

DATA WAREHOUSING AND DATA MINING LABORATORY

(Semester -VI of B.Tech)

As per the curricullam and syllabus
of

Bharath Institute of Higher Education & Research

(DWDM Lab Manual)



Bharath

INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Declared as Deemed - to - be - University under section 3 of UGC Act 1956)

ACCREDITED WITH 'A' GRADE BY NAAC

NEW EDITION

PREPARED BY

DR. M.K.VIDHYALAKSHMI

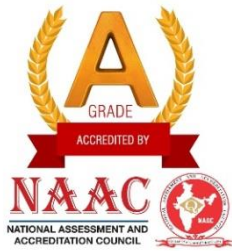


Bharath

INSTITUTE OF HIGHER EDUCATION AND RESEARCH

(Declared as Deemed-to-be University under section 3 of UGC Act, 1956)

(Vide Notification No. F.9-5/2000 - U.3, Ministry of Human Resource Development, Govt. of India, dated 4th July 2002)



ABET



SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

LAB MANUAL

SUBJECT NAME: Data Warehousing and Data Mining Laboratory

SUBJECT CODE: BCS6L1

Regulation R2015
(2015-2016)

BCS6L1	DATA WAREHOUSING AND DATA MINING LABORATORY					L	T	P	C						
	Total Contact Hours - 30					0	0	3	2						
	Prerequisite –Data ware Housing and Data mining														
	Lab Manual Designed by – Dept. of Computer Science and Engineering.														
OBJECTIVES															
Data mining is primarily used by the companies with a strong consumer focus. It enables these companies to determine the factors such as price, product positioning, or staff skills, and economic indicators, competition, and customer demographics															
COURSE OUTCOMES (COs)															
CO1	Provide efficient distribution of information and easy access to data														
CO2	Create user friendly reporting environment.														
CO3	Find the unseen pattern in large volume of historical data that helps to manage an organization efficiently.														
CO4	Understand the concepts of various data mining Techniques.														
CO5	Understand the concepts of Preprocessing.														
CO6	Explain the concept of Data mining.														
MAPPING BETWEEN COURSE OUTCOMES & PROGRAM OUTCOMES (3/2/1 INDICATES STRENGTH OF CORRELATION) 3- High, 2- Medium, 1-Low															
COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3		3							3		2	
CO2	2		3			2						3	2	2	
CO3			3	2	3				3		3	3		2	
CO4	2	2	3		3	2						3		2	
CO5					3							2		2	
CO6					2				2		3	3		2	
Category	Professional Core (PC)														
Approval	37th Meeting of Academic Council, May 2015														

LIST OF EXPERIMENTS:

1. Listing applications for mining
2. File format for data mining
3. conversion of various data files
4. Training the given dataset for an application
5. Testing the given dataset for an application
6. Generating accurate models
7. Data pre-processing – data filters
8. Feature selection
9. Web mining
10. Text mining
11. Design of fact & dimension tables
12. Generating graphs for star schema.

DATA WAREHOUSING AND DATA MINING LABORATORY- BCS6L1

LIST OF EXPERIMENTS

	NAME OF THE EXPERIMENT
1	Listing applications for mining
2	File format for data mining
3	Conversion of various data files
4	Training the given dataset for an application
5	Testing the given dataset for an application
6	Generating accurate models
7	Data pre-processing – data filters
8	Feature selection
9	Web mining
10	Text mining
11	Design of fact & dimension tables
12	Generating graphs for star schema.

CONTENT

	NAME OF THE EXPERIMENT	Page No.
1	Listing applications for mining	6
2	File format for data mining	8
3	conversion of various data files	11
4	Training the given dataset for an application	14
5	Testing the given dataset for an application	16
6	Generating accurate models	19
7	Data pre-processing – data filters	27
8	Feature selection	30
9	Web mining	39
10	Text mining	43
11	Design of fact & dimension tables	46
12	Generating graphs for star schema.	47

EX.NO:1

LISTING APPLICATIONS FOR MINING

AIM:

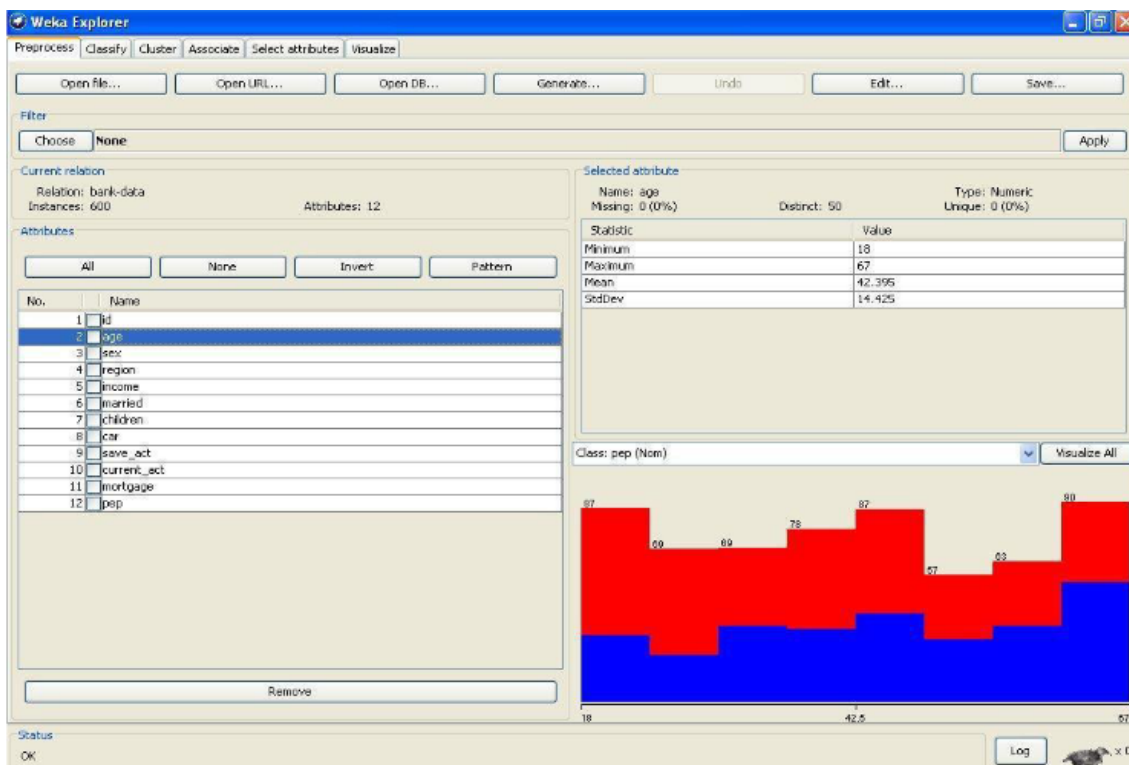
To list all the categorical (or nominal) attributes and the real-valued attributes separately.

RESOURCES: Weka mining tool1.

PROCEDURE:

- 1)Open the Weka GUI Chooser.
- 2)Select EXPLORER present in Applications.
- 3)Select Preprocess Tab.
- 4)Go to OPEN file and browse the file that is already stored in the system “bank.csv”.
- 5)Clicking on any attribute in the left panel will show the basic statistics on that selected attribute.1.4

OUTPUT:



Result:

Thus the listing applications for the data mining was studied.

EX.NO:2

FILE FORMAT FOR DATA MINING

Aim: To study the file formats for the data mining.

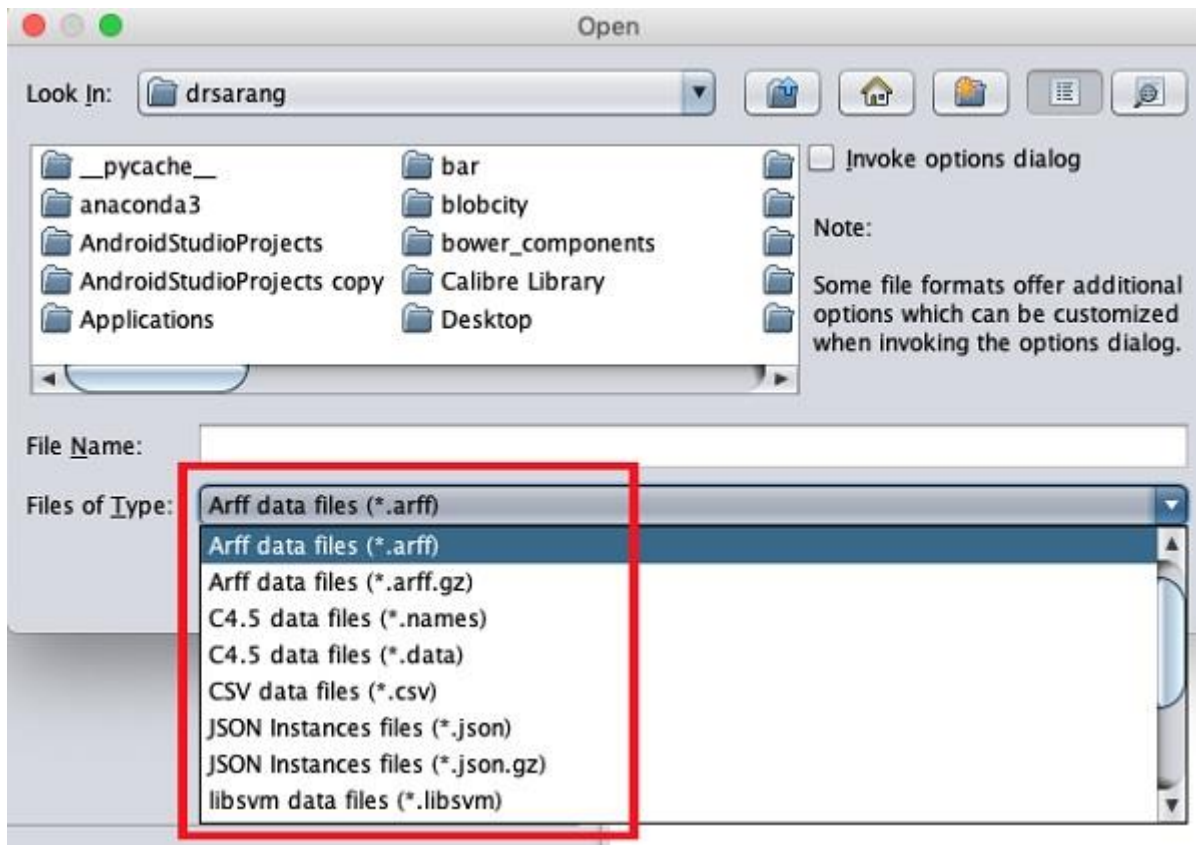
Introduction:

WEKA supports a large number of file formats for the data. The complete list of file formats are given here:

1. arff
2. arff.gz
3. bsi
4. csv
5. dat
6. data
7. json
8. json.gz
9. libsvm
10. m
11. names
12. xrff
13. xrff.gz

The types of files that it supports are listed in the drop-down list box at the bottom of the screen.

This is shown in the screenshot given below.



As you would notice it supports several formats including CSV and JSON.

The default file type is Arff.

Arff Format

An Arff file contains two sections - header and data.

The header describes the attribute types.

The data section contains a comma separated list of data.

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

```

@relation weather.symbolic
@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}
@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no

```

Diagram annotations:

- Dataset name: points to `@relation weather.symbolic`
- Attributes: points to the list of `@attribute` tags
- Target / Class variable: points to the `play` attribute tag
- Data Values: points to the list of data rows under `@data`

From the screenshot, you can infer the following points –

The `@relation` tag defines the name of the database.

The `@attribute` tag defines the attributes.

The `@data` tag starts the list of data rows each containing the comma separated fields.

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

```
@attribute outlook (sunny, overcast, rainy)
```

The attributes can take real values as in this case –

```
@attribute temperature real
```

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

```
@attribute play (yes, no)
```

The Target assumes two nominal values yes or no.

Result:

Thus the different file formats for the data mining was studied.

EX.NO:3a**CONVERSION OF TEXT FILE INTO ARFF FILE****Aim:**

To convert a text file to ARFF(Attribute-Relation File Format) using Weka3.8.2 tool.

Objectives:

Most of the data that we have collected from public forum is in the text format that cannot be read by Weka tool. Since Weka (Data Mining tool) recognizes the data in ARFF format only we have to convert the text file into ARFF file.

