German Credit Card Data Mining Using Weka

Description of the German credit dataset.

1. Title: German Credit data

2. Source Information

Professor Dr. Hans Hofmann

Institut f"ur Statistik und "Okonometrie

Universit"at Hamburg

FB Wirtschaftswissenschaften

Von-Melle-Park 5

2000 Hamburg 13

Original Source:
http://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)

3. Number of Instances: 1000

Two datasets are provided. the original dataset, in the form provided by Prof. Hofmann, contains categorical/symbolic attributes and is in the file "german.data".

For algorithms that need numerical attributes, Strathclyde University produced the file "german.data-numeric". This file has been edited and several indicator variables added to make it suitable for algorithms which cannot cope with categorical variables. Several attributes that are ordered categorical (such as attribute 17) have been coded as integer. This was the form used by StatLog.

6. Number of Attributes german: 20 (7 numerical, 13 categorical)

Number of Attributes german.numer: 24 (24 numerical)

7. Attribute description for german

Attribute 1: (qualitative)

Status of existing checking account

A11 : ... < 0 DM

A12 : 0 <= ... < 200 DM

A13 : ... >= 200 DM /

salary assignments for at least 1 year

A14 : no checking account

Attribute 2: (numerical)

Duration in month

Attribute 3: (qualitative)

Credit history

A30 : no credits taken/

all credits paid back duly

A31 : all credits at this bank paid back duly

A32 : existing credits paid back duly till now

A33 : delay in paying off in the past

A34 : critical account/

other credits existing (not at this bank)

Attribute 4: (qualitative)

Purpose

A40 : car (new)

A41 : car (used)

A42 : furniture/equipment

A43 : radio/television

A44 : domestic appliances

A45 : repairs

A46 : education

A47 : (vacation - does not exist?)

A48 : retraining

A49 : business

A410 : others

Attribute 5: (numerical)

Credit amount

Attribute 6: (qualitative)

Savings account/bonds

A61 : ... < 100 DM

A62 : 100 <= ... < 500 DM

A63 : 500 <= ... < 1000 DM

A64 : .. >= 1000 DM

A65 : unknown/ no savings account

Attribute 7: (qualitative)

Present employment since

A71 : unemployed

A72 : ... < 1 year

A73 : 1 <= ... < 4 years

A74 : 4 <= ... < 7 years

A75 : .. >= 7 years

Attribute 8: (numerical)

Installment rate in percentage of disposable income

Attribute 9: (qualitative)

Personal status and sex

A91 : male : divorced/separated

A92 : female : divorced/separated/married

A93 : male : single

A94 : male : married/widowed

A95 : female : single

Attribute 10: (qualitative)

Other debtors / guarantors

A101 : none

A102 : co-applicant

A103 : guarantor

Attribute 11: (numerical)

Present residence since

Attribute 12: (qualitative)

Property

A121 : real estate

A122 : if not A121 : building society savings agreement/
life insurance

A123 : if not A121/A122 : car or other, not in attribute 6

A124 : unknown / no property

Attribute 13: (numerical)

Age in years

Attribute 14: (qualitative)

Other installment plans

A141 : bank

A142 : stores

A143 : none

Attribute 15: (qualitative)

Housing

A151 : rent

A152 : own

A153 : for free

Attribute 16: (numerical)

Number of existing credits at this bank

Attribute 17: (qualitative)

Job

A171 : unemployed/ unskilled - non-resident

A172 : unskilled - resident

A173 : skilled employee / official

A174 : management/ self-employed/

highly qualified employee/ officer

Attribute 18: (numerical)

Number of people being liable to provide maintenance for

Attribute 19: (qualitative)

Telephone

A191 : none

A192 : yes, registered under the customers name

Attribute 20: (qualitative)

foreign worker

A201 : yes

A202 : no

8. Cost Matrix

This dataset requires use of a cost matrix (see below)

	1	2
1	0	1
2	5	0

(1 = Good, 2 = Bad)

the rows represent the actual classification and the columns

the predicted classification.

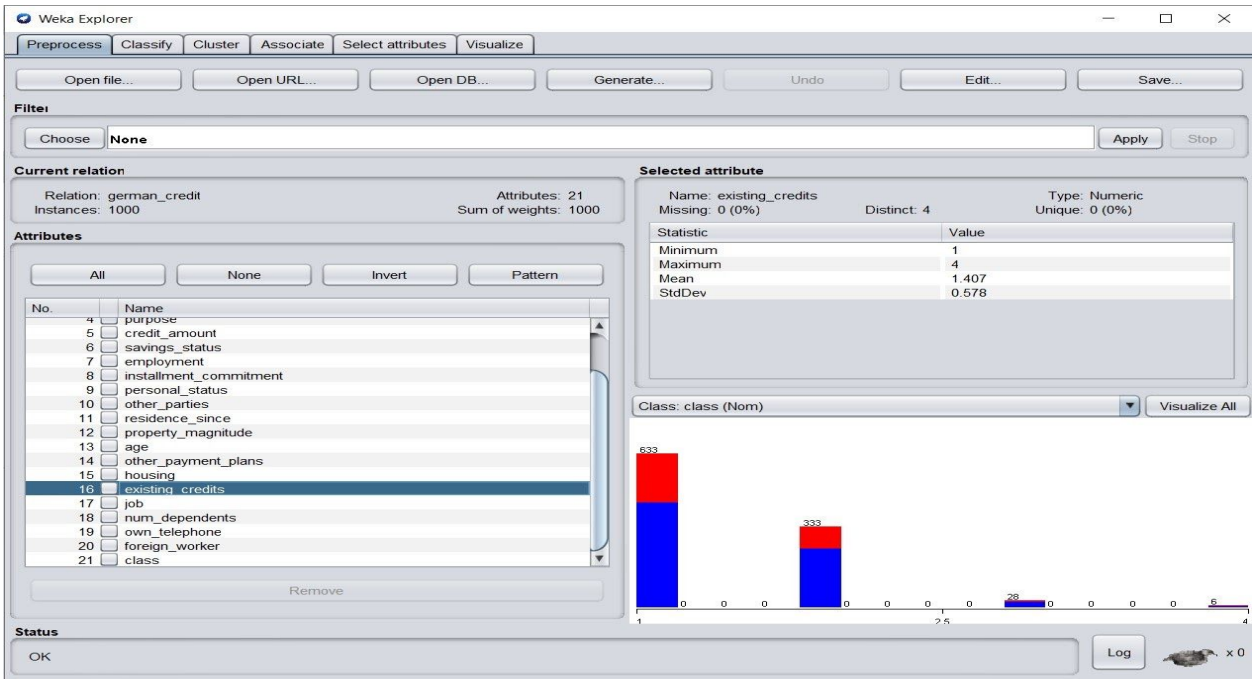
It is worse to class a customer as good when they are bad (5),

than it is to class a customer as bad when they are good (1).

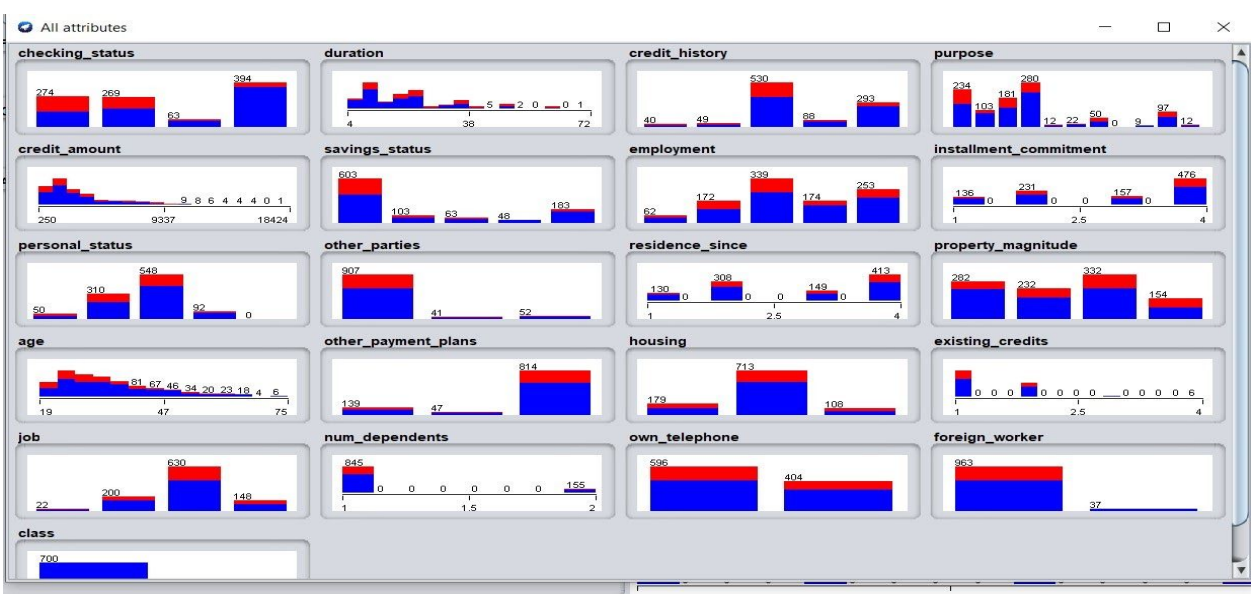
2. Preprocessing of data

Preprocessing techniques that can be applied on german dataset. The data that is collected from the field can contain 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.

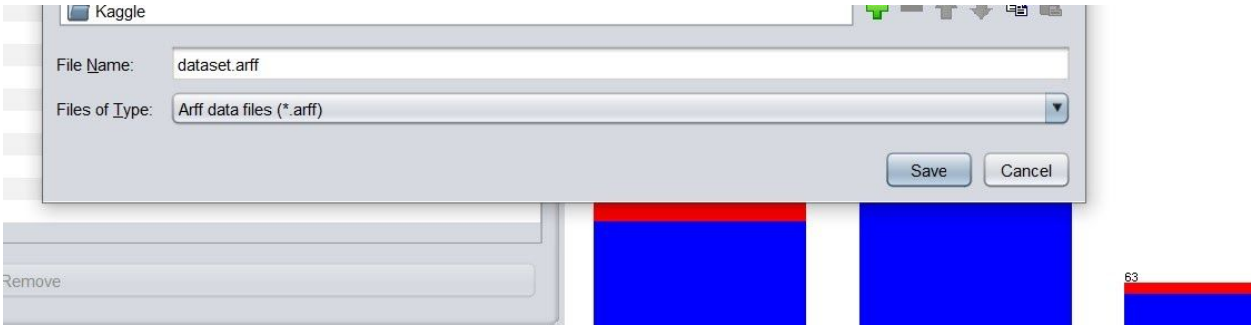
Saving dataset as dataset.arff



Visualization of a dataset.arff file which contains german data of its variables/instances.

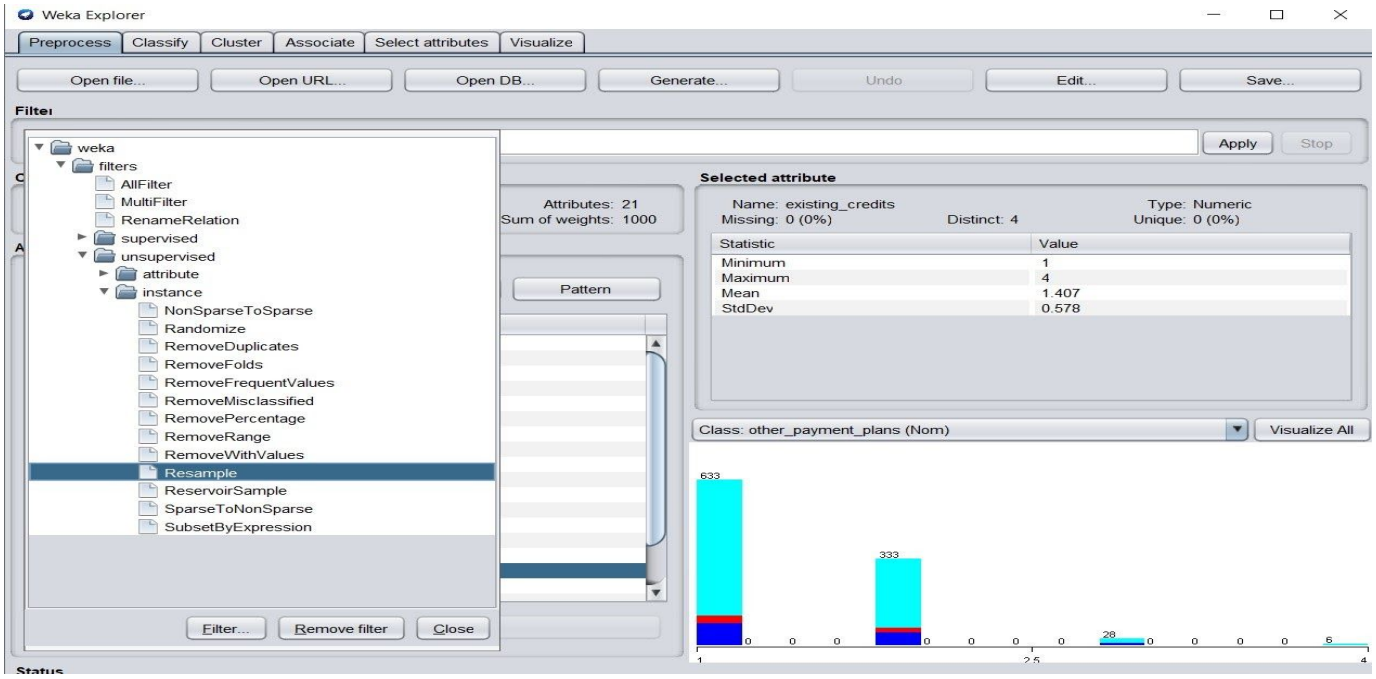


Dataset is saved as dataset.arff format for weka readable. The data is preprocessed and the data is being stored as dataset.arff in the computer.

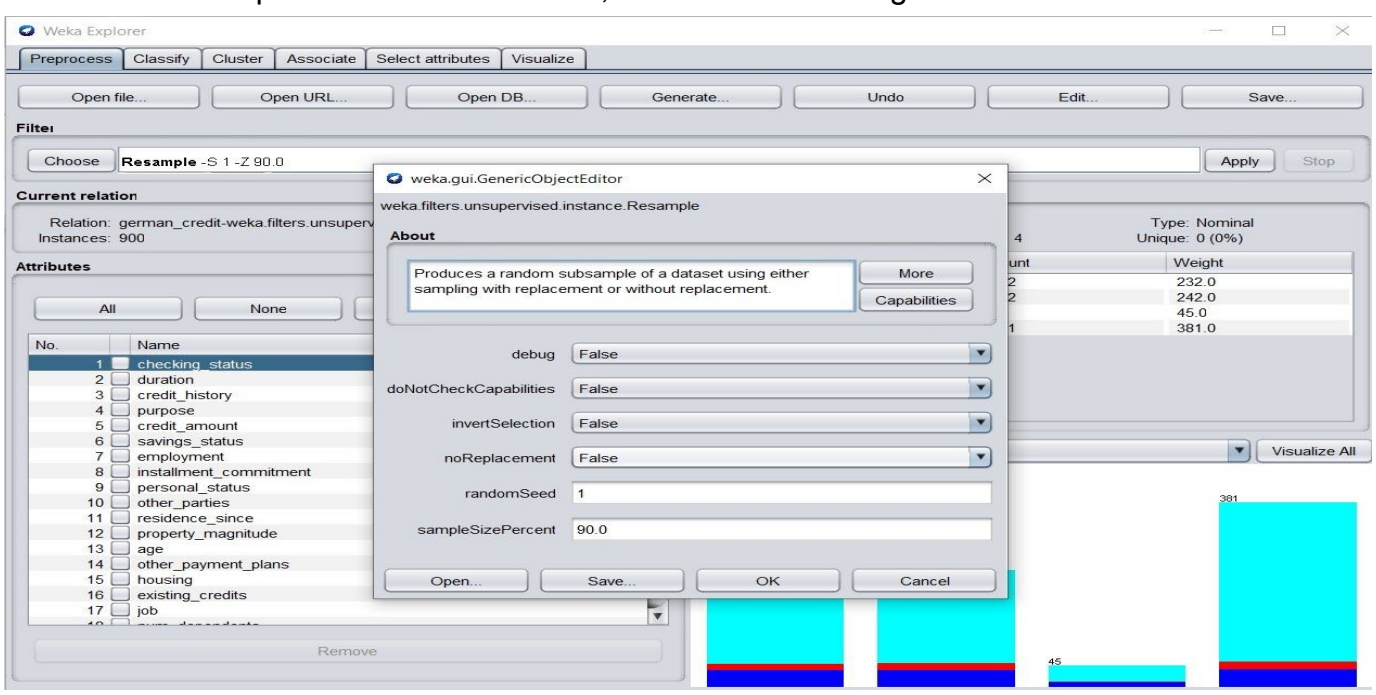


3. Splitting the data into training and testing

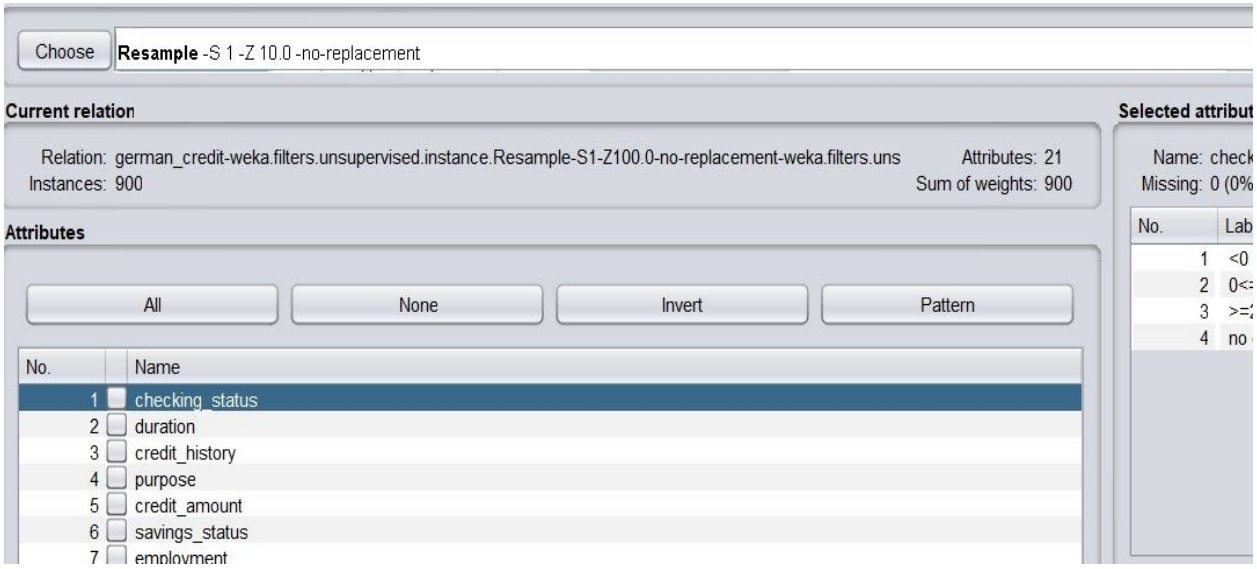
In weka, it is possible to divide data into train and test. The data is being split as 90:10 ratio for train and test. We have to go to “choose” option and under the unsupervised option by selecting an instance there is a “resample option” where splitting of the data is possible with the GUI editor.



