**Title:** Finding best Classifier and evaluating performance using test and training set for Glass Identification dataset

**Objective:**

In this Project, I will Choose five classifier algorithms and apply them into a dataset referred with http://archive.ics.uci.edu/ml/datasets/Glass+Identification . And also, I will find the ROC curve to make a precise decision that which is the best among the selected classifiers. Additionally, we will split our whole dataset into two segments. One is test set and another is training set. I will run this test data set with the training data set and trying to evaluate the performance of the test data. All the task will be done by Weka (a machine learning tool).

**Dataset description:**

In the dataset there are 214 instances and 11 attributes

Attribute Information:

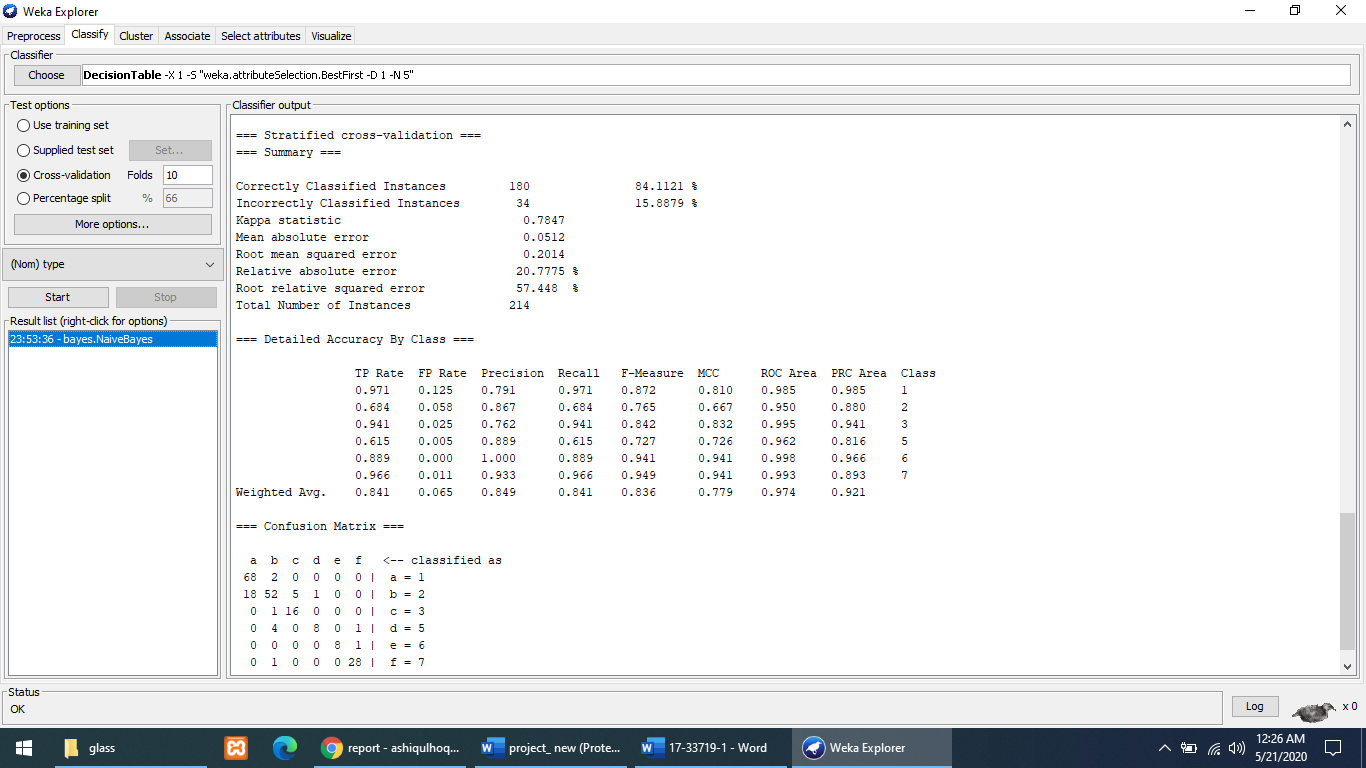
|  |  |
| --- | --- |
| Attributes | values |
| Id: instances number | 1-214 |
| RI: refractive index | Numeric |
| Na: Sodium | Numeric |
| Mg: Magnesium | Numeric |
| Al: Aluminum | Numeric |
| Si: Silicon | Numeric |
| K: Potassium | Numeric |
| Ca: Calcium | Numeric |
| Ba: Barium | Numeric |
| Fe: Iron | Numeric |
| Type of glass: (class attribute) | 1: building\_windows\_float\_processed (70 out of 214) |
| 2: building\_windows\_non\_float\_processed (76 out of 214) |
| 3: vehicle\_windows\_float\_processed (17 out of 214) |
| 4: vehicle\_windows\_non\_float\_processed (0 out of 214) |
| 5: containers (13 out of 214) |
| 6: tableware (9 out of 214) |
| 7: headlamps (29 out of 214) |