Algorithm:

1. Download any data set from UCI data repository.
2. Open the same data file from excel. It will ask for delimiter (which produce column) in excel.
3. Add one row at the top of the data.
4. Enter header for each column.
5. Save file as .CSV (Comma Separated Values) format.
6. Open Weka tool and open the CSV file.
7. Save it as ARFF format.

Output:**Data Text File:**

1	5.1	3.8	14	0.2
2	4.7	3.6	14	0.2
3	4.7	3.2	12	0.2
4	4.9	3.1	15	0.2
5	3	3.4	14	0.2
6	3.4	3.3	17	0.4
7	4.8	3.4	18	0.2
8	3	3.3	17	0.2
9	3.4	3.4	18	0.2
10	4.9	3.1	15	0.2
11	3.4	3.7	15	0.2
12	3.3	3.4	18	0.2
13	4.4	3	14	0.2
14	4.1	3	11	0.2
15	3.4	4	11	0.2
16	5.1	4.4	15	0.4
17	5.4	3.9	11	0.4
18	5.1	3.1	14	0.2
19	3.7	3.3	17	0.2
20	3.1	3.3	15	0.2
21	5.4	3.8	17	0.2
22	3.1	3.7	15	0.2
23	4.8	3.6	1	0.2
24	3.1	3.3	17	0.2
25	4.4	3.4	18	0.2

EX.NO:3b.

CONVERSION OF ARFF TO TEXT FILE

Aim:

To convert ARFF (Attribute-Relation File Format) into text file.

Objectives:

Since the data in the Weka tool is in ARFF file format we have to convert the ARFF file to text format for further processing.

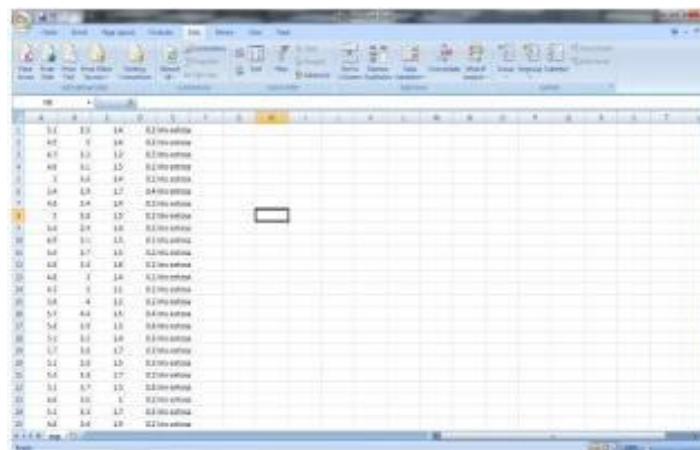
Algorithm:

1. Open any ARFF file in Weka tool.
2. Save the file as CSV format.
3. Open the CSV file in MS-EXCEL.
4. Remove some rows and add coreseponding header to the data.
5. Save it as text file with the desire delimiter.

Data ARFF File:



Data Text File:



Result: Thus conversion of ARFF (Attribute-Relation File Format) into text file is implemented.

EX. No: 4

TRAINING THE GIVEN DATASET FOR AN APPLICATION

Aim:

To apply the concept of Linear Regression for training the given dataset.

Algorithm:

1. Open the weka tool.
2. Download a dataset by using UCI.
3. Apply replace missing values.
4. Apply normalize filter.
5. Click the Classify Tab.
6. Choose the Simple Linear Regression option.
7. Select the training set of data.
8. Start the validation process.
9. Note the output.

LINEAR REGRESSION:

In statistics, Linear Regression is an approach for modeling a relationship between a scalar dependent variable Y and one or more explanatory variables denoted X. the case of explanatory variable is called Simple Linear Regression.

Coefficient of Linear Regression is given by: $Y=ax+b$

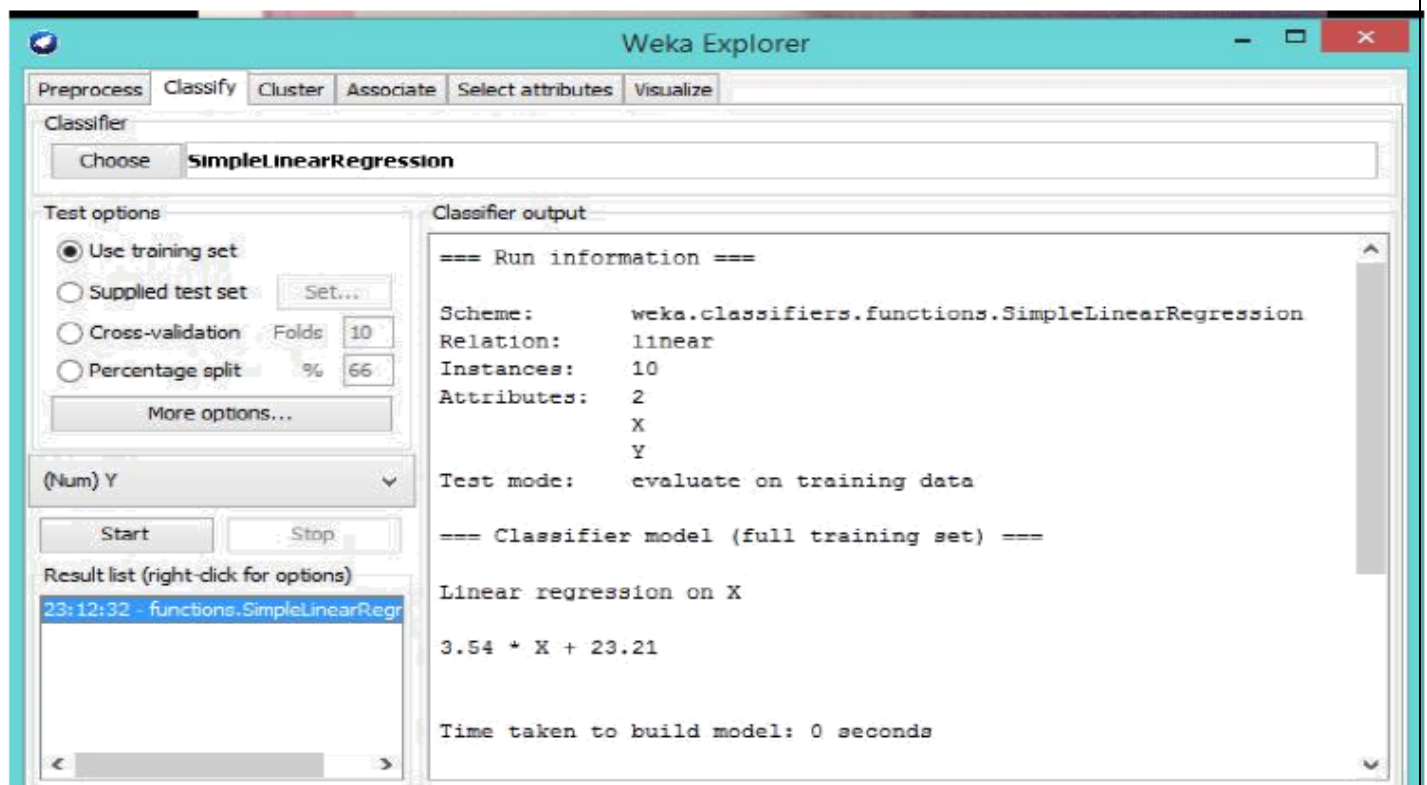
PROBLEM:

Consider the dataset below where x is the number of working experience of a college graduate and y is the corresponding salary of the graduate. Build a regression equation and predict the salary of college graduate whose experience is 10 years.

Input:

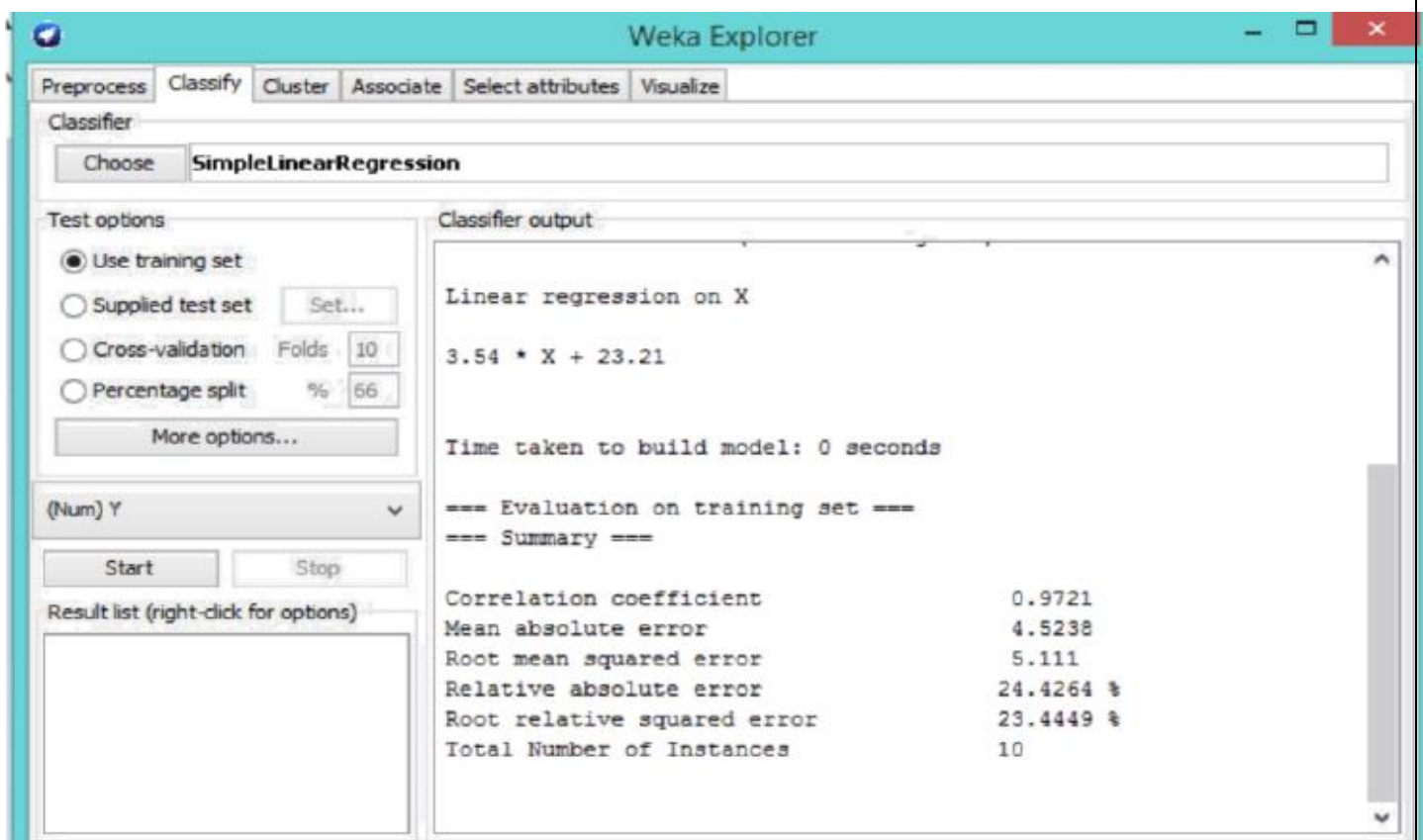
	A	B	C	D	E	F	G	H
1	X	Y						
2		3	30					
3		8	57					
4		9	64					
5		13	72					
6		3	36					
7		6	43					
8		11	59					
9		21	90					
10		1	20					
11		16	83					
12								

Output:



The screenshot shows the Weka Explorer interface with the 'Classify' tab selected. The classifier chosen is 'SimpleLinearRegression'. The 'Test options' section is set to 'Use training set'. The 'Classifier output' pane displays the following information:

```
=== Run information ===  
  
Scheme:      weka.classifiers.functions.SimpleLinearRegression  
Relation:    linear  
Instances:   10  
Attributes:  2  
             X  
             Y  
Test mode:   evaluate on training data  
  
=== Classifier model (full training set) ===  
  
Linear regression on X  
  
3.54 * X + 23.21  
  
Time taken to build model: 0 seconds
```



The screenshot shows the Weka Explorer interface with the 'Classify' tab selected. The classifier chosen is 'SimpleLinearRegression'. The 'Test options' section is set to 'Use training set'. The 'Classifier output' pane displays the following information:

```
Linear regression on X  
  
3.54 * X + 23.21  
  
Time taken to build model: 0 seconds  
  
=== Evaluation on training set ===  
=== Summary ===  
  
Correlation coefficient      0.9721  
Mean absolute error         4.5238  
Root mean squared error    5.111  
Relative absolute error    24.4264 %  
Root relative squared error 23.4449 %  
Total Number of Instances  10
```

Result: Thus the concept of Linear Regression for training the given dataset is applied and implemented.

