) = 0.048 bits. Because age has the highest information gain among the attributes, it is selected as the splitting attribute. Node N is labeled with age.

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo_A(D)}.$$

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right).$$

A test on income splits the data into three partitions, namely low, medium, and high, containing four, six, and four tuples, respectively.

$$SplitInfo_{income}(D) = -\frac{4}{14} \times \log_2\left(\frac{4}{14}\right) - \frac{6}{14} \times \log_2\left(\frac{6}{14}\right) - \frac{4}{14} \times \log_2\left(\frac{4}{14}\right)$$

$$= 1.557.$$

Gain(Income) = 0.029.

Gain Ratio (Income) = 0.029 / 1.557 = 0.019.

The Gini index is used in CART. Using the notation previously described, the Gini index measures the impurity of D, a data partition or set of training tuples, as:

$$Gini(D) = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2 = 0.459.$$

Advantages of the Decision Tree

- It is easy to comprehend because it follows the same steps that a person will take when deciding in the real world.
- It can be extremely helpful in resolving decision-making issues.
- It is beneficial to consider all of the potential solutions to a problem.
- In comparison to other algorithms, data cleaning is not required as much.

Disadvantages of the Decision Tree

- The decision tree is complicated because it has several layers.
- It may have an overfitting problem, which the Random Forest algorithm may solve.
- The computational complexity of the decision tree may increase as more class labels are added.

A company is trying to decide whether to bid for a certain contract or not. They estimate that merely preparing the bid will cost £10,000. If their company bid then they estimate that there is a 50% chance that their bid will be put on the "short-list", otherwise their bid will be rejected.Once "short-listed" the company will have to supply further detailed information (entailing costs estimated at £5,000). After this stage, their bid will either be accepted or rejected.The company estimate that the labor and material costs associated with the contract are £127,000. They are considering three possible bid prices, namely £155,000, £170,000, and £190,000. Theyestimate that the probability of these bids being accepted (once they have been short-listed) is 0.90, 0.75, and 0.35 respectively. What should the company do and what is the expected monetary value of your suggested course of action?

11.11 Random Forest

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

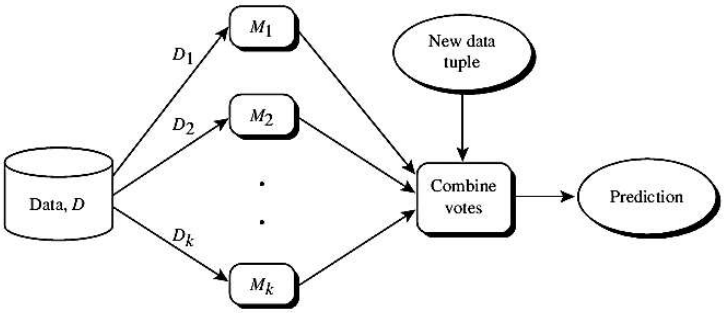


Figure 13: Ensemble Learning

Random Forest is a classifier that contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and predicts the final output.The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Working of Random Forest

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

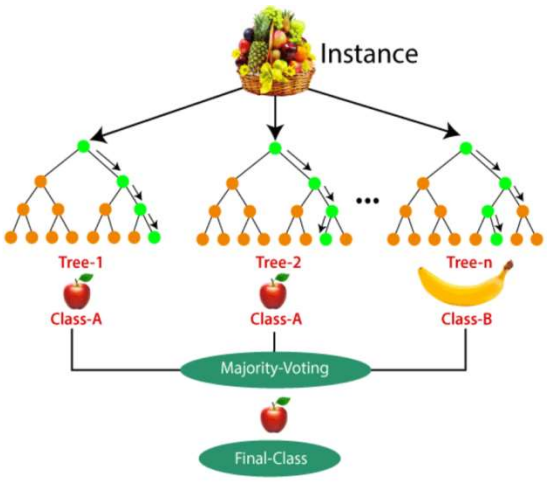


Figure 14:Random Forest

Advantages of Random Forest

- Random Forest is capable of performing both Classification and Regression tasks.
- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue.

Disadvantages of Random Forest

- Complexity is the main disadvantage of Random forest algorithms.
- Construction of Random forests is much harder and time-consuming than decision trees.
- More computational resources are required to implement the Random Forest algorithm.

11.12 Multi-category Classification

The problem of classifying instances into one of three or more classes is known as multiclass or multinomial classification.

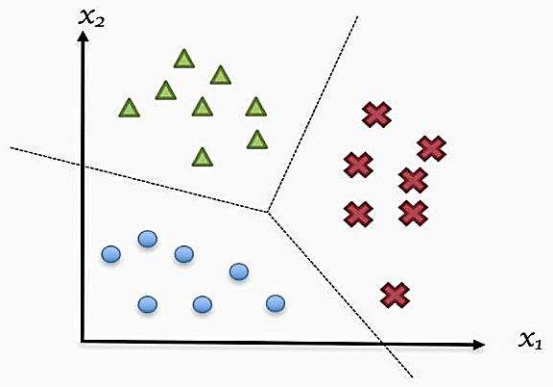


Figure 15: Multi-Category Classification

As a model prediction, multi-class classification is a classification technique that allows us to categorize test data into multiple class labels found in trained data. Multi-class classification methods can be divided into two categories:

- **One-versus-all**
- **One-versus-one**

One-verses-all

The one-versus-all method is usually implemented using a "Winner-Takes-All" (WTA) strategy. It constructs M binary classifier models where M is the number of classes. The i^{th} binary classifier is trained with all the examples from i^{th} class Wi with positive labels (typically +1), and the examples from all other classes with negative labels (typically -1).

One-verses-one

The one-versus-one method is usually implemented using a "Max-Wins" voting (MWV) strategy. This method constructs one binary classifier for every pair of distinct classes and so, all together it constructs M(M - 1)/2 binary classifiers. The binary classifier Cij is trained with examples from ith class Wi and jth class Wj only, where examples from class Wi take positive labels while examples from class Wj take negative labels.

Some researchers also proposed "all-together" approaches that solve the multi-category classification problem in one step by considering all the examples from all classes together at once. However, the training speed of "all-together" methods is usually slow.

11.13 Cross-Validation

Cross-validation is a technique in which we train our model using the subset of the data-set and then evaluate using the complementary subset of the data-set.The following are the three steps involved in cross-validation:

1. A part of the sample data set should be set aside.
2. Train the model with the rest of the data.
3. Use the data-reserve set portion to test the model.

Methods of Cross-Validation

Validation

In this process, 50% of the data set is used for preparation, and the other 50% is used for testing. The biggest disadvantage of this approach is that we only train on 50% of the dataset; the remaining 50% of the data likely includes crucial details that we're leaving when training our model, resulting in higher bias.

LOOCV (Leave One Out Cross Validation)

This approach iterates for each datapoint by training on the entire data-set but leaving just one data-point from the available data-set. It has some benefits as well as some drawbacks.

Advantage

This technique has the advantage of making use of all data points, resulting in low bias.

Disadvantages

Since we are measuring against a single data point, this approach has the main disadvantage of causing more variance in the testing model. If a data point is an outlier, the variance would be higher. Another disadvantage is that it takes a long time to execute because it iterates over the number of data points.

K-Fold Cross-Validation

This approach divides the data set into k subsets (also known as folds), then trains all of the subsets while leaving one (k-1) subset for evaluation of the trained model. We iterate k times with a different subset reserved for testing purposes each time in this process.

The value of k should always be 10 since a lower value of k leads to validation and a higher value of k leads to the LOOCV method.

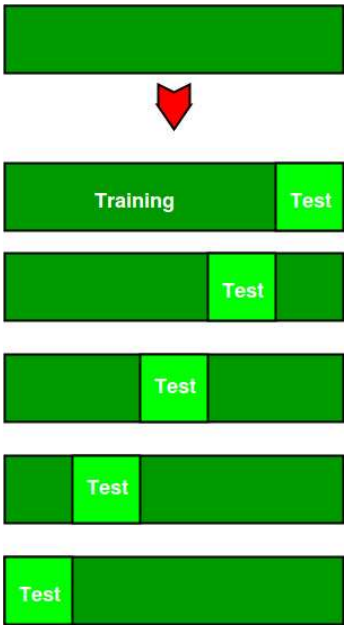


Figure 16: K-Fold cross-validation with K=5

Advantages of a train/test split include:

- Since K-fold cross-validation repeats the train/test split K times, it is K times faster than Leave One Out cross-validation.
- Examining the detailed outcomes of the assessment procedure is easier.

Advantages of Cross-Validation

- Out-of-sample precision can now be estimated more accurately.
- Every observation is used for both training and testing, resulting in a more "effective" use of data.

11.14 Overfitting

When we train a statistical model with a large amount of data (much like fitting ourselves into oversized pants!), it is said to be overfitted. When a model is trained with a large amount of data, it begins to learn from the noise and inaccuracies in the data collection. The model then fails to correctly categorize the data due to too many details and noise.Non-parametric and non-linear

approaches are the causes of overfitting since these types of machine learning algorithms have more flexibility in constructing models based on the dataset and can thus create unrealistic models. If we have linear data, we can use a linear algorithm to avoid overfitting, or we can use decision tree parameters like the maximal depth to avoid overfitting.In a nutshell, overfitting is characterized by a high variance and a low bias.

Bias – A model's assumptions that make a function easier to understand.

Variance: Variance is when you train your data on training data and get a very low error, but when you change the data and then train the same previous model again, you get a very high error.

Summary

- Bayesian classifiers are statistical classifiers. They can predict class membership probabilities, such as the probability that a given sample belongs to a particular class.
- Bayesian classification is based on Bayes theorem. Bayesian classifiers exhibited high accuracy and speed when applied to large databases.
- Classification and prediction methods can be compared and evaluated according to the ⌘⌘criteria of Predictive accuracy, Speed, Robustness, Scalability and Interpretability.
- A decision tree⌘⌘is a flow-chart-like tree structure, where each *internal node* denotes a test on an attribute, each *branch* represents an outcome of the test, and *leaf nodes* represent classes or class distributions. The topmost node in a tree is the *root* node.
- The learning of the model is 'supervised' if it is told to which class each training sample belongs. In contrasts with unsupervised learning (or clustering), in which the class labels of the training samples are not known, and the number or set of classes to be learned may not be known in advance.
- Naïve Bayes classifiers assume that the effect of a variable value on a given class is ⌘⌘independent of the values of other variable. This assumption is called class conditional independence. It is made to simplify the computation and in this sense considered to be "Naïve".
- Non-parametric and non-linear approaches are the causes of overfitting since these types of machine learning algorithms have more flexibility in constructing models based on the dataset and can thus create unrealistic models.
- Cross-validation is a technique in which we train our model using the subset of the data-set and then evaluate using the complementary subset of the data-set.

Keywords

Bayes theorem: Bayesian classification is based on Bayes theorem.

Bayesian belief networks: These are graphical models, which unlike naive Bayesian classifiers, allow the representation of dependencies among subsets of attributes

Bayesian classification: Bayesian classifiers are statistical classifiers.

Classification: Classification is a data mining technique used to predict group membership for data instances.

Decision Tree: A decision tree is a flow-chart-like tree structure, where each *internal node* denotes a test on an attribute, each *branch* represents an outcome of the test, and *leaf nodes* represent classes or class distributions. The topmost node in a tree is the *root* node.

Decision tree induction: The automatic generation of decision rules from examples is known as *rule induction* or *automatic rule induction*.

Overfitting: Decision trees that are too large are susceptible to a phenomenon called as overfitting.

Prediction: Prediction is similar to classification, except that for prediction, the results lie in the future.

Supervised learning: The learning of the model is 'supervised' if it is told to which class each training sample belongs.

Self Assessment

1. Bayesian classifiers are ..... classifiers.

2. Bayesian classification is based on ..... theorem.

3. The learning of the model is ..... if it is told to which class each training sample belongs.

4. A ..... is a flow-chart-like tree structure.

5. The ..... measure is used to select the test attribute at each node in the tree.

6.Which of the following statement is true about the classification?

A. It is a measure of accuracy

B. It is a subdivision of a set

C. It is the task of assigning a classification

D. None of the above

7.Data Mining System Classification consists of?

A. Database Technology

B. Machine Learning

C. Information Science

D. All of the above

8.A rule-based system consists of a bunch of IF-THEN rules.

A. True

B. False

C. Can't Say

D. May be

9. Instead of representing knowledge in a relatively declarative, static way rule-based system represent knowledge in terms of_____ that tell you what you should do or what you could conclude in different situations.

A. Raw Text

B. A bunch of rules

C. Summarized Text

D. Collection of various Texts

10.Which of the following distance metric can not be used in k-NN?

A. Manhattan

B. Minkowski

C. Tanimoto

D. All can be used

11.Which of the following option is true about k-NN algorithm?

A. It can be used for classification

B. It can be used for regression

C. It can be used in both classification and regression

D. None

12.Which of the following will be true about k in k-NN in terms of Bias?

A. When you increase the k the bias will be increases

B. When you decrease the k the bias will be increases

C. Can't say

D. None of these

13.A _____ is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility.

A. Decision tree

B. Graphs

C. Trees

D. Neural Networks

14. What is Decision Tree?

A. Flow-Chart

B. Structure in which internal node represents test on an attribute, each branch represents outcome of test and each leaf node represents class label

C. Flow-Chart & Structure in which internal node represents test on an attribute, each branch represents outcome of test and each leaf node represents class label

D. None of the mentioned

15. Decision Nodes are represented by _____

A. Disks

B. Squares

C. Circles

D. Triangles

Review Questions

Q1. What do you mean by classification in data mining? Write down the applications of classification in business.

Q2. Discuss the issues regarding the classification.

Q3.What is a decision tree? Explain with the help of a suitable example.

Q4. Write down the basic algorithm for decision learning trees.

Q5. Write short notes on the followings:

(a) Bayesian classification

(b) Bayes theorem

(c) Naive Bayesian classification

Q6. Elucidate the usage of cross validation along with its various methods.

Q7. With example explain the KNN algorithm.

Answers for Self Assessment

1.	Statistical	2.	Bayes	3.	Supervised	4.	Decision Tree	5.	Information gain
6.	B	7.	D	8.	A	9.	B	10.	D
11.	C	12.	A	13.	A	14.	C	15.	B

Further Reading

A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.

