

DATA MINING

CS-572

SNEHA VIDHYASHEKAR ASU ID: 1211274841

Saket ashok dumber ASU ID: 1209656536

PROJECT – 1

SUBMISSION-3

FINAL REPORT

Under the guidance of Prof. Frank Calliss

Contents

[PROCEDURE 2](#_Toc486257929)

[PREPROCESS 6](#_Toc486257930)

[CLASSIFICATION 7](#_Toc486257931)

[BUILDING THE MODEL 10](#_Toc486257932)

[EVALUATING THE MODEL 11](#_Toc486257933)

# PROCEDURE

STEP 1: **Analyzing and preprocessing the training data set.**

Removed the first column ID attribute. Since ID is unique to each tuple in the data set, it is not considered during building the data model.

Labeling the attributes as A1, A2, A3,…..A5000

Target variable is named as LABEL.

To make the data set to be applicable to all types of data modeling and making easier to differentiate the target variable, I have changed the target variable from numeric to nominal type.

“Label” 🡪 { -1, 1 } numeric

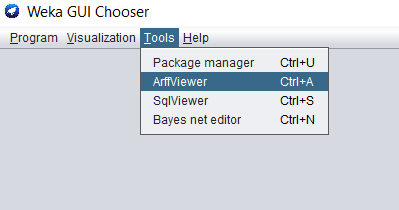
Now I have converted LABEL from numeric to “**nominal**”.

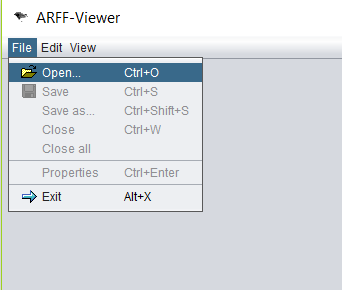
STEP 2: **Changing the data set format** compatible with the weka software.

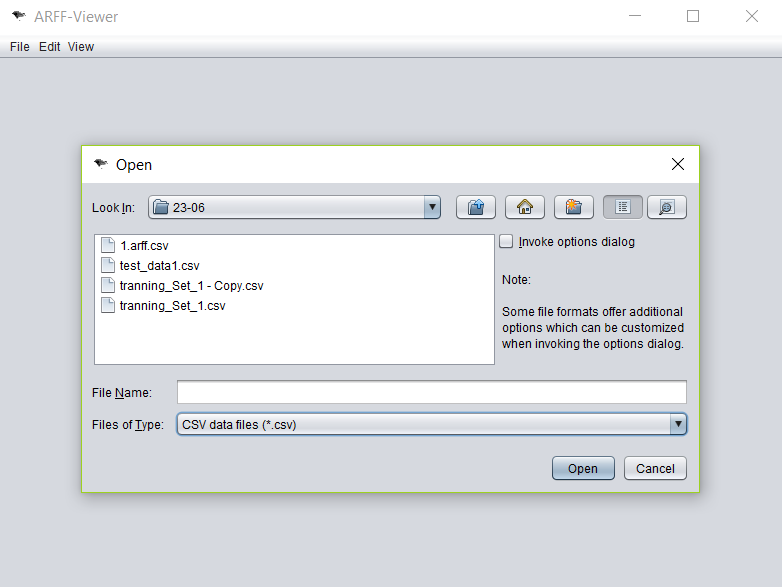
Training data was saved in .csv format. But Weka uses only **“.arff”** format.

Steps followed for this conversion is shown in the screenshots.

**SCREENSHOTS**







After selecting the file and asked for conversion, a file with “.arff “ format appears and is ready to be used in weka software to build the model.

STEP 3: **Building the model** using appropriate method (classification, clustering, associate .. )

STEP 4: **Train the model** using the training set that was converted into “.arff” format.

