Classification Of Complex UCI Datasets Using Machine Learning And Evolutionary Algorithms

Anuj Gupta

Abstract: Classification is an important data mining technique with broad applications. Classification is a gradual practice for allocating a given piece of input into any of the known category. The Data Mining refers to extracting or mining knowledge from huge volume of data. In this paper different classification techniques of Data Mining are compared using diverse datasets from University of California, Irvine (UCI) Machine Learning Repository. Accuracy and time complexity for execution by each classifier is observed. . Finally different classifiers are also compared with the help of Confusion Matrix. Classification is used to classify each item in a set of data into one of predefined set of classes or groups

Index Terms: Classification, Data Mining, Decision Table, Genetic Programming, J48, Logistic, MultilayerPerceptron, NaiveBayes, RandomForest, VFI, ZeroR, .

1 INTRODUCTION

Classification is one of the most researched questions in machine learning and data mining. In machine learning, classification refers to an algorithmic process for designating a given input data into one among the different categories given. A wide range of real problems have been stated as Classification Problems, for example credit scoring, bankruptcy prediction, medical diagnosis, pattern recognition, text categorization and many more. An algorithm that implements classification is known as a classifier. The input data can be termed as an instance and the categories are known as classes. The characteristics of the instance can be described by a vector of features. These features can be nominal, ordinal, integer-valued or real-valued. Classification is a supervised procedure that learns to classify new instances based on the knowledge learnt from a previously classified training set of instances. This work has been carried out to make a performance evaluation of Machine Learning Algorithms: J48, MultilayerPerceptron(MLP), NaiveBayes, Decision Table , Logistic , RandomForest, VFI(Voting Feature Intervals), ZeroR(Zero Rules) and Evolutionary Algorithm: GeneticProgramming (GP) . The paper sets out to make comparative evaluation of these classifiers in the context of 11 different datasets namely iris , abalone , labor Contact Lenses, Soybean , HayesRoth, LungCancer, Glass Identification, Teaching Assistant Evaluation, Vote, Statlog. Performance measures used for comparison are Accuracy, TimeComplexity, Mean Absolute Error(MAE) and Root Mean Squared Error(RMSE). The experiments are carried out using weka 3.6 of Waikato University.Weka library is imported in eclipse and then java programs for different classification algorithms are executed. For Genetic Programming weka 3.4 is used. All these classifications are performed after feature selection to improve the accuracy. After performing all the classification algorithms different conclusions and results are drawn which would be discussed later.

Contribution Of This Paper: In this paper Genetic Programming (GP) is also used for classification . Since genetic programming is an evolutionary algorithm it is better for classification than some traditional algorithms and this is what we have shown in the results.

2 CLASSIFIERS USED:

2.1 J48: J48 can be called as optimized implementation of the C4.5 or improved version of the C4.5. The output given by J48 is the Decision tree. A Decision tree is same as that of the tree structure. having different nodes, such as root node, intermediate nodes and leaf node. Each node in the tree contains a decision and that decision leads to our result as name is decision tree. Decision tree divide the input space of a data set into mutually exclusive areas, where each area having a label, a value or an action to describe or elaborate its data points. Splitting criterion is used in decision tree to calculate which attribute is the best to split that portion tree of the training data that reaches a particular node.

2.2 MultilayerPerceptron: Multilayer Perceptron can be defined as Neural Network and Artificial intelligence without qualification. A MultiLayer Perceptron (MLP) is a feedforward neural network with one or more layers between input and output layer. Basically there are three layers: input layer, hidden layer and output layer. Hidden layer may be more than one. Each neuron (node) in each layer is connected to every neuron (node) in the adjacent layers. The training or testing vectors are connected to the input layer, and further processed by the hidden and output layers.

2.3 VFI: VFI stands for voting feature intervals. Intervals are constructed around each class for each attribute (basically discretization).Class counts are recorded for each interval on each attribute. Classification is done by voting. Higher weight is assigned to more confident intervals, where confidence is a function of entropy:

Weight (att_i) = (entropy of class distrib att_i / max uncertainty) ^-bias

2.4 NaiveBayes: Abstractly, NaiveBayes is a conditional probability model: given a problem instance to be classified, represented by a vector $\mathbf{x} = (x_1, \dots, x_n)$ representing some *n* features (dependent variables), it assigns to this instance probabilities

$$p(C_k|x_1, \dots, x_n)$$

for each of *K* possible outcomes or *classes.* The problem with the above formulation is that if the number of features *n* is large or if a feature can take on a large number of values, then basing such a model on probability tables is infeasible. We therefore reformulate the model to make it more tractable. Using Bayes theorem, the conditional probability can be decomposed as