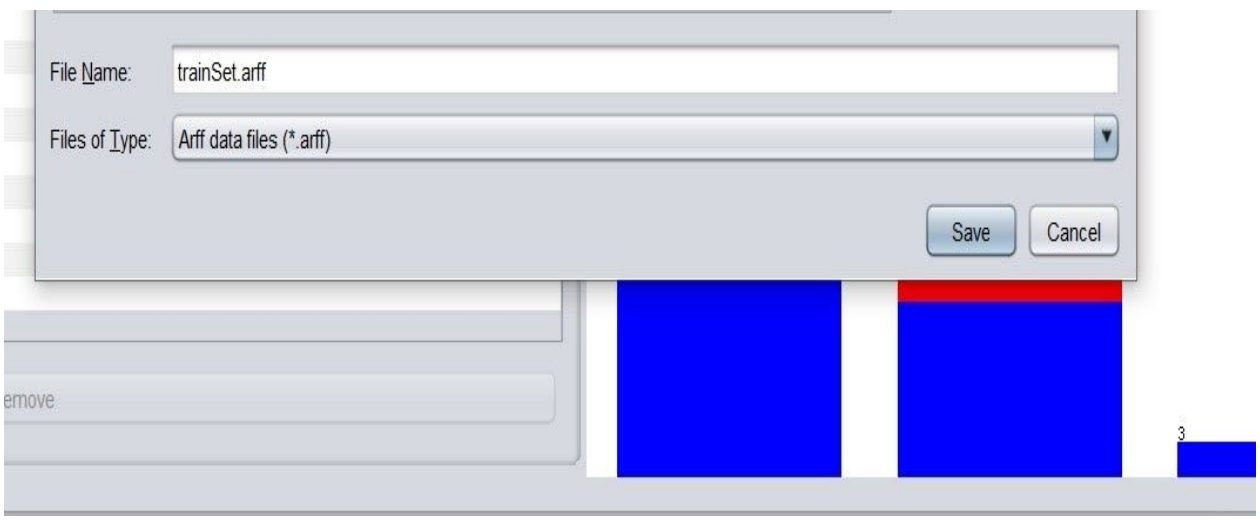
Then opens GUI editor which gives us an option to edit the percentage of the data split for train and test split. For the train dataset, 90% of data is being allocated.



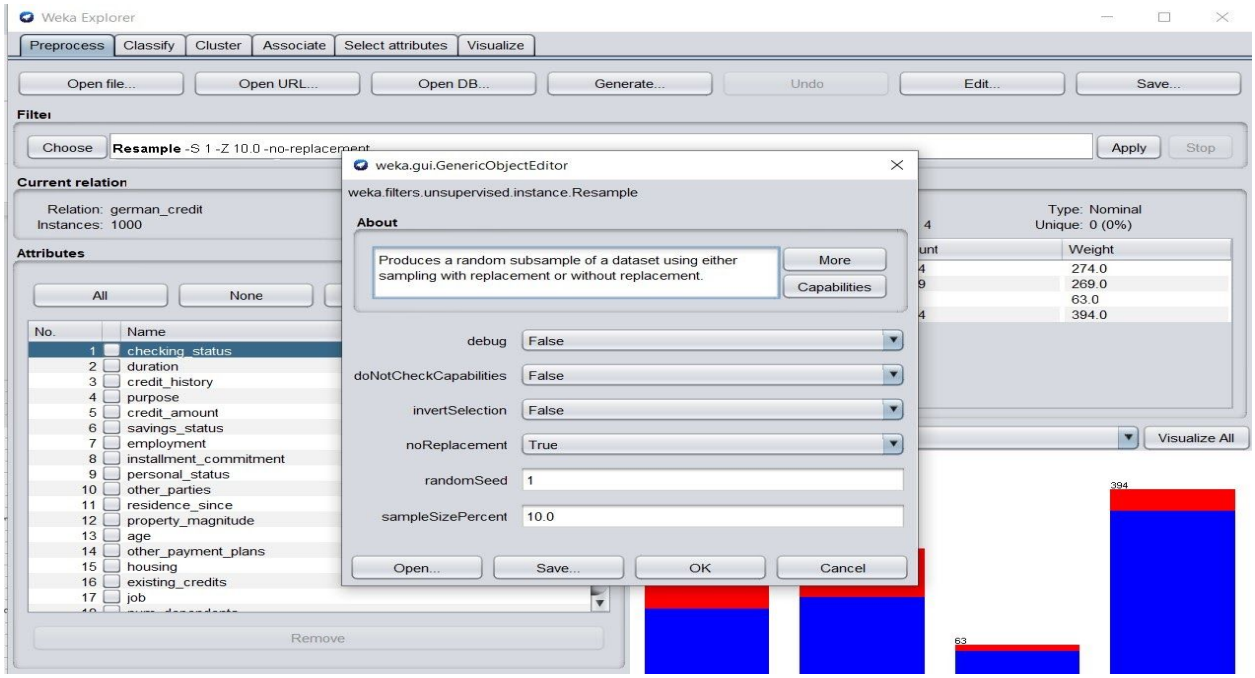
Therefore, it is observed that there are 900 instances in the **train dataset** with replacement being selected at GUI Editor. There are options in the editor where we can select the percentage of data for the train dataset.



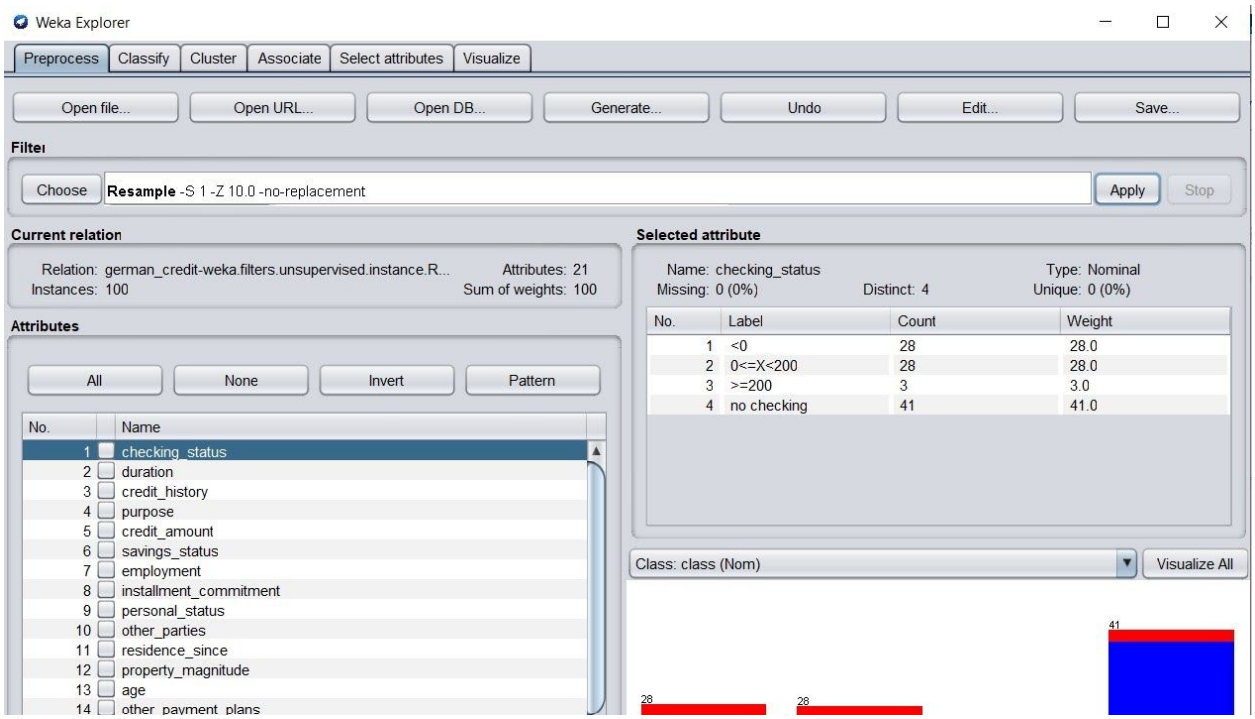
Upon selecting the percentage of the data, the dataset is being saved as **trainSet.arff**. The train set now contains 90% percent of the data which are 900 instances.



Now, preparing for the **test dataset.** The dataset.arff is being split into train and test whereas test split contains 10% of overall data. By selecting the resampling, in the editor by selecting no replacement as true test data is split by giving the percentage count with no replacement.



There are 10% variables assigned in the test data which is observed in the screenshot.



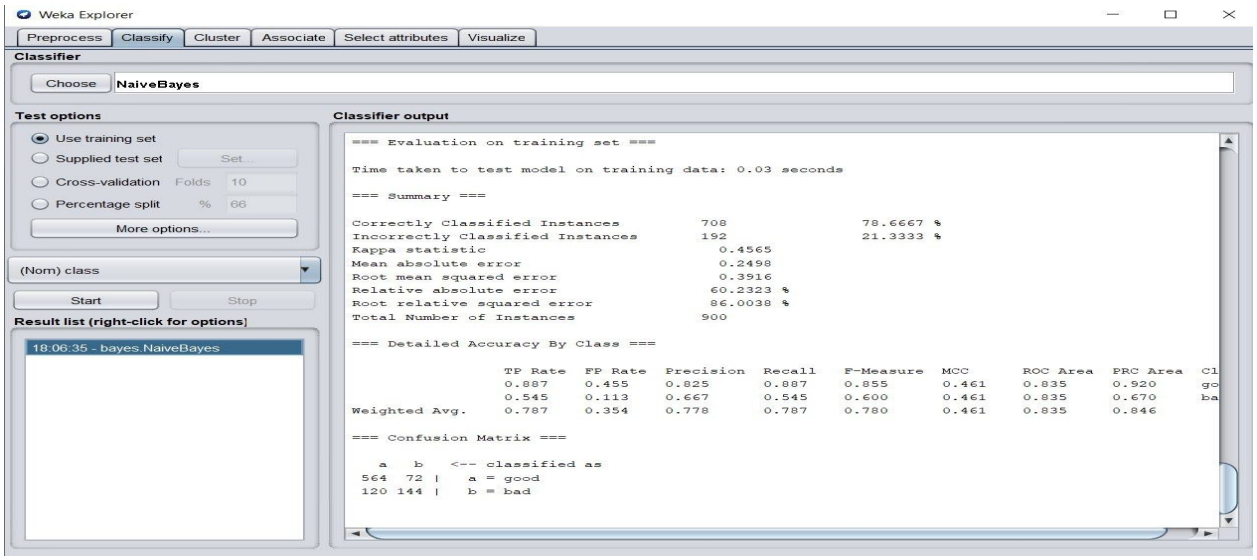
Therefore, the dataset is saved as testSet.arff to the destination.

DATA MINING TECHNIQUES

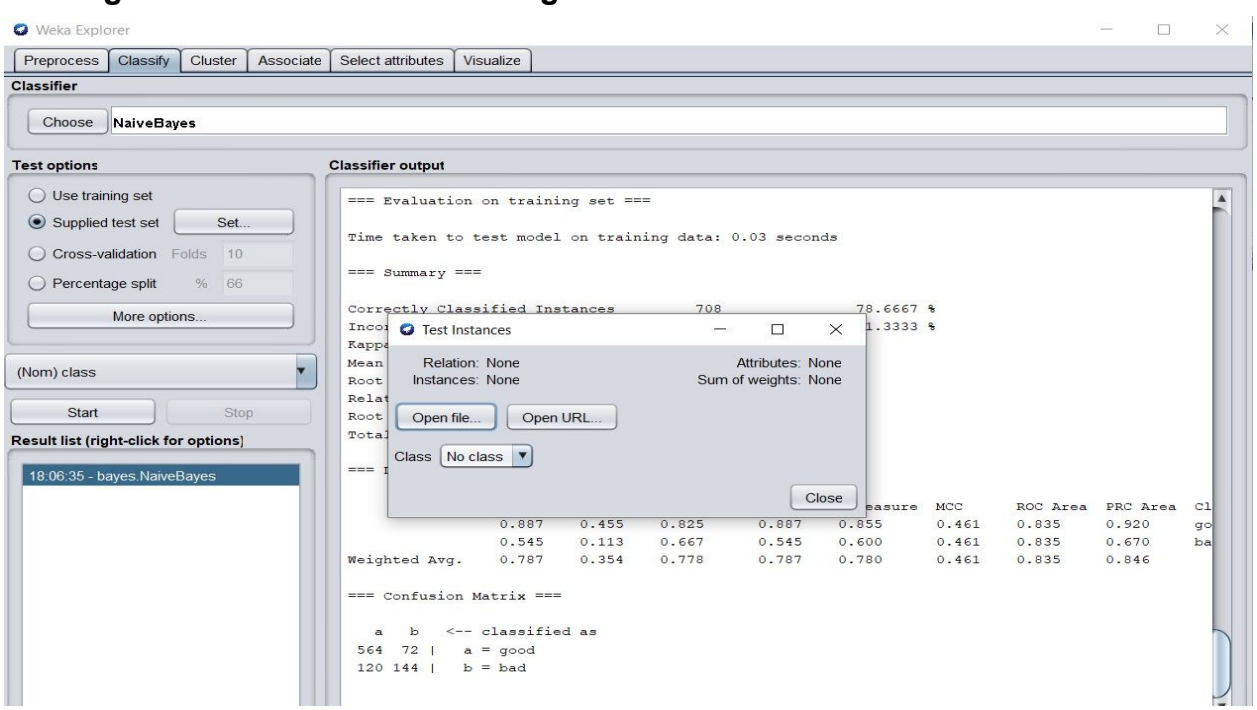
CLASSIFICATION

1. Training model using trainingSet.arff with different classification techniques in weka. There are different classification techniques in weka, here we can see different predicted results of different techniques.

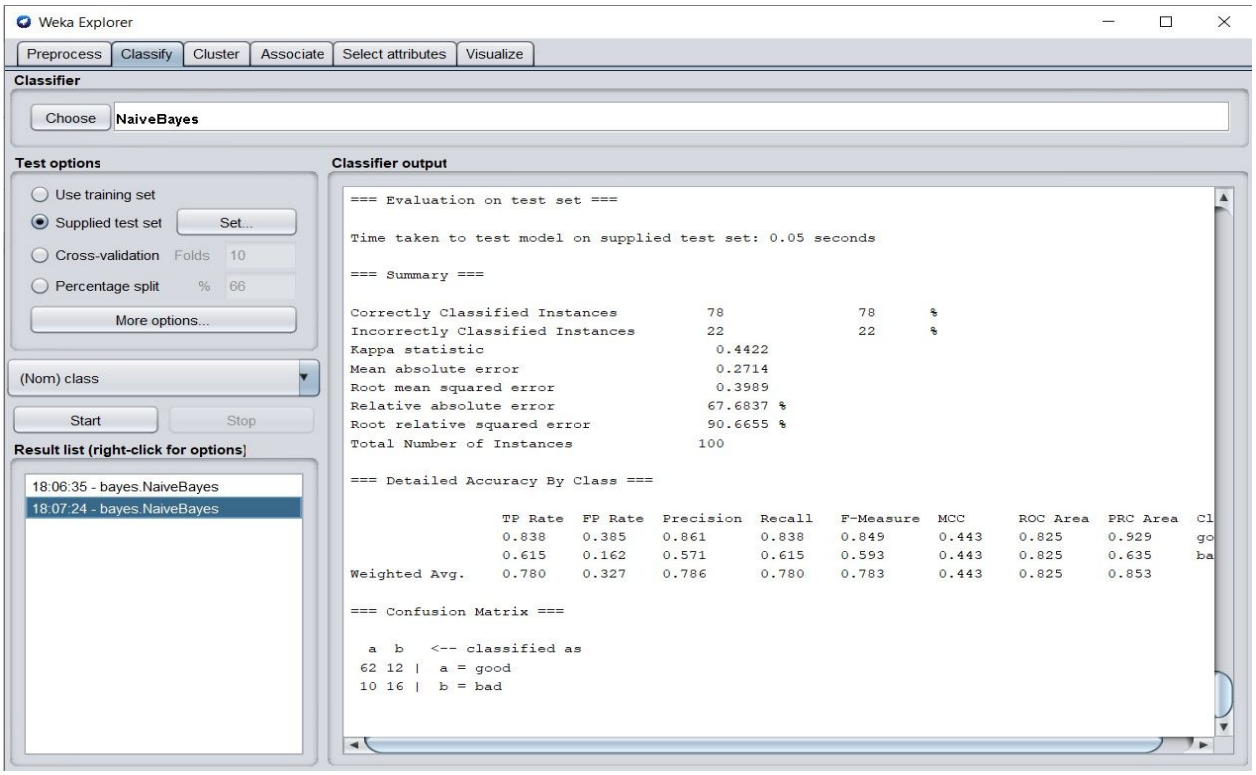
Naive bayes classifier technique.



Testing the trained model on testingSet.arff



Results of predicted model of naive bayes.



