DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

DATA MINING

LAB MANUAL

Subject Code :

Regulation : R18/JNTUH

Academic Year : 2022-2023

III B. TECH I SEMESTER

COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

KG REDDY COLLEGE OF ENGINEERING AND TECHNOLOGY

Autonomous, Chilkur Village, Moinabad Mandal, Hyderabad, Telangana 501504

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

VISION AND MISSION OF THE INSTITUTION

VISION:

To become an institution which is internationally recognized for its holistic approach to engineering, innovative teaching and learning culture, research and entrepreneurial ecosystem, and sustainable social impact in the community.

MISSION:

• To offer undergraduate and post-graduate programs which are supported through industry relevant curriculum and innovative teaching and learning processes that would help students succeed in their professional careers.
• To provide faculty and students with an ecosystem that fosters innovation, research, entrepreneurship, and international exposure through strategic partnerships with government organizations and collaboration with industries.
• To provide holistic learning environment to students, which will contribute to their personal and professional growth and enable them to become leaders in their respective fields.
• To contribute to the development of the region by using our technological expertise to work with nearby communities and support them in their social and economic development.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

VISION AND MISSION OF THE DEPARTMENT

VISION:

To be recognized as a department of excellence by stimulating a learning environment in which students and faculty will thrive and grow to achieve their professional, institutional and societal goals.

MISSION:

• To provide high quality technical education to students that will enable life-long learning and build expertise in advanced technologies in Computer Science and Engineering.
• To promote research and development by providing opportunities to solve complex engineering problems in collaboration with industry and government agencies.
• To encourage professional development of students that will inculcate ethical values and leadership skills through entrepreneurship while working with the community to address societal issues.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

PROGRAM EDUCATIONAL OBJECTIVES

PEO 1: Graduates will provide solutions to difficult and challenging issues in their profession by applying computer science and engineering theory and principles.
PEO 2: Graduates have successful careers in computer science and engineering fields or will be able to successfully pursue advanced degrees.
PEO 3: Graduates will communicate effectively, work collaboratively and exhibit high levels of professionalism, moral and ethical responsibility.
PEO 4: Graduates will develop the ability to understand and analyze engineering issues in a broader perspective with ethical responsibility towards sustainable development.

PROGRAM OUTCOMES

• **PO I: Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
• **PO II: Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
• **PO III: Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
• **PO IV: Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
• **PO V: Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
• **PO VI: The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

professional engineering practice.
• **PO VII: Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of and need for sustainable development.
• **PO VIII: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
• **PO IX: Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
• **PO X: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
• **PO XI: Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
• **PO XII: Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

PSO1: Graduates will be able to apply the knowledge of human cognition and modern tools to solve complex real-world problems to meet the challenges of the future.
PSO2: Graduates will be able to utilize innovative Artificial Intelligence tools and techniques to develop the robotic systems, of inter-disciplinary domains for pursuing higher studies, research and entrepreneurship.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Name of the Laboratory: **Data Mining Lab**

Things to DO:

1. Be on time. At the start of the lab period, there will be a short introduction to the experiment you will perform that day. It is unfair to your partner and to others in the lab if you are not up to speed when the work begins.
2. Inform the instructor and/or TA if there is a problem. You will have their immediate attention if you have cut yourself (even if you consider it minor), if something broke and needs cleaning up, or if you are on fire.
3. Be aware of all the safety devices. Even though the instructor and TA will take care of emergencies, you should know where to find the first aid kit, the chemical spill kit, the eye wash and the safety shower.
4. Keep clutter to a minimum. There is a coat rack to hang your jackets and there are empty cabinets to store your backpacks. Anything left in the aisles is likely to be stepped on and is a hazard to everyone.
5. Wash your hands before you leave the lab for the day.
6. Be aware of others in the lab. Areas of the room may be crowded at times and you should take care not to disturb the experiments of others in the lab.
7. Bring your lab notebook and an open mind to every lab meeting.

Things NOT TO DO:

1. Do not eat, drink, chew gum, smoke or apply cosmetics in the lab. Just being in lab makes your hands dirtier than you can imagine and you don't want to accidentally eat any reagent (see item 5 on 'things to do' list).
2. Do not put pieces of lab equipment in your mouth. It sounds obvious but you'd be surprised!
3. Do not work with chemicals until you are sure of their safe handling. This includes some awareness of their flammability, reactivity, toxicity, and disposal.
4. Do not use the phone or computer with gloves on your hands.

HoD, CSE-CSD **Principal**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

COURSE OBJECTIVES:

1. The course is intended to obtain hands-on experience using data mining software.
2. Intended to provide practical exposure of the concepts in data mining algorithms
3. Practical exposure on implementation of well known data mining tasks.
4. Exposure to real life data sets for analysis and prediction.
5. Handling a small data mining project for a given practical domain.

COURSE OUTCOMES:

1. Apply preprocessing statistical methods for any given raw data.
2. Gain practical experience of constructing a data warehouse.
3. Implement various algorithms for data mining in order to discover interesting patterns from large amounts of data.
4. Analyze the data and apply appropriate algorithm for decision making
5. Apply OLAP operations on data cube construction.

MAPPING OF COURSE OUTCOMES WITH PROGRAM OUTCOMES:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	2								3	2
CO2	3	3			2								3	2
CO3	2	2	3	2	2	2		2	2		2	2	3	2
CO4	2	2	2	3	3			3			3	3	2	3
CO5	2	2	2	2	3	3	3	3	3		3	3	2	3

3-High 2-Medium 1-Low

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

DATAWARE HOUSE TOOLS:

Cloudera	
Teradata	
Oracle	
TabLeau	

OPEN SOURCE DATA MINING TOOLS:

WEKA	
Pentaho	
Orange	
KNIME	
R-Programming	

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Name of the Lab: DATA MINING LAB

INDEX

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 1: Perform data preprocessing tasks

Preprocess Tab

1. Loading Data

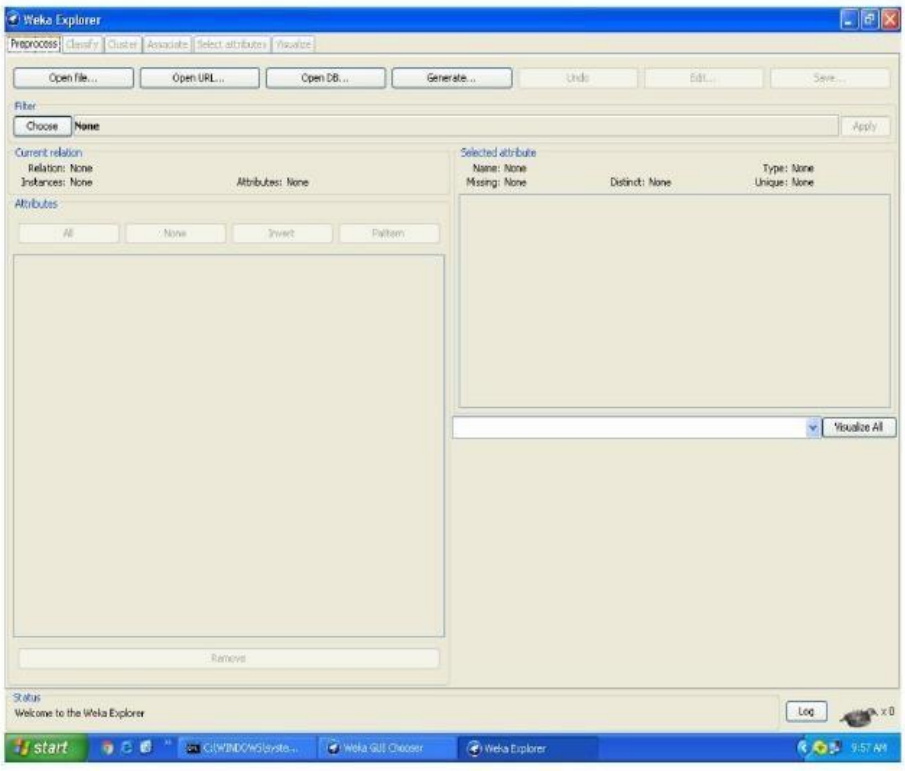
The first four buttons at the top of the preprocess section enable you to load data into WEKA are,

a) **Open file**: Brings up a dialog box allowing you to browse for the data file on the local file system.

b) **Open URL**: Asks for a Uniform Resource Locator address for where the data is stored.

c) **Open DB**: Reads data from a database. (Note that to make this work you might have to edit the file in WEKA/experiment/DatabaseUtils.props.)

d) **Generate**: Enables you to generate artificial data from a variety of Data Generators. Using the Open file button you can read files in a variety of formats: WEKA's ARFF format, CSV format, C4.5 format, or serialized Instances format. ARFF files typically have a .arff extension, CSV files a .csv extension, C4.5 files a .data and .names extension, and serialized Instances objects a .bsi extension.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Current Relation: Once some data has been loaded, the Preprocess panel shows a variety of information.

The Current relation box (the ―current relation‖ is the currently loaded data, which can be interpreted as a single relational table in database terminology) has three entries:

1. **Relation.** The name of the relation, as given in the file it was loaded from. Filters (described below) modify the name of a relation.
2. **Instances.** The number of instances (data points/records) in the data.
3. **Attributes.** The number of attributes (features) in the data.

Working with Attributes

Below the Current relation box is a box titled Attributes. There are four buttons, and beneath them is a list of the attributes in the current relation.

The list has three columns:

1. **No.** A number that identifies the attribute in the order they are specified in the data file.
2. **Selection tick boxes**. These allow you select which attributes are present in the relation.
3. **Name.** The name of the attribute, as it was declared in the data file. When you click on different rows in the list of attributes, the fields change in the box to the right titled selected attribute.

This box displays the characteristics of the currently highlighted attribute in the list:

1. **Name.** The name of the attribute, the same as that given in the attribute list.
2. **Type.** The type of attribute, most commonly Nominal or Numeric.
3. **Missing.** The number (and percentage) of instances in the data for which this attribute ismissing (unspecified).
4. **Distinct.** The number of different values that the data contains for this attribute.
5. **Unique.** The number (and percentage) of instances in the data having a value for this attributethat no other instances have.

Below these statistics is a list showing more information about the values stored in this attribute, which differ depending on its type. If the attribute is nominal, the list consists of each possible value for the attribute along with the number of instances that have that value. If the attribute is numeric, the list gives four statistics describing the distribution of values in the data— the minimum, maximum, mean and standard deviation. And below these statistics there is a coloured histogram, colour-coded according to the attribute chosen as the Class using the box above the histogram. (This box will bring up a drop-down list of available selections when clicked.)Note that only nominal Class attributes will result

in a color-coding. Finally, after pressing the Visualize All button, histograms for all the attributes in the data are shown in a separate window. Returning to the attribute list, to begin with all the tick boxes are unticked. They can be toggled on/off by clicking on them individually. The four buttons above can also be used to change the selection.

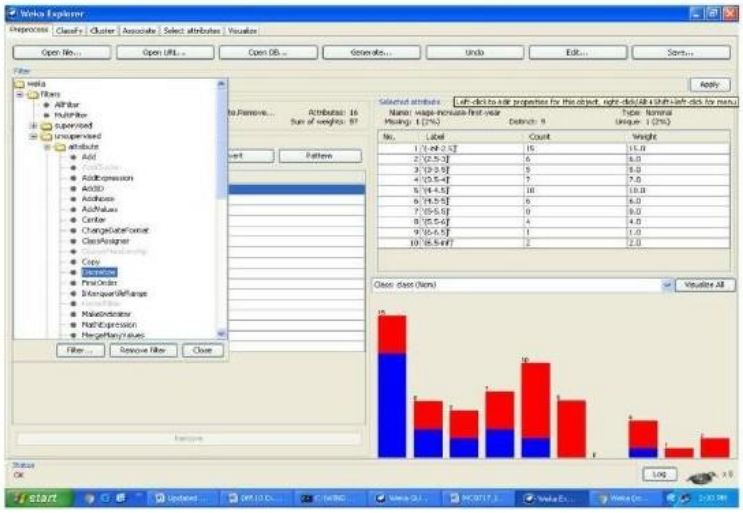
PREPROCESSING

1. **All.** All boxes are ticked.
2. **None.** All boxes are cleared (unticked).
3. **Invert.** Boxes that are ticked become unticked and vice versa.
4. **Pattern.** Enables the user to select attributes based on a Perl 5 Regular Expression. E.g., .*id selects all attributes which name ends with id.

Once the desired attributes have been selected, they can be removed by clicking the Remove button below the list of attributes. Note that this can be undone by clicking the Undo button, which is located next to the Edit button in the top-right corner of the Preprocess panel.

Working with Filters:-

The preprocess section allows filters to be defined that transform the data in various ways. The Filter box is used to set up the filters that are required. At the left of the Filter box is a Choose button. By clicking this button it is possible to select one of the filters in WEKA. Once a filter has been selected, its name and options are shown in the field next to the Choose button. Clicking on this box with the left mouse button brings up a GenericObjectEditor dialog box. A



click with the right mouse button (or Alt+Shift+left click) brings up a menu where you can choose, either to display the properties in a GenericObjectEditor dialog box, or to copy the current setup string to the clipboard.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

The GenericObjectEditor Dialog Box

The GenericObjectEditor dialog box lets you configure a filter. The same kind of dialog box is used to configure other objects, such as classifiers and clusterers. Thefields in the window reflect the available options.

Right-clicking (or Alt+Shift+Left-Click) on such a field will bring up a popup menu, listing thefollowing options:

a) **Show properties...** has the same effect as left-clicking on the field, i.e., a dialog appearsallowing you to alter the settings.

b) **Copy configuration** to clipboard copies the currently displayed configuration string to the **system's clipboar**d and therefore can be used anywhere else in WEKA or in the console. This is rather handy if you have to setup complicated, nested schemes.

c) **Enter configuration... is the —receiving‖ end for configurations that** got copied to the clipboard earlier on. In this dialog you can enter a class name followed by options (if the class supports these). This also allows you to transfer a filter setting from the Preprocess panel to a Filtered Classifier used in the Classify panel.

Left-Clicking on any of these gives an opportunity to alter the filters settings. For example, the setting may take a text string, in which case you type the string into the text field provided. Orit may give a drop-down box listing several states to choose from. Or it may do something else, depending on the information required. Information on the options is provided in a tool tip if you let the mouse pointer h over of the corresponding field. More information on the filter and its options can be obtained by clicking on the More button in the **About** panel at the top of the GenericObjectEditor window.

Applying Filters

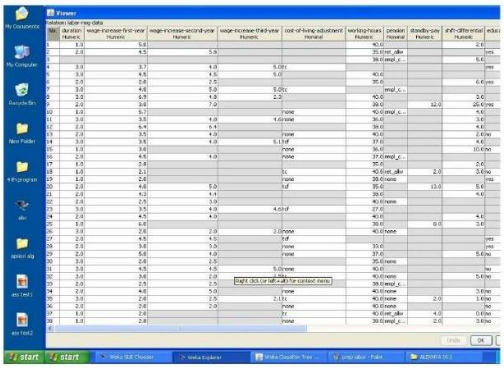
Once you have selected and configured a filter, you can apply it to the data by pressing the Apply button at the right end of the Filter panel in the Preprocess panel. The Preprocess panel will then show the transformed data. The change can be undone by pressing the Undo button. You can also use the Edit...button to modify your data manually in a dataset editor. Finally, the Save...button at the top right of the Preprocess panel saves the current version of the relation in file formats that can represent the relation, allowing it to be kept for future use.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

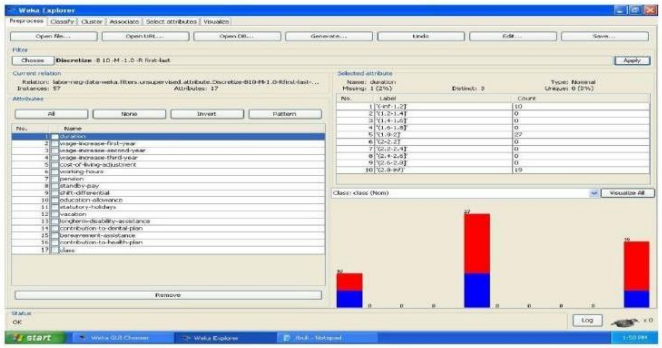
Steps for run preprocessing tab in WEKA:

1. Open WEKA Tool.
2. Click on WEKA Explorer.
3. Click on Preprocessing tab button
4. Click on open file button.
5. Choose WEKA folder in C drive.
6. Select and Click on data option button.
7. Choose labor data set and open file.
8. Choose filter button and select the Unsupervised-Discritize option and apply

Dataset labor.arff



The Following screen shows the Descretization



Exercise 1:

a. **Remove spaces from the file by using Java.**

b. **Normalize the data using min-max normalization**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

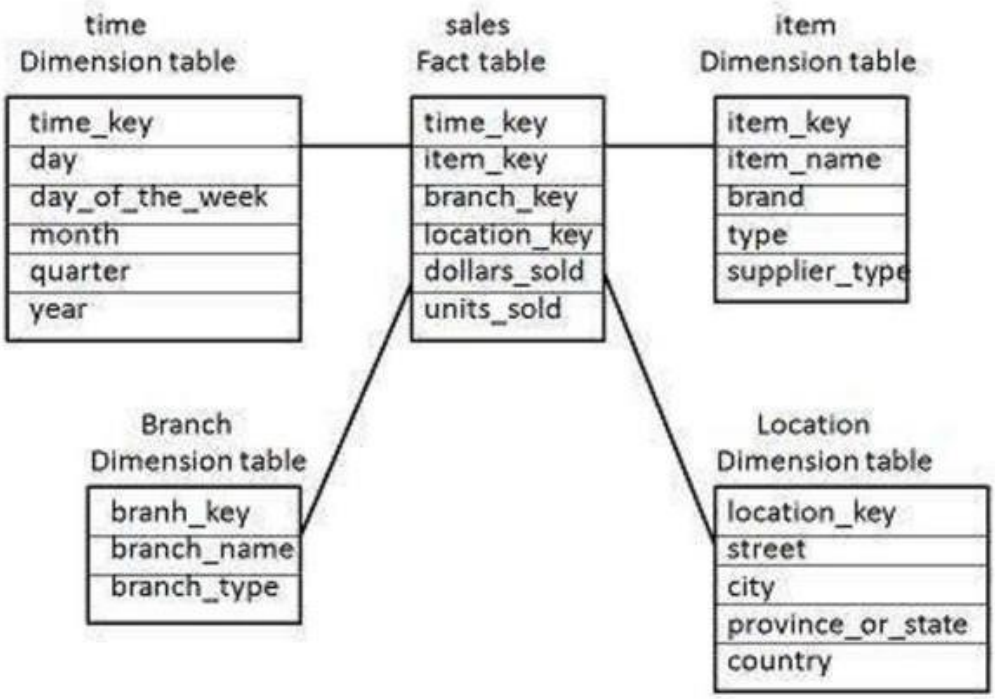
Experiment 2: Design multi-dimensional data models namely Star, Snowflake and Fact Constellation schemas for any one enterprise (ex. Banking, Insurance, Finance, Healthcare, manufacturing, Automobiles, sales etc).