EX. No: 5

TESTING THE GIVEN DATASET FOR AN APPLICATION

Aim:

To apply the Navie Bayes Classification for testing the given dataset.

Algorithm:

1. Open the weka tool.
2. Download a dataset by using UCI.
3. Apply replace missing values.
4. Apply normalize filter.
5. Click the Classification Tab.
6. Apply Navie Bayes Classification.
7. Find the Classified Value.
8. Note the output.

Bayes' Theorem In the Classification Context:

X is a data tuple. In Bayesian term it is considered "evidence". H is some hypothesis that X belongs to a specified class C. $P(H|X)$ is the posterior probability of H conditioned on X.

Example: predict whether a costumer will buy a computer or not " Costumers are described by two attributes: age and income " X is a 35 years-old costumer with an income of 40k " H is the hypothesis that the costumer will buy a computer " $P(H|X)$ reflects the probability that costumer X will buy a computer given that we know the costumers' age and income.

Input Data:

	A	B	C	D	E	F	G	H	I	J
1	age	income	student	credit	buys computer					
2	youth	high	no	fair	no					
3	youth	high	no	excellent	no					
4	middle	high	no	fair	yes					
5	senior	medium	no	fair	yes					
6	senior	low	yes	fair	yes					
7	senior	low	yes	excellent	no					
8	middle	low	yes	excellent	yes					
9	youth	medium	no	fair	no					
10	youth	low	yes	fair	yes					
11	senior	medium	yes	fair	yes					
12	middle	medium	no	excellent	yes					
13	middle	high	yes	fair	yes					
14	senior	medium	no	excellent	no					
15	youth	medium	yes	excellent	yes					
16										

Output data:

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 (%: 66)
 More options...

(Nom) buys computer

Start Stop

Result list (right-click for options)
 17:44:02 - misc.InputMappedClassifier

Classifier output

```

--- Run information ---

Scheme:      weka.classifiers.misc.InputMappedClassifier -I -trim -W weka.classifiers.bayes.1
Relation:    nb
Instances:   14
Attributes:  5
              age
              income
              student
              credit
              buys computer
Test mode:   user supplied test set: 1 instances

--- Classifier model (full training set) ---

InputMappedClassifier:

Naive Bayes Classifier

Attribute    Class
              no    yes
              (0.38) (0.63)
=====
age
  youth      4.0    3.0
  middle     1.0    5.0
  senior     3.0    4.0
  [total]    8.0    12.0

income
  
```

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 (%: 66)
 More options...

(Nom) buys computer

Start Stop

Result list (right-click for options)
 17:44:02 - misc.InputMappedClassifier

Classifier output

```

  [total]    8.0    12.0

income
  high       3.0    3.0
  medium     3.0    5.0
  low        2.0    4.0
  [total]    8.0    12.0

student
  no         5.0    4.0
  yes        2.0    7.0
  [total]    7.0    11.0

credit
  fair       3.0    7.0
  excellent  4.0    4.0
  [total]    7.0    11.0

Attribute mappings:

Model attributes          Incoming attributes
-----
(nominal) age             --> 1 (nominal) age
(nominal) income          --> 2 (nominal) income
(nominal) student         --> 3 (nominal) student
(nominal) credit          --> 4 (nominal) credit
  
```

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier: Choose **NaiveBayes**

Test options:

- Use training set
- Supplied test set
- Cross-validation Folds: 10
- Percentage split %: 65
-

(Nom) buys computer

Result list (right-click for options)

17:44:02 - misc.InputMappedClassifier

Classifier output

Time taken to build model: 0 seconds

--- Evaluation on test set ---

--- Summary ---

Correctly Classified Instances	1	100	%
Incorrectly Classified Instances	0	0	%
Kappa statistic	1		
Mean absolute error	0.1404		
Root mean squared error	0.1404		
Relative absolute error	37.4302 %		
Root relative squared error	37.4302 %		
Coverage of cases (0.95 level)	100	%	
Mean rel. region size (0.95 level)	100	%	
Total Number of Instances	1		

--- Detailed Accuracy By Class ---

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0	0	0	0	0	?	no
	1	0	1	1	1	?	yes
Weighted Avg.	1	0	1	1	1	0	

--- Confusion Matrix ---

a b <-- classified as

0 0 | a = no

0 1 | b = yes

Result:

Thus the Navie Bayes Classification for testing the given dataset is implemented.

EX. No: 6

GENERATE ACCURATE MODEL

Aim:

To find the good result (by improving the performance) using the training set and testing data set for numerical values.

Objectives:

To develop training and testing data using numerical data set in order to get accurate model for classification.

ALGORITHM:

1. Download any data set.
2. Save the file with .ARFF format.
3. Apply 'Replace Missing Values' filter.
4. Normalize the values by applying normalize filter.
5. Go to unsupervised instance remove percentage
6. Right click on that (show properties) option then select 70% true and save it as training.arff
7. Select the original data set then right click on show properties then give 70% false and save it as testing.arff
8. Select classification and apply various algorithms.

TRAINING DATA:

The image shows a screenshot of the Weka software interface. On the left, the 'Viewer' window displays a dataset with the following columns: ID, Last Name, First Name, City, State, Gender, Student Status, Major, Country, Age, SAT, and Average score (grade). The data is presented in a table format with 14 rows of data. On the right, the 'Scatter Plot' window is visible, showing a plot of 'Average score (grade)' versus 'SAT'. The plot shows a positive correlation between the two variables, with data points scattered around a central trend. The 'Scatter Plot' window also includes a legend and a 'Data' table with columns for 'SAT' and 'Average score'.

ID	Last Name	First Name	City	State	Gender	Student Status	Major	Country	Age	SAT	Average score (grade)	
1	DOE01	JANE01	Los An...	Califor...	Female	Graduate	Politics	US	1.571...	1.952...	67.0	
2	0.036...	DOE02	JANE02	Los An...	Arizona	Female	Undergraduate	Math	US	1.047...	1.687...	63.0
3	0.096...	DOE01	ELMER	New Y...	Male	Graduate	Math	US	1.368...	1.769...	79.0	
4	0.090...	DOE02	JANE02	Lacka...	New Y...	Female	Graduate	Econ	US	1.714...	1.385...	78.0
5	0.121...	DOE02	JANE02	Defianco	Ohio	Female	Graduate	Econ	US	1.904...	1.272...	65.0
6	0.131...	DOE04	JANE04	Tel Aviv	Israel	Male	Graduate	ECOM	Israel	1.333...	1.461...	69.0
7	0.181...	DOE05	JANE05	Dmae	Florch...	Male	Graduate	Politics	US	1.0	1.246...	95.0
8	0.113...	DOE03	JANE03	Liberal	Kentuck...	Female	Undergraduate	Politics	US	1.142...	1.618...	87.0
9	0.242...	DOE04	JANE04	Montreal	Canada	Female	Undergraduate	Math	Canada	1.0	1.489...	91.0
10	0.272...	DOE05	JANE05	New Y...	New Y...	Female	Graduate	Math	US	1.714...	1.722...	71.0
11	0.302...	DOE04	JANE04	Holt C...	Missour...	Male	Undergraduate	Econ	US	1.0	1.462...	62.0
12	0.333...	DOE06	JANE06	Jove	Virgini...	Female	Graduate	Math	US	1.952...	1.336...	79.0
13	0.363...	DOE07	JANE07	Varna	Bulgaria	Male	Graduate	Politics	Bulgaria	1.571...	1.307...	79.0
14	0.393...	DOE08	JANE08	Moscow	Russia	Male	Graduate	Politics	Russia	1.571...	1.579...	70.0
15	0.424...	DOE07	JANE07	Drunk...	New Y...	Female	Undergraduate	Math	US	1.142...	1.0	82.0
16	0.454...	DOE08	JANE08	Mexico...	Utah	Female	Undergraduate	Econ	US	1.0	1.497...	80.0
17	0.484...	DOE09	JANE09	Amsche...	Holland	Female	Undergraduate	Math	Holland	1.047...	1.366...	75.0
18	0.515...	DOE10	JANE10	Mexico	Mexico	Female	Graduate	Politics	Mexico	1.618...	1.537...	85.0
19	0.545...	DOE11	JANE11	Caracas	Venezu...	Female	Undergraduate	Math	Venezu...	1.0	1.941...	92.0
20	0.575...	DOE09	JANE09	San Juan	Puerto...	Male	Graduate	PHILOS	US	1.714...	1.682...	85.0
21	0.605...	DOE12	JANE12	Rome	Chago...	Female	Undergraduate	Econ	US	1.047...	1.406...	87.0
22	0.636...	DOE10	JANE10	New Y...	New Y...	Male	Undergraduate	Econ	US	1.142...	1.346...	82.0
23	0.666...	DOE13	JANE13	Tric C...	Moscow	Female	Graduate	PHILOS	US	1.333...	1.441...	89.0
24	0.696...	DOE14	JANE14	Beijing	China	Female	Undergraduate	Math	China	1.0	1.314...	79.0
25	0.727...	DOE11	JANE11	Stockh...	Sweden	Male	Undergraduate	Politics	Sweden	1.047...	1.566...	88.0
26	0.757...	DOE12	JANE12	Ember...	Minne...	Male	Graduate	Econ	US	1.176...	1.059...	90.0
27	0.787...	DOE13	JANE13	Inter...	Penna...	Male	Undergraduate	Math	US	1.099...	1.804...	88.0
28	0.818...	DOE15	JANE15	Loom	Ohio	Female	Undergraduate	Econ	US	1.099...	1.0	64.0
29	0.848...	DOE14	JANE14	Buenos...	Argen...	Male	Graduate	Politics	Argentina	1.571...	1.965...	83.0
30	0.878...	DOE15	JANE15	Acme	Louisian...	Male	Undergraduate	Econ	US	1.047...	1.580...	79.0
31	0.908...	DOE16	JANE16	Los An...	Califor...	Female	Graduate	Politics	US	1.571...	1.952...	67.0
32	0.938...	DOE17	JANE17	Sedona	Arizona	Female	Undergraduate	Math	US	1.047...	1.687...	63.0
33	0.969...	DOE18	JANE18	Elmer	New Y...	Male	Graduate	Math	US	1.368...	1.769...	79.0
34	1.0	DOE19	JANE19	Lacka...	New Y...	Male	Graduate	Econ	US	1.714...	1.385...	78.0

ZeroR:

Classifier
 Choose: **ConjunctiveRule - N3 - M2.0 P-1 - G1**

Test options
 Use training set
 Supplied test set
 Cross-validation: Folds: 10
 Percentage split: % 66
 More options...

Classifier output

```

instances: 34
attributes: 12
ID
Last Name
First Name
City
State
Gender
Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode: evaluate on training data

=== Classifier model (full training set) ===
ZeroR probabilities class values: 79.3238294179471

Time taken to build model: 0 seconds

=== Evaluation on training set ===
=== Summary ===

Correlation coefficient      0
Mean absolute error         0.1262
Root mean squared error     10.0203
Relative absolute error     100 %
Root relative squared error 100 %
Total Number of Instances   34
  
```

Alert
WARNING
 Advanced System Protector has detected 256 items. It is highly recommended to clean them immediately.
 Clean Now

Status: OK

Ridor:

Classifier
 Choose: **ConjunctiveRule - N3 - M2.0 P-1 - G1**

Test options
 Use training set
 Supplied test set
 Cross-validation: Folds: 10
 Percentage split: % 66
 More options...

