PRACTICAL 1

Roll No.: K034	Name: Anushka Mirajkar
Class: B.Tech Cyber Security	Batch: A2
Date of Practical: 08/01/2022	Date of Submission: 12/01/2022
Grade:	

Aim: To study ARFF file format and some features of WEKA Tool.

Prerequisite:

- Understanding of the schema diagram

Outcome: After successful completion of this experiment, students will be able to

- Understand the ARFF file format used in WEKA Tool and create one from an existing CSV file.
- Understand the usage of some filters in preprocessing tab of WEKA Tool.

Theory:

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

Weka can read the files that are in the Attribute-Relation File Format (ARFF). An ARFF (Attribute-Relation File Format) file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files were developed by the Machine Learning Project at the Department of Computer Science of The University of Waikato for use with the Weka machine learning software.

ARFF files have two distinct sections. The first section is the Header information, which is followed the Data information.

The Header of the ARFF file contains the name of the relation, a list of the attributes (the columns in the data), and their types. Lines that begin with a % are comments.

The ARFF Header section of the file contains the relation declaration and attribute declarations.

The @relation Declaration

The relation name is defined as the first line in the ARFF file. The format is:

@relation <relation-name>

where <relation-name> is a string. The string must be quoted if the name includes spaces.

The @attribute Declarations

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

@attribute <attribute-name> <datatype>

where the <attribute-name> must start with an alphabetic character. If spaces are to be included in the name then the entire name must be quoted.

The <datatype> can be any of the four types currently (version 3.2.1) supported by Weka:

- Numeric - can be real or integer numbers
- <nominal-specification> - are defined by providing an <nominal-specification> listing the possible values: {<nominal-name1>, <nominal-name2>, <nominal-name3>, ...}

  For example, the class value of the Weather dataset can be defined as follows:

  @ATTRIBUTE class {Play-Yes,Play-No}
- String- allow us to create attributes containing arbitrary textual values. This is very useful in text-mining applications

  @ATTRIBUTE LCC string
- date [<date-format>] - declarations take the form

  @attribute <name> date [<date-format>]

  where <name> is the name for the attribute and <date-format> is an optional string specifying how date values should be parsed and printed (this is the same format used by SimpleDateFormat). The default format string accepts the ISO-8601 combined date and time format: "yyyy-MM-dd'T'HH:mm:ss".

ARFF Data Section

The ARFF Data section of the file contains the data declaration line and the actual instance lines.

The @data Declaration - is a single line denoting the start of the data segment in the file. The format is:

@data

The instance data is represented on a single line, with carriage returns denoting the end of the instance. Attribute values for each instance are delimited by commas. They must appear in the order that they were declared in the header section (i.e. the data corresponding to the nth @attribute declaration is always the nth field of the attribute). Missing values are represented by a single question mark, as in:

@data

sunny,85,?,FALSE,no

An example header on the Weather dataset looks like this:

% Title: Weather Database

% Creator: Ashwini Rao

% Date: Dec 2021

@relation weather_training

@attribute outlook {sunny, overcast, rainy}

@attribute temperature numeric

@attribute humidity numeric

@attribute windy {TRUE,FALSE}

@attribute play {yes,no}

@data

sunny,85,85,FALSE,no

sunny,80,90,TRUE,no

overcast,83,86,FALSE,yes

rainy,70,96,FALSE,yes

rainy,68,80,FALSE,yes

rainy,65,70,TRUE,no

overcast,64,65,TRUE,yes

sunny,72,95,FALSE,no

sunny,69,70,FALSE,yes

rainy,75,80,FALSE,yes

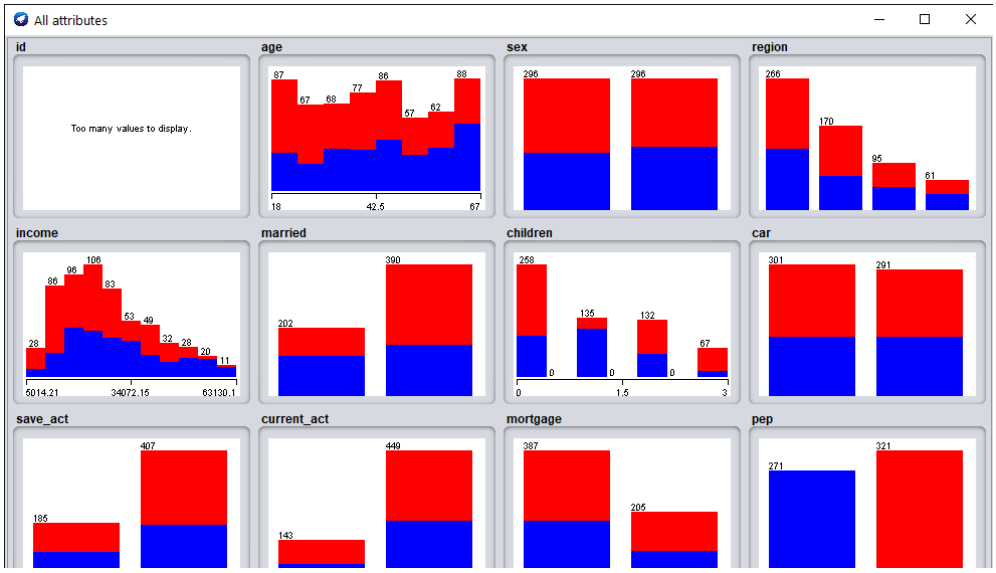
sunny,75,70,TRUE,yes

overcast,72,90,TRUE,yes

overcast,81,75,FALSE,yes

rainy,71,91,TRUE,no

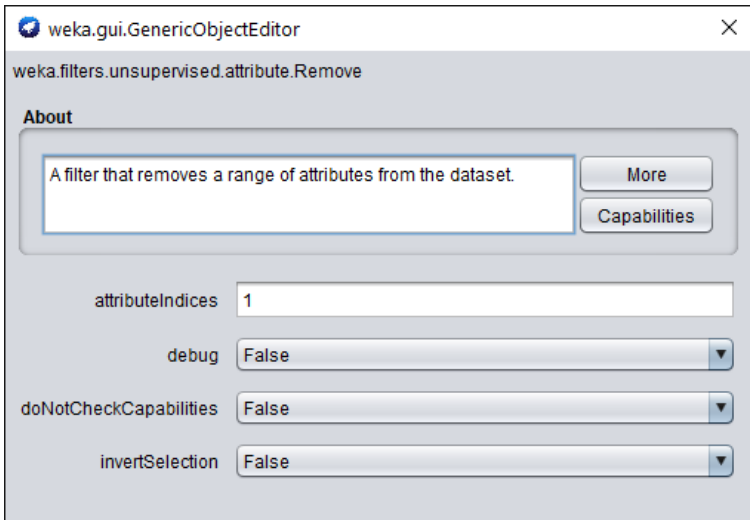
1. Using the appropriate header and data section create an ARFF file from the given CSV file (Bank data set). Load the file onto the Weka tool and note the basic statistics for each attribute computed by the tool.



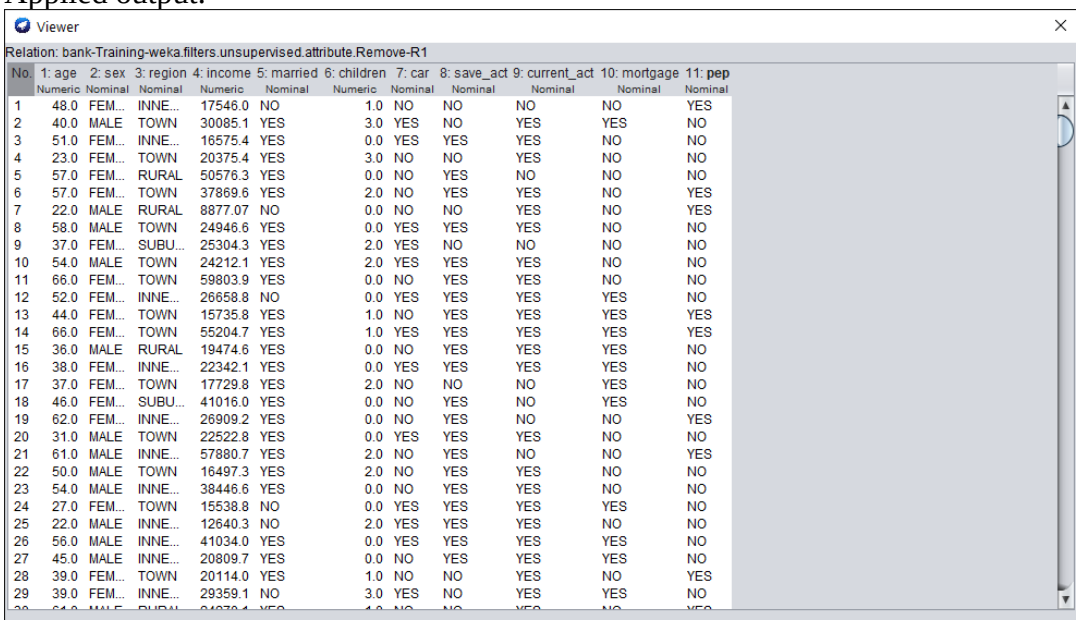
2. Apply the below mentioned filters on the data set.

