Practical No.1

Aim: To build Data warehouse and Explore WEKA.

Date : 01/02/23

Practical No. 1

Aim : To build Data Warehouse and Explore WEKA.

Theory :

Data Warehouse :

Data warehousing is the process of constructing and using a data warehouse. A data warehouse is constructed by integrating data from multiple heterogeneous sources that support analytical reporting, structured and/or ad hoc queries, and decision making. Data warehousing involves data cleaning, data integration, and data consolidations.

Data warehouse characteristics :→

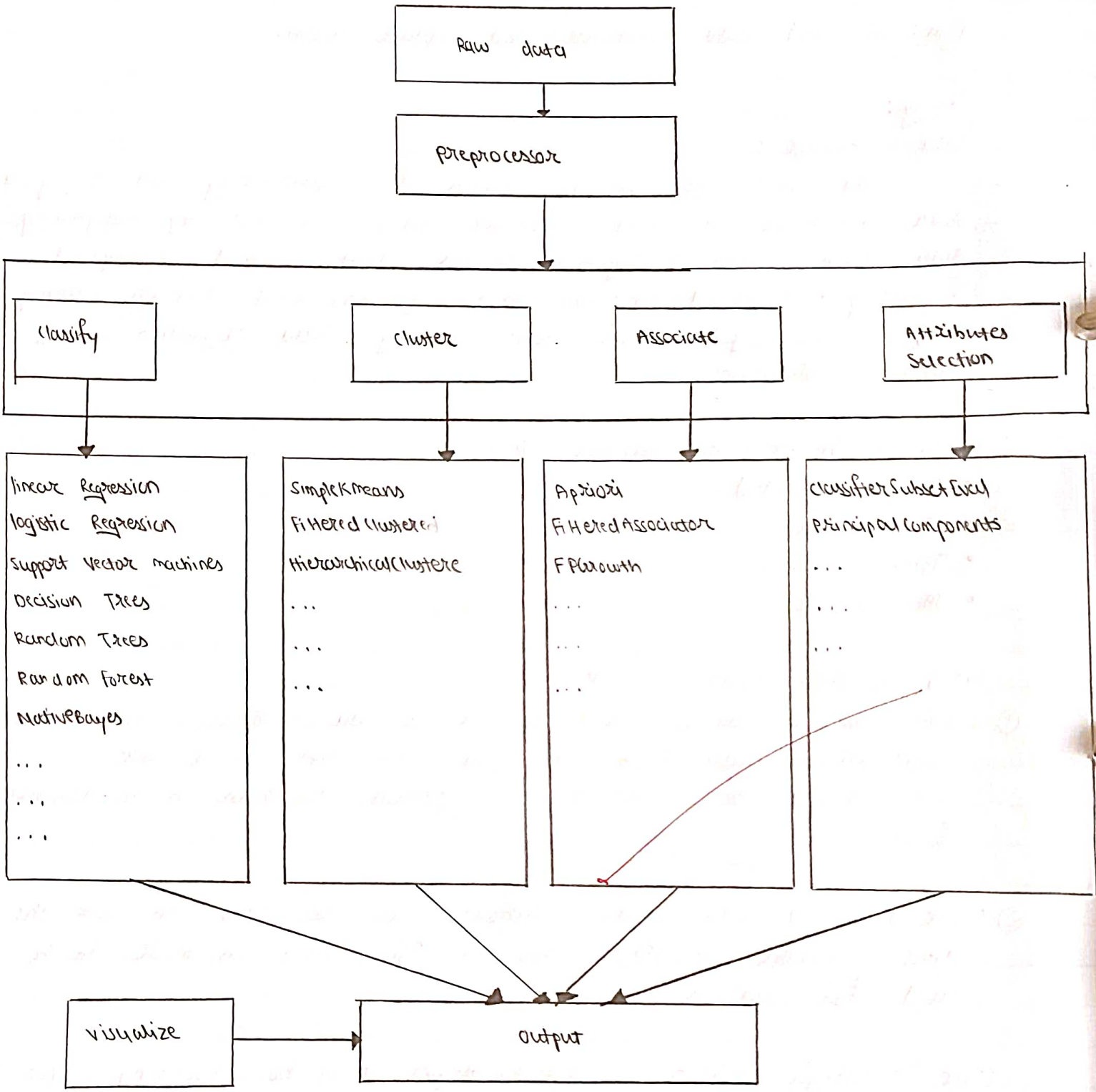
- Subject - oriented
- Integrated
- Time - variant
- Non - Volatile

Need of Data Warehouse :→

① **Business user :** Business users require a data warehouse to view summarized data from the past. Since these people are non-technical, the data may be present to them in an elementary form.

② **Store historical data :** Data warehouse is required to store the time variable data from the past. This input is made to be used for various purposes.

③ **Make strategic decision :** Some strategies may be depending upon the data in the data warehouse. So, data warehouse contributes to making strategic decision.


Raw data
Preprocessor
Classify
Cluster
Associate
Attributes Selection
linear Regression
logistic Regression
Support Vector machines
Decision Trees
Random Trees
Random Forest
NativeBayes
...
...
...
SimpleKmeans
FilteredClusterer
HierarchicalClusterer
...
...
...
Apriori
FilteredAssociator
FPGrowth
...
...
...
ClassifierSubsetEval
PrincipalComponents
...
...
...
Visualize
Output

Date :

④ For data consistency and quality : Bringing the data from different sources at commonplace, the user can effectively undertake to bring the uniformity and consistency in data.

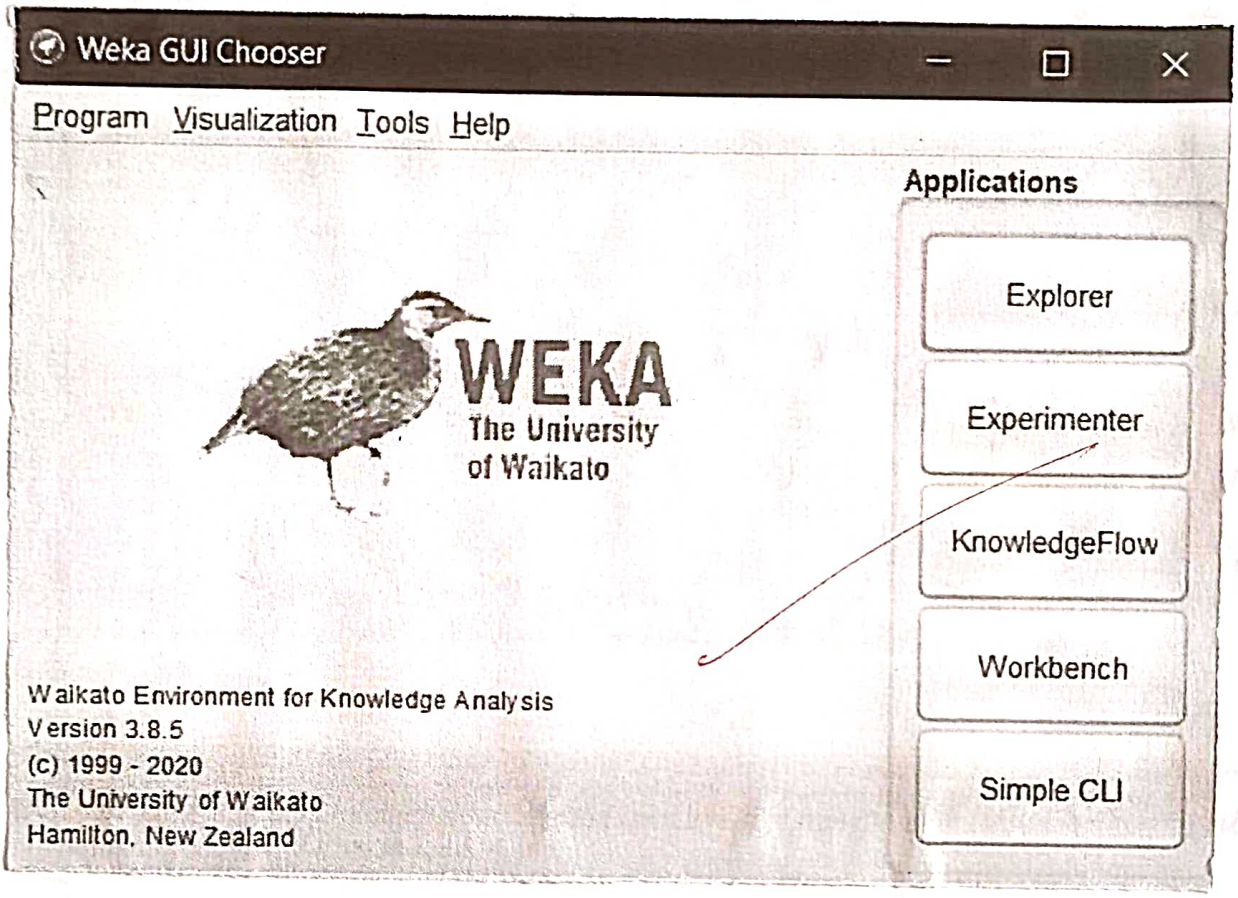
⑤ High response time : Data warehouse has to be ready for somewhat unexpected loads and types of queries, which demands a significant degree of flexibility and quick response time.

Benefits of data warehouse :→

① Understand business trends and make better forecasting decisions.
② Data warehouses are designed to perform well enormous amount of data.
③ The structure of data warehouses is more accessible for end-users to navigate, understand and query.
④ Queries that would be complex in many normalized databases could be easier to build and maintain in data warehouses
⑤ Data warehousing is an efficient method to manage demand for lots of information from lots of users.
⑥ Data warehousing provides the capabilities to analyze a large amount of historical data.

Exploring weka Tool :→

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


Weka GUI Chooser
Program Visualization Tools Help
Applications
Explorer
Experimenter
KnowledgeFlow
Workbench
Simple CLI
WEKA
The University of Waikato
Waikato Environment for Knowledge Analysis
Version 3.8.5
(c) 1999 - 2020
The University of Waikato
Hamilton, New Zealand

Date :

If you observe the beginning of the flow of the image, you will understand that there are many stages indealing with Big data to make it suitable for machine learning.

First, you will start with the raw data collected from the field. This data may contain several null values and irrelevant fields. You use the data preprocessing tools provided in WEKA to cleanse the data.

Then, you would save the preprocessed data in your local storage for applying ML algorithms.

Next, depending on the kind of ML model that you are trying to develop you would select one of the options such as classify, cluster or associate. The attributes selection allows the automatic selection of features to creates a reduced dataset.

WEKA provides the implementation of several algorithms. You would select an algorithm of your choice, set the desired parameters and run it on the dataset.

Then, WEKA would give you the statistical output of the model processing. It provides you a visualization tool to inspect the data.

The various models can be applied on the same dataset. You can then compare the outputs of different models and select the best that meets your purpose.

Thus, the use of WEKA results in quicker development of machine learning models on the whole.

Now that we have seen what WEKA is and what it does, in the next chapter let us learn how to install WEKA on your local computer.

Weka is a data mining slw that uses a collection of machine learning algorithms. These algorithms can be applied directly to the data or called from the Java code

Date :

Weka is a collection of tools for :

- Regression
- clustering
- Association
- Data pre-processing
- classification
- Visualisation weka application Interface.

There are totally five application interfaces available for weka. when we open weka, it will start the weka GUIChooser screen from where we can open the weka application interfaces Explorer preprocessing, attribute selection, learning, visualization Experimenter Testing and evaluating machine learning algorithms knowledge Flow Visual design of KDD process simple command line A simple interface for typing commands.

Weka data formats :

Weka uses the Attribute Relation File format for data analysis. by default. But listed below are some formats that weka supports. from where data can be imported :

- arff
- arff.gz
- bsi
- csv
- dat
- data
- json
- json.gz
- libsvm
- m
- names
- xrff
- xrff.gz

Date :

Arff Formate :

An Arff file contains two sections - headers and data

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

An ARFF file requires the declarations of the relations.

@relation :-

This is the first line in any ARFF file, written in the header sections, followed by the relation / dataset name. The relation name must be a string and if it contains spaces, then it should be enclosed between quotes.

@attribute :-

These are declared with their names and the type or range in the headersections. Weka supports the following data types for attributes :

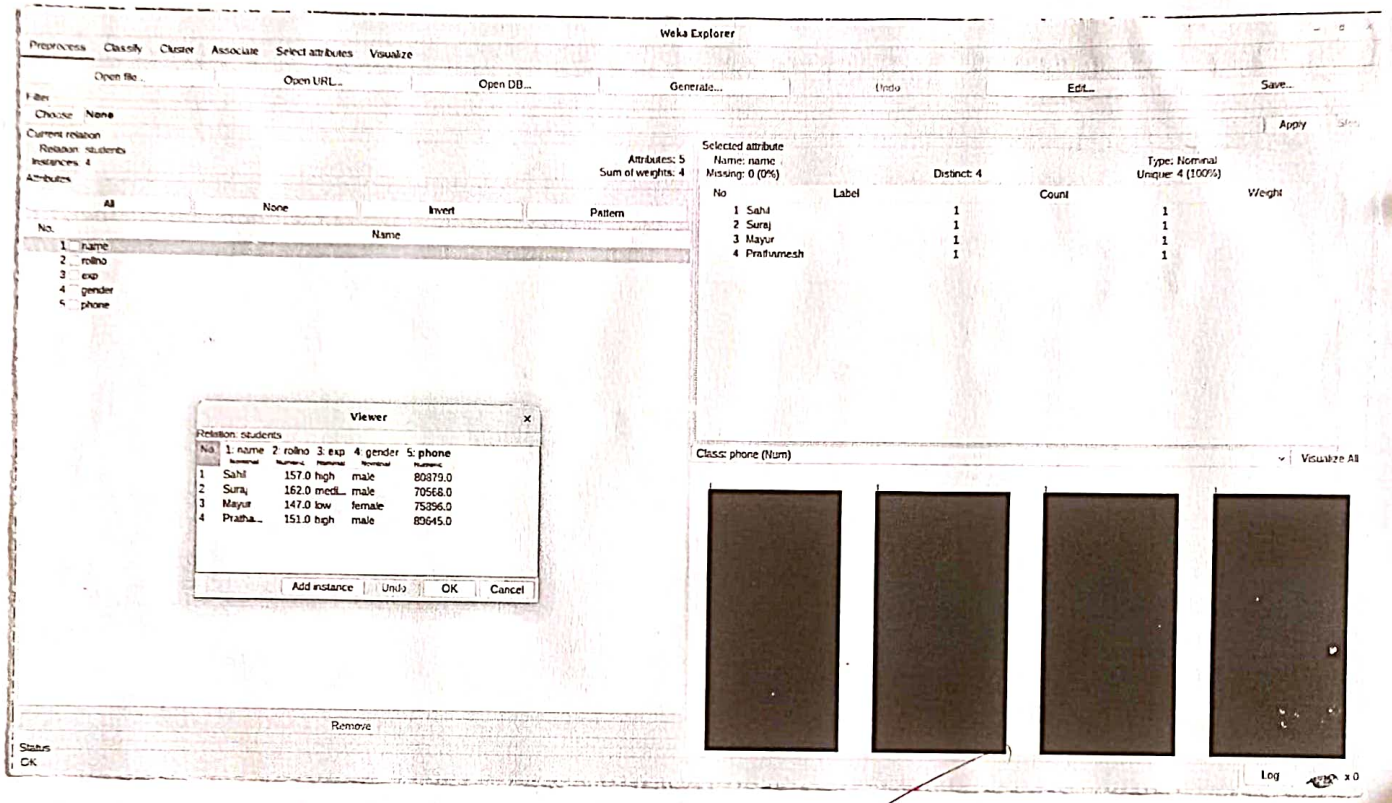
- Numeric
- <nominal-specification>
- String
- date
- @data - defined in the data section followed by the list of all data segments.

Creating a Student Table with the help of data mining Tool weka :

@ relation students

@ attribute name {Sahil, Suraj, mayur, Prathamesh}
@ attribute rollno numeric
@ attribute exp {low, medium, high}
@ attribute gender {male, female}
@ attribute phone numeric

Output:



Conclusion:

Thus, the training data table is created and WEKA tool is explored.

Date :

@ data

Sahil , 157 , high , male , 80879
Suraj , 162 , medium , male , 70568
Mayur , 147 , low , female , 75896
Prathamesh , 151 , high , male , 89645

Conclusion :-

Thus the training data table is created and WEKA tool is explored.

Viva Questions :

① Define Data warehouse.

→ It is a system used for reporting and data analysis and is considered a core component of business intelligence. A datawarehouse is a subject oriented, integrated, non-volatile and time variant collection of data in support of managements decision making process.

② what are the characteristics of Data warehousing?

→ There are four characteristic of data warehousing and they are :
a) Subject oriented
b) Integrated
c) Time - variant
d) Non - volatile.

③ what is weka Tool and what are the significance of weka?

→ ① weka is an open source software software.
② It is collection of machine learning algorithms for data mining tasks.
③ weka contains tools for data pre-processing, classification, regression, clustering, association rules and visualization.

* Significance of weka :

① Free availability under GNU General public License

Date :

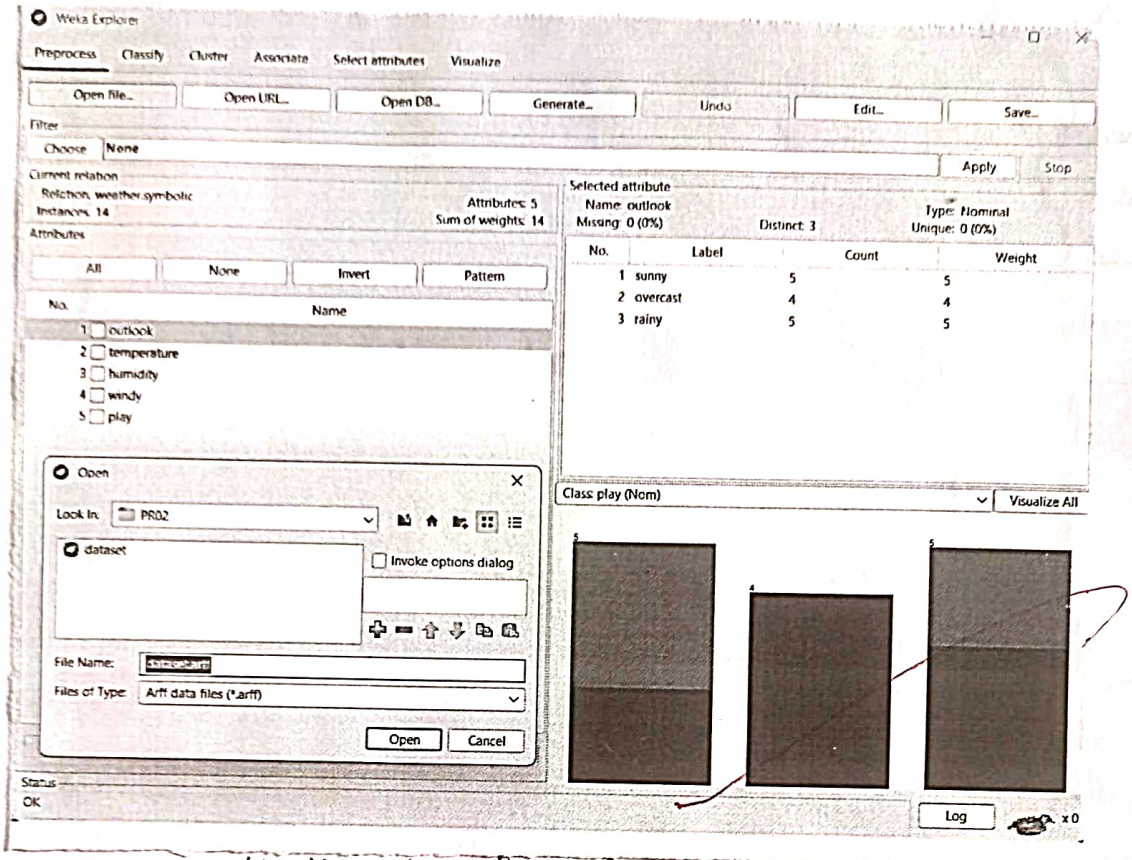
② Portability, since it is fully implemented in Java programming language
③ A comprehensive collection of data preprocessing and modelling techniques.
④ Ease of use due to its graphical user interfaces.
⑤ It provides you a visualization tool to inspect the data.
⑥ The use of ~~WEKA~~ results in quicker development of ML models on the whole.

08/02/2022

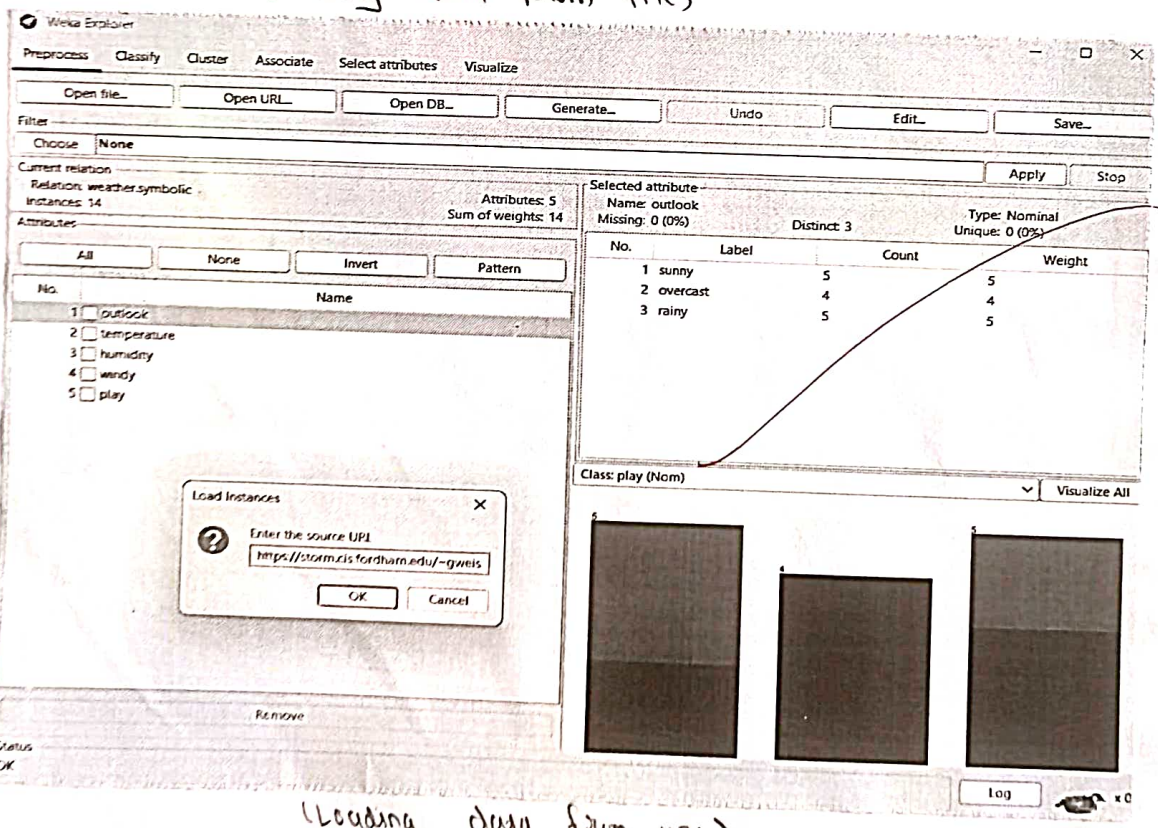
A

Practical No. 02

Aim: To demonstrate pre-processing on provided dataset.