Note: There is no missing value of attributes

**Output:** We select five Classification Algorithms and apply them into the mentioned training set. All the results are as follows:-

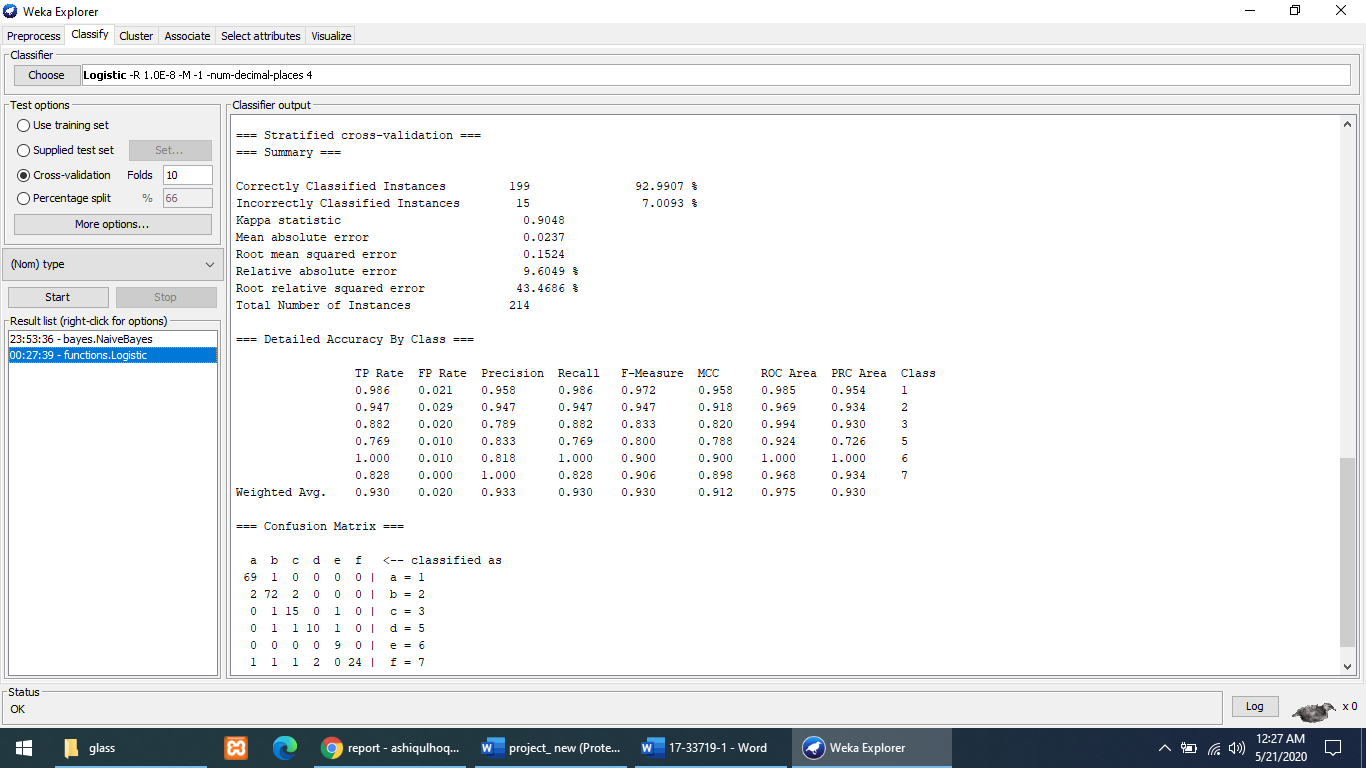
1. NaiveBayes

NaiveBayes Classifier works on Bayes theorem of probability to predict the class of unknown data sets. Here is output of it.