Alex Berson, *Data Warehousing Data Mining and OLAP*, Tata Mcgraw Hill, 1997

Alex Berson, Stephen J. Smith, *Data warehousing, Data Mining & OLAP*, Tata McGraw Hill, Publications, 2004.

Alex Freitas and Simon Lavington, *Mining Very Large Databases with Parallel Processing*, Kluwer Academic Publishers, 1998.

J. Ross Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, 1993.

Jiawei Han, MichelineKamber, Data Mining – Concepts and Techniques, Morgan Kaufmann Publishers, First Edition, 2003.

Matthias Jarke, Maurizio Lenzerini, YannisVassiliou, PanosVassiliadis, Fundamentals of Data Warehouses, Publisher: Springer

Michael Berry and Gordon Linoff, Data Mining Techniques (For Marketing, Sales, and Customer Support), John Wiley & Sons, 1997.

Michael J. A. Berry, Gordon S Linoff, Data Mining Techniques, Wiley Publishing Inc, Second Edition, 2004.

https://www.geeksforgeeks.org/basic-concept-classification-data-mining/

https://www.tutorialspoint.com/data_mining/dm_classification_prediction.htm

https://www.javatpoint.com/data-mining-techniques

Unit 12: Prediction and Classification Using Weka Tool

CONTENTS

Objective

After this lecture, you will be able to

- How to use Weka to create a decision tree model.
- Understand the creation and working of Naïve Bayes in weka.
- Know the various applications of web mining.

Introduction

Weka is a set of data mining-related machine learning techniques. It includes data preparation, categorization, regression, clustering, mining of association rules, and visualization tools. It provides access to a vast variety of classification algorithms. One of the advantages of using the Weka platform to solve your machine learning challenges is the wide variety of machine learning algorithms available. Weka comes with many built-in functions for constructing a variety of machine learning techniques, ranging from linear regression to neural networks. With just a press of a button, you can install even the most complex algorithms on your dataset! Not only that, but Weka also allows you to use Python and R to access some of the most popular machine learning library algorithms.

12.1 Naïve Bayes Using Weka

The Bayes' Theorem is used to create the Naive Bayes classifiers, which are a series of classification algorithms based on the Bayes' Theorem. It's a group of algorithms that all have the same premise: each pair of features to be classified is independent of the others.This experiment's "weather-nominal" data set is available in ARFF format.

Steps to be followed

1. Using the choose file option, we must first load the necessary dataset into the Weka tool. We're going to run the weather-nominal dataset here.

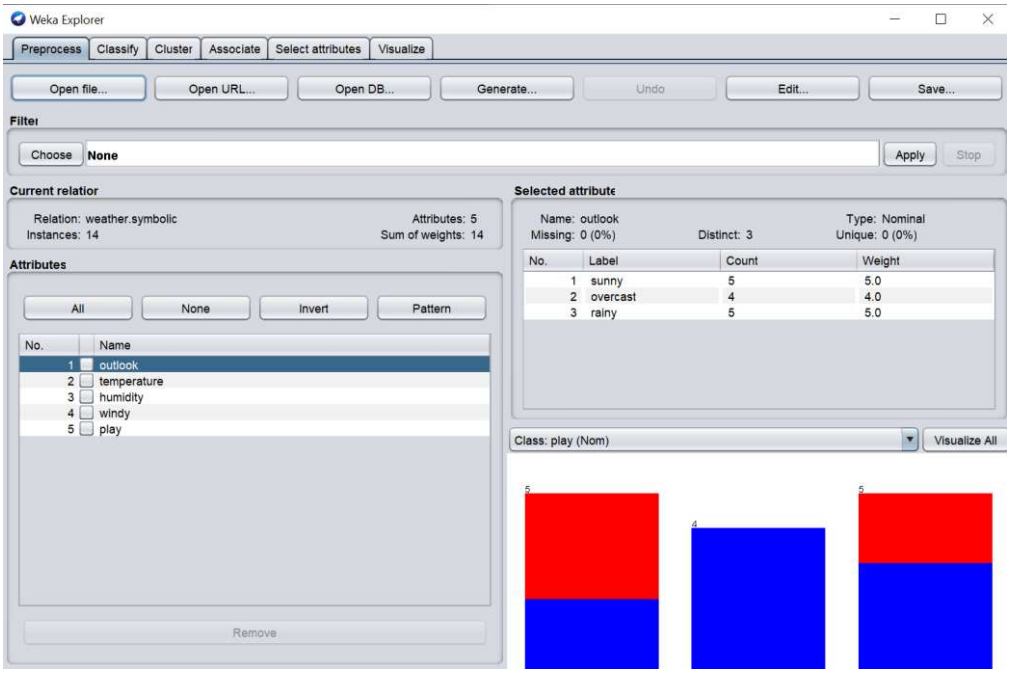


Figure 1: Dataset Selection

2. Now, on the top left side, go to the classify tab, click the choose button, and pick the Naive Bayesian algorithm.

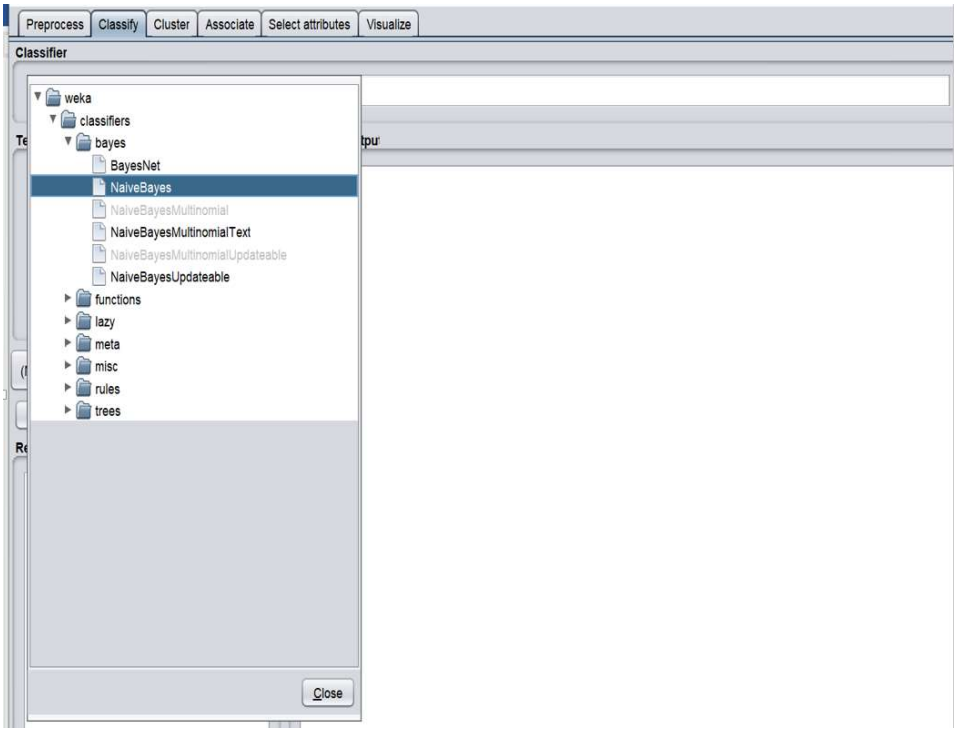


Figure 2: Selection of Naive Bayes

3. To adjust the parameters, click the choose button on the right side, and in this example, we'll accept the default values.

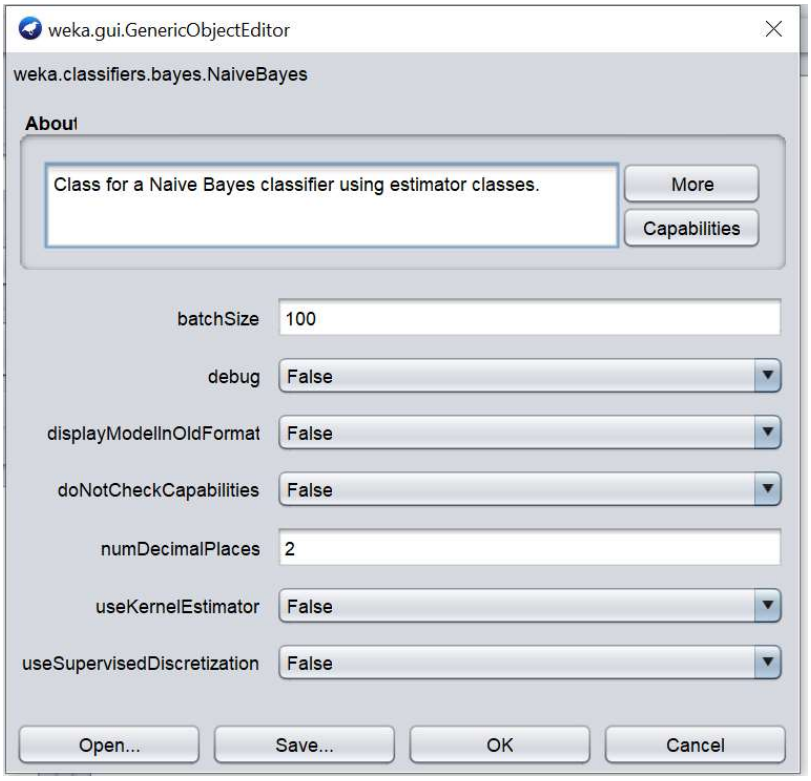


Figure 3: Adjustment of parameters

4. From the "Test" options in the main panel, we choose Percentage split as our measurement process. We'll use the percentage split of 66 percent to get a clear estimate of the model's accuracy since we don't have a separate test data set. There are 14 examples in our dataset, with 9 being used for training and 5 for testing.

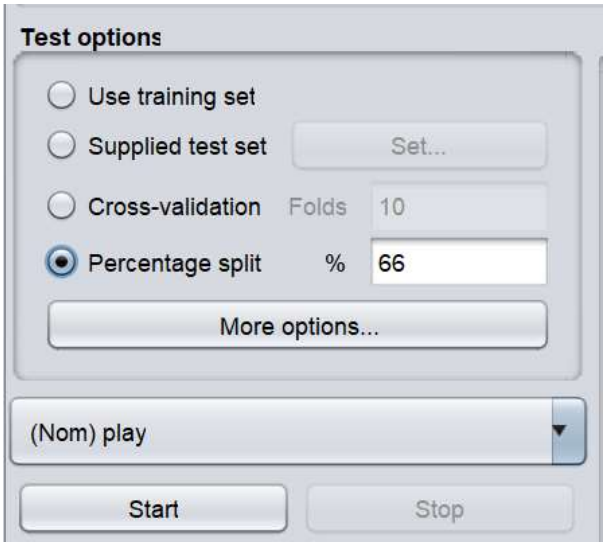


Figure 4: Selection of Split Ratio

5. We'll now click "start" to begin creating the model. The assessment statistic will appear in the right panel after the model has been completed.

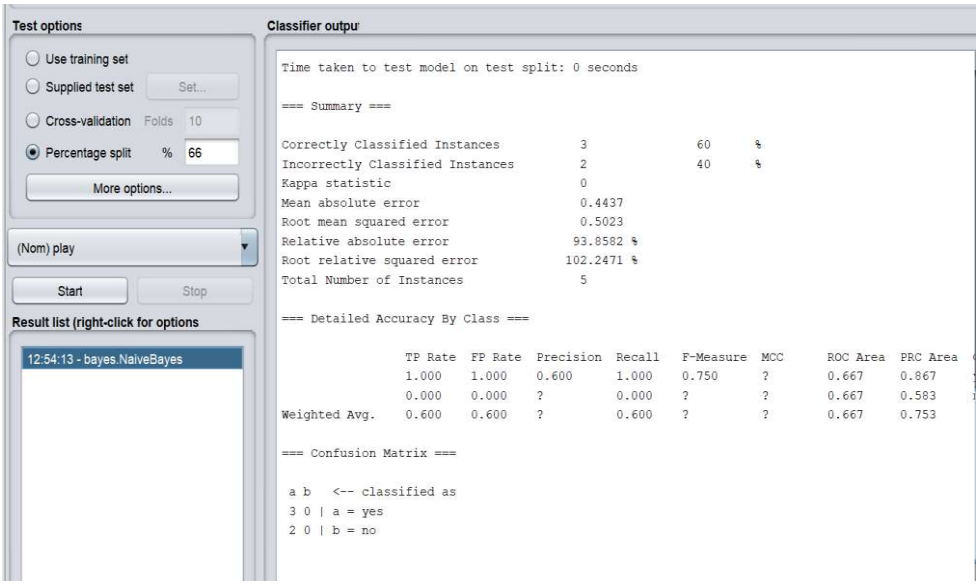


Figure 5: Model Output

6. It's worth noting that the classification accuracy of the model is about 60%. This means that by making certain changes, we would be able to improve the accuracy. (Either during preprocessing or when choosing current classification parameters.)

Furthermore, we can use our models to find new instances. In the main panel's "Test options," select the "supplied test package" radio button, then click the "set" button.

Figure 6: Selection of supplied Test set

This will open a pop-up window that will allow us to open the test instance file. It can be used to further increase the accuracy of the module by using different test sets.

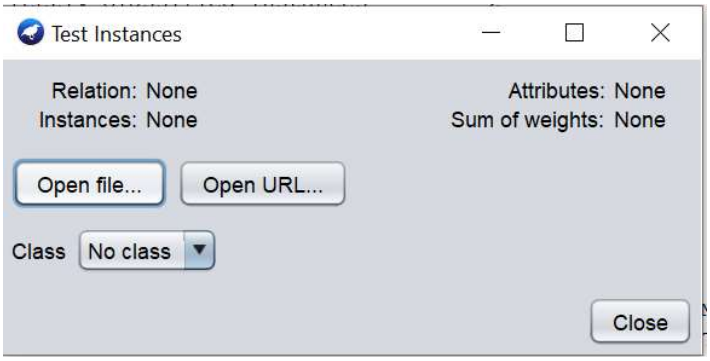


Figure 7: Selection of test Instances

Using the airline.arff inbuilt file from Weka dataset and implement Naïve Bayes algorithm on it.

12.2 Classification using Decision Tree in Weka

Classifier J48

It is a C4.5-generated algorithm for generating a decision tree (an extension of ID3). A statistical classifier is another name for it. A database is needed for decision tree classification.