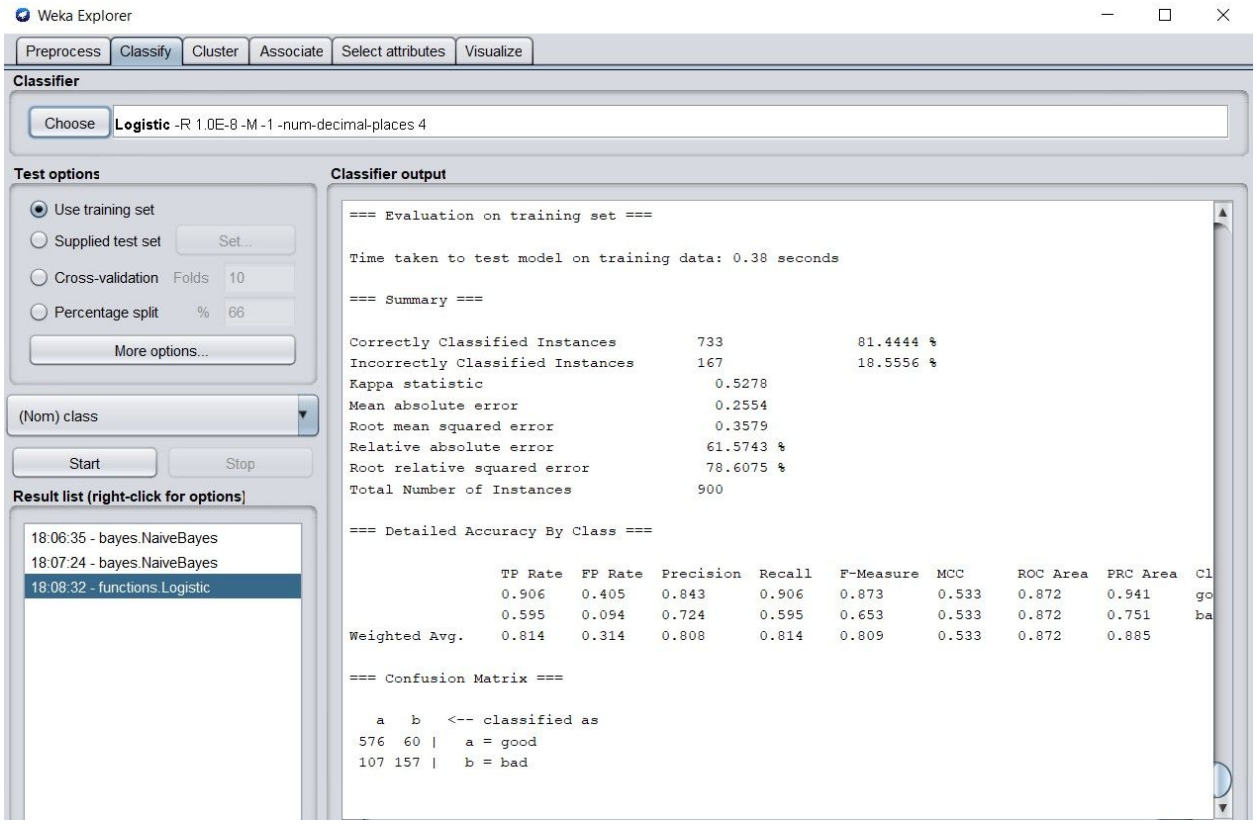
Analyzing/interpreting the results

There are 78 instances which are predicted correct and 22 were incorrectly predicted. And it is observed that MSE is 0.27, RMSE is 0.3989. Since there are 100 instances in total where 78 were predicted correctly.

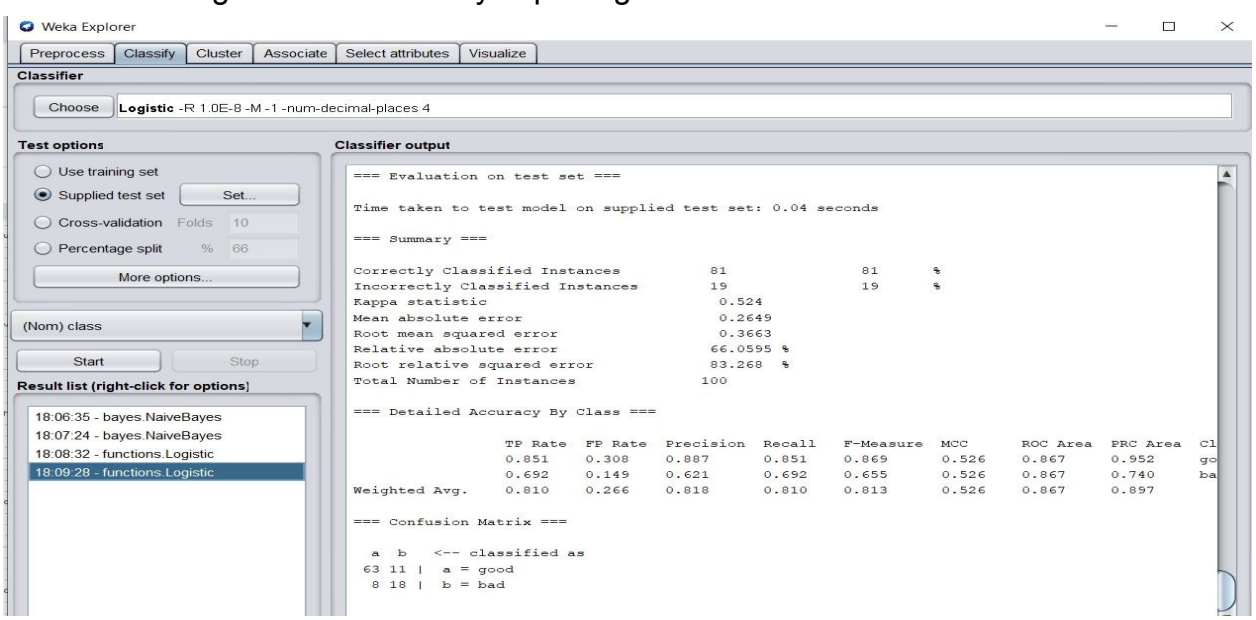
The total time taken to build a model on weka is just 0.05 seconds. Also weka generated results of Precision, recall, false positive rate and true positive rate.

1.2 Functions.logistic

1.2.1 Now, selecting functions.logistic technique and building a model on training dataset.

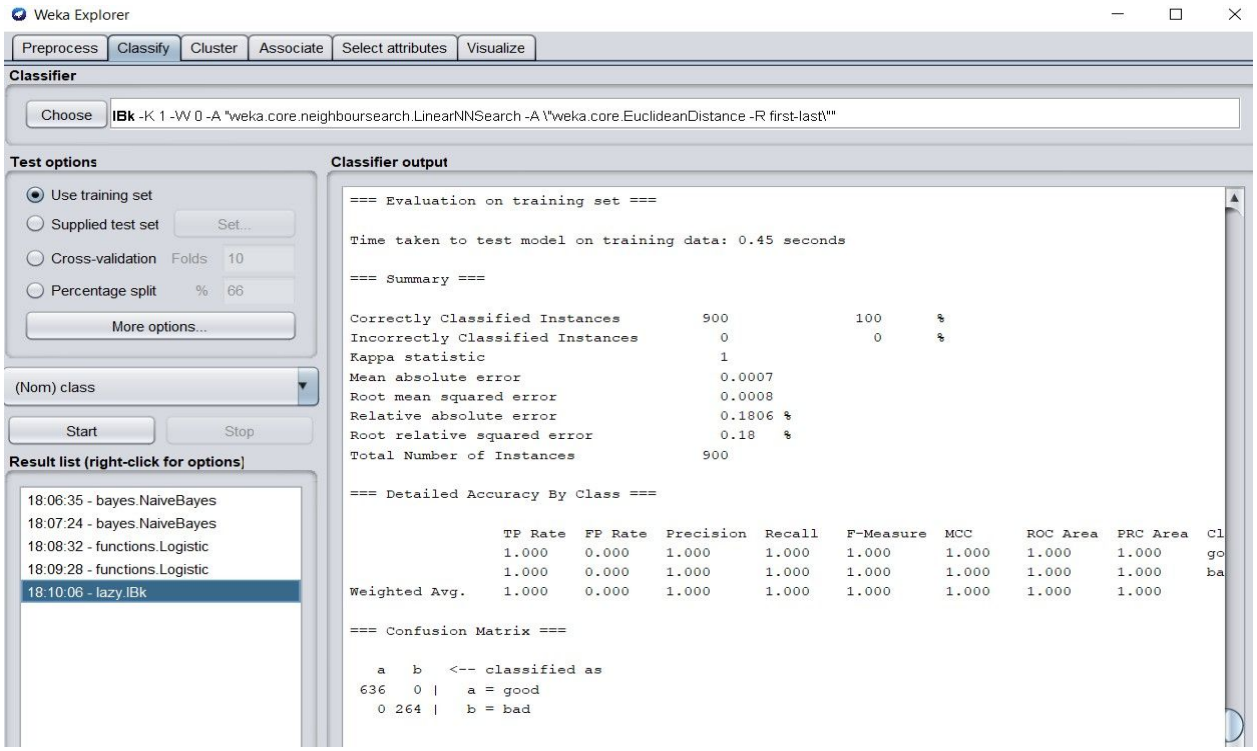


1.2.2 Predicting on test dataset by importing testSet.arff

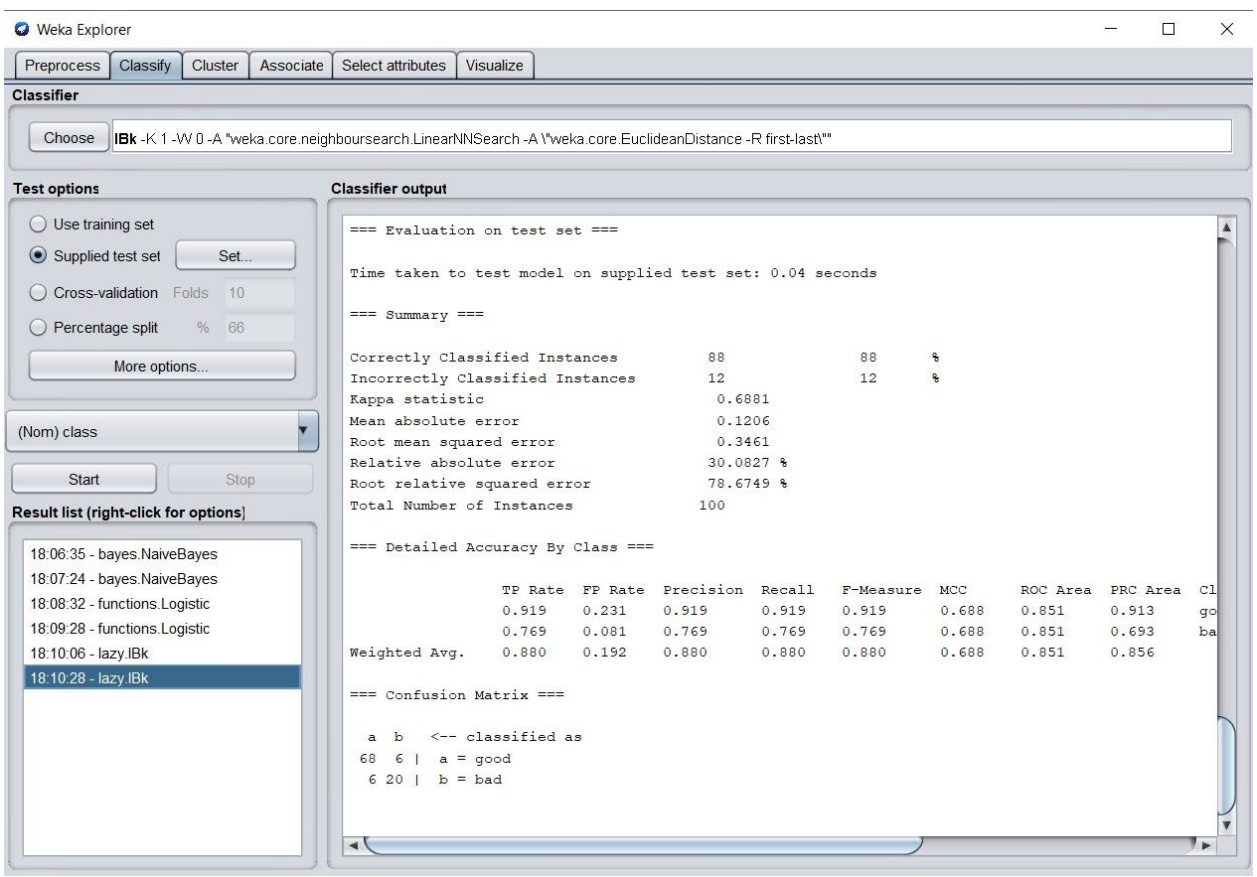


1.3 lazy.IBk

1.3.1 Now, using this technique model is built on a training set.

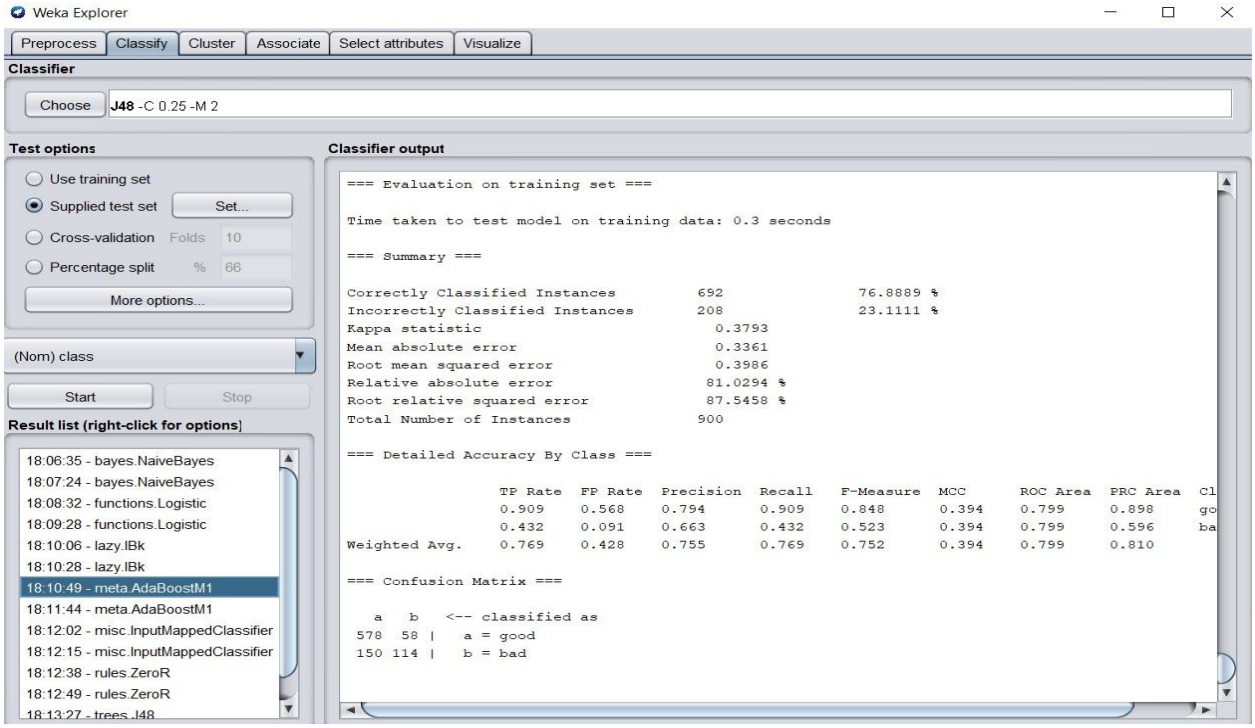


1.3.2 Predicting on test data by using testSet.arff.

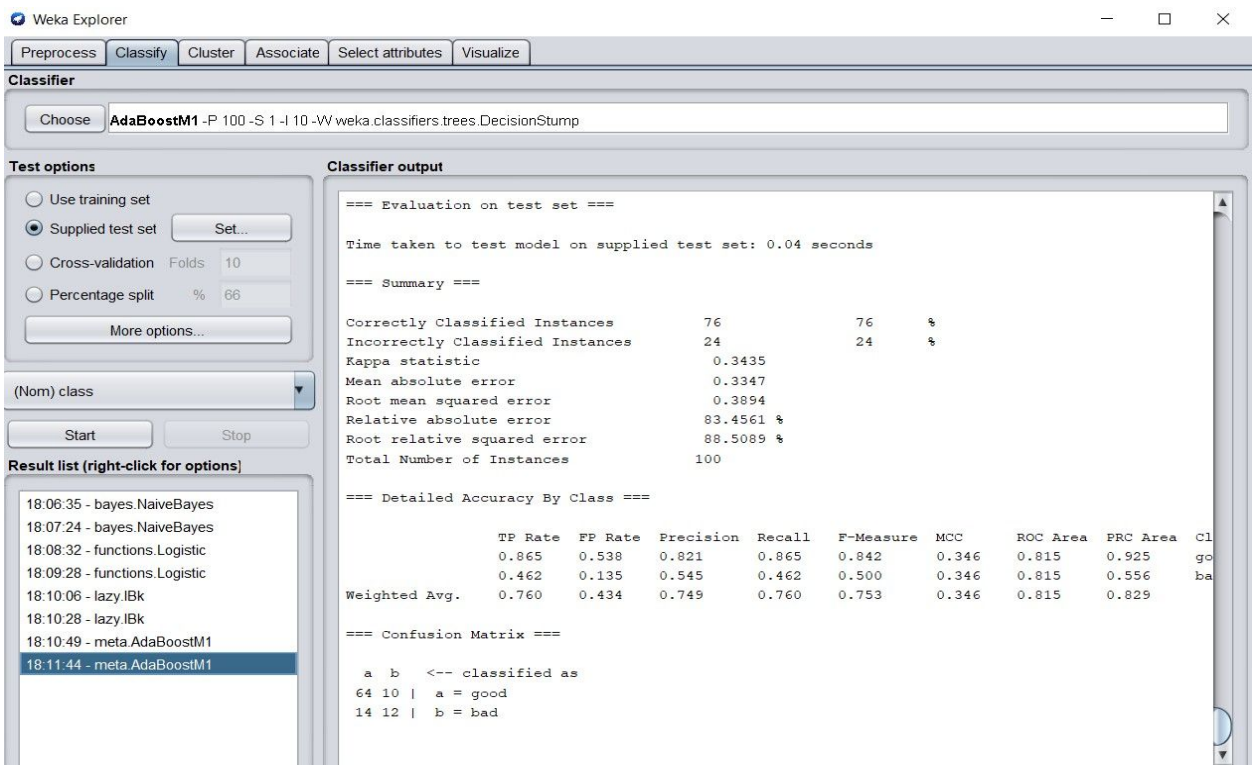


1.4 metaAdaBoostM1

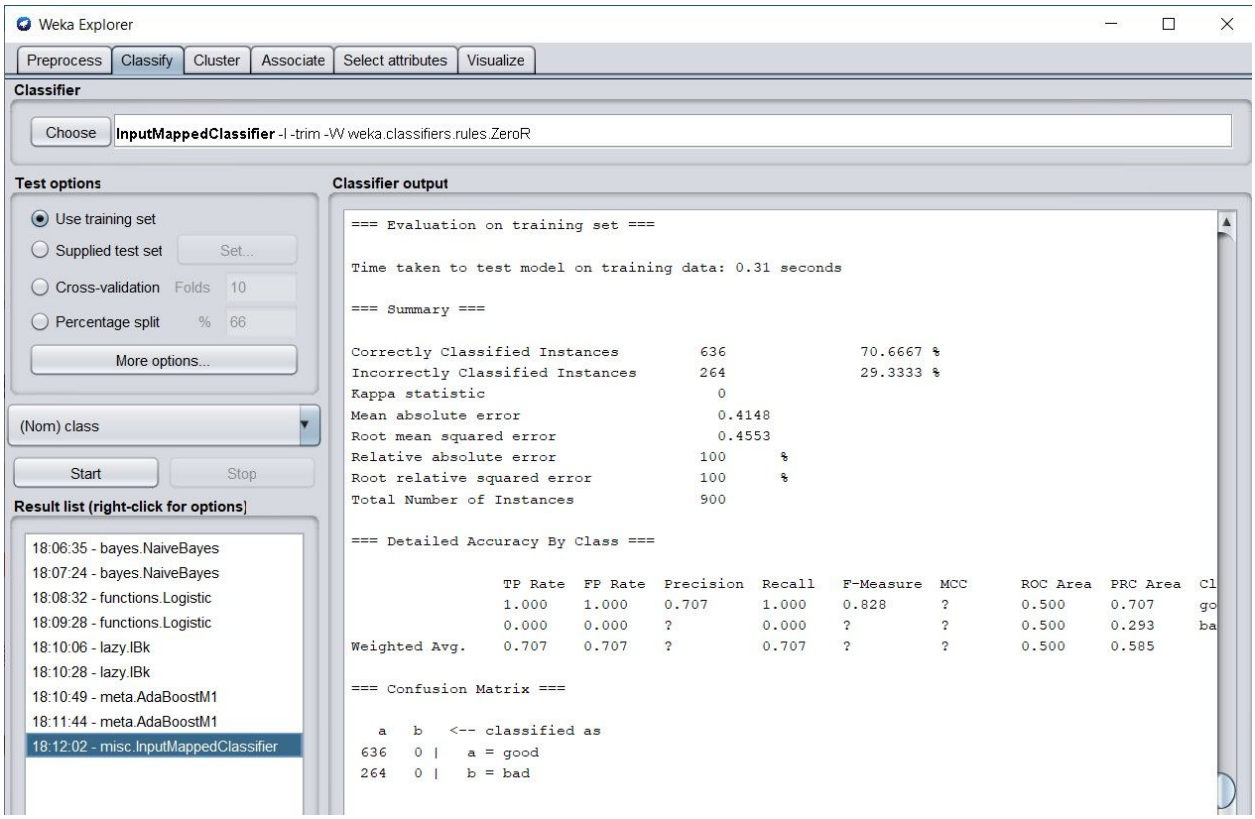
1.4.1 Model is built now using this technique on training dataset by using trainingSet.arff



