**Data Mining Project**

**Damini Patel**

**Aaditya Morey**

**Purpose-**

* **To analyze different types of datasets with different types of algorithms available in the WEKA and command prompt.**
* **To perform different validation techniques and tests to check the accuracy and the error rates in the datasets.**
* **To compare different algorithm performance on the same data sets to find out the best results.**
* **To study about the parameters that plays important role in data mining while using algorithms on dataset.**
* **To study about the behavior of different algorithms on same dataset.**
* **To study about the Datamining techniques in terms of classification, comparison, clustering.**

**Data Sets used –**

1. Teaching Assistant
2. Cars
3. Tic Toc
4. Balance Scale

**Rationale –**

* Datasets were selected to satisfy all the criteria of the project requirements.
* Selection is done in such way that different types of algorithms can be used.
* Different types of data sets covered which can cover different types of class values and attribute values.
* Rationale for choosing database depends on the algorithm behavior according to the data set type. We have done research on algorithms and chose best that could be used on the selected datasets.

1. **Teaching Assistant**

Information

The data consist of evaluations of teaching performance over three regular semesters and two summer semesters of 151 teaching assistant (TA) assignments at the Statistics Department of the University of Wisconsin-Madison. The scores were divided into 3 roughly equal-sized categories ("low", "medium", and "high") to form the class variable. The data has 151 instances, 5 attributes and no missing values.

Attribute characteristics are categorial and integer type.

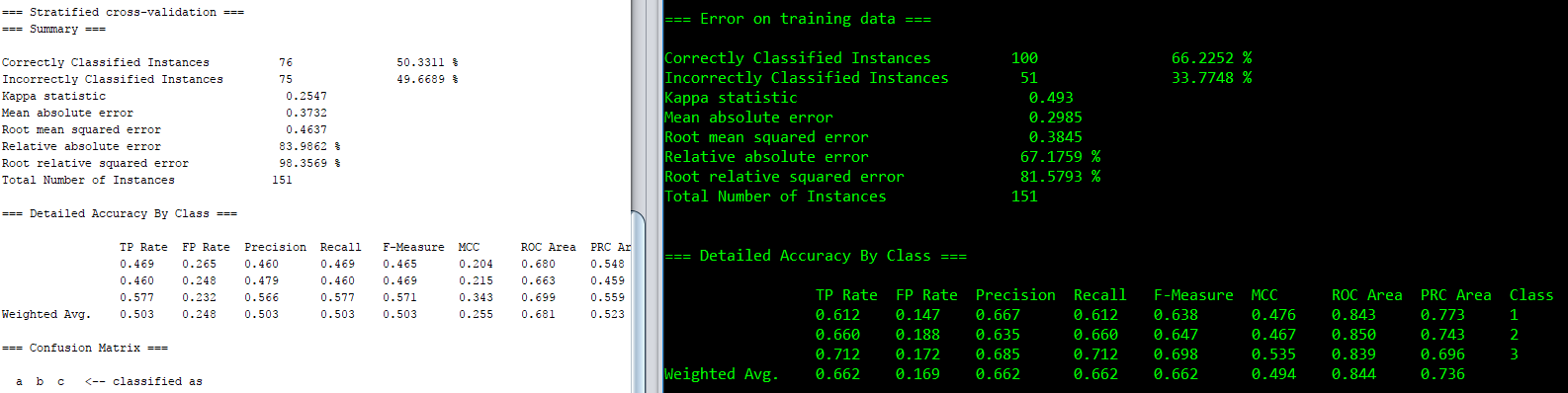
Algorithms used

We decided to use **Bayes Naïve, ZeroR, SVM(SMO)** on this dataset.

Reason for using these classifier algorithms –

Naïve Bayes:

* It is faster and more accurate, highly scalable and good method for text categorization.
* It suits the datasets which are independent and good for classifying in terms of separate attributes.
* It supports Multiclass classification. It performs better with multi-class attributes.



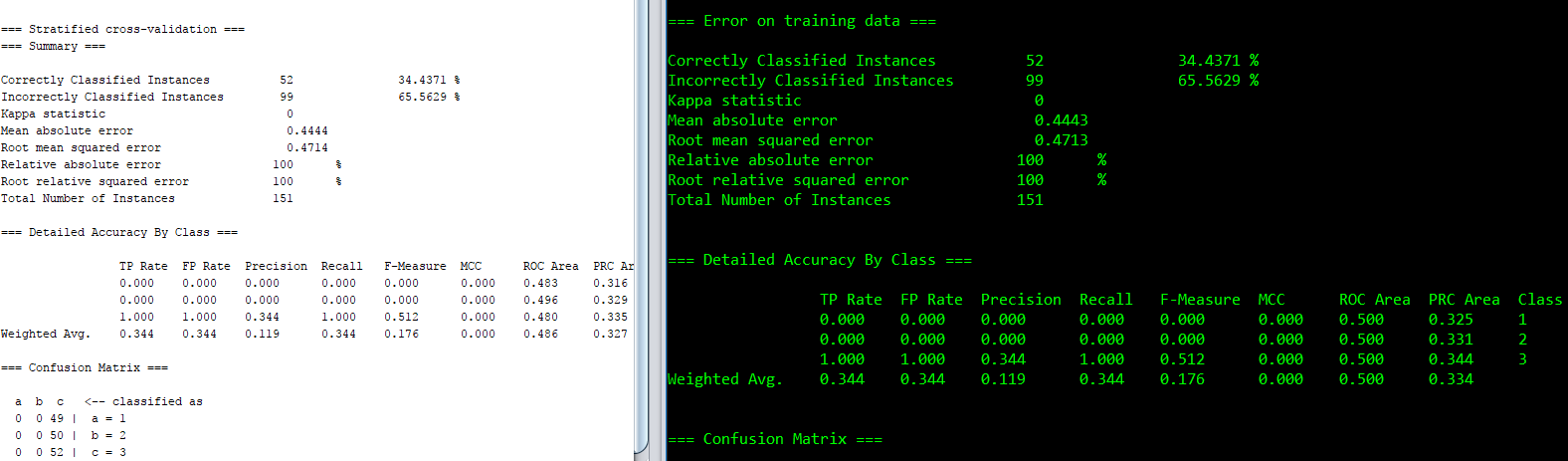
**Correctly Classified instances = 50.33%**