(Loading data from file)



(Loading data from URL)

Date : 08/02/2023 Practical No. 02

Aim : To demonstrate pre-processing on provided datasets.

Theory:

The data that is collected from the field contains many unwanted things that leads to wrong analysis. For example, the data may contain null fields, it may contain columns that are irrelevant to the current analysis, and so on. Thus, the data must be preprocessed to meet the requirements of the type of analysis you are seeking. This is the done in the preprocessing module.

To demonstrate the available features in preprocessing, we can use the database that is provided in the installation

Weka loading Data :→

The data can be loaded from the following sources -

- Local file system
- web
- Database.

Load data from Local File System :→

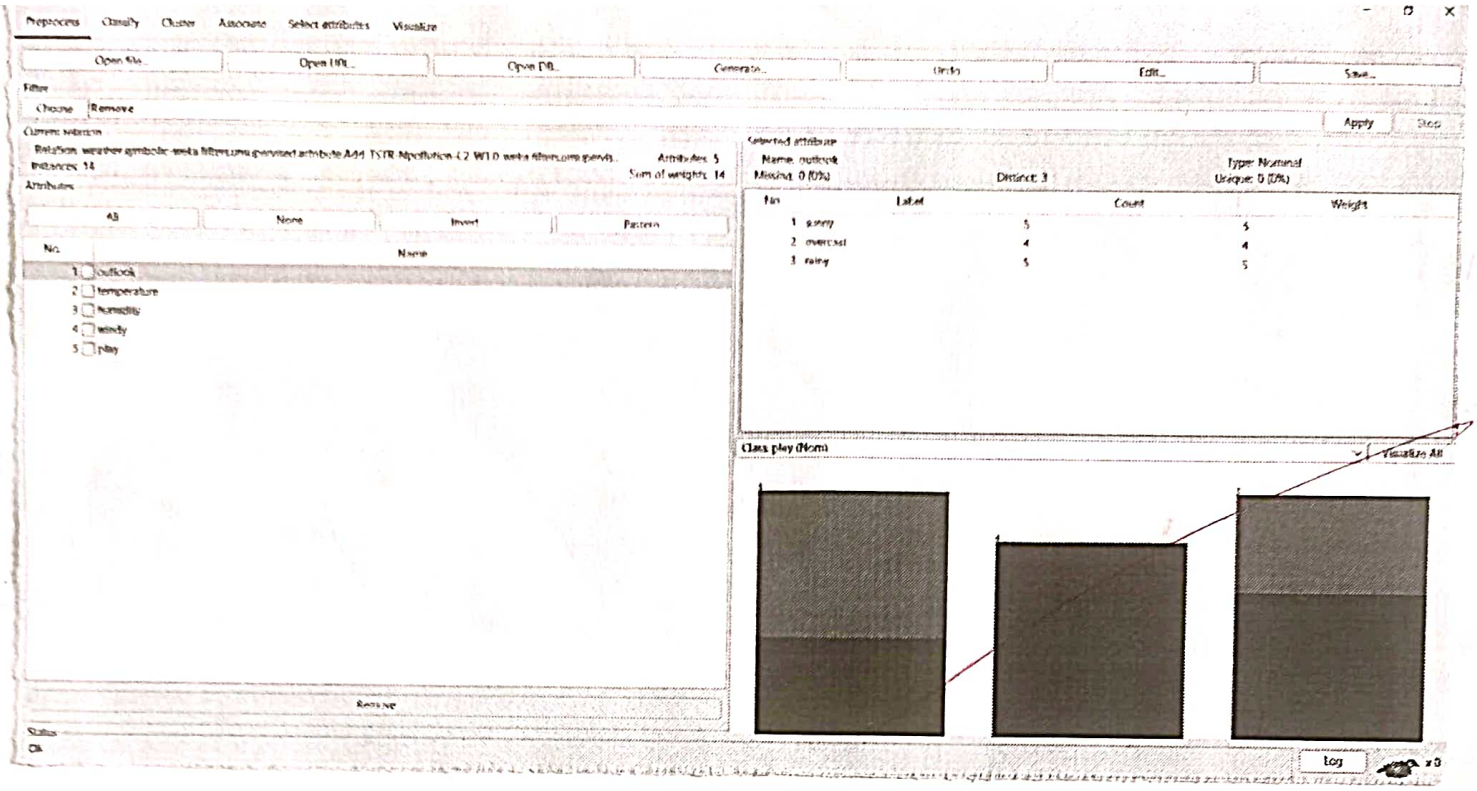
Just under the machine learning tabs that you studies in the previous lesson, you would find the following three buttons -

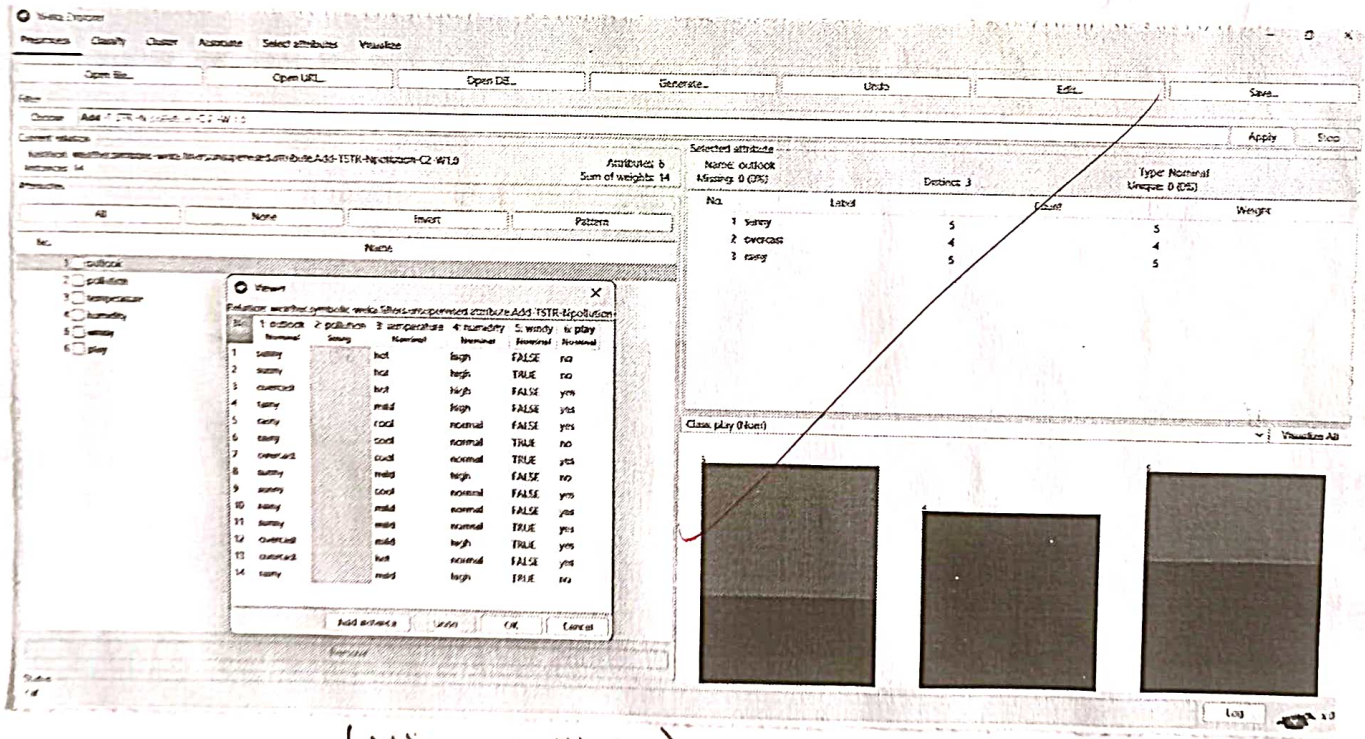
open file...
open URL...
open DB...

click on the open file ... button. A directory navigator window opens as shown in following screen -

Load data from web :→

Once you click on the open URL.. button, you see a window as follows:





(adding attribute)

Date :

We will open the file from a public URL type the following URL in the popup box -

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

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

Understanding Data :→

Let us first look at the highlighted current relation sub window. It shows the name of database that is currently loaded. You can infer two points from this sub window - There are 14 instances - the number of rows in the table. The table contains 5 attributes - the field, which are discussed in the upcoming sections.

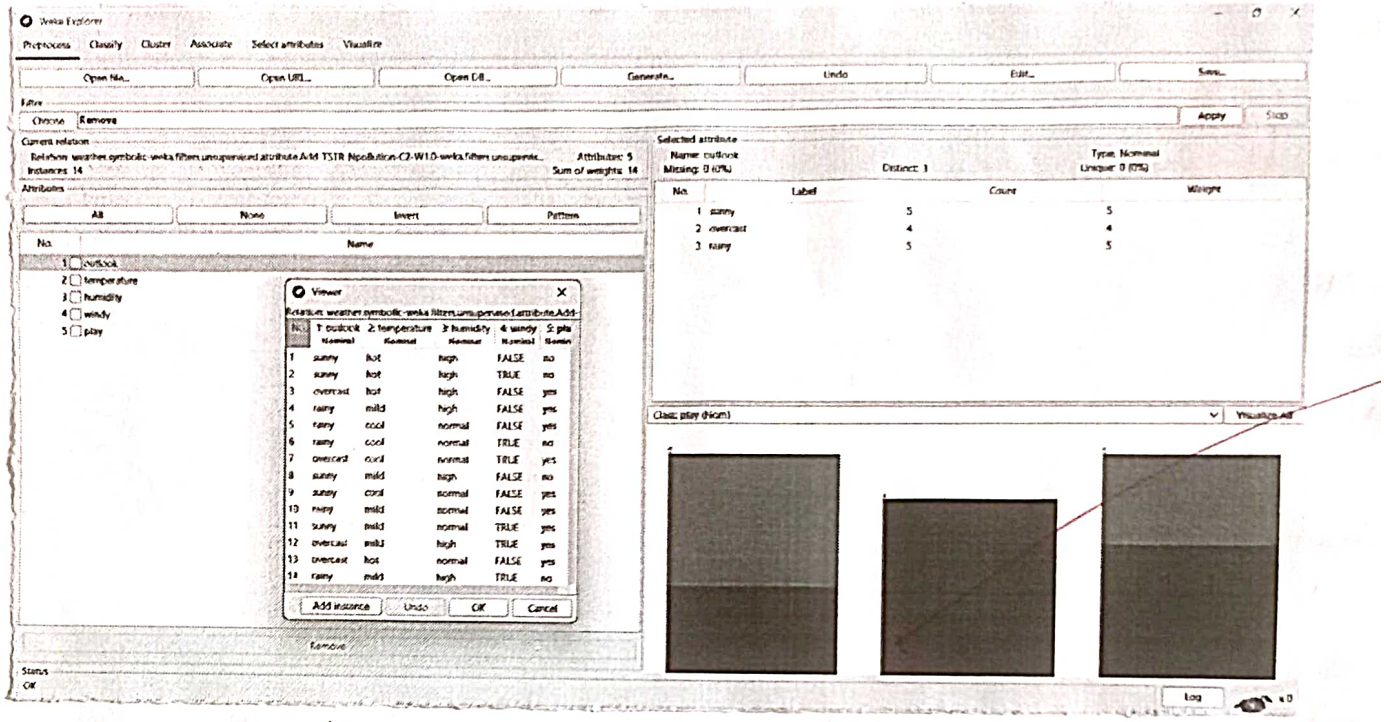
On the left side, notice the attributes sub window that display the various fields in the database.

Applying Filters :→

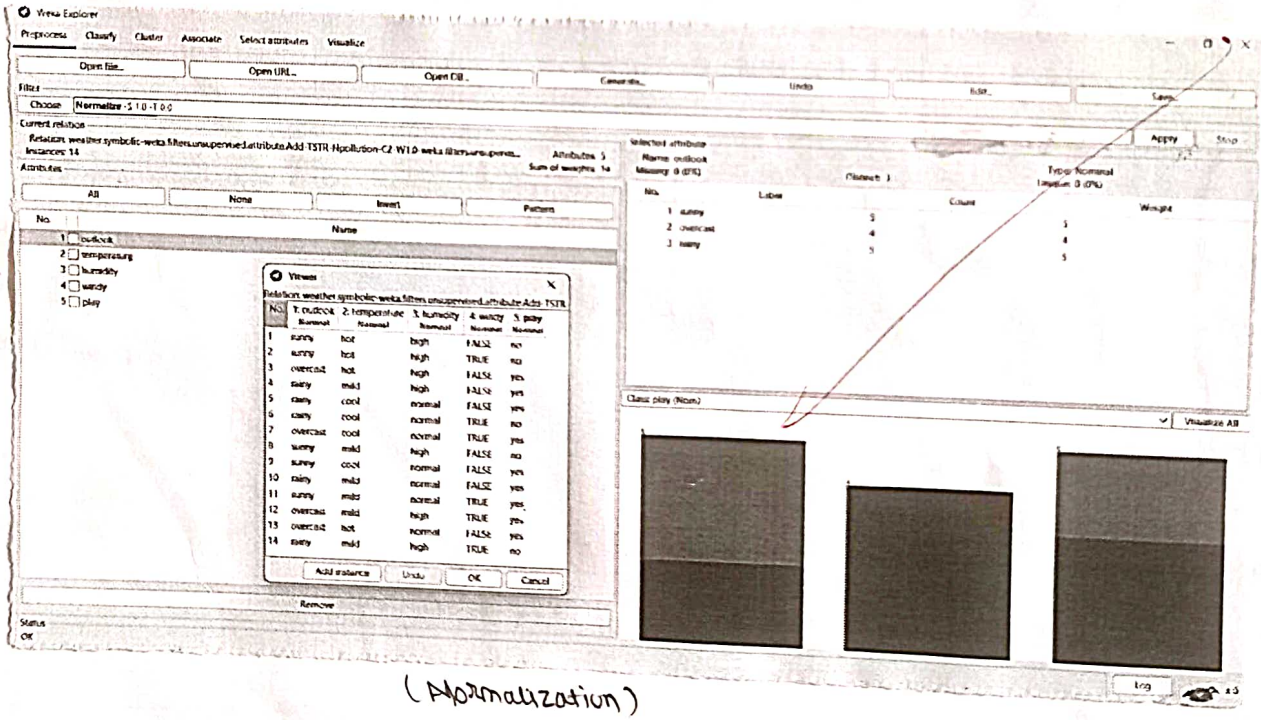
Some of the machine learning techniques such as associations rule mining required categorical data. To illustrate the use of filters, we will use weather-numeric.arff database that contain two numeric attributes - temperature and humidity. We will convert these to nominal by applying a filters on our raw data, click on the choose button in the filter subwindow and select the following filters -

weka → filters → supervised → attribute → Discretize,

Click on the Apply button and examine the temperature and/or humidity attribute. You will notice that these have changed from numeric to nominal types.



(Removing attribute)



(Normalization)

Date :

Like this we can apply following preprocessing on the data set

(1) Add : To add attribute. It is present in unsupervised data. In that we can enter attribute index, type, data format, nominal label values.

(2) Remove : To remove attributes, select them and click on the remove button at the bottom. The selected attributes would be removed from db.

(3) Normalization :- It is in unsupervised data. It is good technique to use when you do not know the distribution of your data. You can normalized all of the your dataset.

Result :

The pre-processing on the given data set is executed.

Viva Questions :→

(1) Define preprocessing Technique ?

→ Preprocessing tools in WEKA are called Filters. The preprocess receives data from a file, SQL db or URL. Some preprocessing are Add, remove, normalization & etc..

(2) What are Data preprocessing steps ?

→ Step 1 : Open weka
Step 2 : load dataset from URL, SQL database or file.
Step 3 : click on filters
Step 4 : click on supervised
Step 5 : click on attribute
Step 6 : AttributeSelection or select the preprocessing technique ie. add, remove or etc..
Step 7 : click on Apply.

Date :

③ What are the 3 stages of data processing?

→ Stage 1: Loading data or dataset

Stage 2: understanding Data

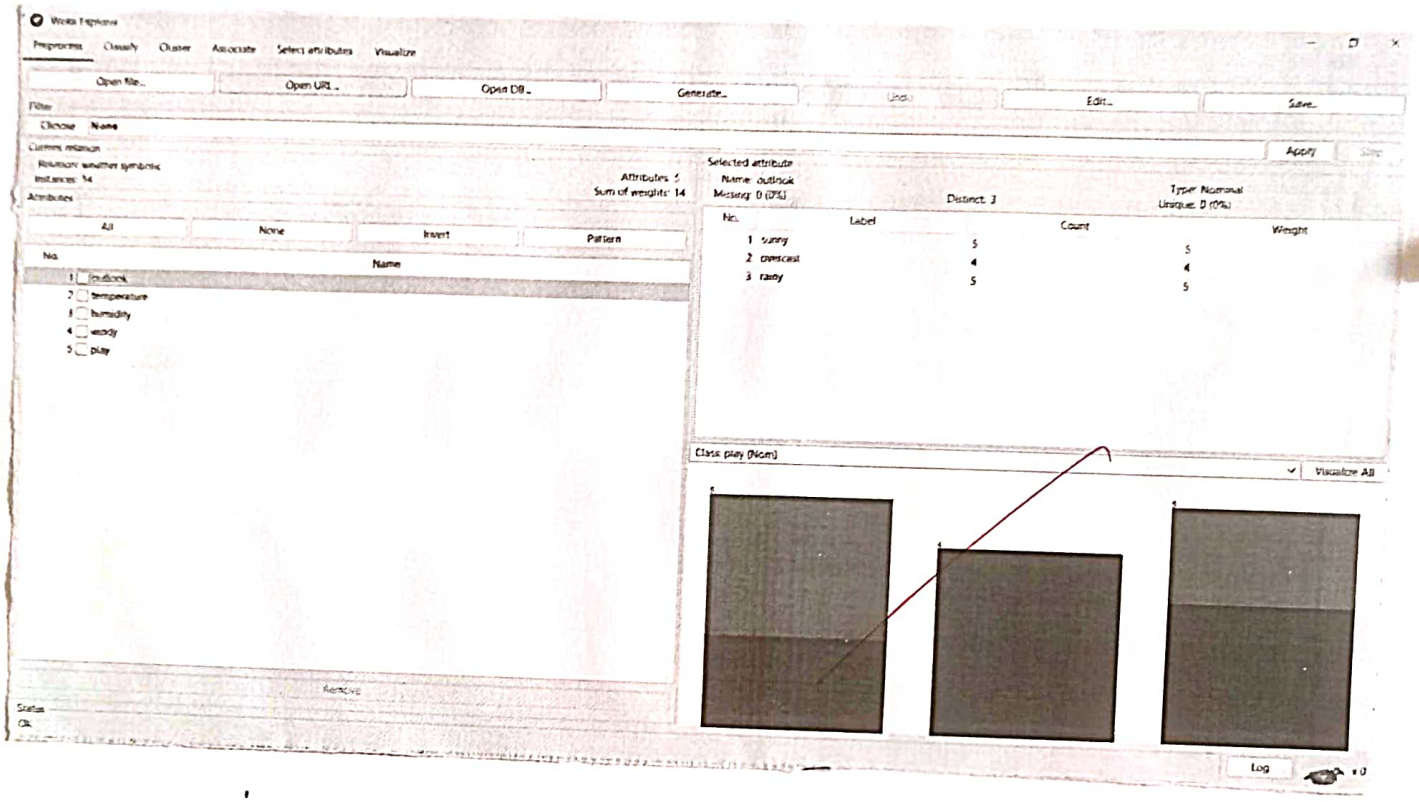
Stage 3: Apply Filters on dataset and Save.

15/02/23

A

Practical No. 03

Aim: To demonstrate performing classification on data set.



Date : 15/02/2023

Practical No. 03

Aim : To demonstrate performing classification on data sets.

Theory :

Classification :

Classification is the process for finding a model that describe the data values and concept for the purpose of prediction.

Classification in data mining is a common technique that separates data points into different classes.

It allows you to organize data set of all sorts, including complex and large datasets as well as small and simples ones. It primarily involves using algorithms that you can easily modify to improve the data quality. The algorithm establishes the link between the variables for predictions. The algorithm you use for classification in data mining is called the classifier, and observations you make through the same are called instances.