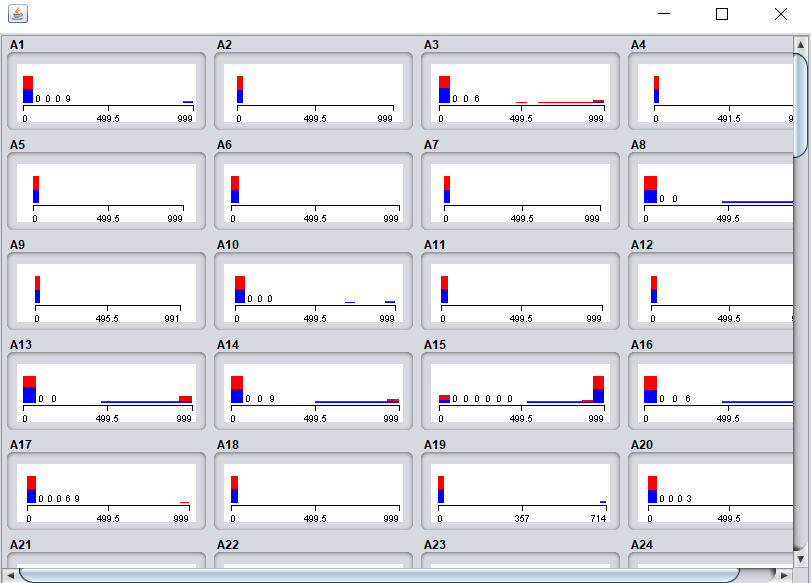
STEP 5: **Test the model** using test set / data that was separated from training data set and not used during model building.

STEP 6: **Evaluate the accuracy of the model** built. Measure the accuracy of the model using precision, confusion matrix and many other metrics used to evaluate the model.

# **PREPROCESS**

1. Select preprocess from weka GUI.

It allows us to select the “train.arff” file to analyze the data w.r.t the target/label variable.

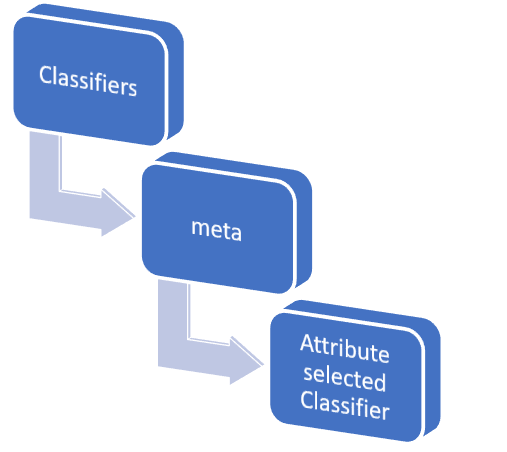
Graph of all attributes w.r.t target attribute can be visualized.

# CLASSIFICATION

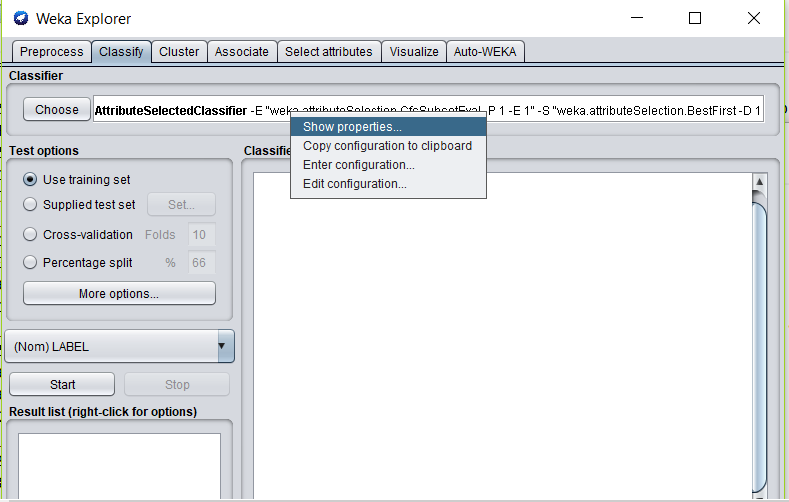
By using classification I have found the best result with 99.2308 % accuracy.

Steps followed are:

* Select classify on the menu window.
* Choose the classifier with the drop-down window as shown below.