Classifier output

```

instances: 34
attributes: 12
ID
Last Name
First Name
City
State
Gender
Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode: evaluate on training data

=== Classifier model (full training set) ===
Ripple Down Rule Learner(Ridor) rules

City = Los Angeles (34.0/0.0)

Total number of rules (incl. the default rule): 1

Time taken to build model: 0 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances      3          8.8235 %
Incorrectly Classified Instances    31         91.1765 %
Kappa statistic                     0
Mean absolute error                 0.0434
Root mean squared error             0.2502
Relative absolute error             94.7139 %
Root relative squared error        137.7271 %
Total Number of Instances          34

--- Detailed Accuracy By Class ---
  
```

Status: OK

Log

PART:

The screenshot shows the Weka Explorer interface with the 'PART' classifier selected. The 'Classifier output' pane displays the following information:

```

Classifier
-----
Chosen: ConjunctiveRule: N3-N2.0-P-1-G1

Test options
- Use training set
- Supplied testset: Set...
- Cross-validation: Folds: 10
- Percentage split: %: 66
- More options...

(Part) City
- Start
- Stop

Result list (right-click for options)
14:46:52 - rules.ZeroR
14:46:59 - rules.DecisionTable
14:54:29 - rules.OneR
14:57:03 - rules.PART
14:55:39 - rules.Rider
14:55:33 - rules.Rip
14:08:40 - rules.DTNE
14:57:03 - rules.ConjunctiveRule

Classifier output
Last Name = JOE00: Mexico (2.0/1.0)
Last Name = JOE01: Amsterdam (2.0/1.0)
Last Name = JOE10: New York (2.0/1.0)
Last Name = JOE11: Caracas (2.0/1.0)
Last Name = JOE13: The X (2.0/1.0)
Last Name = JOE14: Beijing (2.0/1.0)
Last Name = Remote (3.0/2.0)

Number of Rules: 35

Time taken to build model: 0.03 seconds

=== Evaluation on training set ===
--- Summary ---

Correctly Classified Instances      18      55.8824 %
Incorrectly Classified Instances    12      34.1176 %
Kappa Statistic                    0.5993
Mean absolute error                 0.0411
Root mean squared error            0.2047
Relative absolute error             46.8477 %
Root relative squared error        60.4923 %
Total Number of Instances          34

--- Detailed Accuracy By Class ---

```

OneR:

The screenshot shows the Weka Explorer interface with the 'OneR' classifier selected. The 'Classifier output' pane displays the following information:

```

Classifier
-----
Chosen: ConjunctiveRule: N3-N2.0-P-1-G1

Test options
- Use training set
- Supplied testset: Set...
- Cross-validation: Folds: 10
- Percentage split: %: 66
- More options...

(Part) City
- Start
- Stop

Result list (right-click for options)
14:46:52 - rules.ZeroR
14:46:59 - rules.DecisionTable
14:54:29 - rules.OneR
14:57:03 - rules.PART
14:55:39 - rules.Rider
14:55:33 - rules.Rip
14:08:40 - rules.DTNE
14:57:03 - rules.ConjunctiveRule

Classifier output
JOE00 -> Amsterdam
JOE01 -> Mexico
JOE09 -> Caracas
JOE09 -> San Juan
JOE10 -> Remote
JOE10 -> New York
JOE13 -> The X
JOE14 -> Beijing
JOE11 -> Stockholm
JOE13 -> Bahamas
JOE13 -> Intercoorrae
JOE13 -> Loco
JOE14 -> Buenos Aires
JOE15 -> Rome
(83/84 instances correct)

Time taken to build model: 0 seconds

=== Evaluation on training set ===
--- Summary ---

Correctly Classified Instances      33      97.8588 %
Incorrectly Classified Instances     1       2.9412 %
Kappa Statistic                    0.9693
Mean absolute error                 0.002
Root mean squared error            0.045
Relative absolute error             3.0553 %
Root relative squared error        24.7365 %
Total Number of Instances          34

--- Detailed Accuracy By Class ---

```

JRip:

The screenshot shows the Weka Explorer interface with the JRip classifier selected. The 'Classifier output' pane displays the following information:

```

Classifier output
Country
Age
SAT
Average score (grade)
Test mode:  evaluation on training data
=== Classifier model (full training set) ===
JRIP rules)
=====
(First Name = JER02) => City=Lackawanna (2.0/0.8)
=> City=Los Angeles (31.0/29.0)
Number of Rules : 2
Time taken to build model: 0.03 seconds
=== Evaluation on training set ===
--- Summary ---
Correctly Classified Instances      5          14.7059 %
Incorrectly Classified Instances    29          85.2941 %
Kappa statistic                     0.0669
Mean absolute error                  0.0623
Root mean squared error              0.1764
Relative absolute error              81.7591 %
Root relative squared error          98.8957 %
Total Number of Instances           34
--- Detailed Accuracy By Class ---

```

DTNB:

The screenshot shows the Weka Explorer interface with the DTNB classifier selected. The 'Classifier output' pane displays the following information:

```

Classifier output
Major
Country
Age
SAT
Average score (grade)
Test mode:  evaluation on training data
=== Classifier model (full training set) ===
Decision Table)
Number of training instances: 34
Number of Rules : 34
100 percent covered by majority class.
Evaluation (for feature selection): CV (leave one out)
Feature set: 1,3,4
Time taken to build model: 0.15 seconds
=== Evaluation on training set ===
--- Summary ---
Correctly Classified Instances      30          88.2353 %
Incorrectly Classified Instances     4          11.7647 %
Kappa statistic                     0.8771
Mean absolute error                  0.0697
Root mean squared error              0.1643
Relative absolute error              89.9233 %
Root relative squared error          90.2921 %
Total Number of Instances           34
--- Detailed Accuracy By Class ---

```

TEST DATA:

Weka Explorer

Classifier: **ConjunctiveRule** N 3 - M 2.0 P -1 - G 1

Test options:

- Use training set
- Supplied test set
- Cross-validation
- Percentage split

Classifier output:

```

first: name
City
State
Gender
Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode: evaluation on training data

=== Classifier model (full training set) ===
Zero predicted class values: Tel Aviv

Time taken to build model: 0 seconds

=== EVALUATION ON TRAINING SET ===
=== SUMMARY ===
Correctly Classified Instances      2          12.5 %
Incorrectly Classified Instances    14          87.5 %
Mappa statistic                    0           0.0
Mean absolute error                0.1167
Root mean squared error            0.3413
Relative absolute error            100 %
Root relative squared error        100 %
Total Number of Instances          16

```

==== Detailed Accuracy By Class ====

TP Rate FP Rate Precision Recall F-Measure ROC Area Class

ZeroR:

Viewer

Relation: test-test-volts.filters.unsupervised.attribute.RapsoodMeanAverage

No	ID	Last Name	First Name	City	State	Gender	Student Status	Major	Country	Age	SAT	Average score (grade)
Numeric	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Numeric	Numeric	Numeric
1	35.0	DOE20	JOE20	Tel Aviv	Ohio	Male	Graduate	Econ	US	37.0	1701.0	65.0
2	36.0	DOE21	JOE21	Tel Aviv	New Y...	Male	Graduate	Econ	Israel	25.0	1786.0	69.0
3	37.0	DOE22	JOE22	Linde	North...	Male	Graduate	Politics	US	38.0	1577.0	96.0
4	38.0	DOE23	JANE23	Liberal	Kansas	Male	Undergraduate	Politics	US	21.0	3542.0	87.0
5	39.0	DOE24	JANE24	Marshall	Canada	Female	Undergraduate	Math	Canada	18.0	913.0	81.0
6	40.0	DOE25	JANE25	New Y...	New Y...	Female	Graduate	Math	US	33.0	2091.0	71.0
7	41.0	DOE26	JOE26	Hol C...	Massa...	Male	Undergraduate	Econ	US	18.0	1787.0	82.0
8	42.0	DOE27	JANE27	Ilwaco	Virginia	Female	Graduate	Math	US	38.0	1513.0	79.0
9	43.0	DOE28	JOE28	Varna	Bulgaria	Male	Graduate	Politics	Bulgaria	39.0	2637.0	79.0
10	44.0	DOE29	JOE29	Moscow	Russia	Male	Graduate	Politics	Russia	38.0	1512.0	70.0
11	45.0	DOE30	JANE30	Drunk...	New Y...	Female	Undergraduate	Math	US	21.0	1038.0	83.0
12	46.0	DOE31	JANE31	Mic...	Utah	Female	Undergraduate	Econ	US	18.0	3521.0	80.0
13	47.0	DOE32	JANE32	Amste...	Holland	Female	Undergraduate	Math	Holland	18.0	1494.0	75.0
14	48.0	DOE33	JANE33	Mexico	Mexico	Female	Graduate	Politics	Mexico	31.0	2348.0	85.0
15	49.0	DOE34	JOE34	Elira	New Y...	Male	Graduate	Math	US	28.0	2321.0	78.0
16	50.0	DOE35	JOE35	Lacka...	New Y...	Male	Graduate	Econ	US	33.0	1718.0	79.0

Statistics

Type: Numeric
Unique: 10 (100%)

Value

- 35
- 80
- 42.5
- 4.761

Ridor:

Weka Explorer

Classifier: **ConjunctiveRule N3-M2.0-P-1-G1**

Test options:

- Use training set
- Supplied testset
- Cross-validation
- Percentage split

Classifier output:

```

===== Classifier model (full training set) =====
Ripple Open Rule Learner (Ridor) rules
-----
City = Tel Aviv (16.0/1.0)

Total number of rules (incl. the default rule): 1

Time taken to build model: 0 seconds

===== EVALUATION ON TRAINING SET =====
===== SUMMARY =====
Correctly Classified Instances      2      12.5 %
Incorrectly Classified Instances   14      87.5 %
Kappa statistic                    0
Mean absolute error                0.1094
Root mean squared error            0.3307
Relative absolute error            93.7233 %
Root relative squared error       137.055 %
Total Number of Instances         16

===== Detailed Accuracy By Class =====
TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
  
```

Status: OK

PART:

Weka Explorer

Classifier: **ConjunctiveRule N3-M2.0-P-1-G1**

Test options:

- Use training set
- Supplied testset
- Cross-validation
- Percentage split

Classifier output:

```

===== Classifier model (full training set) =====
Ripple Open Rule Learner (Ridor) rules
-----
PART = Yarnlike wool
ID <= 0.460567: Montreal (2.0/1.0)

PART = Gradiste MID
ID <= 0.510233: Linen (1.0/1.0)

PART = Linen
ID <= 0.460567: Linen (2.0/1.0)

PART = Woolen
ID <= 0.730281: Woolen (2.0/1.0)

PART = Amsterdam (2.0/1.0)

Number of Rules: 5

Time taken to build model: 0 seconds

===== EVALUATION ON TRAINING SET =====
===== SUMMARY =====
Correctly Classified Instances      0      0 %
Incorrectly Classified Instances   16     100 %
Kappa statistic                    0.4039
Mean absolute error                0.0651
Root mean squared error            0.2504
Relative absolute error           155.700 %
Root relative squared error       74.7698 %
Total Number of Instances         16

===== Detailed Accuracy By Class =====
TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
  
```

Status: OK

OneR:

The screenshot shows the Weka Explorer interface with the OneR classifier selected. The 'City' attribute is chosen for classification. The classifier output lists 18 instances, each mapped to a city. The evaluation summary shows 100% accuracy on the training set.

Classifier output:

```

DCE23 -> LISBEN
DCE24 -> Montreal
DCE25 -> New York
DCE26 -> Hot Coffe
DCE27 -> Java
DCE28 -> Varza
DCE29 -> Moscow
DCE30 -> Denmark Coast
DCE31 -> Malboro St.
DCE32 -> Amsterdam
DCE33 -> Mexico
DCE34 -> Elvira
DCE35 -> Lockness
(18/18 instances correct)

Time takes to build model: 0 seconds

=== EVALUATION ON TRAINING SET ===
=== Summary ===

Correctly Classified Instances    18      100 %
Incorrectly Classified Instances    0         0 %
Mappa statistic                    1
Mean absolute error                 0
Root mean squared error             0
Relative absolute error             0 %
Root relative squared error         0 %
Total Number of Instances         18

=== Detailed Accuracy By Class ===

```

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
---------	---------	-----------	--------	-----------	----------	-------

JRip:

The screenshot shows the Weka Explorer interface with the JRip classifier selected. The 'City' attribute is chosen for classification. The classifier output shows a single rule: 'City=Bel Aviv (16.0/14.0)'. The evaluation summary shows 10.0% accuracy on the training set.