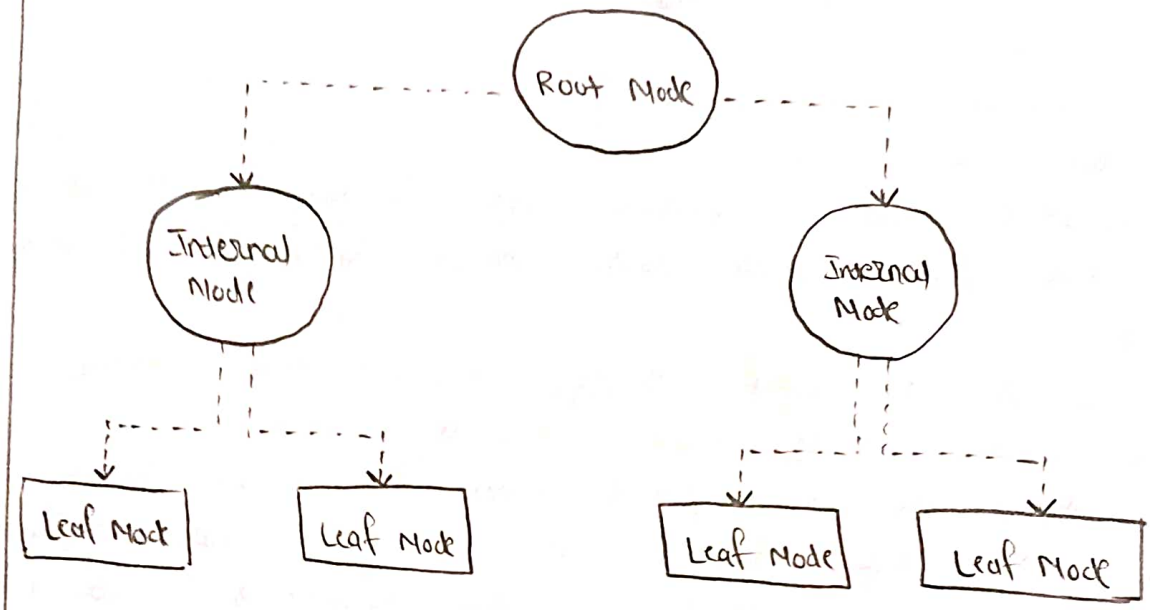
There are multiple type of classification algorithm, each with its unique functionality and application. All of these algorithm are used to extract data from a dataset.

Data Mining Algorithm for classification :

- Decision Trees
- Logistic Regression
- Naive Bayes Classification
- k - nearest neighbors
- Support Vector Machine.

Decision Tree :

A decision tree is a structure that includes a root node, branches and leaf nodes. Each internal node denotes a test on an


Root Node
Internal Node
Internal Node
Leaf Node
Leaf Node
Leaf Node
Leaf Node

Date :

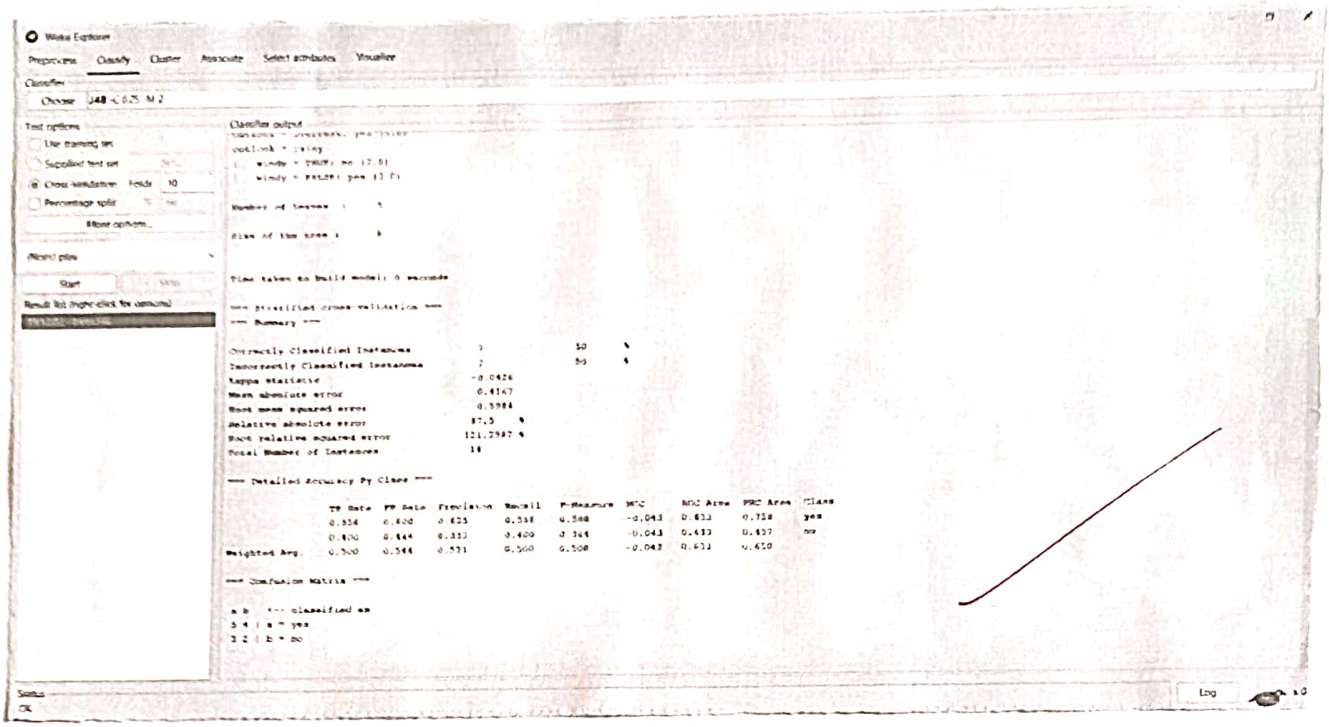
attribute, each branch denotes the outcome of a test and each leaf node holds a class label. The topmost node in the tree is the root node.

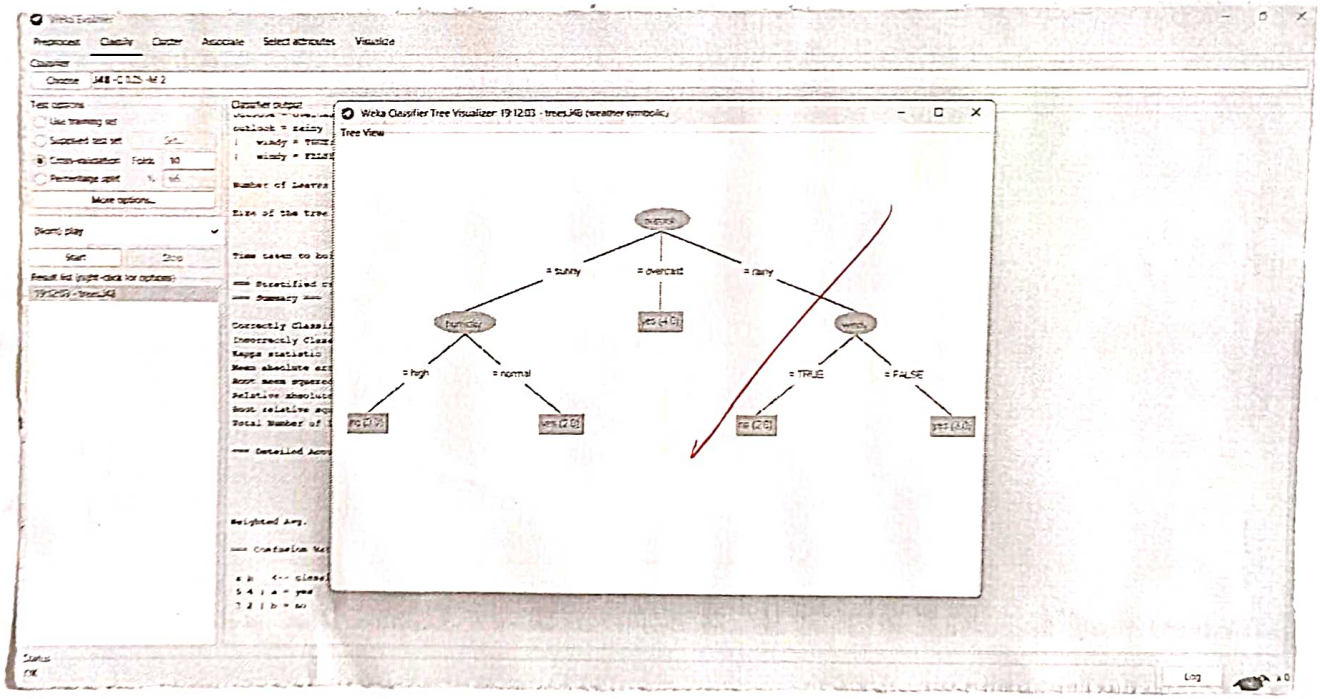
A decision tree is a classification scheme to generate a tree consisting of root node, internal nodes and external nodes. Root nodes representing the attributes. Internal nodes are also the attributes. External nodes are the class and each branch represent the values of the attributes.

Decision tree also contains set of rules for a given data set; there are two subsets in decision tree, one is a training data set and second one is a testing data set. Training data set is previously classified data. Testing data set is newly generated data.

The benefits of having a decision tree are as follows :-

- It does not required any domain knowledge.
- It is easy to comprehend
- The learning and classification steps of a decision tree are simple and fast.
- J48 classification and its decision Tree
- C4.5 algorithm / J48
- The C4.5 algo. is a classification algo. which provides decision trees based on information theory. It is an extension of Ross Quinlan's earlier ID3 algorithm also known in weka as J48, J standing for Java. The decision trees generated by C4.5 are used for classification, and for this reason, C4.5 is often referred to as a statistical classifier.
- The J48 implementation of the C4.5 algo. has many additional features including accounting for missing values, decision trees pruning, continuous attribute value range, derivation of rule and etc..
- In WEKA, data mining tool, J48 is an open Page source Java


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier output
Correctly Classified Instances 9 50 %
Incorrectly Classified Instances 9 50 %
Kappa statistic -0.0426
Mean absolute error 0.4167
Root mean squared error 0.5984
Relative absolute error 87.5 %
Root relative squared error 121.2987 %
Total Number of Instances 18
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.556 0.600 0.625 0.556 0.588 -0.043 0.433 0.738 yes
0.400 0.444 0.333 0.400 0.364 -0.043 0.433 0.457 no
Weighted Avg. 0.500 0.544 0.521 0.500 0.508 -0.043 0.433 0.610
=== Confusion Matrix ===
a b <-- classified as
5 4 | a = yes
5 4 | b = no
Status
OK
Log


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Weka Classifier Tree Visualizer: 19:12:03 - trees.J48 (weather.symbolic)
Tree View
outlook
= sunny
= overcast
= rainy
humidity
windy
= high
= normal
= TRUE
= FALSE
Status
OK
Log

Date :

implementation of the C4.5 algorithm. J48 allows classification via either decision trees or rules generated from them.

Generating a decision tree from training tuples of data partition D
Algorithm : Generate_decision_tree
Input :
Data partition, D, which is a set of training tuples and their associated class labels.
attribute_list, the set of candidate attributes.
Attribute selection method, a procedure to determine the splitting criterion that best partitions that the data tuples into individual classes. This criterion includes a splitting_attribute and either a splitting point or splitting subset.

Output :
A Decision Tree

Method :
create a node N;
if tuples in D are all of the same class, C then
return N as leaf node labeled with class C;

if attribute_list is empty then
return N as leaf node with labeled
with majority class in D; // majority voting

apply attribute_selection_method(D, attribute_list)
to find the best splitting_criterion;
label node N with splitting_criterion;

if splitting_attribute is discrete-valued and
multiway split allowed then // no restriction to binary trees.

Result :

Thus, the classification on Data set is performed by decision tree (J48) method.

Date :

attribute_list = splitting attribute // remove splitting attribute
for each outcome j of splitting criterion

// partition the tuples and grow subtrees for each partitions
let Dj be the set of data tuples in D satisfying outcome j ;
// a partition

if Dj is empty then
    attach a leaf labeled with the majority
    class in D to node N;
else
    attach the node returned by Generate
    decision tree (Dj, attribute list) to node N;
end for
return N;

Result:

Thus the classification on data set is performed by decision tree (ID3) method.

Viva Question :

① what is classification ?

→ classification is the process for finding a model that describe the data values and concept for the purpose of prediction.

② what is the need of classification ?

→ The need of classification is to accurately predict the target class for each case in the data, it allows you to organize data set of all sorts, including complex and large datasets as well as small and simples ones.

Date :

Q.3 What are the different methods of classification?

→ Logistic regression, Naive Bayes, Decision tree, k-Near Neighbours, Support vector machine, Bayes classifiers, function classifier, lazy classifier, meta classifier and so on.

Q.4 What are the advantages of a decision tree classifier?

→ (1) It is easy to comprehend

(2) It does not require any domain knowledge.

(3) The learning and classification steps of decision tree are simple and fast

(4) less data preparation, Non-parametric versatility, Non-Linearity.

02/03/23

A

Practical No: 4

Aim: To demonstrate classification rule process on dataset using naives bayes algorithm.

Date :

Practical No. 4

Aim : To demonstrate classification rule proces on datasets using naives bayes algorithm.

Theory :

Naive Bayes classifier Algorithm :

- Naive Bayes algo. is a supervised learning algo., which is based on Bayes theorem and used for solving classification problems.
- It is mainly used in text classification that includes a high-dimensional training datasets.
- Naive Bayes classifier is one of the simple and most effective classification algo.
- It is a probabilistic classifier, which means it predict on the basis of the probability of an object.
- Some popular example of Naive Bayes algo. are spam filtration, sentimental analysis and classifying articles.

Why is it called Naives Bayes ?

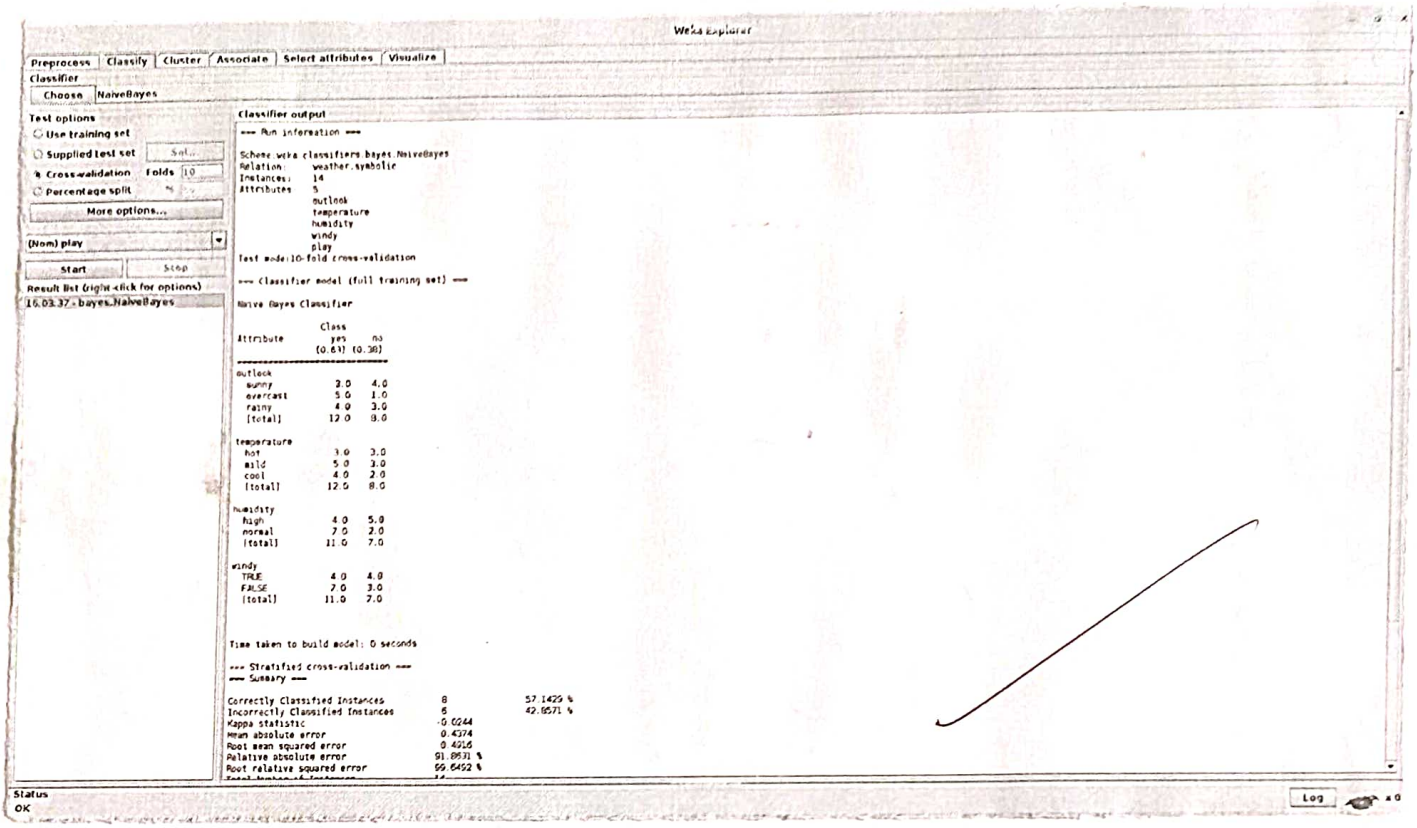
The Naive Bayes algo. is comprised of two words Naive and Bayes, which can be described as :

Naive : It is called Naive because it assume that the occurence of a certain feature is independent of the occurence of other features. Such as if the fruit is identified on the bases of color, shape and taste then red, spherical, and sweet fruit is identified or recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.

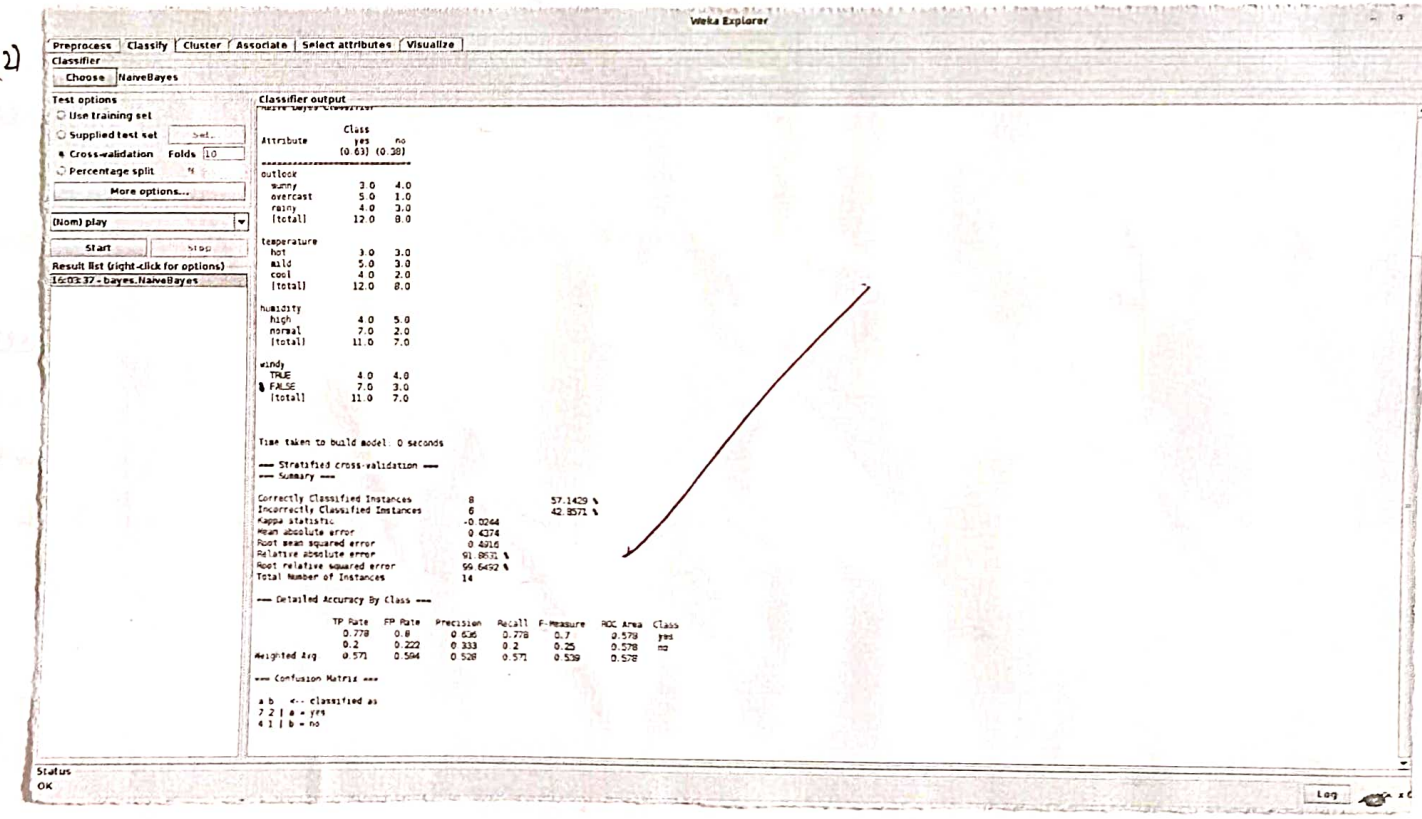
Bayes : It called Bayes because it depends on the principle of Bayes theorem.

output :

(1)


Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Classifier
Choose NaiveBayes
Test options
Use training set
Supplied test set Set...
Cross-validation Folds 10
Percentage split %
More options...
(Nom) play
Start Stop
Result list (right-click for options)
16:03:37 - bayes.NaiveBayes
Classifier output
=== Run information ===
Scheme:weka.classifiers.bayes.NaiveBayes
Relation: weather.symbolic
Instances: 14
Attributes: 5
outlook
temperature
humidity
windy
play
Test mode:10-fold cross-validation
=== Classifier model (full training set) ===
Naive Bayes Classifier
Class
Attribute yes no
(0.63) (0.38)
outlook
sunny 3.0 4.0
overcast 5.0 1.0
rainy 4.0 3.0
[total] 12.0 8.0
temperature
hot 3.0 3.0
mild 5.0 3.0
cool 4.0 2.0
[total] 12.0 8.0
humidity
high 4.0 5.0
normal 7.0 2.0
[total] 11.0 7.0
windy
TRUE 4.0 4.0
FALSE 7.0 3.0
[total] 11.0 7.0
Time taken to build model: 0 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 8 57.1429 %
Incorrectly Classified Instances 6 42.8571 %
Kappa statistic -0.0244
Mean absolute error 0.4374
Root mean squared error 0.4916
Relative absolute error 91.8631 %
Root relative squared error 99.6492 %
Status
OK
Log

(2)


Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Classifier
Choose NaiveBayes
Test options
Use training set
Supplied test set Set...
Cross-validation Folds 10
Percentage split %
More options...
(Nom) play
Start Stop
Result list (right-click for options)
16:03:37 - bayes.NaiveBayes
Classifier output
Attribute Class
yes no
(0.63) (0.38)
outlook
sunny 3.0 4.0
overcast 5.0 1.0
rainy 4.0 3.0
[total] 12.0 8.0
temperature
hot 3.0 3.0
mild 5.0 3.0
cool 4.0 2.0
[total] 12.0 8.0
humidity
high 4.0 5.0
normal 7.0 2.0
[total] 11.0 7.0
windy
TRUE 4.0 4.0
FALSE 7.0 3.0
[total] 11.0 7.0
Time taken to build model: 0 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 8 57.1429 %
Incorrectly Classified Instances 6 42.8571 %
Kappa statistic -0.0244
Mean absolute error 0.4374
Root mean squared error 0.4916
Relative absolute error 91.8631 %
Root relative squared error 99.6492 %
Total Number of Instances 14
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure ROC Area Class
0.778 0.8 0.636 0.778 0.7 0.578 yes
0.2 0.222 0.333 0.2 0.25 0.578 no
Weighted Avg 0.571 0.594 0.528 0.571 0.539 0.578
=== Confusion Matrix ===
a b <-- classified as
7 2 | a = yes
4 1 | b = no
Status
OK
Log

Date :

Bayes' theorem:

- Bayes theorem is also known as Bayes' rule or Bayes law, which is used to determine the probability of a hypothesis with prior knowledge. It depend on the conditional probability.
- The formula for Bayes theorem is given as:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

where,

P(A|B) is posterior probability : probability of hypothesis A on the observed event B.