**Incorrectly classified instances= 49.66%**

**ROC= 0.680**

ZeroR:

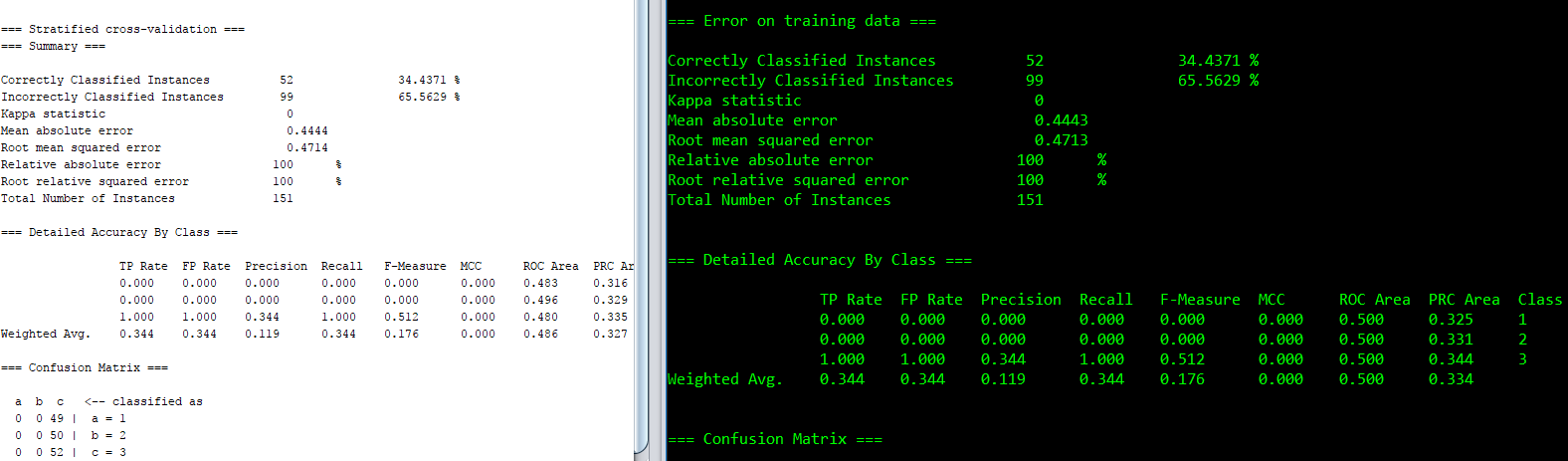
* + ZeroR simply predicts the Majority Class.
  + For the dataset like this this is probably the good algorithm to use.



**Correctly Classified instances =34%**

**Incorrectly classified instances= 65%**

**ROC= 0.483**



**Correctly Classified instances =34%**

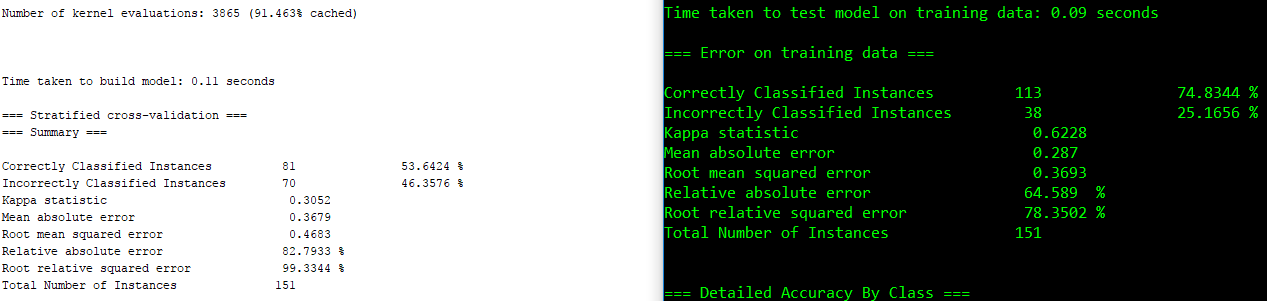
**Incorrectly classified instances= 65%**

**ROC= 0.500**

SVM[functions/SMO]:

It has been used successfully in many real-world problems

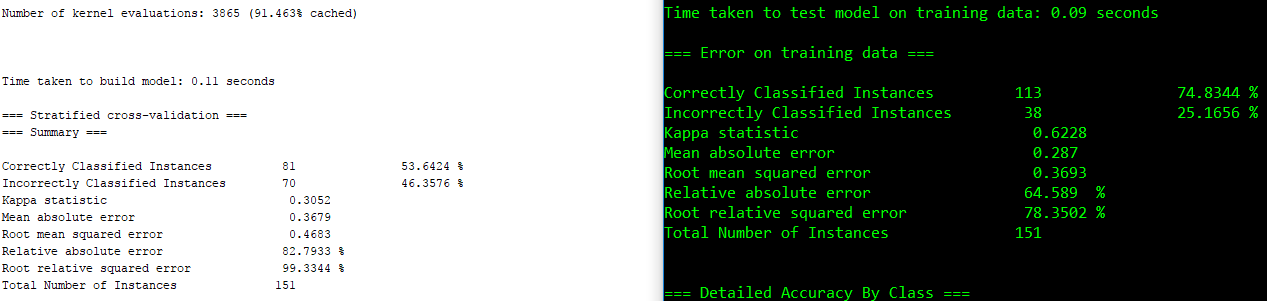
* text (and hypertext) categorization
* image classification
* hand-written character recognition



**Correctly Classified instances =53.83%**

**Incorrectly classified instances= 46.35%**

**ROC= 0.861**



**Correctly Classified instances =74.83%**

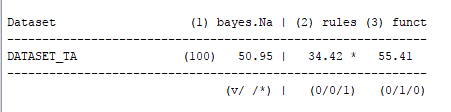
**Incorrectly classified instances= 25.16%**

**ROC= 0.861**

Reason for not selecting following Algorithms:

Linear regression or trees: these types of algorithms need dependent and related attributes.

Comparison



1. **Cars**

Information

This dataset contains the information about cars which helps evaluate cars according to few parameters which are acting as attributes in this data set. This dataset contains data with 1728 instances which are assigned within 6 different attributes and it has no missing values in this dataset. The dataset is good match for the classification. This dataset has multivariate and categorical characteristics.

*Challenges and solution:*

Originally, all the attributes were ordinal, but WEKA doesn’t have option for ordinal values, so we changed it to nominal.

This data is basically classified into four types of class values i.e. Unacceptable, acceptable, good very good.

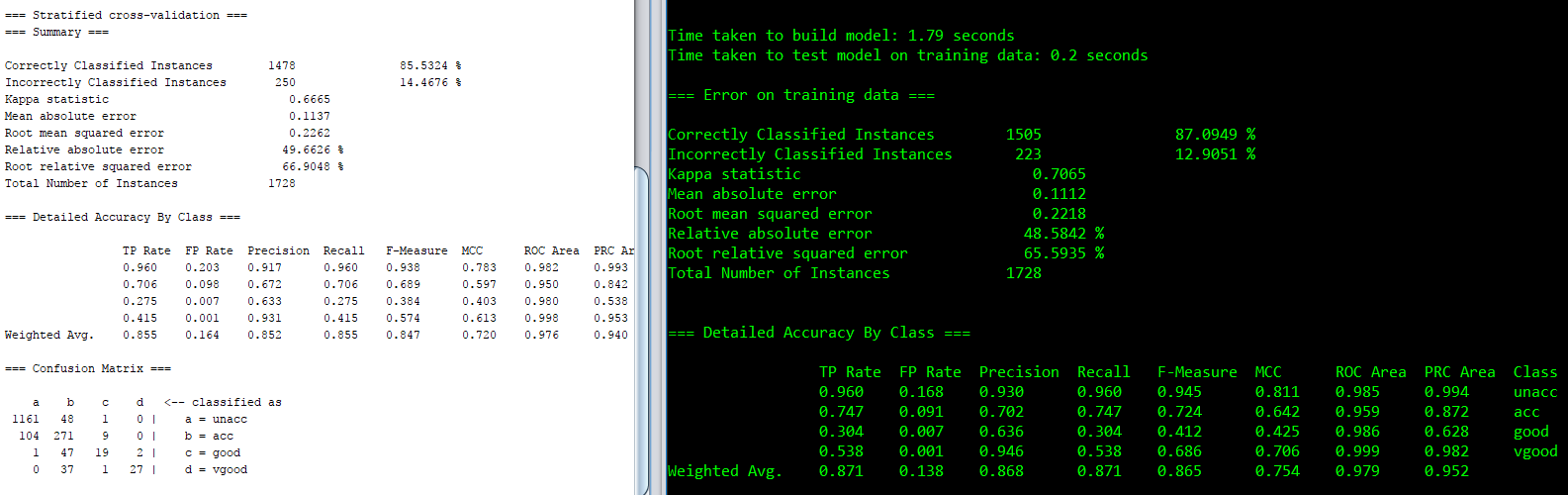
Algorithms used

We decided to use **J48** trees classifier, **SMO** and **Bayes Naïve** on this dataset.

Reason for using these classifier algorithms –

Naïve Bayes:

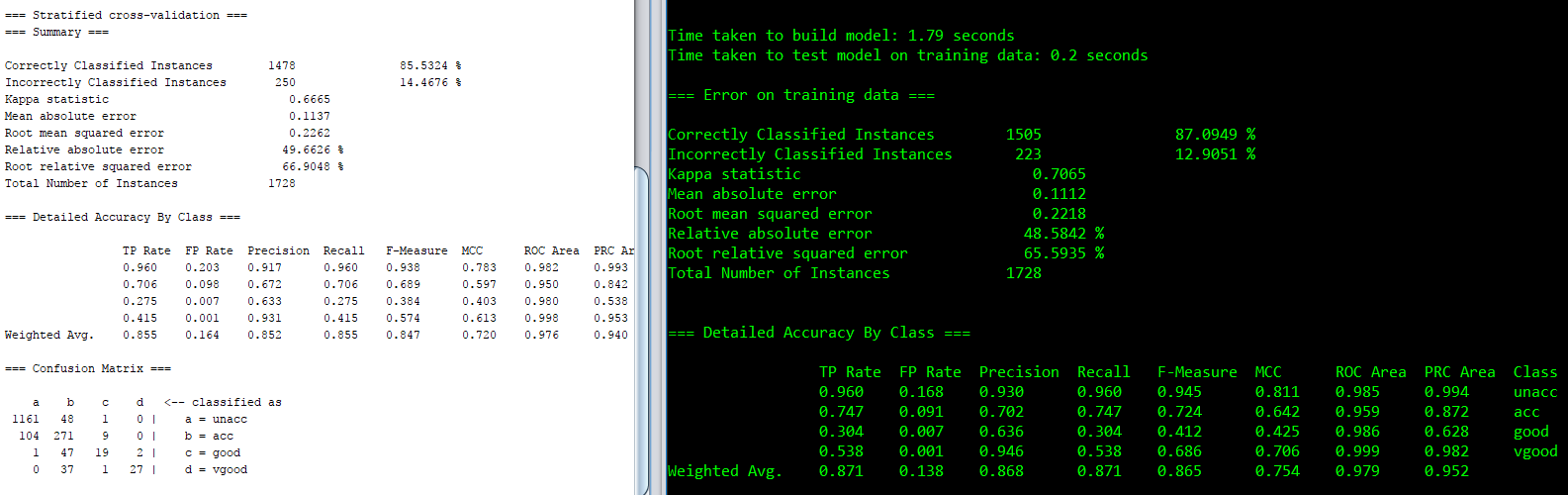
* In this case Naïve Bayes is helpful for the Multi classes attribute values, String values and Text Values.
* Out of other Bayes algorithms Naïve Bayes has better output in comparison



**Correctly Classified instances =85.53%**

**Incorrectly classified instances= 14.46%**

**ROC= 0.982**



**Correctly Classified instances =87.09%**

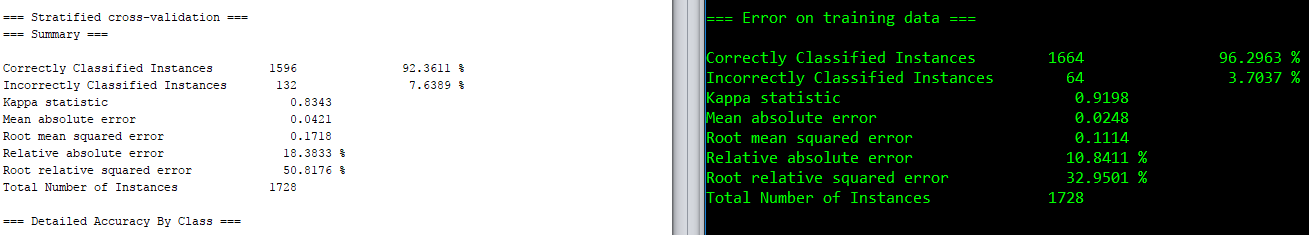
**Incorrectly classified instances= 12.90%**

**ROC= 0.979**

J48 (Decision Tree):

J48 is used because it supports well in the below mentioned attribute types

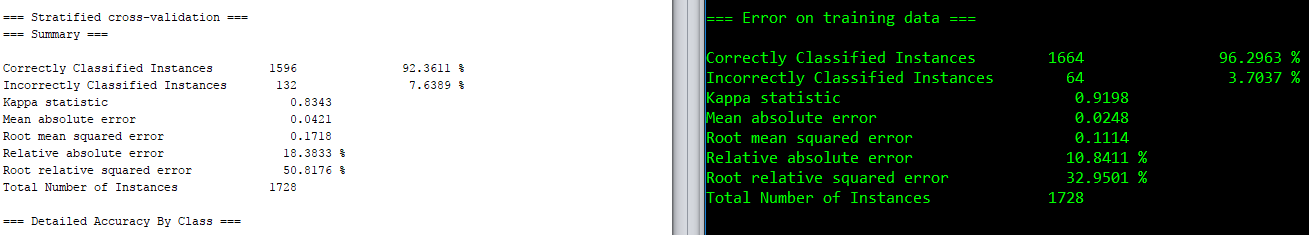
* In this dataset (cars) it demands to take decision between cars. Between acceptable or unacceptable or good or very good.
* When it comes to regression J48 is the one of the best algorithm believed/ and proven to be successful in data mining.
* In this dataset class values are dependent on the attributes.
* This is possibly most flexible Algorithm which can classify the data with different types of attributes. This C4.5 chooses the attribute of the data that most effectively splits list set of samples into subsets and puts them into one class or the other. It simply creates a leaf node to make the decision to decide where to put the next data.



