International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 3 Issue: 3

ISSN: 2321-8169
1216 - 1220

Comparative Study of Advanced Classification Methods

Shruti A[1], B. I. Khodanpur[2]
[1] PG Student, [2]Prof.
CSE Dept., RNSIT, Bangalore, India.

Abstract— The availability of huge amounts of data leads to the need for powerful data analysis tools to extract useful knowledge. Several data mining tools exist to improve data analysis on large data sets. There are a number of data mining tools namely Classification and Regression, Association Rules, Cluster Analysis and Outlier Analysis. Classification in data mining is a form of data analysis that extracts model using a training set, whose class label is known. This model is used as a classifier and is used for predicting the class label of unknown data set. This method of working is known as supervised learning. Various types of classifiers are Support Vector Machine, Bayesian Classification, Decision Tree Induction, Artificial Neural Network, K-Nearest Neighbor and Genetic Algorithms etc. Support Vector Machine (SVM) belongs to the class of supervised learning algorithms, SVMs construct a hyperplane or set of hyperplanes in higher dimensional space that separates two classes. Naive Bayes Classifiers (NBC) are statistical classifiers and are based on Bayes Theorem. They can predict the class membership using probabilities. In this paper study of Support Vector Machine and Naive Bayes Classifier are carried out for various training sets and their efficiency for the unknown set are analyzed for Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity results are documented. Most of the results correlated with the literature.

Keywords *—SVM, NBC, AUC*

*****

I. Introduction

Across a wide range of areas, data are being collected and accumulated at a considerable pace. There is a critical need for a new generation of data mining tools to assist humans in extracting useful knowledge from the rapidly growing volumes of data. These tools are the subject of the emerging field of knowledge discovery in databases (KDD). Data mining is the exploration of large datasets to extract hidden and previously unknown patterns. Data mining is rapidly growing as a successful tool in a wide range of applications such as healthcare and weather forecasting.

Several data mining tools are applied to improve data analysis on large data sets. There are a number of data mining tools namely Classification and Regression, Association Rules, Cluster Analysis and Outlier Analysis. The idea of classification is to place an object into one of the class, based on the model created by a training set.

The basic idea of classification is as follows: First, Choose the classification method, like Support Vector Machine and Naive Bayes Classifier. Second, a sample of training data is needed, whose class labels are known. The training set is given to a learning algorithm, which derives a classifier. Then the classifier is tested with the test set, whose class labels are unknown. If the classifier classifies most of the cases in the test set correctly, then it works accurately on the future data. On the other hand, if the classifier makes too many errors (misclassifications) in the test data, then it's a wrong model. For e.g., To predict a particular disease of a patient classifiers are used, which creates a model using the already available data from various patients of a particular disease. Using the model we can predict that a patient has a particular disease or not.

Support Vector Machine and Naive Bayes Classifier are studied for various training sets and their efficiency for unknown set are analyzed for Accuracy, AUC [10], Error Rate, F-measure, Precision, Recall and Specificity [11].

A. Support Vector Machine

SVMs have been successfully developed and have become powerful tools for solving data mining problems such as classification, regression and feature selection. In classification problems, SVMs determine an optimal separating hyperplane that classifies data points into different classes. Given a training set, each set is marked as belonging to one of the two classes. SVM algorithm builds a model using the training set whose class label is known and constructs the hyperplane which separates the training set based on the class label. Test set are then applied to the model already built and predict the class label based on hyperplane.

Linear Support Vector Machine

SVM [4] [5] belongs to the class of supervised learning algorithms in which the learning machine is given a training set with the associated class labels. SVMs construct a hyperplane that separates two classes. SVM algorithm tries to achieve maximum separation between the classes, Separating the classes with wide margin minimizes the generalization error. When the test set is applied to the model the chance of making error in the prediction should be the minimum.