P(B|A) is likelihood probability : probability of the evidence given that the probability of a hypothesis true.

P(A) is prior probability: probability of hypothesis before observing the evidence.

P(B) is marginal probability : probability of evidence.

working of Naive Bayes' classifier:

working of Naive Bayes classifier can be understood with the help of the below example:

Suppose we have dataset of weather conditions and corresponding target variable "play". So using this dataset we need to decide that whether we should play or not on a particular day according to the weather condition. So to solve this problem, we need to follow the below steps:

- Convert the given dataset into frequency tables.
- Generate likelihood table by finding the probabilities of given frequency.
- Now, use Bayes theorem to calculate the posterior probability.

ep 1: Initially, we have to load the required dataset in the weka tool using choose file options. Here we selecting the weather - nominal dataset to execute.

Result :

Thus, the classification on dataset is performed by Naives Bayes classification.

Date :

Step 2: Now we have to go the classify tab on the left side and click on the choose button and select the naive Bayesian algorithm in it.

Step 3: Now to change the parameters click on the right side at the choose button and we are accepting the default values in this example.

Step 4: We choose the percentage split as our measurement method from the "Test" choices in the main panel. since we don't have a separate test data collection, we'll use the percentage split of 66 percent to get a good idea of the model's accuracy. our dataset contains 14 examples, with 9 being used for training and 5 being used for testing.

Step 5: To generate the model, we now click 'start'. when the model is done, the evaluation statistic will appear in right panel.

Result:

Thus the classification on data set is performed by Naives bayes classification.

Viva Questions:

(1) what is a Naive Bayes classifier?

→ A naive Bayes classifier is an algo. that uses Bayes' theorem to classify objects. Naive Bayes classifiers assume strong or naive, independence between attributes of data points.

(2) what are the basic assumption?

→ The basic assumption in naive Bayes is one of conditional independence between all independent variable features.

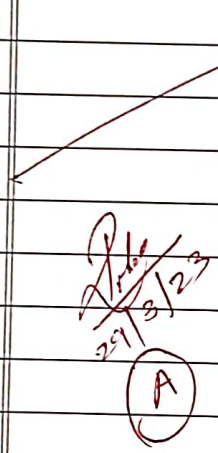
Date :

Ques.3) what are the advantages of Naive Bayes classifier?

→ ① It is simple and easy to implement.
② It doesn't require as much training data.
③ It handle both discrete and continuous data.
④ It is fast and can be used to make real time predictions.
⑤ It is not sensitive to irrelevant features.

Ques.4) Name the different problem statement you can solve using Naive Baye's.

→ ① A fruit may be considered to be an apple if it is red, round and about 3 inch in diameter.
② If the weather is sunny, then the player should play or not?



Date :

Practical No. 5

Aim: To implement k-means algorithm.

Descriptions :

Clustering :→

Clustering is the method of dividing a set of abstract objects into groups. points to keep in mind A set of data objects can be viewed as a single entity. when performing cluster analysis, we divide the data set into groups based on data similarity, then assign labels to the groups.

Simple k means clustering :→

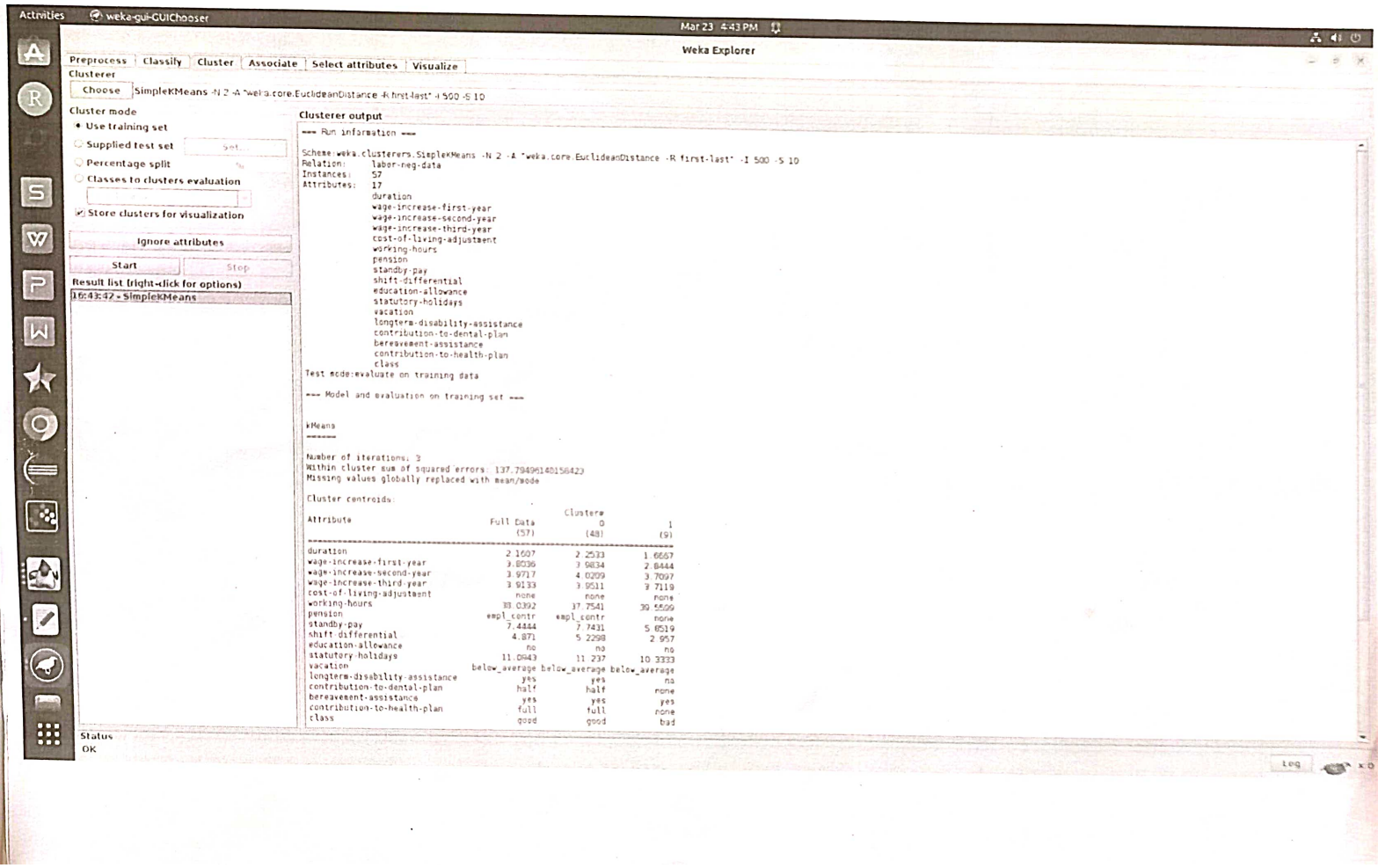
K means clustering is a simple unsupervised learning algorithm. In this, the data objects ('n') are grouped into a total 'k' clusters, with each observation belonging to the cluster with closest mean. It defines 'k' sets, one for each clusters k n . The clusters are separated by a large distance.

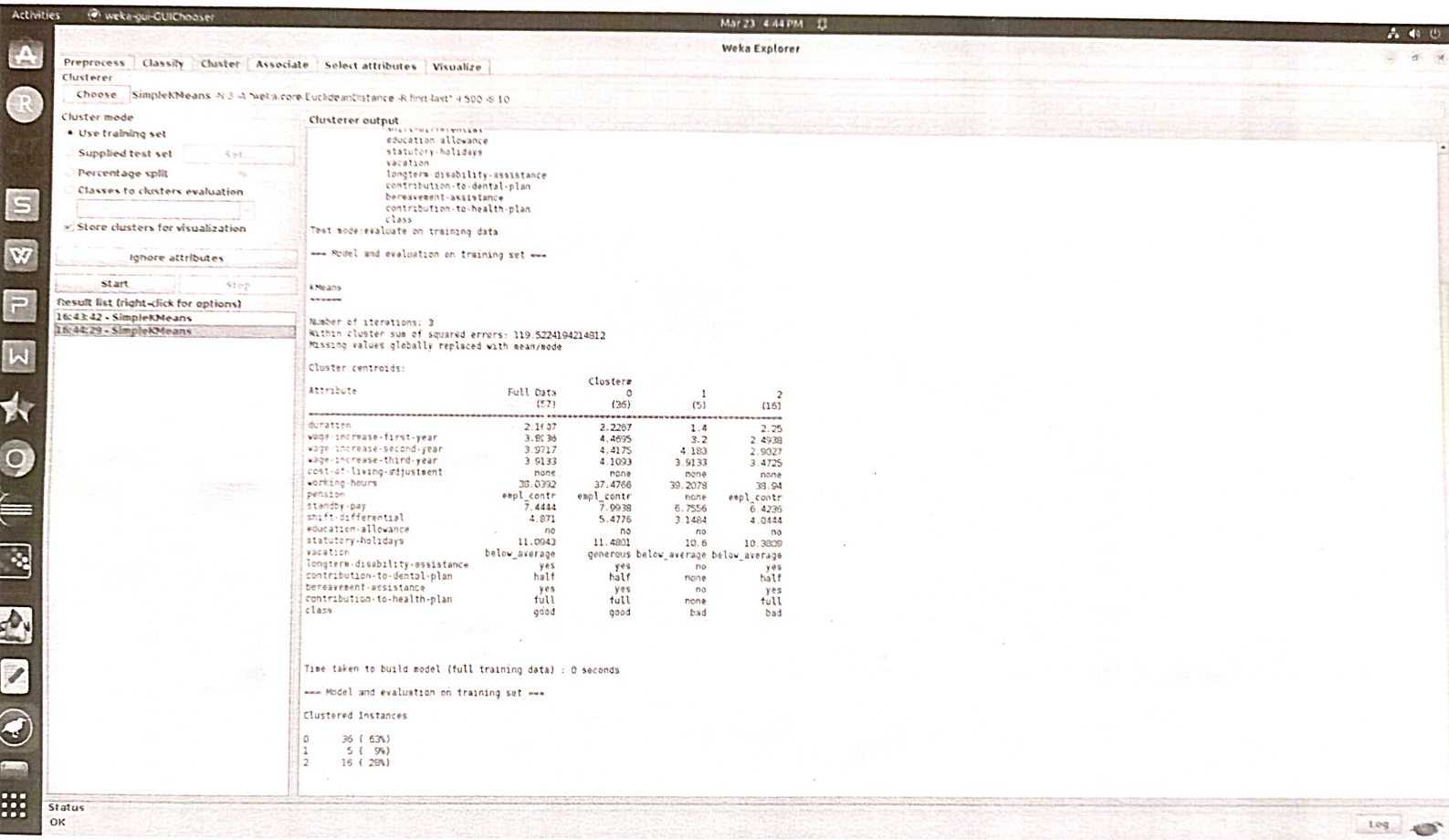
The data is then organized into acceptable data sets and linked to the nearest collection. If no data is pending, the first stage is more difficult to complete; in the case, an early grouping is performed. The 'k' new set must be recalculated as the barycenters of the clusters from the previous stage.

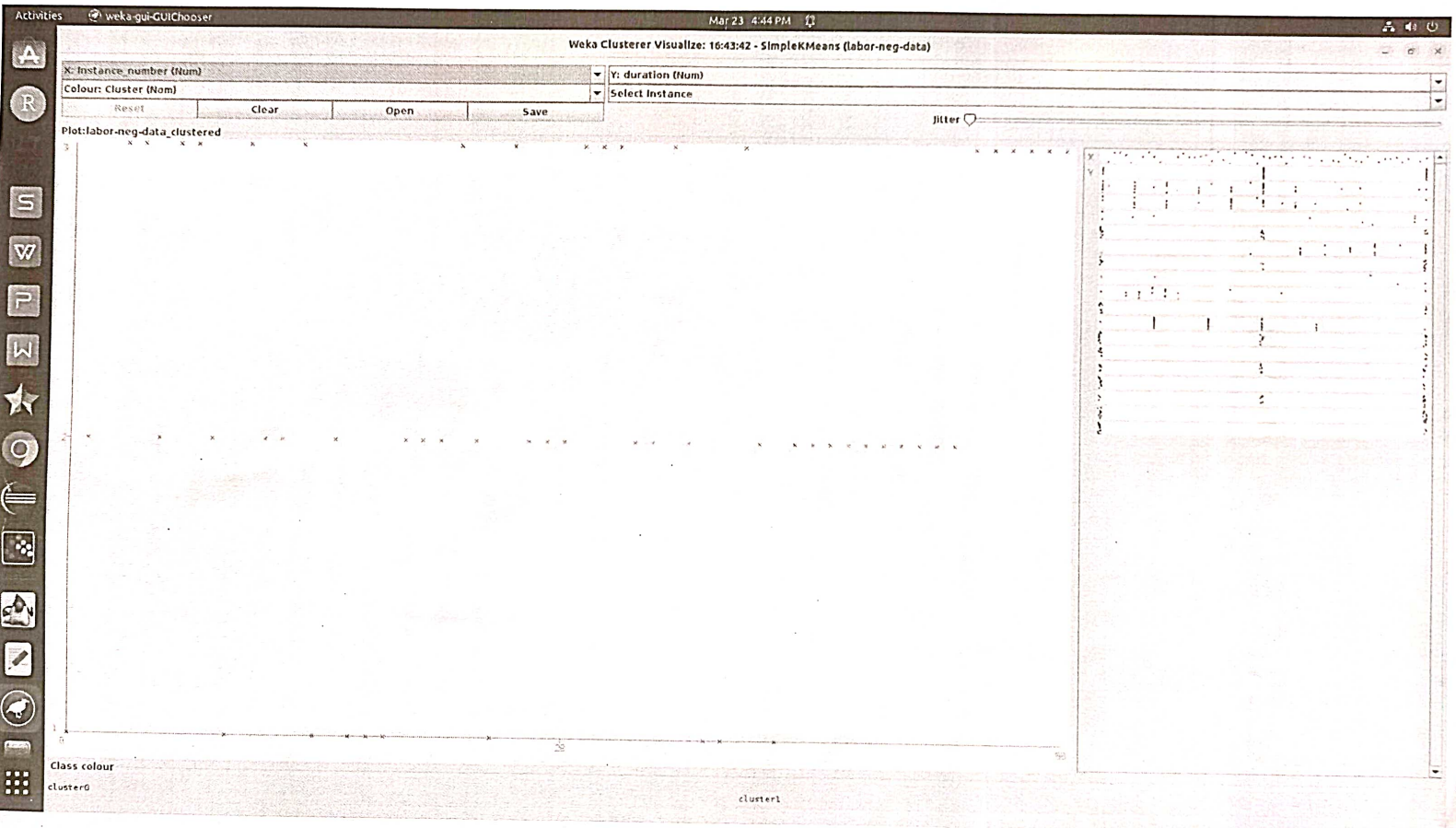
The same data set points and the nearest new sets are bound together after these 'k' new sets have been created. After that, a loop is created. The 'k' sets change their position step by step until no further changes are made as a result of this loop.

K-means clustering algo. computers the centroids and iterates until we it finds optimal centroid. It assumes that the no. of clusters are already known. It is also called flat clustering algo. The no. of clusters identified from data by algo. is represented by 'k' in k-means.

In this algo. the data points are assigned to a cluster in such a manner that the sum of the squared distance between the


Activities
weka-gui-GUIChooser
Mar 23 4:43 PM
Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Clusterer
Choose
SimpleKMeans -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10
Cluster mode
Use training set
Supplied test set
Set...
Percentage split
%
Classes to clusters evaluation
Store clusters for visualization
Ignore attributes
Start
Stop
Result list (right-click for options)
16:43:42 - SimpleKMeans
Clusterer output
=== Run information ===
Scheme:weka.clusterers.SimpleKMeans -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10
Relation: labor-neg-data
Instances: 57
Attributes: 17
duration
wage-increase-first-year
wage-increase-second-year
wage-increase-third-year
cost-of-living-adjustment
working-hours
pension
standby-pay
shift-differential
education-allowance
statutory-holidays
vacation
longterm-disability-assistance
contribution-to-dental-plan
bereavement-assistance
contribution-to-health-plan
class
Test mode:evaluate on training data
=== Model and evaluation on training set ===
kMeans
Number of iterations: 3
Within cluster sum of squared errors: 137.79496140156423
Missing values globally replaced with mean/mode
Cluster centroids:
Attribute Full Data Cluster# 0 1
(57) (48) (9)
duration 2.1607 2.2533 1.6667
wage-increase-first-year 3.8596 3.9834 2.8444
wage-increase-second-year 3.9717 4.0209 3.7097
wage-increase-third-year 3.9133 3.9511 3.7119
cost-of-living-adjustment none none none
working-hours 38.0392 37.7541 39.5699
pension empl_contr empl_contr none
standby-pay 7.4444 7.7431 5.6519
shift-differential 4.871 5.2298 2.957
education-allowance no no no
statutory-holidays 11.0943 11.237 10.3333
vacation below_average below_average below_average
longterm-disability-assistance yes yes na
contribution-to-dental-plan half half none
bereavement-assistance yes yes yes
contribution-to-health-plan full full none
class good good bad
Status
OK
Log
x 0


Activities
weka-gui-GUIChooser
Mar 23 4:44 PM
Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Clusterer
Choose
SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10
Cluster mode
Use training set
Supplied test set
Set...
Percentage split
%
Classes to clusters evaluation
Store clusters for visualization
Ignore attributes
Start
Stop
Result list (right-click for options)
16:43:42 - SimpleKMeans
16:44:29 - SimpleKMeans
Clusterer output
education-allowance
statutory-holidays
vacation
longterm-disability-assistance
contribution-to-dental-plan
bereavement-assistance
contribution-to-health-plan
class
Test mode:evaluate on training data
=== Model and evaluation on training set ===
kMeans
======
Number of iterations: 3
Within cluster sum of squared errors: 119.52241942148 12
Missing values globally replaced with mean/mode
Cluster centroids:
Attribute Full Data (57) Cluster# 0 (36) 1 (5) 2 (16)
duration 2.1137 2.2287 1.4 2.25
wage-increase-first-year 3.8636 4.4695 3.2 2.4938
wage-increase-second-year 3.9717 4.4175 4.183 2.9027
wage-increase-third-year 3.9133 4.1093 3.9133 3.4725
cost-of-living-adjustment none none none none
working-hours 38.0392 37.4766 39.2078 38.94
pension empl_contr empl_contr none empl_contr
standby-pay 7.4444 7.9938 6.7556 6.4236
shift-differential 4.871 5.4776 3.1484 4.0444
education-allowance no no no no
statutory-holidays 11.0943 11.4801 10.6 10.3809
vacation below_average generous below_average below_average
longterm-disability-assistance yes yes no yes
contribution-to-dental-plan half half none half
bereavement-assistance yes yes no yes
contribution-to-health-plan full full none full
class good good bad bad
Time taken to build model (full training data) : 0 seconds
=== Model and evaluation on training set ===
Clustered Instances
0 36 (63%)
1 5 (9%)
2 16 (28%)
Status
OK
Log


Activities
weka-gui-GUIChooser
Mar 23 4:44 PM
Weka Clusterer Visualize: 16:43:42 - SimpleKMeans (labor-neg-data)
X: instance_number (Num)
Y: duration (Num)
Colour: Cluster (Nom)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot:labor-neg-data_clustered
Class colour
cluster0
cluster1

Date :

data points and centroid would be minimum. It is to be understood that less variation within the clusters will leads to more similar data points within same clusters.

working of k-means Algorithm:→

We can understand the working of k-means clustering algo. with the following steps:

Step 1: First, we need to specify the number of clusters, K, need to be generated by this algorithm.

Step 2: Next, randomly select K data points and assign each data point to a cluster.

Step 3: Now it will compute the cluster centroids

Step 4: Next, keep iterating the following until we find optimal centroid which is the assignment of data points to the clusters that are not changing any more.

Result:

This program has been successfully executed.

Viva Questions:

① what is the purpose of k-means algorithm?

→ It is used to find group which have not been explicitly labeled in the data. It is unsupervised learning algo. which groups the unlabeled datasets into different clusters.

② How do we use k means clustering algo. in weka?

→ step 1: open weka and load dataset.

step 2: Find cluster tab in explorer and choose button to execute clustering and select simple - k-means algo.

step 3: Then to the right of choose icon, press text button and enter three for number of clusters and leave the seed value alone.

Date :

Step 4: The choice to use training set is selected and then 'start' button is pressed.

Step 5: The centroid of each cluster is shown in result window and right click the result set on result, selecting to visualize cluster assignment from list column.

(3) what does k-mean in k-mean algorithm?