Schema Definition

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

a). Star Schema

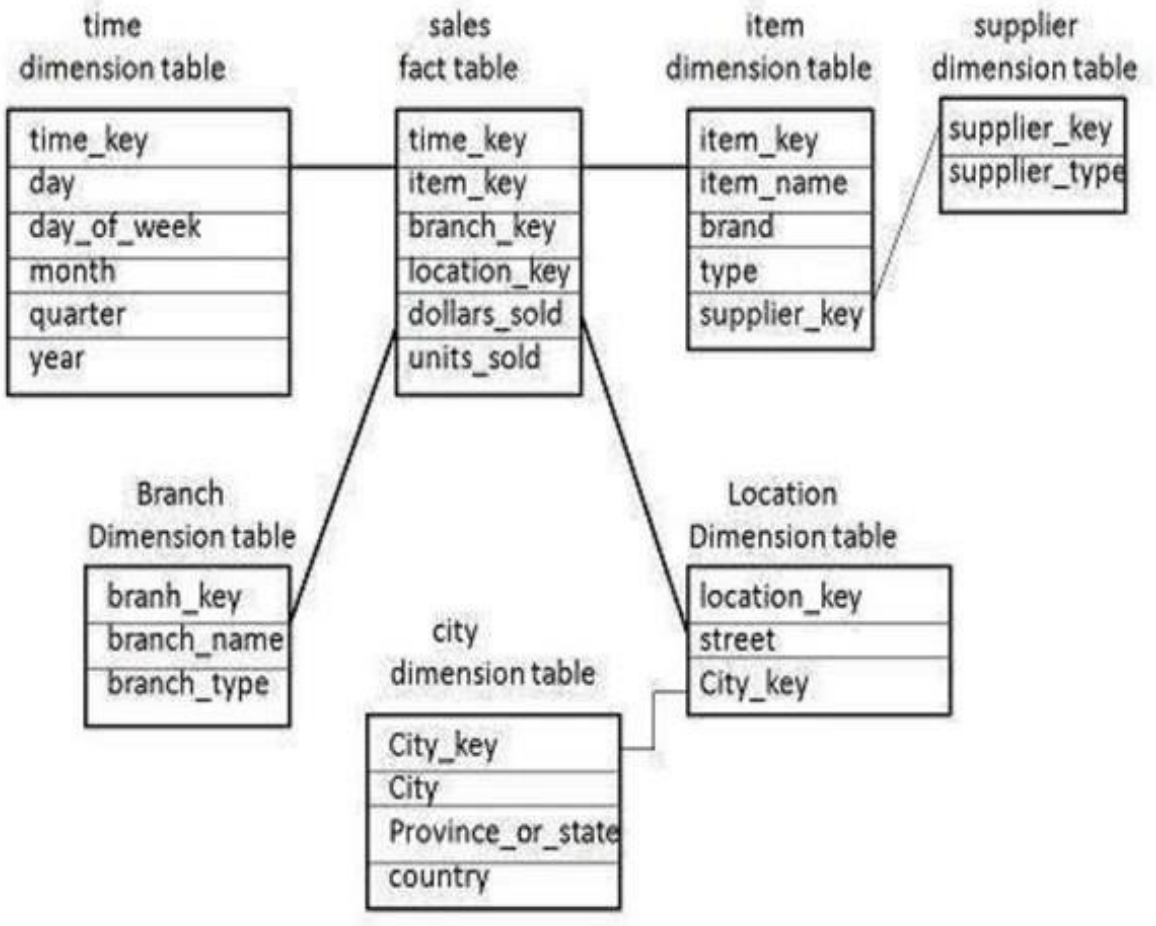
- Each dimension in a star schema is represented with only one-dimension table.
- This dimension table contains the set of attributes.
- The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.
- There is a fact table at the center. It contains the keys to each of four dimensions.
- The fact table also contains the attributes, namely dollars sold and units sold.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

b). Snowflake Schema

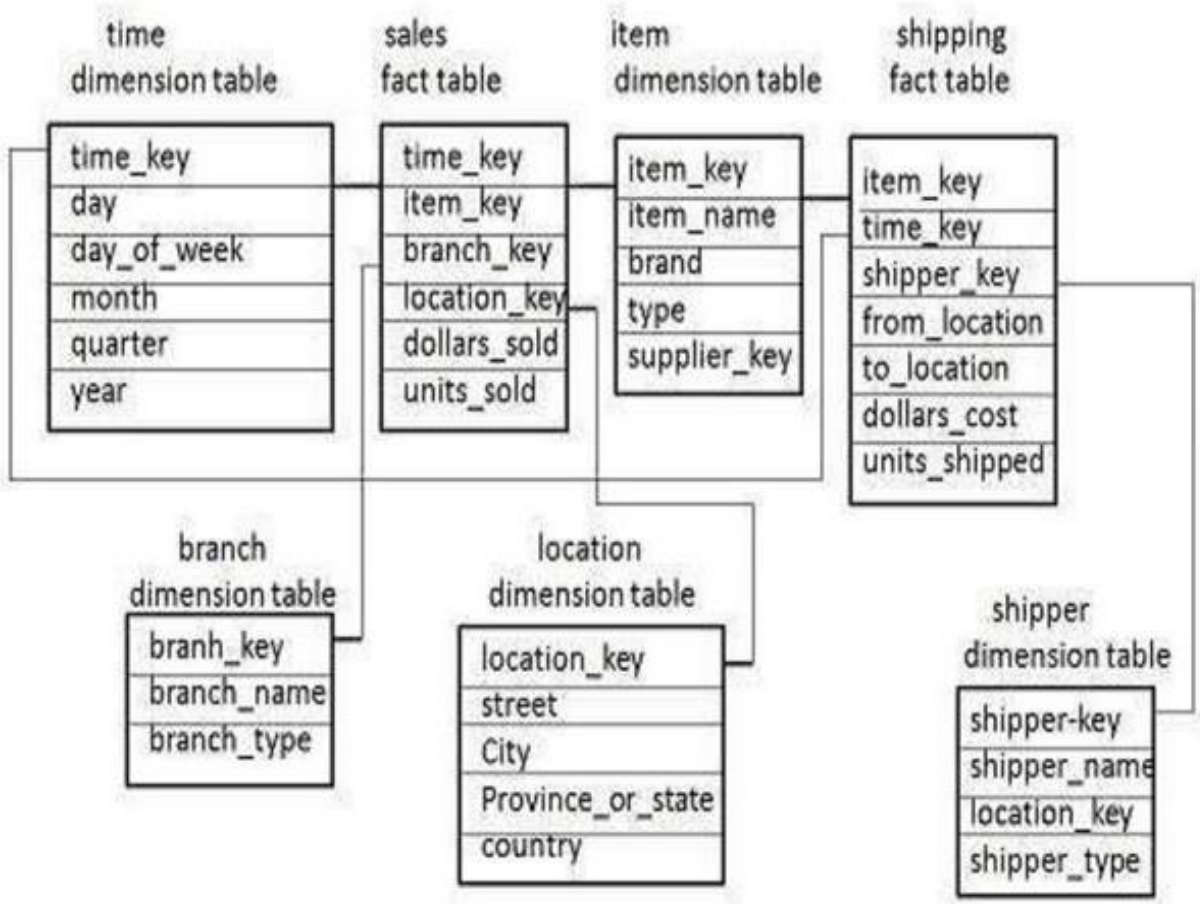
- Some dimension tables in the Snowflake schema are normalized.
- The normalization splits up the data into additional tables.
- Unlike Star schema, the dimensions table in a snowflake schema is normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.
- Now the item dimension table contains the attributes item_key, item_name, type, brand, and supplier-key.
- The supplier key is linked to the supplier dimension table. The supplier dimension table contains the attributes supplier_key and supplier_type.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

c). Fact Constellation Schema

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



Exercise 2: Design data warehouse schemas for Banking application.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 3: Perform Various OLAP operations such slice, dice, roll up, drill up and pivot.

OLAP OPERATIONS

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

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

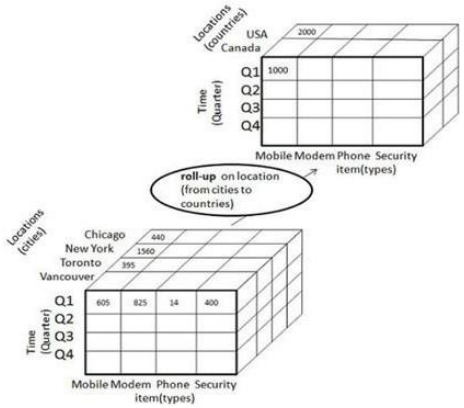
Roll-up

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

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

The following diagram illustrates how roll-up works.

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



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

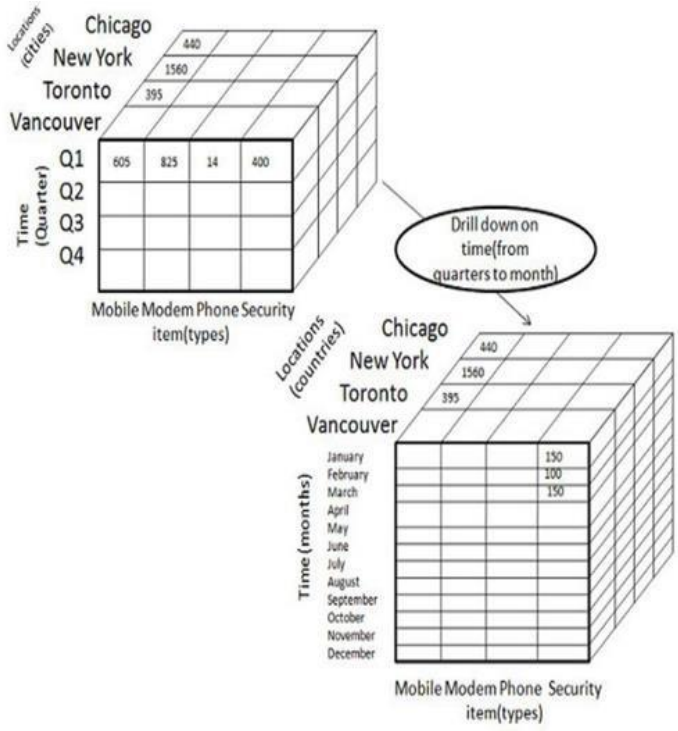
Drill-down

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

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

The following diagram illustrates how drill-down works:

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

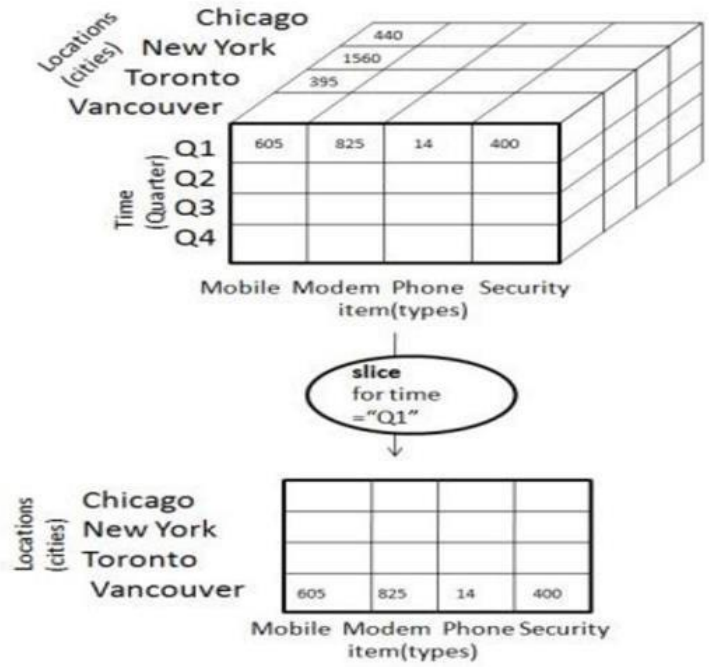


Slice

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

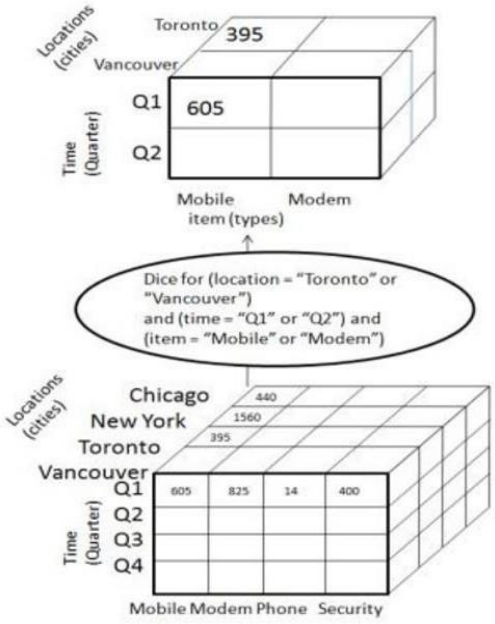
- Here Slice is performed for the dimension "time" using the criterion time = "Q1".
- It will form a new sub-cube by selecting one or more dimensions.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE



Dice

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



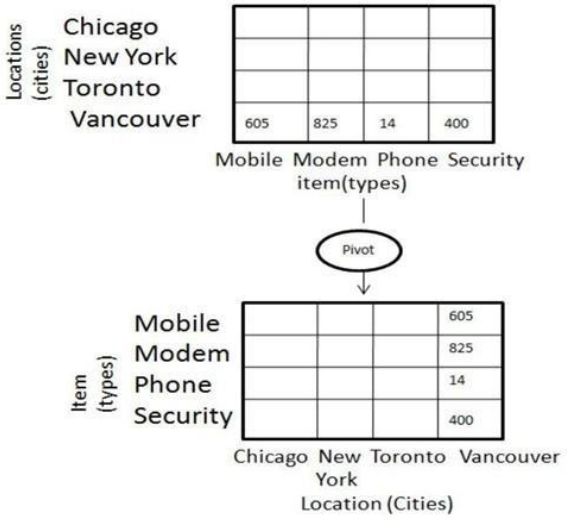
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

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

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

Pivot

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



Exercise 3:

Apply the OLAP operations for the above banking application.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 4: ETL scripts and implement using data warehouse tools.

ETL comes from Data Warehousing and stands for Extract-Transform-Load. ETL covers a process of how the data are loaded from the source system to the data warehouse. Extraction– transformation–loading (ETL) tools are pieces of software responsible for the extraction of data from several sources, its cleansing, customization, reformatting, integration, and insertion into a data warehouse.

Building the ETL process is potentially one of the biggest tasks of building a warehouse; it is complex, time consuming, and consumes most of data warehouse project‘s implementation efforts, costs, and resources.

Building a data warehouse requires focusing closely on understanding three main areas:

a) Source Area- The source area has standard models such as entity relationship diagram.

b) Destination Area- The destination area has standard models such as star schema.

c) Mapping Area- But the mapping area has not a standard model till now.

ETL Process:

Extract

The Extract step covers the data extraction from the source system and makes it accessible for further processing. The main objective of the extract step is to retrieve all the required data from the source system with as little resources as possible. The extract step should be designed in a way that it does not negatively affect the source system in terms or performance, response time or any kind of locking.

There are several ways to perform the extract:

- **Update notification -** if the source system is able to provide a notification that a record has been changed and describe the change, this is the easiest way to get the data.
- **Incremental extract -** some systems may not be able to provide notification that an update has occurred, but they are able to identify which records have been modified and provide an extract of such records. During further ETL steps, the system needs to identify changes and propagate it down. Note, that by using daily extract, we may not be able to handle deleted records properly.
- **Full extract -** some systems are not able to identify which data has been changed at all, so a full extract is the only way one can get the data out of the system. The full extract requires keeping a copy of the last extract in the same format in order to be able to identify changes. Full extract handles deletions as well.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Transform

- The transform step applies a set of rules to transform the data from the source to the target. This includes converting any measured data to the same dimension (i.e. conformed dimension) using the same units so that they can later be joined. The transformation step also requires joining data from several sources, generating aggregates, generating surrogate keys, sorting, deriving new calculated values, and applying advanced validation rules.

Load

- During the load step, it is necessary to ensure that the load is performed correctly and with as little resources as possible. The target of the Load process is often a database. In order to make the load process efficient, it is helpful to disable any constraints and indexes before the load and enable themback only after the load completes. The referential integrity needs to be maintained by ETL tool to ensure consistency.

ETL method

- ETL as scripts that can just be run on the database. These scripts must be re-runnable: they should be able to be run without modification to pick up any changes in the legacy data, and automaticallywork out how to merge the changes into the new schema.

In order to meet the requirements, my scripts must:

a) INSERT rows in the new tables based on any data in the source that hasn‘t already beencreated in the destination
b) UPDATE rows in the new tables based on any data in the source that has already been insertedin the destination
c) DELETE rows in the new tables where the source data has been deleted

Next step is to design the architecture for custom ETL solution.

a) Create two new schemas on the new database: LEGACY and MIGRATE
b) Take a snapshot of all data in the legacy database, and load it as tables in the LEGACY schema.
c) Grant read-only on all tables in LEGACY to MIGRATE
d) Grant CRUD on all tables in the target schema to MIGRATE.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

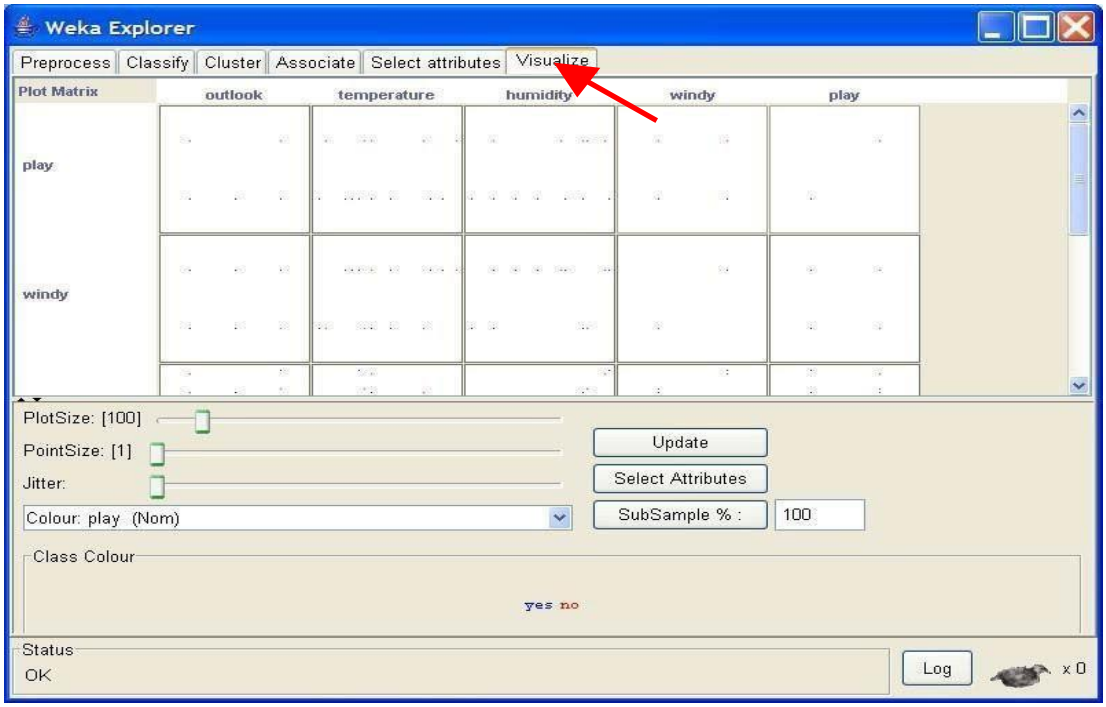
WEKA

Visualization Features:

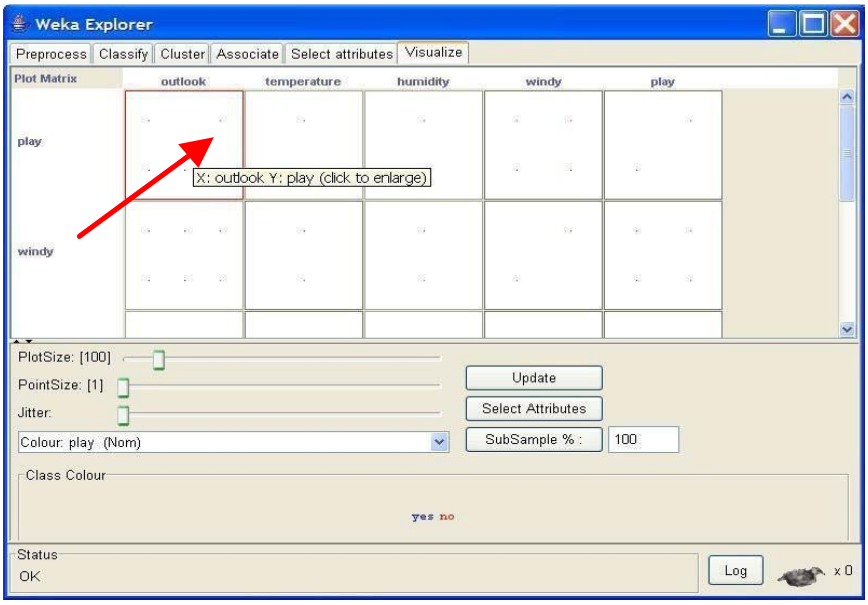
WEKA's visualization allows you to visualize a 2-D plot of the current working relation. Visualization is very useful in practice, it helps to determine difficulty of the learning problem. WEKA can visualize single attributes (1-d) and pairs of attributes (2-d), rotate 3-d visualizations (Xgobi-style). WEKA has "Jitter" option to deal with nominal attributes and to detect "hidden" data points.

Access To Visualization From The Classifier, Cluster And Attribute Selection Panel is Available From A Popup Menu. Click The Right Mouse Button Over An Entry In The Result List To Bring Up The Menu. You Will Be Presented With Options For Viewing Or Saving The Text Output And --- Depending On The Scheme --- Further Options For Visualizing Errors, Clusters, Trees Etc.

- To open Visualization screen, click 'Visualize' tab.
- Select a square that corresponds to the attributes you would like to visualize. For example, let's choose 'outlook' for X – axis and 'play' for Y – axis. Click anywhere inside the square that corresponds to 'play on the left and 'outlook' at the top.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE



Changing the View:

In the visualization window, beneath the X-axis selector there is a drop-down list, 'Colour', for choosing the color scheme. This allows you to choose the color of points based on the attribute selected. Below the plot area, there is a legend that describes what values the colors correspond to. In your example, red represents 'no', while blue represents 'yes'. For better visibility you should change the color of label 'yes'. Left-click on 'yes' in the 'Class colour' box and select lighter color from the colorpalette.

To the right of the plot area there are series of horizontal strips. Each strip represents an attribute, and the dots within it show the distribution values of the attribute. You can choose what axes are used in the main graph by clicking on these strips (left-click changes X-axis, right- click changes Y-axis).

The software sets X - axis to 'Outlook' attribute and Y - axis to 'Play'. The instances are spread out in the plot area and concentration points are not visible. Keep sliding 'Jitter', a random displacement given to all points in the plot, to the right, until you can spot concentration points.