Support vector machine are based on class of hyperplanes

$$w^T \times x - \gamma = 0 \qquad w \text{ € } R^n \qquad (1)$$

i.e., $w = (w_1, w_2, \ldots., w_n)^T \qquad \gamma \text{ € } R$

The corresponding decision function is given by

$$f(x) = \text{sign}\ (w^T \times x - \gamma) \qquad (2)$$

Here $x= (x_1,x_2,\ldots,x_n)^T$ a vector in the n-dimensional real space R^n. In two variable problems, decision plane is given by $w_1 \times x_1 + w_2 \times x_2 - \gamma = 0$

We build a model using a training set, the attributes of training set are represented by m x n matrix A and the class label are represented by m x m diagonal matrix D with the class label along the diagonal according to membership of each point in the class A+ or A-. The decision boundary of linear classifier is given by (1) that bisects the training examples into their respective classes. Test set are then applied to the model, it predicts the class label of test set by using (2).

Non-Linear Support Vector Machine

If the training set are not linearly separable. The trick here is to transform the data from its original input space x into a higher dimensional feature space Φ (x). The new mapping is then linearly separable.
Here (x_1, x_2) maps to $[(x_1^2, x_2^2, \sqrt{2}x_1^2x_2^2)]$

B. Naive Bayes Classifier

A Naive Bayes Classifier [6] [7] [8] is a simple probabilistic classifier based on applying Bayes theorem where every feature is assumed to be class-conditionally independent. The terms generally used in Bayes theorem are prior probability and posterior probability. The prior probability of a hypothesis or event is the original probability obtained before any additional information is obtained. The posterior probability is the revised probability of the hypothesis using some additional information or evidence obtained.
Bayes' Theorem can be written as:

$$P(A|B) = P(B|A) \times P(A)/P(B) \qquad (3)$$

Where, P (A) is the prior probability of A
P (B) is the prior probability of B
P (A|B) is the posterior probability of A given B
P (B|A) is the posterior probability of B given A

Naive Bayesian Classifier estimates the class-conditional probability by assuming that the attributes are conditionally independent, given the class label c. The conditional independence assumption can be formally stated as follows:

$$P(A|C=c) = \prod_{i=1}^{n} P(Ai|C=c) \qquad (4)$$

Where each attribute set A = {A1, A2,…, An} consists of n attribute values. With the conditional independence assumption, instead of evaluating the class conditional probability for every grouping of A, only estimate the conditional probability of each A_i, given C. To classify a test set who class label is unknown, the Naive Bayes Classifier computes the posterior probability of each class of c.

$$P(C|A) = \frac{P(C) * \prod_{i=1}^{n} P(Ai|C)}{P(A)} \qquad (5)$$

Since P(A) is fixed for every A, it is sufficient to choose the class that maximizes the numerator term,

$$P(C) * \prod_{i=1}^{n} P(Ai|C) \qquad (6)$$

The prior probability of each class is calculated by dividing the number of data instances with that class in the training set by the total number of instances in the training set.
The conditional probabilities for each feature value in the test data are calculated by getting the count of instances with that feature value in a particular class and dividing it by the count of instances with the same class in the training set. This is done for each class in the data set.
The posterior probability for each class given the feature values in the test data are calculated by using the naive Bayes classifier formula in (6) on the prior probability and conditional probability values.

II. SYSTEM ARCHITECTURE

A. STRUCTURE CHART

Structure Chart (SC) shows the breakdown of a system to its lowest manageable levels. It depicts the size and complexity of the system, and the number of readily identifiable functions and modules within each function and whether each identifiable function is a manageable entity or should be broken down into smaller components.

Fig. 1 shows as follows

1. Build the model: Build the model for SVM using the training set whose class is known and constructs the hyperplane. The hyperplane separates the two classes.
2. Test the model: Test the model using test set whose class label is unknown and predict the class label of test set based on the hyperplane.
3. Evaluation of the model: Evaluate the model based on Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity.
4. Build the model: Build the model for NBC using the training set whose classification is known. Compute prior probabilities for training set.
5. Test the model: Test the model using the test set whose classification is unknown. Compute conditional probabilities for feature values in the test set. Compute posterior probabilities for each class
6. Evaluation of the model: Evaluate the model based on Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity.
7. Compare SVM with NBC based on Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity.