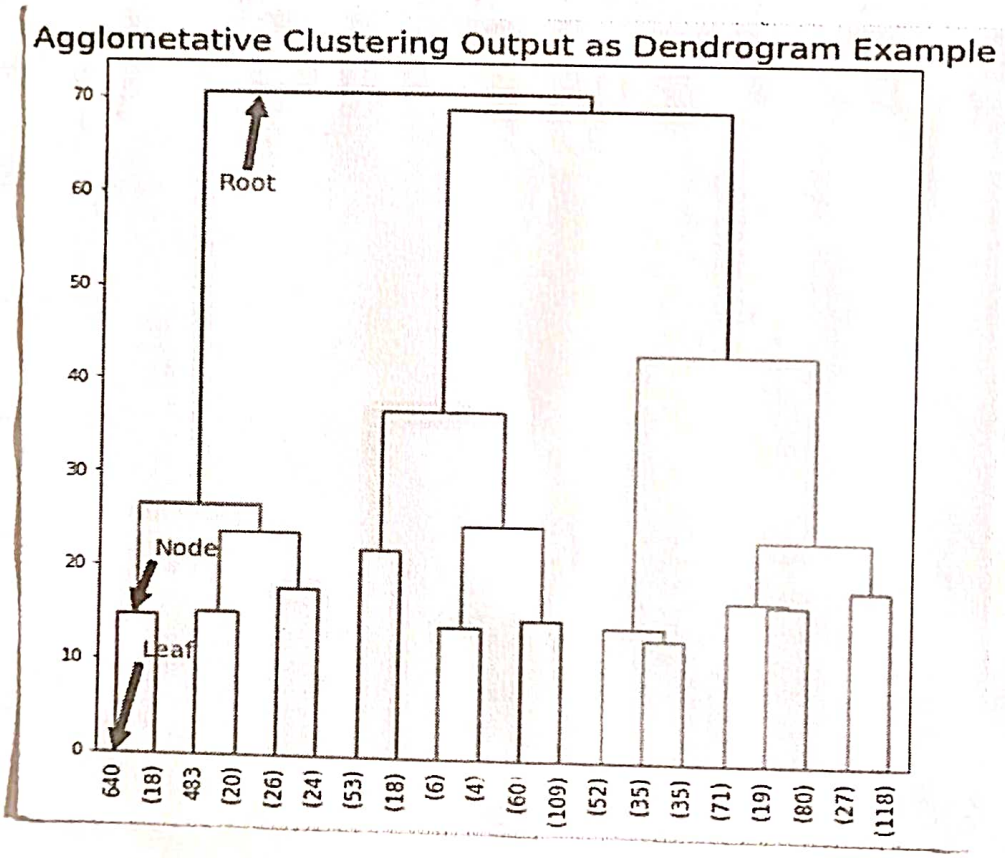
→ The number of clusters found from data by the method is denoted by the letter 'k' in K-means.

(4) what type of algorithm is k-means?

→ It is simple unsupervised learning algorithm. It is widely used centroid-based clustering algorithm.

Practical No·6

Aim : To demonstrate hierarchical clustering on given data set



Date :

Practical No. 6

Aim: To demonstrate hierarchical clustering on given data set.

Theory:

Clustering is one of the most common exploratory data analysis techniques used to get an intuition about the structure of the data. It can be defined as the task of identifying subgroups in the data such that data point in the same subgroup (cluster) are very similar while data point in different clusters are very different. In otherwords, we try to find homogeneous subgroups within the data such that data points in each cluster are as similar as possible acc to the similarity measure such as euclidean-based distance or correlation based distance. The decision of which similarity measure to used is application-specific.

Hierarchical clustering :→

It is also known as Hierarchical cluster analysis or HCA, is an unsupervised clustering approach that includes forming group with a top-to-bottom order.

The hierarchical clustering techniques has two approaches:

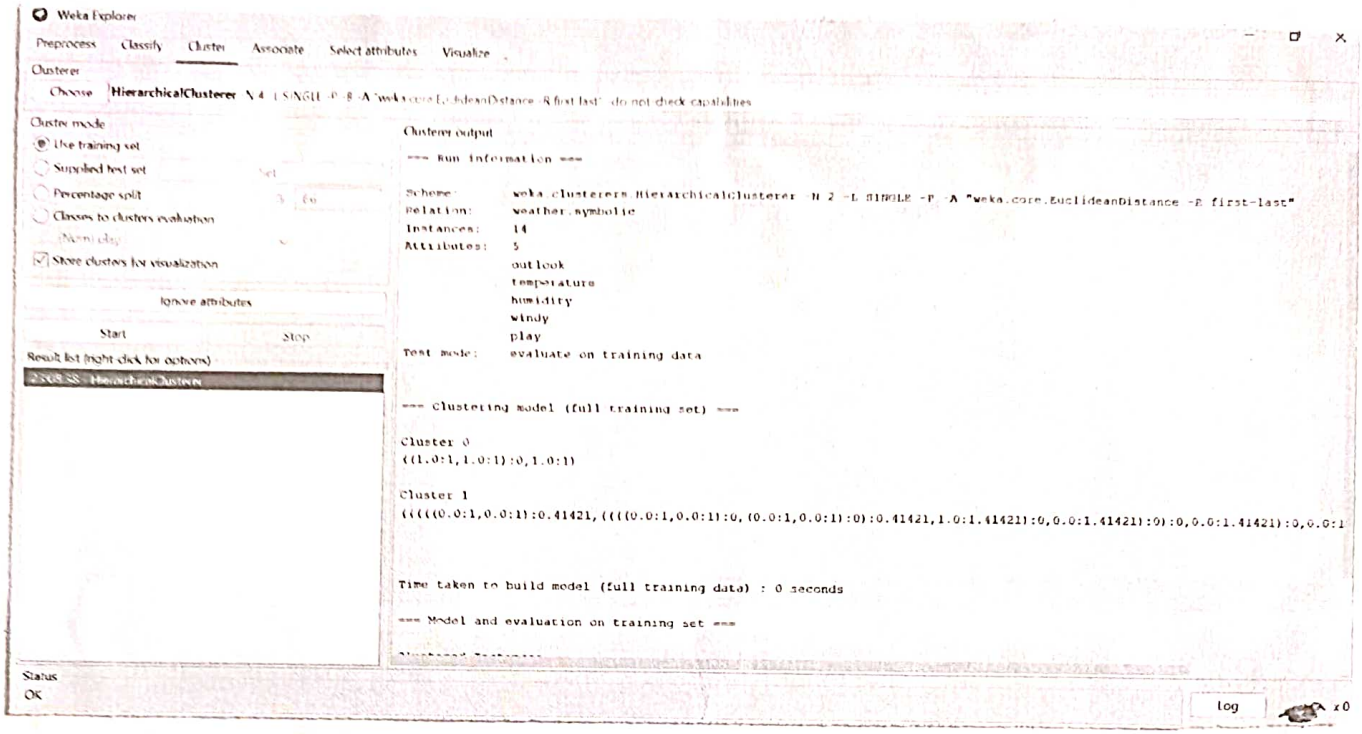
① Agglomerative : It is bottom up approach, in which the algo starts with taking all data points as single clusters and merging them until one cluster is left.

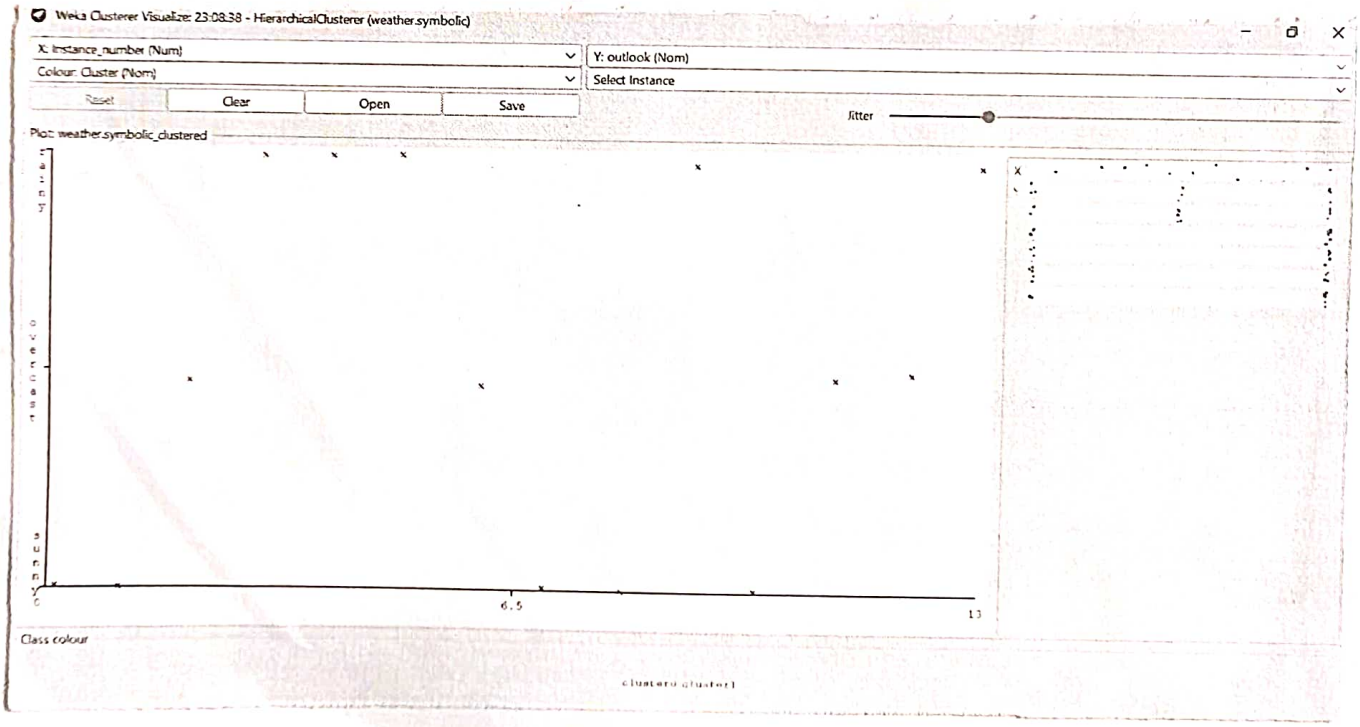
② Divisive : It is the reverse of agglomerative algo. as it is a top-down approach.

Agglomerative hierarchical clustering :→

Agglomerative clustering or bottom up clustering, essentially started from an individual cluster (each data point is considered as an individual cluster, also called leaf), then every cluster calculates their distance with each other. The

Page No.:





Conclusion: Thus, we have implemented hierarchical algo. for clustering successfully

Date :

two clusters with the shortest distance with each other would merge creating what we called node. Newly formed clusters once again calculating the member of their cluster distance with another cluster outside of their cluster. The process is repeated until all the data points assigned to one cluster called root. The result is a tree-based representation of the objects called dendrogram.

Conclusion :

Thus we have implemented hierarchical algorithm for clustering successfully.

Viva Voce :

① what is hierarchical clustering Algorithm ?

→ Hierarchical clustering, also known as hierarchical cluster analysis or HCA, is an unsupervised clustering approach that include forming groups with a top-to-down (bottom) order.

② what are the various types of hierarchical clustering?

→ There are two types of hierarchical clustering, They are:

a) Agglomerative

b) Divisive.

③ What as dendrogram in HCA ?

→ Dendrogram is a diagram that shows the hierarchical relationship betn objects.

④ Explain the different linkage method used in HCA ?

→ ① Single linkage :→

Single linkage returns minimum distance betn two point, where each points belong to two different clusters.

Date :

② Complete linkage :→
It returns the maximum distance betⁿ each data point.

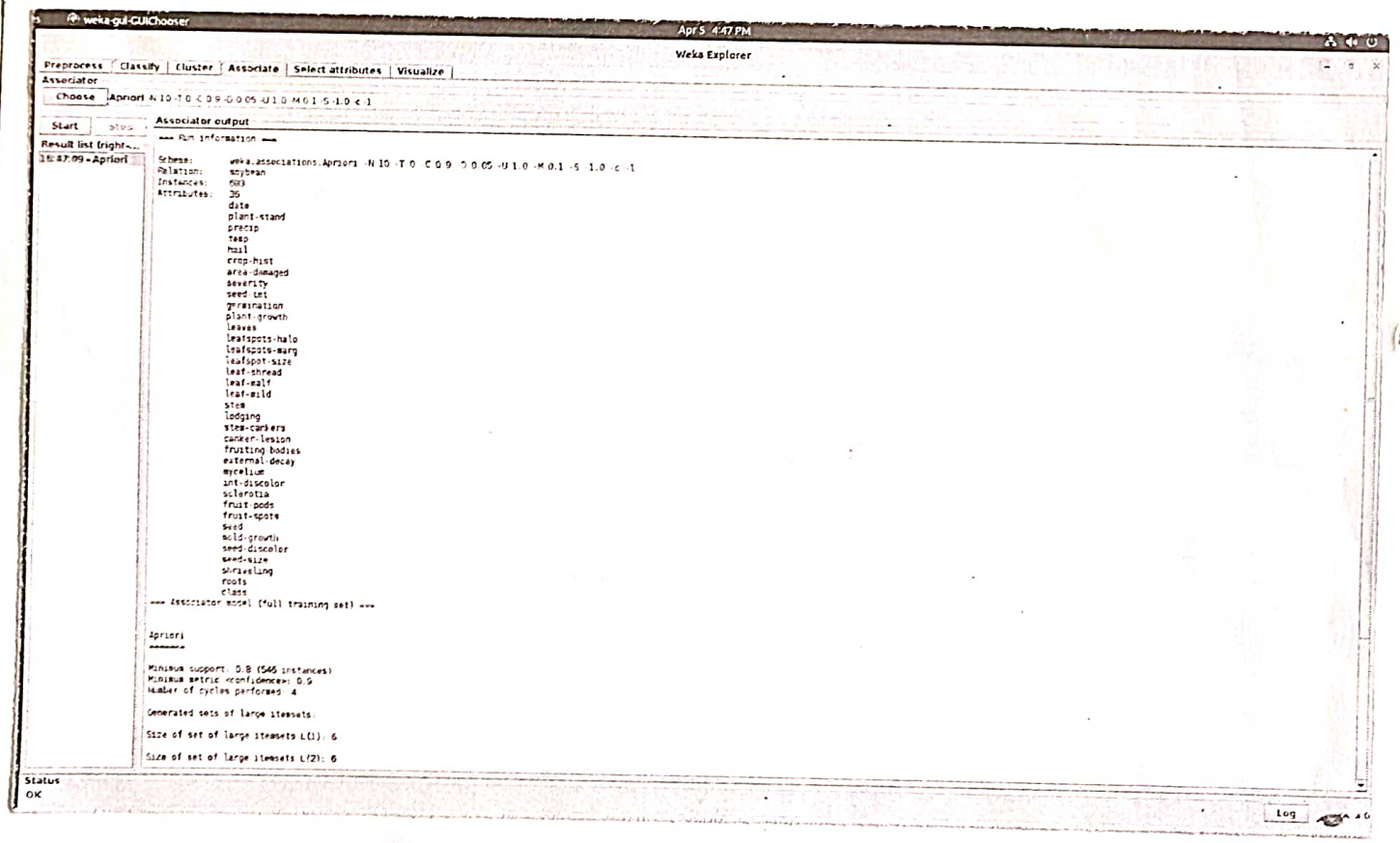
③ Average linkage :→
It returns the average of distances betⁿ all pairs of data points.

④ Centroid linkage :→
It returns the distance betⁿ centroid of clusters.

Practical 7

Aim: Implement Apriori Algorithm.

Transaction ID	onion	Potato	Burger	milk	Beer
t1	1	1	1	0	0
t2	0	1	1	1	0
t3	0	0	0	1	1
t4	1	1	0	1	0
t5	1	1	1	0	1



Date :

Practical No. 7

Aim: Implement Apriori Algorithm.

Theory:

In general association rule mining can be viewed as a two-step process:

(1) Find all frequent itemset: By definition each of these itemsets will occur at least as frequently as a predetermined minimum support count min sup.

(2) Generate strong association rules from the frequency itemsets: By definitions these rule must satisfy minimum support and minimum confidence.

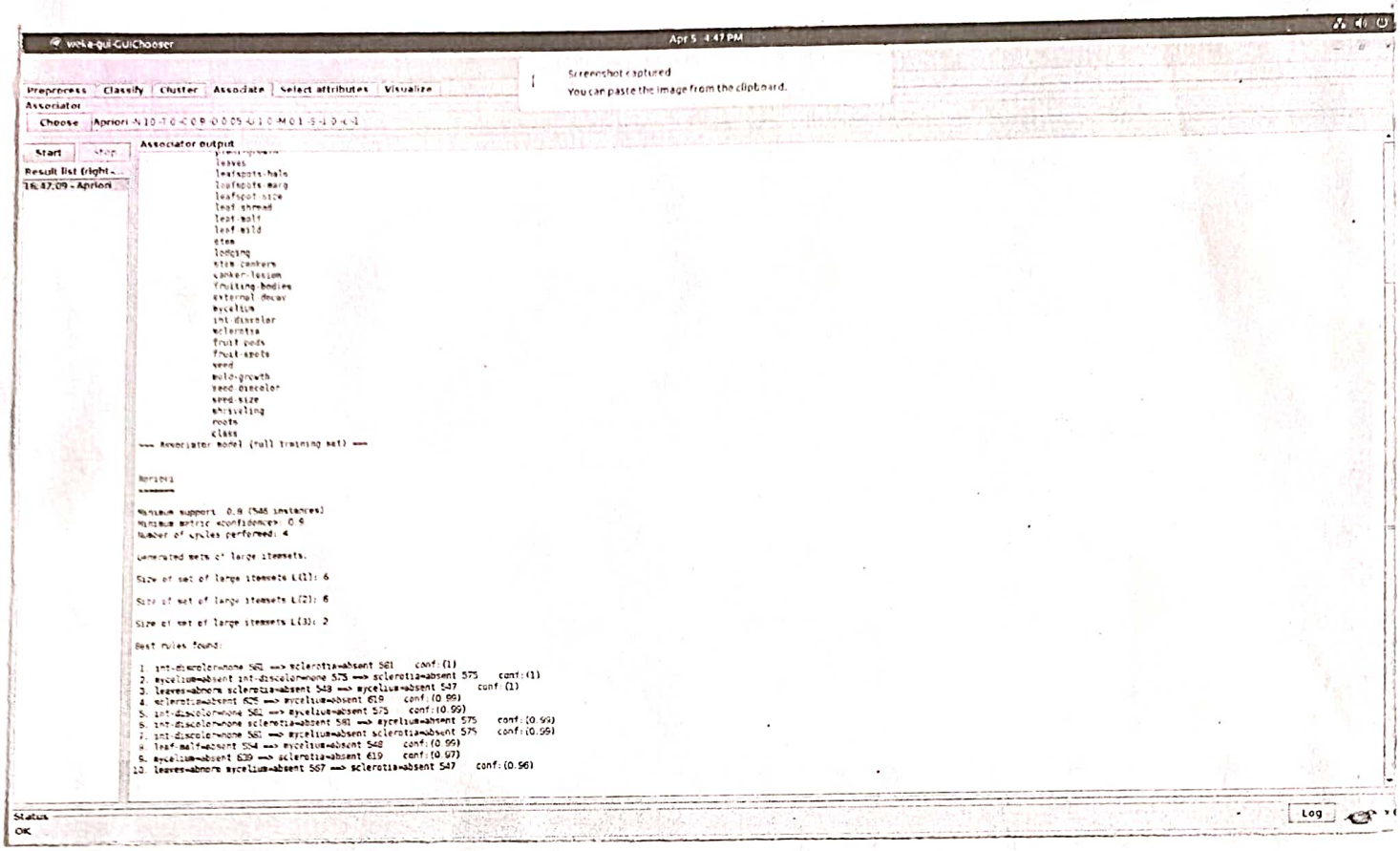
Let $I = \{i1, i2, i3, \ldots, in\}$ be a set of n attributes called items and $D = \{t1, t2, \ldots, tn\}$ be the set of transaction. It is called a database. Every transaction ti in D has a unique transaction ID, and it consist of subsets of itemset in I.

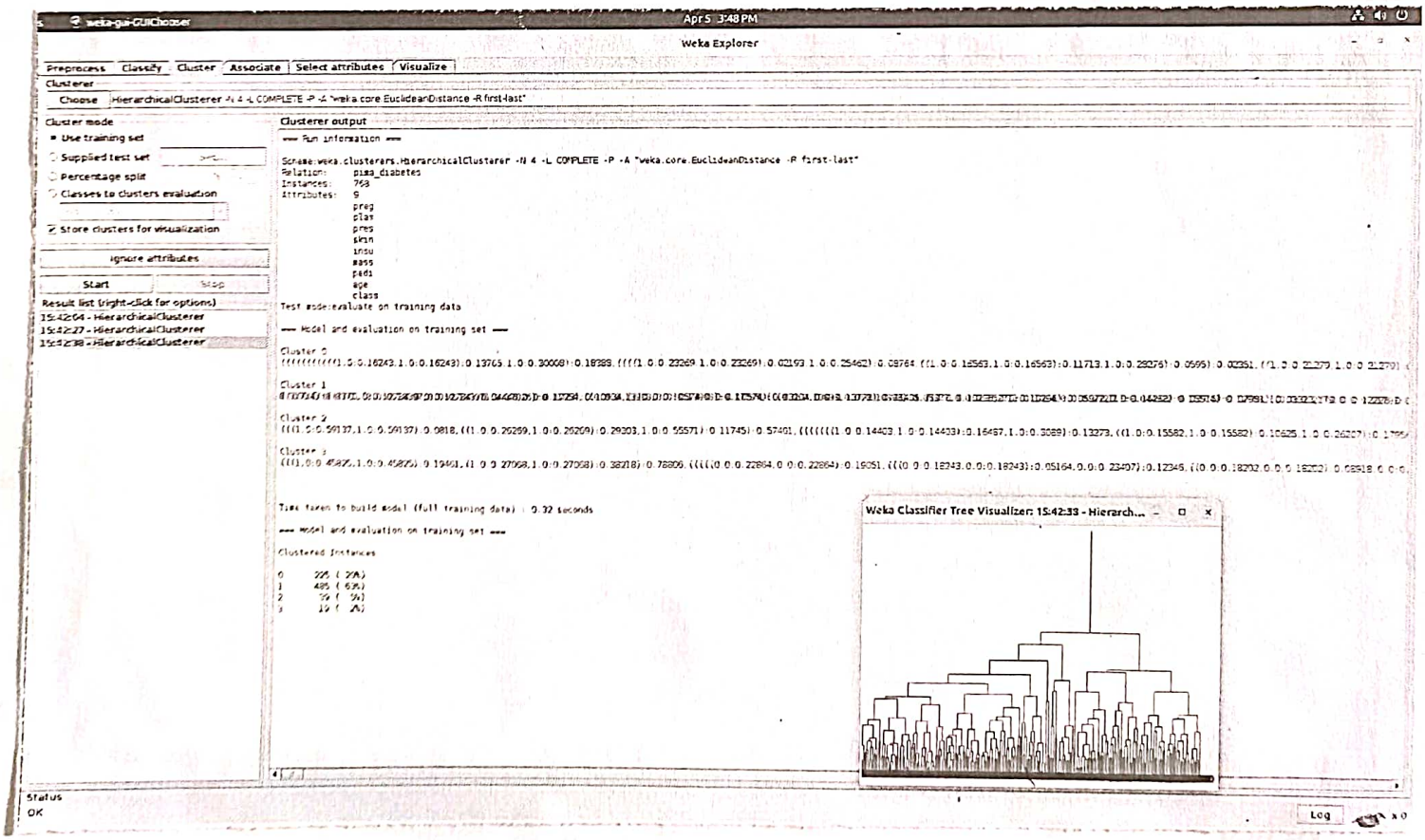
A rule can be defined as an implication, $X \rightarrow Y$ where X and Y are subsets of I ($X, Y \subseteq I$) and they have no element in common ie. $X \cap Y$, X and Y are the antecedent and the consequent of the rule, respectively.