Classifier output:

```

Country
Age
SAT
Average score (grade)
evaluate on training data

=== Classifier model (full training set) ===

JRip rules:
=====
-> City=Bel Aviv (16.0/14.0)

Number of Rules : 1

Time takes to build model: 0.01 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances    2      10.0 %
Incorrectly Classified Instances  14      80.0 %
Mappa statistic                   0
Mean absolute error               0.1162
Root mean squared error           0.2411
Relative absolute error           99.8838 %
Root relative squared error       99.0051 %
Total Number of Instances         18

=== Detailed Accuracy By Class ===

```

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
---------	---------	-----------	--------	-----------	----------	-------

DTNB:

The screenshot shows the Weka Explorer interface with the DTNB classifier selected. The 'Classify' tab is active, and the 'Classifier output' pane displays the following information:

```
Classifier output
Country
Age
SAT
Average score (grade)
Test mode: evaluate on training data

=== Classifier model (full training set) ===

Decision table:
Number of training instances: 16
Number of Rules : 15
Test machines covered by Majority class:
Evaluation (for feature selection): CV (leave one out)
Feature set: 1,4
Time taken to build model: 0.04 seconds

=== EVALUATION ON TRAINING SET ===
=== SUMMARY ===

Correctly Classified Instances      15          93.75 %
Incorrectly Classified Instances     1           6.25 %
Kappa statistic                     0.95238
Mean absolute error                  0.0977
Root mean squared error              0.208
Relative absolute error              03.7359 %
Root relative squared error          04.1356 %
Total Number of Instances           16

=== Detailed accuracy by class ===

TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
```

The 'Result list (right-click for options)' on the left shows a list of classifiers with their execution times. The DTNB classifier is highlighted in blue, indicating it is the selected model.

Result :

Thus, the good result (by improving the performance) using the training set and testing data set for numerical values is found out.

EX. No: 7**DATA PRE-PROCESSING – DATA FILTERS****Aim:**

To perform the data pre-processing by applying filter.

Objectives:

The data collected from public forums have plenty of noise or missing data. Weka provides filter to replace the missing values and to remove the noisy data. So that the result will be more accurate.

Algorithm:

1. Download a complete data set (numeric) from UCI.
2. Open the data set in Weka tool.
3. Save the data set with missing values.
4. Apply replace missing value filter.
5. Calculate the accuracy using the formula

$$\text{Accuracy} = \sqrt{\sum (\text{old} - \text{new})^2}$$


$$\text{Percentage of accuracy} = \frac{\text{Accuracy}}{\sum \text{old value}} \times 100$$

OUTPUT:**Student Details Table: Missing values**

The screenshot shows the Weka Viewer interface for a dataset named 'weather'. The table has 6 columns: 'No.', '1: outlook', '2: temperature', '3: humidity', '4: windy', and '5: play'. The data rows are as follows:

No.	1: outlook	2: temperature	3: humidity	4: windy	5: play
	Nominal	Numeric	Numeric	Nominal	Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy		96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0		TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny			FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast			TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy		91.0	TRUE	no

Student Details Table: Replace Missing values:

 Viewer

Relation: weather-weka.filters.unsupervised.attribute.Replace

No.	1: outlook	2: temperature	3: humidity	4: windy	5: play
	Nominal	Numeric	Numeric	Nominal	Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	74.8	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	83.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	74.8	83.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	74.8	83.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	74.8	91.0	TRUE	no

CALCULATION:

Data Location	Old Data	Predicted data	Errors	(Error)²
J2				
J4				
J6				

Result:

Thus the data pre-processing by applying filter is performed.

EX. No: 8

FEATURE SELECTION

AIM:

To find the good results by feature selection.

OBJECTIVES:

Any classifier/model has internal feature, those feature gives more accurate and optimal result.

ALGORITHM:

1. Download any dataset with nominal values.
2. Save it as text.arff .
3. Split it into training and testing data set.
4. Go to unsupervised instance remove percentage.
5. Right click on that show properties then select 70% true and save it as training.arff
6. Right click on that show properties then select 70% false and save it as testing.arff using original data set.
7. Open the parameter for classifying .
8. Fix the set of changing values.
9. Look at the performance.
10. Go to step 3 until the expected values of maximum value is reached.

Training Data:

The screenshot shows the Weka software interface. On the left, a 'Viewer' window displays a dataset table with columns: No, ID, Last Name, First Name, City, State, Gender, Student Status, Major, Country, Age, SAT, and Average score (grades). The table contains 14 rows of data. On the right, a dialog box is open, showing a 'Type: Numeric' field with a value of '0.302' and a 'Unique: 24 (100%)' label. The dialog box has 'Edit...', 'Save...', and 'Apply' buttons. A tooltip at the bottom of the dialog box reads 'Right click (or left-alt) for context menu'.

No	ID	Last Name	First Name	City	State	Gender	Student Status	Major	Country	Age	SAT	Average score (grades)
1	0.030...	DOE01	JANE01	Sedona	Califor...	Female	Graduate	Politic	US	1.571...	1.992...	67.0
2	0.030...	DOE02	JANE02	Sedona	New Y...	Female	Undergraduate	Math	US	1.047...	1.687...	63.0
3	0.090...	DOE01	JOE01	Elmira	New Y...	Female	Graduate	Math	US	1.386...	1.908...	78.0
4	0.090...	DOE02	JOE02	Lacka...	New Y...	Male	Graduate	Econ	US	1.714...	1.385...	78.0
5	0.221...	DOE02	JOE02	Dufrenoy	Ohio	Male	Graduate	Math	US	1.604...	1.272...	65.0
6	0.391...	DOE04	JOE04	Tel Aviv	Israel	Male	Graduate	Econ	Israel	1.533...	1.461...	69.0
7	0.391...	DOE05	JOE05	Omex	North...	Male	Graduate	Politic	US	1.0	1.248...	56.0
8	0.312...	DOE03	JANE03	Liberal	Kansas	Female	Undergraduate	Politic	US	1.142...	1.514...	67.0
9	0.242...	DOE04	JANE04	Montreal	Canada	Female	Undergraduate	Math	Canada	1.0	1.489...	71.0
10	0.272...	DOE05	JANE05	New Y...	New Y...	Female	Graduate	Math	US	1.714...	1.222...	71.0
11	0.302...	DOE04	JOE04	Massa...	Massa...	Male	Undergraduate	Econ	US	1.0	1.462...	62.0
12	0.333...	DOE06	JOE06	Jays	Virgini	Female	Graduate	Math	US	1.092...	1.388...	79.0
13	0.363...	DOE07	JOE07	Varna	Bulgaria	Male	Graduate	Politic	Bulgaria	1.571...	1.307...	79.0
14	0.393...	DOE08	JOE08	Moscow	Russia	Male	Graduate	Politic	Russia	1.571...	1.175...	70.0
15	0.424...	DOE07	JANE07	Drunk...	New Y...	Female	Undergraduate	Math	US	1.142...	1.0	62.0
16	0.454...	DOE08	JANE08	Mexic...	Utah	Female	Undergraduate	Econ	US	1.0	1.467...	60.0
17	0.484...	DOE09	JANE09	Amste...	Holland	Female	Undergraduate	Math	Holland	1.047...	1.368...	75.0
18	0.515...	DOE10	JANE10	Mexico	Mexico	Female	Graduate	Politic	Mexico	1.618...	1.537...	85.0
19	0.545...	DOE11	JANE11	Conaco	Hawa...	Female	Undergraduate	Math	Hawa...	1.0	1.044...	62.0
20	0.575...	DOE09	JOE09	San Juan	Puerto...	Male	Graduate	Politic	US	1.714...	1.602...	95.0
21	0.606...	DOE12	JANE12	Rumata	Chagat	Female	Undergraduate	Econ	US	1.047...	1.426...	67.0
22	0.636...	DOE10	JOE10	New Y...	New Y...	Male	Undergraduate	Econ	US	1.142...	1.548...	62.0
23	0.666...	DOE13	JANE13	The G...	Massa...	Female	Graduate	Politic	US	1.533...	1.441...	69.0
24	0.696...	DOE14	JANE14	Beijing	China	Female	Undergraduate	Math	China	1.0	1.314...	79.0
25	0.727...	DOE11	JOE11	Stockh...	Sweden	Male	Undergraduate	Politic	Sweden	1.047...	1.566...	68.0
26	0.757...	DOE12	JOE12	Evans...	Illino...	Male	Graduate	Econ	US	1.476...	1.028...	96.0
27	0.787...	DOE13	JOE13	Inter...	Penns...	Male	Undergraduate	Math	US	1.092...	1.304...	60.0
28	0.818...	DOE15	JANE15	Loos	Oklah...	Female	Undergraduate	Econ	US	1.096...	1.0	64.0
29	0.848...	DOE14	JOE14	Buenos...	Argon...	Male	Graduate	Politic	Argentina	1.571...	1.965...	85.0
30	0.878...	DOE15	JOE15	Arms	Louisian	Male	Undergraduate	Econ	US	1.047...	1.380...	79.0
31	0.908...	DOE16	JANE16	Los An...	Califor...	Female	Graduate	Politic	US	1.571...	1.952...	93.0
32	0.939...	DOE17	JANE02	Sedona	Arizona	Female	Undergraduate	Math	US	1.047...	1.687...	63.0
33	0.969...	DOE18	JOE01	Elmira	New Y...	Male	Graduate	Math	US	1.386...	1.908...	78.0
34	1.0	DOE19	JOE02	Lacka...	New Y...	Male	Graduate	Econ	US	1.714...	1.388...	78.0

JRip(seed=1):

Classifier
Choose: JRip-F 3-N 2.0-0.2-5.1

Test options
 Use training set
 Supplied testset: Set...
 Cross-validation: Folds: 10
 Percentage split: %: 100
 More options...

Classifier output

JRIP rules:
 =====
 (First Name = JORDI) => City=Lackawana (2.0/0.3)
 (First Name = JORDI) => City=Elmira (2.0/0.0)
 => City=Sedona (33.0/27.0)

Number of Rules : 1

Time takes to build model: 0.04 seconds

=== Evaluation on training set ===
 === Summary ===

Correctly Classified Instances	7	20.5662 %
Incorrectly Classified Instances	27	79.4338 %
kappa statistic	0.1333	
Mean absolute error	0.0551	
Root mean squared error	0.1707	
Relative absolute error	57.7056 %	
Root relative squared error	93.7599 %	
Total Number of Instances	34	

=== Detailed Accuracy By Class ===

IF Rate	FE Rate	Precision	Recall	F-Measure	ROC Area	Class
0	0	0	0	0	0.561	Los Angeles
1	0.071	0.1	1	0.162	0.563	Sedona
1	0	1	1	1	1	Elmira
1	0	1	1	1	1	Lackawana
0	0	0	0	0	0.561	Defensive
0	0	0	0	0	0.561	Tel Aviv
0	0	0	0	0	0.561	Clmax

Status: OK

JRip(seed=2):

Classifier
Choose: JRip-F 3-S 1-N 2.0

Test options
 Use training set
 Supplied testset: Set...
 Cross-validation: Folds: 10
 Percentage split: %: 100
 More options...