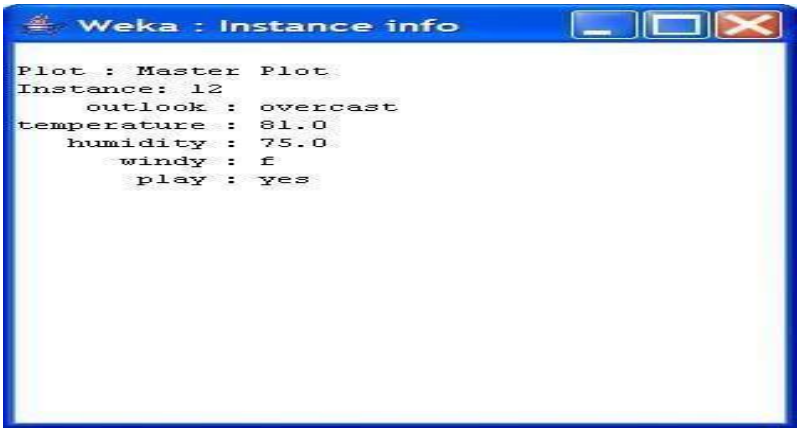
The results are shown below. But on this screen we changed 'Colour' to temperature. Besides 'outlook' and 'play', this allows you to see the 'temperature' corresponding to the 'outlook'. It will affect your result because if you see 'outlook' = 'sunny' and 'play' = 'no' to explain the result, you need to see the 'temperature' – if it is too hot, you do not want to play. Change 'Colour' to 'windy', you can see that if it is windy, you do not want to play as well.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

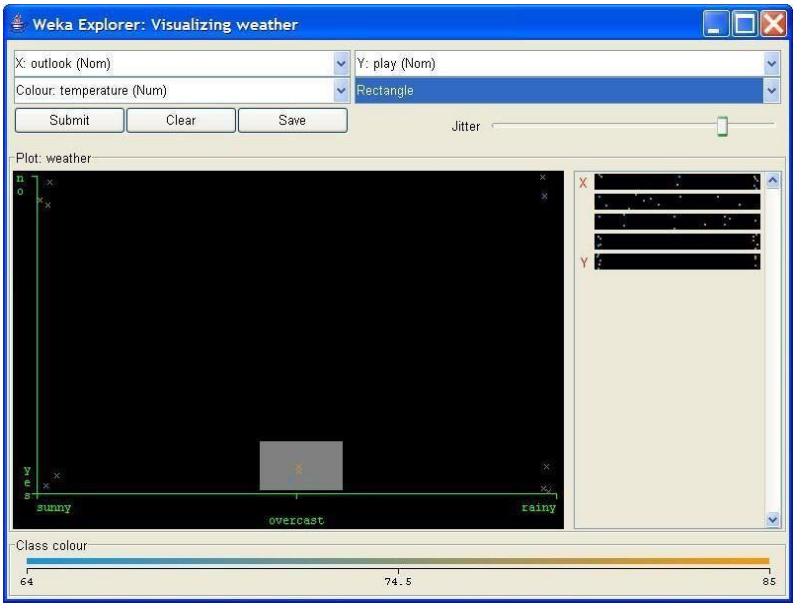
Selecting Instances

Sometimes it is helpful to select a subset of the data using visualization tool. A special case is the 'UserClassifier', which lets you to build your own classifier by interactively selecting instances. Below the Y – axis there is a drop-down list that allows you to choose a selection method. A group of points on the graph can be selected in four ways [2]:

1. **Select Instance.** Click on an individual data point. It brings up a window listing attributes of the point. If more than one point will appear at the same location, more than one set of attributes will be shown.

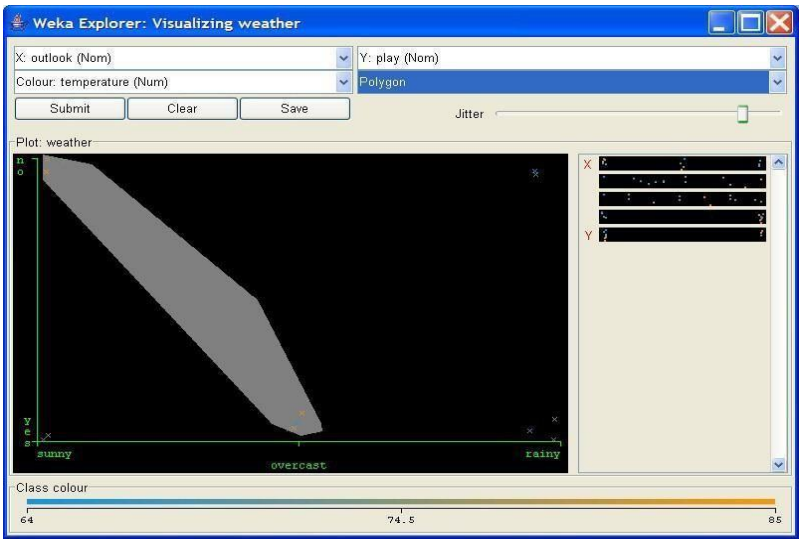


2. **Rectangle.** You can create a rectangle by dragging it around the point.

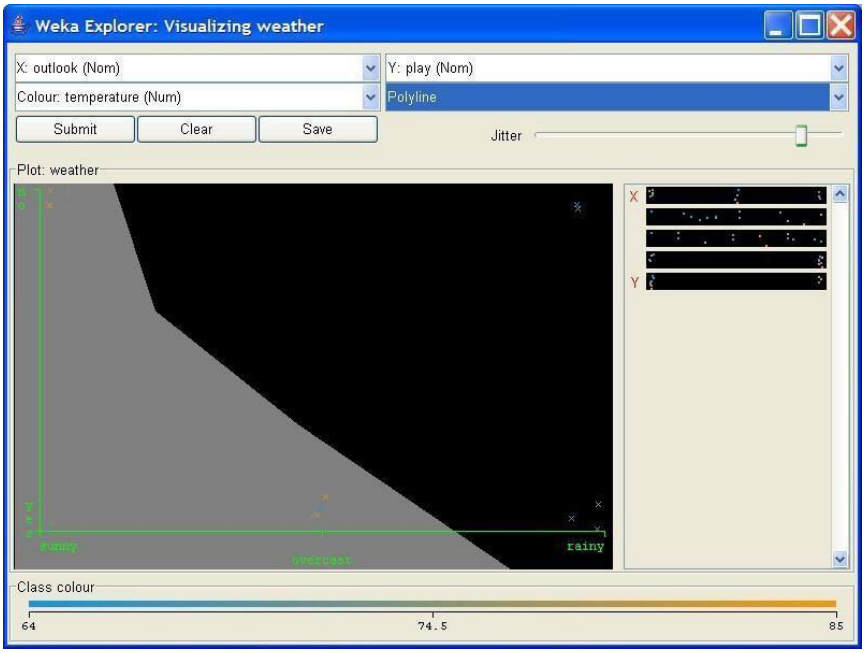


DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

3. **Polygon**. You can select several points by building a free-form polygon. Left-click on thegraph to add vertices to the polygon and right-click to complete it.



4. **Polyline**. To distinguish the points on one side from the once on another, you can build apolyline. Left-click on the graph to add vertices to the polyline and right-click to finish.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

B) Explore WEKA Data Mining/Machine Learning Toolkit.

1. Download the software as your requirements from the below given link. http://www.cs.waikato.ac.nz/ml/WEKA/downloading.html
2. The Java is mandatory for installation of WEKA so if you have already Java on your machine then download only WEKA else download the software with JVM.
3. Then open the file location and double click on the file

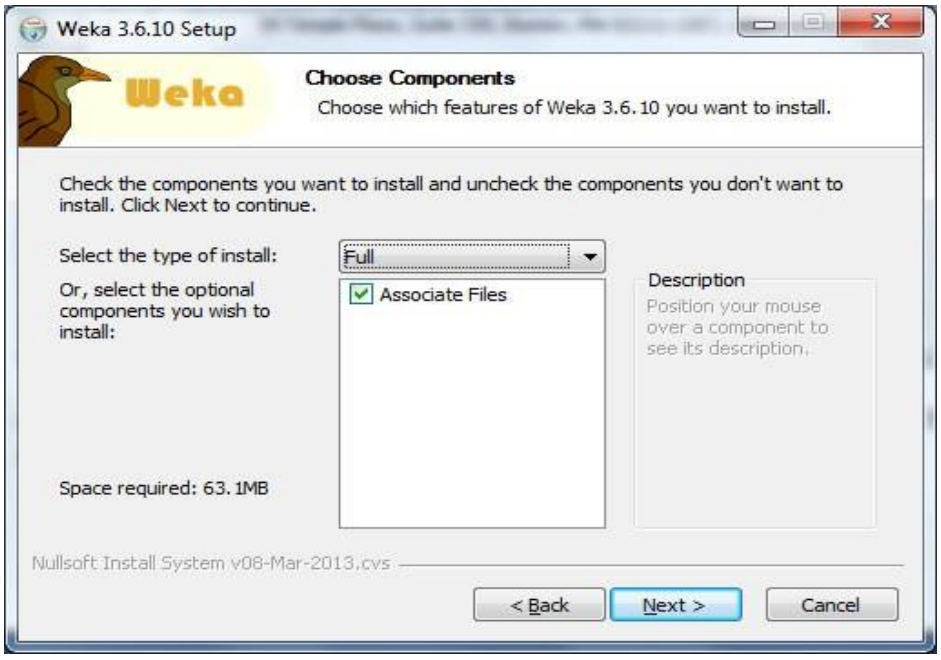
4. Click **Next**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

5. Click **I Agree.**

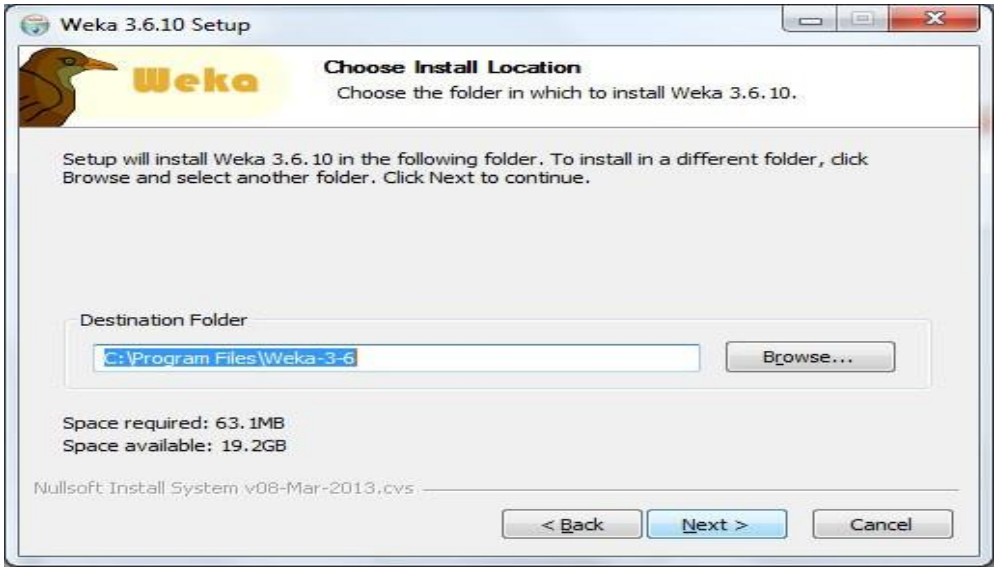
6. As your requirement do the necessary changes of settings and click **Next.**

Full and Associate files are the recommended settings

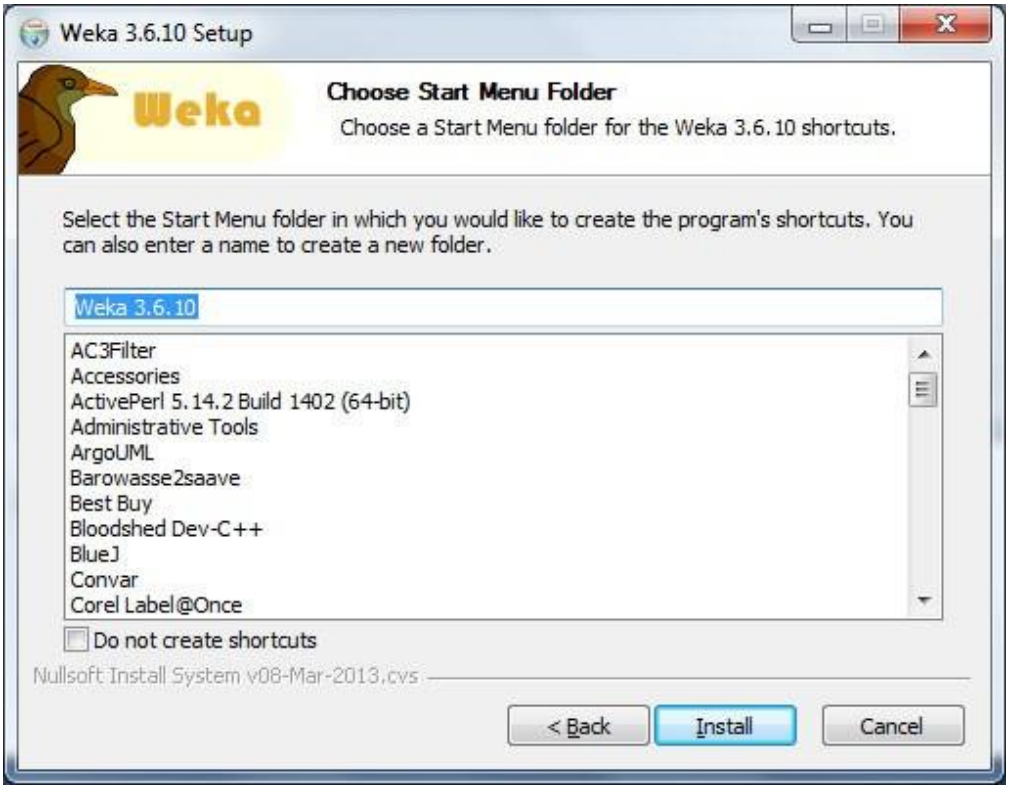


DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

7. Change to your desire installation location.

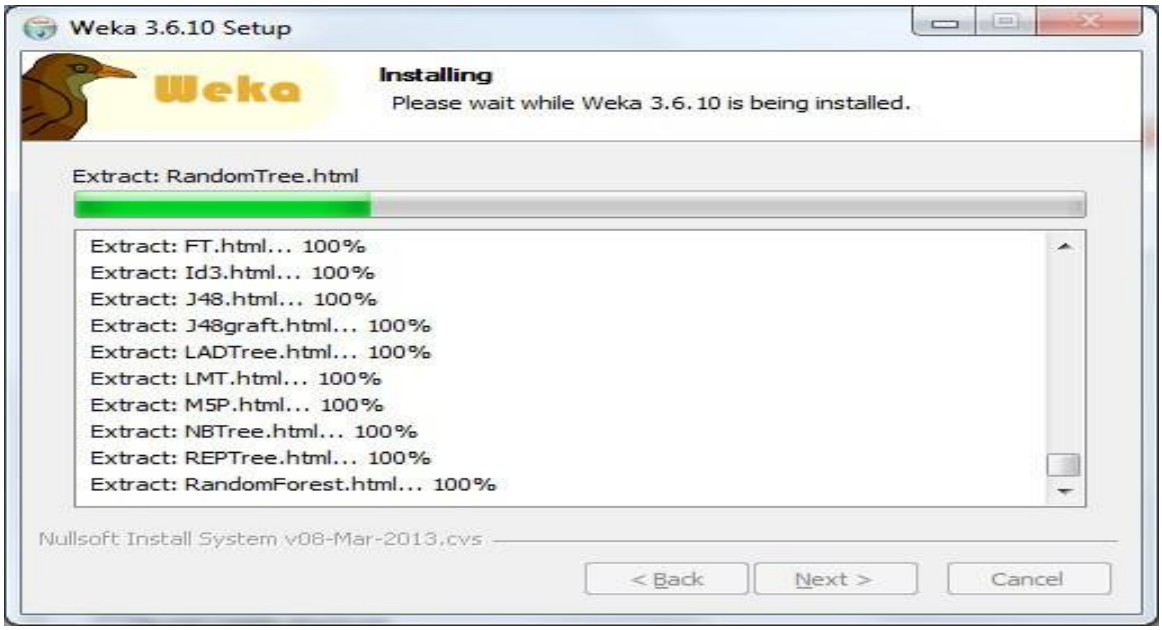


8. If you want a shortcut, then check the box and click **Install.**

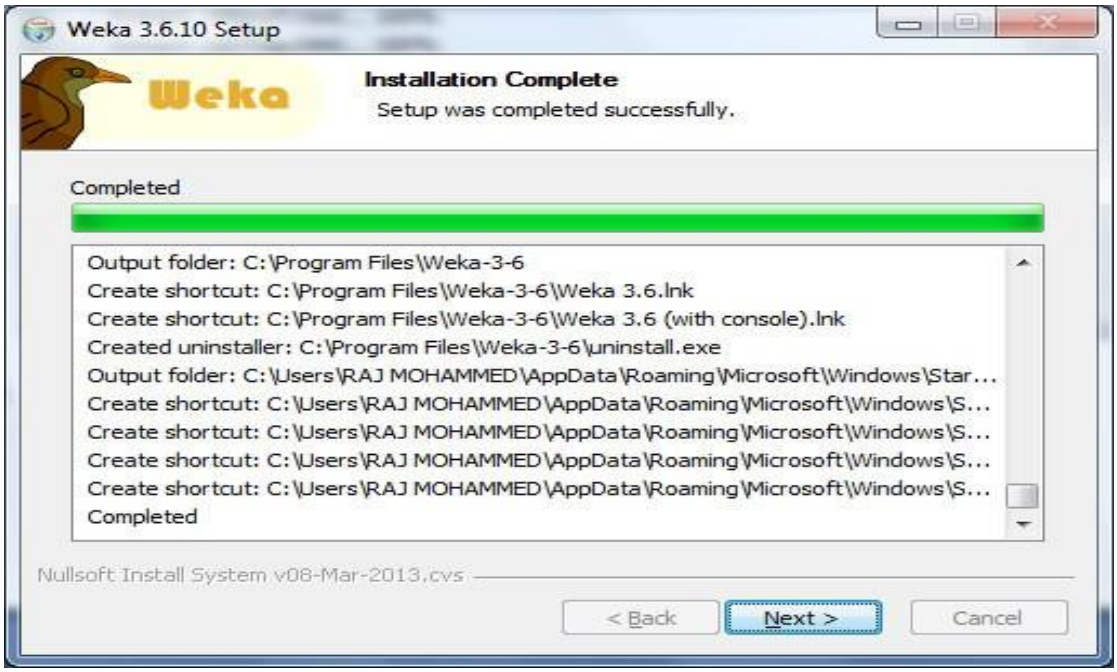


DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

9. The Installation will start wait for a while it will finish within a minute.



10. After complete installation, click on **Next.**



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

11. Click on the **Finish** and

12. Start working with WEKA tool for Data Mining.

This is the GUI you get when started. You have 4 options Explorer, Experimenter, Knowledge Flowand Simple CLI.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

- **Understand the features of WEKA tool kit such as Explorer, Knowledge flow interface, Experimenter, command-line interface.**

You will have 4 options in this WEKA Tool.

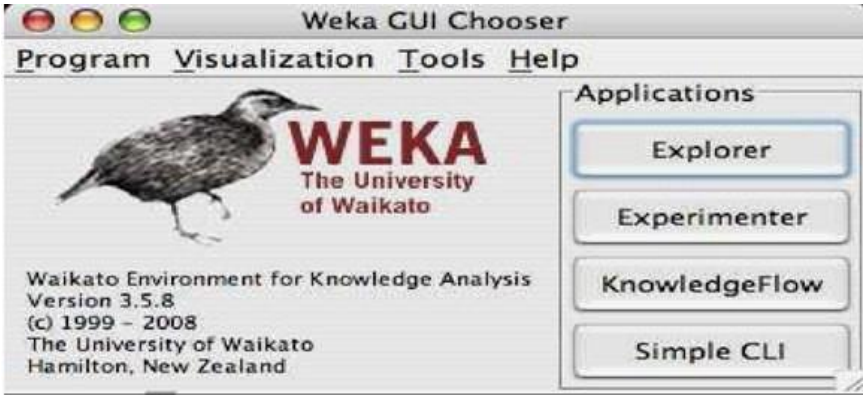
1. Explorer,
2. Experimenter,
3. Knowledge Flow and
4. Simple CLI.

WEKA ***(Waikato Environment for Knowledge Analysis)*** is created by researchers at the university WIKATO in New Zealand.

- It is java based application.
- It is collection often source, Machine Learning Algorithm.
- The routines (functions) are implemented as classes and logically arranged in packages.
- It comes with an extensive GUI Interface.
- WEKA routines can be used standalone via the command lineinterface.

The Graphical User Interface (GUI):

The WEKA GUI Chooser (class WEKA.gui.GUIChooser) provides a starting point for launching WEKA's main GUI applications and supporting tools. If one prefers a MDI ("multiple document interfaces") appearance, then this is provided by an alternative launcher called "Main", (class WEKA.gui.Main). The GUI Chooser consists of four buttons—one for each of the four major WEKA applications—and four menus.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

The buttons can be used to start the following applications:

- **Explorer:** An environment for exploring data with WEKA (the rest of this Documentation deals with this application in more detail).
- **Experimenter:** An environment for performing experiments and conducting statistical tests between learning schemes.
- **Knowledge Flow** This environment supports essentially the same functions as the Explorer but with a drag-and-drop interface. One advantage is that it supports incremental learning.
- **Simple CLI Provides** a simple command-line interface that allows direct execution of WEKA commands for operating systems that do not provide their own command line interface.

1. Explorer:

At the very top of the window, just below the title bar, is a row of tabs. When the Explorer is first started only the first tab is active; the others are grayed out. This is because it is necessary to open (and potentially pre-process) a data set before starting to explore the data.

The tabs are as follows:

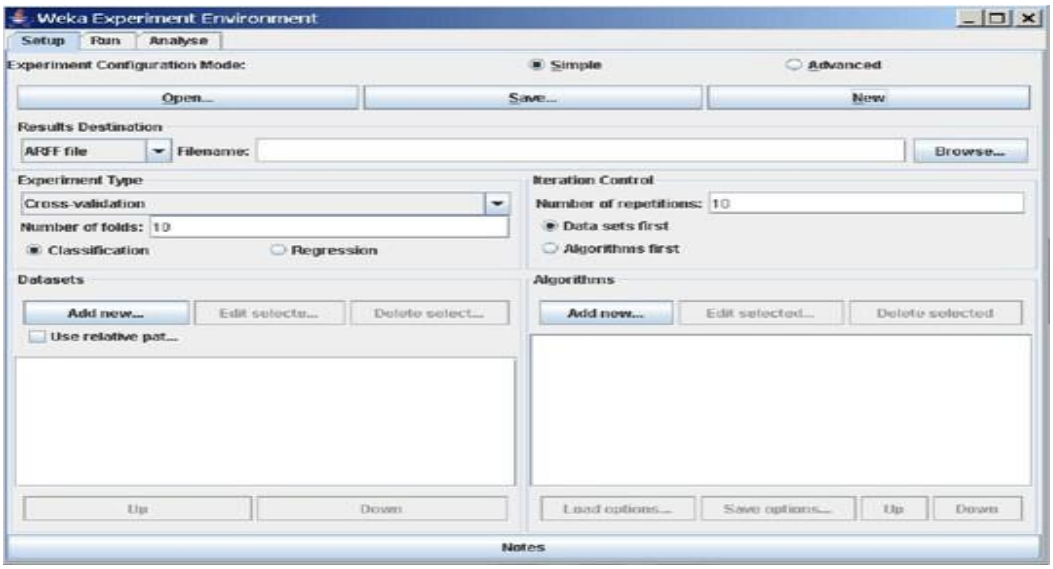
1. **Preprocess.** Choose and modify the data being acted on.
2. **Classify.** Train & test learning schemes that classify or perform regression
3. **Cluster.** Learn clusters for the data.
4. **Associate.** Learn association rules for the data.
5. **Select attributes.** Select the most relevant attributes in the data.
6. **Visualize.** View an interactive 2D plot of the data.

