Volume 3, No. 5, May 2012

Journal of Global Research in Computer Science

ISSN-2229-371X

RESEARCH PAPER

Available Online at www.jgrcs.info

CLASSIFICATION AND COMPARATIVE STUDY OF DATA MINING CLASSIFIERS WITH FEATURE SELECTION ON BINOMIAL DATA SET

Pushpalata Pujari[1]
Department of Computer Science & Information Technology,
Guru Ghasi Das Central University, Bilaspur, Chhattisgarh, India
pujari.lata@rediffmail.com

Abstract*:* This paper describes about the performance analysis of different data mining classifiers before and after feature selection on binomial data set. Three data mining classifiers Logistic Regression, SVM and Neural Network classifiers are considered in this paper for classification. The Congressional Voting Records data set is a binomial data set investigated in this study is taken from UCI machine learning repository. The classification performance of all classifiers is presented by using statistical performance measures like accuracy, specificity and sensitivity. Gain chart and R.O.C (Receiver Operating Characteristics) chart are also used to measure the performances of the classifiers. A comparative study is carried out among the data mining classifiers. Experimental result showed that without feature selection Logistic Regression and SVM classifiers provides 100% accuracy and neural network provides 98.13 % accuracy on test data set. With feature selection SVM classifier provides 100% accuracy. The performance of SVM classifier is found to be the best among all the classifiers with reduced number of features.

Keywords: Data mining, Logistic regression, SVM, neural network, feature selection, Gain chart, R.O.C chart

INTRODUCTION

Classification [1] is a two step process. In the first step, a classifier is built describing a predetermined set of data classes or concepts. This is the learning step or training phase, where a classification algorithm builds the classifiers by analyzing or “learning from” a training set made up of database tuples and their associated class levels. A tuple, X, is represented by an n-dimensional attribute vector, X = ($x_{1,}$ $x_{2\dots}$ x $_{n}$)$_{,}$ depicting n measurements made on the tuple from n database attributes, respectively $A_{1,}$ $A_{2\dots}$ $A_{n.}$ Each, tuple, X, is assumed to belong to a predefined class as determined by another database attribute called the class label attribute.

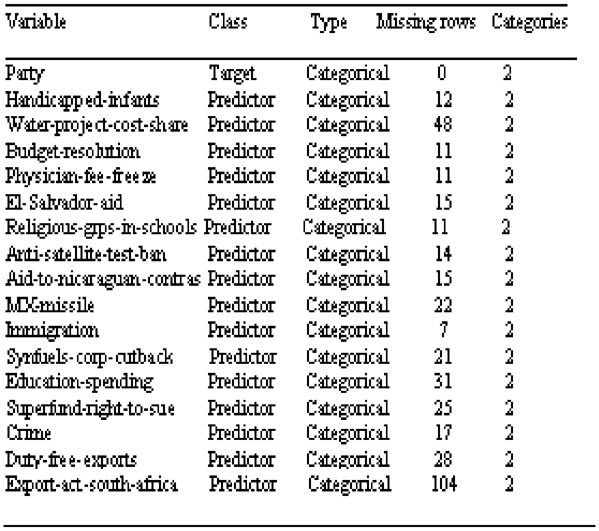
The class level attribute is discrete-valued and unordered. It is categorical in that each value serves as a category or class. The individual tuples making up the training set are referred to as training tuples and are selected from the database under analysis. As the class label of each training tuple is provided, this step is known as supervised learning. The first step of the classification process can be viewed as the learning of a mapping or function, y= *f*(X), that can predict the associated class level y of a given tuple X. This mapping is represented in the form of classification rules, decision trees, or mathematical formula. The rules can be used to categorize future data tuples. In the second step the model is used for classification. A test set is used to test tuples and their associated class labels. These tuples are randomly selected from the general data set. The accuracy of a classifier on a given test set is the percentage of test set tuples that are correctly classified by the classifier.