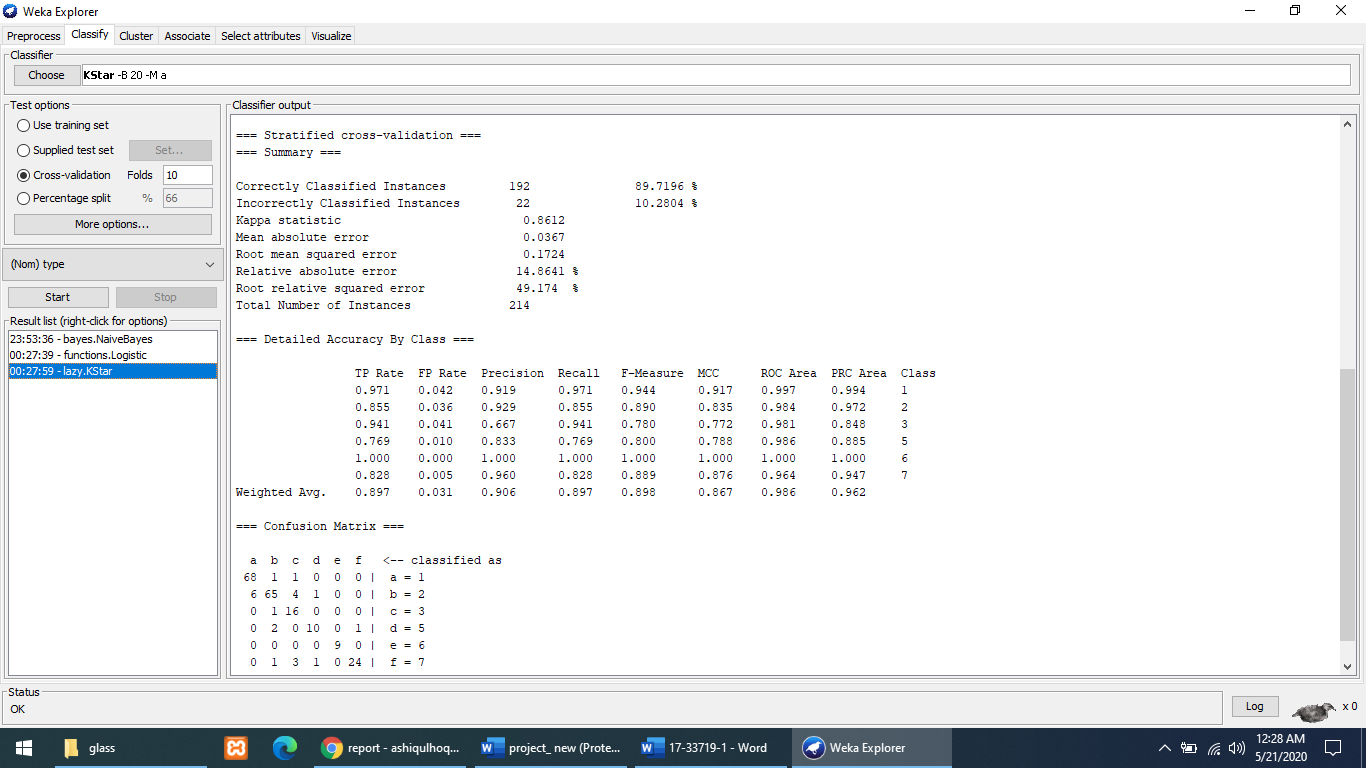
2.Logistic

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. Here is output of it.