Let's take an easy example from the supermarket sphere. The example that we are considering is quite small and in practical situation, datasets contain millions or billions of transactions. The set of itemsets, $I = \{$Onion, Burger, Potato, Milk, Beer$\}$ and a database consisting of six transactions. Each transaction is a tuple of 0's and 1's where 0 represents the absence of an item and 1 the presence.

An example for a rule in this scenario would be $\{$Onion, Potato$\} \Rightarrow \{$Burger$\}$, which means that if Onion and Potato are bought, customers also buy a burger.

There are multiple rules possible even from a very small db, so in order to select the interesting ones, we we constraints on various measures of interest and significance. we will look some


weka-gui-GUIChooser
Apr 5 4:47 PM
Preprocess Classify Cluster Associate Select attributes Visualize
Associator
Choose Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Start Stop
Result list (right-...
16:47:09 - Apriori
Associator output
leaves
leafspots-halo
leafspots-marg
leafspot-size
leaf-shread
leaf-malf
leaf-mild
stem
lodging
stem-cankers
canker-lesion
fruiting-bodies
external decay
mycelium
int-discolor
sclerotia
fruit-pods
fruit-spots
seed
mold-growth
seed-discolor
seed-size
shriveling
roots
class
=== Associator model (full training set) ===
Apriori
Minimum support: 0.9 (546 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 4
Generated sets of large itemsets:
Size of set of large itemsets L(1): 6
Size of set of large itemsets L(2): 6
Size of set of large itemsets L(3): 2
Best rules found:
1. int-discolor=none 561 ==> sclerotia=absent 561 conf:(1)
2. mycelium=absent int-discolor=none 575 ==> sclerotia=absent 575 conf:(1)
3. leaves=abnorm sclerotia=absent 548 ==> mycelium=absent 547 conf:(1)
4. sclerotia=absent 625 ==> mycelium=absent 619 conf:(0.99)
5. int-discolor=none 561 ==> mycelium=absent 575 conf:(0.99)
6. int-discolor=none sclerotia=absent 561 ==> mycelium=absent 575 conf:(0.99)
7. int-discolor=none 561 ==> mycelium=absent sclerotia=absent 575 conf:(0.99)
8. leaf-malf=absent 554 ==> mycelium=absent 548 conf:(0.99)
9. mycelium=absent 639 ==> sclerotia=absent 619 conf:(0.97)
10. leaves=abnorm mycelium=absent 567 ==> sclerotia=absent 547 conf:(0.96)
Screenshot captured
You can paste the image from the clipboard.
Status
OK
Log


weka-gui-GUIChooser
Apr 5 3:48 PM
Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Clusterer
Choose HierarchicalClusterer -N 4 -L COMPLETE -P -A "weka.core.EuclideanDistance -R first-last"
Cluster mode
Use training set
Supplied test set
Percentage split
Classes to clusters evaluation
Store clusters for visualization
Ignore attributes
Start Stop
Result list (right-click for options)
15:42:04 - HierarchicalClusterer
15:42:27 - HierarchicalClusterer
15:42:38 - HierarchicalClusterer
Clusterer output
=== Run information ===
Scheme:weka.clusterers.HierarchicalClusterer -N 4 -L COMPLETE -P -A "weka.core.EuclideanDistance -R first-last"
Relation: pima_diabetes
Instances: 768
Attributes: 9
preg
plas
pres
skin
insu
mass
pedi
age
class
Test mode:evaluate on training data
=== Model and evaluation on training set ===
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Time taken to build model (full training data) : 0.32 seconds
=== Model and evaluation on training set ===
Clustered Instances
0 99 (2%)
1 485 (63%)
2 99 (2%)
3 19 (2%)
Weka Classifier Tree Visualizer: 15:42:33 - Hierarch...
Status
OK
Log

Date :

of these useful measures such as support, confidence, lift and conviction.

Conclusion:

Hence implementation of Apriori algorithm for association rule mining is studied.

Viva Questions:

(1) Define association rule mining?

→ Association rule mining finds interesting associations and relationships among large sets of data items. This rule shows how frequently a itemset occur in a transaction. A typical example is market Based Analysis. market based analysis is one of the key technique used by large relation to show associations betn items. It allows retailers to identify relationship betn the items that people buy together frequently.

(2) Define apriori algorithm?

→ The Apriori algo. is used for data mining frequent itemsets & devising association rules from a transactional database. The parameters 'support' and 'confidence' are used. Support refers to items' frequency of occurrence; confidence is conditional probability.

(3) What is meant by frequent itemset mining?

→ Frequent pattern mining is a data mining subject with the objective of extracting frequent itemsets from a db. Frequent itemsets plays an essential role in many data mining tasks and are related to interesting patterns in data, such as Association Rules.

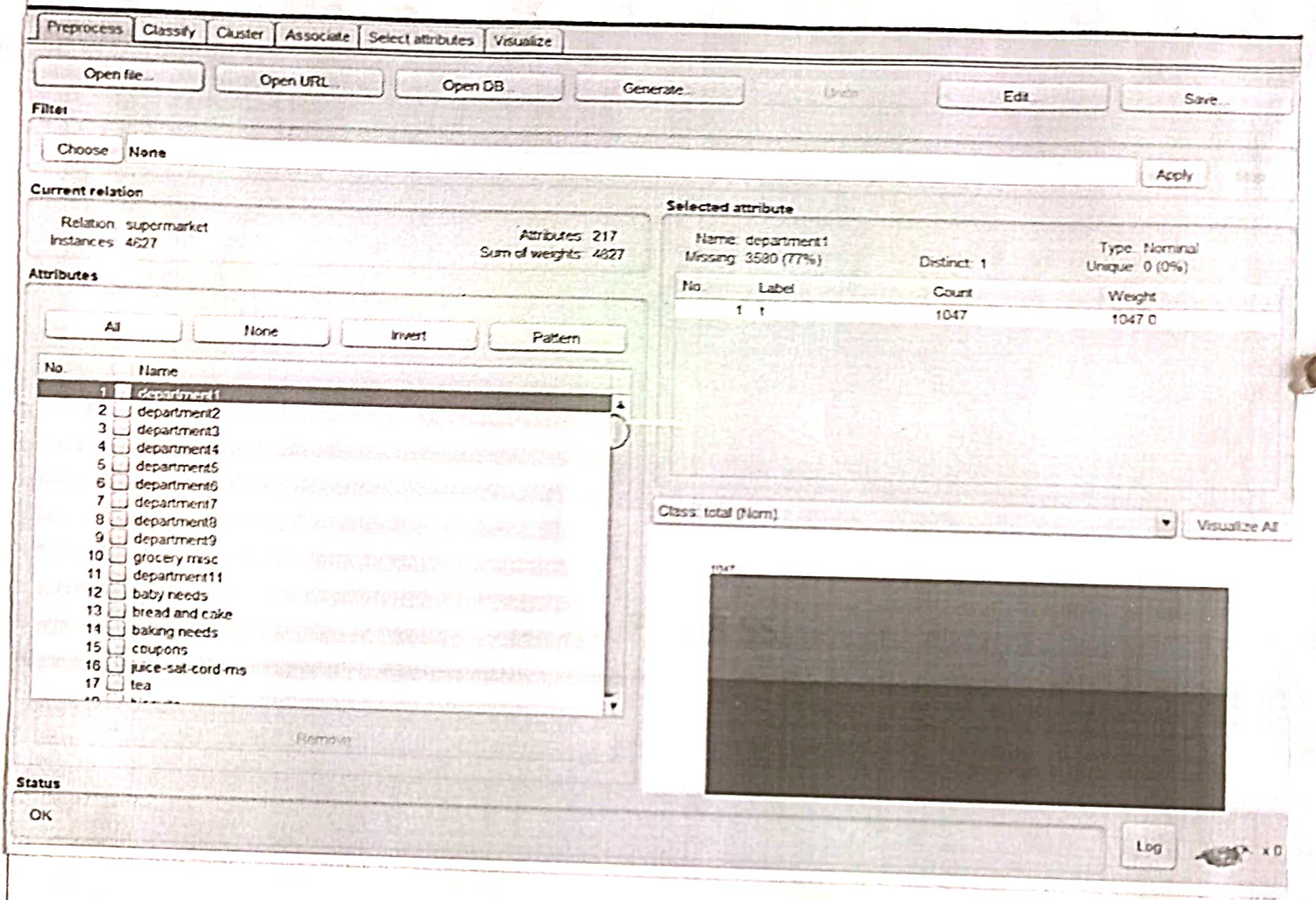
Date :

(4) Define Support and Confidence ?

→ The number of transactions that include items in the {x} and {y} parts of the rule as a percentage of the total no. of transaction. It is a measure of how frequently the collection of items occur together as a percentage of all transactions.

Practical No.8

Aim: To implement Fp Growth algorithm.



Date :

Practical No. 8

Aim: To implement Fp growth algorithm.

Theory:

Frequent pattern Growth Algorithm:

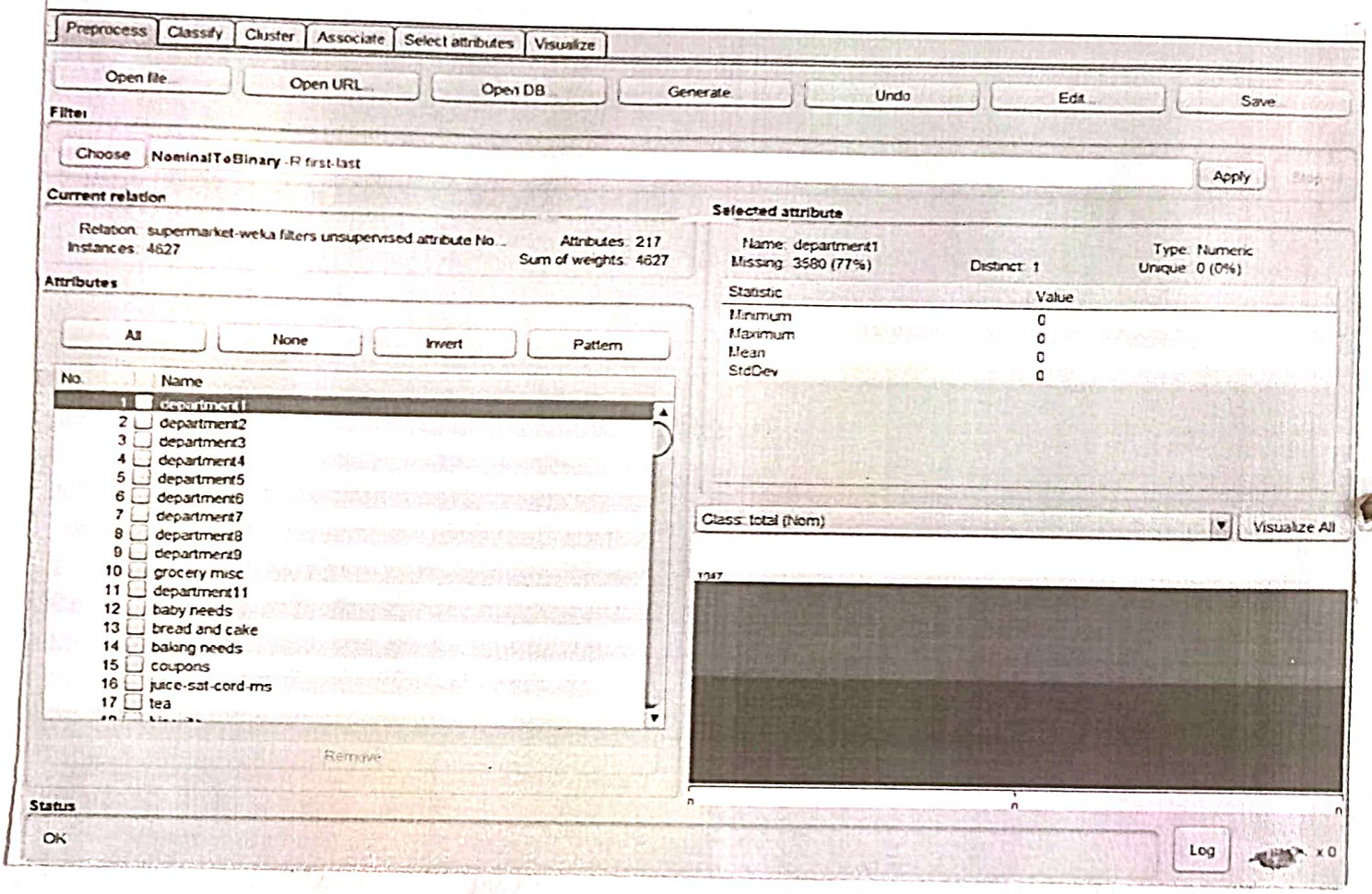
This algo. is an improvement to the Apriori method. A Fp is generated without the need for candidate generation. FP growth algo. represents the db in the form of tree called a Fp tree or Fp tree. This tree structure will maintain the association betn the itemsets. The db is fragmented using one frequent item. This fragmented part is called 'pattern fragment'. The itemset of these fragmented pattern are analyzed. Thus with this method, the search for frequent item sets is reduced comparatively.

FP tree:

FP tree is a tree like structure that is made with the initial itemset of the database. The purpose of FP tree is to mine the most frequent pattern. Each node of Fp tree represent an item of the interest itemset. The root node represents null while the lower nodes represent the itemsets. The association of the nodes with the lower nodes, that is the itemsets with the other itemsets are maintained while forming the tree.

Frequent pattern Algorithm steps:

1. The first step is to scan the db to find the occurences of the itemset in the database. This step is the same as the first step of Apriori. The count of 1-itemsets in the db is called support count or frequency of 1-itemset.
2. The second step is to construct the FP tree. For this, create the root of the tree. The root is represented by Null.


Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
NominalToBinary -R first-last
Apply
Current relation
Relation: supermarket-weka.filters.unsupervised.attribute.No...
Instances: 4627
Attributes: 217
Sum of weights: 4627
Selected attribute
Name: department1
Missing: 3580 (77%)
Distinct: 1
Type: Numeric
Unique: 0 (0%)
Statistic
Value
Minimum 0
Maximum 0
Mean 0
StdDev 0
Attributes
All
None
Invert
Pattern
No.
Name
1 department1
2 department2
3 department3
4 department4
5 department5
6 department6
7 department7
8 department8
9 department9
10 grocery misc
11 department11
12 baby needs
13 bread and cake
14 baking needs
15 coupons
16 juice-sat-cord-ms
17 tea
Remove
Class: total (Nom)
Visualize All
1247
0
0
0
Status
OK
Log
x 0

Date :

(3) The next step is to scan the database again and examine the transaction. Examine the first transaction and findout the itemset in it. The itemset with the max count is taken at top, the next itemset with lower count and so on. It means that the branch of the tree is constructed with transaction item sets in descending order of count.

(4) The next transaction in the database is examined. The item sets are ordered in descending order of count. If any itemset of this transaction is already present in another branch, then this transaction branch would share a common prefix to the root. This means that the common itemset is linked to the new node of another itemset in this transaction.

(5) Also, the count of the itemset is incremented as it occurs in the transaction. Both the common node and new node count is increased by 1 as they are created and linked acc to transaction.

(6) The next step is to mine the created FP tree. The lowest node is examined first along with the links to the lowest node. It represents the frequency pattern length 1. From this, traverse path in the FP tree. This path and paths are called a conditional pattern base.
Conditional pattern base is a sub-database consisting of prefix paths in the FP tree occuring with the lowest node

(7) construct a conditional FP tree which is formed by count of itemsets in the path.

(8) FP are generated from conditional FP tree.

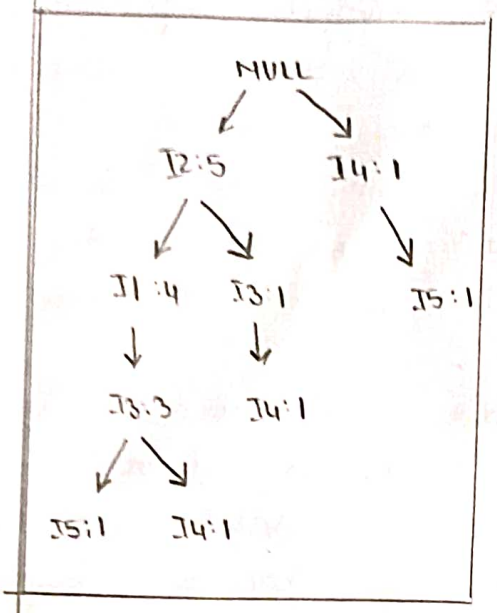


Fig: Build FP Tree

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Associator

Choose FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1

Start | Stop

Result list (right-click f...

11:56:13 - FPGrowth

Associator output

```
=== Run information ===

Scheme:       weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
Relation:     supermarket
Instances:    4627
Attributes:   217
              [list of attributes omitted]
=== Associator model (full training set) ===

FPGrowth found 16 rules (displaying top 10)

 1. [fruit=t, frozen foods=t, biscuits=t, total=high]: 788 ==> [bread and cake=t]: 723   <conf:(0.92)> lift:(1.27)
 2. [fruit=t, baking needs=t, biscuits=t, total=high]: 760 ==> [bread and cake=t]: 696   <conf:(0.92)> lift:(1.27)
 3. [fruit=t, baking needs=t, frozen foods=t, total=high]: 770 ==> [bread and cake=t]: 705   <conf:(0.92)> lift:(1
 4. [fruit=t, vegetables=t, biscuits=t, total=high]: 815 ==> [bread and cake=t]: 746   <conf:(0.92)> lift:(1.27) l
 5. [fruit=t, party snack foods=t, total=high]: 854 ==> [bread and cake=t]: 779   <conf:(0.91)> lift:(1.27) lev:(0
 6. [vegetables=t, frozen foods=t, biscuits=t, total=high]: 797 ==> [bread and cake=t]: 725   <conf:(0.91)> lift:(
 7. [vegetables=t, baking needs=t, biscuits=t, total=high]: 772 ==> [bread and cake=t]: 701   <conf:(0.91)> lift:(
 8. [fruit=t, biscuits=t, total=high]: 954 ==> [bread and cake=t]: 866   <conf:(0.91)> lift:(1.26) lev:(0.04) conv
 9. [fruit=t, vegetables=t, frozen foods=t, total=high]: 834 ==> [bread and cake=t]: 757   <conf:(0.91)> lift:(1.2
10. [fruit=t, frozen foods=t, total=high]: 969 ==> [bread and cake=t]: 877   <conf:(0.91)> lift:(1.26) lev:(0.04)
```

Status

OK

Log x0

Date :

Example :

Support threshold = 50%. Confidence = 60%.

Transaction	List of items
T1	I1, I2, I3
T2	I2, I3, I4
T3	I4, I5
T4	I1, I2, I4
T5	I1, I2, I3, I5
T6	I1, I2, I3, I4

Solution:

Support threshold = 50% $\Rightarrow$ 0.5 * 6 = 3

$\therefore$ min_sup = 3

1. Count each item :

item	count
I1	4
I2	5
I3	4
I4	4
I5	2

6

2. Sort the itemset in descending order :

item	count
I1	5
I2	4
I3	4
I4	4
I5	2

3. Build FP tree :

(1) Considering the root node null.

(3) The first scan of transaction T1 : I1, I2, I3 contain three

Date :

{I1:1}, {I2:1}, {I3:1} where I2 is linked as a child to root, I1 is linked to I2 and I3 is linked to I1.

(3) T2: I2, I3, I4 contain I2, I3 and I4 where I2 is linked to root, I3 is linked to I2 and I4 is linked to I3. But this branch would share a node I2 node as common as it is already used in T1.

(4) Increment the count of I2 by 1 and I3 is linked as a child to I2, I4 is linked as a child to I3. The count is {I2:2}, {I3:1}, {I4:1}

(5) T3: I4, I5. Similarly, a new branch with I5 is linked to I4 as a child is created.

(6) T4: I1, I2, I4. The sequence will be I2, I1, and I4. I2 is already linked to root node, hence it will be incremented by 1.

(7) T5: I1, I2, I3, I5. The sequence will be I2, I1, I3 & I5
Thus, {I2:4}, {I1:3}, {I3:2}, {I5:1}.

(8) T6: I1, I2, I3, I4. The sequence will be I2, I1, I3, I4.
Thus, {I2:5}, {I1:4}, {I3:3}, {I4:1}

4. Mining of FP-tree is summarized below:

(1) The lowest node item I5 is not considered as it does not have a min support count, hence it is deleted.