Once the tabs are active, clicking on them flicks between different screens, on which the respective actions can be performed. The bottom area of the window (including the status box, the log button, and the WEKA bird) stays visible regardless of which section you are in. The Explorer can be easily extended with custom tabs. The Wiki article **"Adding tabs in the Explorer" explains this in detail.**

2. EXPERIMENTER:

The WEKA Experiment Environment enables the user to create, run, modify, and analyze experiments

in a more convenient manner than is possible when processing the schemes individually. For example, the user can create an experiment that runs several schemes against a series of datasets and then analyze the results to determine if one of the schemes is (statistically) better than the other schemes.



The Experiment Environment can be run from the command line using the Simple CLI. For example, the following commands could be typed into the CLI to run the OneR scheme on the Iris dataset using a basic train and test process. (Note that the commands would be typed on one line into the CLI.)

While commands can be typed directly into the CLI, this technique is not particularly convenient and the experiments are not easy to modify. The Experimenter comes in two flavors', either with a simple interface that provides most of the functionality one needs for experiments, or with an interface with full access to the Experimenter's capabilities. You can choose between those two with the Experiment Configuration Mode radio buttons:

- Simple
- Advanced

Both setups allow you to setup standard experiments that are run locally on a single machine, or remote experiments, which are distributed between several hosts. The distribution of experiments cuts down the time the experiments will take until completion, but on the other hand the setup takes more time. The next section covers the standard experiments (both, simple and advanced), followed by the remote experiments and finally the analyzing of the results.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

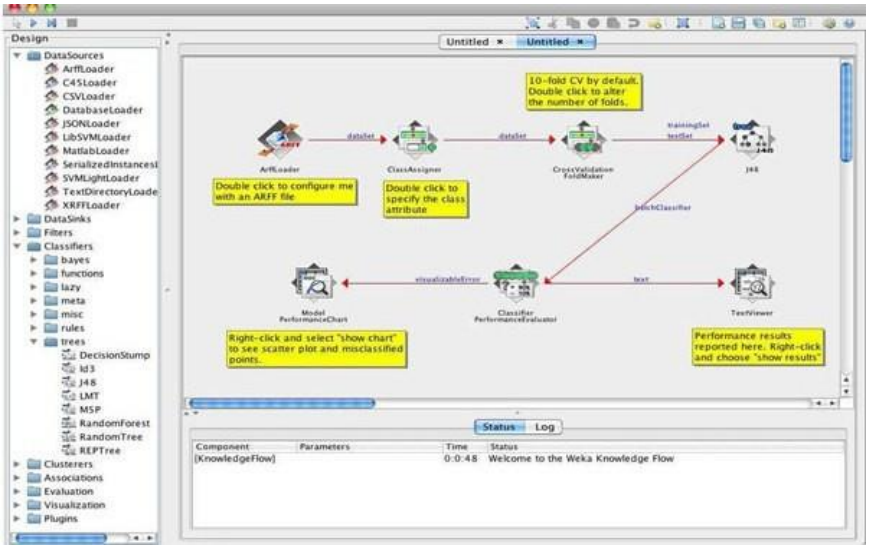
3. Knowledge Flow:

The Knowledge Flow provides an alternative to the Explorer as a graphical front end to WEKA's core algorithms. The Knowledge Flow presents a data-flow inspired interface to WEKA. The user can select WEKA components from a palette, place them on a layout canvas and connect them together in order to form a knowledge flow for processing and analyzing data.

At present, all of **WEKA's classifiers, filters, clusterers, associators, loaders and savers a**re available in the Knowledge Flow along with some extra tools. The Knowledge Flow can handle data either incrementally or in batches (the Explorer handles batch data only). Of course learning from data incrementally requires a classifier that can be updated on an instance by instance basis. Currently in WEKA there are ten classifiers that can handle data incrementally.

The Knowledge Flow offers the following features:

- ✓ **Intuitive** data flow style layout.
- ✓ **Process** data in batches or incrementally.
- ✓ **Process multiple batches** or streams in parallel (each separate flow executes in its own thread).
- ✓ **Process multiple streams sequentially** via a user-specified order of execution.
- ✓ **View models** produced by classifiers for each fold in a crossvalidation.
- ✓ **Visualize performance** of incremental classifiers during processing.
- ✓ **Plugin "perspectives"** that add major new functionality.
- ✓ **Chain filters** together**.**



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

4. **Simple CLI**:

The Simple CLI provides full access to all WEKA classes, i.e., classifiers, filters, clusterers, etc., but without the hassle of the CLASSPATH (it facilitates the one, with which WEKA was started). It offers a simple WEKA shell with separated command line and output.

Commands:

The following commands are available in the Simple CLI:

Sl. No	Command	Description
1	java <classname><args>]	Invokes a java class with the given arguments (if any).
2	break	Stops the current thread, e.g., a running classifier, in a friendly manner kill stops the current thread in an unfriendly fashion.
3	cls	Clears the output area.
4	capabilities <classname>[<args>]	Lists the capabilities of the specified class, e.g., for a classifier with its option: **Capabilities WEKA.classifiers.meta.Bagging -W WEKA.classifiers.trees.Id3**
5	exit	Exits the Simple CLI
6	help[<command>]	Provides an overview of the available commands if without a command name as argument, otherwise more help on the specified command

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Invocation:

In order to invoke a WEKA class, one has only to prefix the class with ***"java".*** This command tells the Simple CLI to load a class and execute it with any given parameters. E.g., the J48 classifier can be invoked on the iris dataset with the following command: **java WEKA.classifiers.trees.J48 -t c:/temp/iris.arff**
This results in the following output:

Command redirection:

Starting with this version of WEKA one can perform a basic redirection:

java WEKA.classifiers.trees.J48 -t test.arff > j48.txt

Note: The **'>'** must be preceded and followed by a space, otherwise it is not recognized as redirection, but part of another parameter.

Command completion:

Commands starting with java support completion for classnames and filenames via Tab (Alt+BackSpace deletes parts of the command again). In case that there are several matches, WEKA lists all possible matches.

- **Package Name Completion javaWEKA.cl<Tab> r**esults in the following output of possible matches of package names:
  - ✓ WEKA.classifiers
  - ✓ WEKA.clusterers
- **Classname completion**
  - ✓ **java WEKA.classifiers.meta.A<Tab>** lists the following classes.
  - ✓ WEKA.classifiers.meta.AdaBoostM1 WEKA.classifiers.meta.AdditiveRegression
  - ✓ WEKA.classifiers.meta.AttributeSelectedClassifier
- **Filename Completion**

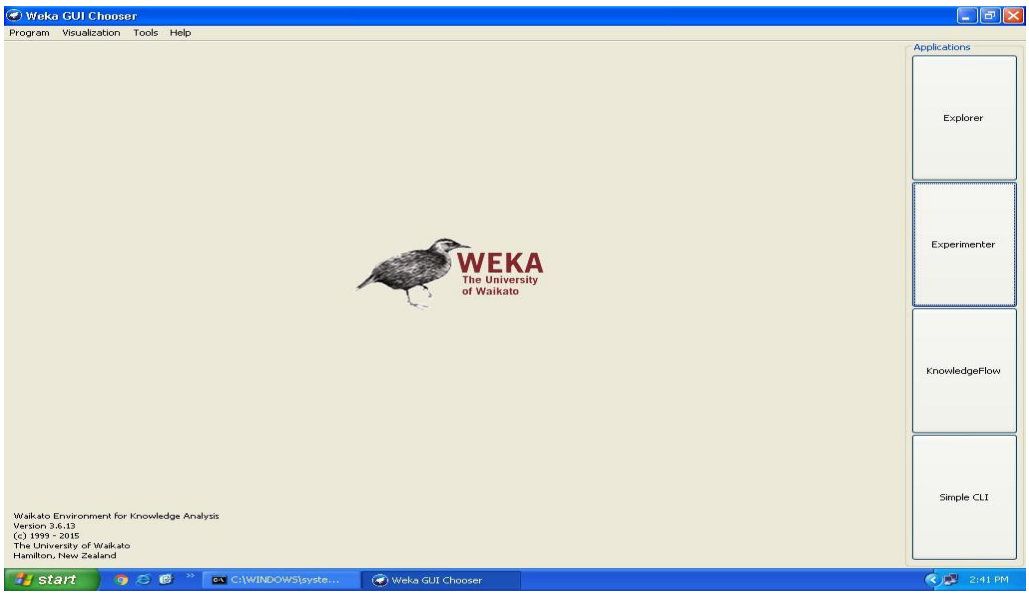
In order for WEKA to determine whether a the string under the cursor is a classname or a filename, filenames need to be absolute (Unix/Linx: /some/path/file;Windows: C:\Some\Path\file) or relative and starting with a dot (Unix/Linux:./some/other/path/file; Windows:.\Some\Other\Path\file).

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

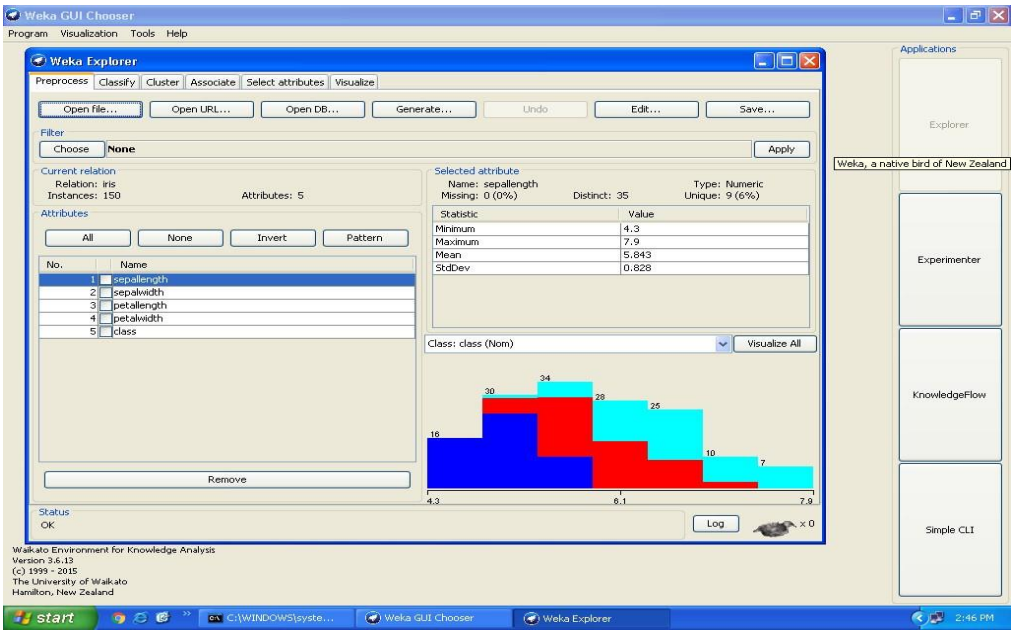
B.(iii)Navigate the options available in the WEKA (ex: select attributes panel, preprocess panel, classify panel, cluster panel, associate panel and visualize)

Ans: Steps for identify options in WEKA

1. Open WEKA Tool.
2. Click on WEKA Explorer.
3. Click on Preprocessing tab button.
4. Click on open file button.
5. Choose WEKA folder in C drive.
6. Select and Click on data option button
7. Choose iris data set and open file.
8. All tabs available in WEKA homepage.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE



(iv) Study the ARFF file format:

Ans: ARFF File Format

An ARFF (Attribute-Relation File Format) file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files are not the only format one can load, but all files that can be converted with WEKA's ***"core converters".*** The following formats are currently supported:

- ARFF (+compressed)
- C4.5
- CSV
- libsvm
- binary serialized instances
- XRFF (+compressed)

Overview

ARFF files have two distinct sections. The first section is the **Header** information, which is followed the **Data** information. The Header of the ARFF file contains the name of the relation, a list of the attributes (the columns in the data), and their types.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

An example header on the standard IRIS dataset looks like this:

1. *Title: Iris Plants Database*
2. **Sources:**
   a. Creator: R.A.Fisher
   b. Donor: Michael Marshall(MARSHALL%PLU@io.arc.nasa.gov)
   c. Date: July,1988

```
@RELATION iris
@ATTRIBUTE sepal length NUMERIC
@ATTRIBUTE sepal width NUMERIC
@ATTRIBUTE petal length NUMERIC
@ATTRIBUTE petal width NUMERIC
@ATTRIBUTE class {Iris-setosa, Iris-versicolor, Iris-irginica}
//The Data of the ARFF file looks like the following:
@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
```

- Note: Lines that begin with a % are comments.
- The @RELATION, @ATTRIBUTE and @DATA declarations are case insensitive.

The ARFF Header Section

The ARFF Header section of the file contains the relation declaration and at• tribute declarations.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

The @relation Declaration

The relation name is defined as the first line in the ARFF file. The format is**: @relation <relation-name>**, where <relation-name> is a string. The string must be quoted if the name includes spaces.

The @attribute Declarations

Attribute declarations take the form of an ordered sequence of @attribute statements. Each attribute in the data set has its own @attribute statement which uniquely defines the name of that attribute and its data type. The order the attributes are declared indicates the column position in the data section of the file. For example, if an attribute is the third one declared then WEKA expects that all that attributes values will be found in the third comma delimited column.

The format for the @attribute statement is: @attribute <attribute-name><datatype>, where the <attribute-name> must start with an alphabetic character. If spaces are to be included in the name, then the entire name must be quoted.

The <datatype> can be any of the four types supported by WEKA:

1. numeric
2. integer is treated as numeric
3. real is treated as numeric
4. <nominal-specification>
5. string
6. date [<date-format>]
7. relational for multi-instance data (for future use)

where <nominal-specification> and <date-format> are defined below. The keywords numeric, real, integer, string and date are case insensitive.

Numeric attributes:

Numeric attributes can be real or integer numbers.

Nominal attributes:

Nominal values are defined by providing an <nominal-specification> listing the possible values: <nominal-name1>, <nominal-name2>, <nominal-name3>, For example, the class value of the Iris dataset can be defined as follows: @ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica} Values that contain spaces must be quoted.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

String attributes:

String attributes allow us to create attributes containing arbitrary textual values. This is very useful in text-mining applications, as we can create datasets with string attributes, then write WEKA Filters to manipulate strings (like String- ToWordVectorFilter). String attributes are declared as follows:

@ATTRIBUTE LCC string

Date attributes:

Date attribute declarations take the form: @attribute <name> date [<date-format>] where <name> is the name for the attribute and <date-format> is an optional string specifying how date values should be parsed and printed (this is the same format used by Simple Date Format). The default format string accepts the ISO-8601 combined date and time format: yyyy-MM-dd'T'HH:mm:ss. Dates must be specified in the data section as the corresponding string representations of the date/time (see example below).

Relational attributes:

Relational attribute declarations take the form:

@attribute <name> relational <further attribute definitions>

@end <name> For the multi-instance dataset MUSK1 the definition would look like this (" ..." denotes an omission):

@attribute molecule_name {MUSK-jf78..., NON-MUSK-199}

@attribute bag relational @attribute f1 numeric

...

@attribute f166 numeric @end bag @attribute class {0,1}

- ✓ **The ARFF Data Section:** The ARFF Data section of the file contains the data declaration line and the actual instance lines.
- ✓ **The @data Declaration:** The @data declaration is a single line denoting the start of the data segment in the file. The format is: **@data The instance data**

Each instance is represented on a single line, with carriage returns denoting the end of the instance. A percent sign (%) introduces a comment, which continues to the end of the line. Attribute values for each instance are delimited by commas. They must appear in the order that they were declared in the header section (i.e. the data corresponding to the nth @attribute declaration is always the nth field of the attribute).

Missing values are represented by a single question mark, as in: @data 4.4,?,1.5,?,Iris-setosa

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Values of string and nominal attributes are case sensitive, and any that contain space or the comment-delimiter character % must be quoted.

✓ **string**

An example follows: @relation LCCvsLCSH @attribute LCC string @attribute LCSH

✓ **@data**

AG5, 'Encyclopedias and dictionaries.; Twentieth century.'

AS262, 'Science -- Soviet Union -- History.'

AE5, 'Encyclopedias and dictionaries.'

AS281, 'Astronomy, Assyro-Babylonian.;Moon -- Phases.'

AS281, 'Astronomy, Assyro-Babylonian.;Moon -- Tables.'

Dates must be specified in the data section using the string representation specified in the attribute declaration.

For example:

@RELATION Timestamps

@ATTRIBUTE timestamp DATE "yyyy-MM-dd HH:mm:ss"

@DATA

"2001-04-03 12:12:12"

"2001-05-03 12:59:55"

Relational data must be enclosed within double quotes(“). For example an instance of the MUSK1 dataset (”...” denotes an omission): MUSK-188,"42,...,30",1

B.(v) Explore the available data sets in WEKA.

Steps for identifying data sets in WEKA

1. Open WEKA Tool.
2. Click on WEKA Explorer.
3. Click on open file button.
4. Choose WEKA folder in C drive.
5. Select and Click on data option button.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 5: Implementation of Apriori algorithm

Association rule mining is defined as: Let be a set of n binary attributes called items. Let be a set of transactions called the database. Each transaction in D has a unique transaction ID and contains a subset of the items in I. A rule is defined as an implication of the form X=>Y where X,Y C I and X Π Y=Φ . The sets of items (for short item sets) X and Y are called antecedent (left hand sideor LHS) and consequent (right hand side or RHS) of the rule respectively. To illustrate the concepts, we use a small example from the supermarket domain.

The set of items is I = {milk, bread, butter, beer} and a small database containing the items (1 codes presence and 0 absence of an item in a transaction) is shown in the table to the right. An example rule for the supermarket could be meaning that if milk and bread is bought, customers also buy butter.

To select interesting rules from the set of all possible rules, constraints on various measures of significance and interest can be used. The best known constraints are minimum thresholds on support and confidence. The support supp(X) of an itemset X is defined as the proportion of transactions in the data set which contain the itemset. In the example database, the itemset {milk, bread} has a support of 2 / 5 = 0.4 since it occurs in 40% of all transactions (2 out of 5 transactions).

The confidence of a rule is defined. For example, the rule has a confidence of 0.2 / 0.4 = 0.5 in the database, which means that for 50% of the transactions containing milk and bread the rule is correct. Confidence can be interpreted as an estimate of the probability P(Y | X), the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.

ALGORITHM:

Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two sub problems. One is to find those itemsets whose occurrences exceed a predefined threshold in the database; those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence.

Suppose one of the large itemsets is Lk, Lk = {I1, I2, … , Ik}, association rules with this itemsets are generated in the following way: the first rule is {I1, I2, … , Ik1} and {Ik}, by checking the confidence this rule can be determined as interesting or not.

Then other rule are generated by deleting the last items in the antecedent and inserting it to the consequent, further the confidences of the new rules are checked to determine the interestingness of them. Those processes iterated until the antecedent becomes empty. Since the second sub problem is quite straight forward, most of the researches focus on the first sub problem.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

The Apriori algorithm finds the frequent sets *L* In Database *D*.Find frequent set *Lk* − 1.

Join Step.

. *Ck* is generated by joining *Lk* − 1with itself

Prune Step.

o Any (*k* − 1) itemset that is not frequent cannot be a subset of a frequent *k* itemset, hence shouldbe removed.

Where · (*Ck*: Candidate itemset of size *k*)

· (*Lk*: frequent itemset of size *k*) Apriori Pseudocode

Apriori (T,£)

L<{ Large 1itemsets that appear in more than transactions }

while L(k1)≠ Φ *C(k)<Generate*(*Lk* − 1) *for transactions* t € TC(t)Subset(*Ck*,*t*)

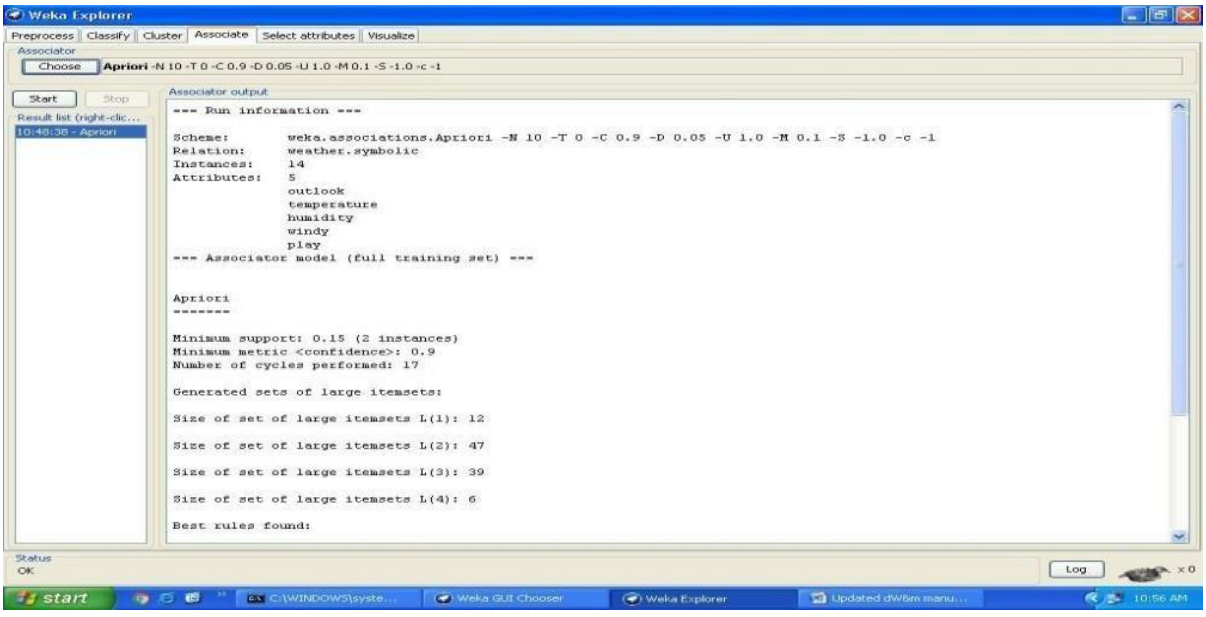
for candidates c € C(t)

count[c]<count[c]+1 L(k)<{ c

€ C(k)| count[c] ≥ *£ K<K+ 1 return* Ụ L(k) k.