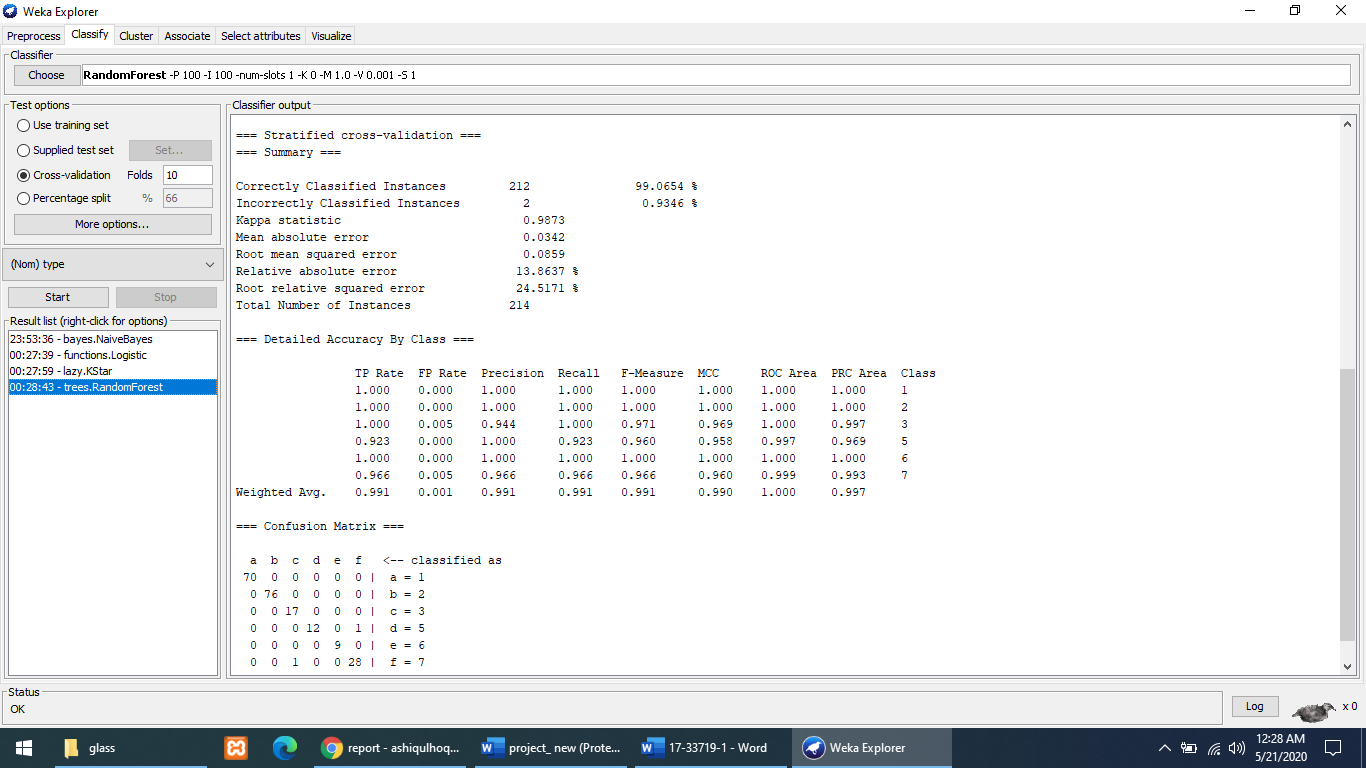
3.Kstar

K\* is an instance-based classifier, that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners in that it uses an entropy-based distance function. Here is output of it.