The associated class level of each test tuple is compared with the learned classifier’s class prediction for that tuple. If the accuracy of the classifier is considered acceptable, the classifier can be used to classify future data tuples for which the class label is not known. In this paper different classification techniques of data mining such as Logistic regression, Neural Network and SVM are analyzed on house vote dataset. A comparative study is carried out among the classification algorithms for the prediction of republic or Democratic Party from U.S house of representative votes. The performance of individual models is evaluated by using different statistical measures including classification accuracy, specificity and sensitivity. Each sample of the dataset is classified into two categories: republic or Democratic Party.

DATA SET DESCRIPTION

The Congressional Voting Records data set used in this study is taken from UCI machine learning dataset [7]. This data set includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes identified by the CQA. The CQA lists nine different types of votes: voted for, paired for, and announced for, voted against, paired against, and announced against, voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known. Each sample of the dataset is classified into two categories: democrat and republican. The dataset contains 17 number of attributes out of which first attributes is taken as target output and rest of the attributes are taken as input attribute. Table I represents the attributes of house vote data set.

Table 1: Attributes of house vote dataset



Variable	Class	Type	Missing rows	Categories
Party	Target	Categorical	0	2
Handicapped-infants	Predictor	Categorical	12	2
Water-project-cost-share	Predictor	Categorical	48	2
Budget-resolution	Predictor	Categorical	11	2
Physician-fee-freeze	Predictor	Categorical	11	2
El-Salvador-aid	Predictor	Categorical	15	2
Religious-grps-in-schools	Predictor	Categorical	11	2
Anti-satellite-test-ban	Predictor	Categorical	14	2
Aid-to-nicaraguan-contras	Predictor	Categorical	15	2
MX-missile	Predictor	Categorical	22	2
Immigration	Predictor	Categorical	7	2
Synfuels-corp-cutback	Predictor	Categorical	21	2
Education-spending	Predictor	Categorical	31	2
Superfund-right-to-sue	Predictor	Categorical	25	2
Crime	Predictor	Categorical	17	2
Duty-free-exports	Predictor	Categorical	28	2
Export-act-south-africa	Predictor	Categorical	104	2

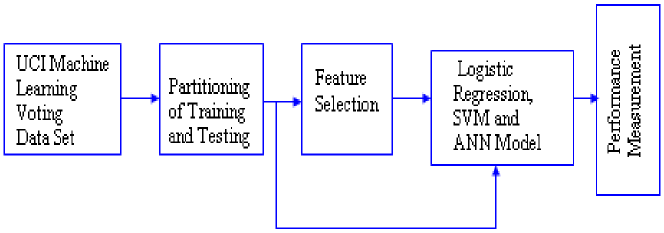
Models [2] are developed in two phases: training and testing Training refers to building a new model by using historical data, and testing refers to trying out the model on new, previously unseen data to determine its accuracy and physical performance characteristics. Training is typically done on a large proportion of the total data available, where as testing is done on some small percentage of the data. The training dataset is used to train or build a model. Once a model is built on training data, the accuracy of the model on unseen data (testing) can be found. Two mutually exclusive datasets, a training dataset comprising 80% of the total dermatology dataset, and a testing dataset of 20% is created by using partitioning node and balanced node portioning techniques. Classification techniques are applied on this data. In all there are 435 numbers of instances in house vote dataset out of which 335 instances are taken as training set and 100 instances are taken as testing set by using balanced node concept of Clementine data mining tool. Out of 267 democrat class 199 instances and 68 instances are taken for training and testing respectively. Out of 168 democrat class 129 instances and 39 instances are taken for training and testing respectively. Table-II shows the number of instances taken for training and testing data set.

Table 2 Instances of training and testing dataset.