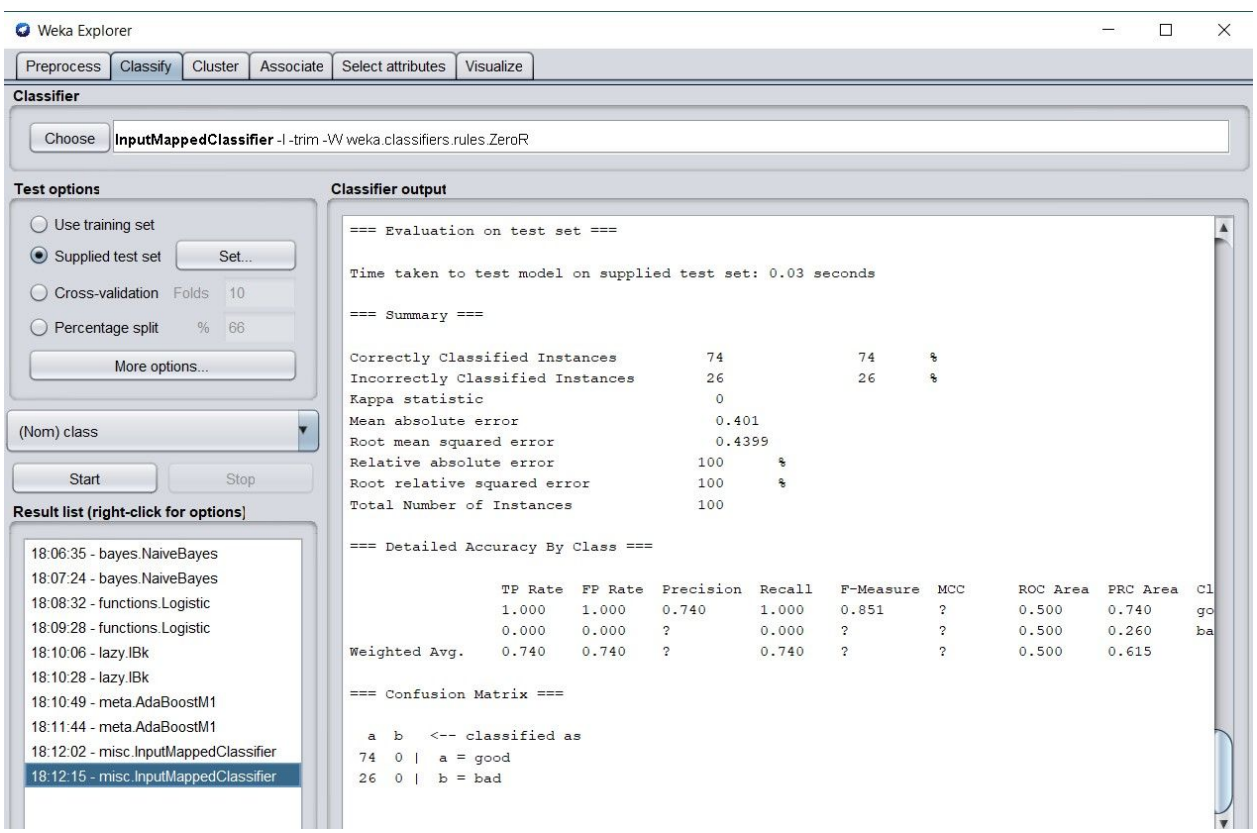
1.4.2 Predicting on test data by using testSet.arff



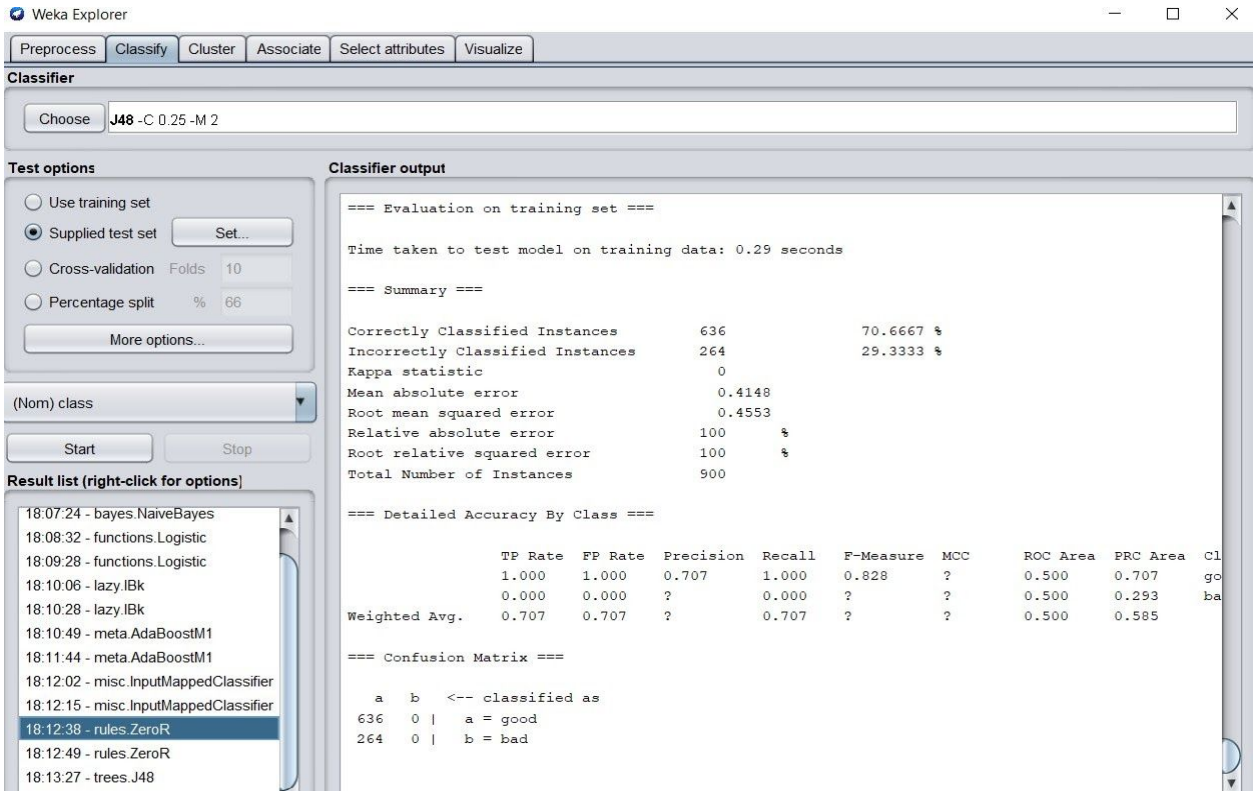
1.5.1 Building model using misc.inputMappedClassifier



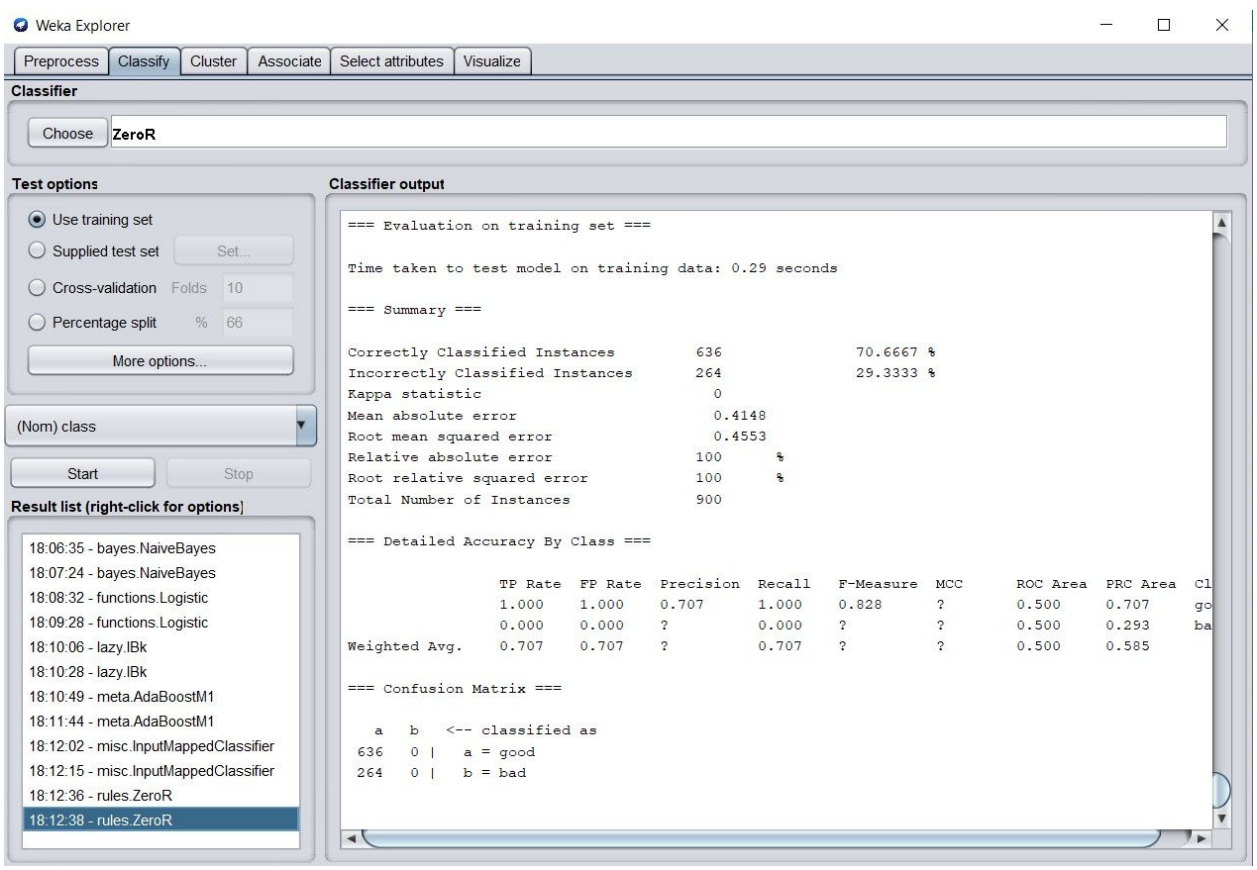
1.5.2 Predicting on test data by using testSet.arff



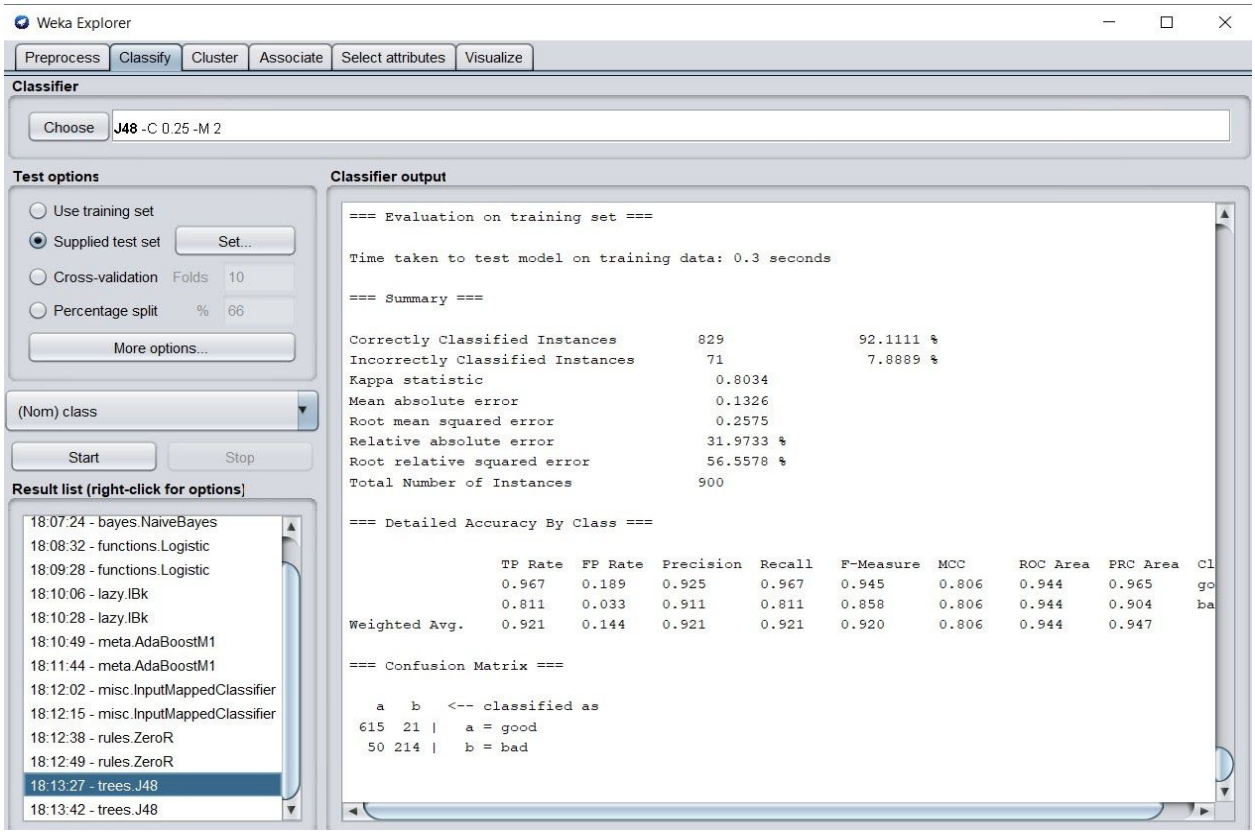
1.6.1 Building model using rules.ZeroR technique using trainSet.arff



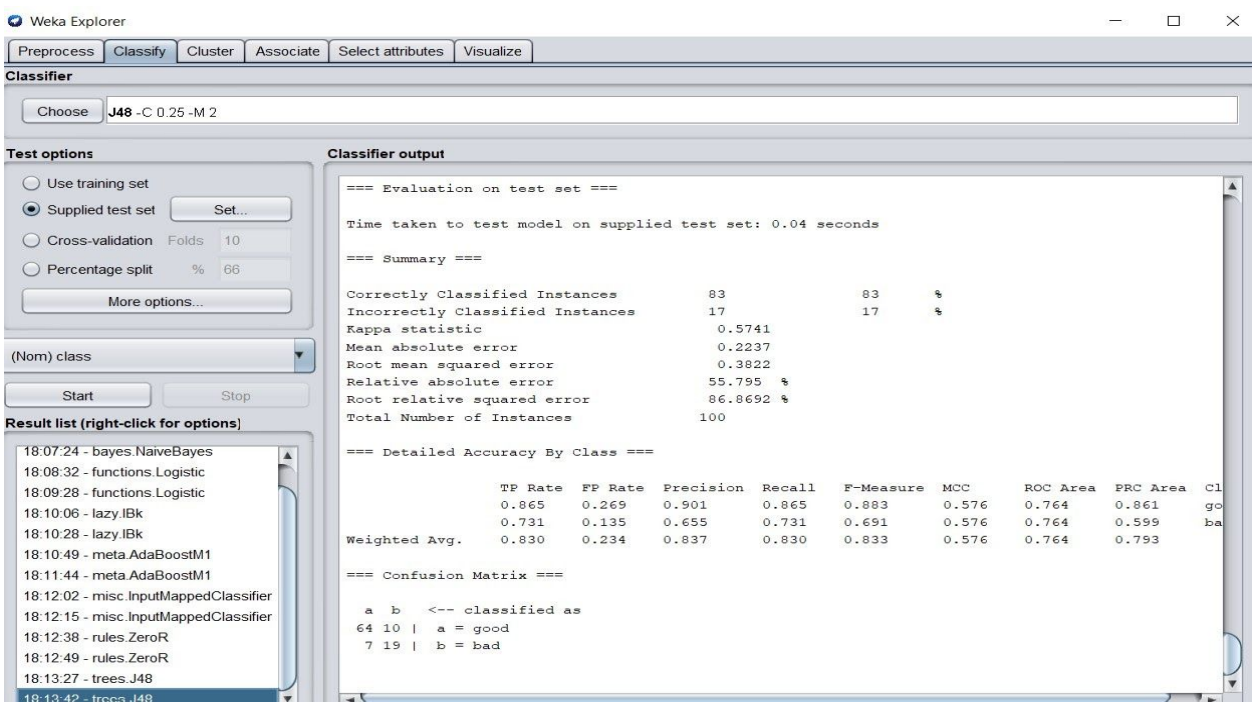
1.6.2 Testing the trained model on testSet.arff