**Correctly Classified instances =92.36%**

**Incorrectly classified instances= 7.63%**

**ROC= 0.947**



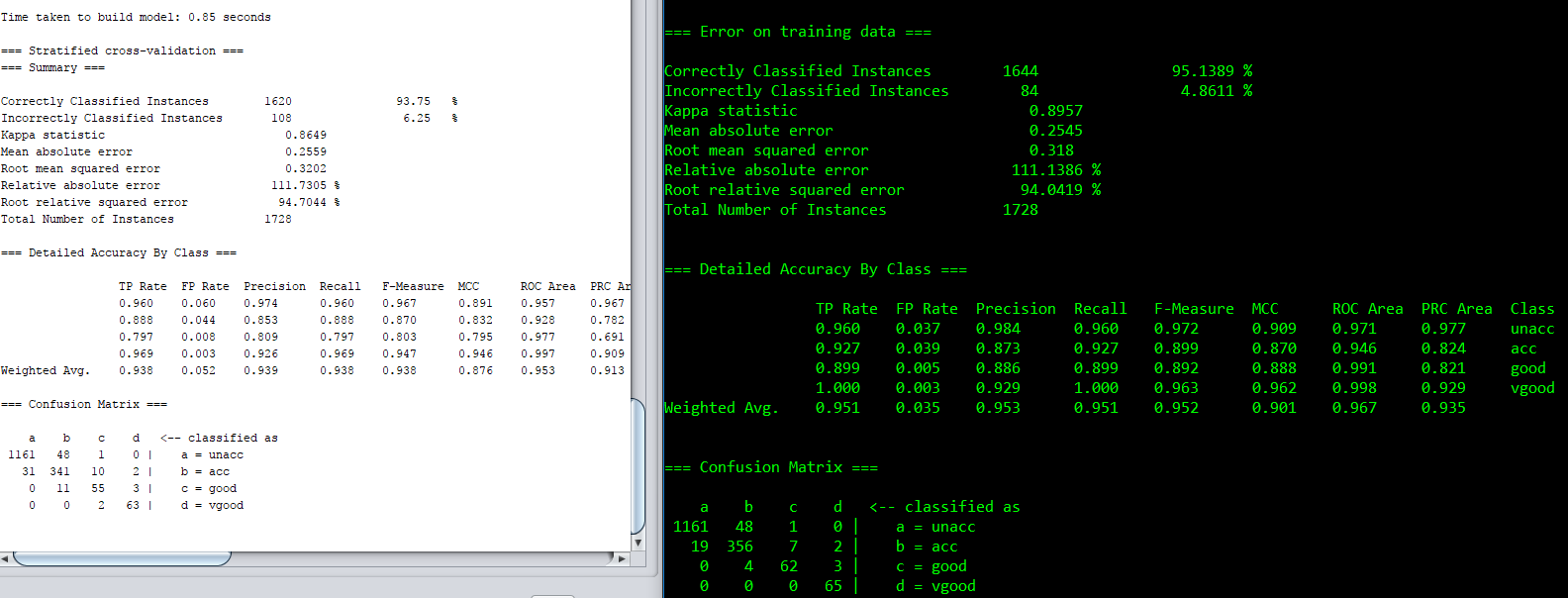
**Correctly Classified instances =96.26%**

**Incorrectly classified instances= 3.70%**

**ROC= 0.997**

SMO:

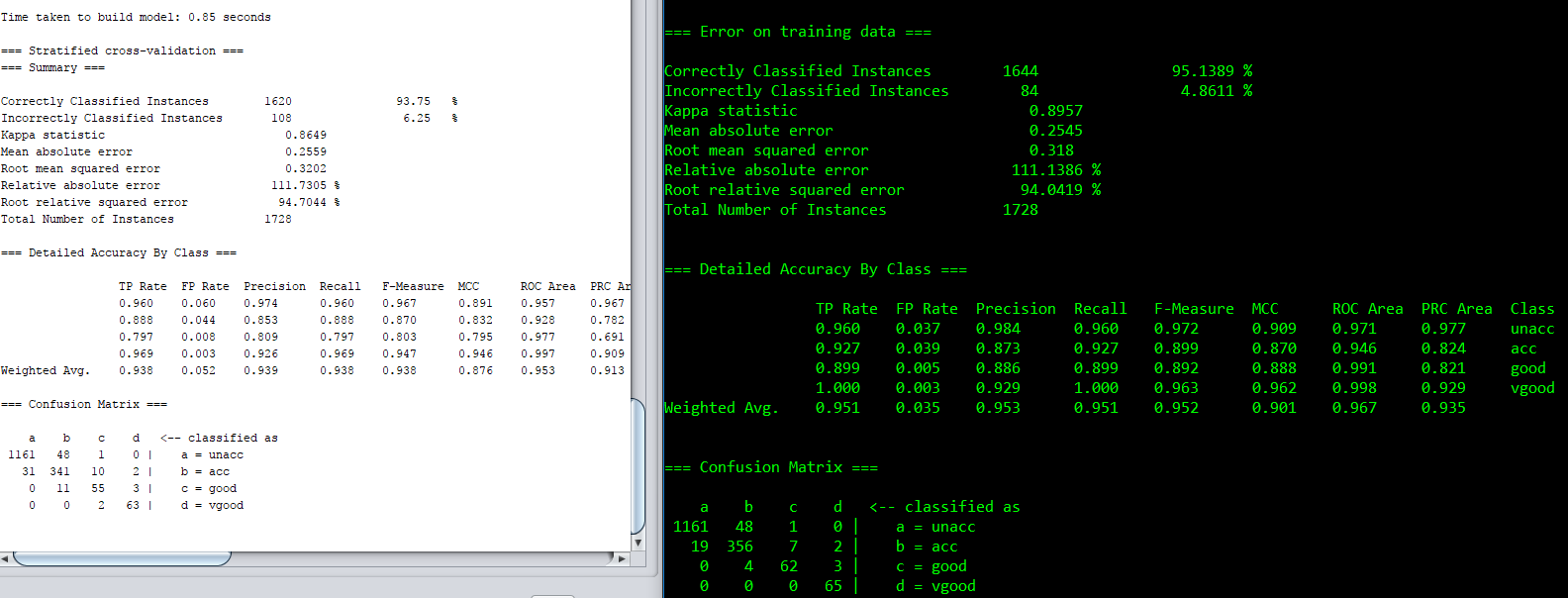
* Reason for using SMO algorithm is that it is suitable for many real-world problems.
* It works better on the text categorization application.
* In this case SMO has given more accuracy then the other function algorithms