- Remove the column attribute ID

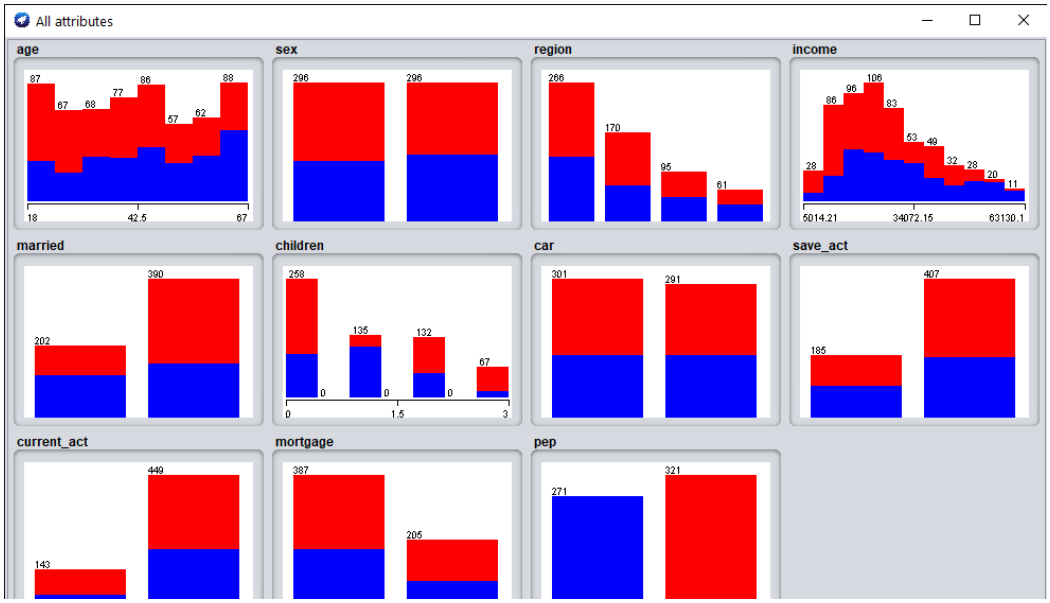
Filter-unsupervised-attribute-remove



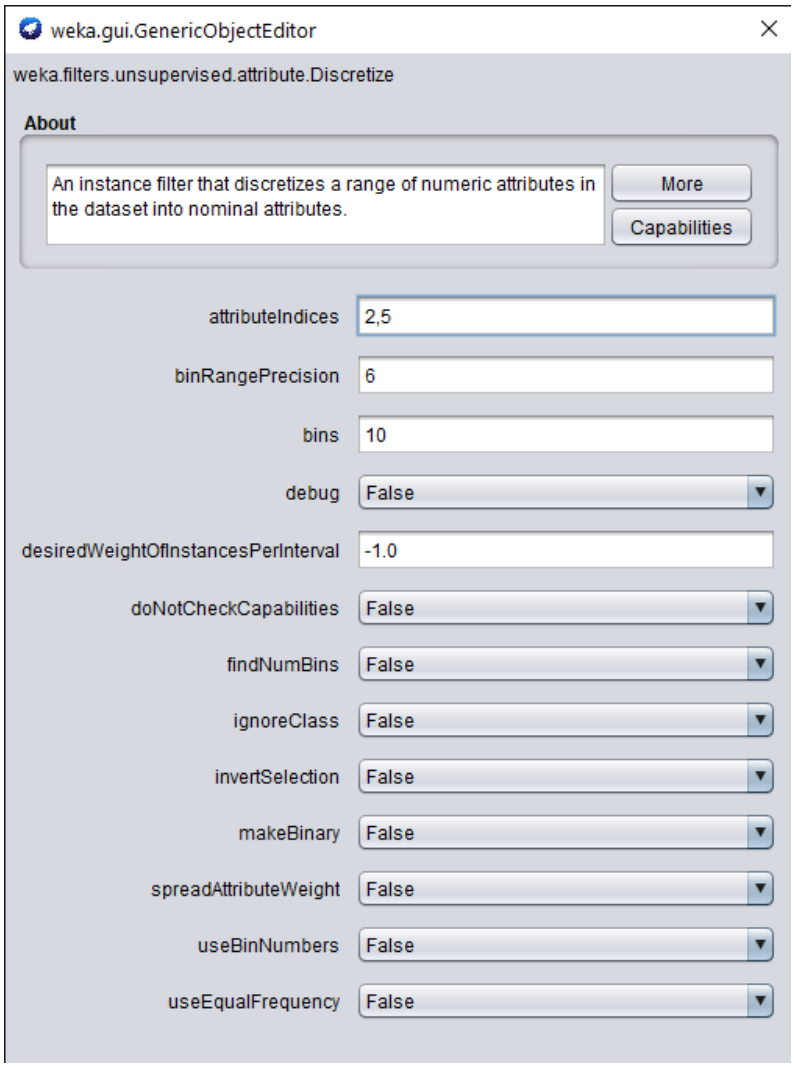
Applied output:



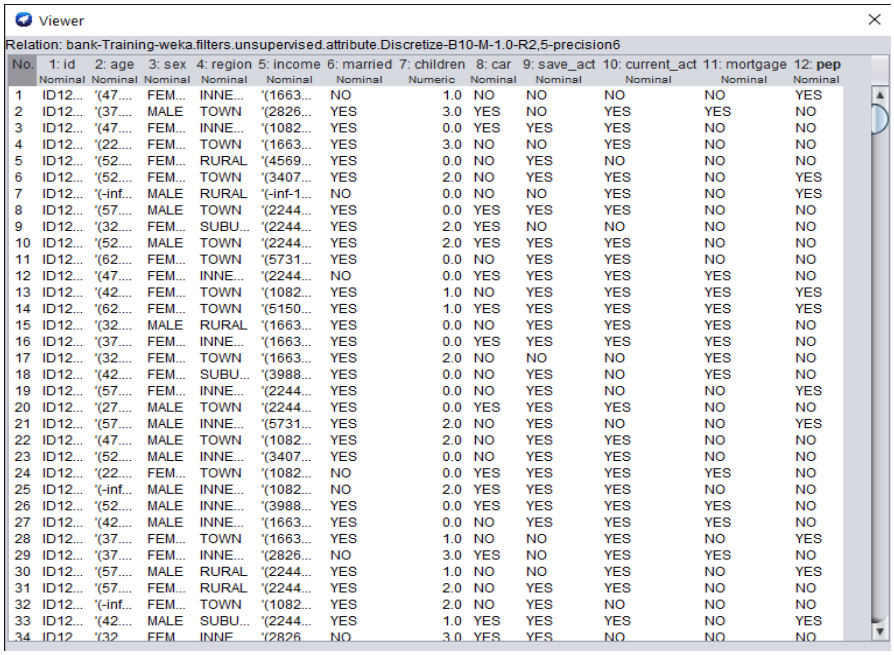
No	1: age	2: sex	3: region	4: income	5: married	6: children	7: car	8: save_act	9: current_act	10: mortgage	11: pep
	Numeric	Nominal	Nominal	Numeric	Nominal	Numeric	Nominal	Nominal	Nominal	Nominal	Nominal
1	48.0	FEM...	INNE...	17546.0	NO	1.0	NO	NO	NO	NO	YES
2	40.0	MALE	TOWN	30085.1	YES	3.0	YES	NO	YES	YES	NO
3	51.0	FEM...	INNE...	16575.4	YES	0.0	YES	YES	YES	NO	NO
4	23.0	FEM...	TOWN	20375.4	YES	3.0	NO	NO	YES	NO	NO
5	57.0	FEM...	RURAL	50576.3	YES	0.0	NO	YES	NO	NO	NO
6	57.0	FEM...	TOWN	37869.6	YES	2.0	NO	YES	YES	NO	YES
7	22.0	MALE	RURAL	8877.07	NO	0.0	NO	NO	YES	NO	YES
8	58.0	MALE	TOWN	24946.6	YES	0.0	YES	YES	YES	NO	NO
9	37.0	FEM...	SUBU...	25304.3	YES	2.0	YES	NO	NO	NO	NO
10	54.0	MALE	TOWN	24212.1	YES	2.0	YES	YES	YES	NO	NO
11	66.0	FEM...	TOWN	59803.9	YES	0.0	NO	YES	YES	NO	NO
12	52.0	FEM...	INNE...	26658.8	NO	0.0	YES	YES	YES	YES	NO
13	44.0	FEM...	TOWN	15735.8	YES	1.0	NO	YES	YES	YES	YES
14	66.0	FEM...	TOWN	55204.7	YES	1.0	YES	YES	YES	YES	YES
15	36.0	MALE	RURAL	19474.6	YES	0.0	NO	YES	YES	YES	NO
16	38.0	FEM...	INNE...	22342.1	YES	0.0	YES	YES	YES	YES	NO
17	37.0	FEM...	TOWN	17729.8	YES	2.0	NO	NO	NO	YES	NO
18	46.0	FEM...	SUBU...	41016.0	YES	0.0	NO	YES	NO	YES	NO
19	62.0	FEM...	INNE...	26909.2	YES	0.0	NO	YES	NO	NO	YES
20	31.0	MALE	TOWN	22522.8	YES	0.0	YES	YES	YES	NO	NO
21	61.0	MALE	INNE...	57880.7	YES	2.0	NO	YES	NO	NO	YES
22	50.0	MALE	TOWN	16497.3	YES	2.0	NO	YES	YES	NO	NO
23	54.0	MALE	INNE...	38446.6	YES	0.0	NO	YES	YES	NO	NO
24	27.0	FEM...	TOWN	15538.8	NO	0.0	YES	YES	YES	YES	NO
25	22.0	MALE	INNE...	12640.3	NO	2.0	YES	YES	YES	NO	NO
26	56.0	MALE	INNE...	41034.0	YES	0.0	YES	YES	YES	YES	NO
27	45.0	MALE	INNE...	20809.7	YES	0.0	NO	YES	YES	YES	NO
28	39.0	FEM...	TOWN	20114.0	YES	1.0	NO	NO	YES	NO	YES
29	39.0	FEM...	INNE...	29359.1	NO	3.0	YES	NO	YES	YES	NO