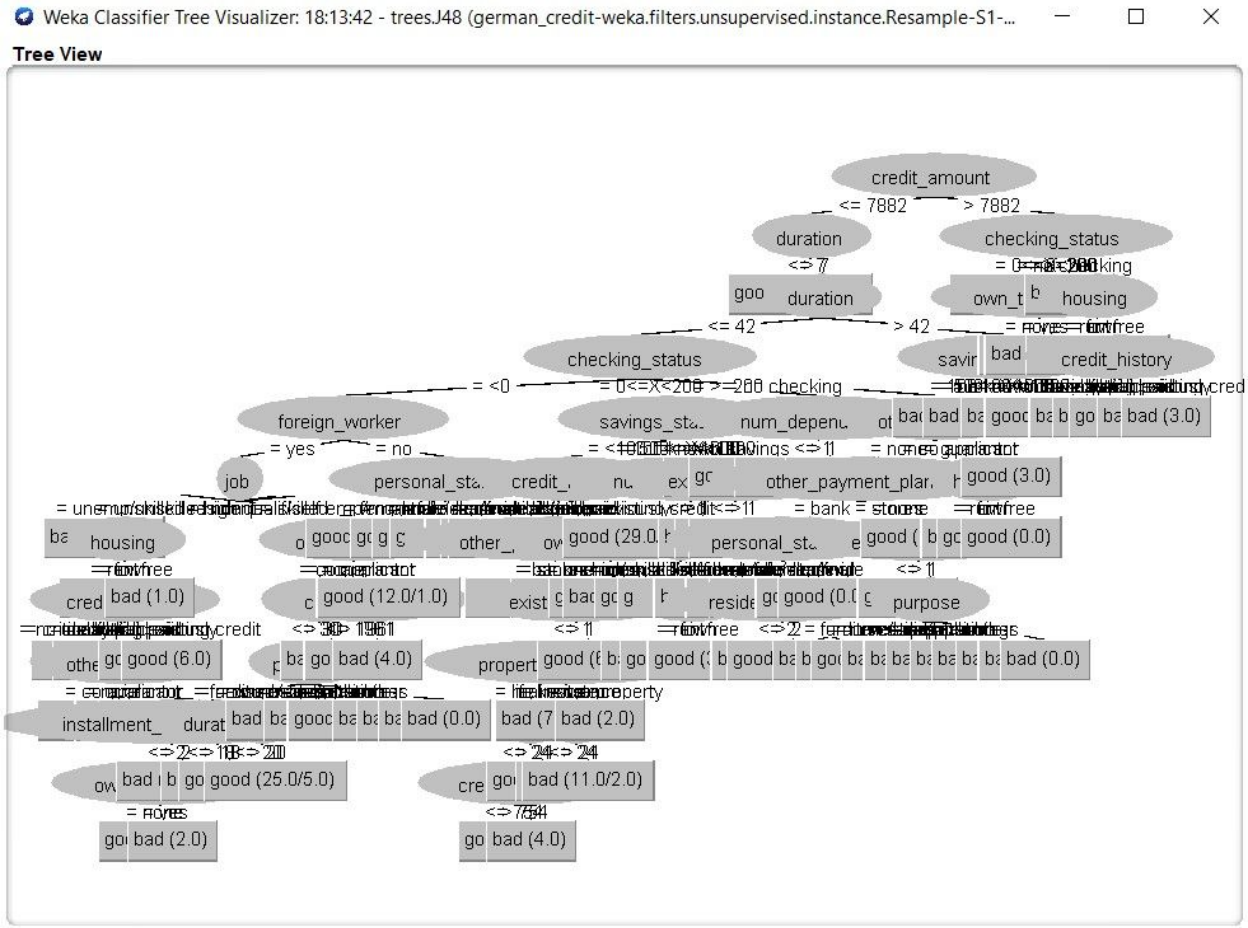
1.7.1 Building model using trees.J48 technique



1.7.2 Predicting the trained model on test data using testSet.arff



VISUALIZATION OF J48.Tree



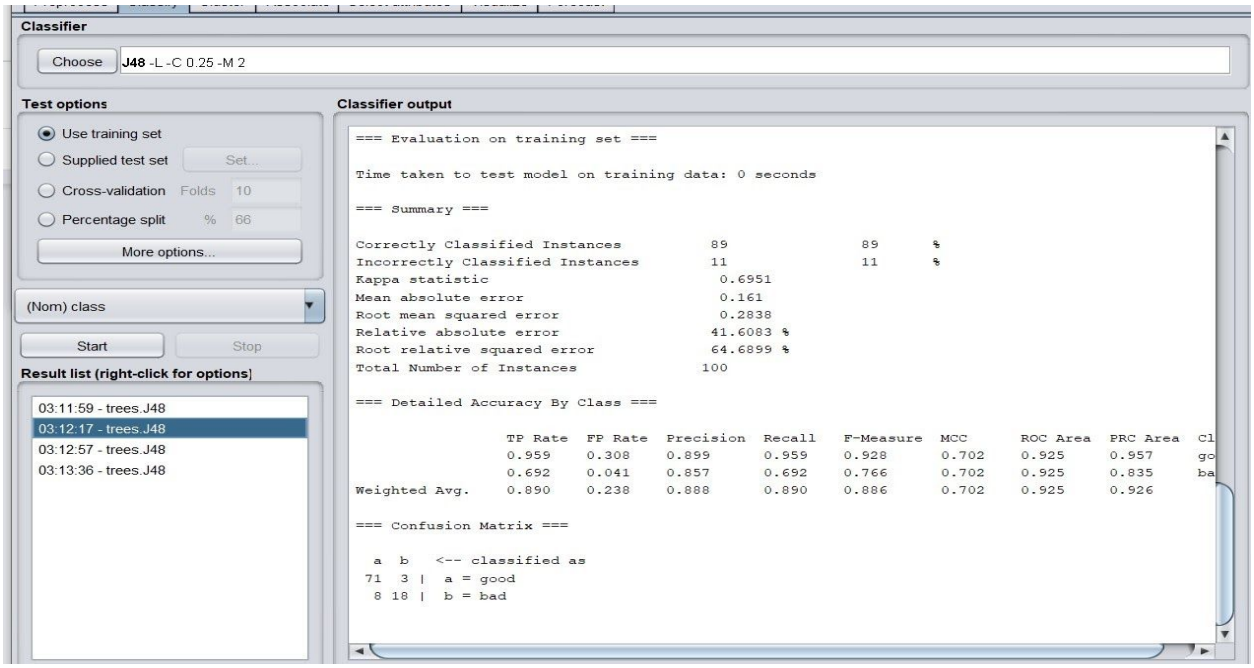
Interpreting the results of j48.tree model

Model is built on trainSet.arff and then predicted on the test dataset that is testSet.arff. In weka j48.tree is a classification technique which is used to classify the instances based on their ranking.

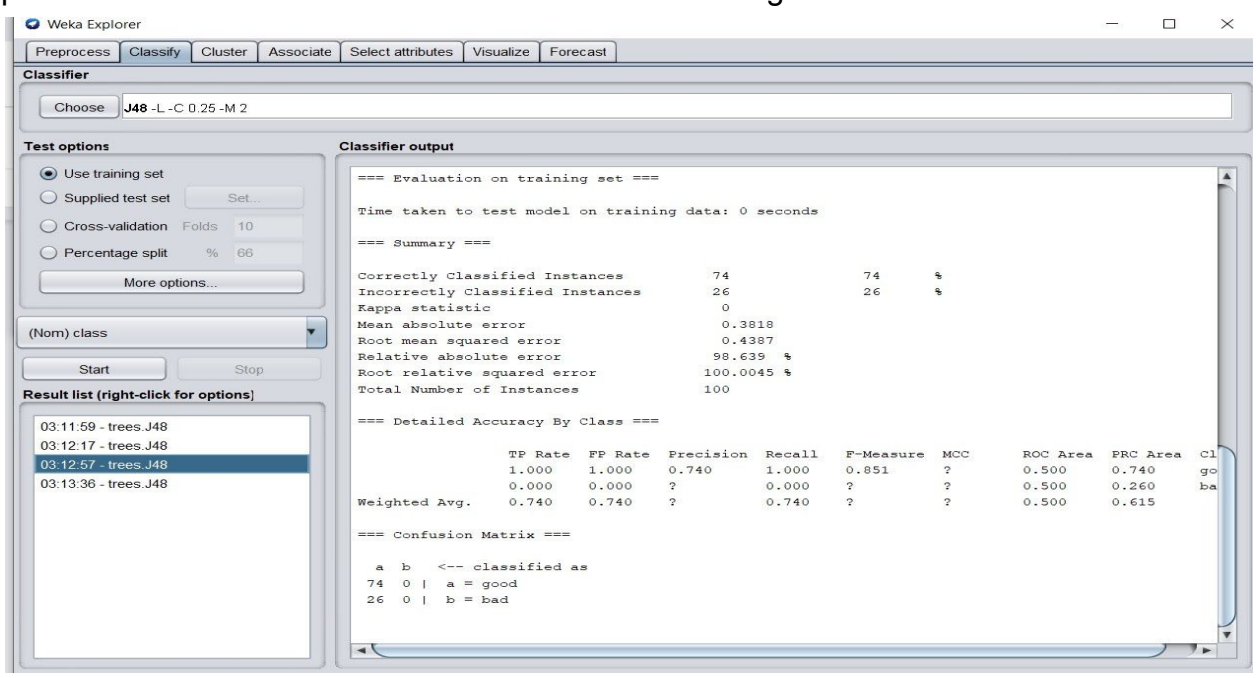
It is also possible to visualize the tree in weka which is presented above. There are 83 instances which are predicted correctly and 17 are incorrectly predicted. The RMSE of the model is 0.3822, MSE is 0.2237. Also precession, recall, true positive rate and false positive rate.

Varying parameters on j48.tree classification technique

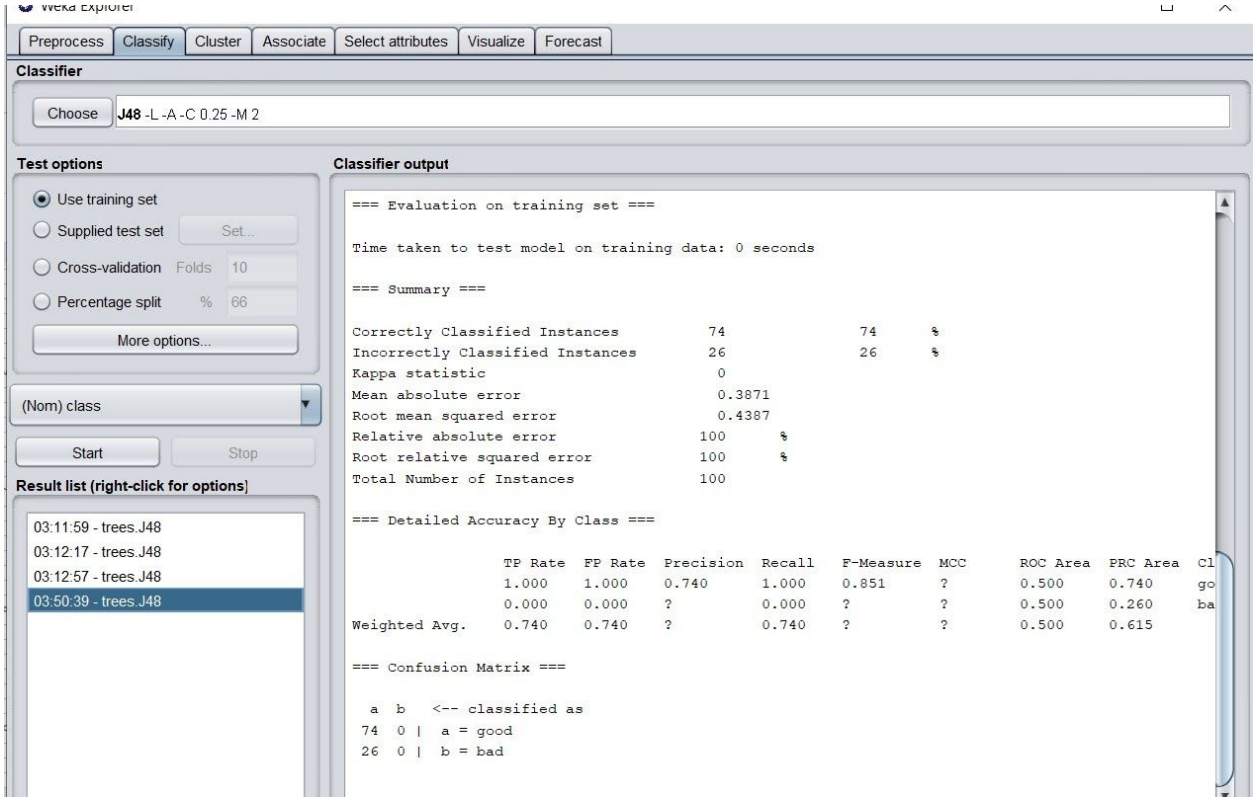
Varying parameters model generates different results. Here the correctly predicted and incorrectly predicted changed on changing parameters compared to previous model results.



Here, correctly and incorrectly predicted instances have changed on changing parameters. As well as RMSE and MSE are also changed.

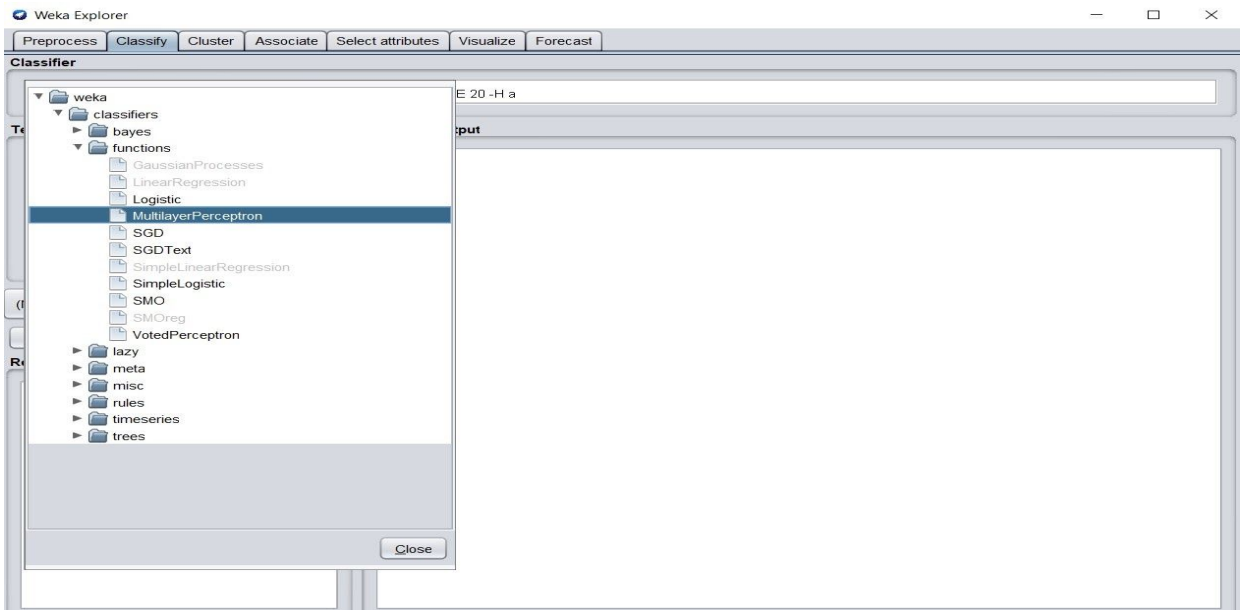


Here, we can observe that RMSE and MSE have changed on changing parameters.

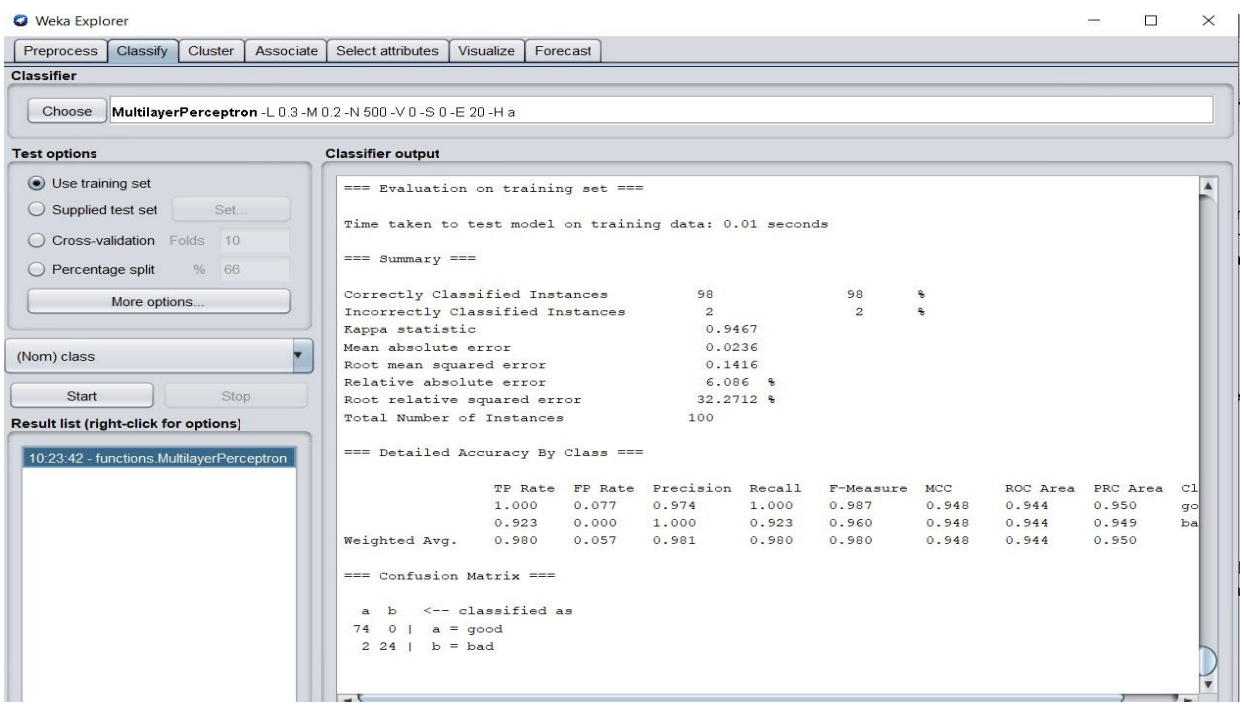


MULTILAYER PERCEPTRON(MLP)