Class	***Training***	***Testing***	***Total***
Democrat Republican	199 129	68 39	267 168
Total	328	107	435

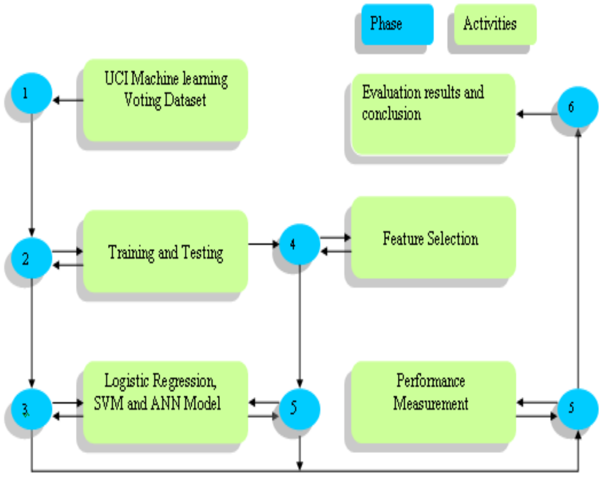
METHODOLOGY

Different data mining classifiers are used to meet the objective of this piece of research work is explored herewith. Mainly Logistic regression, Neural Network and SVM based classification algorithm is considered to classify the house vote data. The house vote data set is partitioned into 80 % of training dataset and 20% of testing data set. The data set is applied for the three classifiers to build models. Feature selection technique is carried out to skip unimportant attributes from the data set. After skipping unimportant attributes the data set is applied to the three classifiers. A comparative analysis is carried out on the performances of classifiers before and after feature selection. Fig.1 shows the block diagram of the proposed model. Fig. 2 shows the phases and activities of the model.



UCI Machine Learning Voting Data Set
Partitioning of Training and Testing
Feature Selection
Logistic Regression, SVM and ANN Model
Performance Measurement

Figure.1. Block diagram of the proposed model



Phase
Activities
1
UCI Machine learning Voting Dataset
Evaluation results and conclusion
6
2
Training and Testing
4
Feature Selection
3
Logistic Regression, SVM and ANN Model
5
Performance Measurement
5

Figure.2. Phases of the proposed model

Neural Network:

Neural networks [1] [6] are simple models of the way the nervous system operates. The basic units are neurons, which are typically organized into layers. There are typically three parts in a neural network: an input layer, with units representing the input fields; one or more hidden layers; and an output layer, with a unit or units representing the output field(s). Each layer is made up of units. The inputs to the network correspond to the attributes measured for each training tuple. The inputs are fed simultaneously into the units making up the input layer. These inputs pass through the input layer and are then weighted and fed simultaneously to a second layer of "neuron like" units, known as a hidden layer. The units are connected with varying connection strengths (or weights). Input data are presented to the first layer, and values are propagated from each neuron to every neuron in the next layer. Eventually, a result is delivered from the output layer, which emits the network's prediction for given tuples. The network learns by examining individual records, generating a prediction for each record, and making adjustments to the weights whenever it makes an incorrect prediction. This process is repeated many times, and the network continues to improve its predictions until one or more of the stopping criteria have been met. Initially, all weights are random, and the answers that come out of the net are probably nonsensical.

The network learns through training. Examples for which the output is known are repeatedly presented to the network, and the answers it gives are compared to the known outcomes. Information from this comparison is passed back through the network, gradually changing the weights. As training progresses, the network becomes increasingly accurate in replicating the known outcomes. Once trained, the network can be applied to future cases where the outcome is unknown. Fig. 3 shows the architecture of neural network. In this research work ANN architecture of 34 X 34 X 2 is created and trained with Error back propagation algorithm (EBPA).

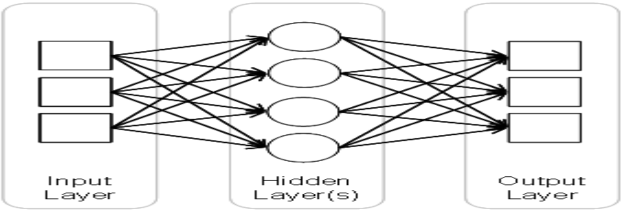


Figure.3. Architecture of a neural network

SVM (Support Vector Machine):

Support Vector Machine [3] (SVM) is a robust classification and regression technique that maximizes the predictive accuracy of a model without over fitting the training data. SVM is particularly suited to analyzing data with very large numbers (for example, thousands) of predictor fields. SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, and then the data are transformed in such a way that the separator could be drawn as a hyper plane. Following this, characteristics of new data can be used to predict the group to which a new record should belong. The Support Vector Machines (SVM) are a general class of learning architectures, inspired by the statistical learning theory that performs structural risk minimization on a nested set structure of separating hyper planes. Given a training data, the SVM learning techniques generates the optimal separating hyper plane in terms of generalization error.