Classifier output

JRIP rules:
 =====
 (First Name = JORDI) => City=Lackawana (3.0/0.3)
 => City=Sedona (32.0/29.0)

Number of Rules : 2

Time takes to build model: 0.02 seconds

=== Evaluation on training set ===
 === Summary ===

Correctly Classified Instances	5	14.7059 %
Incorrectly Classified Instances	29	85.2941 %
kappa statistic	0.0663	
Mean absolute error	0.0622	
Root mean squared error	0.1764	
Relative absolute error	93.7599 %	
Root relative squared error	96.6957 %	
Total Number of Instances	34	

=== Detailed Accuracy By Class ===

Status: OK

JRip(seed=3):

The screenshot shows the Weka Explorer interface with the JRip classifier selected. The classifier output displays the following information:

```

Classifier
Chosen: JRip-F 3 N 2.0 -0.2 -0.0

Test options
Use training set: [x]
Supplied testset: [Set...]
Cross-validation: [Folds: 10]
Percentage split: [%: 66]
More options...

File: City
Start [ ] Stop [ ]
Result list [right-click for options]
00:04:52 - rules.Rip
00:05:15 - rules.Rip
00:05:24 - rules.Rip
00:05:53 - rules.Rider
11:06:03 - rules.Rider
11:01:15 - rules.Rider
11:01:31 - rules.Rip

Classifier output
Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode: evaluate on training data

--- Classifier model (full training set) ---
JRIP rules:
=====
(First Name = JOHNN) => City=LosAngeles (3.0/0.0)
=> City=Seattle (12.0/29.0)

Number of Rules : 2

Time taken to build model: 0.04 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances      5          14.7059 %
Incorrectly Classified Instances    29          85.2941 %
Kappa statistic                    0.0683
Mean absolute error                 0.0622
Root Mean Squared Error             0.1764
Relative absolute error             91.7591 %
Root relative squared error         96.8997 %
Total Number of Instances          34

=== Detailed Accuracy By Class ===

```

Ridor(seed=1):

The screenshot shows the Weka Explorer interface with the Ridor classifier selected. The classifier output displays the following information:

```

Classifier
Chosen: Ridor-F 1.5 1-N 2.0

Test options
Use training set: [x]
Supplied testset: [Set...]
Cross-validation: [Folds: 10]
Percentage split: [%: 66]
More options...

File: City
Start [ ] Stop [ ]
Result list [right-click for options]
00:04:52 - rules.Rip
00:05:15 - rules.Rip
00:05:24 - rules.Rip
00:05:53 - rules.Rider
11:06:03 - rules.Rider
11:01:15 - rules.Rider
11:01:31 - rules.Rip
11:01:38 - rules.Rider

Classifier output
Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode: evaluate on training data

--- Classifier model (full training set) ---
NIPPED DOWN NAIVE LEARNER(RIDGE) rules
-----
City = Seattle (34.0/0.0)

Total number of rules (incl. the default rule): 1

Time taken to build model: 0 seconds

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances      3          8.8235 %
Incorrectly Classified Instances    31          91.1765 %
Kappa statistic                    0
Mean absolute error                 0.0628
Root Mean Squared Error             0.2305
Relative absolute error             94.7189 %
Root relative squared error         137.7271 %
Total Number of Instances          34

=== Detailed Accuracy By Class ===

TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
0        0        0          0        0          0.5      Los Angeles

```

Ridor(seed=2):

The screenshot shows the Weka Explorer interface with the Ridor classifier selected. The 'Classifier output' pane displays the following information:

```

--- Classifier model (full training set) ---
Ripple Down Rule Learner (Ridor) rules

City - Sedona (34.0/0.6)

Total number of rules (incl. the default rule): 1

Time taken to build model: 0 seconds

--- Evaluation on training set ---
--- Summary ---

Correctly Classified Instances      3          0.0230 %
Incorrectly Classified Instances    31          91.1765 %
Kappa statistic                     0
Mean absolute error                 0.0629
Root mean squared error             0.2501
Relative absolute error             94.7139 %
Root relative squared error        137.7271 %
Total Number of Instances          34

--- Detailed Accuracy By Class ---

```

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
Los Angeles	0	0	0	0	0	0.5	Los Angeles
Sedona	1	1	0.000	1	0.142	0.5	Sedona
Kimira	0	0	0	0	0	0.5	Kimira
Lackawana	0	0	0	0	0	0.5	Lackawana
Defiance	0	0	0	0	0	0.5	Defiance

Test Data:

The screenshot shows the 'Viewer' window in Weka Explorer displaying a list of test data instances. The table below represents the data shown in the viewer:

No.	ID	Last Name	First Name	City	State	Gender	Student Status	Major	Country	Age	SAT	Average score (grade)
1	0.0	DOE20	DOE20	Tel Aviv	Ohio	Male	Graduate	Econ	US	9.904	8.598	65.0
2	0.056	DOE21	DOE24	Tel Aviv	New Y...	Male	Graduate	Econ	Israel	8.333	8.492	69.0
3	0.133	DOE22	DOE25	Limbe	North...	Female	Graduate	PHYS	US	1.0	8.261	96.0
4	0.2	DOE23	DOE27	Liberal	Kansas	Female	Graduate	Politic	US	8.140	8.551	87.0
5	0.36	DOE24	DOE24	Marina	Canada	Female	Undergraduate	Econ	Canada	8.0	8.521	81.0
6	0.533	DOE25	DOE29	New Y...	New Y...	Female	Graduate	Math	US	8.714	8.771	71.0
7	0.400	DOE26	DOE28	Hill C...	Monta...	Male	Undergraduate	Econ	US	8.0	8.493	82.0
8	0.466	DOE27	DOE26	Iowa	Virginia	Female	Graduate	Math	US	8.962	8.392	79.0
9	0.533	DOE28	DOE27	Varna	Bulgaria	Male	Graduate	PHYS	Bulgaria	8.571	8.328	79.0
10	0.6	DOE29	DOE28	Moscow	Russia	Male	Graduate	Politic	Russia	8.571	8.391	70.0
11	0.666	DOE30	DOE27	Durk...	New Y...	Female	Undergraduate	Math	US	8.140	8.0	83.0
12	0.733	DOE31	DOE28	Mexic...	Utah	Female	Undergraduate	Econ	US	8.0	8.538	80.0
13	0.8	DOE32	DOE26	Arcata...	Poland	Female	Undergraduate	Math	Poland	8.047	8.271	75.0
14	0.866	DOE33	DOE18	Mexico	Mexico	Female	Graduate	Politic	Mexico	8.619	8.0	85.0
15	0.933	DOE34	DOE21	Ethra	New Y...	Male	Graduate	Math	US	8.386	8.078	78.0
16	1.0	DOE35	DOE22	Laska...	New Y...	Male	Graduate	Econ	US	8.714	8.415	78.0

JRip(seed=1):

The screenshot shows the Weka Explorer interface with the JRip classifier selected. The classifier output window displays the following information:

```

Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode:user supplied test set: size unknown (reading incrementally)
=== Classifier model (full training set) ===
JRIP rules:
=====
=> C10j=Del ANLV (16/0/14/0)

Number of Rules : 1

Time taken to build model: 0.01 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances      1          6.25 %
Incorrectly Classified Instances    15         93.75 %
Kappa statistic                     0
Mean absolute error                 0.1171
Root mean squared error             0.2431
Relative absolute error             100 %
Root relative squared error         100.8114 %
Total Number of Instances          16

=== Detailed Accuracy By Class ===

```

At the bottom of the output window, there is a table header for performance metrics: TP Rate, FP Rate, Precision, Recall, F-Measure, ROC Area, and Class.

JRip(seed=2):

The screenshot shows the Weka Explorer interface with the JRip classifier selected. The classifier output window displays the following information:

```

Student Status
Major
Country
Age
SAT
Average score (grade)
Test mode:user supplied test set: size unknown (reading incrementally)
=== Classifier model (full training set) ===
JRIP rules:
=====
=> C10j=Del ANLV (16/0/14/0)

Number of Rules : 1

Time taken to build model: 0.02 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances      1          6.25 %
Incorrectly Classified Instances    15         93.75 %
Kappa statistic                     0
Mean absolute error                 0.1171
Root mean squared error             0.2431
Relative absolute error             100 %
Root relative squared error         100.8114 %
Total Number of Instances          16

=== Detailed Accuracy By Class ===

```

At the bottom of the output window, there is a table header for performance metrics: TP Rate, FP Rate, Precision, Recall, F-Measure, ROC Area, and Class.

JRip(seed=3):

Classifier output

```

Age
SAT
Average score (grade)
Test mode:user supplied test set: size unknown (reading incrementally)

=== Classifier model (full training set) ===

JRip rules:
=====
=> City=Tel Aviv (16.0/14.0)

Number of Rules : 1

Time taken to build model: 0.01 seconds

=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances      1          6.25 %
Incorrectly Classified Instances    15         93.75 %
Kappa statistic                    0
Mean absolute error                0.1172
Root mean squared error            0.2431
Relative absolute error             100 %
Root relative squared error        100.0114 %
Total Number of Instances         16

=== Detailed Accuracy By Class ===

    TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
-----
    0      0      0          0      0          0.5      Defiance
    1      1      0.062    1      0.118     0.5      Tel Aviv
    0      0      0          0      0          0.5      Class
  
```

Ridor(seed=1):

Classifier output

```

Average score (grade)
Test mode:user supplied test set: size unknown (reading incrementally)

=== Classifier model (full training set) ===

Ripple Down Rule Learner(Ridor) rules
-----
City = Tel Aviv (16.0/14.0)

Total number of rules (incl. the default rule): 1

Time taken to build model: 0 seconds

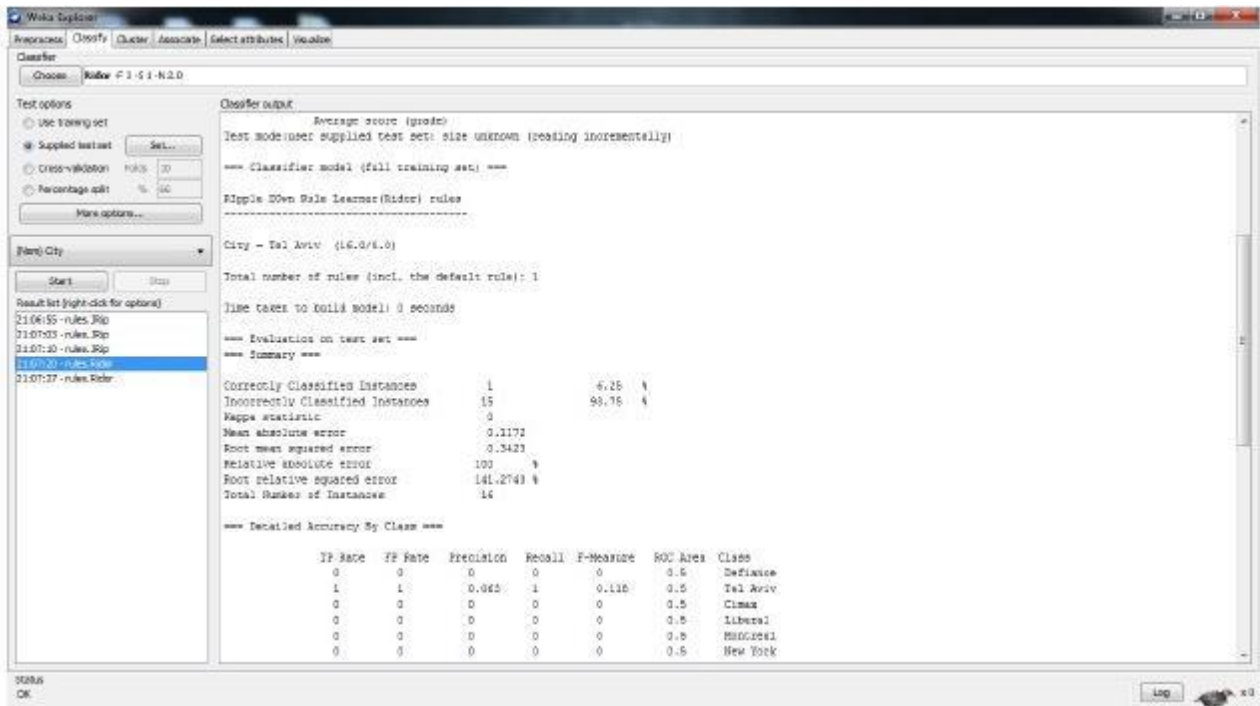
=== Evaluation on test set ===
=== Summary ===

Correctly Classified Instances      1          6.25 %
Incorrectly Classified Instances    15         93.75 %
Kappa statistic                    0
Mean absolute error                0.1172
Root mean squared error            0.2423
Relative absolute error             100 %
Root relative squared error        141.2743 %
Total Number of Instances         16

=== Detailed Accuracy By Class ===

    TP Rate  FP Rate  Precision  Recall  F-Measure  ROC Area  Class
-----
    0      0      0          0      0          0.5      Defiance
    1      1      0.062    1      0.118     0.5      Tel Aviv
    0      0      0          0      0          0.5      Class
    0      0      0          0      0          0.5      Liberal
    0      0      0          0      0          0.5      McGreal
    0      0      0          0      0          0.5      New York
  
```

Ridor(seed=2):



Training Data Set Performance:

TRAINING SET		
CLASSIFIER	PARAMETER SETTING	PERFORMANCE
JRip	Seed=1	Root Mean Squared Error=0.1707 Mean Absolute Error=0.0583
JRip	Seed =2	Root Mean Squared Error=0.1764 Mean Absolute Error=0.0622
JRip	Seed =3	Root Mean Squared Error=0.1764 Mean Absolute Error=0.0622
Ridor	Seed =1	Root Mean Squared Error=0.2508 Mean Absolute Error=0.0629
Ridor	Seed=2	Root Mean Squared Error=0.2508 Mean Absolute Error=0.0629

Testing Data set Performance:

TEST SET		
CLASSIFIER	PARAMETER SETTING	PERFORMANCE
<u>JRip</u>	Seed=1	Root Mean Squared Error=0.2431 Mean Absolute Error=0.1172
<u>JRip</u>	Seed =2	Root Mean Squared Error=0.2431 Mean Absolute Error=0.1172
<u>JRip</u>	Seed =3	Root Mean Squared Error=0.2431 Mean Absolute Error=0.1172
<u>Ridor</u>	Seed =1	Root Mean Squared Error=0.3423 Mean Absolute Error=0.1172
<u>Ridor</u>	Seed=2	Root Mean Squared Error=0.3423 Mean Absolute Error=0.1172

Comparison between training and testing data set:

TRAINING		
<u>JRip</u>	Seed=1	Root Mean Squared Error=0.1707 Mean Absolute Error=0.0583
<u>Ridor</u>	Seed =1	Root Mean Squared Error=0.2508 Mean Absolute Error=0.0629

TEST		
<u>JRip</u>	Seed=1	Root Mean Squared Error=0.2431 Mean Absolute Error=0.1172
Rider	Seed =1	Root Mean Squared Error=0.3423 Mean Absolute Error=0.1172

Result:

Thus the good results by feature selection were found.