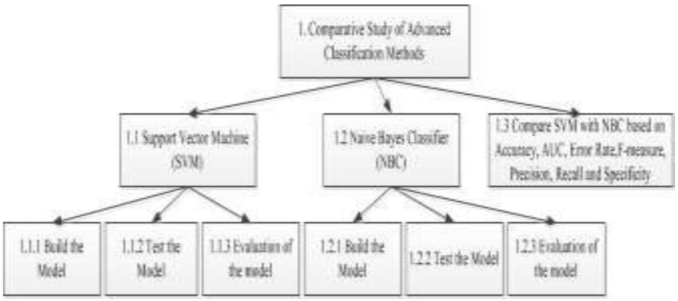

1. Comparative Study of Advanced Classification Methods
1.1 Support Vector Machine (SVM)
1.2 Naive Bayes Classifier (NBC)
1.3 Compare SVM with NBC based on Accuracy, AUC, Error Rate,F-measure, Precision, Recall and Specificity
1.1.1 Build the Model
1.1.2 Test the Model
1.1.3 Evaluation of the model
1.2.1 Build the Model
1.2.2 Test the Model
1.2.3 Evaluation of the model

Fig 1. Structure Chart

B. FLOWCHART

Fig.2 shows the flowchart of our system. Flowchart helps visualize what is going on and thereby helps the viewer to understand a process.

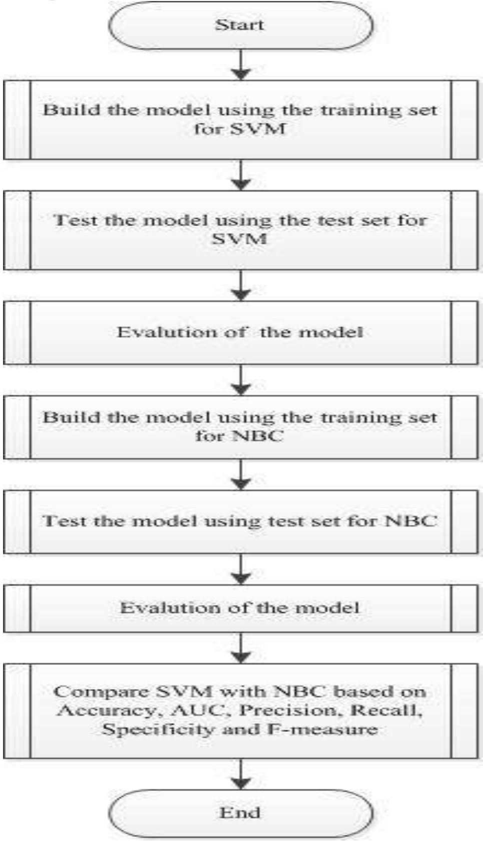

Start
Build the model using the training set for SVM
Test the model using the test set for SVM
Evalution of the model
Build the model using the training set for NBC
Test the model using test set for NBC
Evalution of the model
Compare SVM with NBC based on Accuracy, AUC, Precision, Recall, Specificity and F-measure
End

Fig 2. Flowchart

1. Build the model using training for SVM

- Input : Training Set.
- Output : Weight vector and bias.
- Description: Calculates the weight vector and bias.

2.Test the model using test set for SVM

- Input: Test Set.
- Output: Predicts class label.
- Description: Predicts the class label of the test set based on the model already built.

3.Evaluation of the model

- Input: Actual and Predicted class label
- Output: Computes Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity.
- Description: Computes Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity of the classifier.

4. Build the model using training set for NBC

- Input: Training Set.
- Output: Prior probabilities
- Description: Computes the prior probabilities of the training set.

5. Test the model using test set for NBC

- Input: Test Set
- Output: Predicts the class label of test set
- Description: Computes conditional probabilities and posterior probabilities of each class. If the posterior probability of one class is greater than other class then class label is assigned to class whose posterior probability is high.

6. Evaluation of the model

- Input: Actual and Predicted class label
- Output: Computes Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity.
- Description: Computes Accuracy, AUC,Error Rate, F-measure, Precision, Recall and Specificity of the classifier.

III. IMPLEMENTATION

Implementation of Support Vector Machine is done using MATLAB (Matrix Laboratory) programming and Naive Bayes Classifier is done using C# programming.