Certain algorithms are greyed out depending on whether the problem is a classification or regression problem (piecewise linear function M5P for regression). Besides that, certain decision trees have distinct Weka implementations (for example, J48 implements C4.5 in Java)

In Weka, creating a decision tree is relatively easy. Simply follow the steps below:

1. Open WEKA explorer.
2. Choose the weather.nominal.arff file from the preprocess tab's "choose file" option.

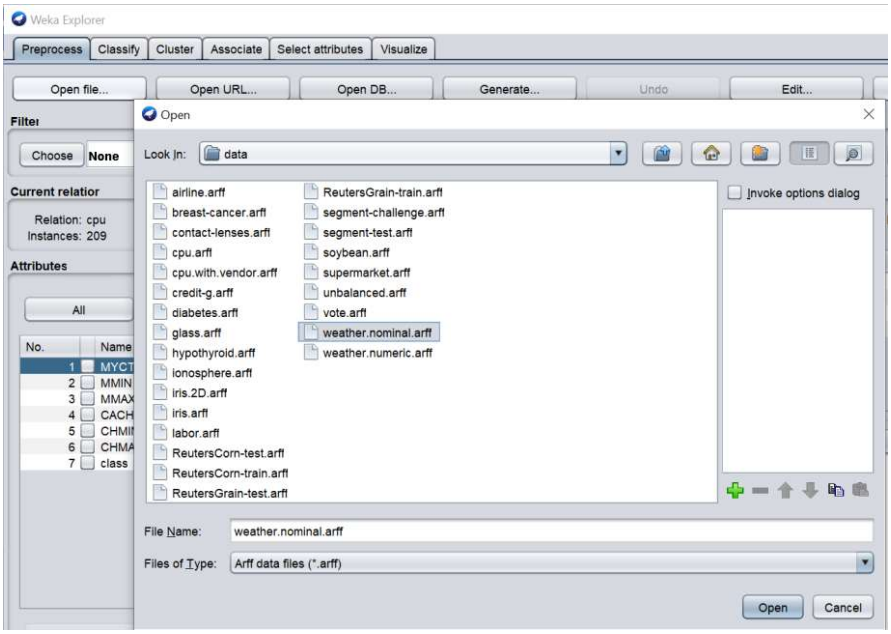


Figure 8: Selection of Dataset

3. To classify the unclassified data, go to the "Classify" tab. Select "Choose" from the drop-down menu. Select "trees -> J48" from this menu.

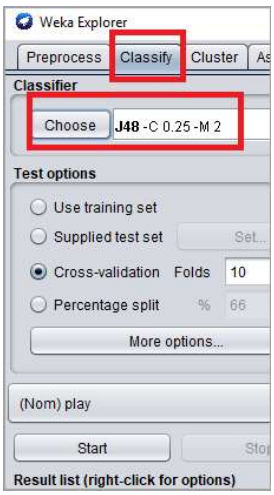


Figure 9: Selection of Decision tree(J48)

4. To begin, press the Start Button. The performance of the classifier will be visible on the right-hand panel. The run information is shown in the panel as follows:

Scheme: The classification algorithm that was used is referred to as the scheme.

Instances: Number of data rows in the dataset.

Attributes: There are five attributes in the dataset.

The decision tree is defined by the number of leaves and the size of the tree.

Time taken to build the model: Time for the output.
Full classification of the J48 pruned with the attributes and number of instances.

"A decision tree divides nodes based on all available variables and then chooses the split that produces the most homogeneous sub-nodes." The homogeneity of a sample at a split is calculated using Information Gain.

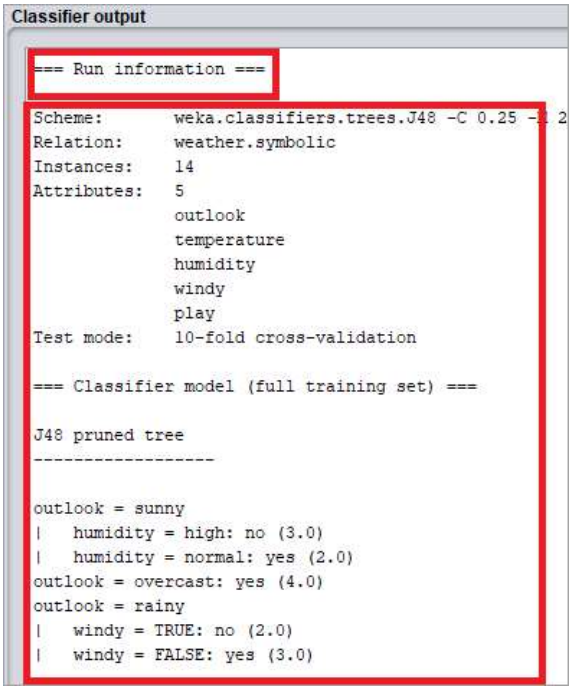


Figure 10: J48 output for the selected dataset

Right-click on the result and choose to visualize the tree from the list.

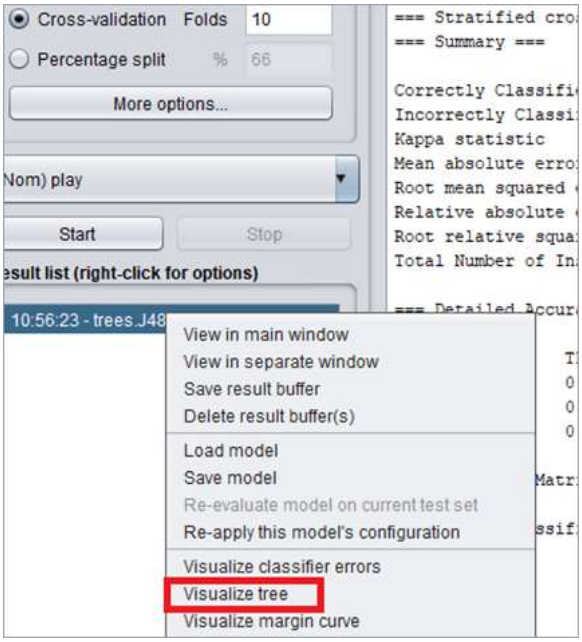


Figure 11: Selection of visualizing tree

5. The output is in the form of a decision tree. The main attribute is "outlook".

If the outlook is sunny, then the tree further analyzes the humidity. If humidity is high then the class label play= "yes".
If the outlook is overcast, the class label, play is "yes". The number of instances which obey the classification is 4.
If the outlook is rainy, further classification takes place to analyze the attribute "windy". If windy=true, the play = "no". The number of instances which obey the classification for outlook= windy and windy=true is 2.

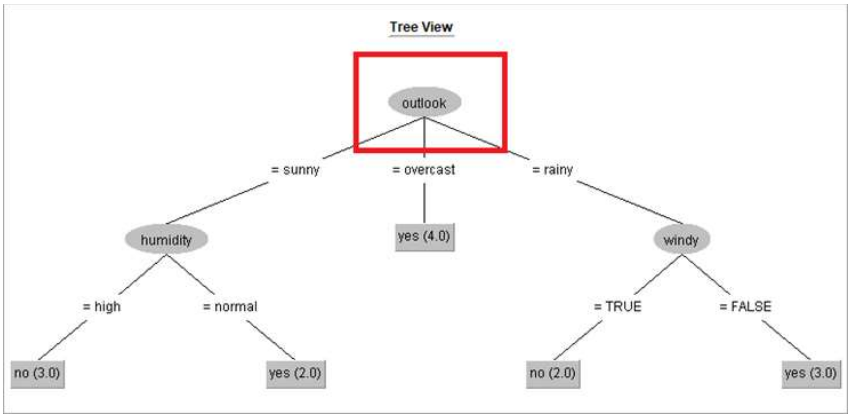


Figure 12: Tree view of J48

How would this instance be classified using the decision tree? outlook = sunny, temperature = cool, humidity = high, windy = TRUE.

12.3 Applications of classification for web mining

Web mining is the process of extracting knowledge from web data, such as web content, web structure, and web usage data, using data mining techniques. Web mining's major goal is to extract relevant information from the World Wide Web and its user trends. Web mining is classified into three categories: web content mining, web structure mining, and web usage mining. These are explained in the following sections.

Web Content Mining

The application of obtaining relevant information from the content of web documents is known as web content mining. Text, image, audio, video, and other sorts of data form web content.

A web page's content data is a collection of information. It has the potential to provide useful and interesting patterns about user requirements. Text mining, machine learning, and natural language processing are all connected to text documents. Text mining is another name for this type of mining. According to the content of the input, this sort of mining scans and mines text, images, and groups of web pages.It can be further classified into:

- **Web page content mining:**The typical search of a web page by content is known as content mining.
- **Search result mining:**The term "search result mining" refers to the second search of sites found in a prior search.

In web content mining, there are two approaches:

1. Agent-based approach
2. Database approach

In the Agent-based approach, there are three types of agents

● Intelligent search agents

● Information filtering/Categorizing agent

● Personalized web agents.

Intelligent search agents: Intelligent Search agents automatically searches for information according to a particular query using domain characteristics and user profiles.

Information filtering/Categorizing agent: To filter data according to predefined instructions, information agents used a variety of techniques.

Personalized web agents: Personalized web agents learn about user preferences and find documents that are relevant to those profiles.

It comprises a well-formed database comprising schemas and attributes with defined domains in the Database approach.

If a user wants to search for a particular book, then the search engine provides a list of suggestions.

Web structure mining

The application of discovering structural information from the web is known as web structure mining. The web graph's structure is made up of nodes (web pages) and edges (links) that connect related pages. Structure mining essentially displays a structured summary of a webpage. It establishes a link between web pages that are linked by information or a direct link.To determine the connection between two commercial websites, Web structure mining can be very useful.

Web structure mining, one of three types of data mining, is a technique for determining the relationship between Web pages linked by information or direct links. It provides details on how various pages are connected to construct this massive web.

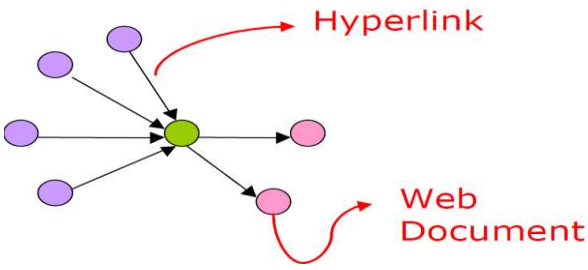


Figure 13: Connection between webpages

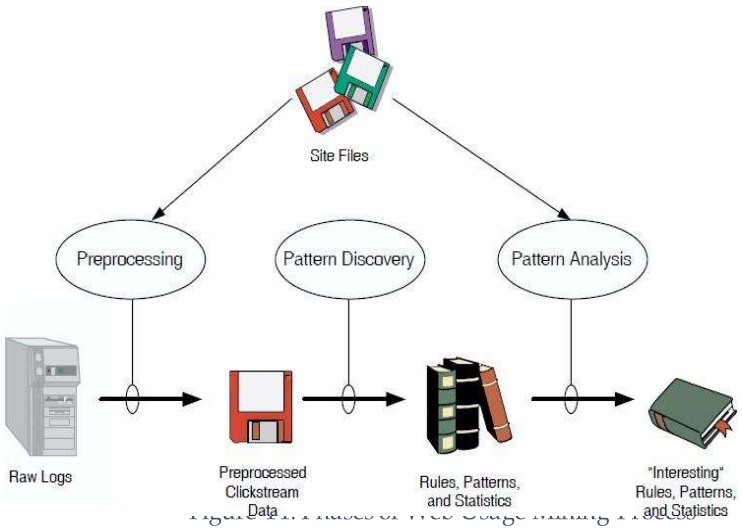
Web pages serve as nodes in a typical Web graph, while hyperlinks serve as edges linking two related pages.Web Structural Mining is the method of extracting structure data from the internet. This type of mining can be done at the document (intra-page) level or the hyperlink (inter-page) level. Hyperlink Analysis is a term used to describe study at the level of hyperlinks.

Web structure mining can be very useful to companies to determine theconnection between two commercial websites.

Web Usage Mining

Web usage mining, a subset of data mining, is the extraction of various types of interesting data that is readily available and accessible on the Internet—or formally known as the World Wide Web—in the ocean of massive web pages (WWW). It has aided in the analysis of user actions on various web pages and tracking them over time as one of the uses of data mining methodology. Web usage mining may reveal relationships that were not intended by the pages' creators.

Some of the methods to identify and analyze the web usage patterns are given below:



Preprocessing: Preprocessing entails translating the following information from diverse data sources:

- usage information
- content information
- structure information

into the data abstractions required for pattern identification from the many available data sources.

Pattern Discovery: Pattern recognition uses methods and algorithms from a variety of domains, including statistics, data mining, machine learning, and pattern recognition.

Pattern Analysis: Pattern analysis is the final step in the whole Web Usage mining process. Its goal is to eliminate uninteresting rules or patterns from the set discovered during the pattern discovery phase. The application for which Web mining is done usually determines the exact analytic methodology.The most common form of pattern analysis consists of:

- A knowledge query technique, such as SQL, is the most prevalent kind of pattern analysis.
- Another method is to input user data into a data cube and use it for Online Analytical Processing (OLAP).
- Visualization techniques like graphing patterns or assigning colors to distinct values can often reveal broad patterns or trends in data.
- Patterns having pages of a specific usage kind, content type, or pages that fit a specific hyperlink structure can be filtered out using content and structure information.

Challenges in Web Mining

Based on the following observations, the web pretends to present incredible challenges for resource discovery and knowledge discovery:

The complexity of Web Pages

There is no unified structure to the web pages. When compared to regular text documents, they are exceedingly complex. In the web's digital library, there are tremendous volumes of documents. These libraries are not arranged in any particular sequence.

The web is a Dynamic Data Source

The information on the internet is constantly updated. News, weather, shopping, financial news, sports, and so on are only a few examples.

Client networks are diverse

The web's client network is rapidly growing. These customers have a variety of hobbies, backgrounds, and purposes for using the service. There are over a hundred million internet-connected workstations, and this number is rapidly growing.

Relevance of Data

It is assumed that a specific person is only concerned with a limited area of the internet, while the rest of the web contains data that is unfamiliar to the user and may lead to unexpected results.