EX. No: 9

Web Mining

Aim:

To apply the web mining technique clustering algorithm for the given dataset.

Introduction to Web Mining:

Web mining is an application of data mining techniques to find information patterns from the web data. Web mining helps to improve the power of web search engine by identifying the web pages and classifying the web documents. Web mining is very useful to e-commerce websites and e-services.

Web Content Mining :

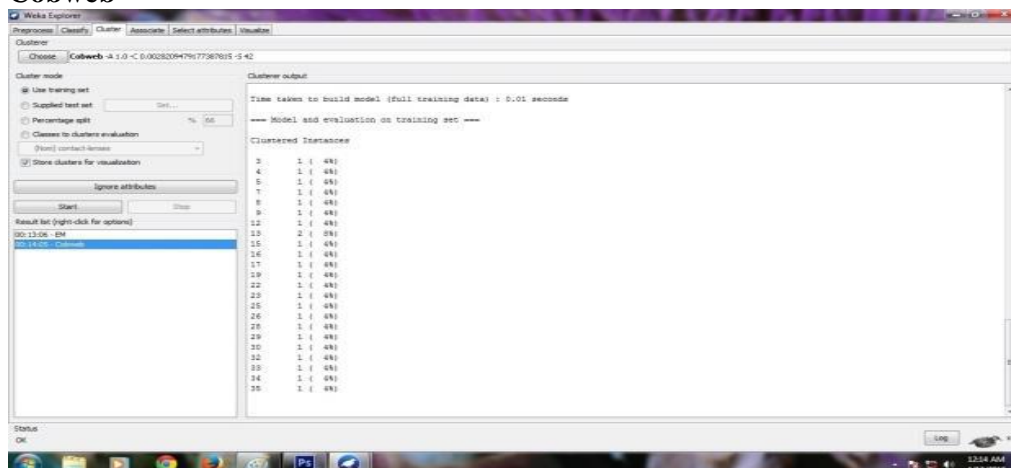
Web content mining can be used for mining of useful data, information and knowledge from web page content. Web structure mining helps to find useful knowledge or information pattern from the structure of hyperlinks. Due to heterogeneity and absence of structure in web data, automated discovery of new knowledge pattern can be challenging to some extent. Web content mining performs scanning and mining of the text, images and groups of web pages according to the content of the input (query), by displaying the list in search engines. For example: If an user wants to search for a particular book, then search engine provides the list of suggestions.

ALGORITHM:

1. Open the weka tool.
2. Download a dataset by using UCI.
3. Apply replace missing values.
4. Apply normalize filter.
5. Click the cluster tab.
6. Apply all algorithms one by one.
7. Find the no of clusters that are formed
8. Note the output.

Output:

Cobweb



EM

Weka Explorer interface showing EM clustering results. The 'Cluster' tab is active, displaying the 'EM-1 100-N-1' model. The 'Cluster mode' section shows 'Use training set' selected. The 'Result list' shows three entries: 'EM-1 100-N-1', 'EM-1 100-N-1', and 'EM-1 100-N-1'. The 'Cluster output' section displays the following information:

```

=== Run information ===
Scheme: weka.clusterers.EM -I 100 -N 1 -K 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -num-slots 1 -S 100
Relation: contact-lenses
Instances: 24
Attributes: 5
  age
  spectacle-prescrip
  astigmatism
  tear-prod-rate
  contact-lenses
Test mode: evaluate on training data

=== Clustering model (full training set) ===

EM
==

Number of clusters selected by cross validation: 2
Number of iterations performed: 10

Attribute      Cluster
              0      1
-----
age
young          5.1102  4.6898
pre-presbyopic 5.0346  4.1654
presbyopic     4.4083  3.3917
    
```

Farthest First

Weka Explorer interface showing Farthest First clustering results. The 'Cluster' tab is active, displaying the 'FarthestFirst -N 2 -S 1' model. The 'Cluster mode' section shows 'Use training set' selected. The 'Result list' shows three entries: 'EM-1 100-N-1', 'EM-1 100-N-1', and 'FarthestFirst'. The 'Cluster output' section displays the following information:

```

=== Run information ===
Scheme: weka.clusterers.FarthestFirst -N 2 -S 1
Relation: contact-lenses
Instances: 24
Attributes: 5
  age
  spectacle-prescrip
  astigmatism
  tear-prod-rate
  contact-lenses
Test mode: evaluate on training data

=== Clustering model (full training set) ===

FarthestFirst
=====

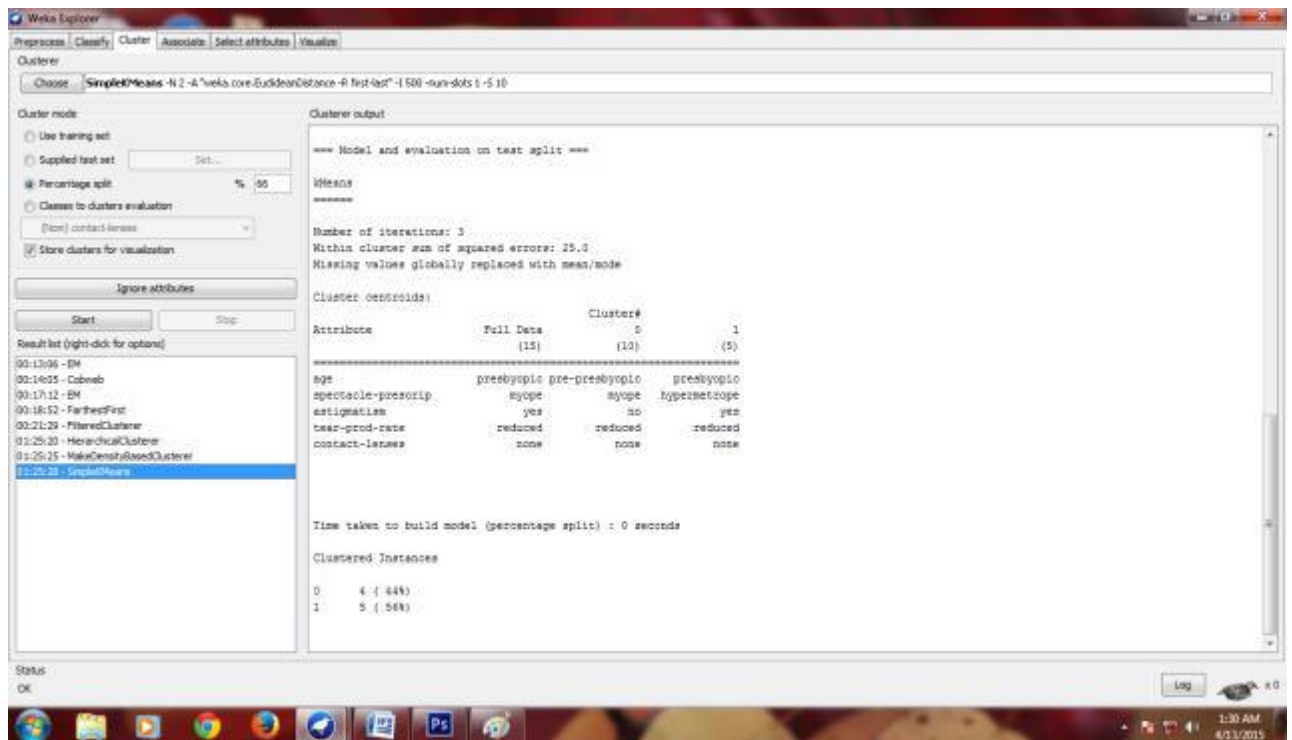
Cluster centroids:

Cluster 0
pre-presbyopic myope no normal soft
Cluster 1
young hypermetrope yes reduced none

Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===
    
```


Simple KMeans:



The screenshot shows the Weka Explorer interface with the Simple KMeans algorithm selected. The 'Clusterer' dropdown is set to 'SimpleKMeans -N 2 -A "weka core.EuclideanDistance-R test-test"-I 500 -num-slots 1 -S 10'. The 'Cluster mode' section has 'Percentage split' checked at 50%. The 'Clusterer output' window displays the following information:

```
== Model and evaluation on test split ==
KMeans
=====
Number of iterations: 3
Within cluster sum of squared errors: 25.3
Missing values globally replaced with mean/mode

Cluster centroids:
Attribute          Full Data          Cluster#
                   (15)              (10)              (5)
-----
age                presbyopic pre-presbyopic  presbyopic
spectacle-prescrip myope          myope          hypermetrope
astigmatism        yes            no             yes
tear-prod-rate     reduced       reduced       reduced
contact-lenses     none          none          none
```

Time taken to build model (percentage split) : 0 seconds

Clustered Instances

Cluster#	Count	Percentage
0	4	(44%)
1	5	(56%)

The bottom status bar shows 'OK' and a 'Log' button. The system tray at the bottom right indicates the time is 1:30 AM on 4/13/2015.

Result:

Thus the web mining technique clustering algorithm for the given dataset is implemented.

Aim:

To find association between data and to find the frequent item set for text mining.

Text Data Mining

Text data mining can be described as the process of extracting essential data from standard language text. All the data that we generate via text messages, documents, emails, files are written in common language text. Text mining is primarily used to draw useful insights or patterns from such data. The purchasing of one product when another product is purchased represents an association rule. Association rules are frequently used by retail store to assist in marketing, advertising, floor placement, and inventory control. Association rules are used to show the relationship between data items.

Keyword-based Association Analysis in text mining:

It collects sets of keywords or terms that often happen together and afterward discover the association relationship among them. First, it preprocesses the text data by parsing, stemming, removing stop words, etc. Once it pre-processed the data, then it induces association mining algorithms. Here, human effort is not required, so the number of unwanted results and the execution time is reduced.

ALGORITHM:

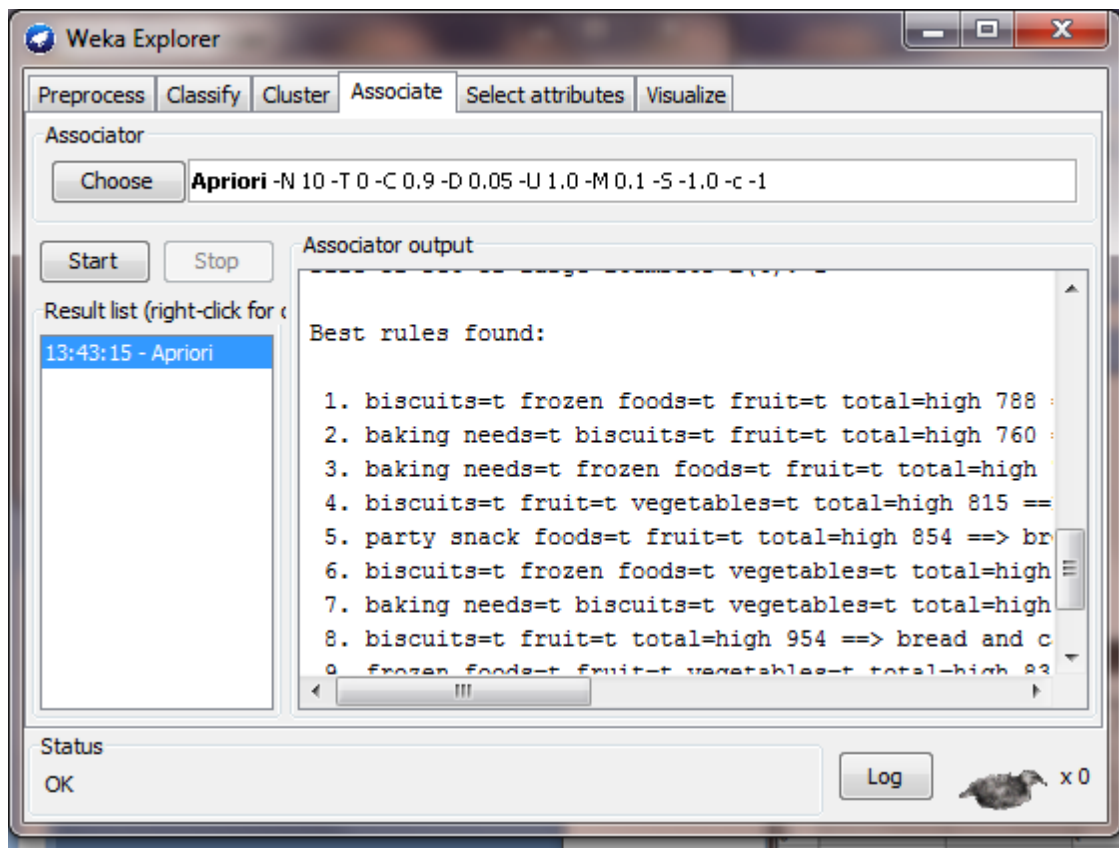
1. Open dataset
2. Select associate
3. Choose different algorithm for association
4. Observe the performance
5. Select the association rule with the maximum confidence rule.