SVM was implemented in MATLAB. It is a high-level language and interactive environment for numerical computation, visualization, programming, analyze data, develop algorithms, create models and applications. The language and built-in math functions enable us to explore multiple approaches and reach a solution faster. Support Vector Machine has been developed using MATLAB 7.5(R2007b).

NBC was implemented in C# using Microsoft Visual Studio. Microsoft Visual Studio (VS) is an Integrated Development Environment (IDE).The .NET framework is a software framework that provides a large library and language interoperability between several programming languages. C# is an object oriented programming language designed to be fully compatible with the Microsoft .NET framework. The naive Bayes classifier has been developed using Visual Studio 2012, .Net Framework 4.5, and C#.

IV. EXPERIMENTAL RESULTS

Problem Definition: The problem is to implement the advanced classification methods i.e., Support Vector Machine and Naive Bayes Classifier. Compare both the classifiers based on the following evaluation criteria Accuracy, AUC, Error Rate, F-measure, Precision, Recall and Specificity and suggest which classifier works well.

Experimental results have been carried out on two data sets. Experimental results show the comparison of Support Vector Machine with Naive Bayes Classifier based on Accuracy, AUC, Error Rate, F-measure, Precision, Recall, Specificity and the results are tabulated in Table.1 and Table.2

A. Description of Haberman's Survival data set

Haberman data set [2] contains cases from the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer. The task is to determine if the patient survived 5 years or longer (positive) or if the patient died within 5 year (negative).

@relation Haberman
@attribute Age integer [30, 83]
@attribute Year integer [58, 69]
@attribute Positive integer [0, 52]
@attribute Survival {positive (1), negative (-1)}
@inputs Age, Year, Positive
@outputs Survival

Table.1. Results of Haberman Data Set

Haberman Data Set	Support Vector Machine	Naive Bayes Classifier
Accuracy	77.124183%	76.4706%
AUC	0.517792	0.5202
Error Rate	22.875817%	23.5294%
F-measure	31.372549%	35.7143%
Precision	50%	47.6191%
Recall	22.857143%	28.5714%
Specificity	93.220339%	90.678%

B. Description of Titanic dataset

The titanic dataset [1] gives the values of four attributes. The attributes are social class (first class, second class, third class, and crew member), age (adult or child), sex, and whether or not the person survived.

@relation titanic
@attribute Class real [-1.87, 0.965]
@attribute Age real [-0.228, 4.38]
@attribute Sex real [-1.92, 0.521]
@attribute Survived {-1.0, 1.0}
@inputs Class, Age, Sex
@outputs Survived

Table.2. Results of Titanic Data Set

Titanic Data Set	Support Vector Machine	Naive Bayes Classifier
Accuracy	77.727273%	74%
AUC	0.426392	0.4782
Error Rate	22.272727%	26%
F-measure	55.596203%	57.9412%
Precision	70.642201%	57.1015%
Recall	45.970149%	58.806%
Specificity	91.633987%	80.6536%

Accuracy: percentage of test tuples that are correctly classified by the classifier.

AUC: is the measure for evaluating the predictive ability of learning algorithms.

Error Rate: percentage of test tuples that are incorrectly classified by the classifier.

F-measure: is a measure of test's accuracy. It considers both precision and recall to compute the score.

Precision: is the proportion of the true positives against all the positive results.

Recall: is the proportion of positive tuples that are correctly identified.

Specificity: is the proportion of negative tuples that are correctly identified.

C. Confusion Matrix of SVM and NBC

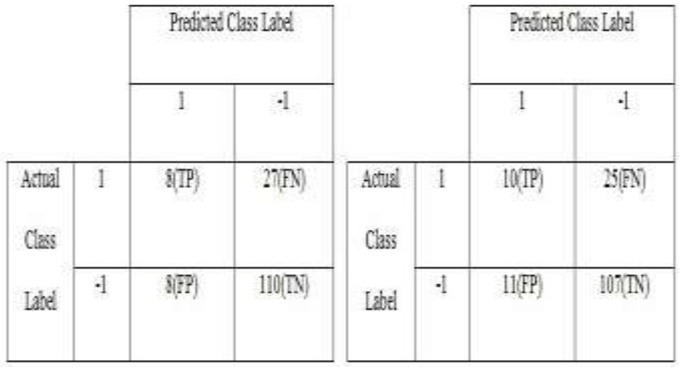
Confusion Matrix [6] of SVM and NBC for Haberman Data Set and Confusion Matrix of SVM and NBC for Titanic Data Set are shown in Fig. 3.