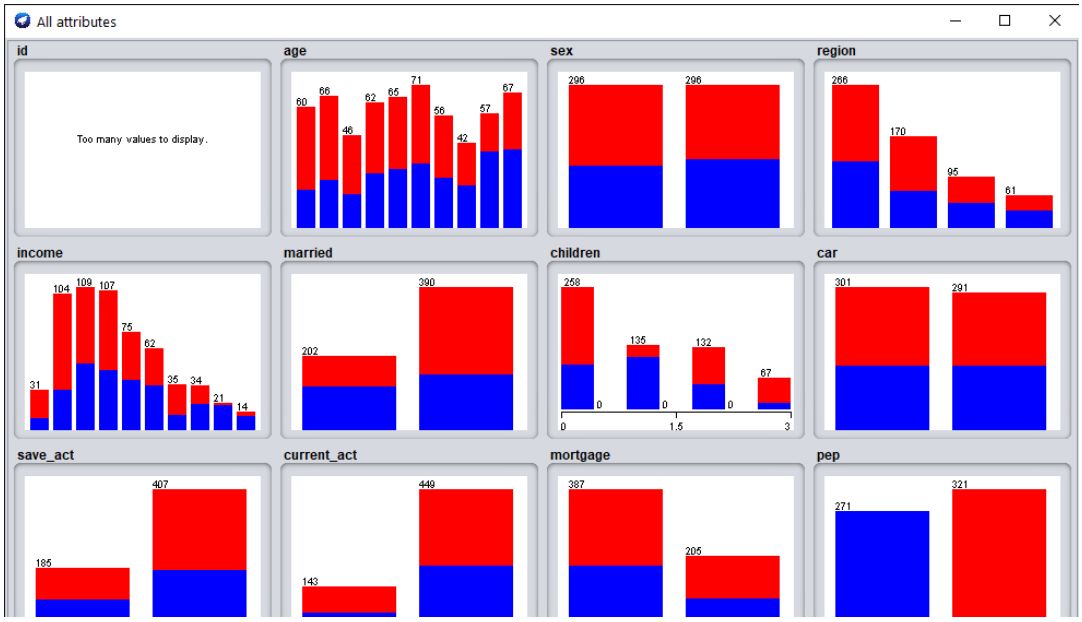
- Discretize the column attributes, income and age
  Filter-unsupervised-attribute-discretize



Applied Output:

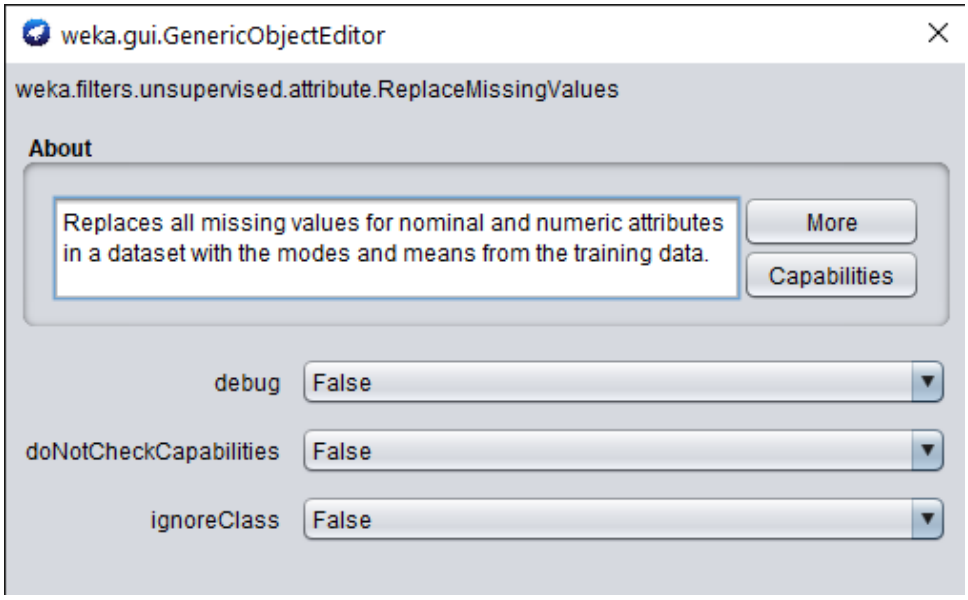


No.	1: id	2: age	3: sex	4: region	5: income	6: married	7: children	8: car	9: save_act	10: current_act	11: mortgage	12: pep
	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Numeric	Nominal	Nominal	Nominal	Nominal	Nominal
1	ID12...	'(47....	FEM...	INNE...	'(1663...	NO	1.0	NO	NO	NO	NO	YES
2	ID12...	'(37....	MALE	TOWN	'(2826...	YES	3.0	YES	NO	YES	YES	NO
3	ID12...	'(47....	FEM...	INNE...	'(1082...	YES	0.0	YES	YES	YES	NO	NO
4	ID12...	'(22....	FEM...	TOWN	'(1663...	YES	3.0	NO	NO	YES	NO	NO
5	ID12...	'(52....	FEM...	RURAL	'(4569...	YES	0.0	NO	YES	NO	NO	NO
6	ID12...	'(52....	FEM...	TOWN	'(3407...	YES	2.0	NO	YES	YES	NO	YES
7	ID12...	'(-inf...	MALE	RURAL	'(-inf-1...	NO	0.0	NO	NO	YES	NO	YES
8	ID12...	'(57....	MALE	TOWN	'(2244...	YES	0.0	YES	YES	YES	NO	NO
9	ID12...	'(32....	FEM...	SUBU...	'(2244...	YES	2.0	YES	NO	NO	NO	NO
10	ID12...	'(52....	MALE	TOWN	'(2244...	YES	2.0	YES	YES	YES	NO	NO
11	ID12...	'(62....	FEM...	TOWN	'(5731...	YES	0.0	NO	YES	YES	NO	NO
12	ID12...	'(47....	FEM...	INNE...	'(2244...	NO	0.0	YES	YES	YES	YES	NO
13	ID12...	'(42....	FEM...	TOWN	'(1082...	YES	1.0	NO	YES	YES	YES	YES
14	ID12...	'(62....	FEM...	TOWN	'(5150...	YES	1.0	YES	YES	YES	YES	YES
15	ID12...	'(32....	MALE	RURAL	'(1663...	YES	0.0	NO	YES	YES	YES	NO
16	ID12...	'(37....	FEM...	INNE...	'(1663...	YES	0.0	YES	YES	YES	YES	NO
17	ID12...	'(32....	FEM...	TOWN	'(1663...	YES	2.0	NO	NO	NO	YES	NO
18	ID12...	'(42....	FEM...	SUBU...	'(3988...	YES	0.0	NO	YES	NO	YES	NO
19	ID12...	'(57....	FEM...	INNE...	'(2244...	YES	0.0	NO	YES	NO	NO	YES
20	ID12...	'(27....	MALE	TOWN	'(2244...	YES	0.0	YES	YES	YES	NO	NO
21	ID12...	'(57....	MALE	INNE...	'(5731...	YES	2.0	NO	YES	NO	NO	YES
22	ID12...	'(47....	MALE	TOWN	'(1082...	YES	2.0	NO	YES	YES	NO	NO
23	ID12...	'(52....	MALE	INNE...	'(3407...	YES	0.0	NO	YES	YES	NO	NO
24	ID12...	'(22....	FEM...	TOWN	'(1082...	NO	0.0	YES	YES	YES	YES	NO
25	ID12...	'(-inf...	MALE	INNE...	'(1082...	NO	2.0	YES	YES	YES	NO	NO
26	ID12...	'(52....	MALE	INNE...	'(3988...	YES	0.0	YES	YES	YES	YES	NO
27	ID12...	'(42....	MALE	INNE...	'(1663...	YES	0.0	NO	YES	YES	YES	NO
28	ID12...	'(37....	FEM...	TOWN	'(1663...	YES	1.0	NO	NO	YES	NO	YES
29	ID12...	'(37....	FEM...	INNE...	'(2826...	NO	3.0	YES	NO	YES	YES	NO
30	ID12...	'(57....	MALE	RURAL	'(2244...	YES	1.0	NO	NO	YES	NO	YES
31	ID12...	'(57....	FEM...	RURAL	'(2244...	YES	2.0	NO	YES	YES	NO	NO
32	ID12...	'(-inf...	FEM...	TOWN	'(1082...	YES	2.0	NO	YES	NO	NO	NO
33	ID12...	'(42....	MALE	SUBU...	'(2244...	YES	1.0	YES	YES	YES	NO	YES
34	ID12...	'(32....	FEM...	INNE...	'(2826...	NO	3.0	YES	YES	NO	NO	NO