Summary

- Naive Bayes is a classification algorithm. Traditionally it assumes that the input values are nominal, although numerical inputs are supported by assuming a distribution.
- Naïve Bayes classifier is a statistical classifier. It assumes that the values of attributes in the classes are independent.
- J48 can deal with both nominal and numeric attributes
- Decision trees are also known as Classification And Regression Trees (CART).
- Each node in the tree represents a question derived from the features present in your dataset.
- Weka is free open-source software that comes with several machine learning algorithms that can be accessed via a graphical user interface.
- Pattern analysis is the final step in the whole Web Usage mining process.
- Naïve Bayes classifiers assume that the effect of a variable value on a given class is independent of the values of other variables. This assumption is called class conditional independence. It is made to simplify the computation and in this sense considered to be "Naïve".

Keywords

Bayesian classification: Bayesian classifiers are statistical classifiers.

Classification: Classification is a data mining technique used to predict group membership for data instances.

Decision Tree: A decision tree is a flow-chart-like tree structure, where each *internal node* denotes a test on an attribute, each *branch* represents an outcome of the test, and *leaf nodes* represent classes or class distributions. The topmost node in a tree is the *root* node.

Predictive accuracy: This refers to the ability of the model to correctly predict the class label of new or previously unseen data.

***Intelligent search agents*:**Intelligent Search agents automatically searches for information according to a particular query using domain characteristics and user profiles.

***Web page content mining*:**The typical search of a web page by content is known as content mining.

Instances: Number of data rows in the dataset.

Self Assessment Questions

1. In WEKA explorer _____ tab we need to open data file.

A. Preprocess

B. Cluster

C. Select Attribute

D. Classify

2. Which of the following tab is used to build a Naive Bayes model.

A. Preprocess

B. Cluster

C. Select Attribute

D. Classify

3. To build a classifier in Weka we need to click on

A. Build Button

B. Start Button

C. Create Button

D. None

4. Under classify you need to select_____ button, and from the tree menu, select Naïve Bayes.

A. Choose Classifier

B. Select Classifier

C. Create Classifier

D. Build Classifier

5. In Weka which of the following options is/are available under test options.

A. Use Training Set

B. Supplied Test Set

C. Cross-Validation

D. All of the above.

6. _____ work by learning answers to a hierarchy of if/else questions leading to a decision.

A. Decision trees

B. KNN

C. Naïve Bayes

D. Perceptron

7. Which of the following steps are used to load the dataset in Weka.

A. Open Weka GUI

B. Select the "Explorer" option.

C. Select "Open file" and choose your dataset.

D. All of the above

8. Implementing a decision tree in Wekafrom the drop-down list, select _____which will open all the tree algorithms.

A. Tree

B. Trees

C. Decision Tree

D. ALL Tree

9. In decision tree _____ is used to calculate the homogeneity of the sample at a split.

A. Information Gain

B. No of attributes

C. Entropy

D. Kappa Statistics

10. Decision tree splits the nodes on all available variables and then selects the split which results in the most _____sub-nodes.

A. Homogeneous

B. Heterogeneous

C. Both

D. None

11. For what purpose, the analysis tools pre-compute the summaries of the huge amount of data?

A. To maintain consistency

B. For authentication

C. For data access

D. To obtain the queries response

12. Which of the following statement is true about the classification?

A. It is a measure of accuracy

B. It is a subdivision of a set

C. It is the task of assigning a classification

D. None of the above

13. Task of inferring a model from training labeled data is called

A. Unsupervised Learning

B. Supervised Learning

C. Reinforcement Learning

D. All of the above

14. _____are used in data mining to classify data based on class labels.

A. Classification algorithms

B. Clustering Algorithms

C. Both

D. None

15. _____is the process of Data Mining techniques to automatically discover and extract information from Web documents and services.

A. Text Mining

B. Data Mining

C. Web Mining

D. None

Answers for Self Assessment

1.	A	2.	D	3.	B	4.	A	5.	D
6.	A	7.	D	8.	B	9.	A	10.	A
11.	D	12.	B	13.	B	14.	A	15.	C

Review Questions

Q1. Create a dataset and apply a decision tree algorithm on it and display the results in the tree form.

Q2. Consider training and predicting with a naive Bayes classifier for two document classes. The word "booyah" appears once for class 1, and never for class 0. When predicting new data, if the classifier sees "booyah", what is the posterior probability of class 1?

Q3. Elucidate the concept of web mining along with its various categories.

Q4. With the help of example explain the various challenges of Web mining.

Q5. Consider the weather.arff file and discuss why does it make more sense to test the feature "outlook" first? Implement the dataset using decision tree.

Further Reading

Thuraisingham, B., & Maning, D. (1999). *technologies, techniques, tools, and Trends*. CRC press.

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2005). Practical machine learning tools and techniques. *Morgan Kaufmann*, 578.

Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

Ngo, T. (2011). Data mining: practical machine learning tools and technique, by ian h. witten, eibe frank, mark a. hell. *ACM Sigsoft Software Engineering Notes*, *36*(5), 51-52.

Kaluža, B. (2013). *Instant Weka How-to*. Packt Publishing Ltd.

Veart, D. (2013). *First, Catch Your Weka: A Story of New Zealand Cooking*. Auckland University Press.

https://machinelearningmastery.com/use-classification-machine-learning-algorithms-weka/

https://scienceprog.com/building-and-evaluating-naive-bayes-classifier-with-weka/

https://weka.sourceforge.io/doc.dev/weka/classifiers/bayes/package-summary.html

https://www.analyticsvidhya.com/blog/2020/03/decision-tree-weka-no-coding/

https://weka.sourceforge.io/doc.dev/weka/classifiers/trees/J48.html

Unit 13: Clustering methods using Weka Tool

CONTENTS

Objectives

After this unit, you will be able to

- Understand the concept of unsupervised learning.
- Learn the various clustering algorithms.
- Know the difference between clustering and classification.
- Implementation of K-Means and Hierarchal clustering using Weka.

Introduction

Clustering is the organization of data in classes. However, unlike classification, it is used to place data elements into related groups without advanced knowledge of the group definitions i.e. class labels are unknown and it is up to the clustering algorithm to discover acceptable classes. Clustering is also called unsupervised classification because the classification is not dictated by given class labels. There are many clustering approaches all based on the principle of maximizing the similarity between objects in the same class (intra-class similarity) and minimizing the similarity between objects of different classes (inter-class similarity). Clustering can also facilitate taxonomy formation, that is, the organization of observations into a hierarchy of classes that group similar events together.

Example: For a data set with two attributes: AGE and HEIGHT, the following rule represents most of the data assigned to cluster 10:

If AGE >= 25 and AGE <= 40 and HEIGHT >= 5.0ft and HEIGHT <= 5.5ft then CLUSTER = 10

13.1 Unsupervised Learning

Unsupervised Learning, as clearly defined through its name, uses no classification or labeled information as input to the algorithm. Unsupervised learning is another machine learning method that uses unlabeled input data to discover patterns. Unsupervised learning aims to extract structure and patterns from unstructured data. There is no need for monitoring when learning unsupervised. Instead, it searches the data for patterns on its own.

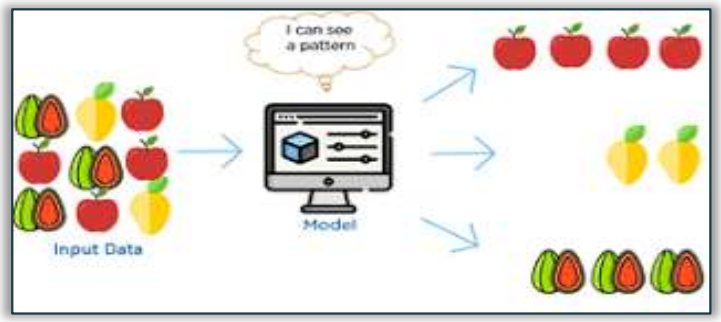


Figure 1: Unsupervised Learning

In unsupervised learning "The outcome or output for the given inputs is unknown"

Example: If the machine wants to differentiate between fishes and birds. If we do not tell the machine which animal is a fish and which is a bird (that is until we do not label them) machine will have to differentiate them by using similarities or patterns found in their attributes.

13.2 Introduction to Clustering

Clustering is the process of dividing a group of abstract items into classes of similar objects. Clustering is the grouping of similar objects, keeping in mind that: -

- objects of one cluster are similar to one another.
- objects of two different clusters differ from each other.

Clustering Requirements

The reasons why clustering is significant in data mining are as follows:

1. Scalability

To work with huge databases, we need highly scalable clustering techniques.

2. Ability to deal with a variety of attributes

Algorithms should be able to work with a variety of data types, including categorical, numerical, and binary data.

3. Cluster discovery with arbitrary shape

The method should be able to detect clusters of any shape and not be limited by distance measures.

4. Interpretability

The results should be complete, usable, and interpretable.

5. High dimensionality

Instead of merely dealing with low-dimensional data, the algorithm should be able to deal with high-dimensional space.

13.3 Clustering Methods

Clustering can be divided into two categories: hard clustering and soft clustering. One data point can only belong to one cluster in hard clustering. In soft clustering, however, the result is a probability likelihood of a data point belonging to each of the pre-defined clusters.

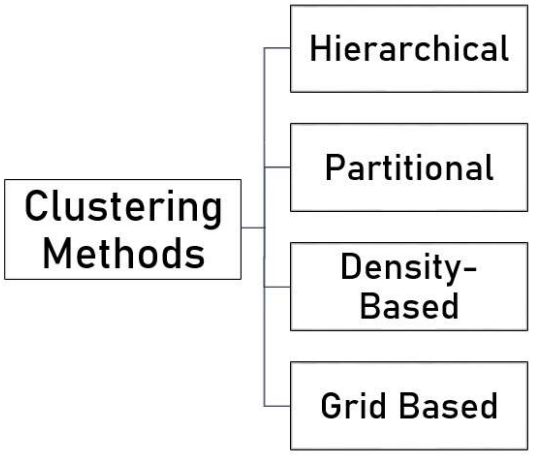


Figure 2: Clustering Methods

Hierarchical Clustering Methods

The hierarchical method decomposes the specified set of data items in a hierarchical manner. Methods can be classified based on how the hierarchical decomposition is generated. There are two approaches :

- Agglomerative
- Divisive

1. Agglomerative

The bottom-up technique is another name for this method. We begin by separating each object into its group. It continues to merge objects or groupings that are close together.

It continues to do so until all of the groups have been merged into one of the termination condition has been met.

2. Divisive

The opposite of Agglomerative, Divisive starts with all of the points in one cluster and splits them to make other clusters. These algorithms generate a distance matrix for all existing clusters and link them together based on the linkage criteria. A dendrogram is used to show the clustering of data points.

Partitional Method

This is one of the most popular methods for creating clusters among analysts. Clusters are partitioned based on the properties of the data points in partitioning clustering. For this clustering procedure, we must provide the number of clusters to be produced. These clustering algorithms use an iterative procedure to allocate data points between clusters based on their distance from one another. The following are the algorithms that fall within this category: –

1. K-Means Clustering

One of the most extensively used methods is K-Means clustering. Based on the distance metric used for clustering, it divides the data points into k clusters. The user is responsible for determining the value of 'k.' The distance between the data points and the cluster centroids is calculated. The cluster is assigned to the data point that is closest to the cluster's centroid. It computes the centroids of those clusters again after each iteration, and the procedure repeats until a pre-determined number of iterations have been completed or the centroids of the clusters have not changed after each iteration.

2. PAM (Partitioning Around Medoids)

The k-medoid algorithm is another name for this approach. It works similarly to the K-means clustering algorithm, except for how the cluster's center is assigned. The cluster's medoid must be an input data point in PAM, however, this is not true in K-means clustering because the average of all data points in a cluster may not be an input data point.

3. CLARA (Clustering Large Applications)

CLARA is a modification of the PAM method that reduces computing time to improve performance for huge data sets. To do so, it chooses a random portion of data from the entire data set to serve as a representation of the actual data. It uses the PAM algorithm to analyze several samples of data and selects the best clusters after several iterations.

Density-Based Methods

Clusters are produced using this method depending on the density of the data points represented in the data space. Clusters are locations that become dense as a result of the large number of data points that reside there.

The data points in the sparse region (the region with the few data points) are referred to as noise or outliers. These methods allow for the creation of clusters of any shape. Examples of density-based clustering methods are as follows:

1. DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

The distance metric and criterion for a minimum number of data points are used by DBSCAN to group data points. It requires two inputs: eps and minimum points. The Eps value indicates how near data points should be to be considered neighbors. To consider the region as dense, the condition for minimum points should be completed.

Notes: The DBSCAN Clustering Algorithm's complexity

Best Case: We attain O(N log N) average runtime complexity if an indexing system is employed to store the dataset and neighborhood queries are done in logarithmic time.

Worst Case: Without using an index structure or working with degraded data (e.g., all points within a distance of less than), the worst-case run time complexity remains O(n2).

Average Case: Based on the data and algorithm implementation, the average case is the same as the best/worst case.

2. OPTICS (Ordering Points to Identify Clustering Structure)

It works similarly to DBSCAN, but it addresses one of the latter's flaws: the inability to generate clusters from data of arbitrary density. It also takes into account two other parameters: core distance and reachability distance. By selecting a minimal value for core distance, it is possible to determine if a data point is a core or not. The maximum core distance and the value of the distance metric used to calculate the distance between two data points are called reachability distances. One thing to keep in mind concerning reachability distance is that if one of the data points is a core point, its value is undefined.

Grid-Based methods

The collection of information is represented in a grid structure that consists of grids in grid-based clustering (also called cells). This method's algorithms take a different approach from the others in terms of their overall approach. They're more concerned about the value space that surrounds the data points than the data points themselves. One of the most significant benefits of these algorithms is their reduced computational complexity. As a result, it's well-suited to coping with massive data sets. It computes the density of the cells after partitioning the data sets into cells, which aids in cluster identification. The following are a few grid-based clustering algorithms: –

Statistical Information Grid Approach (STING)

The data set is partitioned recursively and hierarchically in STING. Each cell is subdivided further into a distinct number of cells. It records the statistical measurements of the cells, making it easier to respond to requests in a short amount of time.

CLIQUE (Clustering in Quest)

CLIQUE is a clustering technique that combines density-based and grid-based clustering. Using the Apriori principle, it partitions the data space and identifies the sub-spaces. It determines the clusters by calculating the cell densities.