Steps for run Apriori algorithm in WEKA:

- Open WEKA Tool.
- Click on WEKA Explorer.
  - Click on Preprocessing tab button.
  - Click on open file button.
  - Choose WEKA folder in C drive.
- Select and Click on data option button.
- Choose Weather data set and open file.
- Click on Associate tab and Choose Apriori algorithm
- Click on start button.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Association Rule:

An association rule has two parts, an antecedent (if) and a consequent (then). An antecedent is an item found in the data. A consequent is an item that is found in combination with the antecedent. Association rules are created by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships.

Support is an indication of how frequently the items appear in the database. Confidence indicates the number of times the if/then statements have been found to be true. In data mining, association rules are useful for analyzing and predicting customer behavior. They playan important part in shopping basket data analysis, product clustering, catalog design and store layout.

Support and Confidence values:

- Support count: The support count of an itemset *X*, denoted by *X.count*, in a data set *T* is the number of transactions in *T* that contain *X*. Assume *T* has *n* transactions.
- Then,

$$support = \frac{(X \cup Y).count}{n}$$

$$confidence = \frac{(X \cup Y).count}{X.count}$$

support = support({*A* U *C*})

confidence = support({*A* U *C*})/support({*A*})

Exercise 5:

Apply the Apriori algorithm on Airport noise monitoring data set, Discriminating between patients with Parkinsons and neurological diseases using voice recordings dataset. [https://archive.ics.uci.edu/ml/machine-learning-databases/00000/ refer this link for datasets]

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 6: Implementation of FP-Growth algorithm

By using the FP-Growth method, the number of scans of the entire database can be reduced to two. The algorithm extracts frequent item sets that can be used to extract association rules. This is done using the support of an item set. The main idea of the algorithm is to use a divide and conquer strategy:

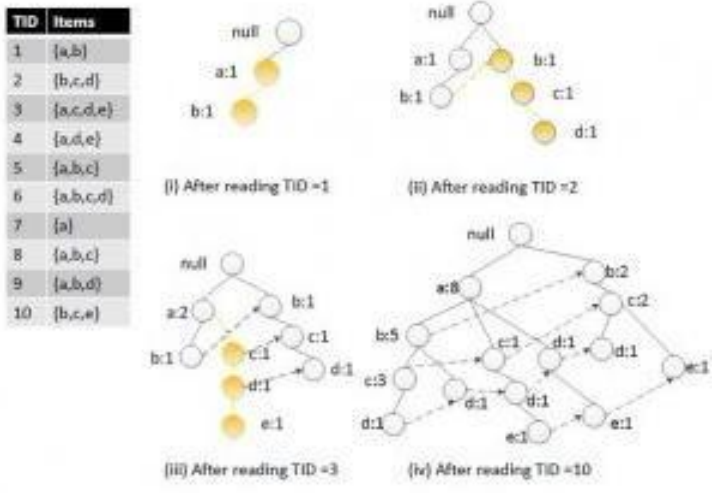
Compress the database which provides the frequent sets; then divide this compressed database into a set of conditional databases, each associated with a frequent set and apply data mining on each database.

To compress the data source, a special data structure called the FP-Tree is needed [26]. The tree is used for the data mining part. Finally the algorithm works in two steps:

1. **Construction of the FP-Tree**
2. **Extract frequent item sets**

1. Construction of the FP-Tree

The FP-Tree is a compressed representation of the input. While reading the data source each transaction t is mapped to a path in the FP-Tree. As different transaction can have several items in common, their path may overlap. With this it is possible to compress the structure.



The below figure shows an example for the generation of an FP-tree using 10 transactions.

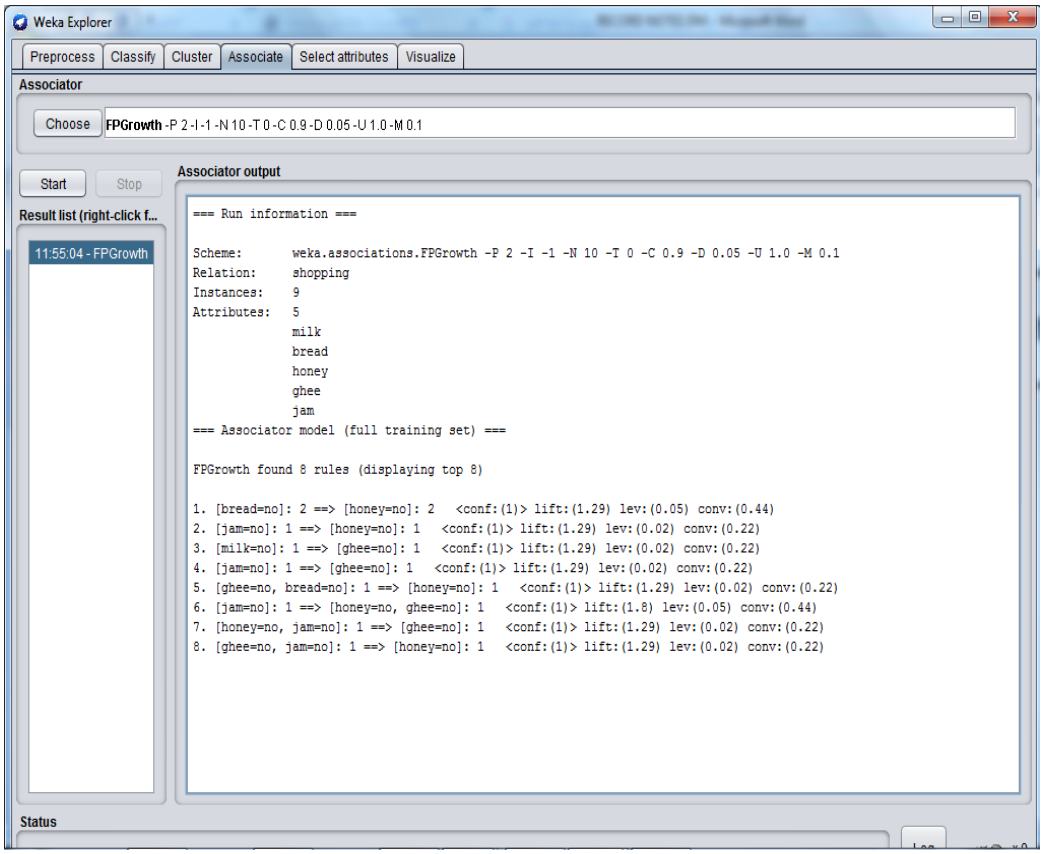
2. Extract frequent item sets

A bottom-up strategy starts with the leaves and moves up to the root using a divide and conquer strategy. Because every transaction is mapped on a path in the FP-Tree, it is possible to mine frequent item sets ending in a particular item.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Steps:

- Open WEKA Tool.
- Click on WEKA Explorer.
  - Click on Preprocessing tab button.
  - Click on open file button.
  - Choose WEKA folder in C drive.
- Select and Click on data option button.
- Choose a data set and open file.
- Click on Associate tab and Choose FP-Growth algorithm
- Click on start button.



Exercise 6:

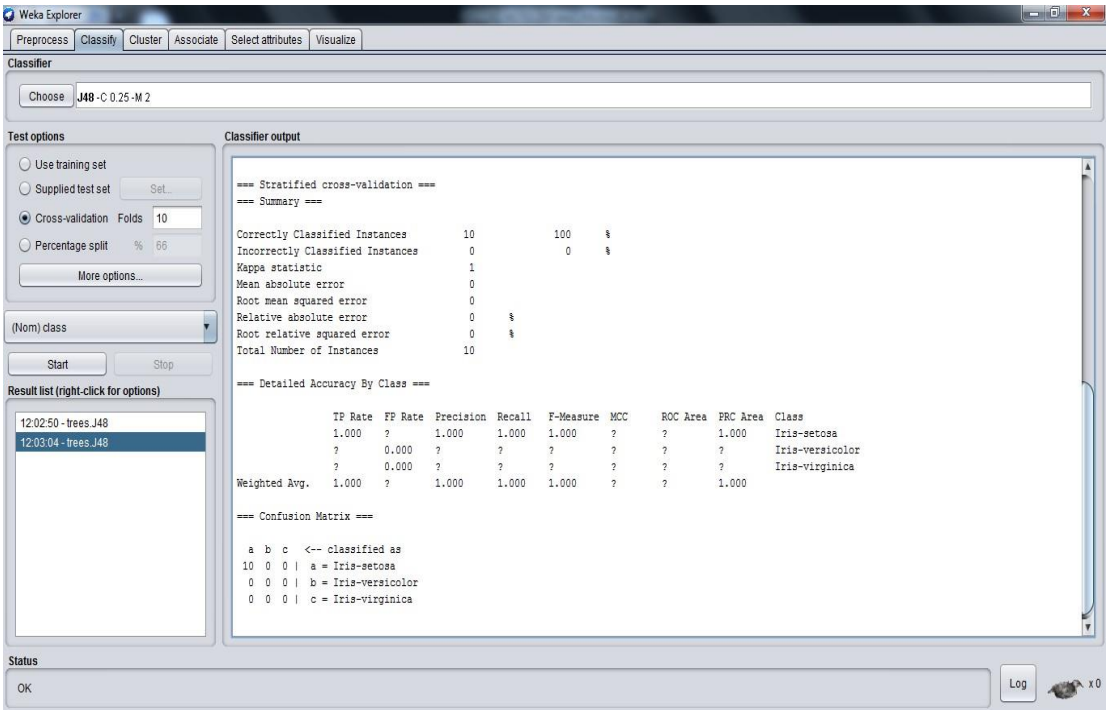
Apply FP-Growth algorithm on Blood Transfusion Service Center data set

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 7: Implementation of Decision Tree InductionSteps to model decision tree.

1. Double click on **credit-g.arff** file.
2. Consider all the **21** attributes for making decision tree.
3. Click on classify tab.
4. Click on choose button.
5. Expand tree folder and select J48
6. Click on use training set in test options.
7. Click on start button.
8. Right click on result list and choose the visualize tree to get decision tree.

We created a decision tree by using J48 Technique for the complete dataset as the training data.The following model obtained after training.



Exercise 7:

Apply decision tree algorithm to book a table in a hotel/ book a train ticket/ movie ticket**.**

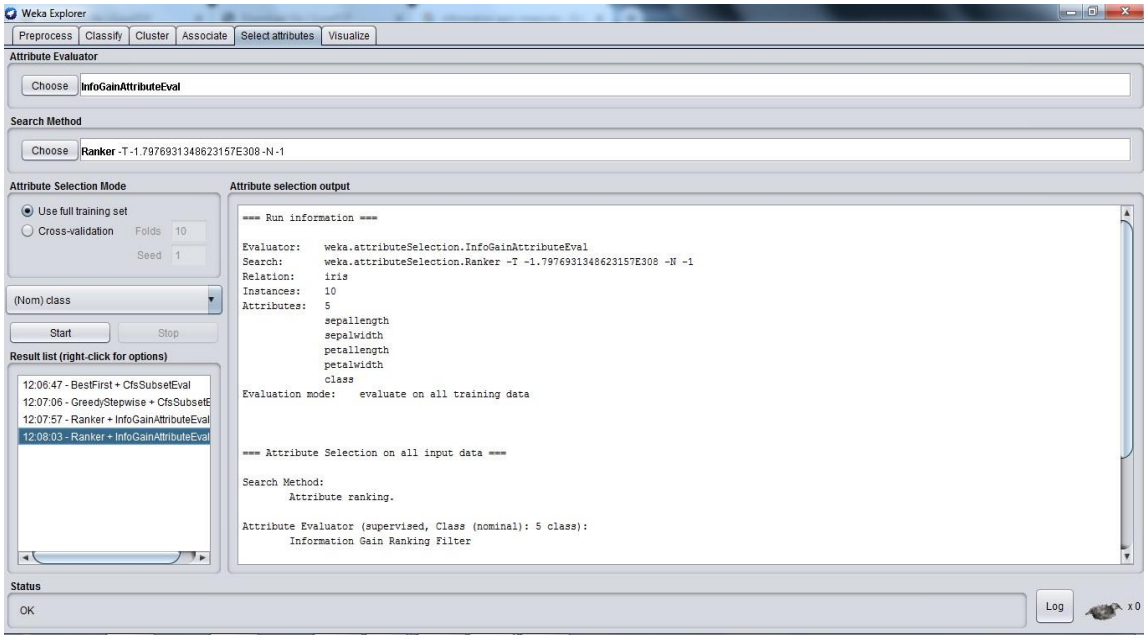
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 8: calculating information gain measures.

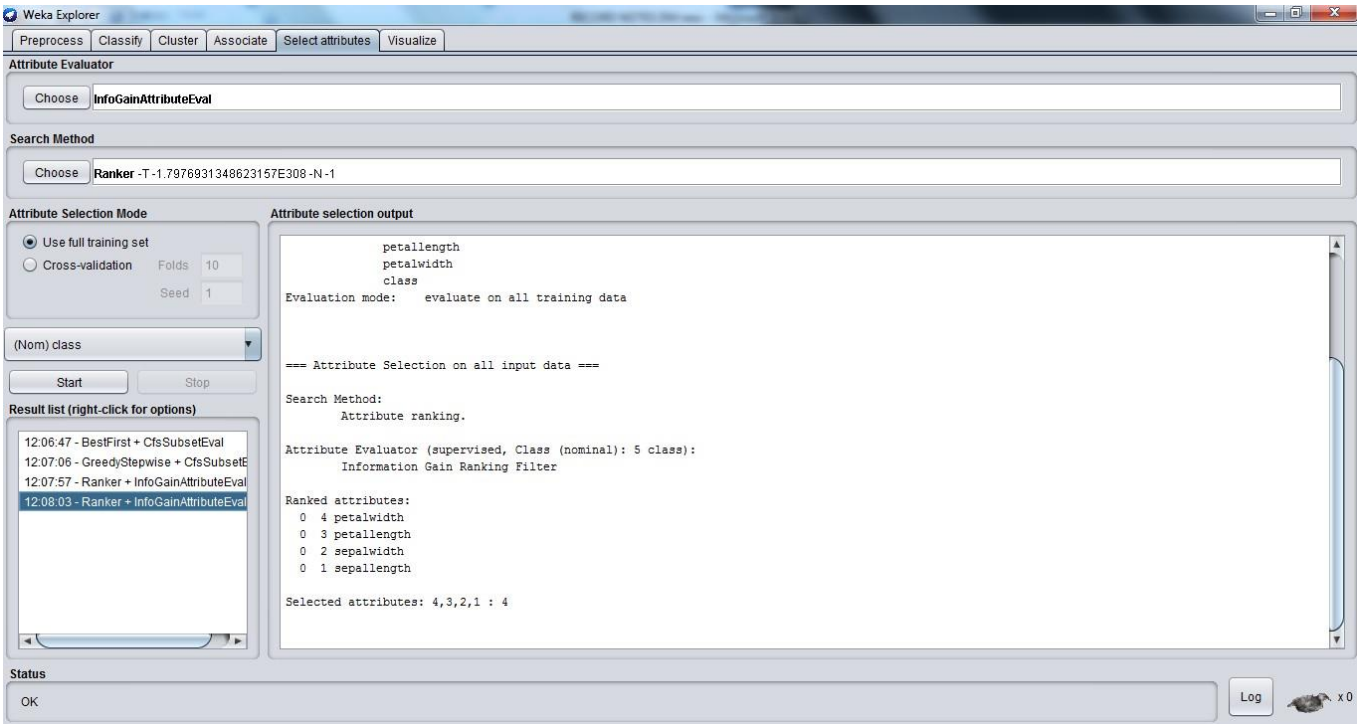
Information gain (IG) measures how much "information" a feature gives us about the class. – Features that perfectly partition should give maximal information. – Unrelated features should give no information. It measures the reduction in entropy. CfsSubsetEval aims to identify a subset of attributes that are highly correlated with the target while not being strongly correlated with one another. It searches through the space of possible attribute subsets for the "best" one using the BestFirst search method by default, although other methods can be chosen. To use the wrapper method rather than a filter method, such as CfsSubsetEval, first select WrapperSubsetEval and then configure it by choosing a learning algorithm to apply and setting the number of cross-validation folds to use when evaluating it on each attribute subset.

Steps:

- Open WEKA Tool.
- Click on WEKA Explorer.
  - Click on Preprocessing tab button.
  - Click on open file button.
- Select and Click on data option button.
- Choose a data set and open file.
- Click on select attribute tab and Choose attribute evaluator, search method algorithm
- Click on start button.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE



Exercise 8:

Calculate the information gain on weather data set(for each attributes separately).

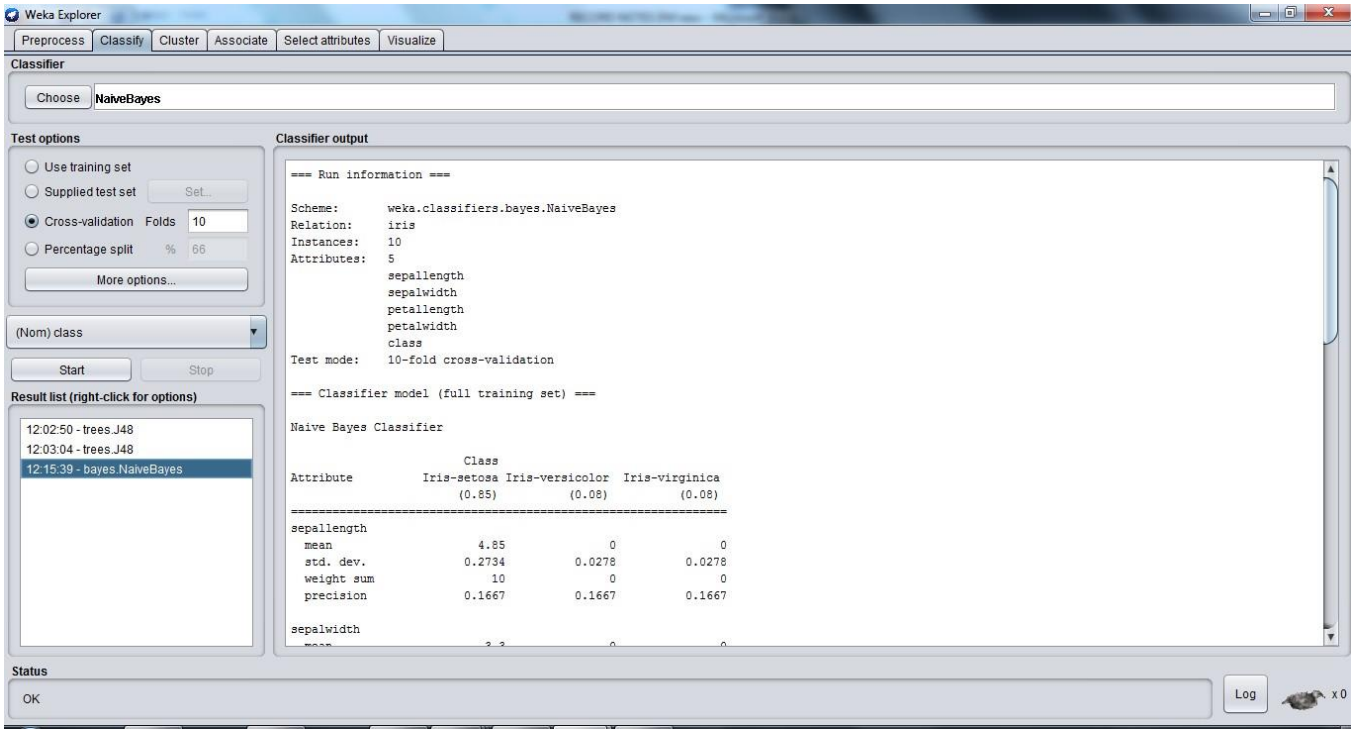
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 9: classification of data using Bayesian approach

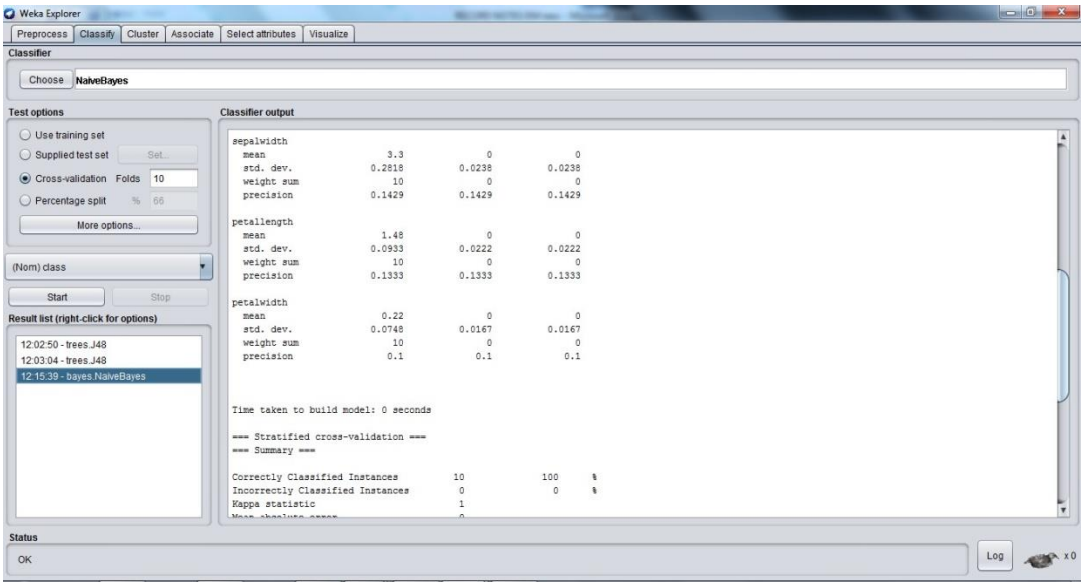
Bayesian classification is based on Bayes' Theorem. Bayesian classifiers are the statistical classifiers. Bayesian classifiers can predict class membership probabilities such as the probability that a given tuple belongs to a particular class.