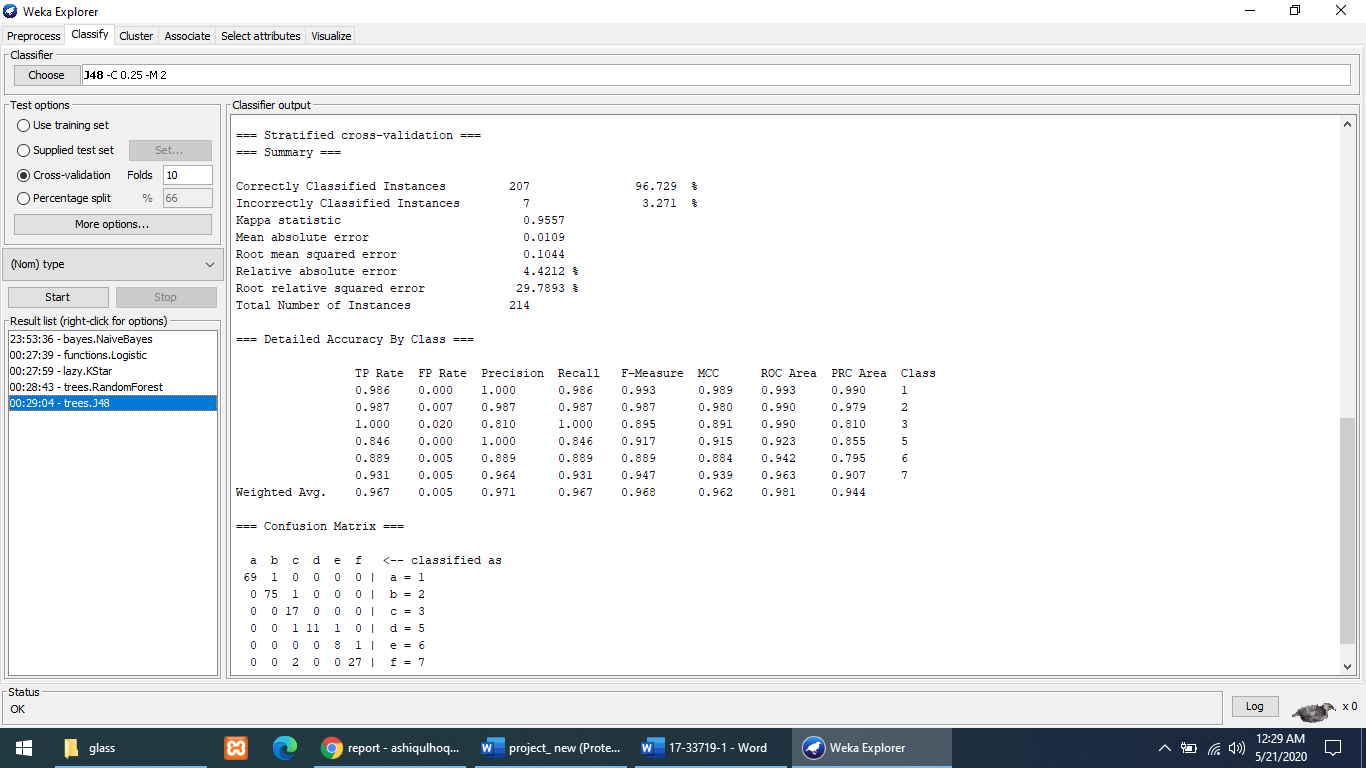
4.RandomForest

RandomForest Classifier Works like ‘Decision tree’ but the difference is it split tree randomly rather than traditional split of tree. Here is output of it.



5.J48

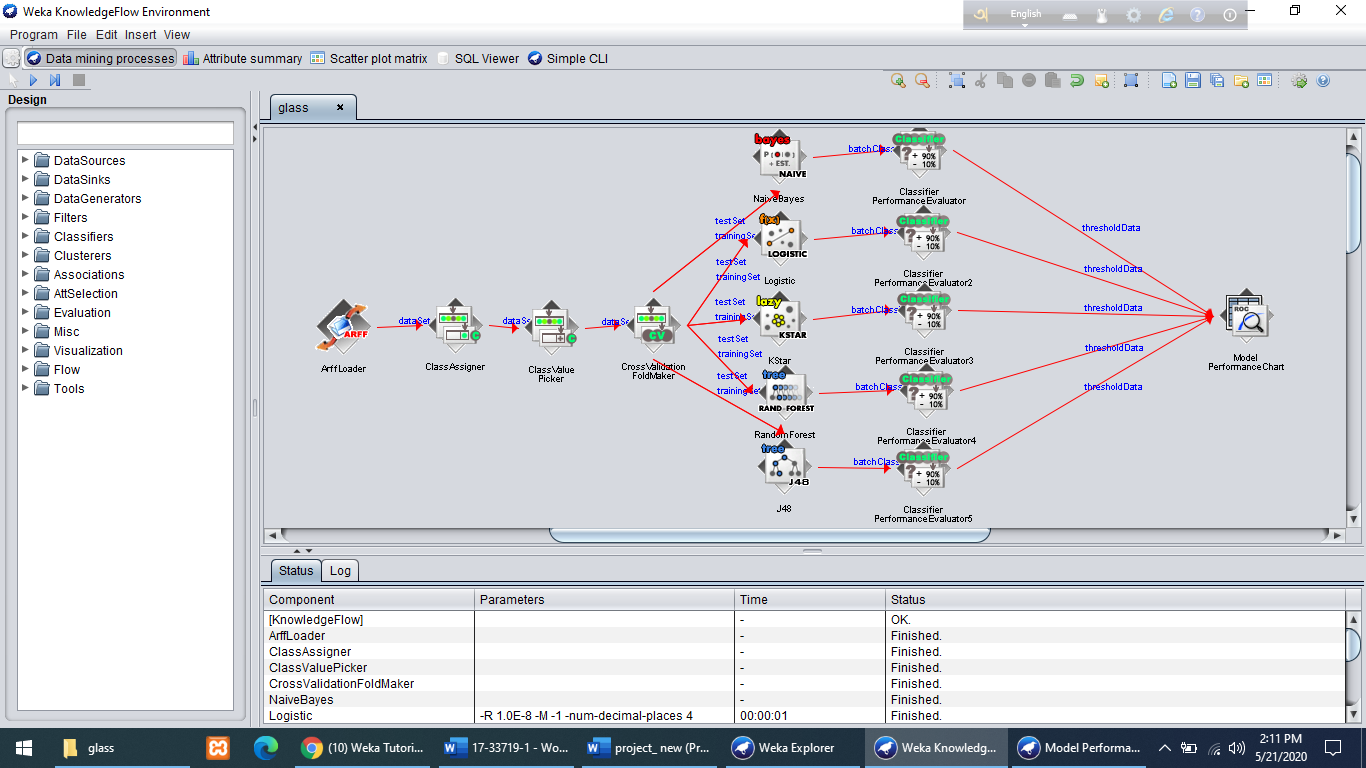
J48 algorithm is one of the best machine learning algorithms to examine the data categorically and continuously. Here is output of it.