The support vector machine is very popular as a high-performance classifier in several domains in classification. It obtains a set of support vector which characterizes a given classification task. The basic idea is to construct a hyper plane as the decision surface such that the margin of separation between positive and negative examples is maximized. The structural risk minimization principle is used for this purpose. Here the error rate of a learning machine is considered to be bounded by the sum of the training error rate and a term depending on the Vapnik Chervonenkis (VC) 1 dimension. Given a labeled set of N training samples (X_i, Y_i), where $X_i \in R^n$ and $Y_i \in \{-1, 1\}$, the discriminate hyper plane is defined as

$$f(X_q)= \Sigma Y_i \propto_i K(X_q, X_i)+b$$

Here K (.) is a kernel function and the sign of $f(X_q)$ determines the membership of query sample X_q .Constructing an optimal hyper plane is equivalent to determining all nonzero $\propto_{i's}$, which corresponds to the support vectors, and the bias b. The expected loss of making decision is the minimum.

Logistic Regression:

Logistic regression, [6] also known as nominal regression, is a statistical technique for classifying records based on values of input fields. It is analogous to linear regression but takes a categorical target field instead of a numeric one. Both binomial models (for targets with two discrete categories) and multinomial models (for targets with more than two categories) are supported. Logistic regression works by building a set of equations that relate the input field values to the probabilities associated with each of the output field categories. Once the model is generated, it can be used to estimate probabilities for new data. For each record, a probability of membership is computed for each possible output category. The target category with the highest probability is assigned as the predicted output value for that record. Logistic regression models are often quite accurate. They can handle symbolic and numeric input fields. They can give predicted probabilities for all target categories. Logistic models are most effective when group membership is a truly categorical field

Feature Selection:

Feature selection [1] [6] helps to identify the fields that are most important in predicting a certain outcome. Feature selection is a process that can be used to remove terms in the training documents that are statistically uncorrelated with class labels. It reduces the set of terms to be used in classification, improving both efficiency and accuracy. Feature selection consists of three steps. Screening: It removes unimportant and problematic predictors and records or cases, such as predictors with too many missing values or predictors with too much or too little variation to be useful. Ranking: Sorts remaining predictors and assigns ranks based on importance. Selecting: It identifies the subset of features by preserving only the most important predictors and filtering or excluding all others. From a set of hundreds or even thousands of predictors, the Feature Selection screens, ranks, and selects the predictors that are most important.

The predictors which contribute less in prediction can be skipped from the data set. Ultimately, it ends up with a quicker, more efficient model that uses fewer predictors, executes more quickly, and may be easier to understand. In this piece of research work importance of attributes are ranked based on Pearson chi-square measure. The unimportant features are skipped and the performances are compared against the performances of the classifiers before feature selection. Table III Shows the list of important and unimportant attribute after carrying out feature selection technique. The unimportant attributes are skipped from the data set as they do not contribute much more in prediction. The Congressional Voting Records data set contains sixteen input attributes and one target attribute.

Out of the sixteen attribute fourteen attributes are found to be important attributes with value 1.0 and two attributes Export-act-south-africa and Water-project-cost-share are found to be unimportant attributes with values 0.783 , 0.104 respectively , hence skipped from the data set.