13.4 Applications of Clustering

Many applications employ data clustering analysis. Market research, pattern recognition, data analysis, and picture processing are just a few examples.

Data clustering can also aid marketers in identifying separate client groupings. They can also categorize their customers based on their purchase habits.

It can be used to create plant and animal taxonomies in the realm of biology. Genes with comparable functions should be grouped to provide insight into population structures.

Clustering aids in the identification of areas in Data Mining. In an earth observation database, this is of comparable land usage. It can also be used to locate groupings of residences in a city. This varies depending on the type of home, its value, and its location.

Clustering in Data Mining also aids in the classification of web documents for information discovery.

Data clustering is also used in outlier identification applications. For example, detecting credit card fraud.

Cluster analysis is a data mining function that can be used as a tool. That is, to get knowledge about how data is distributed. It's also necessary to pay attention to the characteristics of each cluster.

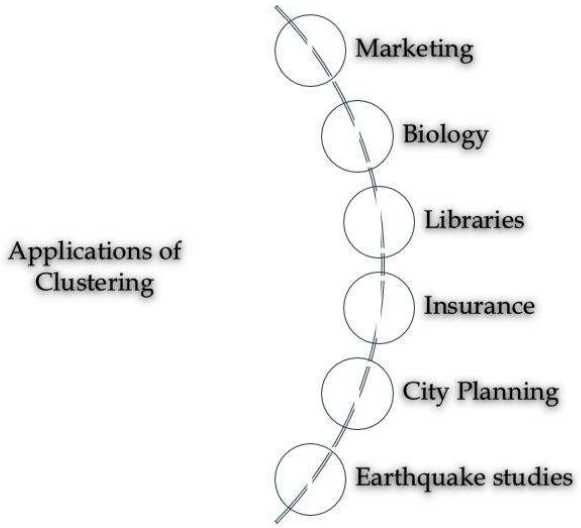


Figure 3: Application areas of clustering

13.5 The difference in Clustering and Classification

The classification of things into one or more classes based on features is done using both Classification and Clustering. They appear to be the same process since the differences are minor. In the case of classification, each input instance has predefined labels assigned to it based on its properties, whereas in the case of clustering, these labels are missing. The following table shows the difference between classification and clustering:

Table 1: Difference in classification and clustering

Parameter	Classification	Clustering
Type	Used for supervised learning	Used for unsupervised learning
Basic	Process of classifying the input instances based on their corresponding class labels	Grouping the instances based on their similarity without the help of class labels
Need	It has labels so there is a need for training and testing dataset	There is no need for training and testing dataset

	for verifying the model created	
Complexity	More complex as compared to clustering	Less complex as compared to classification
Example Algorithms	Logistic regression, Naive Bayes classifier, Support vector machines, etc.	k-means clustering algorithm, Fuzzy c-means clustering algorithm, Gaussian (EM) clustering algorithm, etc.

Differences between Classification and Clustering

1. Classification is used for supervised learning whereas clustering is used for unsupervised learning.

2. The process of classifying the input instances based on their corresponding class labels is known as classification whereas grouping the instances based on their similarity without the help of class labels is known as clustering.

3. As Classification have labels so there is need of training and testing dataset for verifying the model created but there is no need for training and testing dataset in clustering.

4. Classification is more complex as compared to clustering as there are many levels in the classification phase whereas only grouping is done in clustering.

5. Classification examples are Logistic regression, Naive Bayes classifier, Support vector machines, etc. Whereas clustering examples are k-means clustering algorithm, Fuzzy c-means clustering algorithm, Gaussian (EM) clustering algorithm, etc.

13.6 K-Mean Clustering Algorithm

The following steps show the working of the K-Means Clustering algorithm:

Step 1: Select a K value, where K denotes the number of clusters.

Step 2: Iterate through each point, assigning it to the cluster with the closest center. The centroid of all the clusters should be computed once each element has been iterated.

Step 3: Iterate through the dataset, calculating the Euclidean distance between each point and the cluster's centroid. If any point in the cluster is not nearest to it, reassign that point to the nearest cluster, and then calculate the centroid of each cluster again after doing so for all of the points in the dataset.

Step 4: Repeat Step 3 until there is no new assignment that occurred between the two iterations.

Task: What criteria can be used to decide the number of clusters in k-means statistical analysis?

Implementation of K-means in Weka

The following are the steps for implementing K-means using Weka:

1) Go to the Preprocess tab in WEKA Explorer and select Open File. Select the dataset "vote.arff."

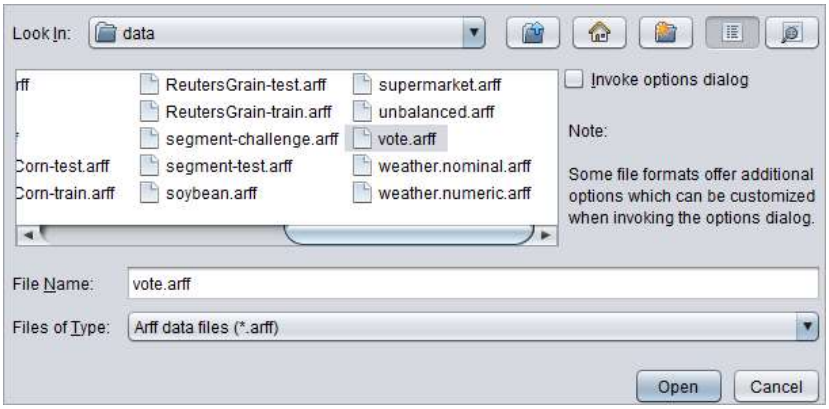


Figure 4: Dataset Selection

2) Select the "Choose" button from the "Cluster" menu. Choose "SimpleKMeans" as the clustering algorithm.

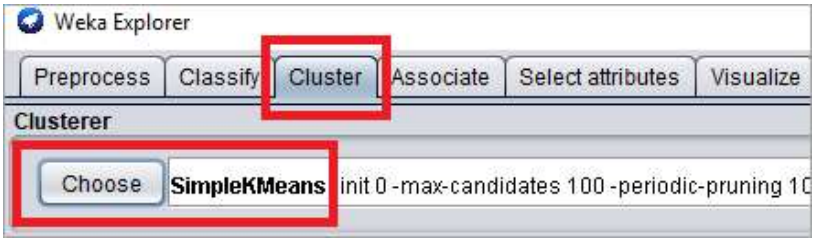


Figure 5: Selection of K-means

3) Select Settings, then fill in the following fields:

- Euclidian distance function
- There are six clusters in total. The sum of squared error will decrease as the number of clusters increases.

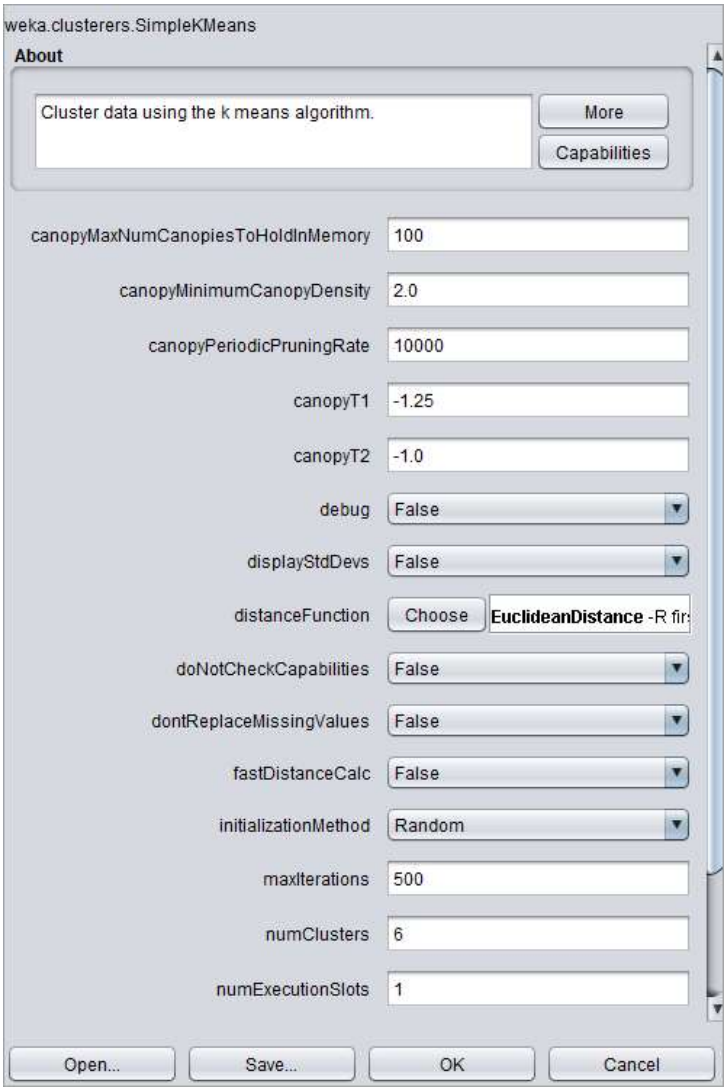


Figure 6: Parameters Setting

4. In the left panel, select Start. The algorithm's output is displayed on a white screen. Let us analyze the run information:

The properties of the dataset and the clustering process are described by the terms Scheme, Relation, Instances, and Attributes. The vote. arff dataset has 435 instances and 13 attributes in this scenario. The number of iterations with the Kmeans cluster is 5. The sum of the squared error is 1098.0. As the number of clusters grows, this inaccuracy will decrease. A table is used to represent the 5 final clusters with centroids. Cluster centroids are 168.0, 47.0, 37.0, 122.0.33.0, and 28.0 in our example.

```
=== Run information ===

Scheme:       weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-prun
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===
```

Figure 7: K-means Run information

5. Choose "Classes to Clusters Evaluations" and click on Start.

The algorithm will assign the class label to the cluster. Cluster 0 represents republican and Cluster 1 represents democrat. The Incorrectly clustered instance is 14.023 % which can be reduced by ignoring the unimportant attributes.

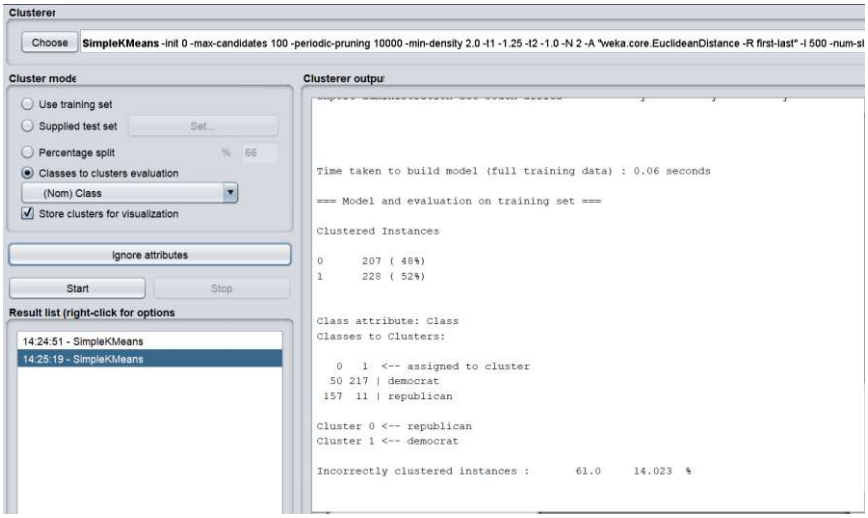


Figure 8: Cluster Formation using K-means

6. Use the "Visualize" tab to visualize the Clustering algorithm result. Go to the tab and click on any box. Move the Jitter to the max.

- The X-axis and Y-axis represent the attribute.
- The blue color represents class label democrat and the red color represents class label republican.
- Jitter is used to view Clusters.
- Click the box on the right-hand side of the window to change the x coordinate attribute and view clustering concerning other attributes.

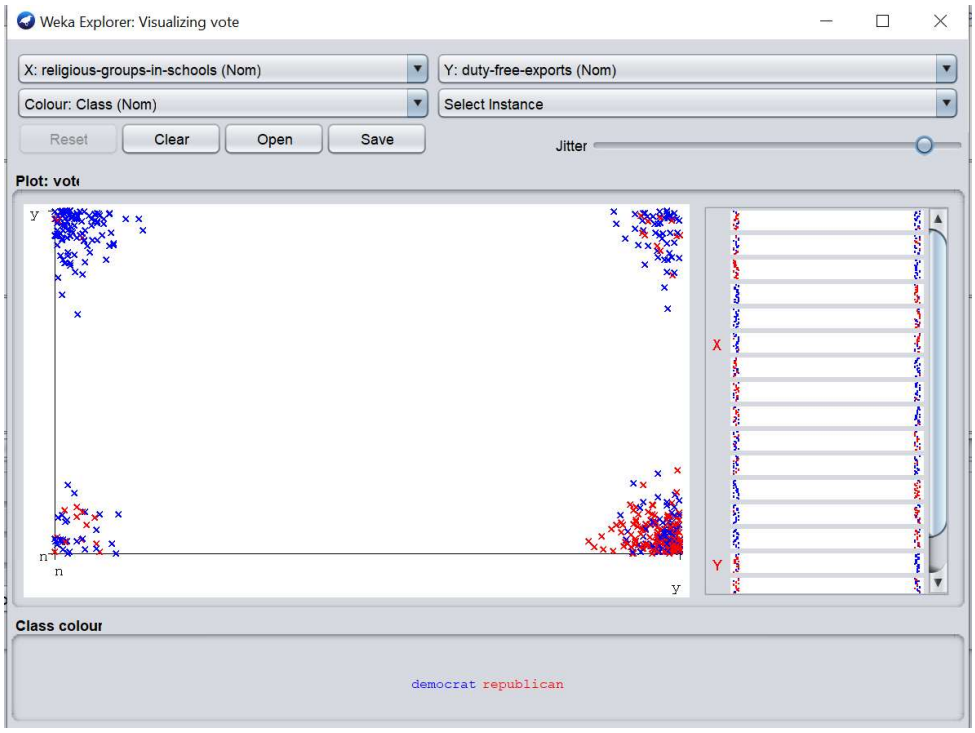


Figure 9: Visualization of Result

Caution: The number of clusters to use necessitates a precise balancing act. Larger k values can increase cluster homogeneity, but they also risk overfitting.