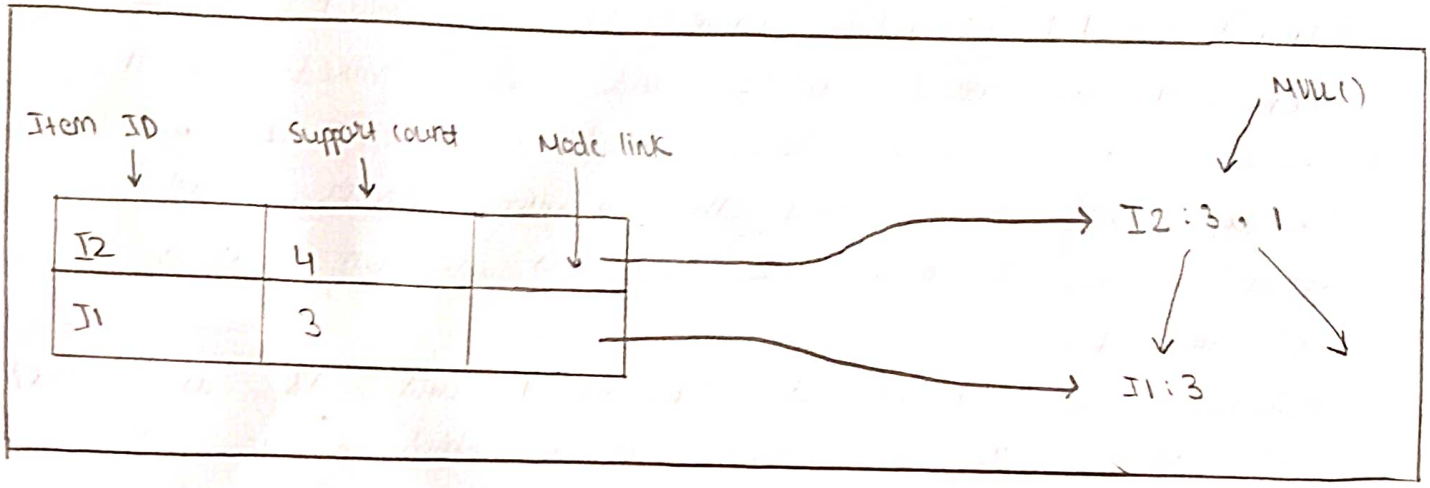
(2) The next lower node is I4. I4 occurs in 2 branches {I2, I1, I3:I4:1}, {I2, I3, I4:1}. Therefore considering I4 as suffix the prefix paths will be {I2, I1, I3:1}, {I2, I3:1}. This forms the conditional pattern base.

(3) The conditional pattern base is considered a transaction db, an FP tree is constructed. This will contain {I2:2, I3:2} I1 is not considered as it does not meet the main support count.

(4) This path will generate all combinations of FP:
{I2, I4:2}, {I3, I4:2}, {I2, I3, I4:2}

(5) For I3, the prefix path would be {I2:I1:3}, {I2:1} this will generate a 2 node FP-tree {I2:4, I1:3} and FP generated {I2, I3:4}, {I1:I3:3}, {I2, I1, I3:3}.

Page No.:


Item ID
Support count
Node link
I2
4
I1
3
NULL()
I2 : 3
I1 : 3

Date :

⑥ For I1, the prefix path would be : {I2:4} this will generate a single node FP-tree : {I2:4} and FP are generate {I2, I1:4}

Item	conditional pattern Base	conditional FP-tree	Frequent pattern generated
I4	{I2, I1, I3:1} {I2, I3:1}	{I2:2, I3:2}	{I2, I4:2}, {I3, I4:2}, {I2, I3, I4:2}
I3	{I2, I1:3}, {I2:1}	{I2:4, I1:3}	{I2, I3:4}, {I1, I3:3} {I2, I1, I3:3}
I1	{I2:4}	{I2:4}	{I2, I1:4}

Advantages:

① This algo needs to scan the db only twice when compared to Apriori which scans the transaction for each iteration

② The pairing of items is not done in this algo, and this makes it faster.

③ The db is stored in a compact version in memory.

④ It is efficient and scalable for mining both long and short frequent patterns.

Disadvantages:

① FP trees is more cumbersome and difficult to build than Apriori.

② It may be expensive.

③ When the db is large, the algo. may not fit in the shared memory.

Date :

Viva Voce :→

① Define Fp-Growth algorithm ?

→ This algo. is an improvement to the Apriori method. A FP is generated without the need for candidate Generation. Fp growth algo. represents the db in the form of tree called a Fp tree.

② How to construct Fp tree ?

→ To put it simply, an FP tree is a compressed representation of the input data. It is constructed by reading the dataset one transaction at one time and mapping each transaction onto a path in the FP-tree structure. As different transactions can have the same items, their paths may overlap.

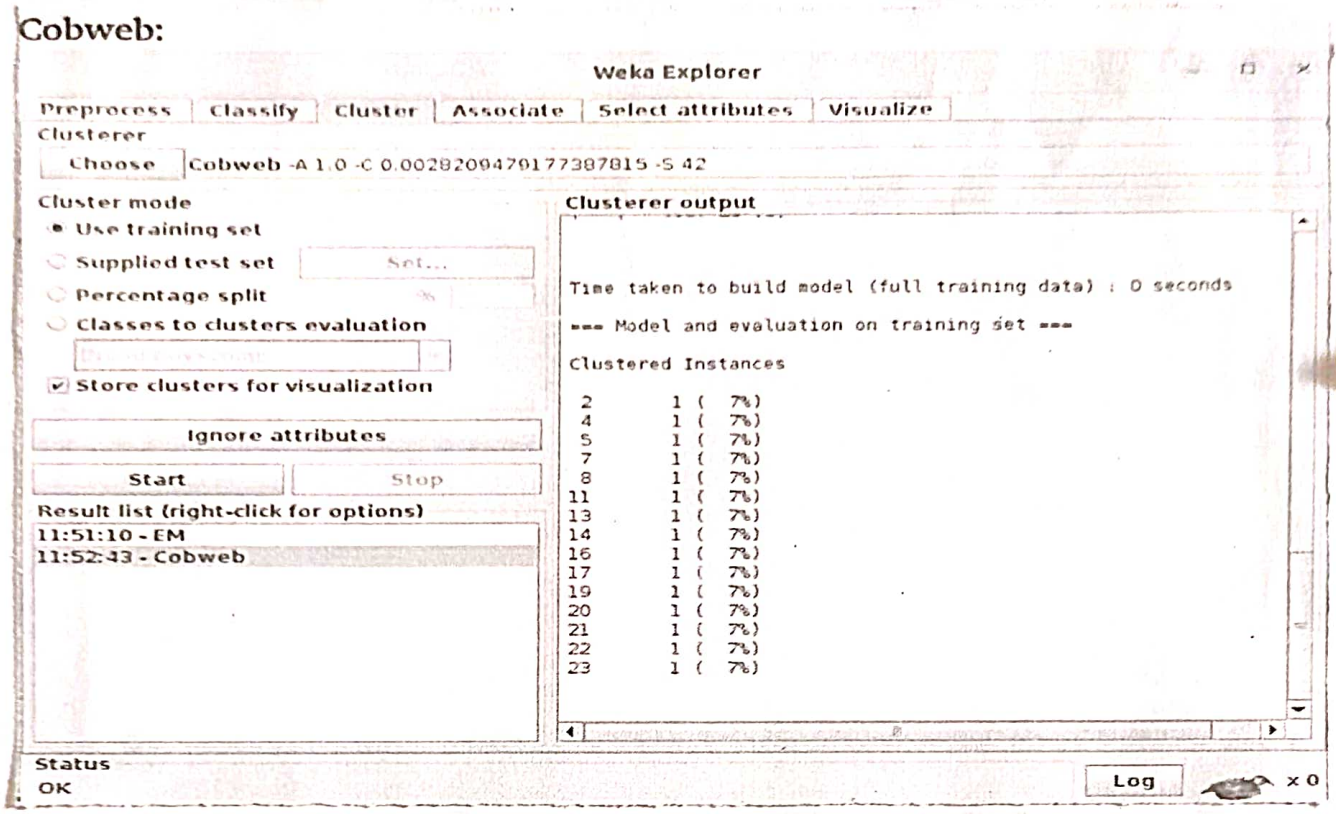
③ Define frequent pattern ?

→ Fp mining (also Association Rule mining) is an analytical process that finds frequent patterns, association or causal structures from data set found in various kinds of databases such as Relational db, transactional db and other databases repositories.

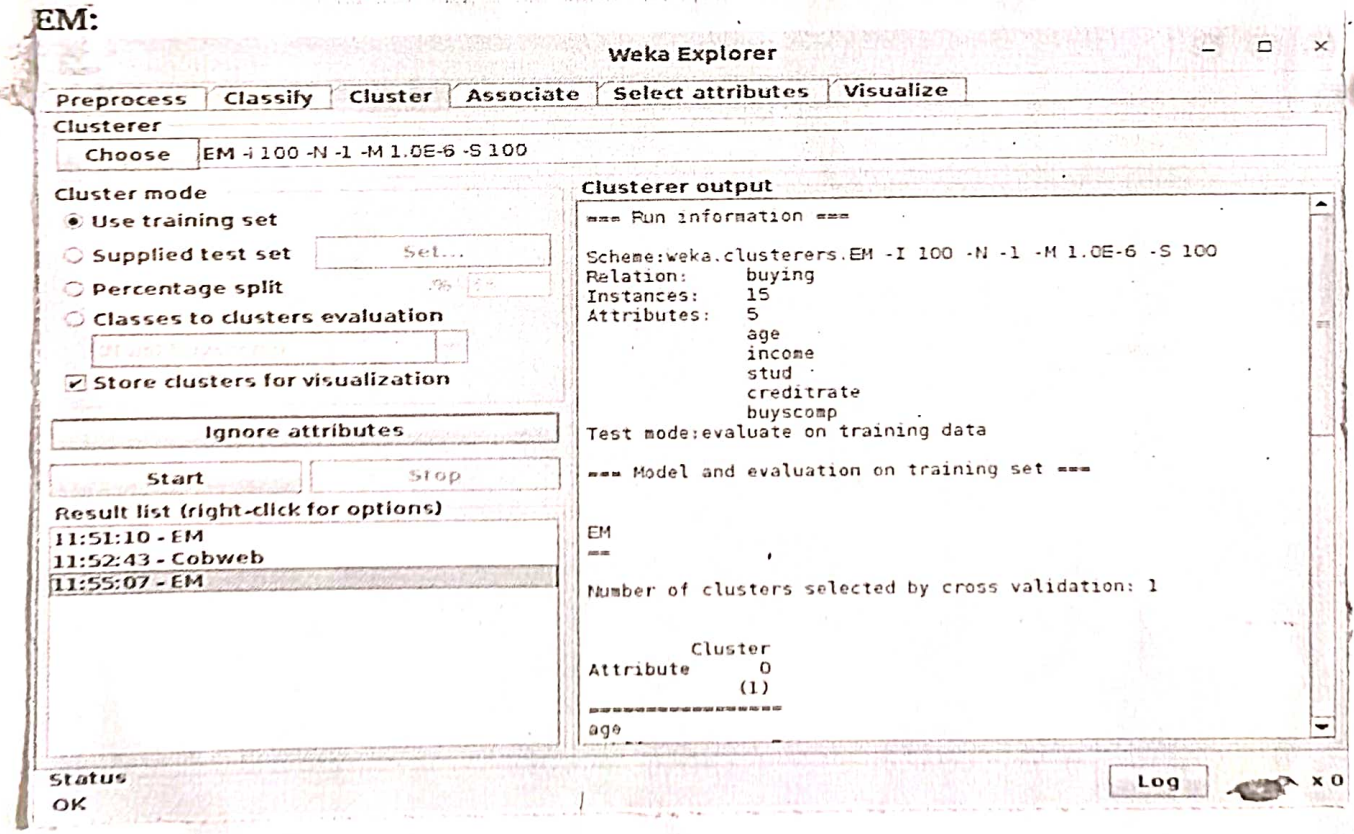
Practical No.9

Aim: Apply Cobweb, EM, Farthest first algorithm on banking data set

Cobweb:



EM:



Date :

Practical No. 9

Aim: Apply Cobweb, EM, Farthest First algorithm on Banking data set.

Theory:

Cobweb: COBWEB is an incremental system for hierarchical conceptual clustering. COBWEB was invented by professor Douglas H. Fisher.

COBWEB incrementally organizes observations into a classification tree. Each node in a classification tree represent a class and is labeled by a probalistic concept that summarizes the attribute-value distribution of object classified under the node. This classification treee can be used to predict missing attribute or the class of a new object.

There are four basic operation COBWEB employs in building the classification tree, which operation is selected depends on the category utility of the classification achieved by applying it. The operations are:

merging two node:

merging two node means replacing them by node whose children is union of the original nodes set of children and which summarizes the attribute-value distribution of all objects classified under them.

Splitting a node:

A node is split by replacing it with its children

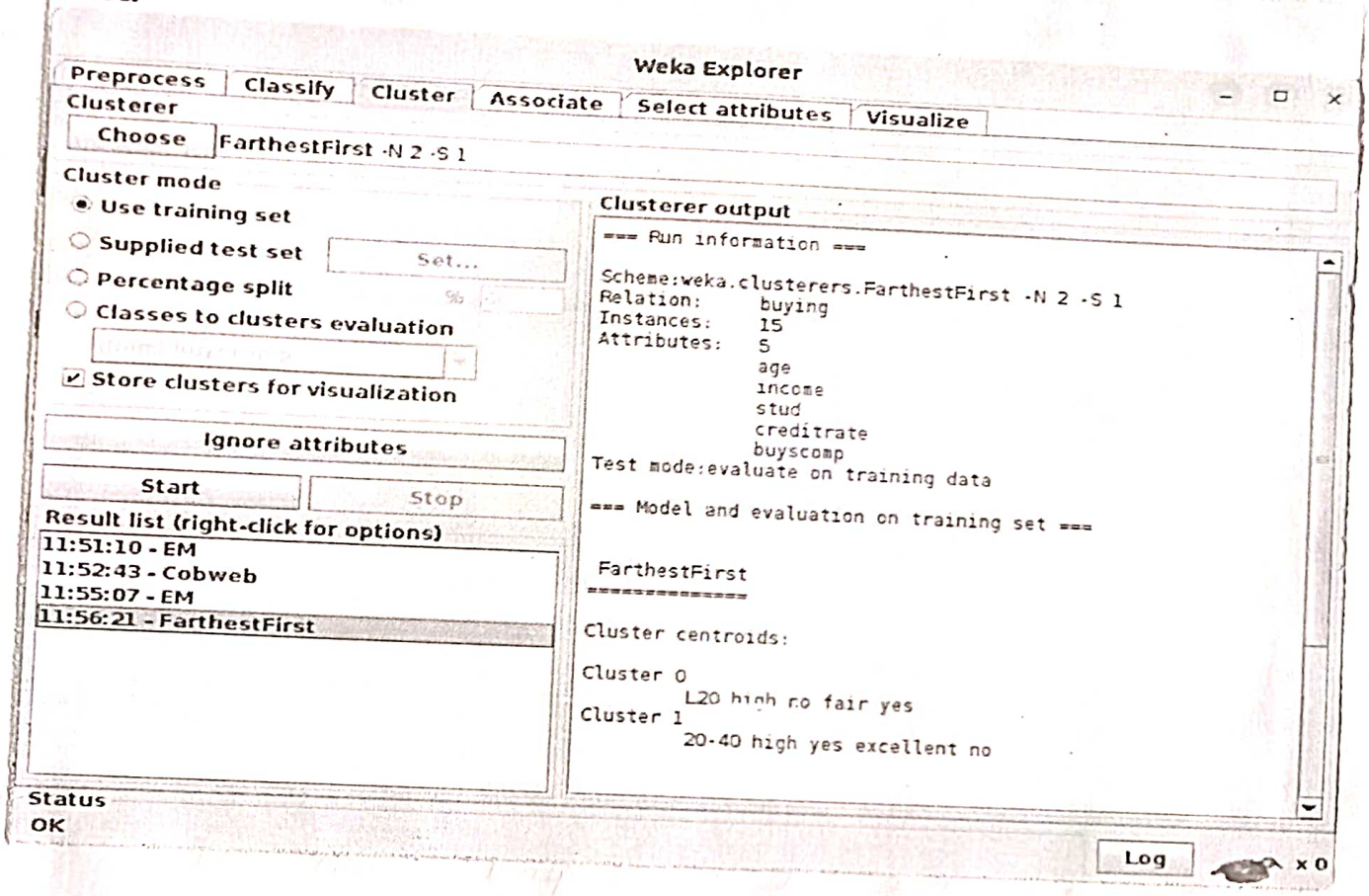
Inserting a new node:

A node is created corresponding to the object being insert into the tree.

Passing an object down the hierarchy:

Effectively calling the COBWEB algorithm on the object and subtree rooted in the node.

Farthest First:



Date :

The COBWEB algorithm:

COBWEB (root, record):

Input: A COBWEB node root, an instance to insert record

if root has no children then
  children := {copy (root)}
  new category (record) // adds child with record's feature value
  insert (record, root) // updates root's statistics.
else
  insert (record, root)
  for child in root's children do
    calculate category utility for insert (record, child),
    set best 1, best 2 children w. best CU.
  end for
  if new category (record) yields best CU then
    new category (record)
  else if merge (best 1, best 2) yields best CU then
    merge (best 1, best 2)
    COBWEB (root, record)
  else if split (best 1) yields best CU then
    split (best 1)
    COBWEB (root, record)
  else
    COBWEB (best 1, record)
  end if
end

EM algorithm :

The Expectation - maximization (EM) algorithm is defined as the combination of various unsupervised machine learning algorithm, which is used to determine the local maximum likelihood estimates (MLE) or maximum a posteriori estimate (MAP) for unobservable variables in statistical models. It is also referred to as the latent variable model.

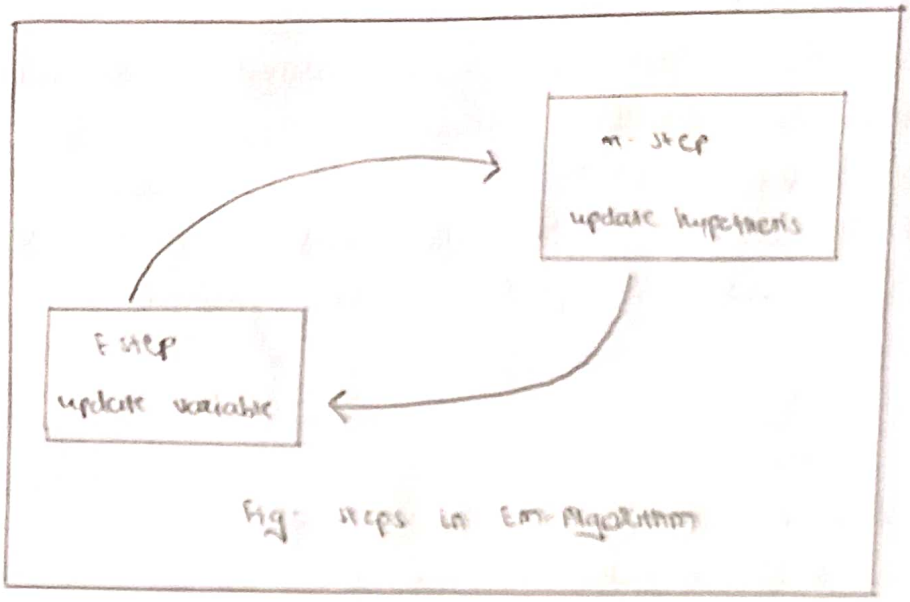


Fig: steps in Em-Algorithm

Date :

The EM algo. is the combination of various unsupervised ML algo., such as K-means clustering algo. The other model is used to optimize the parameters of the models so that it can explain the data more clearly. The second mode is known as the maximization-step or M-step.

- Expectation step (E-step): It involves the estimation of all missing values in the dataset so that after completing this step, there should not be any missing value.

- maximization step (M-step): This step involves the use of estimated data in the E-step and updating the parameter.

- Repeat E step and M-step until the convergence of the values occurs.

Farthest First Algorithm:

A farthest-first traversal is a sequence of point in a compact metric space, with each point appearing at most once. The first point of the sequence may be in the space. The first point of the or each point P after the first must have the maximum possible distance from a point to a set is defined as the minimum of the pairwise distances to set points in the set.

Farthest-point traversals may be characterized by the following properties. Fix a number K, and consider the prefix formed by the first K points of the farthest point of the farthest-first traversal of any metric space. Let r be the distance between the final point of the prefix and other point in the prefix. Then the subset has the following two properties:

Page No.

e of Practical

- All pairs of the selected points are at distance at least r from each other, and
- All points of the metric space are at distance at most r from the subset.

Working as described here, it also defines initial seeds and then on basis of "k" no. of cluster which we need to know prior. In farthest first it takes a point P_i then choose next point P_i which is at maximum distance. P_i is the centroid and $P_1, P_2 \ldots P_n$ are points or object of dataset belongs to cluster from equation.

$\min \{ \max \text{ dist}(P_i, P_1), \max \text{ dist}(P_i, P_2) \ldots \}$.

Farthest first actually solves problem of k-center and it is very efficient for large set of data. It takes centroid arbitary and distance of one centroid from other is maximum figure show cluster assignment using farthest-first.

Result:

This program has been successfully executed.

Viva Question:

(1) What is the use of COBWEB clustering?

→ It is machine learning algo. that is used to generate predictions based on data.

Teacher's Signature .....

CHANDAN
PAGE NO.:
DATE :

ame of Practical

(2) What is EM algorithm used for?

→ It is used to find local maximum likelihood parameters of a statistical model in cases where the equation cannot be solved directly.

(3) What are the steps of EM algorithm?

→ There are two steps of EM algorithm, they are

a) Expectation step (E-step):→

It is used to estimate the missing data in dataset

b) Maximization step (M-step):→

It is used to update the parameter after the complete data is generated in E-step.