Table 3. Importance of attributes with values

Rank	*Attribute*	*Importance*	*Value*
1	Handicapped-infants	Important	1.0
2	Budget-resolution	Important	1.0
3	Physician-fee-freeze	Important	1.0
4	El-salvador-aid	Important	1.0
5	Religious-grps-in-schools	Important	1.0
6	Anti-satellite-test-ban	Important	1.0
7	Aid-to-nicaraguan-contras	Important	1.0
8	Immigration	Important	1.0
9	MX-missile	Important	1.0
10	Synfuels-corp-cutback	Important	1.0
11	Education-spending	Important	1.0
12	Superfund-right-to-sue	Important	1.0
13	Crime	Important	1.0
14	Duty-free-exports	Important	1.0
15	Export-act-south-africa	Unimportant	0.783
16	Water-project-cost-share	Unimportant	0.104

PERFORMANCE MEASUREMENT

Performance of each classifier can be evaluated by using some very well known statistical measures[4] classification accuracy, sensitivity and specificity. These measures are defined by true positive (TP), true negative (TN), false positive (FP) and false negative (FN). Say we test some people for the presense of a disease. Some of these people have the disease, and our test says they are positive. They are called true positives. Some have the disease, but the test claims they don't. They are called false negatives. Some don't have the disease, and the test says they don't - true negatives. Finally, we might have healthy people who have a positive test result false positives. Table-IV represents a matrix showing number of TP, TN, FP, and FN cases.

Table 4. Matrix for Actual and Predicted cases

	P'(predicted)	*N'(predicted)*
P(Actual)	True Positive(TP)	False Negative(FN)
N(Actual)	False Positive (FP)	True Negative(TN)

If the total number of cases is N then based on the above table following statistical performance measures can be evaluated.

Classification Accuracy:

It measures the proportion of correct predictions considering the positive and negative inputs. It is highly dependant of the data set distribution which can easily lead to wrong conclusions about the system performance. It is calculated as follows

Classification accuracy = Total hits / Number of entries in the set

$$= (TP + TN) / (P + N) \qquad \ldots (1)$$

Classification Sensitivity:

It measures the proportion of the true positives, that is, the ability of the system on predicting the correct values in the cases presented. It is calculated using the following formula.

Sensitivity = Positive hits / Total positives

$$= TP/ (TP+FN) \qquad \ldots (2)$$

Classification Specificity:

It measures the proportion of the true negatives, that is, the ability of the system on predicting the correct values for the cases that are the opposite of the desired one. It is calculated as follows

Specificity = Negative hits / Total

$$= TN / (TN+FP) \qquad \ldots (3)$$

EXPERIMENTAL RESULTS AND DISCUSSION

First of all the performance of each classifier is analyzed by using all the input attributes. Then feature selection technique is applied to the data set and unimportant attributes are skipped from the data set. Again the performance of each classifier is analyzed with reduced number of attributes. The experimental study is carried out by using Clementine software. After applying training data and testing data set to each classifier a confusion matrix is obtained to identify true positive, true negative, false positive, and false negative values as follows. Table V & VI shows the confusion matrices for training and testing data set before and after feature selection. Table VII & VIII shows comparative statistical measures of different models for training and testing data set before and after feature selection.

Each cell of the table V and VI below contains the row number of samples classified for the corresponding combination of desired and actual model output. The prediction are compared with original classes to identify true positive, true negative, false positive and false negative. Table VII & VIII represents the value of three statistical measures classification accuracy, sensitivity and specificity of the three models.

Table 5 Confusion matrices of different model for training and testing data set before feature selection

		Training data		*Testing data*	
Models	**Desired output**	**Democrat**	**Republican**	**Democrat**	**Republican**
Logistic Regression	Democrat	199	0	68	0
	Republican	0	129	0	39
SVM	Democrat	198	1	68	0
	Republican	0	129	0	39
Neural Network	Democrat	197	2	66	2
	Republican	2	127	0	39

Table 6. Confusion matrices of different model for training and testing data set after feature selection

		Training data		*Testing data*	
Models	**Desired output**	**Democrat**	**Republican**	**Democrat**	**Republican**
Logistic Regression	Democrat	196	3	66	2
	Republican	3	126	0	39
SVM	Democrat	198	1	68	0
	Republican	0	129	0	39
Neural Network	Democrat	193	6	67	1
	Republican	4	125	0	39

Table 7: Comparative statistical measures of different models for training and test data set before feature selection