Implementation of Hierarchal clustering in Weka

1. Go to the Preprocess tab in WEKA Explorer and select Open File. Select the dataset "Iris.arff."

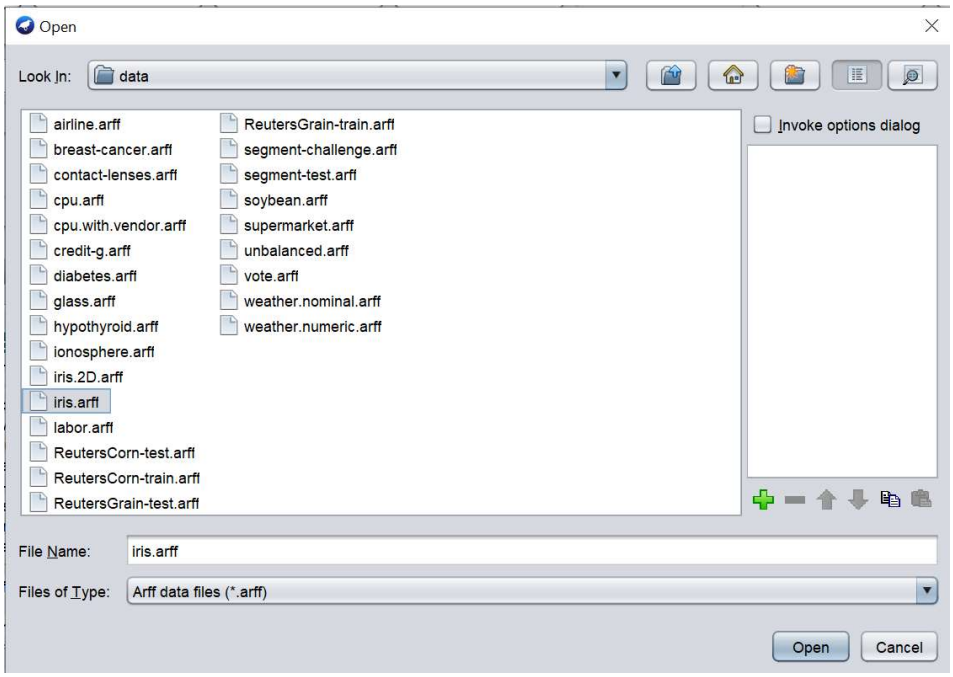


Figure 10: Selection of dataset

2. Select the "Choose" button from the "Cluster" menu. Choose "HierarichicalCluster" as the clustering algorithm.

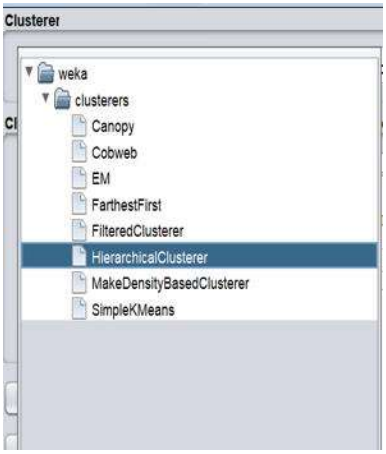


Figure 11: Selection of Algorithm

3. In the left panel, select Start. The algorithm's output is displayed on a white screen. Let us analyze the run information:

The properties of the dataset and the clustering process are described by the terms Scheme, Relation, Instances, and Attributes. The Iris. arff dataset has 150 instances and 5 attributes in this scenario.

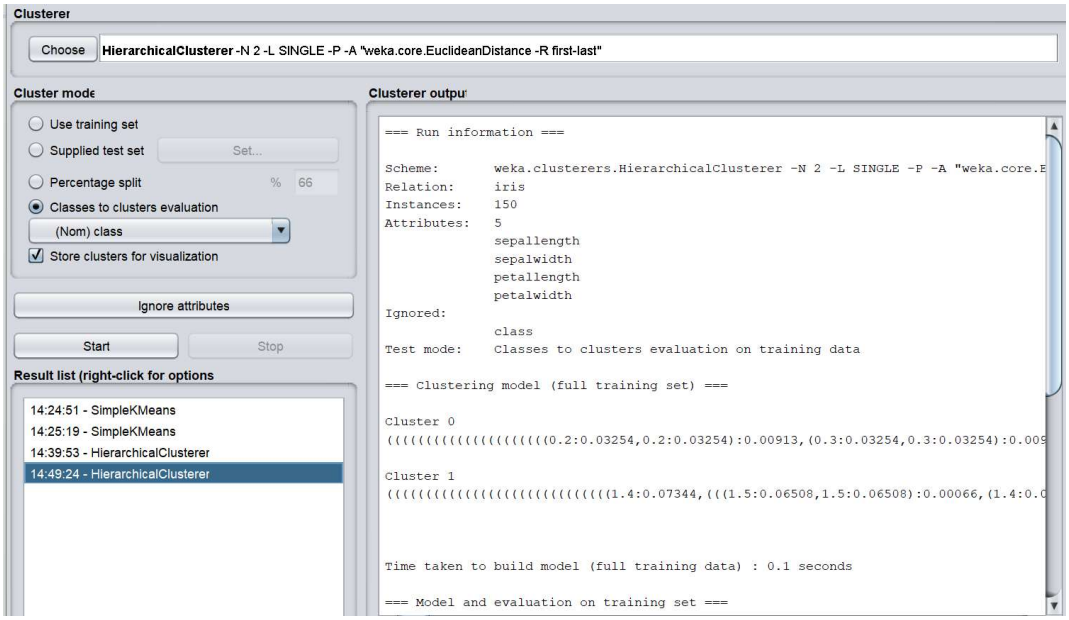


Figure 12: Run Information of Iris dataset

4. Choose "Classes to Clusters Evaluations" and click on Start.

The algorithm will assign the class label to the cluster. Cluster 0 represents Iris-setosa and Cluster 1 represents Iris-versicolor. The Incorrectly clustered instance is 33.3333 % which can be reduced by ignoring the unimportant attributes.

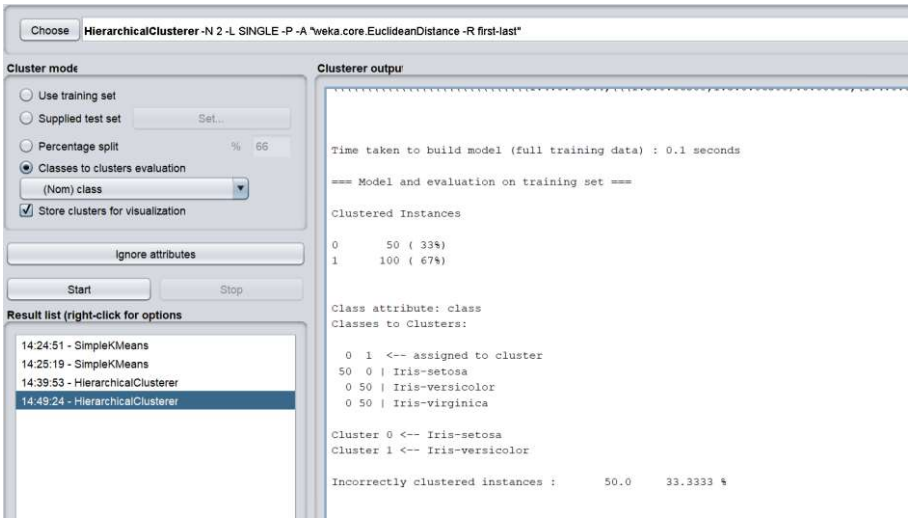


Figure 13: Cluster Formation

5 . Use the "Visualize" tab to visualize the Clustering algorithm result. Go to the tab and click on any box. Move the Jitter to the max. The X-axis and Y-axis represent the attribute.

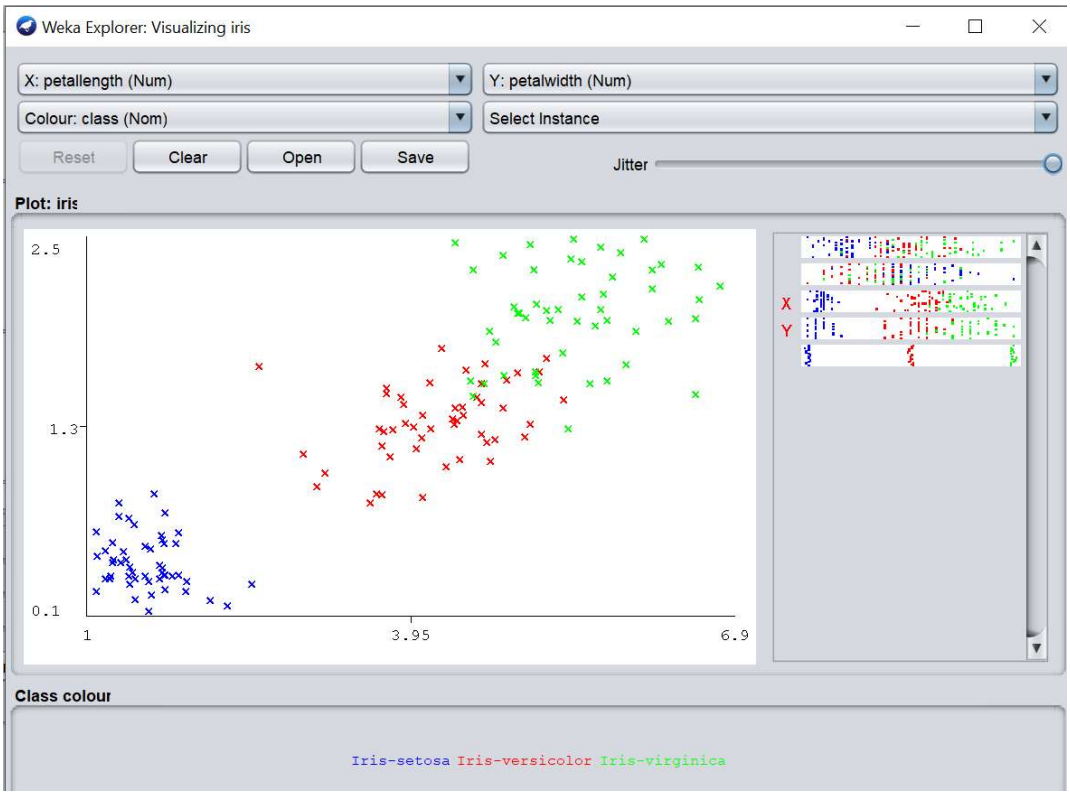


Figure 14: Visualization of Clusters

Task: Create your dataset and implement a hierarchical clustering algorithm on it and visualize the results.

Summary

- The learning of the model is 'supervised' if it is told to which class each training sample ⌘⌘belongs. In contrasts with unsupervised learning (or clustering), in which the class labels of the training samples are not known, and the number or set of classes to be learned may not be known in advance.

- The properties of the dataset and the clustering process are described by the terms Scheme, Relation, Instances, and Attributes.
- Classification is more complex as compared to clustering as there are many levels in the classification phase whereas only grouping is done in clustering.
- Data clustering can also aid marketers in identifying separate client groupings. They can also categorize their customers based on their purchase habits.
- One data point can only belong to one cluster in hard clustering. In soft clustering, however, the result is a probability likelihood of a data point belonging to each of the pre-defined clusters.
- Unsupervised learning is another machine learning method that uses unlabeled input data to discover patterns.

Keywords

Unsupervised Learning: Unsupervised Learning is a machine learning technique in which the users do not need to supervise the model.

Dendrogram: In a hierarchal clustering algorithm, we develop the hierarchy of clusters in the form of a tree, and this tree-shaped structure is known as the dendrogram.

Clustering: Grouping of unlabeled examples is called clustering.

Grid-based Methods: In this method, the data space is formulated into a finite number of cells that form a grid-like structure.

Similarity measure: You can measure similarity between examples by combining the examples' feature data into a metric, called a similarity measure.

High Dimensional Data: High-dimensional data is characterized by multiple dimensions. There can be thousands, if not millions, of dimensions.

Self Assessment Questions

1. _____uses no classification or labelled information as input to the algorithm.

A. Unsupervised Learning

B. Supervised Learning

C. Reinforcement Learning

D. None

2. In unsupervised learning the outcome or output for the given inputs is

A. Known

B. Unknown

C. Available

D. Not Available

3. Unlabeled data is used in

A. Unsupervised Learning

B. Supervised Learning

C. Reinforcement Learning

D. None

4. Clustering is the grouping of similar objects, keeping in mind that

A. Objects of one cluster are similar to one another.

B. Objects of two different clusters differ from each other.

C. Both

D. None

5. DBSCAN is an example of which of the following clustering methods.

A. Density-Based Methods

B. Partitioning Methods

C. Grid-based Methods

D. Hierarchical Based Methods

6. STING is an example of which of the following clustering methods.

A. Density-Based Methods

B. Partitioning Methods

C. Grid-based Methods

D. Hierarchical Based Methods

7. Which of the following comes under applications of clustering.

A. Marketing

B. Libraries

C. City Planning

D. All of the above

8. In _____ there is no need of training and testing dataset.

A. Classification

B. Clustering

C. Support vector Machines

D. None

9. Implementations of K-means only allow _____values for attributes.

A. Numerical

B. Categorical

C. Polynomial

D. Text

10. The WEKA SimpleKMeans algorithm uses_____distance measure to compute distances between instances and clusters.

A. Euclidean

B. Manhattan

C. Correlation

D. Eisen

11. To perform clustering, select the _____tab in the Explorer and click on the _____button.

A. Choose, Cluster

B. Cluster, Choose

C. Create, Cluster

D. Cluster, Create

12. Click on the _____tab to visualise the relationships between variables.

A. View

B. Show

C. Visualize

D. Graph

13. Examine the results in the Cluster output panel which give us:

A. The number of iterations of the K-Means algorithm to reach a local optimum.

B. The centroids of each cluster.

C. The evaluation of the clustering when compared to the known classes.

D. All of the Above

14. In _____ Weka can evaluate clustering on separate test data if the cluster representation is probabilistic.

A. In Supplied test set or Percentage split

B. Classes to clusters evaluation

C. Use training set

D. Visualize the cluster assignments

15. In this mode Weka first ignores the class attribute and generates the clustering.

A. In Supplied test set or Percentage split

B. Classes to clusters evaluation

C. Use training set

D. Visualize the cluster assignments

Review Questions

Q1) Briefly describe and give examples of each of the following approaches to clustering: partitioning methods, hierarchical methods, density-based and grid-based methods.