**Correctly Classified instances =93.75%**

**Incorrectly classified instances= 6.25%**

**ROC= 0.957**

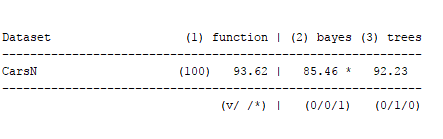


**Correctly Classified instances =95.13%**

**Incorrectly classified instances= 4.86%**

**ROC= 0.967**

Comparison between used algorithms



1. **TIC TAC TOE**

Information:

This database encodes the complete set of possible board configurations at the end of tic-tac-toe games, where "x" is assumed to have played first. The target concept is "win for x" (i.e., true when "x" has one of 8 possible ways to create a "three-in-a-row").

This dataset is classified into two class values i.e. Positive and Negative.

Algorithms used:

We decided to use **Logistic, J48 and SMO**

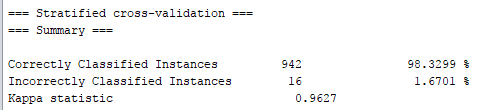
Reason For using These Algorithms

The data type in this data set was in ordinal class form and Binary

In this case we used decision trees and tested since the requirement is decision making in the game and then tried logistic which seemed to be the best out of all for such dataset.

Logistic:

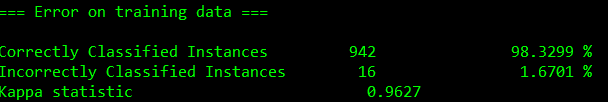
* logistic algorithm which made us realized there is huge difference between the accuracy. Logistic algorithm works way better on the binary data.
* This experiment reveled the technique to data mining. – It is very Important to observe the relation between class and attributes moreover its important to know or predict which algorithm to use on which data set.



**Correctly Classified instances =98.32%**

**Incorrectly classified instances= 1.67%**

**ROC = 0.996**



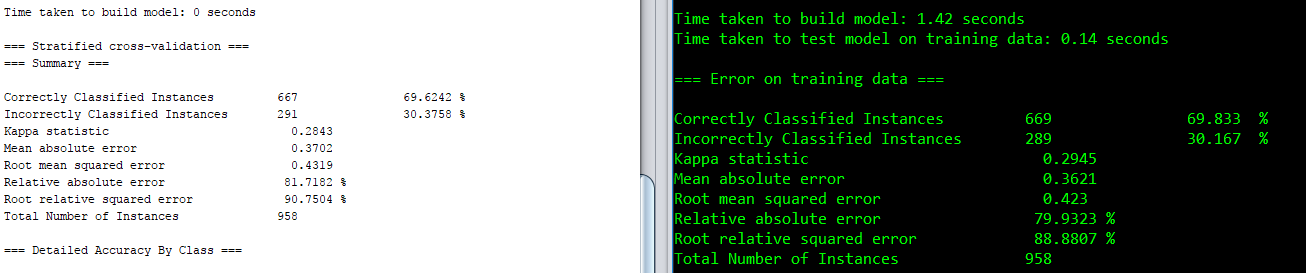
**Correctly Classified instances =98.32%**

**Incorrectly classified instances= 1.67%**

**ROC = 0.998**

Naïve Bayes:

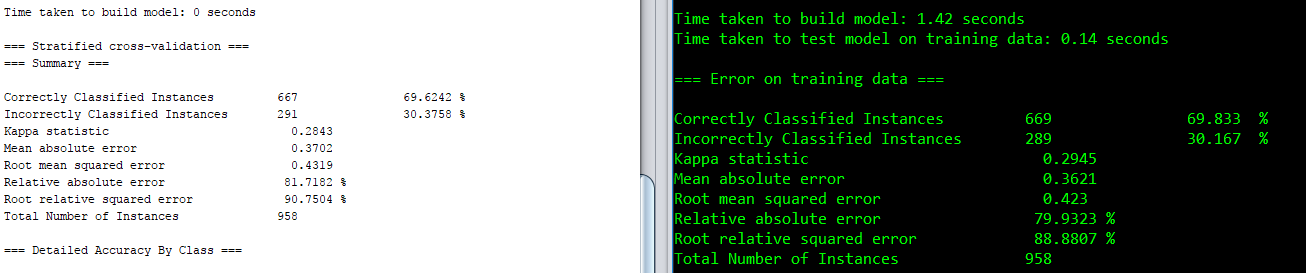
* We decided to use this one because the values are dependent and Naïve Bayes is more effective on such data sets.
* Classification/ Rules



**Correctly Classified instances =69.833%**

**Incorrectly classified instances= 30.167%**

**ROC= 0.758**

.