Measures %				
Model	**Partition**	**Accuracy**	**Sensitivity**	**Specificity**
Logistic Regression	Training	100	100	100
	Test	100	100	100
SVM	Training	99.7	99.49	100
	Test	100	100	100
Neural Network	Training	98. 78	98..99	98.44
	Test	98.13	97.05	100

Table 8: Comparative statistical measures of different models for training and testing data set after feature selection

Measures %				
Model	**Partition**	**Accuracy**	**Sensitivity**	**Specificity**
Logistic Regression	Training	98.17	98.49	97.67
	Test	98.13	97.05	100
SVM	Training	99.7	99.49	100
	Test	100	100	100
Neural Network	Training	96.95	96.98	96.89
	Test	99.07	98.52	100

Another way to compare the performance of different classifier is gain chart and ROC (Receiver Operating Characteristics) [14].The gains chart [6] plots the values in the Gains % column from the table. Gains are defined as the proportion of hits in each increment relative to the total number of hits in the tree, using the equation:

(Hits in increment / total number of hits) x 100% ... (4)

Cumulative gains charts always start at 0% and end at 100% as we go from left to right. For a good model, the gains chart will rise steeply toward 100% and then level off. A model that provides no information will follow the diagonal from lower left to upper right the steeper the curve, the higher the gain. Fig 5 shows the cumulative gain chart of three models for training dataset before feature selection. Fig 6 shows the cumulative gain chart of three models for testing dataset after carrying out feature selection.

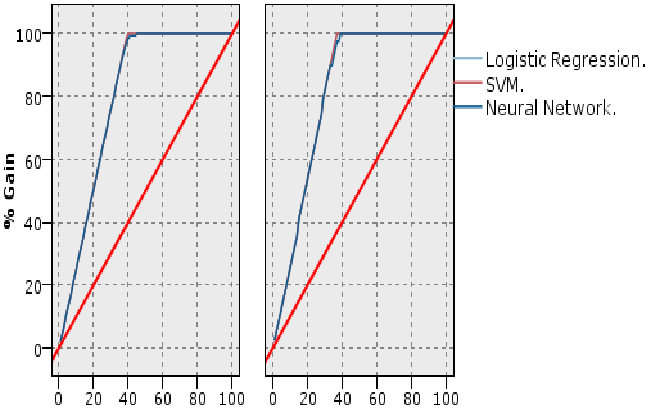


Figure.5. Gain chart for three models before feature selection

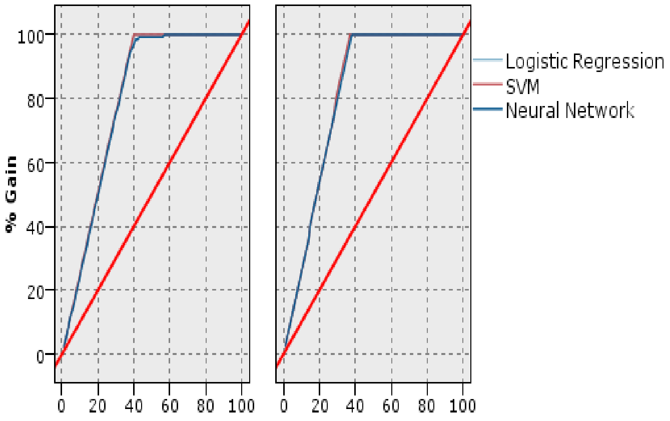


Figure.6. Gain chart for three models after feature selection

R.O.C chart [1] [6] is a useful visual tool for comparing classification methods. It shows the trade-off between the true positive rate and the false positive rate for a given model. The area under the R.O.C chart is a measure of the accuracy of the model. R.O.C chart plots the values in the Response (%) column of the table. The response is a percentage of records in the increment that are hits, using the equation:

(Responses in increment / records in increment) x 100% … (5)

ROC chart is based on the conditional probabilities sensitivity and specificity [11]. The vertical axis of an ROC curve represents the true positive rate and the horizontal axis represents the false-positive rate. It is a plot of sensitivity on the vertical axis and one minus the specificity on horizontal axis for different values of the thresholds. Response charts usually start near 100% and gradually descend until they reach the overall response rate (total hits / total records) on the right edge of the chart. For a good model, the line will start near or at 100% on the left, remain on a high plateau as you move to the right, and then trail off sharply toward the overall response rate on the right side of the chart. For a model that provides no information, the line will hover around the overall response rate for the entire graph. Fig 7 and 8 shows the ROC chart of three models for training and testing dataset before and with feature selection respectively.