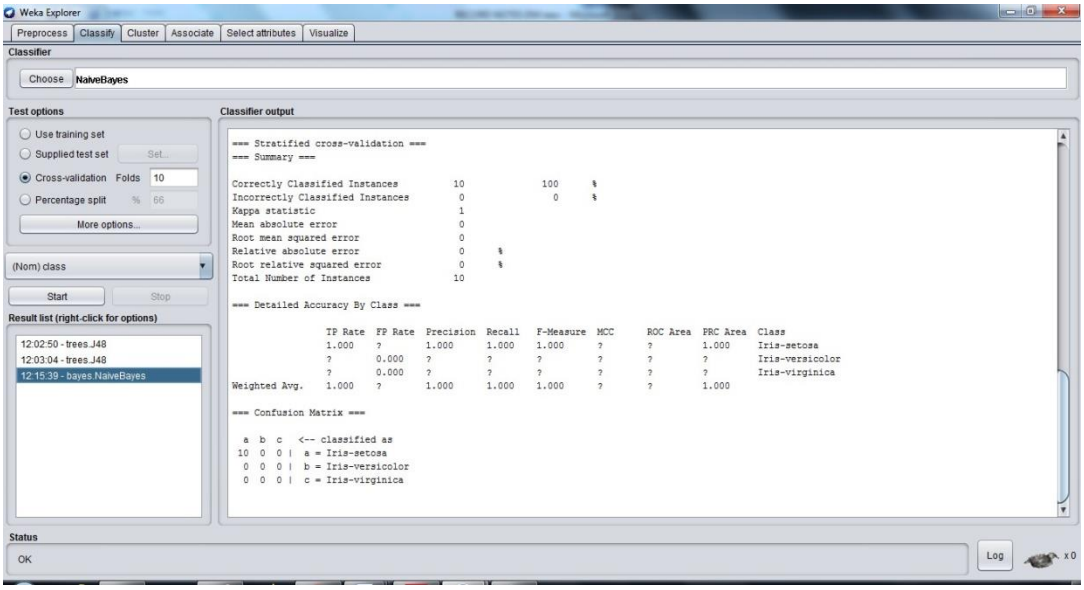
Steps:

1. Open WEKA Tool.
2. Click on WEKA Explorer.
3. Click on Preprocessing tab button.
4. Click on open file button.
5. Choose WEKA folder in C drive.
6. Select and Click on data option button.
7. Choose a data set and open file.
8. Click on classify tab and Choose Naïve-bayes algorithm and select use training set test option.'
9. Click on start button.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE





Exercise 9

Classify data (lung cancer/ diabetes/liver disorder) using Bayesian approach

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

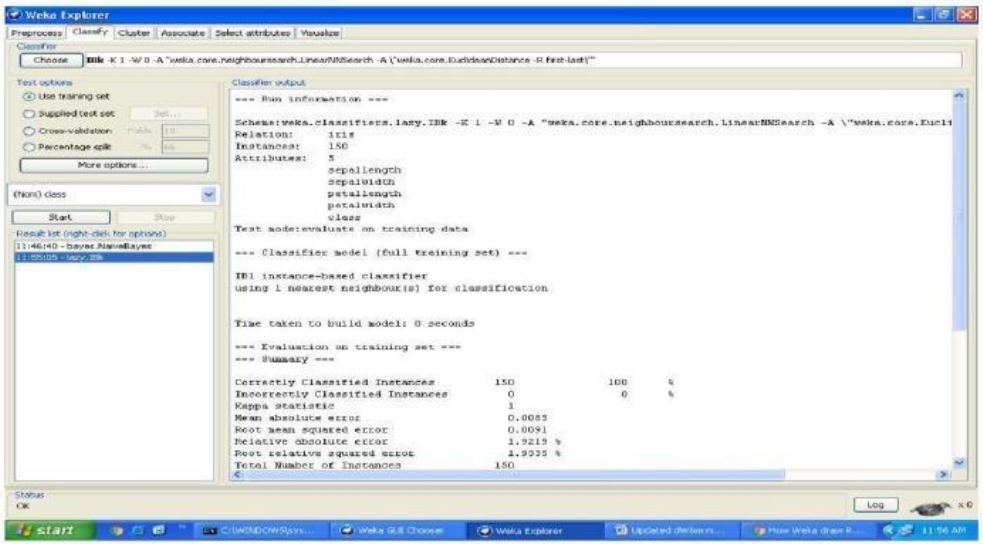
Experiment 10: classification of data using K-nearest neighbor approach

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems. However, it is mainly used for classification predictive problems in industry. The following two properties would define KNN well −

- **Lazy learning algorithm** − KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification.
- **Non-parametric learning algorithm** − KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data.

Steps:

1. Open WEKA Tool.
2. Click on WEKA Explorer.
3. Click on Preprocessing tab button.
4. Click on open file button.
5. Choose WEKA folder in C drive.
6. Select and Click on data option button.
7. Choose a data set and open file.
8. Click on classify tab and Choose k-nearest neighbor and select use training set test option.
9. Click on start button.



Exercise 10: Perform analysis on iris data set and build cluster using K-nearest neighbor approach

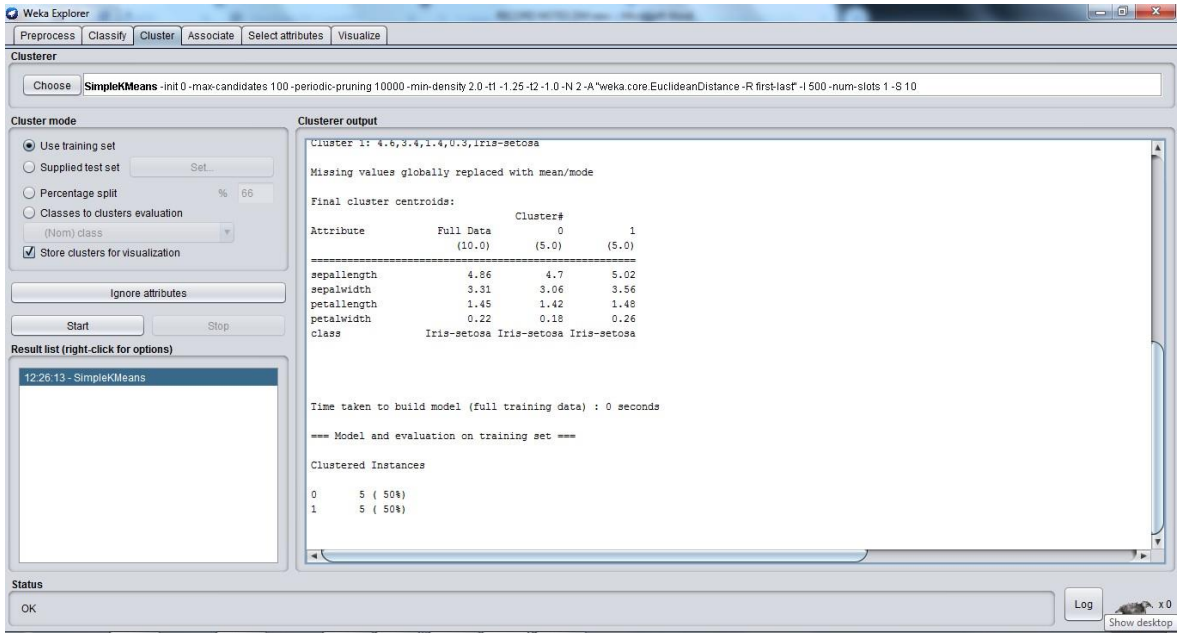
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment 11: implementation of K-means algorithm

K-means algorithm is an iterative algorithm that tries to partition the dataset into ***K*** pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. It tries to make the inter-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

Steps:

1. Open WEKA Tool.
2. Click on WEKA Explorer.
3. Click on Preprocessing tab button.
4. Click on open file button.
5. Choose WEKA folder in C drive.
6. Select and Click on data option button.
7. Choose a data set and open file.
8. Click on cluster tab and Choose k-means algorithm.
9. Click on start button.



Exercise 11: Implement of K-means clustering using crime dataset.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment-12: Partitioning - Horizontal, Vertical, Round Robin, Hash based

Partitioning is needed to

- ✓ Improve scalability.
- ✓ Improve performance.
- ✓ Improve security.
- ✓ Provide operational flexibility.
- ✓ Matches the data store to the pattern of use.
- ✓ Improve availability.

Designing partitions

There are Four typical strategies for partitioning data:

- **Horizontal partitioning (often called sharding).** In this strategy, each partition is a separate data store, but all partitions have the same schema. Here, each partition is known as a shard and holds a specific subset of the data, such as all the orders for a specific set of customers.
- **Vertical partitioning.** In this strategy, each partition holds a subset of the fields for items in the data store. The fields are divided according to their pattern of use.
- **Round-Robin Partitioning:**
  - Data is distributed evenly among all partitions.
  - This partitioning is used where the number of rows to process in each partition are approximately same
- **Hash Portioning:**
  - Hash function is applied for the purpose of partitioning keys to group data among partitions.
  - It is used where ensuring the processes groups of rows with the same partitioning key in the same partition, need to be ensured.

a). Horizontal Partitioning

There are various ways in which a fact table can be partitioned. In horizontal partitioning, we have to keep in mind the requirements for manageability of the data warehouse**.**

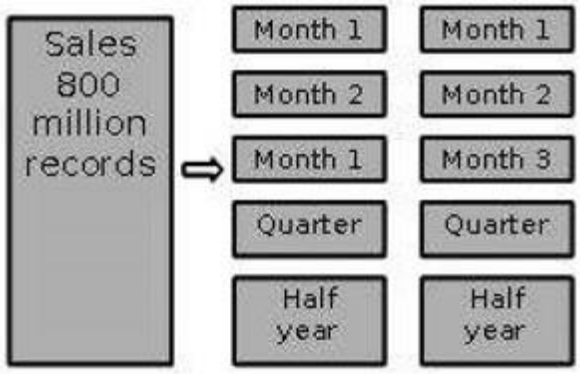
Partitioning by Time into Equal Segments

In this partitioning strategy, the fact table is partitioned on the basis of time period. Here each time period represents a significant retention period within the business. For example, if the user queries for **month to date data** then it is appropriate to partition the data into monthly segments. We can reuse the partitioned tables by removing the data in them.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Partition by Time into Different-sized Segments

This kind of partition is done where the aged data is accessed infrequently. It is implemented as a set of small partitions for relatively current data, larger partition for inactive data.



Points to Note

- The detailed information remains available online.
- The number of physical tables is kept relatively small, which reduces the operating cost.
- This technique is suitable where a mix of data dipping recent history and data mining through entire history is required.
- This technique is not useful where the partitioning profile changes on a regular basis, because repartitioning will increase the operation cost of data warehouse.

Partition on a Different Dimension

The fact table can also be partitioned on the basis of dimensions other than time such as product group, region, supplier, or any other dimension. Let's have an example.

Suppose a market function has been structured into distinct regional departments like on a **state by state** basis. If each region wants to query on information captured within its region, it would prove to be more effective to partition the fact table into regional partitions. This will cause the queries to speed up because it does not require to scan information that is not relevant.

Points to Note

- The query does not have to scan irrelevant data which speeds up the query process.
- This technique is not appropriate where the dimensions are unlikely to change in future. So, it is worth determining that the dimension does not change in future.
- If the dimension changes, then the entire fact table would have to be repartitioned.

Note − We recommend to perform the partition only on the basis of time dimension, unless you are certain that the suggested dimension grouping will not change within the life of the data warehouse.

Partition by Size of Table

When there are no clear basis for partitioning the fact table on any dimension, then we should **partition the fact table on the basis of their size.** We can set the predetermined size as a critical point. When the table exceeds the predetermined size, a new table partition is created.

Points to Note

- This partitioning is complex to manage.
- It requires metadata to identify what data is stored in each partition.

Partitioning Dimensions

If a dimension contains large number of entries, then it is required to partition the dimensions. Here we have to check the size of a dimension.

Consider a large design that changes over time. If we need to store all the variations in order to apply comparisons, that dimension may be very large. This would definitely affect the response time.

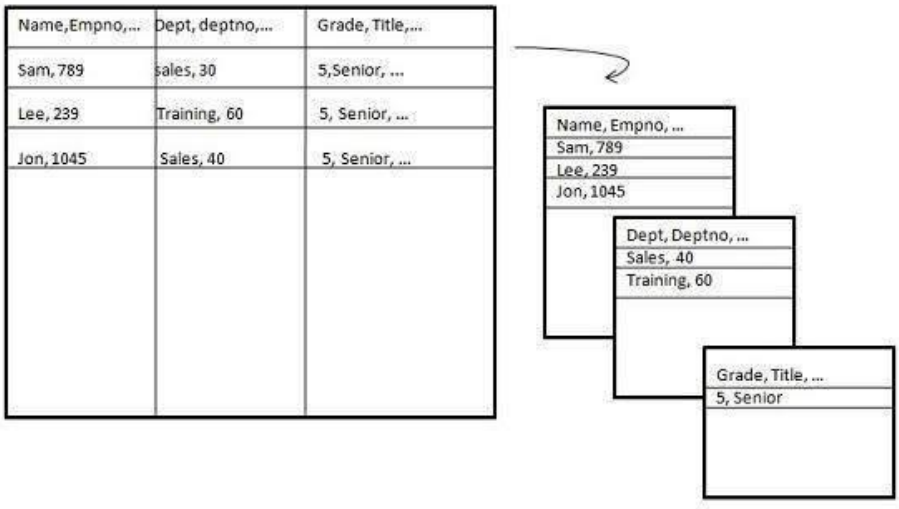
Round Robin Partitions

In the round robin technique, when a new partition is needed, the old one is archived. It uses metadata to allow user access tool to refer to the correct table partition.

This technique makes it easy to automate table management facilities within the data warehouse.

Vertical Partition

Vertical partitioning, splits the data vertically. The following images depicts how vertical partitioning is done.



Vertical partitioning can be performed in the following two ways −

- Normalization
- Row Splitting

Normalization

Normalization is the standard relational method of database organization. In this method, the rows are collapsed into a single row, hence it reduce space. Take a look at the following tables that show how normalization is performed.

Table before Normalization

Product_id	Qty	Value	sales_date	Store_id	Store_name	Location	Region
30	5	3.67	3-Aug-13	16	sunny	Bangalore	S
35	4	5.33	3-Sep-13	16	sunny	Bangalore	S
40	5	2.50	3-Sep-13	64	san	Mumbai	W
45	7	5.66	3-Sep-13	16	sunny	Bangalore	S

Table after Normalization

Store_id	Store_name	Location	Region
16	sunny	Bangalore	W
64	san	Mumbai	S

Product_id	Quantity	Value	sales_date	Store_id
30	5	3.67	3-Aug-13	16
35	4	5.33	3-Sep-13	16
40	5	2.50	3-Sep-13	64
45	7	5.66	3-Sep-13	16

Row Splitting

Row splitting tends to leave a one-to-one map between partitions. The motive of row splitting is to speed up the access to large table by reducing its size.

Note − While using vertical partitioning, make sure that there is no requirement to perform a major join operation between two partitions.

Identify Key to Partition

It is very crucial to choose the right partition key. Choosing a wrong partition key will lead to reorganizing the fact table. Let's have an example. Suppose we want to partition the following table.

Account_Txn_Table

transaction_id

account_id

transaction_type

value

transaction_date

region

branch_name

We can choose to partition on any key. The two possible keys could be

- region
- transaction_date

Suppose the business is organized in 30 geographical regions and each region has different number of branches. That will give us 30 partitions, which is reasonable. This partitioning is good enough because our requirements capture has shown that a vast majority of queries are restricted to the user's own business region.

If we partition by transaction_date instead of region, then the latest transaction from every region will be in one partition. Now the user who wants to look at data within his own region has to query across multiple partitions.

Hence it is worth determining the right partitioning key.

Hash Partitioning

Hash partitioning is a method to separate out information in a randomized way rather than putting the data in the form of groups. This partitioning system can be used efficiently to manage data on a particular platform. However, there are no performance benefits associated with hash partitioning, as it shuffles the data across the table space randomly. The partitioning system can be used to efficiently match queries.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

It makes use of hashing algorithms to distribute the data across the device to space out the load. By this method, the partitions are approximately the same size. The data that can be partitioned is not historical in nature, and thus this method is very easy to use.

The Hash Partitioning is used

1. To enable partial or full parallel partition-wise joins with likely equisized partitions.
2. To distribute data evenly among the nodes of an MPP platform that uses Oracle Real Application Clusters. Consequently, you can minimize interconnect traffic when processing inter node parallel statements.
3. To use partition pruning and partition-wise joins according to a partitioning key that is mostly constrained by a distinct value or value list.
4. To randomly distribute data to avoid I/O bottlenecks if you do not use a storage management technique that stripes and mirrors across all available devices.

```
CREATE TABLE sales_hash
  (s_productid  NUMBER,
   s_saledate   DATE,
   s_custid     NUMBER,
   s_totalprice NUMBER)
PARTITION BY HASH(s_productid)
( PARTITION p1 TABLESPACE tbs1
, PARTITION p2 TABLESPACE tbs2
, PARTITION p3 TABLESPACE tbs3
, PARTITION p4 TABLESPACE tbs4
);
```

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment-13: Implementation of Attribute oriented induction algorithm

The Attribute-Oriented Induction (AOI) approach to data generalization and summarization – based characterization was first proposed in 1989 (KDD '89 workshop) a few years before the introduction of the data cube approach. The data cube approach can be considered as a data warehouse – based, pre computational – oriented, materialized approach. It performs off-line aggregation before an OLAP or data mining query is submitted for processing.

On the other hand, the attribute oriented induction approach, at least in its initial proposal, a relational database query – oriented, generalized – based, on-line data analysis technique. However, there is no inherent barrier distinguishing the two approaches based on online aggregation versus offline precomputation.

Basic Principles of Attribute Oriented Induction:

a) **Data focusing**: Analyzing task-relevant data, including dimensions, and the result is the initial relation.

b) **Attribute-removal:** To remove attribute A if there is a large set of distinct values for A but (1) there is no generalization operator on A, or (2) A's higher-level concepts are expressed in terms of other attributes.

c) Attribute-generalization: If there is a large set of distinct values for A, and there exists a set of generalization operators on A, then select an operator and generalize A.

d) **Attribute-threshold control**: Typical 2-8, specified/default.

e) **Generalized relation threshold control (10-30):** To control the final relation/rule size.

Algorithm for Attribute Oriented Induction:

1. **InitialRel:** It is nothing but query processing of task-relevant data and deriving the initial relation.
2. **PreGen:** It is based on the analysis of the number of distinct values in each attribute and to determine the generalization plan for each attribute: removal? or how high to generalize?
3. **PrimeGen:** It is based on the PreGen plan and performing the generalization to the right level to derive a "prime generalized relation" and also accumulating the counts.
4. **Presentation:** User interaction: (1) adjust levels by drilling, (2) pivoting, (3) mapping into rules, cross tabs, visualization presentations.

Example: Let's say there is a University database that is to be characterized,

For that its corresponding DMQL will be

use University_DB

mine characteristics as "Science_Students"

in relevance to name, gender, major, birth_place, birth_date, residence, phone_no, GPA

from student

Its corresponding SQL statement can be:

Select name, gender, major, birth_place, birth_date, residence, phone_no, GPA from student where status in {"Msc", "MBA", "Ph.D." }

Now for this database let's create a characterized view:

InitialRel:

1. From this table, we are querying task-relevant data.
2. From this table, we also removed a few attributes like name and phoneno, because they make no sense in concluding insights.

PreGen

1. Now, we have generalized these results by removing a few attributes and retaining important attributes.
2. And also we have generalized a few attributes by naming them "Country" rather than "Birth_Place", "Age Range" rather than "Birth_data", "City" rather than "Residence" and so on as per the table given below.

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Jim Woodman	M	CS	Vancouver,BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
…	…	…	…	…	…	…	…
Removed	Retained	Sci,Eng, Bus	Country	Age range	City	Removed	Excl, VG,..

PrimeGen

1. Based on the PreGen plan we've performed generalization to the right level to derive a "prime generalized relation" and also we've accumulated the counts.

Gender	Major	Birth_region	Age_range	Residence	GPA	Count
M	Science	Canada	20-25	Richmond	Very-good	16
F	Science	Foreign	25-30	Burnaby	Excellent	22
…	…	…	…	…	…	…

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Final Results

Now we've and analyzed and concluded our final generalized results as shown below.

Birth_Region / Gender	Canada	Foreign	Total
M	16	14	30
F	10	22	32
Total	26	36	62

Presentation of Results:

Generalized relation: Relations where some or all attributes are generalized, with counts or other aggregation values accumulated.

Cross-tabulation: Mapping results into cross-tabulation form (similar to contingency tables).

Visualization techniques: Pie charts, bar charts, curves, cubes, and other visual forms.

Quantitative characteristic rules: Mapping generalized results in characteristic rules with quantitative information associated with it.

Experiment-14: Implementation of BIRCH algorithm

Existing data clustering methods do not adequately address the problem of processing large datasets with a limited amount of resources (i.e. memory and cpu cycles). In consequence, as the dataset size increases, they scale poorly in terms of running time, and result quality.