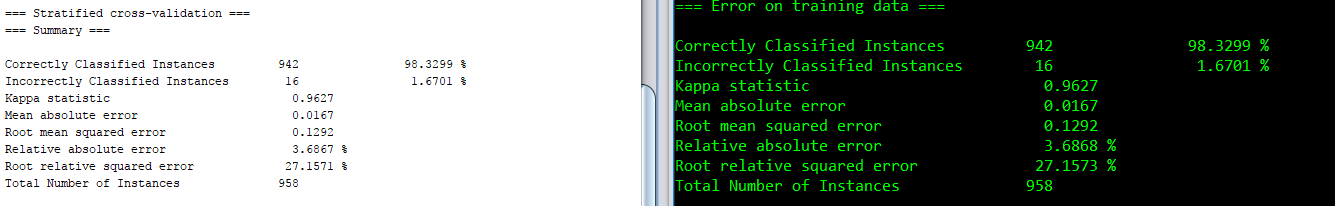
**Correctly Classified instances =69.833%**

**Incorrectly classified instances= 30.167%**

**ROC= 0.769**

SMO:

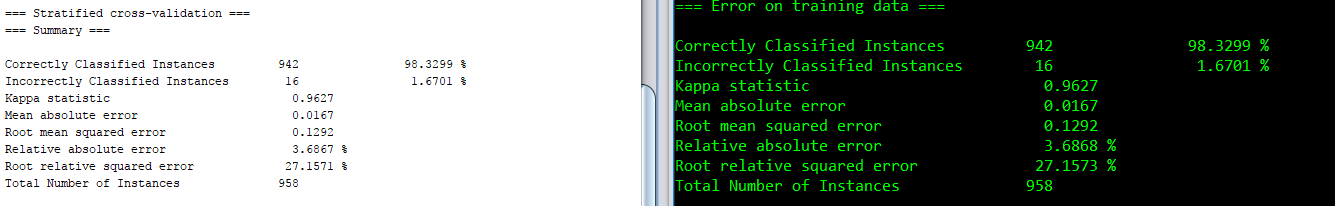
* This We used specifically for binary data.
* SMO is so far proven to be the best on binary data.



**Correctly Classified instances =98.32%**

**Incorrectly classified instances= 1.6701%**

**ROC= 0.976**

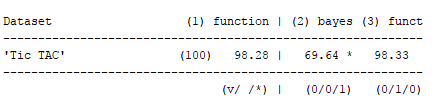


**Correctly Classified instances =98.32%**

**Incorrectly classified instances= 1.6701%**

**ROC= 0.976**

Comparison:



1. **Balance Scale**

Information

This data set was generated to model psychological experimental results. Each example is classified as having the

balance scale tip to the right, tip to the left, or be balanced. The attributes are the left weight, the left distance, the right weight, and the right distance. The correct way to find the class is the greater of (left-distance \* left-weight) and (right-distance \*right-weight). If they are equal, it is balanced.

This dataset is classified into 3 class values i.e. L,,B,R.

All the attribute types are nominal. But originally data set was in ordinal but since ordinal isn’t supported by Weka we changed it to nominal to function properly.

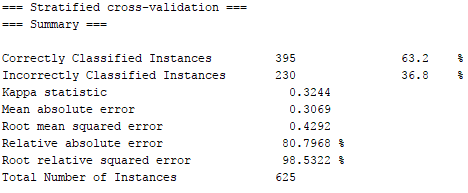
Algorithms used:

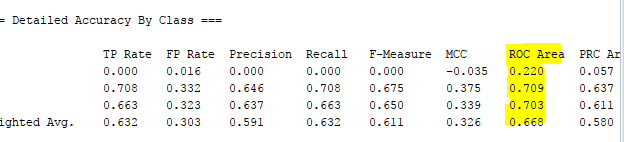
We decided to go ahead with **J48(decision Tree), Naïve Bayes and Logistic.**

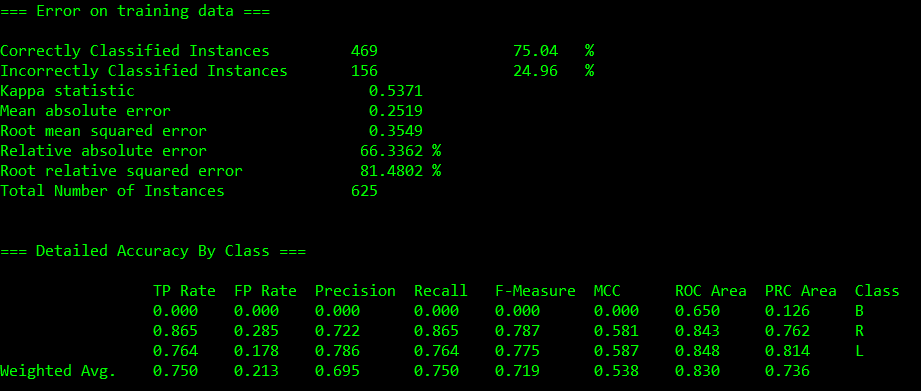
Reason for using these algorithms:

J48:

* The balance scale dataset has class values dependent on other values.
* The relation can be found between attributes.
* So, it is best to apply Decision Tree on this dataset.







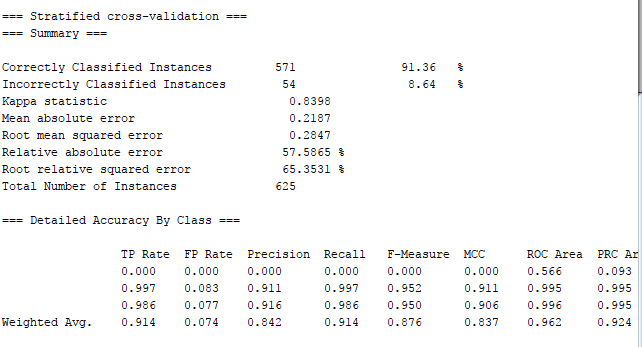
**Correctly Classified instances =75.04%**