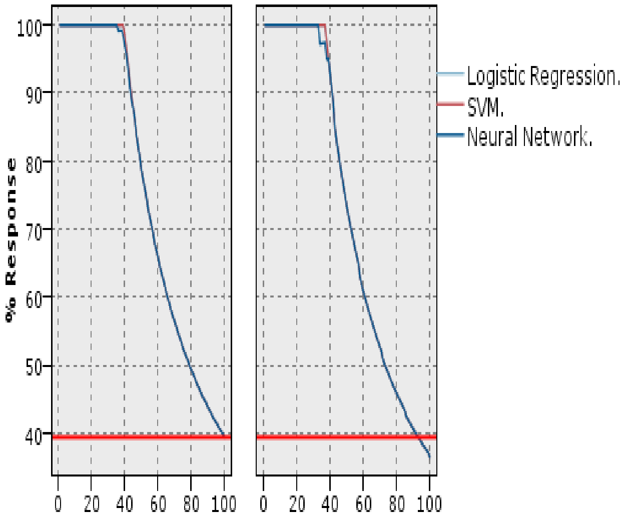


Figure.7. ROC chart for the three models before feature selection

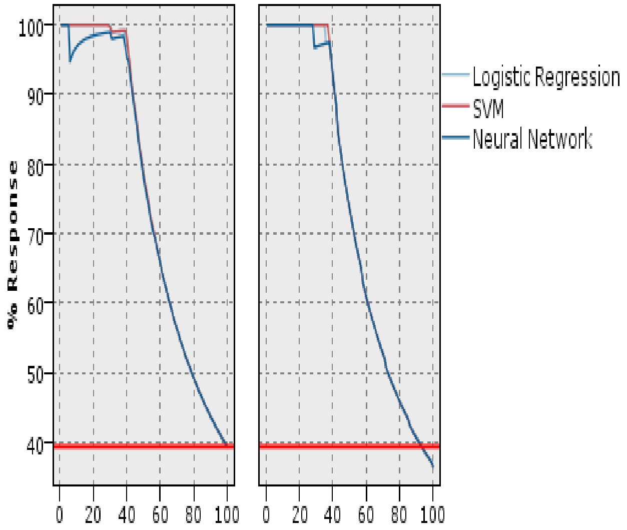


Figure.8. ROC chart for the three models after feature selection

Table 9. Accuracy of different classifiers for training data set before feature selection.

Model	***Cases***	***Number of instances***	***Accuracy (%)***
Logistic Regression	Correct Wrong	328 0	100 0
SVM	Correct Wrong	327 1	99.7 0.3
Neural Network	Correct Wrong	324 4	98.78 1.22

Table 10. Accuracy of different classifiers for test data set before feature selection.

Model	***Cases***	***Number of instances***	***Accuracy (%)***
Logistic Regression	Correct Wrong	107 0	100 0
SVM	Correct Wrong	107 0	100 0
Neural Network	Correct Wrong	105 2	98.13 1.87

Table 11. Accuracy of different classifiers for training data set after feature selection.

Model	*Cases*	*Number of instances*	*Accuracy (%)*
Logistic Regression	Correct Wrong	322 6	98.17 1.83
SVM	Correct Wrong	327 1	99.7 0.3
Neural Network	Correct Wrong	318 10	96.97 3.03

Table 12. Accuracy of different classifiers for test data set after feature selection.