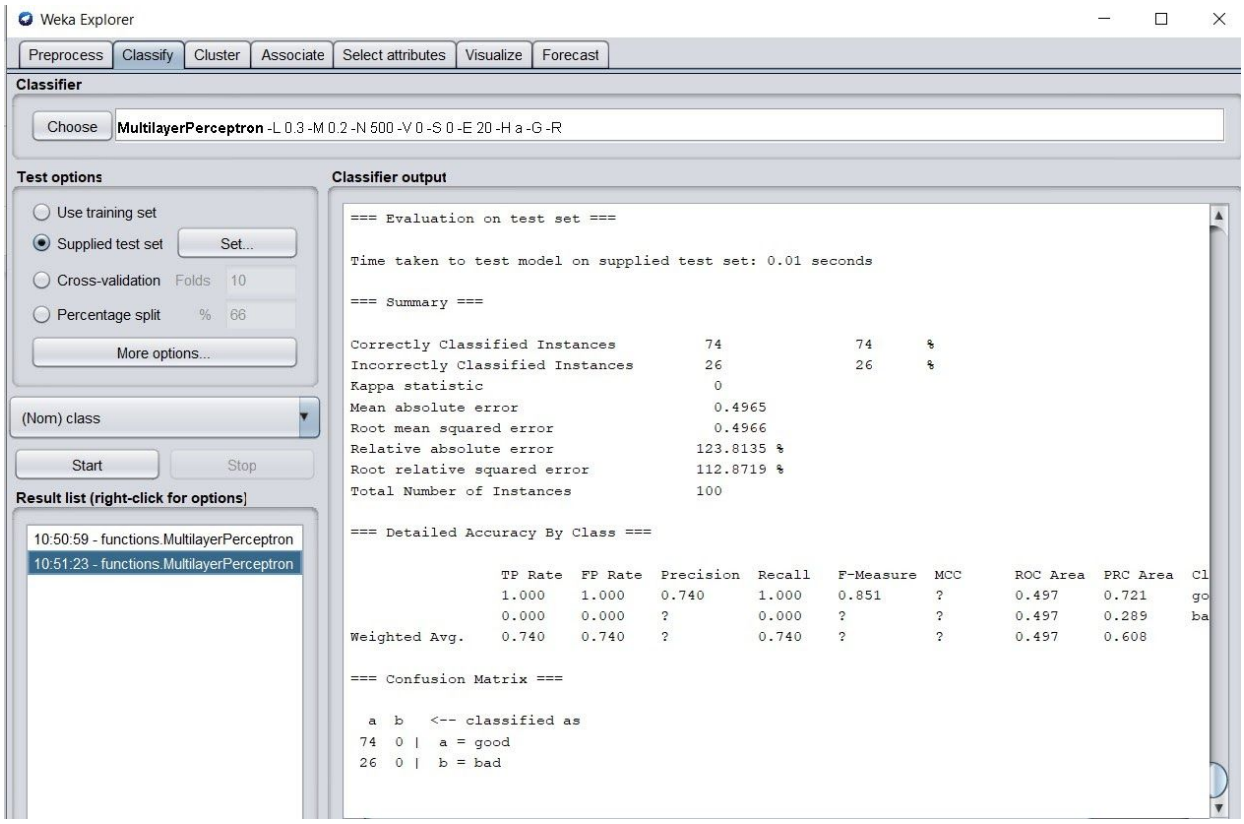
Multilayer perceptrons are networks of perceptrons, networks of linear classifiers. In fact, they can implement arbitrary decision boundaries using “hidden layers”. Weka has a graphical interface that lets you create your own network structure with as many perceptrons and connections as you like.



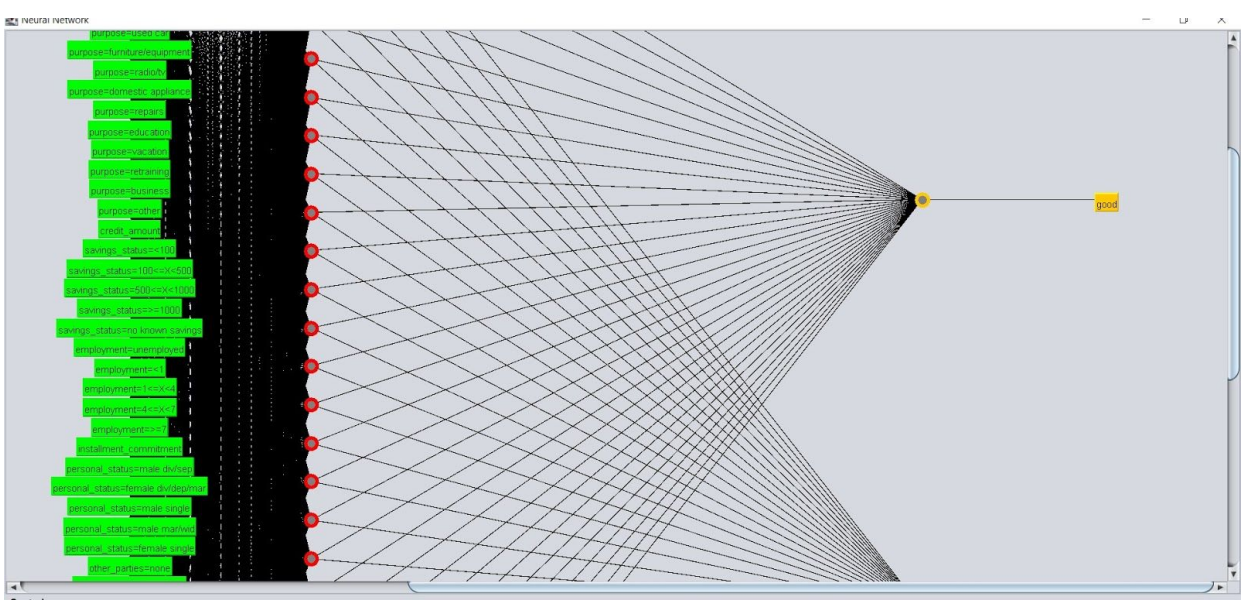
Model is trained using the MLP algorithm on a training data set.



Predicting on the test data set using testSet.arff



Visualizing the neural network



REFERENCES

https://www.tutorialspoint.com/weka/weka_preprocessing_data.htm

https://www.youtube.com/watch?v=FMiCOx95lAc&t=215s

https://machinelearningmastery.com/use-classification-machine-learning-algorithms-weka/

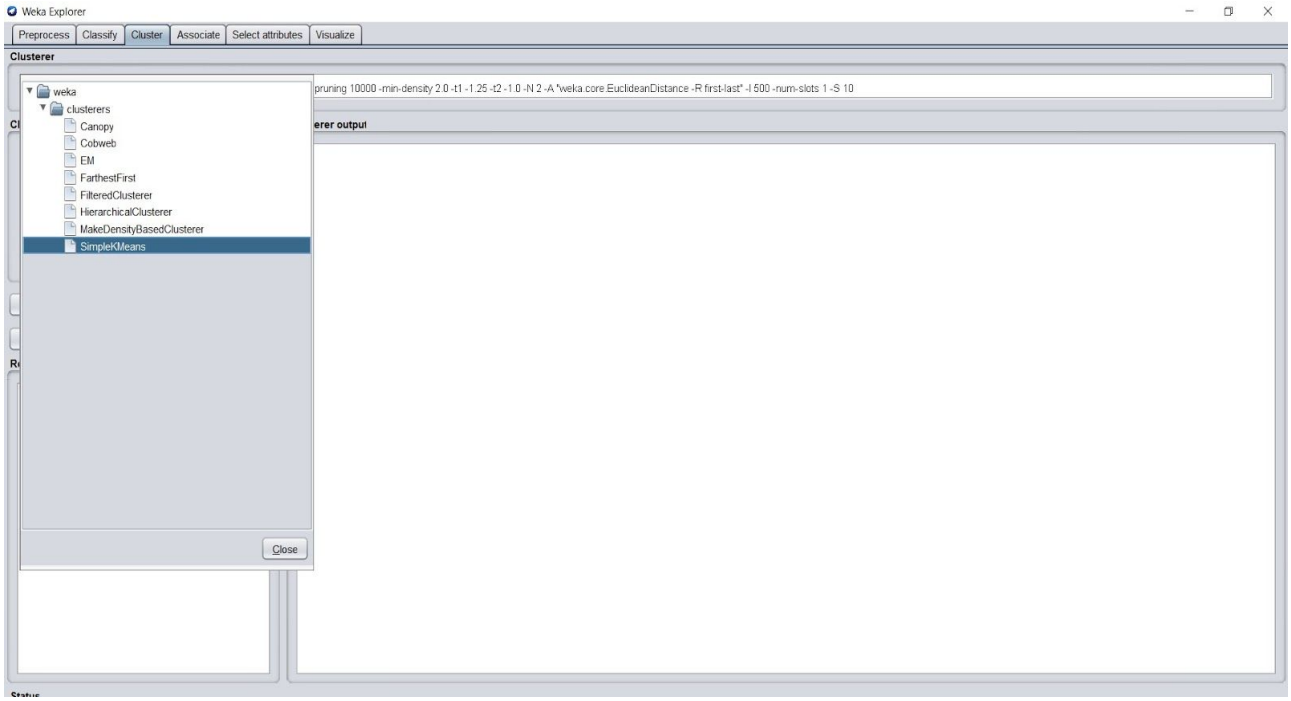
https://docs.rapidminer.com/latest/studio/operators/data_access/files/write/write_arff.html

https://waikato.github.io/weka-wiki/formats_and_processing/arff_stable/

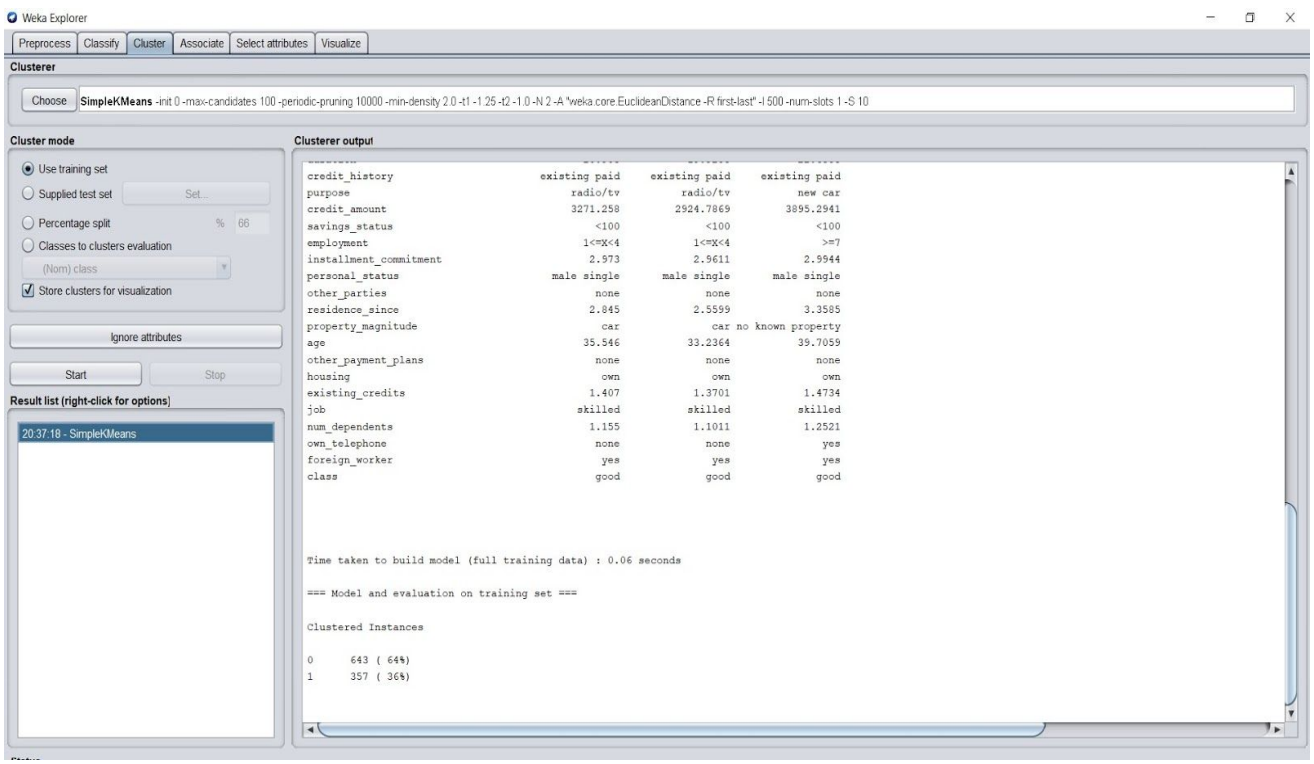
http://statweb.stanford.edu/~lpekelis/13_datafest_cart/WekaManual-3-7-8.pdf

CLUSTERING

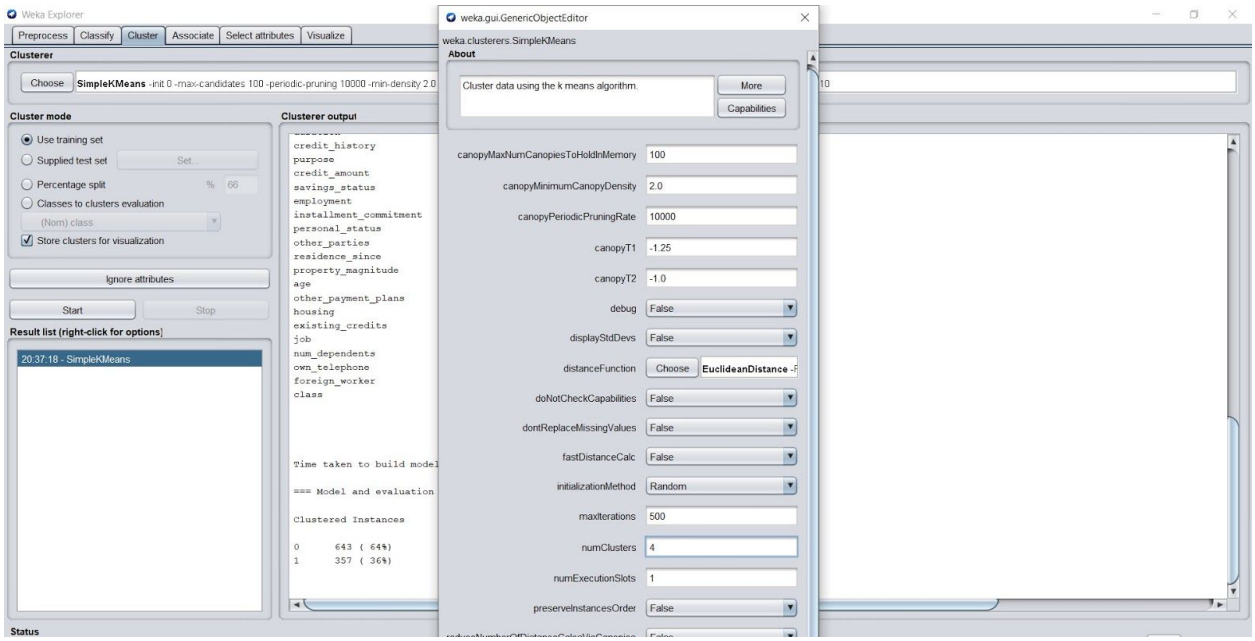
In weka, there is a package called cluster which can be imported which is used to perform clustering on the data. Here, a simple k-means clustering algorithm is used to build a model.



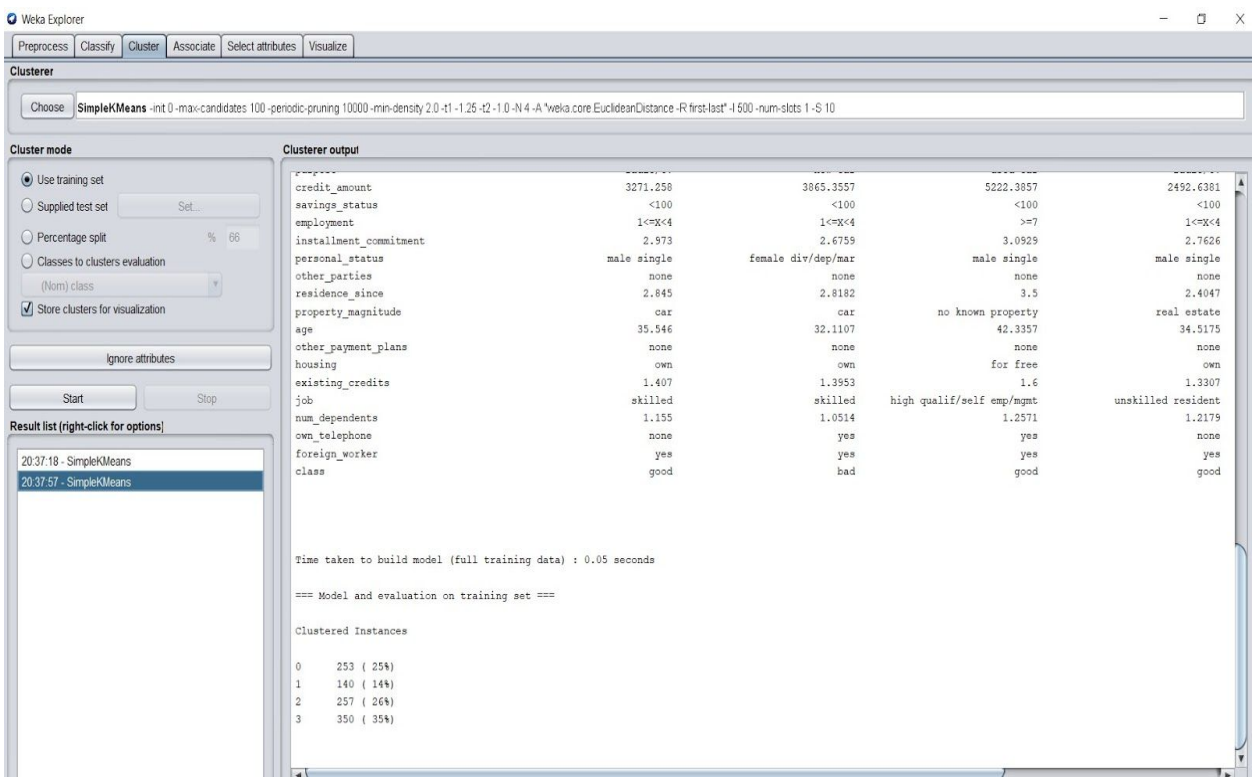
Simple K-means is selected and the following results have been generated below. 64 % of the data fall into the first cluster and 36% is under the second cluster.