TP (True Positive): These refer to the positive tuples that were correctly labeled by the classifier.

TN (True Negative): These are negative tuples that were correctly labeled by the classifier.

FP (False Positive): These are the negative tuples that were incorrectly labeled as positive.

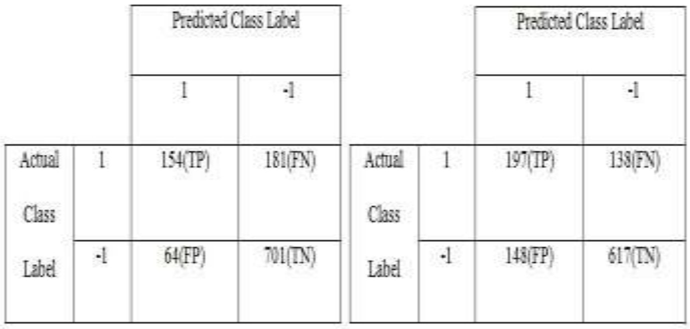
FN (False Negative): These are positive tuples that were mislabeled as negative.



		Predicted Class Label	
		1	-1
Actual Class Label	1	8(TP)	27(FN)
	-1	8(FP)	110(TN)

		Predicted Class Label	
		1	-1
Actual Class Label	1	10(TP)	25(FN)
	-1	11(FP)	107(TN)

Fig. a. Confusion Matrix of SVM and NBC for Haberman Data Set



		Predicted Class Label	
		1	-1
Actual Class Label	1	154(TP)	181(FN)
	-1	64(FP)	701(TN)

		Predicted Class Label	
		1	-1
Actual Class Label	1	197(TP)	138(FN)
	-1	148(FP)	617(TN)

Fig. b. Confusion Matrix of SVM and NBC for Titanic Data Set

Fig 3. Confusion Matrix

V. CONCLUSION

In this paper, various data sets have been used for training as well as for testing. We designed and developed Support Vector Machine and Naive Bayes Classifier that is generalized to read any data set with various attributes and a prescribed structure in the input excel file. The classifier is tested using two different data sets and the evaluation of the classifier is carried out by calculating Accuracy, AUC, Error Rate, F-Measure, Precision, Recall and Specificity.

REFERENCES

[1] KEEL (Knowledge Extraction based on Evolutionary Learning) http://sci2s.ugr.es/keel/datasets.php

[2] UCI (University of California, Irvine) machine learning repository http://archive.ics.uci.edu/ml.

[3] Glen Fung and Olvi Mangasarian, Proximal Support Vector Machine Classifiers, proceedings, KDD 2001

[4] Olvi Mangasarian, Data Mining with support vector machines, Data Mining Institute Report 01-05, University of Wisconsin, Madison. May 2001

[5] K. P. Soman, Insight into Data Mining Theory and Practice, New Delhi: PHI, 2006

[6] J. Han and M. Kamber, Data Mining Concepts and Techniques, Elsevier, 2011

[7] Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Introduction to Data Mining, Pearson Education, 2012

[8] V Susheela Devi, M Narasimha Murty, Pattern Recognition an Introduction, Universities Press, 2011

[9] Sergios Theodoridis, Konstantinos Koutroumbas, Pattern Recognition, Elsevier Academic Press, 2003.

[10] Jin Huang, Charles X.Ling, Using AUC and Accuracy in Evaluating Learning Algorithms, Department of Computer Science, The University of Western Ontario, London, Ontario, Canada, December 2,2003

[11] Ahmad Ashari, Iman Paryudi, A Min Tjoa, Performance Comparison between Naive Bayes, Decision Tree and k-Nearest Neighbor in Searching Alternative Design in an Energy Simulation Tool, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 4, No. 11, 2013.