Q2) Elucidate the step-by-step working of K-Means with examples.

Q3) Explain the concept of unsupervised learning with examples. "Clustering is known as unsupervised learning" justify the statement with an appropriate example.

Q4) Differentiate between classification and clustering. Discuss the various applications of clustering.

Q5) With example discuss the various partitioning-based clustering methods.

Q6) Elucidate the various types of hierarchical clustering algorithms by giving the example of each type.

Answers: Self Assessment

1	A	2	B
3	A	4	C
5	A	6	C
7	D	8	B
9	A	10	A
11	B	12	C
13	D	14	A
15	B		

Further Readings

King, R. S. (2015). *Cluster analysis and data mining: An introduction*. Stylus Publishing, LLC.

Wu, J. (2012). *Advances in K-means clustering: a data mining thinking*. Springer Science & Business Media.

Mirkin, B. (2005). *Clustering for data mining: a data recovery approach*. Chapman and Hall/CRC.

Tan, P. N., Chawla, S., Ho, C. K., & Bailey, J. (Eds.). (2012). *Advances in Knowledge Discovery and Data Mining, Part II: 16th Pacific-Asia Conference, PAKDD 2012, Kuala Lumpur, Malaysia, May 29-June 1, 2012, Proceedings, Part II* (Vol. 7302). Springer.

Liu, H., & Motoda, H. (2012). *Feature selection for knowledge discovery and data mining* (Vol. 454). Springer Science & Business Media.

Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: Practical machine learning tools and techniques with Java implementations.

A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, 1988.

https://www.tutorialspoint.com/data_mining/dm_cluster_analysis.htm

https://www.upgrad.com/blog/cluster-analysis-data-mining/

https://data-flair.training/blogs/clustering-in-data-mining/

Unit 14: Applications of Data Warehousing and Data Mining

CONTENTS

Objectives

After this unit, you will be able to

- Understand the importance of data warehouses in financial data analysis and the retail industry.
- Know the importance of data warehouses in the Indian Railway reservation system and other industries.
- Learn the importance of weather forecasting in real life.

Introduction

Online transaction processing (OLTP) systems address an organization's operational data needs, which are critical to the day-to-day operation of a business. However, they are not well suited to sustaining decision support or business questions that managers are frequently faced with. Analytics, such as aggregation, drill-down, and slicing/dicing of data, are best supported by online analytical processing (OLAP) systems for such questions. By storing and managing data in a multidimensional manner, data warehouses aid OLAP applications. Extract, Transfer, and Load (ETL) tools are used to extract and load data into an OLAP warehouse from numerous OLTP data sources. The requirement for a data warehouse is driven by a business need and a business strategy. A data warehouse is a foundation for sophisticated data analysis; it aids business decision-making by allowing managers and other company users to analyze data and conduct analysis more efficiently.

14.1 Case Study of data warehouse in Financial Data Analysis

Financial Data Analysis

A financial data warehouse built with cutting-edge technology can help us improve the quality of our data and obtain insights into client behavior. As a result of these new insights, clients frequently increase the performance of their marketing campaigns and loyalty programs. Our financial analytics services are business-driven, and our professionals take the time to understand our specific needs and goals. Our consulting services assist financial companies in achieving the following goals:

- Make more informed decisions that are simple to access.
- Introduce new products to the marketplace.

- There are an increasing amount of restrictions.

Financial data warehouses work in the same way as regular data warehouses do. After data is acquired and put in a data warehouse, it is structured into a specific schema that categorizes the data. When the data needs to be analyzed, this allows for easy access.

A financial data warehouse using the latest technologies can increase the quality of our data and help us to gain insights into customer behavior. Clients often improve the effectiveness of their marketing campaigns and loyalty programs as a result of these new insights. Our financial analytics services are business-driven and our experts spend time establishing our requirements and task.

The Challenge

Every client stakeholder wanted an answer to the same question: How are sales going? There was no single location where executives, finance managers, and sales managers could get a comprehensive picture of the sales team's operations and evaluate key data such as present performance, prior performance, and pipeline. The client's IT department was having trouble keeping its governance and planning initiatives on track. They had taken measures to centralize the numerous data sources they needed to deal with and get the ball rolling on accurate, actionable reporting on those assets, but their infrastructure was outdated and no longer capable of handling the increased information demands. They engaged Ironside to help them upgrade their old data warehouse to a more current system that would help them achieve three important department goals:

- Remove warehouse infrastructure from its existing state of obsolescence.
- For more accurate referencing of past events, save historical point-in-time data.
- Increase query efficiency and enable more timely analyses by addressing the pain points that exist between the data warehouse and the Cognos BI reporting layer.

The Journey

The Ironside Information Management team agreed to collaborate with the client's database engineers to make the move to a modern data warehouse a success. The Ironside resources assigned to the project outlined a full-scale migration and redesign plan that would bring about 20 tables from a variety of data sources, including TM1, A-Track, Remedy, Oracle GL, and Excel, into an IBM PureData for Analytics (Netezza) implementation capable of meeting and exceeding the client's requirements, based on discovery conversations with IT leadership. Ironside was also tasked with reengineering both the ETL processes required to transport and transform all of the numerous information streams, as well as the reporting layer that makes that information available for analysis, as part of the migration.

So far, Ironside has completed the following phases:

- For the new data warehouse, They gathered requirements and use cases.
- The existing warehouse logic was documented, including all data extraction, transformations, schedules, and so on.
- Served as the solution architect for a data warehouse redesign proof of concept.
- Set specifications for new ETL operations to feed the POC design with the client's database engineers.
- The reporting layer was rebuilt to work in the new environment.
- Using the POC system, they tested data handling and report output.

The Results

The initial findings coming out of the concepting and testing phases of the project are very promising, and Ironside's team is confident that the final solution will deliver the modern data handling functionality that the client needs to continue their success.

The positive outcomes are:

1. Improved Cognos reporting usability and efficiency.

2. Support time comparison reporting, such as comparing data from the previous month to the current day.

3. During testing, a performance boost of roughly 60 times faster than achievable in the old environment was noticed.

Ironside and the customer are now making modifications and ready to roll out the full-scale data warehouse solution, using this strong proof of concept as a springboard. The IT staff will be able to easily fulfill the severe expectations of the financial services industry and give the sort of answers that will propel the company forward with this level of data performance at their disposal.

Advantages of Data Warehouses in the Financial Industry

Following are some of the advantages:

1. Analytic Types

Many businesses are interested in using financial data for a range of analytics purposes. Predictive and real-time analytics are the most popular types of analytics in finance. The goal of predictive analytics is to find patterns in financial data to anticipate future events. Real-time analytics is employed in a variety of applications, including consumer intelligence, fraud detection, and more.

2. Capturing Customer Data

Customers today use various channels on multiple devices, making data collection more challenging and likely to become more complex in the future. Data warehouses enable businesses to record every connection with a consumer, providing them with unprecedented insight into what motivates them.

3. Personalization

Many financial institutions have pushed to engage more with clients to stay competitive. One approach to achieve this is to send more tailored messages, which is possible thanks to data science. The ability to create more tailored interactions means that you can reach out to customers at the most appropriate times and with the most effective message.

4. Risk Management

Many financial institutions have pushed to engage more with clients to stay competitive. One approach to achieve this is to send more tailored messages, which is possible thanks to data science. The ability to create more tailored interactions means that you can reach out to customers at the most appropriate times and with the most effective message.

Example: Data warehouses are largely utilized in the investing and insurance industries to assess consumer and market trends, as well as other data patterns. Data warehouses are critical in two key sub-sectors: forex and stock markets, where a single point discrepancy can result in enormous losses across the board. Data warehouses are typically shared in these industries and focus on real-time data streaming.

14.2 Data Warehouse in Retail Industry

The retail industry collects a large amount of database and information on sales and customer shopping history. The quantity and quality of data and information collected continue to rapidly, especially due to the increasing ease, availability, and popularity the business conducted on the web, or retail industry provides a rich source for data mining. Retail database mining can help identify :

- Customer behavior,
- Discover customer shopping trends,
- Improve the quality of customer service and patterns,
- Achieve better customer retention and satisfaction and feedback,
- Enhance goods consumption ratios design more effective goods transportation and distribution policies and reduce the cost of business.

Why the Retail Industry Needs Data Warehousing and Analytics?

Every business, including retail, is being disrupted by the digital revolution, which is offering tremendous new opportunities. AI, robotics, blockchain, and data analytics are all transforming the way the retail industry interacts with and serves its customers. Retailers are utilizing data warehousing and analytics in novel and unique ways, all with the goal of better serving their customers and improving their shopping experience. Data warehousing and analytics have various

advantages, including increased operational efficiency, improved customer experience, and customer loyalty and retention.

The following are four ways that data warehousing and analytics are transforming the retail industry:

1. A Data Warehouse Saves Time

Data warehousing allows you to develop a centralized system for operations, data collection, and analysis. This eliminates the requirement for manual data exporting to excel sheets for reporting across corporate functions. Business users may get vital information from a variety of sources in one place. They would not squander time obtaining information from different sources.

2. Enhanced Business Intelligence

Data warehousing and analytics gather information from several departments and show it in reports and dashboards. Furthermore, the dashboard's self-service capabilities allow users to slice and dice data, execute drill-down and drill-up tasks, and see data at both a holistic and granular level. This allows business users to gain key business insights and make quick, well-informed decisions to improve operational efficiency and growth.

3. Demand Forecasting and scaling of operations

Retailers require demand forecasting because their scope of operation does not remain constant throughout the year. It varies dramatically depending on the season. For retailers, for example, the holiday season accounts for a significant portion of annual revenues. As a result, it is vital that they scale up to meet higher demand during peak seasons and then scale down to eliminate excess inventory when sales return to normal.

4. Better understanding of customers

Retailers can employ data warehousing and analytics to acquire a better understanding of their customers' behavior, tastes, and buying habits, and use the information for personalized offers, contextual marketing, and event planning shop design and aesthetics.

Data warehousing and analytics have become important to any retailer's success as the retail business grows and becomes more competitive. To meet business goals and generate important customer insights, merchants must be able to connect the proper, relevant data.

Notes: The Data Warehouse Data Model for Retail and Distribution (Retail DWH model ®) is a standard industry data warehouse model that covers traditional BI requirements, regulatory needs, and Big Data Analytics requirements for retailers and wholesalers.

Background of Apex Supermarket

Apex Grocery is a supermarket chain with 50 locations across the United States, including the West Coast, East Coast, Midwest, and Southwest. Georgia, New York, and Virginia are on the East Coast. Illinois and Kansas are in the Midwest. Texas is in the southwest.

Traditional System Limitation

- There is currently a lack of ability to make strategic decisions.
- No drill-down or roll-up reports are possible.
- It is not possible to conduct multi-dimensional analysis and make decisions.

Why a data warehouse

Allow Decision Support

1. Provide an information system for executives to use while making strategic decisions, such as:

- What is the most popular time for school supplies to be purchased?
- Which racks must school supplies be put to attract clients during the "school season"?

2. Establish an information system for executives to use in making strategic decisions, such as:

- Where in the store do perishable commodities such as fruits and vegetables need to be moved quickly?

- How long can meat meals be kept on the rack at Apex's Illinois location?

3. Roll-up and drill-down capabilities in the Central Repository provide sales by category, region, store, and more.

Example: For distribution and marketing, data warehouses are commonly employed. It also aids in the tracking of items, client purchasing patterns, and promotions, as well as for deciding pricing strategy.

14.3 Data Warehouse in Railway Reservation System

A data warehouse is a centralized storage location for all or large portions of the data collected by an organization's numerous business systems. Bill Inmon invented the phrase. The term "information warehouse" is sometimes used by IBM.

The passenger reservation system (PRS) is an OLTP system with a large number of concurrent reservation users actively adding and changing data. As the online PRS generates an increasing volume of data, the need to analyze data is becoming more and more by higher management for

1. Managing and utilizing the existing resources.
2. Creating new resources for the convenience of passengers.

When railway management tries to examine the data, however, so many of these issues often prevent it from being done:

- Railway executives (and even railway programmers) are unable to develop ad hoc inquiries.
- The application database is fragmented over several servers around the country, making it difficult for railway users to locate data in the first place.
- Ad-hoc querying of PRS OLTP systems is prohibited by database administrators to prevent analytical users from running queries (e.g., how many passengers are booked for a specific journey Day) that slow down mission-critical PRS databases.

Aim of study

We have recorded all of the information on the trains scheduled and the users booking tickets, as well as the condition of the trains, seats, and so on, in our study of the railway reservation system. This database is useful for applications that allow customers to book train tickets and check train details and status from their home, avoiding the inconvenience of having to go to the railway station for every question. Passengers can use the Railway Reservation System to inquire about:

1. The trains that are available based on their origin and destination.

2 . Ticket booking and cancellation, as well as inquiring about the status of a booked ticket, and so on.

3. The goal of this case study is to design and create a database that stores information about various trains, train status, and passengers.

The major goal of keeping a database for the Railway Reservation System is to limit the number of manual errors that occur during ticket booking and cancellation. and make it easy for customers and providers to keep track of information about their clients as well as available seats. Many flaws in manual record-keeping can be eliminated due to automation.

The data will be obtained and processed at a rapid pace. The suggested system could be web-enabled in the future to allow clients to make numerous inquiries about trains between stations. As a result, a variety of issues arise from time to time, and they are frequently involved in consumer disputes. Some assumptions have been made to implement this sample case study, which is as follows:

1. The number of trains is limited to five.

2. The reservation is only available for the following seven days from the present date.

3. There are only two types of tickets available for purchase: AC and General.

4. There are a total of 10 tickets available in each category (AC and General).

5. The maximum number of tickets that can be given a waiting status is two.

6. The time between halt stations and their reservations is not taken into account.

Process

A train list must be kept up to date. Passenger information must be kept up to date. The passenger's train number, train date, and category are read throughout the booking process. An appropriate record from the Train Status is fetched based on the values provided by the passenger. If the desired category is AC, then the total number of AC seats and the number of booked AC seats are compared to find whether a ticket can be booked or not.