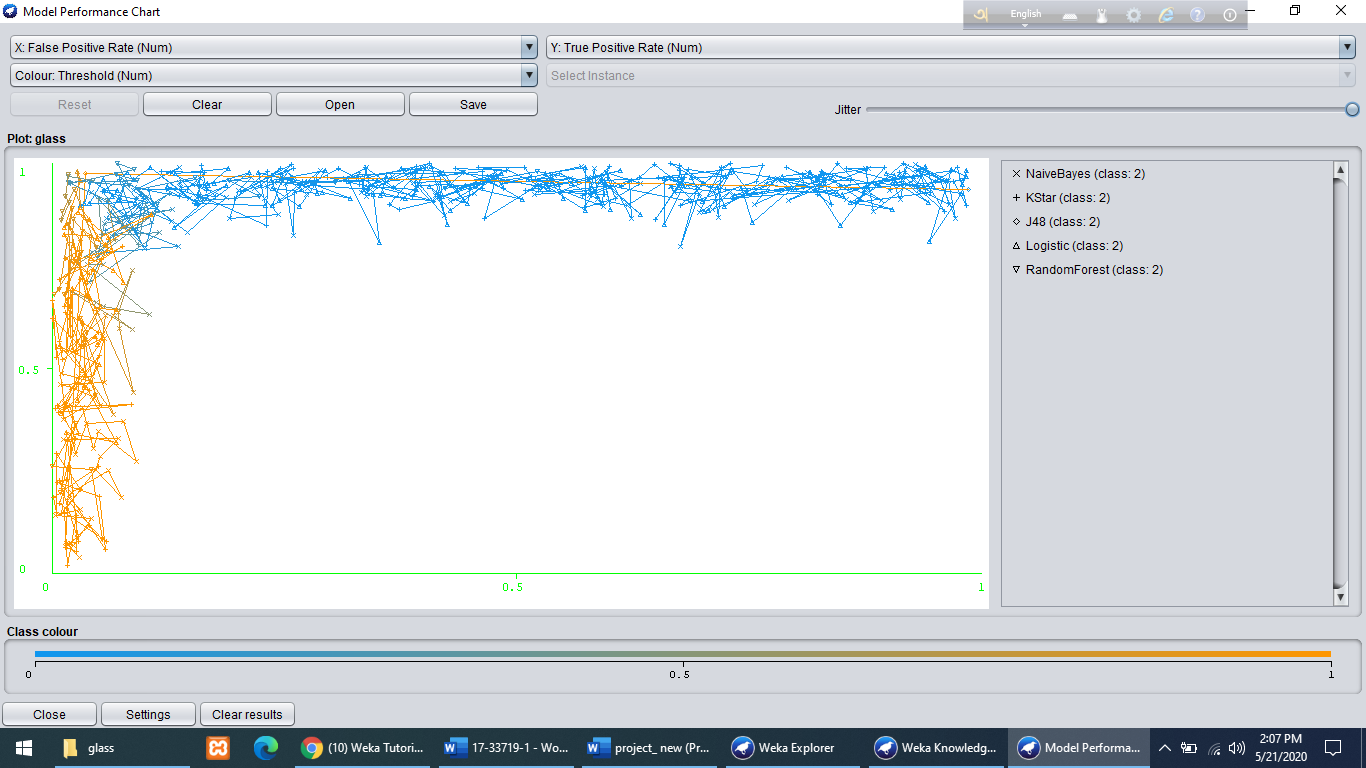


**Performance Analysis:**

|  |  |  |  |
| --- | --- | --- | --- |
| Classifiers | Correctly Classified instance | F-Measure (Weighted Avg.) | Total number of instances |
| NaiveBayes | 180 (84.1121 %) | 0.836 | 214 |
| Logistic | 199 (92.9907 %) | 0.930 |
| Kstar | 192 (89.7196 %) | 0.898 |
| **RandomForest** | **212 (99.0654 %)** | **0.991** |
| J48 | 207 (96.729 %) | 0.968 |

Here we see, among 214 instances Random Forest Correctly Classified 212 instance which is 99.0654 % and it’s the highest percentage among the other Classifiers. So ‘RandomForest’ is the best classifier for the dataset.