INPUT:

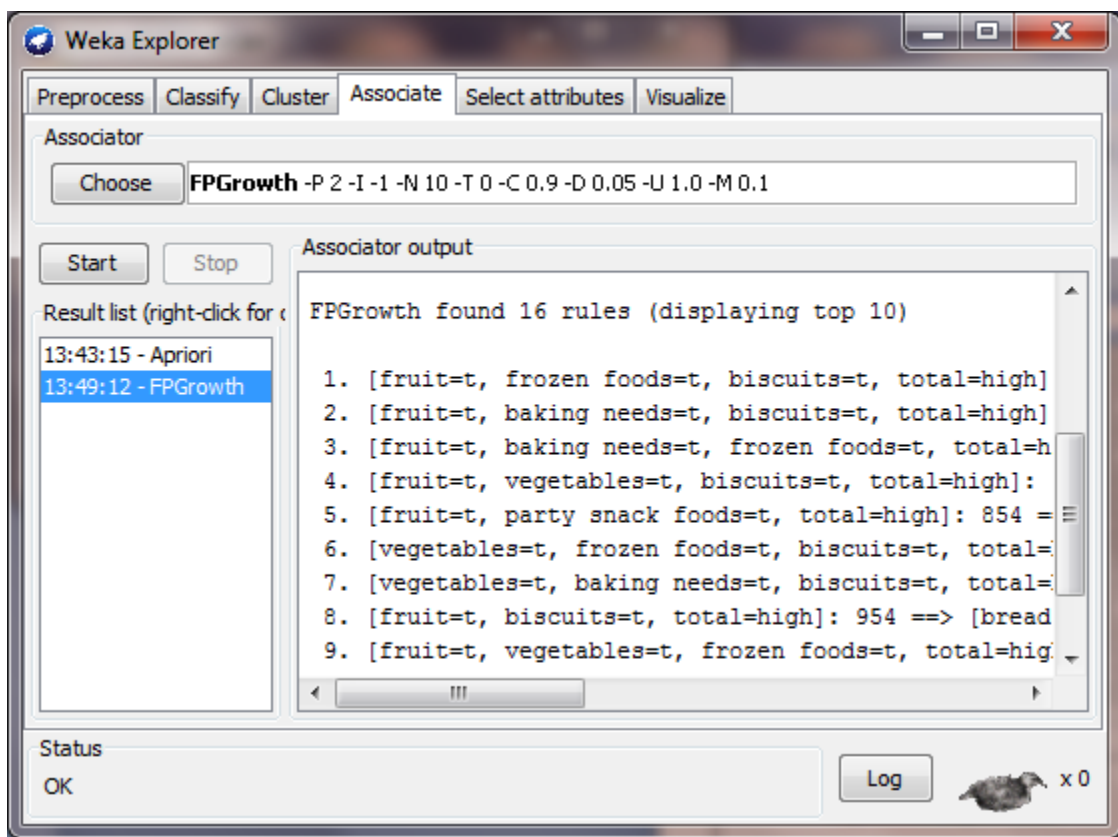
SuperMarket data set

No.	1: department1 Nominal	2: department2 Nominal	3: department3 Nominal	4: department4 Nominal	5: department5 Nominal
1					
2	t				
3					
4	t				
5					
6			t		
7	t				
8					
9	t		t		
10					
11					
12	t				
13	t	t			
14					
15					
16	t				t
17					
18	t		t		
19	t				
20	t				
21		t			t
22	t	t			
23					

**OUTPUT:
Apriori Algorithm**



FP-Growth Algorithm:



Result:

Thus association between data and to find the frequent item set for text mining was found.

EX. No: 11

DESIGN OF FACT AND DIMENSION TABLES

Aim:

To design fact and dimension tables.

Fact Table :

A fact table is used in the dimensional model in data warehouse design. A fact table is found at the center of a star schema or snowflake schema surrounded by dimension tables. A fact table consists of facts of a particular business process e.g., sales revenue by month by product. Facts are also known as measurements or metrics. A fact table record captures a measurement or a metric.

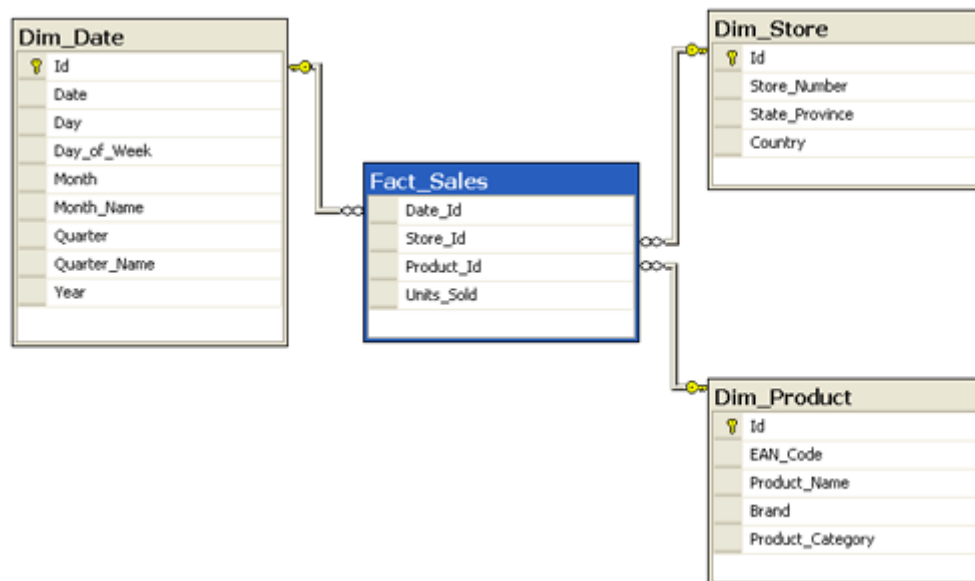
Designing fact table steps

Here is overview of four steps to designing a fact table:

1. **Choosing business process to model** – The first step is to decide what business process to model by gathering and understanding business needs and available data
2. **Declare the grain** – by declaring a grain means describing exactly what a fact table record represents
3. **Choose the dimensions** – once grain of fact table is stated clearly, it is time to determine dimensions for the fact table.
4. **Identify facts** – identify carefully which facts will appear in the fact table.

Fact table FACT_SALES that has a grain which gives us a number of units sold by date, by store and by product.

All other tables such as DIM_DATE, DIM_STORE and DIM_PRODUCT are dimensions tables. This schema is known as the star schema.



Result: Thus design fact and dimension tables are created.

EX. No: 12

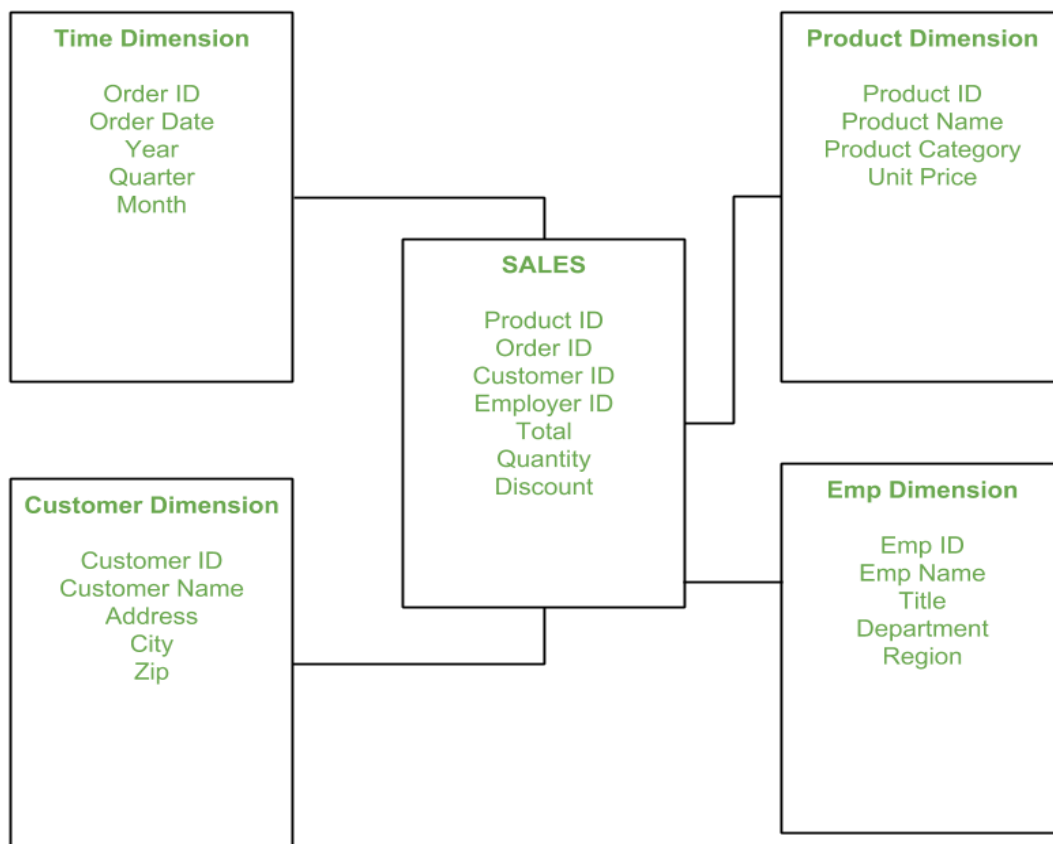
GENERATING GRAPHS FOR STAR SCHEMA

Aim:

To generate graphs for star schema.

Introduction:

Star schema is the fundamental schema among the data mart schema and it is simplest. This schema is widely used to develop or build a data warehouse and dimensional data marts. It includes one or more fact tables indexing any number of dimensional tables. The star schema is a necessary case of the snowflake schema. It is also efficient for handling basic queries. It is said to be star as its physical model resembles to the star shape having a fact table at its center and the dimension tables at its peripheral representing the star's points.



In the above demonstration, SALES is a fact table having attributes i.e. (Product ID, Order ID, Customer ID, Employer ID, Total, Quantity, Discount) which references to the dimension tables. **Employee dimension table** contains the attributes: Emp ID, Emp Name, Title, Department and Region. Product dimension table contains the attributes: Product ID, Product Name, Product Category, Unit Price. Customer dimension table contains the attributes: Customer ID, Customer Name, Address, City, Zip. Time dimension table contains the attributes: Order ID, Order Date, Year, Quarter, Month.

In Star Schema, Business process data, that holds the quantitative data about a business is distributed in fact tables, and dimensions which are descriptive characteristics related to fact data. Sales price, sale quantity, distant, speed, weight, and weight measurements are few examples of fact data in star schema. Often, A Star Schema having multiple dimensions is termed as Centipede Schema. It is easy to handle a star schema which has dimensions of few attributes.

Result: Thus the graphs for star schema are generated.