It can be checked in the same way for the general category. Passenger details are read and stored in the Passenger database if a ticket may be booked. The passenger's ticket ID is read during the cancellation procedure, and the Passenger database is checked for a matching record. The record is erased if it exists. After deleting the record (if it is confirmed), the Passenger table is searched for the first record with a waiting status for the same train and category, and its status is updated to confirm.

CRIS: The Need

For the following reasons, it was thought that a separate organization would be better prepared to take over all computer activity on IR:

1. To prevent individual railways from duplicating their efforts.

2.To ensure that computer hardware and software on the railways are all the same.

3. To design and create big railway applications that necessitate higher degrees of expertise, faster decision-making, and system-wide applicability.

4. to protect the organization from the day-to-day operations of the railways so that its goals are not forgotten.

5. The necessity for a collaborative effort between railways and computer specialists who are best qualified to build computer applications for railways.

6. Expertise development in highly specialized disciplines such as operations research, simulation, expert systems, CAD/CAM, and process control is required.

7. A greater degree of flexibility is required to keep up with rapidly changing technology.

14.4 Applications of Data Warehouse

The following are the various applications of data warehouse in various industries:

Government and Education: Data warehouses are used by the government to preserve and analyze tax records, health policy records, and their related providers, and the state's data warehouse is also related to the criminal law database. Patterns and trends, which are the outcome of analyzing historical data linked with prior criminals, are used to anticipate criminal activity.

Healthcare: All of their financial, clinical, and employee data is loaded into warehouses, which allows them to strategize and anticipate outcomes, track and analyze customer feedback, make patient reports, and exchange data with tie-in insurance companies, medical aid agencies, and so on.

Hospitality Industry: Hotel and restaurant services, automobile rental services, and vacation home services account for a large percentage of this industry. They employ warehousing services to create and assess their ad and marketing programs, which target customers based on their feedback and travel patterns.

Insurance: Apart from keeping track of existing participants, the warehouses are largely used to evaluate data patterns and customer trends. Warehouses can also be used to create customized consumer offers and promotions.

Manufacturing and Distribution Industry: This industry is one of a state's most important sources of revenue. A manufacturing company must make several make-or-buy decisions that can have a significant impact on the industry's future, which is why they use high-end OLAP tools as part of data warehouses to forecast market changes, analyze current business trends, detect warning conditions, track marketing trends, and, ultimately, to predict market changes. Data warehouses are used to handle the supply chain management of products in distributions.

Telephone Industry: The telephone industry uses both offline and online data, resulting in a large amount of historical data that must be aggregated and integrated.

Apart from those processes, a data warehouse is required for the study of fixed assets, the study of customer calling habits for salespeople to push advertising campaigns, and the tracking of customer inquiries.

Transportation Industry: Client data is recorded in data warehouses in the transportation industry, allowing traders to experiment with target marketing, where marketing campaigns are created with the needs of the customer in mind. They are used in the industry's internal environment to monitor customer feedback, performance, crew management on board, and customer financial information for pricing strategies.

14.5 Data Warehouse in Weather Forecasting

Weather forecasting has traditionally relied on huge, complicated physics models that take into account a variety of atmospheric circumstances over a lengthy period. Because of weather system perturbations, these conditions are frequently unstable, causing models to make erroneous forecasts. We describe a weather prediction methodology in this study that uses historical data from several weather stations to train simple machine learning models that can produce meaningful forecasts for specific weather conditions soon in a short amount of time.

Weather conditions around the world change rapidly and continuously. In today's world, accurate forecasts are critical. We rely significantly on weather forecasts in all we do, from agriculture to industry, from travel to the daily commute. Because the entire world is affected by climate change and its consequences, it is critical to accurately predict the weather to maintain smooth and seamless movement as well as safe day-to-day activities.

Weather Forecasting

The first step in constructing a data warehouse is to gather forecaster knowledge to construct the subject system. Weather forecasting is the use of science and technology to anticipate the state of the atmosphere in a specific region. Temperature, rain, cloudiness, wind speed, and humidity are all factors to consider. Weather warnings are a type of short-range forecast that is used to protect people's lives.

According to forecasters, there are currently six categories of subjects in our subject system:

1. **Target Element**: The target element of the forecast, such as rainfall, temperature, humidity, and wind, must be included in the subject system.

2.**Index Data**: Some Index data are useful; they are usually flags or indicators of future weather, such as the surface wind direction in Wutaishan, the 500hPa height change in west Siberia, and so on.

3. **Statistical Data**: Subjects like the average of 5 stations, the maximum temperature gradient in someplace, minimum relative humidity in the last 5 days, and others should be included in the subject system to analyze statistical features in a region or period.

4. **Transformed Data**: orthogonal transformations, such as wavelet transformation. And data that has been filtered using lowpass, highpass, and other techniques.

5. **Weather System:** Even though the forecaster's knowledge is based on the study of model output, synoptic expertise is vital. As a result, the weather system should be part of the topic system. High, low, large gradient area, saddle area for scale element and convergence center, and divergence are all examples of weather systems.

The Forecasting Process

Making a weather forecast involves three steps:

- Observation and analysis
- Extrapolation to find the future state of the atmosphere.
- Prediction of particular variables.

Methods of weather Prediction

In Weather prediction, there are three methods available:

- **Synoptic Weather Prediction**
- **Numerical Weather Prediction**
- **Statistical Weather Prediction**

Synoptic Weather Prediction

It is the most basic and conventional method of weather forecasting. Until the late 1950s, this approach was still in use. The term "synoptic" refers to the observation of various weather elements at a certain period. To a meteorologist, a weather map that represents the atmospheric conditions at a specific time is a synoptic chart. Creating synoptic charts regularly necessitates a massive gathering and analysis of observational data from thousands of weather stations. Certain empirical laws have been formed based on years of careful study of weather charts. These criteria aid forecasters in forecasting the rate and direction of weather system migration.

Numerical Weather Prediction

The numerical method necessitates a significant amount of mathematics. Numerical Weather Prediction techniques are now used in modern weather forecasting (NWP). This strategy is based on the fact that atmospheric gases adhere to a set of physical laws. The theoretical models of the general circulation of the atmosphere are developed using a series of mathematical equations. These equations are used to describe how the atmosphere evolves. Certain meteorological factors, such as air movements, temperatures, humidity, evaporation at the ground, clouds, rain, snow, and interactions of air with the earth and oceans, are taken into account in these equations. The daily weather prediction model is one such thing. On mobile phones, we see these forecasts.

Statistical Weather Prediction

Along with numerical weather prediction computations, statistical methods are primarily utilized. These methods are frequently used in conjunction with numerical methods. Statistical methods rely on weather data from the past, assuming that the future will be similar to the past. The major goal of looking at historical weather data is to figure out which features of the weather are good predictors of future events. Correct data can be safely utilized to anticipate future conditions after these linkages have been established. This method can only predict overall weather. It's especially useful for forecasting just one aspect of the weather at a time.

How is your business linked to the weather?

- Knowing the accurate situation of the weather is an important element for individuals and organizations.
- Many businesses are directly or indirectly linked with weather conditions.
- For instance, agriculture relies on perfect weather forecasting for when to plant, irrigate and harvest.
- Similarly, other occupations like construction, airport control authorities, and many more businesses are dependent on the weather.
- With weather forecasting, your organization can work more accurately without any disturbance.

Given the wide and essential need for accurate forecasting of weather conditions, data intelligence is powered by AI techniques that leverage real-time weather feeds and historical data. Live weather feeds from different locations (Latitude/Longitude) are available. Temperature, water level, wind, and other sensors continuously transmitting data. Historical weather data are publically available. Weather maps (precipitations, clouds, pressure, temperature, wind, weather stations) provide necessary information which will further help in the forecasting process.

Advantages of weather forecasting for businesses

- People get warned earlier of what the weather will be like for that particular day.
- Help people to take appropriate precautions to stay safe in case of unwanted occurrences.
- With forecasting methods, companies can get better outcomes with the help of accurate predictions.
- Delivers visual forecasts by methods most companies prefer.

- Helps agricultural organizations in buying/selling livestock.
- Helps the farming industry in planting crops, pastures, water supplies.
- The best method for inventory management, selling strategies, and crop forecasts.
- It provides the business with valuable information that the business can use to make decisions about future business strategies.

Challenges

Weather forecasts still have their limitations despite the use of modern technology and improved techniques to predict the weather. Weather forecasting is complex and not always accurate, especially for days further in the future, because the weather can be chaotic and unpredictable. If weather patterns are relatively stable, the persistence method of forecasting provides a relatively useful technique to predict the weather for the next day. Weather observation techniques have improved and there have been technological advancements in predicting the weather in recent times. Despite this major scientific and technical progress, many challenges remain regarding long-term weather predictability. The accuracy of individual weather forecasts varies significantly.

Summary

- Every business, including retail, is being disrupted by the digital revolution, which is offering tremendous new opportunities.
- The major goal of keeping a database for the Railway Reservation System is to limit the number of manual errors that occur during ticket booking and cancellation. and make it easy for customers and providers to keep track of information about their clients as well as available seats.
- A data warehouse is a centralized storage location for all or large portions of the data collected by an organization's numerous business systems.
- Hospitality Industry employs warehousing services to create and assess their ad and marketing programs, which target customers based on their feedback and travel patterns.
- Weather forecasting is complex and not always accurate, especially for days further in the future, because the weather can be chaotic and unpredictable.
- Data warehousing and analytics have various advantages, including increased operational efficiency, improved customer experience, and customer loyalty and retention.

Keywords

Key attribute: The key attribute is the attribute in a dimension that identifies the columns in the dimension main table that is used in foreign key relationships to the fact table.

Integrated: A data warehouse is developed by combining data from multiple heterogeneous sources, such as flat files and relational databases, which consequently improves data analysis.

Data Mart: A data mart performs the same function as a data warehouse, but with a smaller scope. It could be tailored to a certain department or line of business.

Self Assessment Questions

1. Using _____ Clients often improve the effectiveness of their marketing campaigns and loyalty programs as a result of these new insights.

A. Financial Data Analysis

B. Data Analysis

C. Cash data Analysis

D. None

2. Financial data consulting services help financial businesses to:

A. Make decisions more effectively and easy to access.

B. Bring new products onto the market.

C. A growing number of regulations.

D. All of the Above

3. _____collects large amount of database and information on sales and customer shopping history.

A. Retail Industry

B. Finance Industry

C. Manufacturing Industry

D. Production Industry

4. Retail database mining can help identify

A. Customer behavior

B. Discover customer shopping trends

C. Improve the quality of customer service and patterns

D. All of the Above

5. Which of the following is/are new-age technologies

A. AI

B. Data Analytics

C. Blockchain

D. All of the Above

6. A _____ is a central repository for all or significant parts of the data that an enterprise's various business systems collect.

A. Data warehouse

B. Data Mart

C. DataBase

D. None

7. _____is an OLTP system characterized by large numbers of concurrent reservation user's actively adding and modifying data.

A. The person reservation systems

B. The passenger reservation systems

C. The personal reservation systems

D. None

8. Railway reservation system stored all the information about

A. Trains scheduled

B. Users booking tickets

C. Status of trains and seats

D. All of the Above

9. The Railway Reservation System facilitates the passengers to enquire about

A. The trains available based on source and destination.

B. Booking and Cancellation of tickets

C. Enquire about the status of the booked ticket

D. All of the above

10. The main purpose of maintaining a database for the Railway Reservation System is

A. To reduce the manual errors involved in the booking and canceling of tickets

B. Make it convenient for the customers and providers to maintain the data about their customers

C. Data about the seats available

D. All of the Above

11. _____is the prediction of the state of the atmosphere for a given location using the application of science and technology.

A. Weather forecasting

B. Temperature forecasting

C. Rain forecasting

D. Humidity forecasting

12. _____ are a special kind of short-range forecast carried out for the protection of human life.

A. Weather Messages

B. Weather Warnings

C. Weather Information

D. None

13. On an everyday basis, many use _____to determine what to wear on a given day.

A. Rain forecasting

B. Temperature forecasting

C. Weather forecasting

D. None

14. Which of the following steps making a weather forecast

A. Observation and analysis

B. Extrapolation to find the future state of the atmosphere.

C. Prediction of particular variables.

D. All of the Above

15. It is the traditional and basic approach adopted in weather prediction.

A. Synoptic Weather Prediction

B. Numerical Weather Prediction

C. Statistical Weather Prediction

D. None

Review Questions

Q1) What is the significance of weather forecasting in today's business? Explain the various methods used in weather prediction.

Q2) Discuss the various applications of Data Warehouse by giving the example of each application.

Q3) "Every business, including retail, is being disrupted by the digital revolution, which is offering tremendous new opportunities". Justify the statement by giving the appropriate example.

Q4) Discuss a scenario that shows how a data warehouse is helpful in the railway reservation system.

Q5) Elucidate the role of Data Warehouse in financial data analysis.

Q6) Discuss the various challenges faced during weather forecasting.

Answers: Self Assessment

1	A	2	D
3	A	4	D
5	D	6	A
7	B	8	D
9	D	10	D
11	A	12	B
13	C	14	D
15	A		

Further Readings

Wang, J. (Ed.). (2005). Encyclopedia of data warehousing and mining. iGi Global.

Inmon, W. H. (2005). Building the data warehouse. John Wiley & sons.

Adamson, C., & Venerable, M. (1998). Data warehouse design solutions. J. Wiley & Sons.

Blackwood, B. D. (2015). QlikView for Finance. Packt Publishing Ltd.

Pover, K. (2016). Mastering QlikView Data Visualization. Packt Publishing Ltd.

Kimball, R., & Ross, M. (2011). The data warehouse toolkit: the complete guide to dimensional modeling. John Wiley & Sons.

https://www.import.io/post/using-a-data-warehouse-in-financial-services-how-web-data-integration-helps-you-win-in-the-financial-industry/

https://www.csub.edu/training/pgms/fdwp2/index.html

https://www.voicendata.com/retail-industry-needs-data-warehousing-analytics/

https://dwh-models.com/solutions/retail/

LOVELY PROFESSIONAL UNIVERSITY
Jalandhar-Delhi G.T. Road (NH-1)
Phagwara, Punjab (India)-144411
For Enquiry: +91-1824-521360
Fax.: +91-1824-506111
Email: odl@lpu.co.in