* Now changing the minute details of the classifier (attribute classifier) which is suitable to the training data set.

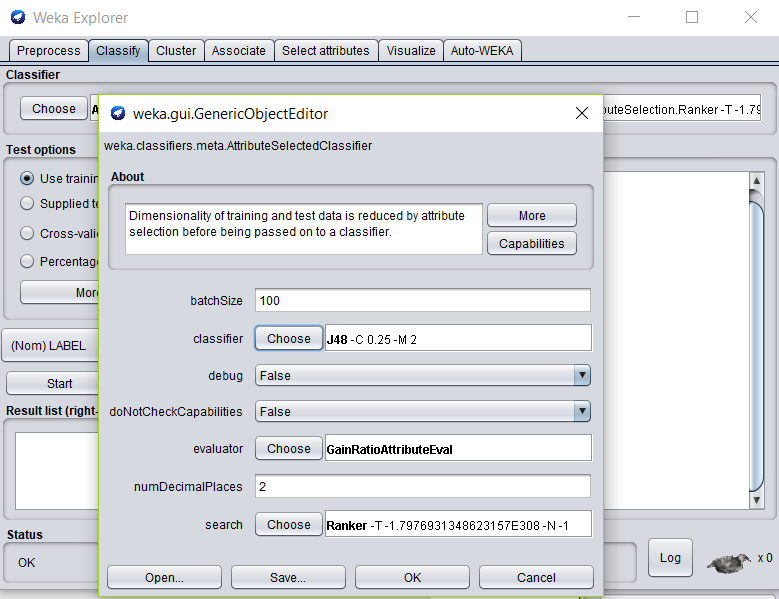


* Choose the properties to change the default settings in the “Attribute Selected Classifier” method.
* Classifier – j48

Evaluator = Gain Ratio

Search method = ranker

* Selects the first 100 attributes after ranking them w.r.t it contribution to the target variable.



* Click OK and continue.

# BUILDING THE MODEL

Once the classifier method is specified along with the number of attributes to be used in building the model according to the gain ratio.

Files attached in the document are:

1. Training data set (modified) 🡪 .csv and .arff format
2. Java file 🡪 used to build the model
3. Screenshots
4. Ranking of all attributes in excel sheet.
5. Evaluation metrics used to evaluate the model

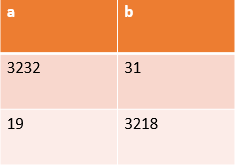
Steps followed are:

1. Rank the attributes based on the Gain Ratio.
2. List of all the attributes according to ranks w.r.t target attribute is attached in the excel sheet.

# EVALUATING THE MODEL

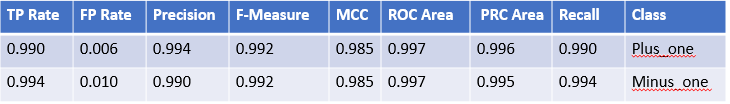
# Model is evaluated using the test data set which was not fed to the model during training.

=== Confusion Matrix ===

a = 1

b = -1

=== Detailed Accuracy by Class ===



**EVALUATION METRICS**

Correctly Classified Instances 6450 99.2308 %

Incorrectly Classified Instances 50 0.7692 %

Kappa statistic 0.9846

K&B Relative Info Score 633592.8053 %

K&B Information Score 6335.855 bits 0.9747 bits/instance

Class complexity | order 0 6499.925 bits 1 bits/instance

Class complexity | scheme 336.8349 bits 0.0518 bits/instance

Complexity improvement(Sf) 6163.0901 bits 0.9482 bits/instance

Mean absolute error 0.0143

Root mean squared error 0.0847

Relative absolute error 2.8683 %

Root relative squared error 16.936 %

Total Number of Instances 6500

**CONCLUSION**

Training data set had 6500 instances and 5000 attributes and one label class.

Test data set had 1000 instances.

After applying all the above-mentioned data mining techniques, a model was built to predict the classification of previously unseen data set with 99.2308% precision.