At a high level, **Balanced Iterative Reducing and Clustering using Hierarchies**, or BIRCH for short, deals

with large datasets by first generating a more compact summary that retains as much distribution information as possible, and then clustering the data summary instead of the original dataset. BIRCH actually complements other clustering algorithms by virtue if the fact that different clustering algorithms can be applied to the summary produced by BIRCH.

BIRCH can only deal with metric attributes (similar to the kind of features KMEANS can handle). A metric attribute is one whose values can be represented by explicit coordinates in an Euclidean space (no categorical variables).

Clustering Feature (CF)

BIRCH attempts to minimize the memory requirements of large datasets by summarizing the information contained in dense regions as Clustering Feature (CF) entries.

CF Definition : *Given N d-dimensional data points in a cluster: $\{\vec{X}_i\}$ where $i = 1, 2, ..., N$, the* **Clustering Feature (CF)** *entry of the cluster is defined as a triple:* $\mathbf{CF} = (N, \vec{LS}, SS)$, *where N is the number of data points in the cluster, $\vec{LS}$ is the linear sum of the N data points, i.e., $\sum_{i=1}^{N} \vec{X}_i$, and SS is the square sum of the N data points, i.e., $\sum_{i=1}^{N} \vec{X}_i^{\,2}$.*

As we're about to see, it's possible to have CFs composed of other CFs. In this case, the subcluster is equal to the sum of the CFs.

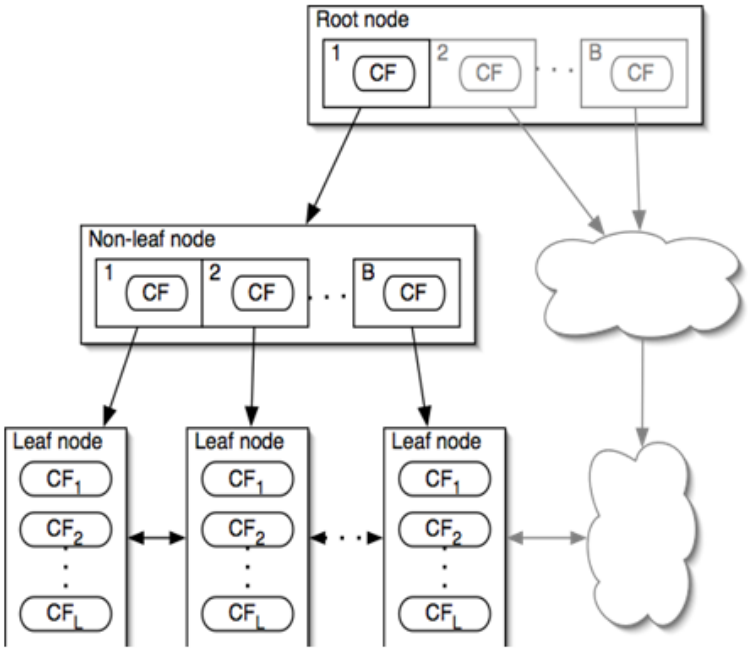
CF Additivity Theorem : *Assume that* $\mathbf{CF_1} = (N_1, \vec{LS}_1, SS_1)$, *and* $\mathbf{CF_2} = (N_2, \vec{LS}_2, SS_2)$ *are the CF entries of two disjoint subclusters. Then the CF entry of the subcluster that is formed by merging the two disjoint subclusters is:*

$$\mathbf{CF_1} + \mathbf{CF_2} = (N_1 + N_2, \vec{LS}_1 + \vec{LS}_2, SS_1 + SS_2) \quad (11)$$

CF-tree

The CF-tree is a very compact representation of the dataset because each entry in a leaf node is not a single data point but a subcluster. Each nonleaf node contains at most B entries. In this context, a single entry contains a pointer to a child node and a CF made up of the sum of the CFs in the child (subclusters of subclusters). On the other hand, a leaf node contains at most L entries, and each entry is a CF (subclusters of data points). All entries in a leaf node must satisfy a threshold

requirement. That is to say, the diameter of each leaf entry has to be less than Threshold. In addition, every leaf node has two pointers, prev and next, which are used to chain all leaf nodes together for efficient scans.



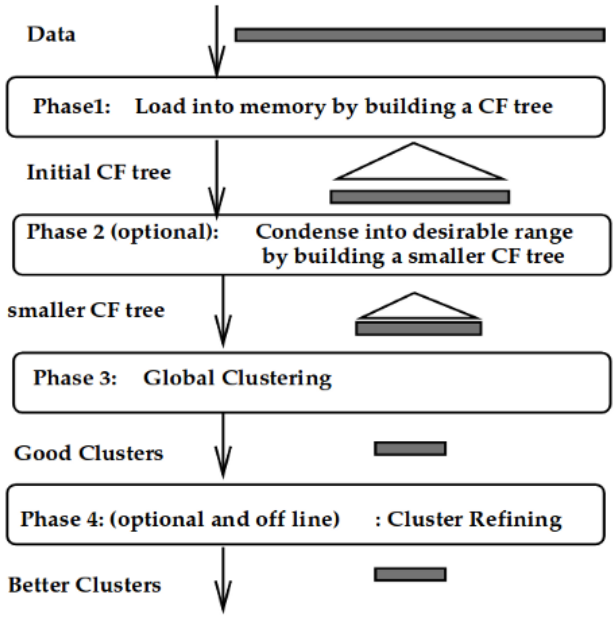
Insertion Algorithm	Let's describe how we'd go about inserting a CF entry (a single data point or subcluster) into a CF-tree. 1. Identify the appropriate leaf: Starting from the root, recursively descend the DF-tree by choosing the closest child node according to the chosen distance metric (i.e. euclidean distance). 2. Modify the leaf: Upon reaching a leaf node, find the closest entry and test whether it can absorb the CF entry without violating the threshold condition. If it can, update the CF entry, otherwise, add a new CF entry to the leaf. If there isn't enough space on the leaf for this new entry to fit in, then we must split the leaf node. Node splitting is done by choosing the two entries that are farthest apart as seeds and redistributing the remaining entries based on distance. 3. Modify the path to the leaf: Recall how every nonleaf node is itself a CF composed of the CFs of all its children. Therefore, after inserting a CF entry into a leaf, we update the CF information for each nonleaf entry on the path to the leaf. In the event of a split, we must insert a new nonleaf entry into the parent node and have it point to the newly formed leaf. If according to B, the parent doesn't have enough room, then we must split the parent as well, and so on up to the root.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Clustering Algorithm

Now that we've covered some of the concepts underlying BIRCH, let's walk through how the algorithm works.

Figure 2. BIRCH Overview



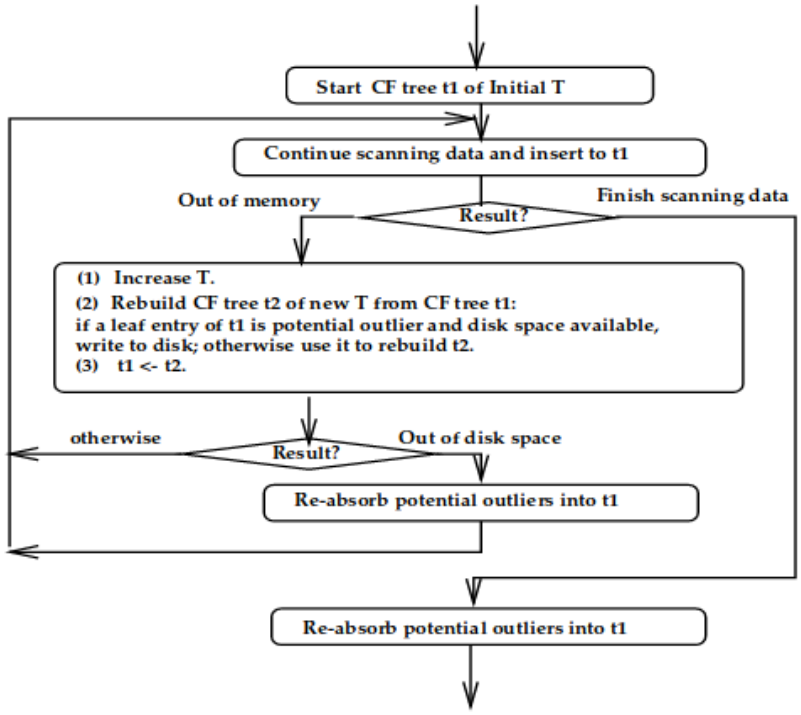
Phase 1

The algorithm starts with an initial threshold value, scans the data, and inserts points into the tree. If it runs out of memory before it finishes scanning the data, it increases the threshold value, and rebuilds a new, smaller CF-tree, by re-inserting the leaf entries of the old CF-tree into the new CF-tree. After all the old leaf entries have been re-inserted, the scanning of the data and insertion into the new CF-tree is resumed from the point at which it was interrupted.

A good choice of threshold value can greatly reduce the number of rebuilds. However, if the initial threshold is too high, we will obtain a less detailed CF-tree than is feasible with the available memory.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Figure 3. Flow Chart of Phase 1



Optionally, we can allocate a fixed amount of disk space for handling outliers. Outliers are leaf entries of low density that are judged to be unimportant with respect to the overall clustering pattern. When we rebuild the CF-tree by reinserting the old leaf entries, the size of the new CF-tree is reduced in two ways. First, we increase the threshold value, thereby allowing each leaf entry to absorb more points. Second, we treat some leaf entries as potential outliers and write them out to disk. An old leaf entry is considered a potential outlier if it has far fewer data points than average. An increase in the threshold value or a change in the distribution in response to the new data could well mean that the potential outlier no longer qualifies as an outlier. In consequence, the potential outliers are scanned to check if they can be re-absorbed in the tree without causing the tree to grow in size.

Phase 2

Given that certain clustering algorithms perform best when the number of objects is within a certain range, we can group crowded subclusters into larger ones resulting in an overall smaller CF-tree.

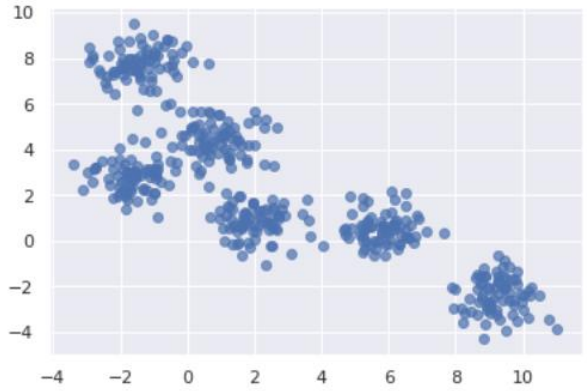
Phase 3

Almost any clustering algorithm can be adapted to categorize Clustering Features instead of data points. For instance, we could use KMEANS to categorize our data, all the while deriving the benefits from BIRCH (i.e. minimize I/O operations).

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Phase 4

Up until now, although the tree may have been rebuilt multiple times, the original data has only been scanned once. Phase 4 involves additional passes over the data to correct inaccuracies caused by the fact that the clustering algorithm is applied to a coarse summary of the data. Phase 4 also provides us with the option of discarding outliers.



Next, we initialize and train our model, using the following parameters:

1. **threshold:** The radius of the subcluster obtained by merging a new sample and the closest subcluster should be lesser than the threshold.
2. **branching_factor:** Maximum number of CF subclusters in each node
3. **n_clusters:** Number of clusters after the final clustering step, which treats the subclusters from the leaves as new samples. If set to None, the final clustering step is not performed and the subclusters are returned as they are.

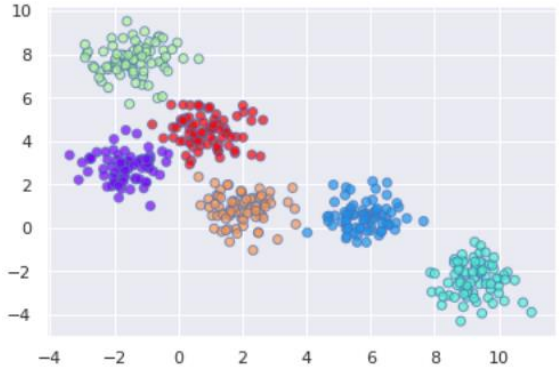
```
brc = Birch(branching_factor=50, n_clusters=None, threshold=1.5)brc.fit(X)
```

4. We use the predict method to obtain a list of points and their respective cluster.

```
labels = brc.predict(X)
```

Finally, we plot the data points using a different color for each cluster.

```
plt.scatter(X[:,0], X[:,1], c=labels, cmap='rainbow', alpha=0.7, edgecolors='b')
```



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment-15: Implementation of PAM algorithm

PAM stands for **"partition around medoids".** The algorithm is intended to find a sequence of objects called medoids that are centrally located in clusters. Objects that are tentatively defined as medoids are placed into a set S of selected objects. If O is the set of objects that the set U = O − S is the set of unselected objects. The goal of the algorithm is to minimize the average dissimilarity of objects to their closest selected object. Equivalently, we can minimize the sum of the dissimilarities between object and their closest selected object.

The algorithm has two phases:

(i) In the first phase, **BUILD,** a collection of k objects are selected for an initial set S.

(ii) In the second phase, **SWAP,** one tries to improve the quality of the clustering by exchanging selected objects with unselected objects.

The goal of the algorithm is to minimize the average dissimilarity of objects to their closest selected object. Equivalently, we can minimize the sum of the dissimilarities between object and their closest selected object.

For each object p we maintain two numbers:

- Dp, the dissimilarity between p and the closest object in S, and
- Ep, the dissimilarity between p and the second closest object in S.

These numbers must be updated every time when the sets S and U change. Note that Dj ≤ Ej and that we have p ∈ S if and only if Dp = 0.

The **BUILD** phase entails the following steps:

1. Initialize S by adding to it an object for which the sum of the distances to all other objects is minimal.
2. Consider an object i ∈ U as a candidate for inclusion into the set of selected objects.
3. For an object j ∈ U − {i} compute Dj , the dissimilarity between j and the closest object in S.
4. If Dj > d(i, j) object j will contribute to the decision to select object i (because the quality of the clustering may benefit); let Cji = max{Dj − d(j, i), 0}.
5. Compute the total gain obtained by adding i to S as gi = P j∈U Cji.
6. Choose that object i that maximizes gi ; let S := S ∪ {i} and U = U − {i}.

These steps are performed until k objects have been selected. The decisions taken in assessing object I, as shown in fig-1.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

The second phase, **SWAP,** attempts to improve the the set of selected objects and, therefore, to improve the quality of the clustering. This is done by considering all pairs (i, h) ∈ S ×U and consists of computing the effect Tih on the sum of dissimilarities between objects and the closest selected object caused by swapping i and h, that is, by transferring i from S to U and transferring h to from U to S.

The computation of T_{ih} involves the computation of the contribution K_{jih} of each object j ∈ U − {h} to the swap of i and h. Note that we have either d(j, i) > Dj or d(j, i) = Dj .

1. Kjih is computed taking into account the following cases

(a) if d(j, i) > Dj , then two subcases occur:

i. if d(j, h) ≥ Dj , then Kjih = 0;

ii. if d(j, h) < Dj , then Kjih = d(j, h) − Dj .

In both subcases, Kjih = min{d(j, h) − Dj , 0}.

(b) if d(j, i) = Dj , we have two subcases:

i. if d(j, h) < Ej , where Ej is the dissimilarity between j and the second closest selected object, then Kjih = d(j, h) − Dj ; note that Kjih can be either positive or negative.

ii. if d(j, h) ≥ Ej , then Kjih = Ej − Dj ; in this case Kjih > 0.

In each of the above subcases we have

Kjih = min{d(j, h), Ej} − Dj .

2. Compute the total result of the swap as Tih = X{Kjih | j ∈ U}.

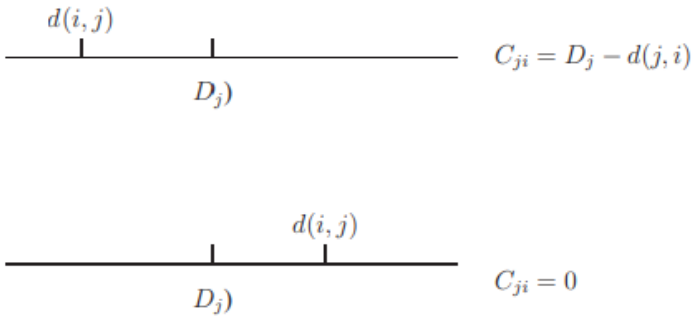


Figure 1: Computation of the Contribution C_{ji}

3. Select a pair (i, h) ∈ S × U that minimizes Tih.
4. If Tih < 0 the swap is carried out, Dp and Ep are updated for every object p, and we return at Step 1. If min Tih > 0, the value of the objective cannot be decreased and the algorithm halts. Of course, this happens when all values of Tih are positive and this is precisely the halting condition of the algorithm.

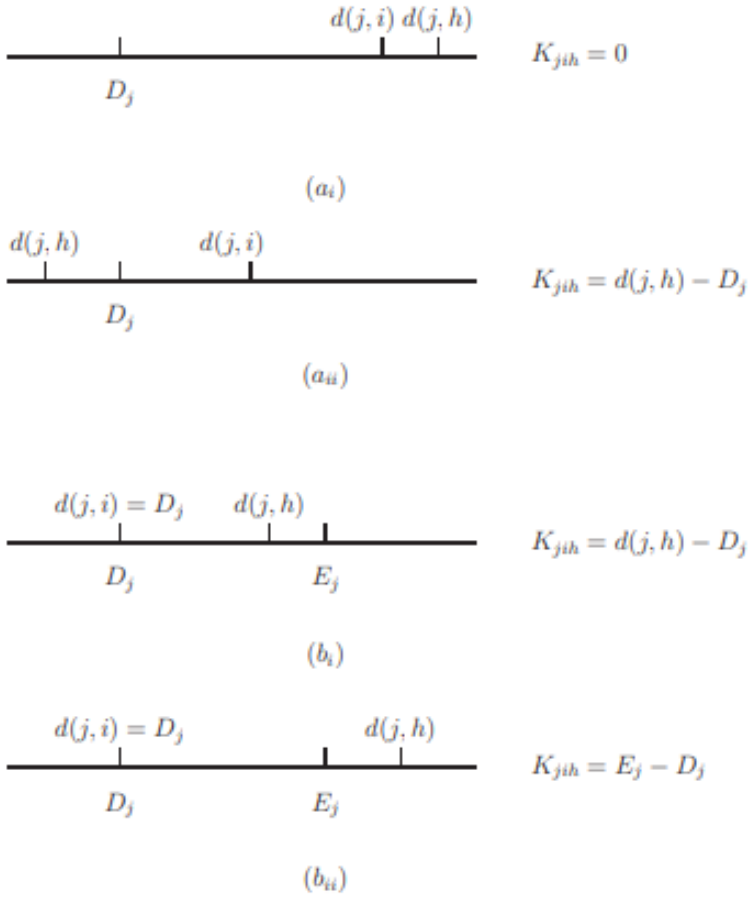


Figure 2: Computation of the contribution K_{jih} of object j to the swap of $i \in S$ with $u \in U$

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Experiment-16: Implementation of DBSCAN Clustering algorithm

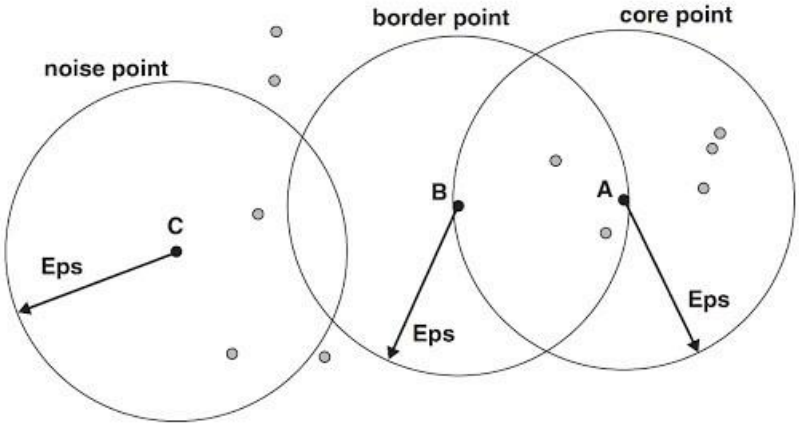
The full name of the DBSCAN algorithm is **Density-based Spatial Clustering of Applications with Noise**. Well, there are three particular words that we need to focus on from the name. They are density, clustering, and noise.

Important parameters of the DBSCAN algorithm

The first one is epsilon.

- **Epsilon:** It is a measure of the neighborhood.
  - **Neighborhood:** Suppose, this is the point we are considering right now, and let me draw a circle around this point making this as a center and add a distance Epsilon. So, we are gonna say this circle as this point's neighborhood. So, epsilon is just a number that represents the radius of the circle around a particular point that we are going to consider the neighborhood of that point.
- **min_sample**

min_samples are threshold on the least number of points that we want to see in a point's neighborhood. Suppose we are taking z = 3.



If we have 4 points in our neighborhood, this will also satisfy our threshold z = 3.

Because this threshold represents the minimum number of samples in a neighborhood.

- **Classification of data points**

Now based on these two parameters i.e., epsilon and min_samples, we are first going to classify every point in our dataset into three categories. They are

- Core points
- Boundary points or border points
- Noise points

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

- **Core points**
  - ✓ Now see the above-mentioned figure. I represented a core point A.
  - ✓ If I say a point as a core point then it must satisfy one condition. The condition is the number of neighbors must be greater than or equal to our threshold min_samples or z. If I set z = 3, then this point satisfies this condition. Hence, we say this is the core point.
  - ✓ Let's see the second type of point.
- **Boundary points**
  - ✓ If I say one point as a boundary point, then it has to satisfy the following two conditions.
  - ✓ The number of neighbors must be less than z.
  - ✓ The point should be in the neighborhood of a core point.
  - ✓ Consider the same figure mentioned above. I represented a border point B. The point has less than the number of neighbors in its neighborhood and it is in the neighborhood of another core point. So, this point B is a boundary point or border point.
- **Noise points**
  - ✓ The definition of noise point is very simple. If a point is neither a core point nor a boundary point, then it is called a noise point. In the above-mentioned figure, point C is neither a core point nor a boundary point. So, we can say that as a noise point.
  - ✓ Now we have classified every single data point into three categories. This is the first step in the DBSCAN algorithm.
  - ✓ Now you need to understand another concept.