Model	*Cases*	*Number of instances*	*Accuracy (%)*
Logistic Regression	Correct Wrong	105 2	98.13 1.87
SVM	Correct Wrong	107 0	100 0
Neural Network	Correct Wrong	106 1	99.07 0.93

CONCLUSION

In this paper the performance of three different classifiers Logistic regression, SVM & Neural Network is analyzed on house vote dataset. The classification performance of all algorithms is investigated by using statistical performance measures like accuracy, specificity and sensitivity. Also the performance of all classifier is investigated with the help of gain chart and ROC chart for both training and testing set. Table IX. & X shows the classification accuracy for training and testing data set for the three models before feature selection. Table XI. & XII shows the classification accuracy for training and testing data set for the three models after feature selection .From the experimental results the accuracy of Logistic regression, SVM and Neural network model is found to be 100%, 99.07% and 98.78 %respectively for training data set before feature selection. Similarly the accuracy is found to be 100%, 100% and 98.13 % respectively for test data set before feature selection. After carrying out feature selection the accuracy of Logistic regression, SVM and Neural network model is found to be 98.17%, 99.7% , 96.97% respectively for training data set and 98.13 %, 100% and 99.07% respectively for test data set. The SVM Model with feature selection has achieved a remarkable performance with accuracy of 100.00% on test data set which is a competitive technique for prediction of Republic or Democratic Party from the dataset.

REFERENCES

[1]. Jiwaei Han, Kamber Micheline, Jian Pei Data mining: Concepts and Techniques, Morgam Kaufmann Publishers (Mar 2006).

[2]. Cabena, Hadjinian, Atadler, Verhees, Zansi “Discovering data mining from concept to implementation” International Technical Support Organization, Copyright IBM corporation 1998.

[3]. S.Mitra, T. Acharya “Data Mining Multimedia, Soft computing and Bioinformatics, A john Willy & Sons, INC , Publication, 2004.

[4]. Alaa M. Elsayad “Predicting the severity of breast masses with ensemble of Bayesian classifiers” journal of computer science 6 (5): 576-584, 2010, ISSN 1549-3636

[5]. Alaa M. Elsayad “ Diagnosis of Erythemato-Squamous diseases using ensemble of data mining methods” ICGST-BIME Journal Volume 10, Issue 1, December 2010

[6]. SPSS Clementine help file. http//www.spss.com

[7]. UCI Machine Learning Repository of machine learning databases. University of California, school of Information and Computer Science, Irvine. C.A. http://www.ics.uci.edu/~mlram,?ML.Repositary.html

[8]. Michael J. A. Berry Gordon Linoff, “Data Mining Techniques “, John Wiley and Sons, Inc.

[9]. Gajendra Sharma “Data Mining, Data Warehousing and OLAP”, S.K kataria and sons New Delhi,2nd edition 2008-2009.

[10]. Harleen Kaur and Siri Krishan Wasan “Empirical Studies on applications of Data Mining” Techniques in health care., Journal of computer Science, 2(2), 194-200, 2006, ISSN 1549-3636.

[11]. Jozef Zurada and Subash Lonial “Comparison of The Performance of Several Data Mining methods for Bed Debt Recovery in The Health Care Industry”.

[12]. Matthew N Anyanwu & Sajjan G Shiva “Comparative Analysis of serial Decision Trees Classification Algorithms”,(IJCSS), Volume (3) : Issue (3)

[13]. Mahesh Pal “Ensemble Learning With Decision Tree for Remote Sensing Classification”, World Academy of Science, Engineering and Technology 36 2007.

[14]. Kelly H. Zou, PhD; A. James O’Malley, PhD; Laura Mauri, MD, M.Sc “ROC Analysis for Evaluating Diagnostic Test and Predictive Models.

Short Bio Data for the Author

Pushpalata Pujari has received her master degree in Computer Application from Berhampur University, India. Currently she is working as an assistant professor in the department of computer science and IT, Guru Ghasi Das Central University, India. Pushpalata Pujari has published several papers in national and internal conferences and in international journals. All focusing in classification, data mining and soft computing. Her current research interest involves improving classification accuracy of data mining algorithms. 400-450-1-SM.doc