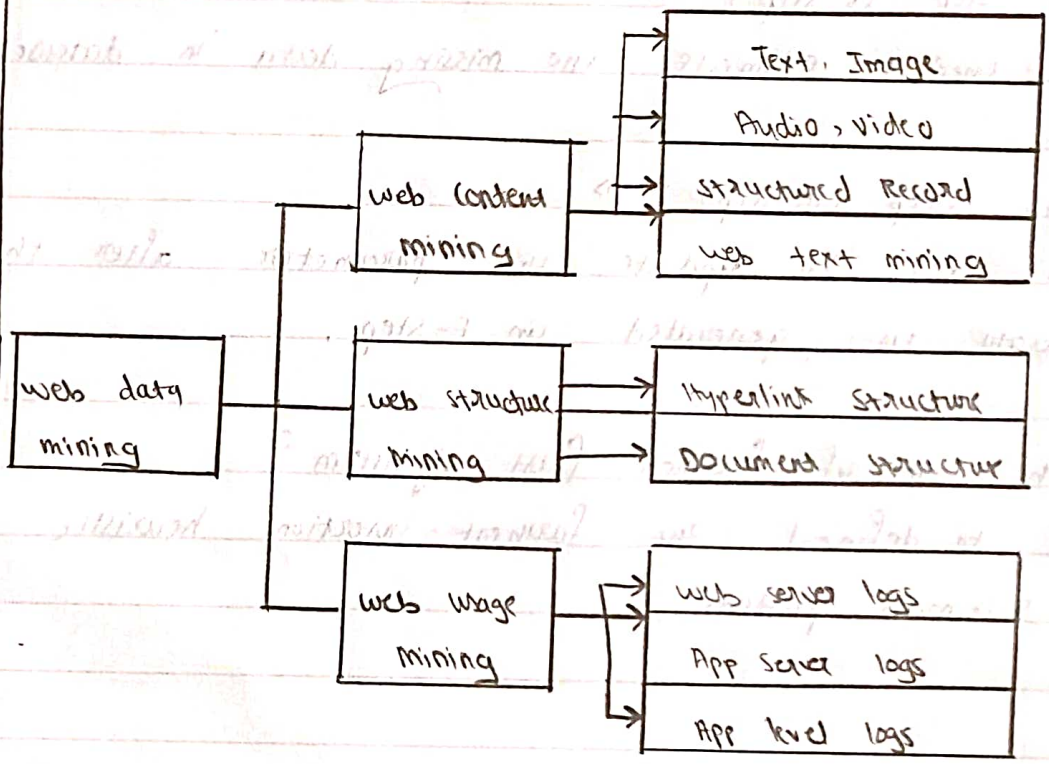
(4) What is the use of farthest first algorithm?

→ It is used to defined the farthest-insertion heuristic for travelling Salesman problem.

Teacher's Signature .....

Practical No. 10

Aim: To compare web data mining techniques, tools and algorithm.



Date :

Practical No. 10

Aim : To compare web data mining techniques, tools and algorithms.

Theory:

web mining is one of the types of techniques uses in data mining. The main purpose of web mining is to automatically extract info. from the web. Info. over the internet is huge and increasing with passage to time due to which size of data bases are also growing. It could be unstructured data, multimedia, table, tag.

web mining is actually an area of data mining related to the information available on internet. It is concept of extracting informative data available on web pages over the internet. user use different search to fetch their required data from the internet, that informative and user needed data is discovered through mining technique called web mining. Usage, content, are included in information gathered through web mining.

Web mining is sub categorized into three types as shown in figure.

a) web content mining.
b) web structure mining
c) web usage mining.

A) Web Content Mining :→

Content mining is a process of web mining in which needful informative data is extracted from web sites. Content includes audio, video, text, document & etc.. web contents are designed to deliver data to users in the form of text, list, images, video and tables. web content over a last few decades the amount of web pages (HTML) increases to billions and still continues to grow.

web mining categories	Techniques	Tools	Algorithms
web content mining	- Unstructured data mining - Structured data mining - Semi-structure data mining - multimedia data mining	- Screen scaper - mozenda - Automation Anywhere7 - web content extractor - web info extractor - Rapid miner	- Decision tree - Naives Bayes - Support Vector machines - Neural Networks
web structure mining	- hared-board classification - hared-based cluster analysis - linked Type - link strength - link cardinality	- Google PR checker - link viewer	- Page Rank algorithm - HITS algorithm - weighted page Rank algorithm - distance Rank algorithm - Eigen Runner algorithm - weighted page content Rank algorithm - web page Ranking using link attribute - Query dependent Ranking algorithm
web usage mining	- Data preprocessing: Data cleaning User and session Identification - pattern Discovery Statistical Analysis Association Rule Clustering Classification Sequential pattern	- Data preprocessing Tools: Data preparator Sumatra TT Lisp Miner Speed Tracer - pattern Discovery Tools: SEWEBAR - CMS i-Miner Argunaut Mi Das	- Association Rule: Apriori Algo. Maxi-max forward References markov chains Fp growth prefix span - clustering: self organized maps Graph partitioning Ant based Technique K-mean algo. Fuzzy c-mean algo.

CHANDAN
PAGE NO.:
DATE :

Name of Practical

B) web structure mining :→

web mining technique are very useful to discover knowledgeable data from web. structure mining is one of core techniques of web mining which deals with hyperlinks structure. structure mining basically shows the structured summary of the websites. structure mining analyze hyperlinks of the websites to collect informative data and sort out in categories like similarities and relationship. structure analysis is also called as link-mining.

c) web usage mining Techniques :→

Following three techniques are described in detail with their sub approaches use in web usage mining. Each technique performs different tasks in a hierarchy.

- Data preprocessing.
- pattern discovery.
- pattern Analysis.

Conclusion :

Thus, the web data mining techniques, tools and algorithm are studied and compared.

Viva Voce :

(1) what is web mining ? How does it differ from regular data mining or text mining ?

→ web mining is the process of using data mining technique & algo. to extract info. directly from web from documents, services, hyperlinks and servers logs.

Teacher's Signature .....

Web usage mining	- pattern analysis Knowledge query mechanism OLAP Intelligent Agents	- pattern Analysis Tools Webalizer Mavis Web Viz web miner Stratdyn	-Classification : Decision trees Naives Bayesian classifier K-nearest neighbour classification Support vector machine - sequential patterns MIDAS algorithm. (mining interest Data for association sequence)

Usage mining techniques comparison :→

Usage mining Technique	method	Data Gathering	Data store	Advantages	Important Algorithm.
Data preprocessing	Web status codes	- Data logs - website - user login info. - web access log - caches - cookies & etc.	- web logs	- convert raw data to understandable common LF and extended CLF for recording	- Apriori algo. - FP Growth.
pattern discovery	- Frequent median, mode used to show length, recently accessed, view time of pages	- filtered data from preprocessing section	- session log	- Extract useful info. from discovered pattern correlation	-K-means with genetic algo. -Fuzzy c-mean algo.
pattern Analysis	- Roll up - Drill Down /up	- pattern discovery	Data cube	Irrelevant rules and pattern are separated	- SQL language - OLAP

CHANDAN
PAGE NO.:
DATE :

Practical

Web mining is a subset of data mining that involves processing the data related to the web.

(2) What are the three main areas of web mining?

→ (1) web content mining
(2) web structure mining
(3) web usage mining.

(3) What is web content mining?

→ web content mining is the browsing and mining of text, images, graphs of web pages to decide the relevance of the content to the search query.

(4) What are three web usage mining techniques?

→ (1) Data preprocessing
(2) pattern discovery
(3) pattern analysis.

Teacher's Signature .....

Practical No. 11

Aim: To install HADOOP.

CHANDAN
PAGE NO.:
DATE :

Practical No. 11

of Practical

Aim: To install HADOOP (Advance Topic).

Theory:

What is Hadoop?

Hadoop is an open source framework from Apache and is used to store process and analyze data which are very huge in volume. Hadoop is written in Java and is not OLAP. It is used for batch / offline processing. It is being used by Facebook, yahoo, Google, Twitter, Linkedin and many more. Moreover it can be scaled up just by adding nodes in the cluster.

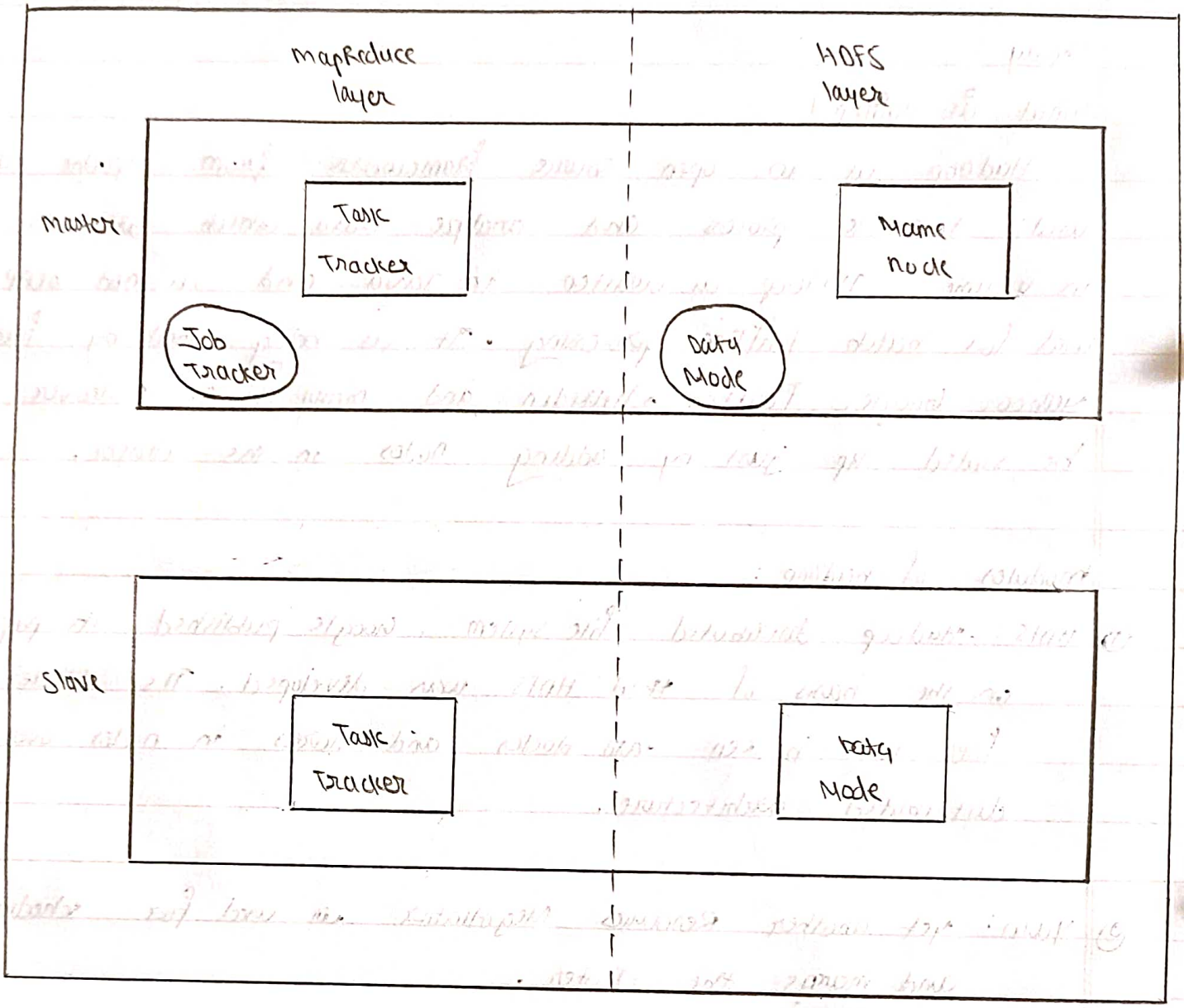
modules of hadoop:

(1) HDFS: Hadoop distributed file system. Google published its paper GFS on the basis of that HDFS was developed. It states that the files will be broken into blocks and stores in nodes over the distributed architecture.

(2) Yarn: Yet another Resource Negotiator is used for scheduling and manage the cluster.

(3) Map reduce: This is framework which help Java programs to do the parallel computation on data using key value pair. The map task takes input data and converts it into a data set which can be computed in key value pair. The output of map task is consumed by reduce task and then the out of reducer gives the desired result.

Teacher's Signature .....


mapReduce layer
HDFS layer
Master
Task Tracker
Name node
Job Tracker
Data Node
Slave
Task Tracker
Data Node

ame of Practical

(4) Hadoop Common: These java libraries are used to start Hadoop and are used by other Hadoop modules.

Hadoop Architecture:

The hadoop architecture is a package of the file system, mapreduce engine and the HDFS. The mapReduce engine can be mapReduce / MR1 or YARN / MR2.

A Hadoop cluster consist of a single master and multiple slave nodes. The master node includes Job trackers, Task trackers Namenode, datanode whereas the slave node include Datanode and tasktracker.

Advantages of Hadoop:

- Fast: In HDFS the data distributed over the cluster and are mapped which helps in faster retrieval. It also able to process tb of data in minutes and peta bytes in hours.

- Scalable: Hadoop cluster can be extended by just adding nodes in clusters.

- Cost effective: Hadoop is open source and uses commodity hardware to store data so it really cost effective as compared to traditional relational DBMS.

- Resilient to failure: HDFS has the property with which it can replicate data over the network, so if one node is down or some other network failure happens, then hadoop takes the other copy of data and use it.

Teacher's Signature .....

CHANDAN
PAGE NO.:
DATE :

Practical

Introduction to R and R studio:

R is a programming language and s/w environment for statistical computing and graphics supported by the R foundation for statistical computing. R is an integrated suite of s/w facilities for data manipulation, calculation and graphical display. It includes

The term 'environment' is intended to characterises it as a fully planned and coherent system, rather than an incremental accumulation of very specific and inflexible tools, as is frequently the case with other data analysis software. Many users think of R as a statistics system. It is preferable to this ws of it of an environment within which statistical techniques are implemented. R can be extended via packages.

The R language is widely used among statisticians and data miners for developing statistical software and data analysis.

Rstudio is an integrated development environment (IDE) for R. It include a console, syntax-highlighting editor that support direct code execution, as well as tool for plotting, history, debugging and workspace management.

Rstudio available in open source and commercial editions and runs on the desktop or in a browser connected to Rstudio server or Rstudio server pro. Rstudio is available in two editions Rstudio Desktop, where the program is run locally as a regular desktop application, and Rstudio server, which allows accessing Rstudio using a web browser while it is running on a remote linux server. Rstudio is written in the c++ programming languages and uses the Qt frameworks for its graphical user interface.

Teacher's Signature .....

```
daneshva@MoeVB:/$ R

R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

of Practical

The R hadoop methods are the collection of packages. It contains three packages ie. rmr, rhbase and rhdfs.

The rmr package:

For hadoop framework, the rmr package provides mapReduce functionality by executing the mapping and Reducing codes in R.

The rhbase packages:

This packages provides R database management capability with integration with HBASE.

The rhdfs package:

This package provides file management capabilities by integrating with HDFS.

* Installation of R, Rstudio and packages for RHadoop:

To install the latest version of R package, CRAN repository should be added to the system. use the following code for this purpose:

~ $ sudo sh -c 'echo "deb http://cran.cnr.berkeley.edu/bin/linux/ubuntu xenial/" >> /etc/apt/sources.list'

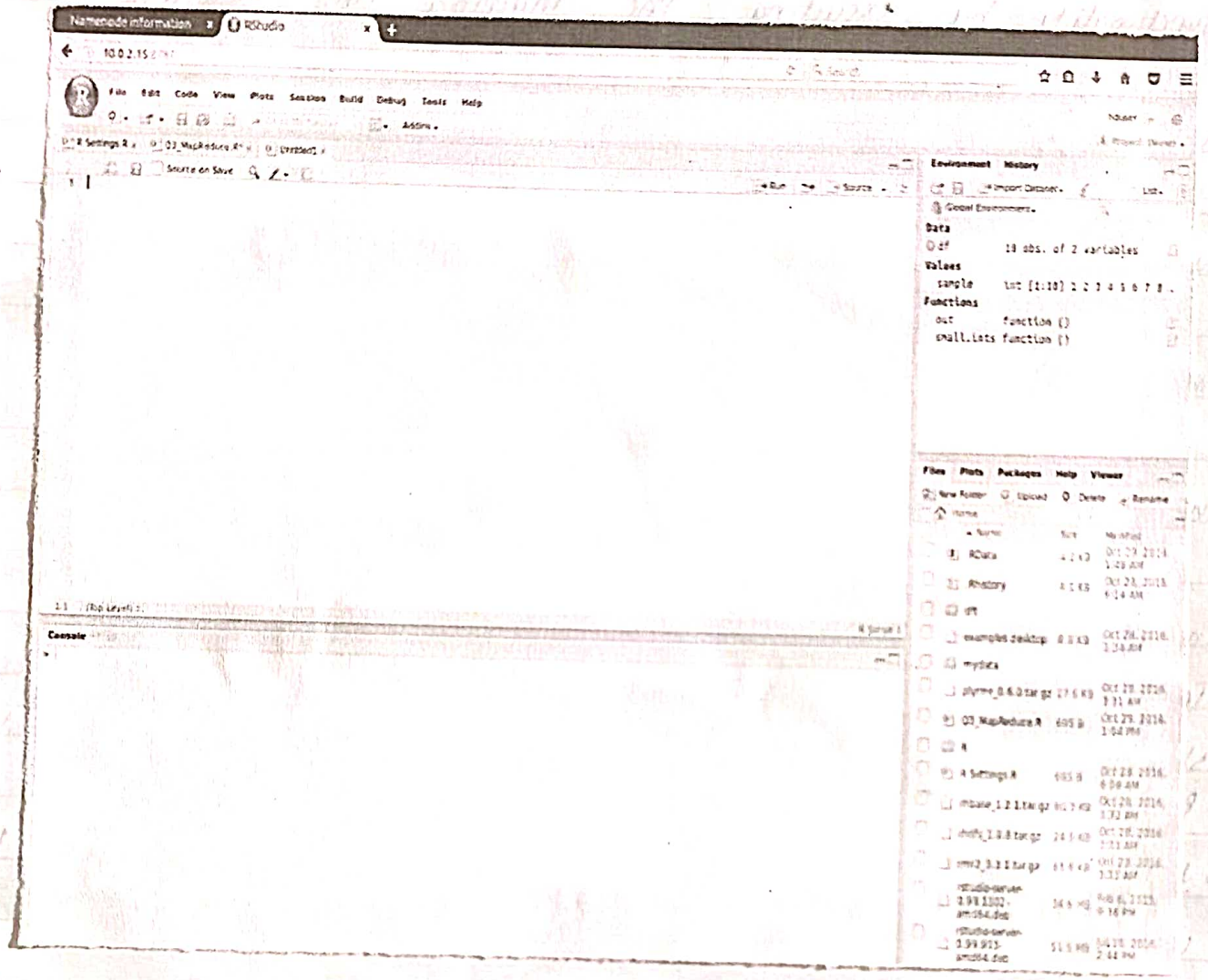
the following commands to install the complete R system and compile R package from the source.

~ $ sudo apt-get update

~ $ sudo apt-get install r-base

~ $ sudo apt-get install r-base-dev

Teacher's Signature .....


Namenode information
RStudio
10.0.2.15
File Edit Code View Plots Session Build Debug Tools Help
Addins
Source on Save
Run
Source
Console
Environment History
Import Dataset
Global Environment
Data
df 10 obs. of 2 variables
Values
sample int [1:10] 1 2 3 4 5 6 7 8 ...
Functions
out function ()
small.lists function ()
Files Plots Packages Help Viewer
New Folder
Upload
Delete
Rename
Home
Name
Size
Modified
RData
Rhistory
df
examples.desktop
mystica
plyrmr_0.6.0.tar.gz
03_MapReduce.R
R
R Settings.R
rmr2_3.3.1.tar.gz

me of Practical

At this point the installation is complete and you can run R with following command.
To quit R, type q() and select to save (or not as prompted)
~ $ R

To install R studio server, use following commands:
~$ sudo apt-get install gdebi-core
~ $ wget https://download2.rstudio.org/rstudio-server-1.0.44-amd64.deb
~$ sudo gdebi rstudio-server-1.0.44-amd64.deb

The following command is useful for identification of your VM IP address:
~ $ ifconfig

RHadoop helps in an integration interaction of R with Hadoop.
R Hadoop is collection of R packages that enables R to use Hadoop for data management. These packages are.
① plyrmr ③ rhdfs
② rmr ④ rhbase.

Installing the packages in R environment:
> install.packages(c("Rcpp", "RJSONIO", 'bitops', "digest", "functional", "stringr", "plyr", "reshape2"))

The next step is to install rJava through following commands in terminal:
~ $ sudo apt-get install oracle-java9-installer.
~ $ sudo apt-get install openjdk-9-jdk

Teacher's Signature .....

Conclusion:

Thus, Hadoop is installed successfully.

CHANDAN
PAGE NO.:
DATE :

Name of Practical

~$ sudo apt-get install libbzma-dev
~$ sudo apt-get install x-cran-xpara

To install the downloaded packages, move to the directory that the packages are downloaded in. So the following command:

~$ cd /
~$ cd home/user/Downloads
~$ HADOOP_CMD="/usr/local/hadoop/bin/hadoop"
~$ sudo R CMD INSTALL plyrmr 0.6.0.tar.gz
~$ sudo R CMD INSTALL rmr2_3.3.1.tar.gz
~$ sudo R CMD INSTALL rhdfs_1.0.8.tar.gz
~$ sudo R CMD INSTALL rhbase 1.2.1.tar.gz

Now, the RHadoop is ready to use.

Conclusion:

Thus Hadoop is installed successfully.

Viva questions:

Q Explain big data and list the characteristics?

→ Big data is a larger, complex set of data acquired from diverse, new, and old sources of data.

characteristic of big data:

a) Variety: Variety of big data refers to structured, unstructured and semistructured data that is gathered from multiple resources.

b) Velocity: Velocity essentially refers to the speed at which data is being created in realtime.

c) Volume: It indicate large volume of data that is being generated

Teacher's Signature .....

CHANDAN
PAGE NO.:
DATE :

Name of Practical

on the daily basis.

(2) Explain Hadoop. List the core components of Hadoop.

→ Hadoop is an open source framework from apache and is used to store process and analyze data which are very huge in volume.

The core components of hadoop are HDFS, YARN, Mapreduce and Hadoop common.

(3) What is Rstudio?

→ Rstudio is an integrated development environment (IDE) for R. It includes a console, syntax highlighting editor.

(4) What is the need of R-Studio?

→ It is needed for data analysis to import, access, transform, explor, plot and model data

Teacher's Signature .....