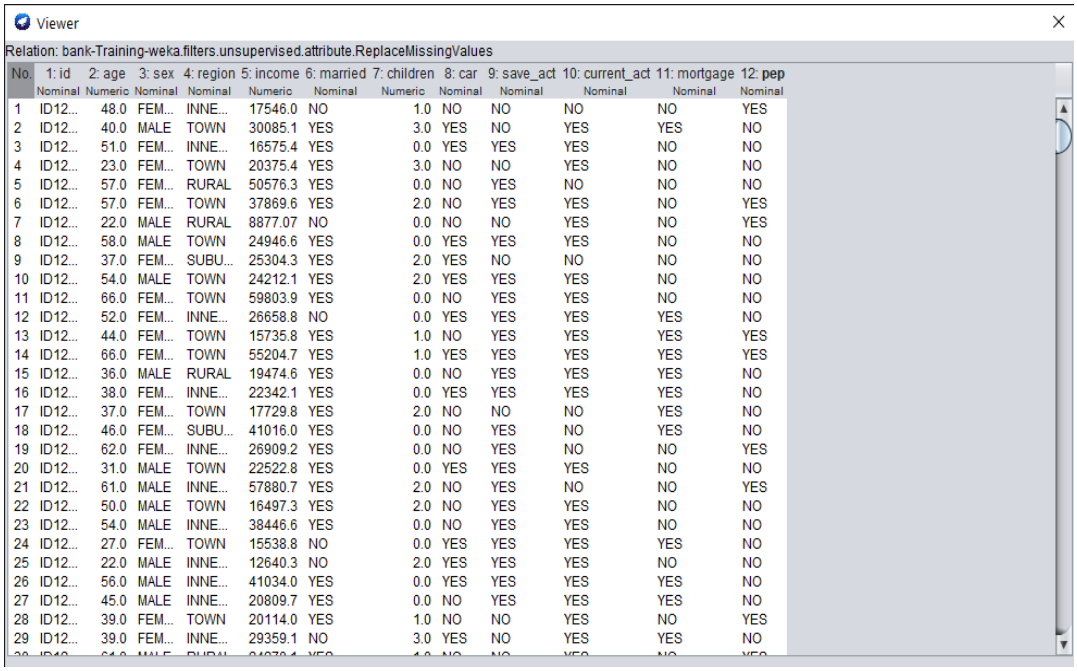


- Create a new data set with some missing values. Apply the replace missing value filter.

Filter-unsupervised-attribute-replacemissingvalues



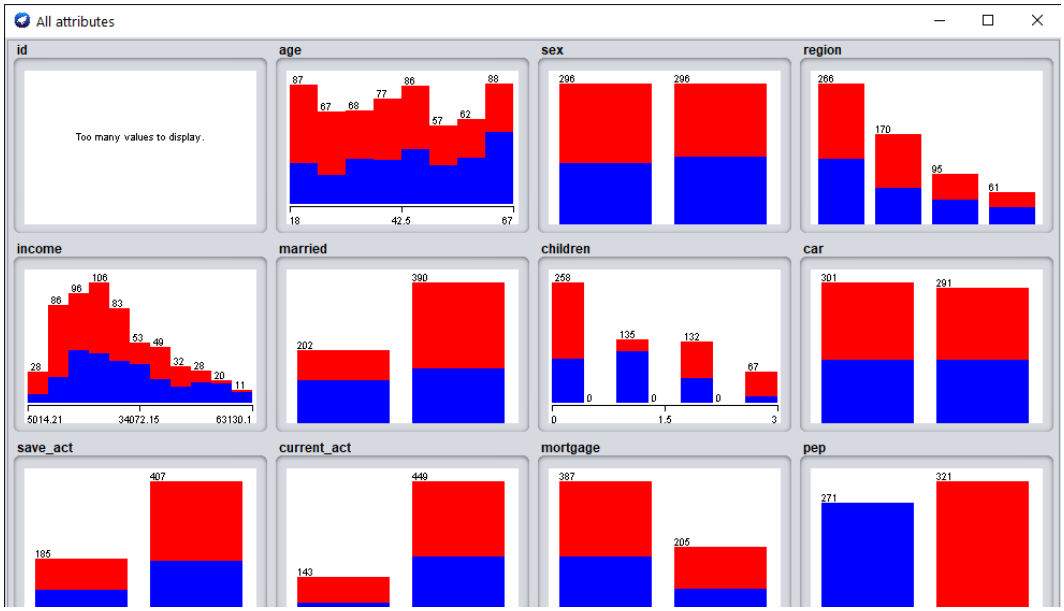
Applied Output:



Viewer

Relation: bank-Training-weka.filters.unsupervised.attribute.ReplaceMissingValues

No.	1: id	2: age	3: sex	4: region	5: income	6: married	7: children	8: car	9: save_act	10: current_act	11: mortgage	12: pep
	Nominal	Numeric	Nominal	Nominal	Numeric	Nominal	Numeric	Nominal	Nominal	Nominal	Nominal	Nominal
1	ID12...	48.0	FEM...	INNE...	17546.0	NO	1.0	NO	NO	NO	NO	YES
2	ID12...	40.0	MALE	TOWN	30085.1	YES	3.0	YES	NO	YES	YES	NO
3	ID12...	51.0	FEM...	INNE...	16575.4	YES	0.0	YES	YES	YES	NO	NO
4	ID12...	23.0	FEM...	TOWN	20375.4	YES	3.0	NO	NO	YES	NO	NO
5	ID12...	57.0	FEM...	RURAL	50576.3	YES	0.0	NO	YES	NO	NO	NO
6	ID12...	57.0	FEM...	TOWN	37869.6	YES	2.0	NO	YES	YES	NO	YES
7	ID12...	22.0	MALE	RURAL	8877.07	NO	0.0	NO	NO	YES	NO	YES
8	ID12...	58.0	MALE	TOWN	24946.6	YES	0.0	YES	YES	YES	NO	NO
9	ID12...	37.0	FEM...	SUBU...	25304.3	YES	2.0	YES	NO	NO	NO	NO
10	ID12...	54.0	MALE	TOWN	24212.1	YES	2.0	YES	YES	YES	NO	NO
11	ID12...	66.0	FEM...	TOWN	59803.9	YES	0.0	NO	YES	YES	NO	NO
12	ID12...	52.0	FEM...	INNE...	26658.8	NO	0.0	YES	YES	YES	YES	NO
13	ID12...	44.0	FEM...	TOWN	15735.8	YES	1.0	NO	YES	YES	YES	YES
14	ID12...	66.0	FEM...	TOWN	55204.7	YES	1.0	YES	YES	YES	YES	YES
15	ID12...	36.0	MALE	RURAL	19474.6	YES	0.0	NO	YES	YES	YES	NO
16	ID12...	38.0	FEM...	INNE...	22342.1	YES	0.0	YES	YES	YES	YES	NO
17	ID12...	37.0	FEM...	TOWN	17729.8	YES	2.0	NO	NO	NO	YES	NO
18	ID12...	46.0	FEM...	SUBU...	41016.0	YES	0.0	NO	YES	NO	YES	NO
19	ID12...	62.0	FEM...	INNE...	26909.2	YES	0.0	NO	YES	NO	NO	YES
20	ID12...	31.0	MALE	TOWN	22522.8	YES	0.0	YES	YES	YES	NO	NO
21	ID12...	61.0	MALE	INNE...	57880.7	YES	2.0	NO	YES	NO	NO	YES
22	ID12...	50.0	MALE	TOWN	16497.3	YES	2.0	NO	YES	YES	NO	NO
23	ID12...	54.0	MALE	INNE...	38446.6	YES	0.0	NO	YES	YES	NO	NO
24	ID12...	27.0	FEM...	TOWN	15538.8	NO	0.0	YES	YES	YES	YES	NO
25	ID12...	22.0	MALE	INNE...	12640.3	NO	2.0	YES	YES	YES	NO	NO
26	ID12...	56.0	MALE	INNE...	41034.0	YES	0.0	YES	YES	YES	YES	NO
27	ID12...	45.0	MALE	INNE...	20809.7	YES	0.0	NO	YES	YES	YES	NO
28	ID12...	39.0	FEM...	TOWN	20114.0	YES	1.0	NO	NO	YES	NO	YES
29	ID12...	39.0	FEM...	INNE...	29359.1	NO	3.0	YES	NO	YES	YES	NO



Observations:

1. We used the remove attribute filter to remove columns.
2. We used the discretize filter to change numeric attributes to nominal attributes.
3. We replaced some values in the file with ? and applied the replace missing value filter to get it back.

Conclusion:

1. We learnt the process to convert .csv file into .arff file and how to apply filters in Weka and get the outputs

****