❖ **Steps in the DBSCAN algorithm**

1. Classify the points.
2. Discard noise.
3. Assign cluster to a core point.
4. Color all the density connected points of a core point.
5. Color boundary points according to the nearest core point.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

VIVA –VOCE QUESTIONS

1. What is Data mining ?

Data mining is knowledge discovery in databases. It is extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) information or patterns from data in large databases.

2. What is difference between OLAP and data mining?

OLAP - (On-line Analytical Processing) provides you with a very good view of what is happening, but can not predict what will happen in the future or why it is happening where as data mining is group of techniques that find relationships that have not previously been discovered.

3. What are the types of tasks that are carried out during data mining?

Data mining involves 2 types of tasks

- **Prediction Tasks-** Use some variables to predict unknown or future values of other variables
- **Description Tasks-** Find human-interpretable patterns that describe the data.

4. What are some of the tasks of data mining?

A Following activities are carried out during data mining

- ✓ Classification [Predictive]
- ✓ Clustering [Descriptive]
- ✓ Association Rule Discovery [Descriptive]
- ✓ Sequential Pattern Discovery [Descriptive]
- ✓ Regression [Predictive]
- ✓ Deviation Detection [Predictive]

5. What do you mean by preprocessing of data in data mining?

A Before data is mined it has to be preprocessed. It consists of following three stages

- ✓ **Data cleaning** - Real world data is dirty so need to be cleaned
- ✓ **Data reduction**- Remove data not useful for mining
- ✓ **Data transformation** - Syntactic transformation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

6. What is Data cleaning?

Causes of Dirty Data

- ✓ Missing values
- ✓ Noisy data (Human/Machine Errors)
- ✓ Inconsistent data

Data cleaning tasks

- ✓ Handling missing values
- ✓ Identify outliers and smooth out noisy data
- ✓ Correct inconsistent data

7. Explain Data reduction?

It consists of following three tasks -

- ✓ **Dimensionality reduction** - Attribute subset selection
- ✓ **Numerosity reduction** - Tuple subset selection
- ✓ **Discretization** - Reduce the cardinality of active domain

8. What is Data Transformation?

It consist of following tasks

- ✓ **Generalization** - concept hierarchy climbing
- ✓ **Attribute/feature construction** - New attributes are constructed and added to the tuple
- ✓ **Normalization** - scaled to fall within a small, specified range

9. What are the different tasks of Data Mining?

The following activities are carried out during data mining:

- ✓ Classification
- ✓ Clustering
- ✓ Association Rule Discovery
- ✓ Sequential Pattern Discovery
- ✓ Regression
- ✓ Deviation Detection

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

10. Discuss the Life cycle of Data Mining projects?

The life cycle of Data mining projects:

- ✓ Business understanding: Understanding projects objectives from a business perspective, data mining problem definition.
- ✓ Data understanding: Initial data collection and understand it.
- ✓ Data preparation: Constructing the final data set from raw data.
- ✓ Modeling: Select and apply data modeling techniques.
- ✓ Evaluation: Evaluate model, decide on further deployment.
- ✓ Deployment: Create a report, carry out actions based on new insights.

11. Explain the process of KDD?

Data mining treat as a synonym for another popularly used term, Knowledge Discovery from Data, or KDD. In others view data mining as simply an essential step in the process of knowledge discovery, in which intelligent methods are applied in order to extract data patterns.

Knowledge discovery from data consists of the following steps:

- ✓ Data cleaning (to remove noise or irrelevant data).
- ✓ Data integration (where multiple data sources may be combined).
- ✓ Data selection (where data relevant to the analysis task are retrieved from the database).
- ✓ Data transformation (where data are transmuted or consolidated into forms appropriate for mining by performing summary or aggregation functions, for sample).
- ✓ Data mining (an important process where intelligent methods are applied in order to extract data patterns).
- ✓ Pattern evaluation (to identify the fascinating patterns representing knowledge based on some interestingness measures).
- ✓ Knowledge presentation (where knowledge representation and visualization techniques are used to present the mined knowledge to the user).

12. **What is Classification?**

Classification is the processing of finding a set of models (or functions) that describe and distinguish data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown. Classification can be used for predicting the class label of data items. However, in many applications, one may like to calculate some missing or unavailable data values rather than class labels.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

13. Explain Evolution and deviation analysis?

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

In the analysis of time-related data, it is often required not only to model the general evolutionary trend of the data but also to identify data deviations that occur over time. Deviations are differences between measured values and corresponding references such as previous values or normative values. A data mining system performing deviation analysis, upon the detection of a set of deviations, may do the following: describe the characteristics of the deviations, try to describe the reason behindhand them, and suggest actions to bring the deviated values back to their expected values.

14. What is Prediction?

Prediction can be viewed as the construction and use of a model to assess the class of an unlabeled object, or to measure the value or value ranges of an attribute that a given object is likely to have. In this interpretation, classification and regression are the two major types of prediction problems where classification is used to predict discrete or nominal values, while regression is used to predict incessant or ordered values.

15. Explain the Decision Tree Classifier?

A Decision tree is a flow chart-like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test and each leaf node (or terminal node) holds a class label. The topmost node of a tree is the root node.

A Decision tree is a classification scheme that generates a tree and a set of rules, representing the model of different classes, from a given data set. The set of records available for developing classification methods is generally divided into two disjoint subsets namely a training set and a test set. The former is used for originating the classifier while the latter is used to measure the accuracy of the classifier. The accuracy of the classifier is determined by the percentage of the test examples that are correctly classified.

In the decision tree classifier, we categorize the attributes of the records into two different types. Attributes whose domain is numerical are called the numerical attributes and the attributes whose domain is not numerical are called categorical attributes. There is one distinguished attribute called a class label. The goal of classification is to build a concise model that can be used to predict the class of the records whose class label is unknown. Decision trees can simply be converted to classification rules.

16. **What are the advantages of a decision tree classifier?**

Decision trees are able to produce understandable rules.

- ✓ They are able to handle both numerical and categorical attributes.
- ✓ They are easy to understand.
- ✓ Once a decision tree model has been built, classifying a test record is extremely fast.
- ✓ Decision tree depiction is rich enough to represent any discrete value classifier.
- ✓ Decision trees can handle datasets that may have errors.
- ✓ Decision trees can deal with handle datasets that may have missing values.

They do not require any prior assumptions. Decision trees are self-explanatory and when compacted they are also easy to follow. That is to say, if the decision tree has a reasonable number of leaves it can be grasped by non-professional users. Furthermore, since decision trees can be converted to a set of rules, this sort of representation is considered comprehensible.

17. Explain Bayesian classification in Data Mining?

A Bayesian classifier is a statistical classifier. They can predict class membership probabilities, for instance, the probability that a given sample belongs to a particular class. Bayesian classification is created on the Bayes theorem. A simple Bayesian classifier is known as the naive Bayesian classifier to be comparable in performance with decision trees and neural network classifiers. Bayesian classifiers have also displayed high accuracy and speed when applied to large databases.

18. Why Fuzzy logic is an important area for Data Mining?

Rule-based systems for classification have the disadvantage that they involve exact values for continuous attributes. Fuzzy logic is useful for data mining systems performing classification. It provides the benefit of working at a high level of abstraction. In general, the usage of fuzzy logic in rule-based systems involves the following:

- ✓ Attribute values are changed to fuzzy values.
- ✓ For a given new sample, more than one fuzzy rule may apply. Every applicable rule contributes a vote for membership in the categories. Typically, the truth values for each projected category are summed.
- ✓ The sums obtained above are combined into a value that is returned by the system. This process may be done by weighting each category by its truth sum and multiplying by the mean truth value of each category. The calculations involved may be more complex, depending on the difficulty of the fuzzy membership graphs.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

19. What is Classification Accuracy?

Classification accuracy or accuracy of the classifier is determined by the percentage of the test data set examples that are correctly classified. The classification accuracy of a classification tree = (1 – Generalization error).

20. Define Clustering in Data Mining?

Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them.

21. Write a difference between classification and clustering?

Parameters	CLASSIFICATION	CLUSTERING
Type	Used for supervised need 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 data set for verifying the model created.	There is no need for training and testing dataset
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.

22. What is Supervised and Unsupervised Learning?[TCS interview question]

Supervised learning, as the name indicates, has the presence of a supervisor as a teacher. Basically supervised learning is when we teach or train the machine using data that is well labeled. Which means some data is already tagged with the correct answer. After that, the machine is provided with a new set of examples(data) so that the supervised learning algorithm analyses the training data(set of training examples) and produces a correct outcome from labeled data.

Unsupervised learning is the training of a machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Here the task of the machine is to group unsorted information according to similarities, patterns, and differences without any prior training of data.

Unlike supervised learning, no teacher is provided that means no training will be given to the machine. Therefore, the machine is restricted to find the hidden structure in unlabeled data by itself.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

23. Name areas of applications of data mining?

Data Mining Applications for

- ✓ Finance
- ✓ Healthcare
- ✓ Intelligence
- ✓ Telecommunication
- ✓ Energy
- ✓ Retail
- ✓ E-commerce
- ✓ Supermarkets
- ✓ Crime Agencies
- ✓ Businesses Benefit from data mining

24. What are the issues in data mining?

A number of issues that need to be addressed by any serious data mining package

- ✓ Uncertainty Handling
- ✓ Dealing with Missing Values
- ✓ Dealing with Noisy data
- ✓ Efficiency of algorithms
- ✓ Constraining Knowledge Discovered to only Useful
- ✓ Incorporating Domain Knowledge
- ✓ Size and Complexity of Data
- ✓ Data Selection
- ✓ Understandably of Discovered Knowledge: Consistency between Data and Discovered Knowledge.

25. Give an introduction to data mining query language?

DBQL or Data Mining Query Language proposed by Han, Fu, Wang, et.al. This language works on the DBMiner data mining system. DBQL queries were based on SQL(Structured Query language). We can this language for databases and data warehouses as well. This query language support ad hoc and interactive data mining.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

26. Differentiate Between Data Mining And Data Warehousing?

Data Mining: It is the process of finding patterns and correlations within large data sets to identify relationships between data. Data mining tools allow a business organization to predict customer behavior. Data mining tools are used to build risk models and detect fraud. Data mining is used in market analysis and management, fraud detection, corporate analysis, and risk management. It is a technology that aggregates structured data from one or more sources so that it can be compared and analyzed rather than transaction processing.

Data Warehouse: A data warehouse is designed to support the management decision-making process by providing a platform for data cleaning, data integration, and data consolidation. A data warehouse contains subject-oriented, integrated, time-variant, and non-volatile data.

Data warehouse consolidates data from many sources while ensuring data quality, consistency, and accuracy. Data warehouse improves system performance by separating analytics processing from transnational databases. Data flows into a data warehouse from the various databases. A data warehouse works by organizing data into a schema that describes the layout and type of data. Query tools analyze the data tables using schema.

27. What is Data Purging?

The term purging can be defined as Erase or Remove. In the context of data mining, data purging is the process of remove, unnecessary data from the database permanently and clean data to maintain its integrity.

28. What are the differences between OLAP And OLTP?[IMP]

OLAP (Online Analytical Processing)	OLTP (Online Transaction Processing)
Consists of historical data from various Databases.	Consists only of application-oriented day-to-day operational current data.
Application-oriented day-to-dayIt is subject-oriented. Used for Data Mining, Analytics, Decision making, etc.	It is application-oriented. Used for business tasks.
The data is used in planning, problem-solving, and decision-making.	The data is used to perform day-to-day fundamental operations.
It reveals a snapshot of present business tasks.	It provides a multi-dimensional view of different business tasks.
A large forex amount of data is stored typically in TB, PB	The size of the data is relatively small as the historical data is archived. For example, MB, GB
Relatively slow as the amount of data involved is large. Queries may take hours.	Very Fast as the queries operate on 5% of the data.
It only needs backup from time to time as compared to OLTP.	The backup and recovery process is maintained

religiously

This data is generally managed by the CEO, MD, GM. This data is managed by clerks, managers.

Only read and rarely write operation. Both read and write operations.

29. Explain Association Algorithm In Data Mining?

Association analysis is the finding of association rules showing attribute-value conditions that occur frequently together in a given set of data. Association analysis is widely used for a market basket or transaction data analysis. Association rule mining is a significant and exceptionally dynamic area of data mining research. One method of association-based classification, called associative classification, consists of two steps. In the main step, association instructions are generated using a modified version of the standard association rule mining algorithm known as Apriori. The second step constructs a classifier based on the association rules discovered.

30. Explain how to work with data mining algorithms included in SQL server data mining?

SQL Server data mining offers Data Mining Add-ins for Office 2007 that permits finding the patterns and relationships of the information. This helps in an improved analysis. The Add-in called a Data Mining Client for Excel is utilized to initially prepare information, create models, manage, analyze, results.

31. **Explain Over-fitting?**

The concept of over-fitting is very important in data mining. It refers to the situation in which the induction algorithm generates a classifier that perfectly fits the training data but has lost the capability of generalizing to instances not presented during training. In other words, instead of learning, the classifier just memorizes the training instances. In the decision trees over fitting usually occurs when the tree has too many nodes relative to the amount of training data available. By increasing the number of nodes, the training error usually decreases while at some point the generalization error becomes worse. The Over-fitting can lead to difficulties when there is noise in the training data or when the number of the training datasets, the error of the fully built tree is zero, while the true error is likely to be bigger.

There are many disadvantages of an over-fitted decision tree:

- ✓ Over-fitted models are incorrect.
- ✓ Over-fitted decision trees require more space and more computational resources.
- ✓ They require the collection of unnecessary features.

32. Define Tree Pruning?

When a decision tree is built, many of the branches will reflect anomalies in the training data due to noise or outliers. Tree

pruning methods address this problem of over-fitting the data. So the tree pruning is a technique that removes the overfitting problem. Such methods typically use statistical measures to remove the least reliable branches, generally resulting in faster classification and an improvement in the ability of the tree to correctly classify independent test data. The pruning phase eliminates some of the lower branches and nodes to improve their performance. Processing the pruned tree to improve understandability.

33. What is a Sting?

Statistical Information Grid is called STING; it is a grid-based multi-resolution clustering strategy. In the STING strategy, every one of the items is contained into rectangular cells, these cells are kept into different degrees of resolutions and these levels are organized in a hierarchical structure.

34. Explain the Issues regarding Classification And Prediction?

Preparing the data for classification and prediction:

- ✓ Data cleaning
- ✓ Relevance analysis
- ✓ Data transformation
- ✓ Comparing classification methods
- ✓ Predictive accuracy
- ✓ Speed
- ✓ Robustness
- ✓ Scalability
- ✓ Interpretability

35. Explain the use of data mining queries or why data mining queries are more helpful?

The data mining queries are primarily applied to the model of new data to make single or multiple different outcomes. It also permits us to give input values. The query can retrieve information effectively if a particular pattern is defined correctly. It gets the training data statistical memory and gets the specific design and rule of the common case addressing a pattern in the model. It helps in extracting the regression formulas and other computations. It additionally recovers the insights concerning the individual cases utilized in the model. It incorporates the information which isn't utilized in the analysis, it holds the model with the assistance of adding new data and perform the task and cross-verified.

36. What is a machine learning-based approach to data mining?

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Machine learning is basically utilized in data mining since it covers automatic programmed processing systems, and it depended on logical or binary tasks. . Machine learning for the most part follows the rule that would permit us to manage more general information types, incorporating cases and in these sorts and number of attributes may differ. Machine learning is one of the famous procedures utilized for data mining and in Artificial intelligence too.

37. What is the K-means algorithm?

K-means clustering algorithm – It is the simplest unsupervised learning algorithm that solves clustering problems. K-means algorithm partition n observations into k clusters where each observation belongs to the cluster with the nearest mean serving as a prototype of the cluster.

38. What are precision and recall?[IMP]

- ✓ Precision is the most commonly used error metric in the n classification mechanism. Its range is from 0 to 1, where 1 represents 100%.
- ✓ Recall can be defined as the number of the Actual Positives in our model which has a class label as Positive (True Positive)". Recall and the true positive rate is totally identical. Here's the formula for it:
- ✓ Recall = (True positive)/(True positive + False negative)

39. Why is KNN preferred when determining missing numbers in data?

K-Nearest Neighbour (KNN) is preferred here because of the fact that KNN can easily approximate the value to be determined based on the values closest to it. The k-nearest neighbor (K-NN) classifier is taken into account as an example-based classifier, which means that the training documents are used for comparison instead of an exact class illustration, like the class profiles utilized by other classifiers.

As such, there's no real training section. once a new document has to be classified, the k most similar documents (neighbors) are found and if a large enough proportion of them are allotted to a precise class, the new document is also appointed to the present class, otherwise not. Additionally, finding the closest neighbors is quickened using traditional classification strategies.

40. Explain Prepruning and Post pruning approach in Classification?

Prepruning: In the prepruning approach, a tree is "pruned" by halting its construction early (e.g., by deciding not to further split or partition the subset of training samples at a given node). Upon halting, the node becomes a leaf. The leaf may hold the most frequent class among the subset samples, or the probability distribution of those samples. When constructing a tree, measures such as statistical significance, information gain, etc., can be used to assess the goodness of

a split. If partitioning the samples at a node would result in a split that falls below a pre-specified threshold, then further partitioning of the given subset is halted. There are problems, however, in choosing a proper threshold. High thresholds could result in oversimplified trees, while low thresholds could result in very little simplification.

Postpruning: The postpruning approach removes branches from a "fully grown" tree. A tree node is pruned by removing its branches. The cost complexity pruning algorithm is an example of the post pruning approach. The pruned node becomes a leaf and is labeled by the most frequent class among its former branches. For every non-leaf node in the tree, the algorithm calculates the expected error rate that would occur if the subtree at that node were pruned. Next, the predictable error rate occurring if the node were not pruned is calculated using the error rates for each branch, collective by weighting according to the proportion of observations along each branch. If pruning the node leads to a greater probable error rate, then the subtree is reserved. Otherwise, it is pruned. After generating a set of progressively pruned trees, an independent test set is used to estimate the accuracy of each tree. The decision tree that minimizes the expected error rate is preferred.

41. What is the difference between Data Mining and Data Analysis?

Data Mining	Data Analysis
Used to perceive designs in data stored.	Used to arrange and put together raw information in a significant manner.
Mining is performed on clean and well-documented.	The analysis of information includes Data Cleaning. So, information is not available in a well-documented format.
Results extracted from data mining are difficult to interpret.	Results extracted from information analysis are not difficult to interpret.

42. What is the difference between Data Mining and Data Profiling?

- ✓ **Data Mining:** Data Mining refers to the analysis of information regarding the discovery of relations that have not been found before. It mainly focuses on the recognition of strange records, conditions, and cluster examination.
- ✓ **Data Profiling:** Data Profiling can be described as a process of analyzing single attributes of data. It mostly focuses on giving significant data on information attributes, for example, information type, recurrence, and so on.

43. What are the important steps in the data validation process?

As the name proposes Data Validation is the process of approving information. This progression principally has two methods associated with it. These are Data Screening and Data Verification.

- ✓ Data Screening: Different kinds of calculations are utilized in this progression to screen the whole information to

discover any inaccurate qualities.

- ✓ Data Verification: Each and every presumed value is assessed on different use-cases, and afterward a final conclusion is taken on whether the value must be remembered for the information or not.

44. What is Visualization?

Visualization is for the depiction of data and to gain intuition about the data being observed. It assists the analysts in selecting display formats, viewer perspectives, and data representation schema.

45. Give some data mining tools?

- ✓ DBMiner
- ✓ GeoMiner
- ✓ Multimedia miner
- ✓ WeblogMiner

46. What are the most significant advantages of Data Mining?

There are many advantages of Data Mining. Some of them are listed below:

- ✓ Data Mining is used to polish the raw data and make us able to explore, identify, and understand the patterns hidden within the data.
- ✓ It automates finding predictive information in large databases, thereby helping to identify the previously hidden patterns promptly.
- ✓ It assists faster and better decision-making, which later helps businesses take necessary actions to increase revenue and lower operational costs.
- ✓ It is also used to help data screening and validating to understand where it is coming from.
- ✓ Using the Data Mining techniques, the experts can manage applications in various areas such as Market Analysis, Production Control, Sports, Fraud Detection, Astrology, etc.
- ✓ The shopping websites use Data Mining to define a shopping pattern and design or select the products for better revenue generation.
- ✓ Data Mining also helps in data optimization.
- ✓ Data Mining can also be used to determine hidden profitability.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

47. What is reinforcement learning?

Reinforcement Learning is a learning mechanism about how to map situations to actions. The end result should help you to increase the binary reward signal. In this method, a learner is not told which action to take but instead must discover which action offers a maximum reward. This method is based on the reward/penalty mechanism.

48. What is Visualization?

Visualization is for the depiction of information and to acquire knowledge about the information being observed. It helps the experts in choosing format designs, viewer perspectives, and information representation patterns.

49. Name some best tools which can be used for data analysis.

The most common useful tools for data analysis are:

- ✓ Google Search Operators
- ✓ KNIME
- ✓ Tableau
- ✓ Solver
- ✓ RapidMiner
- ✓ Io
- ✓ NodeXL

50. What do you understand by data aggregation and data generalization?

- ✓ **Data Aggregation:** Data aggregation is a process where data is aggregated altogether, and we can construct a cube for data analysis purposes.
- ✓ **Data generalization:** Data generalization is a process where high-level data replace low-level data to make it more meaningful and generalized.

←→←→←→******←→←→←→