**Incorrectly classified instances= 24.96%**

**ROC= 0.650**

Naïve Bayes:

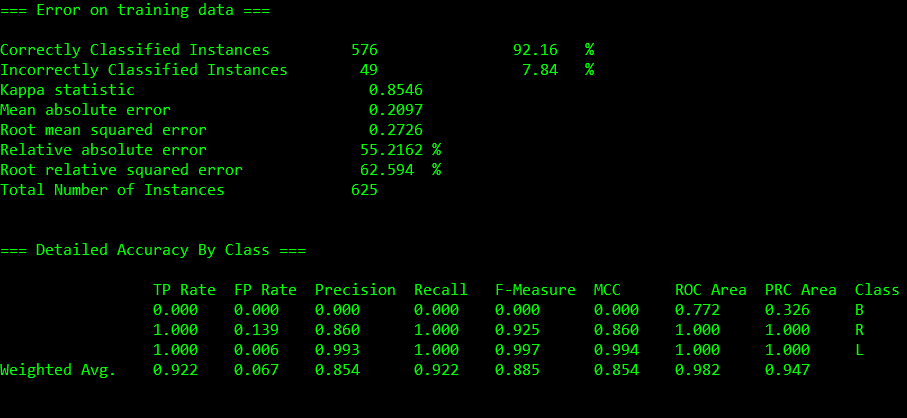
* Reason to use Naïve Bayes in this case was that the data is in text format
* Multiclass values exist in the data set.
* Naïve Bayes worked one of the best on this dataset



**Correctly Classified instances = 92.36%**

**Incorrectly classified instances= 6.64%**

**ROC= 0.566**



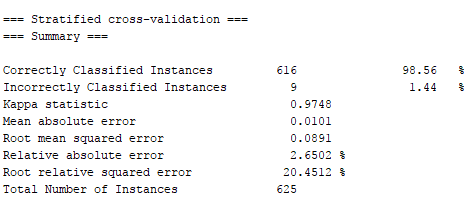
**Correctly Classified instances = 92.16%**

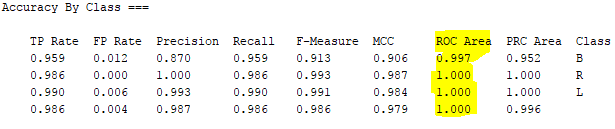
**Incorrectly classified instances= 7.84%**

**ROC= 0.982**

Logistic:

* We used this Algorithm because of the dependency on attribute was established.
* From the previous records and after checking other algorithms on this it was found one of the best.



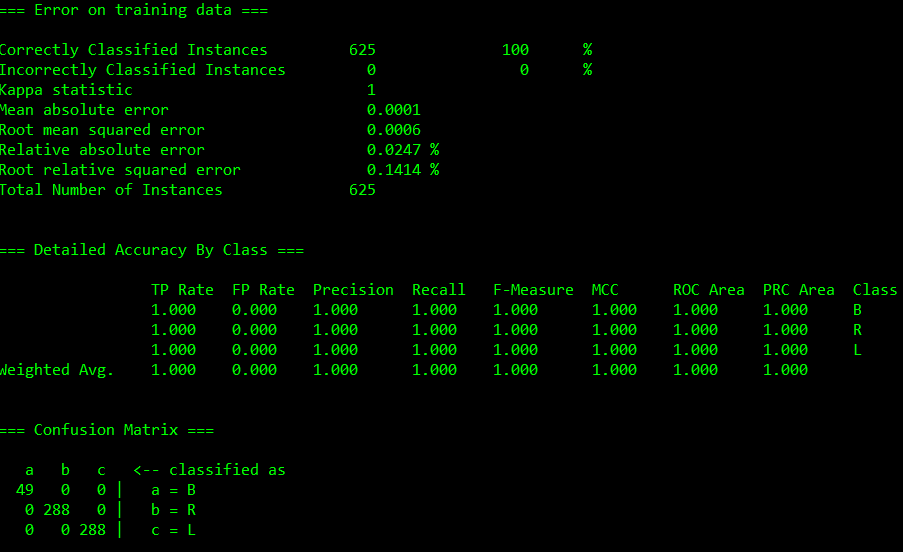


**Correctly Classified instances =98.56%**

**Incorrectly classified instances= 1.44%**

**ROC= 1.0**

**IT takes longer to calculate but its more accurate.**

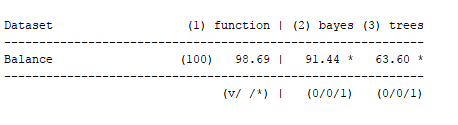


**Correctly Classified instances =100%**

**Incorrectly classified instances= 0%**

**ROC= 1.0**

Comparison:



**Results and Discussion**











Conclusion Statement-

* In this Project we have successfully learnt how to find the perfect and most effective algorithms for the datasets for data mining.
* There are many parameters that has to be taken under consideration to get the best output
* We learnt how WEKA uses the data for datamining and how it shows different outputs while using command prompt.

References-

* (<http://archive.ics.uci.edu/ml/>)
* <http://www.cs.waikato.ac.nz/ml/weka/>.
* CS 515 Lecture notes by Dr. Liu
* <https://en.wikipedia.org/wiki/Naive_Bayes_classifier>
* <https://en.wikipedia.org/wiki/Sequential_minimal_optimization>
* <https://en.wikipedia.org/wiki/C4.5_algorithm>
* <https://en.wikipedia.org/wiki/Logistic_distribution>
* YouTube Videos about WEKA tutorials to convert datasets into ARFF format