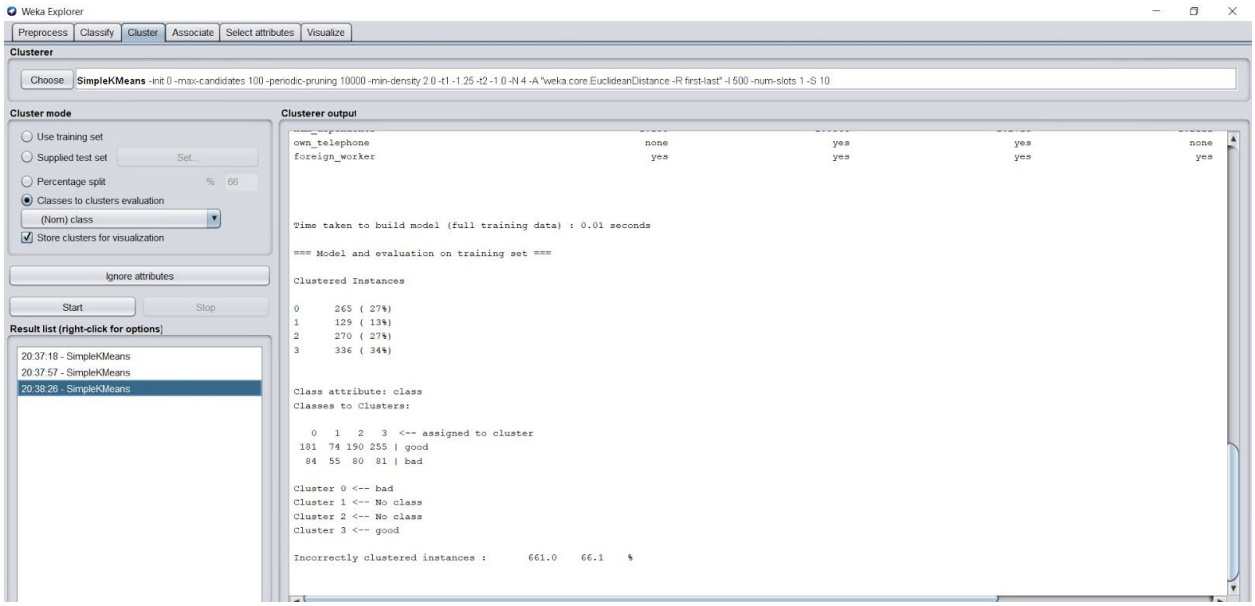
Here, parameters are changed. Where the number of clusters are increased and produced the results. The cluster number is changed to 4 in the following screenshot.



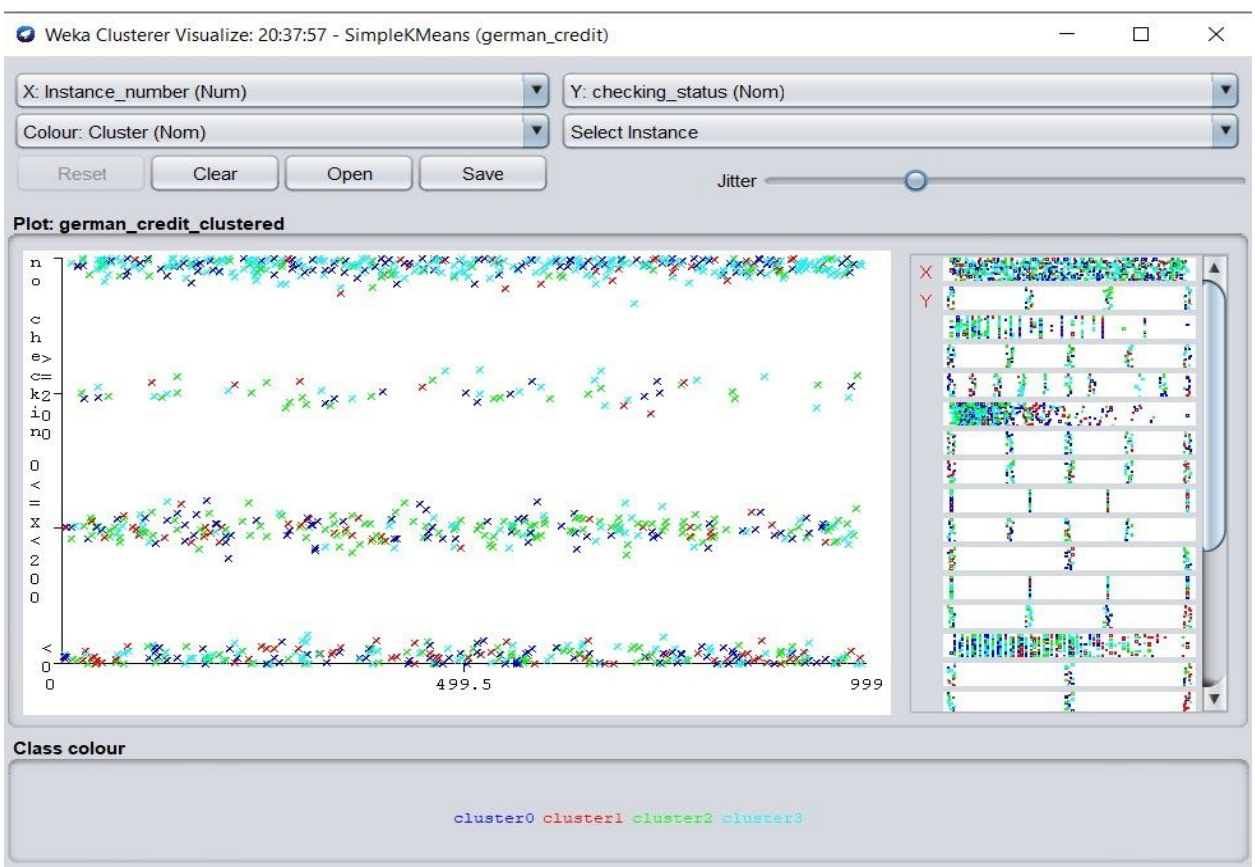
Here we generated 4 clusters. The 25% of the data is under cluster 1 where as cluster 2 contains 14% of the data. Cluster 3 contains 26% of the data and cluster 4 contains 35% which is highest.



Here we changed cluster mode to classes to cluster evaluation. The following results have been generated. It says cluster 0 is bad and cluster 3 is good. Whereas cluster 1 and cluster 2 had no class.



This is the visualization of the cluster which is generated for the german credit card data. Clusters are visible upon adjusting the jitter.



REFERENCES

http://facweb.cs.depaul.edu/mobasher/classes/ect584/WEKA/k-means.html

https://www.tutorialspoint.com/weka/weka_clustering.htm

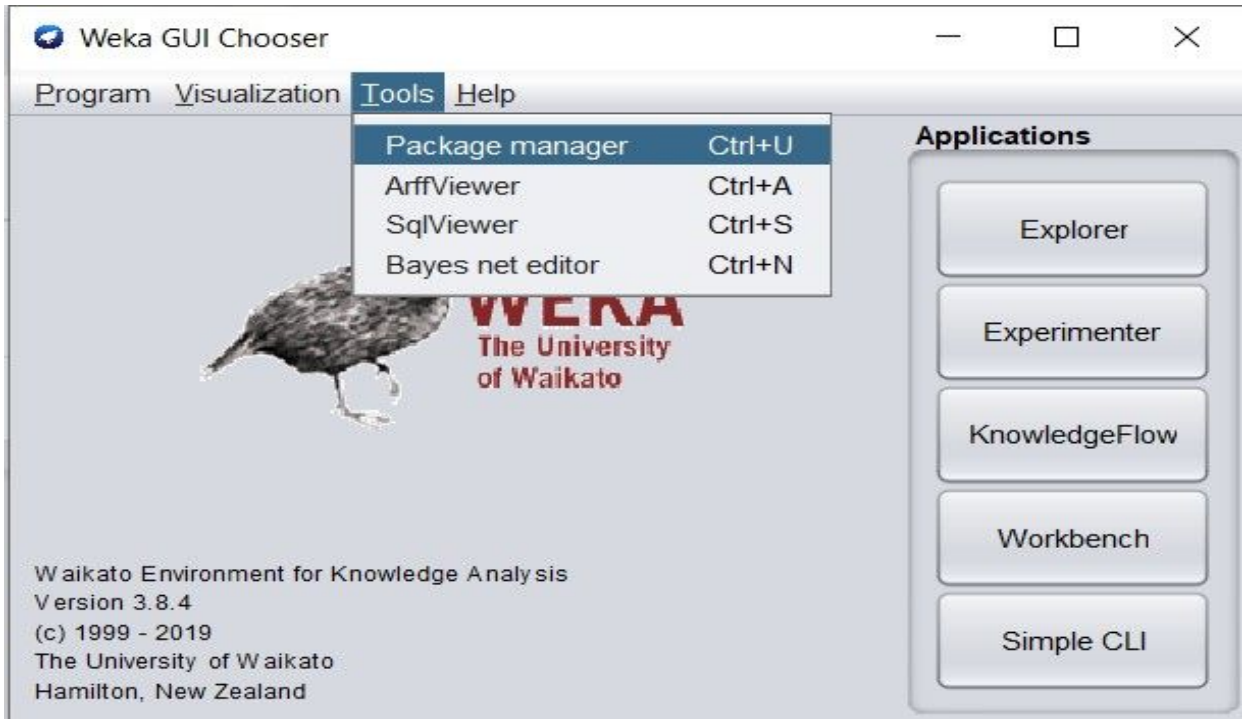
http://modelai.gettysburg.edu/2016/kmeans/assets/iris/Clustering_Iris_Data_with_Weka.pdf

https://developer.ibm.com/technologies/analytics/articles/os-weka2/

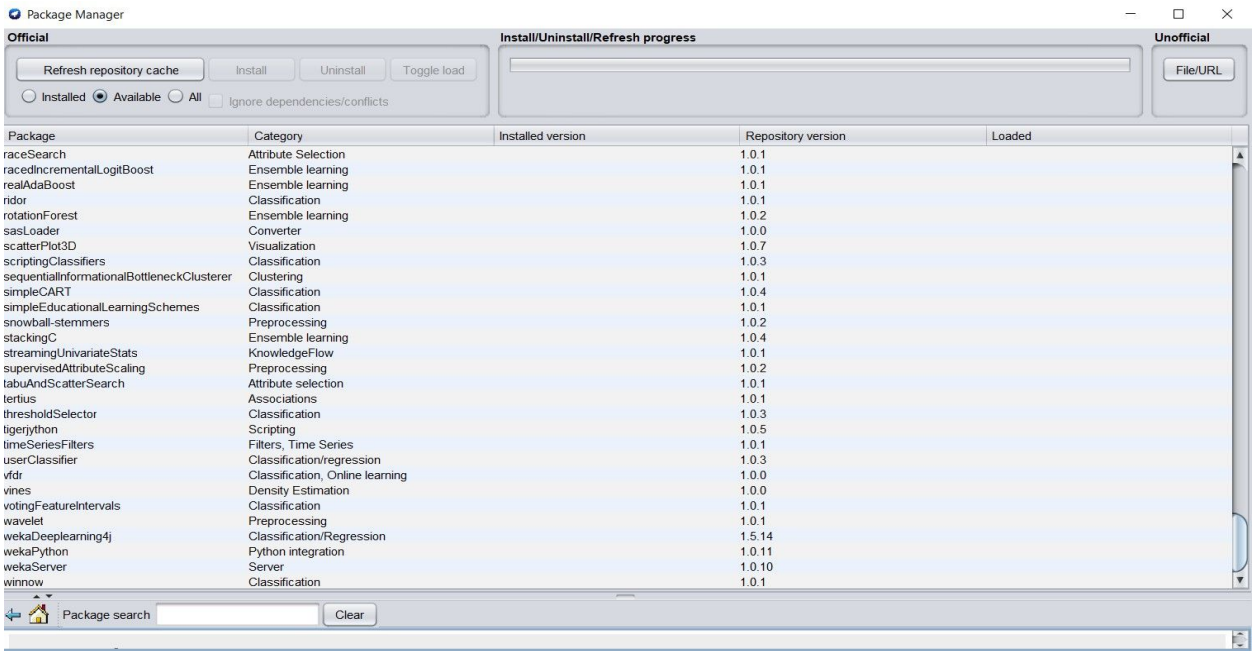
TIME SERIES FORECASTING

Time series forecasting package is installed through the package manager which is required to install.

Installing the forecasting package from the package manager.

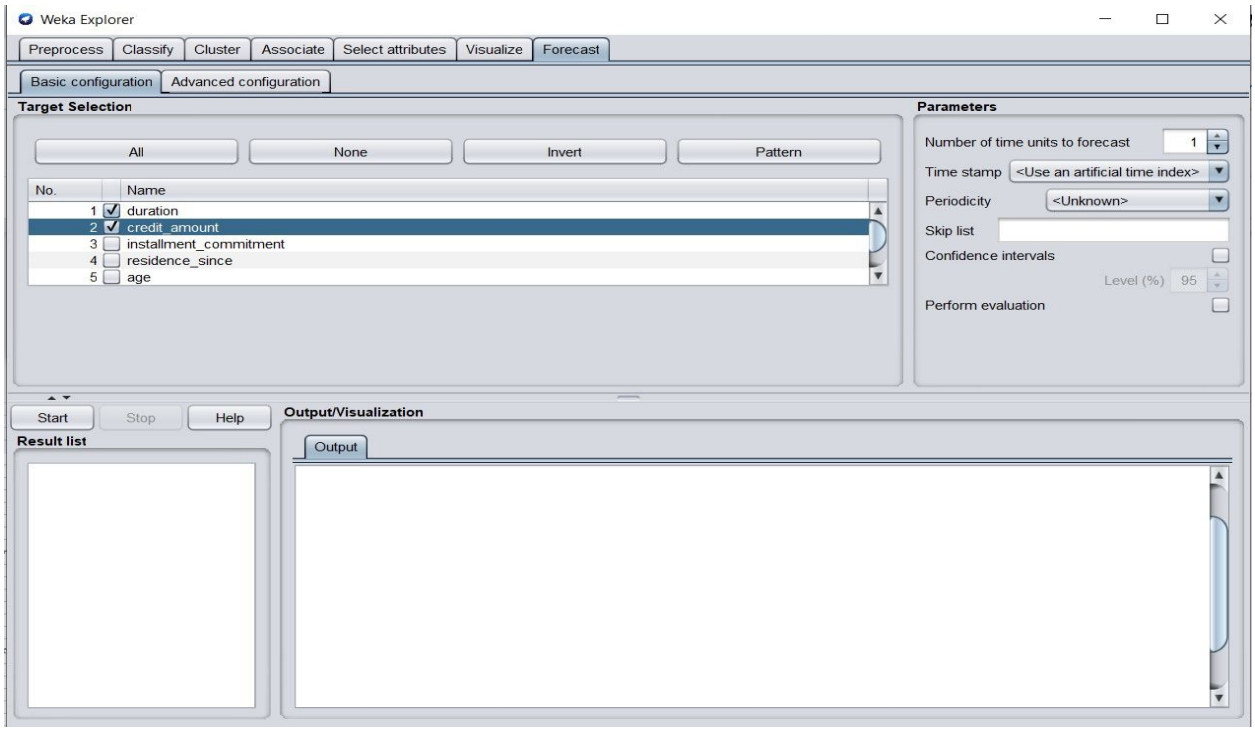


After Opening the package manager, a time series forecasting package is installed from here which can be seen in the screenshot.

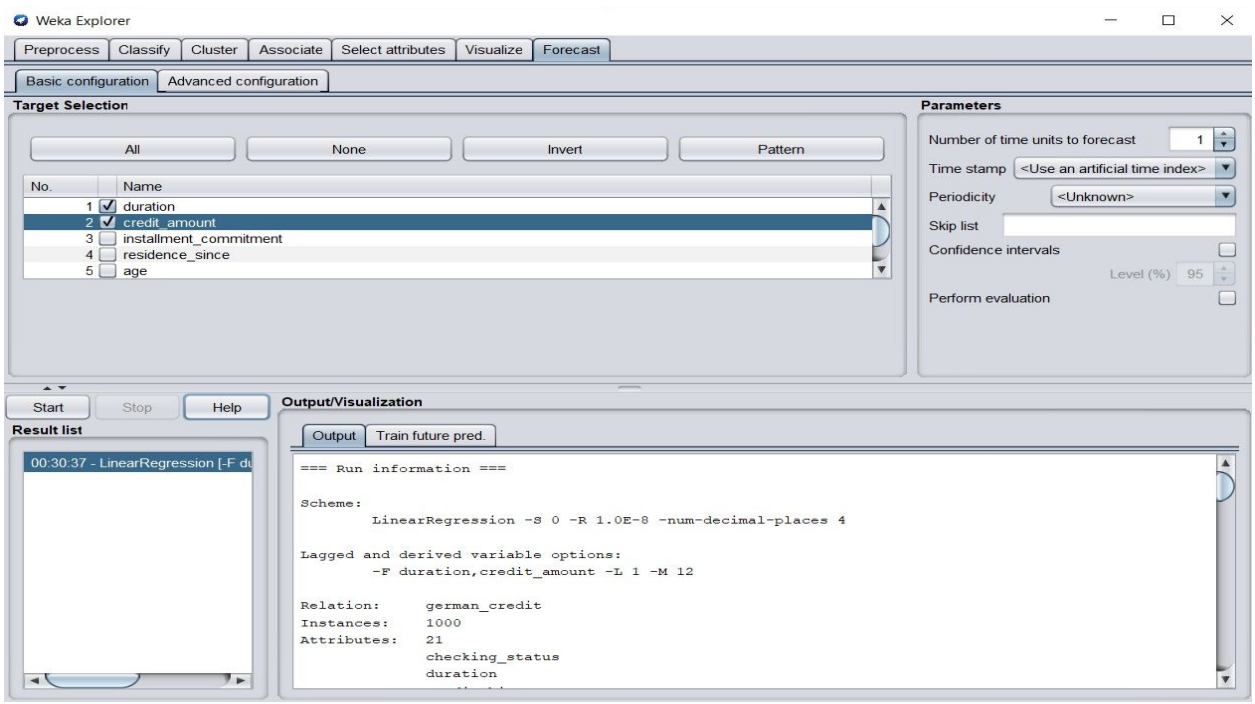


Package	Category	Installed version	Repository version	Loaded
raceSearch	Attribute Selection		1.0.1	
racedIncrementalLogitBoost	Ensemble learning		1.0.1	
realAdaBoost	Ensemble learning		1.0.1	
ridor	Classification		1.0.1	
rotationForest	Ensemble learning		1.0.2	
sasLoader	Converter		1.0.0	
scatterPlot3D	Visualization		1.0.7	
scriptingClassifiers	Classification		1.0.3	
sequentialInformationalBottleneckClusterer	Clustering		1.0.1	
simpleCART	Classification		1.0.4	
simpleEducationalLearningSchemes	Classification		1.0.1	
snowball-stemmers	Preprocessing		1.0.2	
stackingC	Ensemble learning		1.0.4	
streamingUnivariateStats	KnowledgeFlow		1.0.1	
supervisedAttributeScaling	Preprocessing		1.0.2	
tabuAndScatterSearch	Attribute selection		1.0.1	
tertius	Associations		1.0.1	
thresholdSelector	Classification		1.0.3	
tigerjython	Scripting		1.0.5	
timeSeriesFilters	Filters, Time Series		1.0.1	
userClassifier	Classification/regression		1.0.3	
vfdr	Classification, Online learning		1.0.0	
vines	Density Estimation		1.0.0	
votingFeatureIntervals	Classification		1.0.1	
wavelet	Preprocessing		1.0.1	
wekaDeeplearning4j	Classification/Regression		1.5.14	
wekaPython	Python integration		1.0.11	
wekaServer	Server		1.0.10	
winnow	Classification		1.0.1	