**Knowledge Flow Diagram**

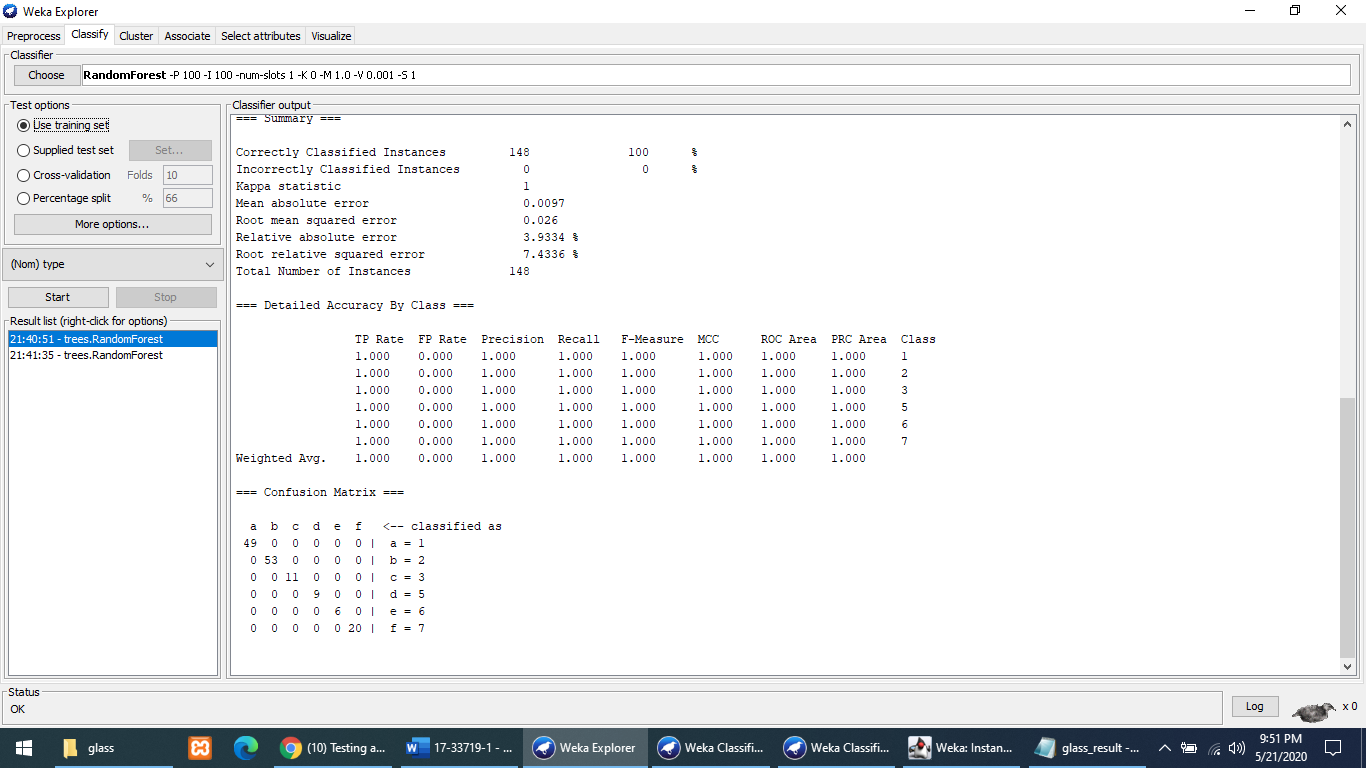
**ROC Curve**

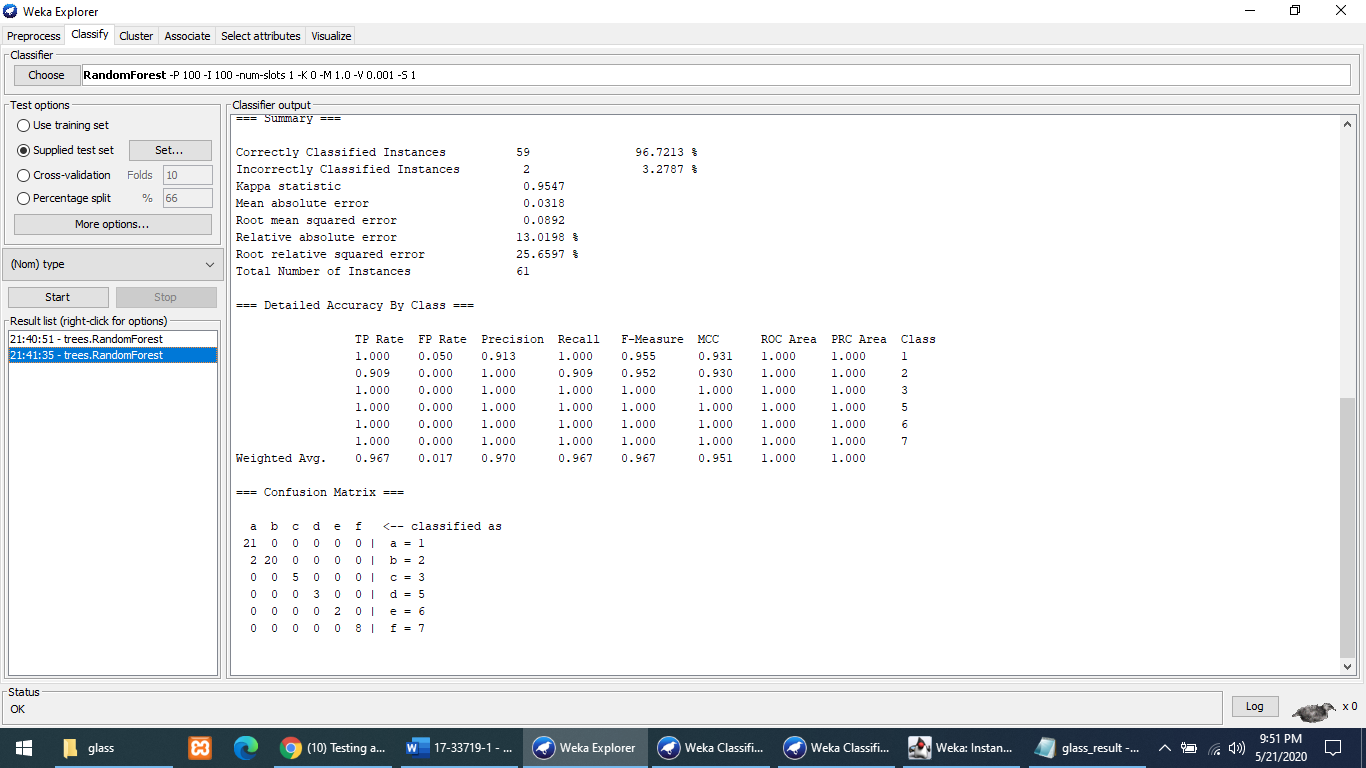
This ROC curve is merged of all algorithm that we used, considering class value 2. We know that the curve goes nearer to the y-axis claimed as the better classifier. From the ROC curve we see that curve of RandomForest is nearer than the other curves. So, we can say that ‘RandomForest’ is best for the dataset.

**Additional task**

In additionally, we split data into two segments. I take 70% of original data as training set(148 out of 214 instances) and 30% of data as test set(61 out of 214 instances). This task is done by weka, and run this test data set with the training data set. Here we evaluate the performance of test data by visualizing the classifiers error.

**Training set**



**Run with test data**

From the Classifiers output, I determined that the performance is 96.7213 %. It Classified 59 instances correctly out of 61 test data set.

**Classifiers error visualization** 