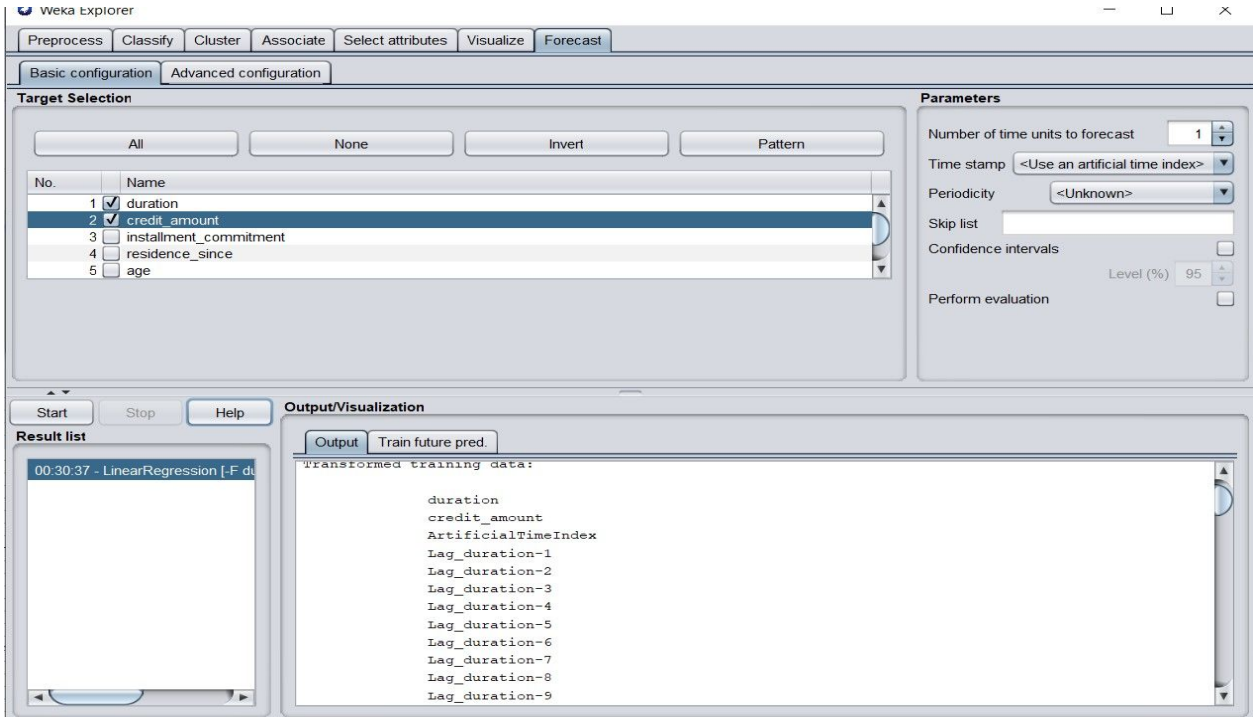
We have selected duration and credit_amount to build the time series model to forecast with the data. Number of units to forecast is assigned as 1 to forecast.



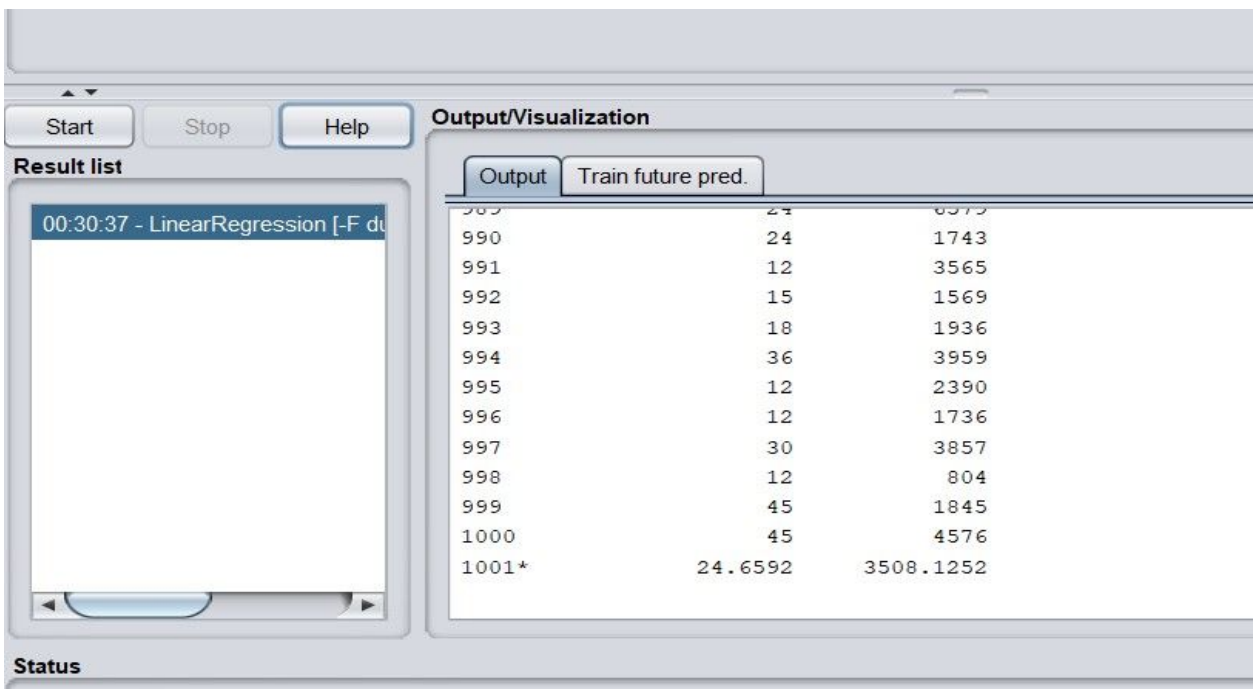
Here, we can see a linear regression equation which is generated on building a model. There are 1000 instances that can be observed; we forecasted the value of 1001 of duration and credit_amount.



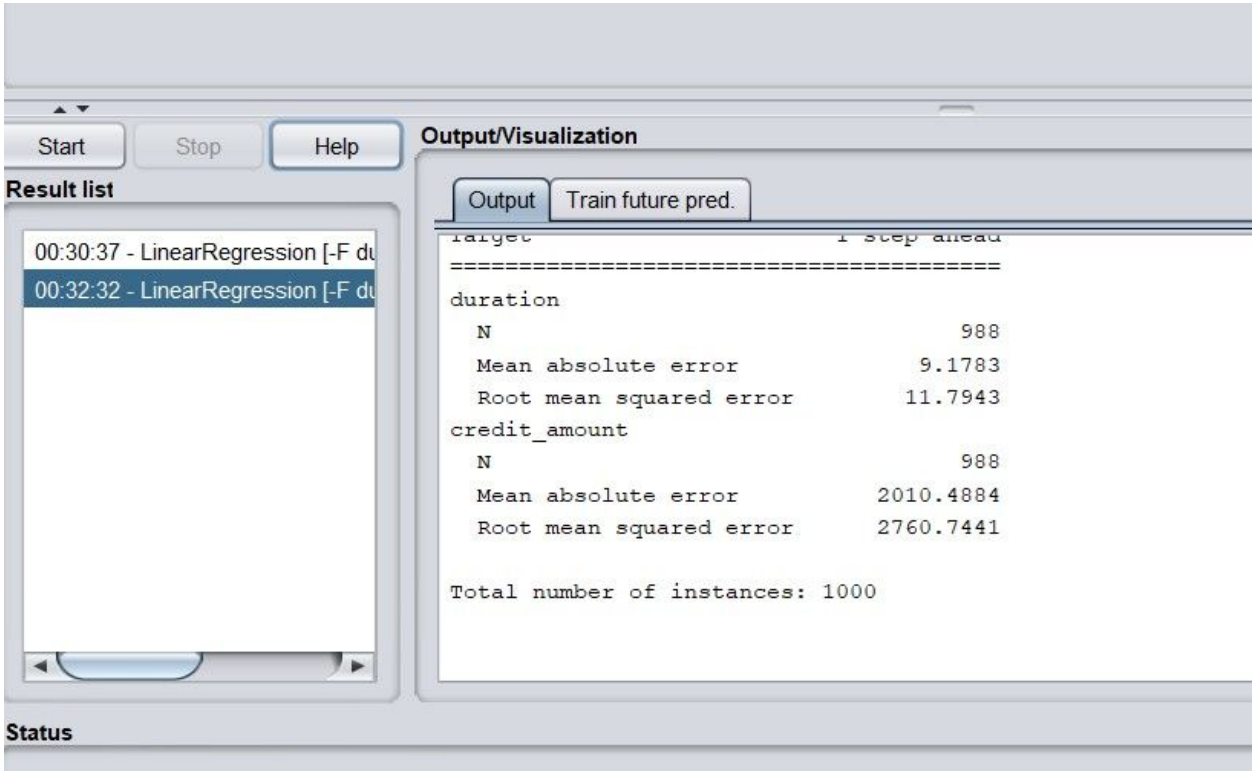
On building the model, lag values have been produced which can be seen in the screenshot below.



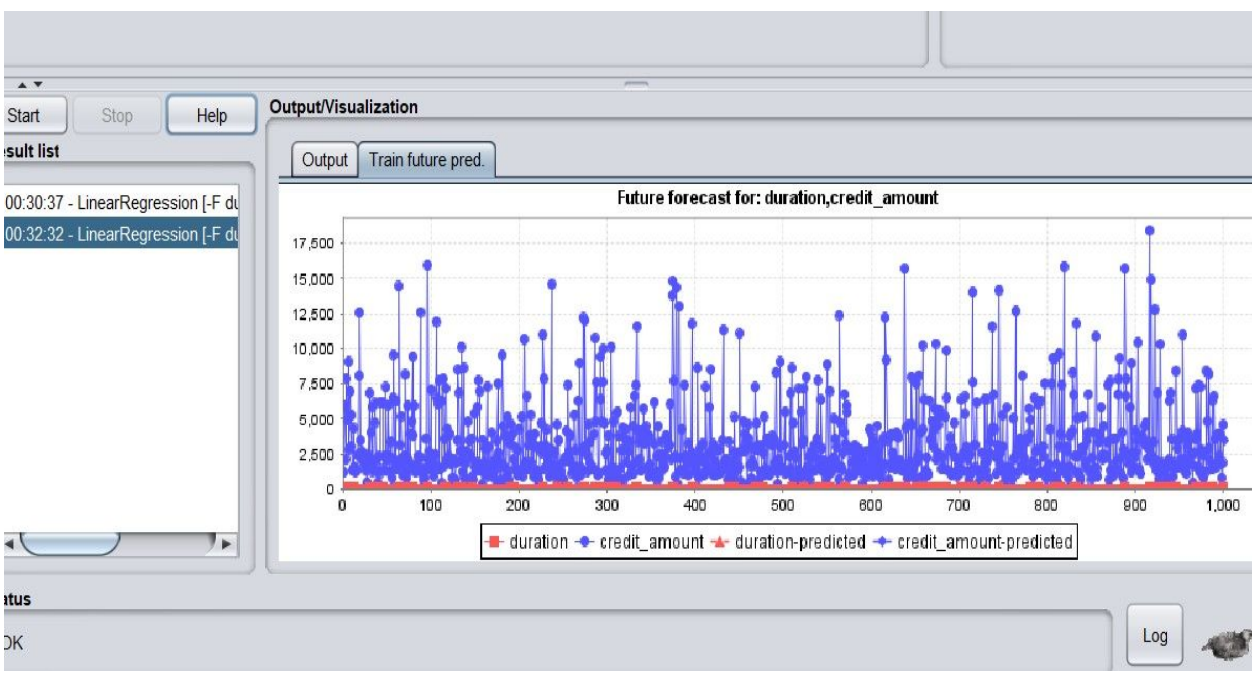
We can observe the 1001th value which is forecasted using weka time series package. Here we can also observe historical and predicted values.



By selecting the evaluation metrics, we can see the following results of MSE, RMSE of duration and credit_amount.



Visualization of the forecasted model.



REFERENCES

https://wiki.pentaho.com/display/DATAMINING/Time+Series+Analysis+and+Forecasting+with+Weka

https://medium.com/@kiaorahao/time-series-forecasting-of-china-stock-market-using-weka-part-1-introduction-71b68aed6d18

https://www.futurelearn.com/courses/advanced-data-mining-with-weka/0/steps/29457

https://www.cs.waikato.ac.nz/ml/weka/mooc/advanceddataminingwithweka/slides/Class1-AdvancedDataMiningWithWeka-2016.pdf

RESEARCH PUBLICATION

PUBLICATION AND RESEARCHERS.

Farhad Alam1 and Sanjay Pachauri2 1Research Scholar, Himalayan University, Arunachal Pradesh, India.

2Assistant Professor, IMS Unison University, Dehradun, Uttarakhand, India.

Breakthroughs in information and communication technology establish payment-collection technologies like Debit and Credit card systems at the level where their rapid penetration in the commercial market have led to an ever larger share of the world payment system and leading to a higher rate of stolen account numbers and subsequent losses by banks. Improved fraud detection thus has become essential to maintain the viability of the payment system, especially for e-commerce.

This tremendous growth in databases, fraud detection and payment security systems has spawned a pressing need for new techniques and tools that can intelligently and automatically transform data into useful information and knowledge. In recent years, data-mining (DM) has become one of the most valuable tools for extracting and manipulating data and for establishing patterns in order to produce useful information for decision-making.

This paper introduces three important data mining techniques J48, Naive Bayes and One-R classifier algorithm using weka workbench to achieve classification response for fraud detection dataset. Beside the basic description, paper compares these classifiers over different parameters and helps the e-commerce companies to select optimal classification algorithms.

DATASET : Data Used The current study uses German Credit fraud data proposed by Professor Dr. Hans Hofmann from the Institute for Statistics Hamburg having 1000 instances with categorical/symbolic attributes.

TECHNIQUES USED: Data Mining Tools; J48; Naive Bayes; One-R; Classification Methods.

CLASSIFICATION METHODS

Classification is a classic data mining technique based on the concepts of machine learning. General applications of classification are used as a tool to categorize items in a set of data into one of predefined sets of classes or groups. Classification method

makes use of mathematical techniques such as decision trees, linear programming, neural network and statistics. In classification, once the software is made that can learn how to classify the data items into groups. Classification—A Two-Step Process 1. Training: describing a set of predetermined classes. Each sample is assumed to belong to a predefined class, as determined by the class label attribute. The set of tuples used for model construction: training set.

The model is represented as classification rules, decision trees, or mathematical formula. 2. Classification: for classifying future or unknown objects. Estimate accuracy of the model. The known label of the test sample is compared with the classified result from the model. Accuracy rate is the percentage of test set samples that are correctly classified by the model. Test set is independent of the training set, otherwise over-fitting will occur.

NAIVE BAYES ALGORITHM

The Naive Bayes algorithm is an intuitive method that uses the conditional probabilities of each attribute belonging to each class to make a prediction. It uses Bayes' Theorem, a formula that calculates a probability by counting the frequency of values and combinations of values in the historical data. Parameter estimation for naive Bayes models uses the method of maximum likelihood. In spite of oversimplified assumptions, it often performs better in many complex real world situations. One of the major advantages of Naïve Bayes theorem is that it requires a small amount of training data to estimate the parameters.

DECISION TREE J48 ALGORITHM

J48 is an open source Java implementation of a simple C4.5 decision tree algorithm. J48 is an extension of ID3. The additional features of J48 are accounting for missing values, decision trees pruning, continuous attribute value ranges, derivation of rules, etc. Being a decision tree classifier J48 uses a predictive machine-learning model which calculates the resultant value of a new sample based on various attribute values of the available data. The internal nodes of a decision tree denote the different attributes; the branches between the nodes tell us the possible values that these attributes can have in the observed samples, while the terminal nodes tell us the final

1R OR ONE R CLASSIFIER ALGORITHM

The 1R or One R classifier for machine learning classification problems is one of the very simple and most effective classifier algorithms. In comparison to its Zero R Classifier, 1 R does not rely on the frequency of target but induces classification rules based on the value of a single predictor. In order to develop a rule set for a predictor, a frequency table corresponding to each predictor against the target is created. It is

evident that 1 R produces rules only little less accurate than cutting edge classifiers while producing rules that are simple for humans to interpret

MAJOR FINDINGS

There are so many benchmarks comparing the performance and accuracy of different classification algorithms but there are still very few experiments carried out on Credit card risk assessment and fraud detection datasets. In this work, we focus on various classification techniques most frequently used in data mining and compare the performance and the interpretation level of confidence on different classification techniques applied on Credit card datasets in order to determine which one is more suitable. From the result we see that time to build the model is less when using j48 and correctly classified instances are more and prediction accuracy is also greater in j48 than the other two.

Hence it is concluded that j48 performed better on credit card dataset. Beside the credit card dataset this can also be concluded that the different classification algorithms are designed to perform better for certain types of dataset. The dataset classifies people described by a set of attributes as good or bad credit risks. It Comes in two formats (one all numeric) with a cost matrix. Algorithms are applied to the dataset the confusion matrix is generated. Experiments are performed on Weka with 10 fold cross validation. Ten fold cross validation has been proved to be statistically good enough in evaluating the performance of the classifier.

RELEVANCE

My work is very much relevant to these researchers. Research paper published which is very much similar to my experiments performed during this assignment. This dataset classifies people described by a set of attributes as good or bad credit risks. My work was very much relevant to the researcher where they came up with an additional algorithm called **1R OR ONE R CLASSIFIER ALGORITHM** which i have not come across. Improved fraud detection thus has become essential to maintain the viability of the payment system to keep the devices and systems secure from threats. These experiments performed by the researchers are very much similar and relevant to my work. J48.Tree classification technique and Naive Bayes classification technique is being used by the researchers where I have used the same algorithms for the analysis in addition to time series, K-means clustering etc..