$$p(C_k|\mathbf{x}) = \frac{p(C_k)\ p(\mathbf{x}|C_k)}{p(\mathbf{x})}.$$

2.5 Logistic: Logistic classification measures the relationship between the categorical dependent variable and one or more independent variables, which are usually (but not necessarily) continuous, by using probability scores as the predicted values of the dependent variable. An explanation of logistic regression begins with an explanation of the logistic function. The logistic function is useful because it can take an input with any value from negative to positive infinity, whereas the output always takes values between zero and one and hence is interpretable as a probability. The logistic function $\sigma(t)$is defined as follows:

$$\sigma(t) = \frac{e^t}{e^t+1} = \frac{1}{1+e^{-t}},$$

If t is viewed as a linear function of an explanatory variable x(or of a linear combination of explanatory variables), then we express t as follows:

$$t = \beta_0 + \beta_1 x$$

And the logistic function can now be written as:

$$F(x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}}$$

$F(x)$ is interpreted as the probability of the dependent variable equaling a "success" or "case" rather than a failure or non-case. It's clear that the response variables Y_i are not identically distributed: $P(Y_i = 1 \mid X)$ differs from one data point $X_i$to another, though they are independent given design matrix $X$and shared with parametersβ.

2.6 RandomForest: Random forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random forests correct for decision trees' habit of overfitting to their training set.

Each tree is grown as follows:

1. If the number of cases in the training set is N, sample N cases at random - but *with replacement*, from the original data. This sample will be the training set for growing the tree.
2. If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
3. Each tree is grown to the largest extent possible. There is no pruning.

2.7 Genetic Programming: It is an evolutionary learning technique that offers a great potential for classification. The application of GP to classification offers some interesting advantages such as flexibility which allows the technique to be adapted to the needs of each particular problem. GP can be employed to construct classifiers using different kinds of representation e.g. decision trees, classification rules, discriminant functions and many more. The automatic feature selection performed by GP and different mechanisms available to controlling the size of the resulting classifiers contribute to improve interpretability. GP uses tree structure as its representation formalism. In tree structure internal nodes represent non-terminal set which includes functions and operators whereas external nodes represent terminal set which includes variables and constants.

2.8 ZeroR: ZeroR is the simplest classification method which relies on the target and ignores all predictors. ZeroR classifier simply predicts the majority category (class). Although there is no predictability power in ZeroR, it is useful for determining a baseline performance as a benchmark for other classification methods. It constructs a frequency table for the target and selects its most frequent value. It predicts the mean for a numeric class and mode for a nominal class.

2.9 DecisionTable: Decision tables, like decision trees or neural nets, are classification models used for prediction. They are induced by machine learning algorithms. A decision table consists of a hierarchical table in which each entry in a higher level table gets broken down by the values of a pair of additional attributes to form another table. The structure is similar to dimensional stacking. A visualization method is presented that allows a model based on many attributes to be understood even by those unfamiliar with machine learning. Various forms of interaction are used to make this visualization more useful than other static designs. Many decision tables include in their condition alternatives the don't care symbol, a hyphen. Using don't cares can simplify decision tables, especially when a given condition has little influence on the actions to be performed. In some cases, entire conditions thought to be important initially are found to be irrelevant when none of the conditions influence which actions are performed.

3 DATASET DESCRIPTION:

3.1 IRIS: It has 150 instances, 4 attributes namely sepal length, sepal width, petal length and petal width and 3 classes namely iris setosa, iris virginica and iris versicolor.

3.2 ABALONE: Predicting the age of abalone from physical measurements. It has 4177 instances, 8 attributes namely sex, length, diameter, height, whole-height, shucked-weight, Viscera-weight, Shell-weight and 29 classes numbered from 1-29.

3.3 LABOR: The data includes all collective arguments reached in the business and personal services sector for locals with at least 500 members**.** It has 57 instances, 16 attributes and 2 classes namely good and bad.

3.4 CONTACT-LENSES: It is the database for fitting contact lenses. It has 24 instances, 4 attributes namely age, spectacle-prescription, astigmatism, tear-production-rate and 3 classes namely soft, hard and none.

3.5 SOYBEAN: It is the description of Soybean disease**.** It has 683 instances, 35 attributes and 19 classes.

3.6 HAYESROTH: It has 160 instances, 5 attributes namely Name, Hobby, Age, Education Level, Marital Status and 3 classes numbered from 1-3.

3.7 LUNG CANCER: The data describes 3 types of pathological lung cancer**.** It has 32 instances, 56 attributes and 3 classes numbered from 1-3.

3.8 GLASS IDENTIFICATION DATASET: It has 9 attributes(Refractive index, Sodium, Pottasium, Magnesium, Aluminium, calcium, Silicon, Barium and iron content) and consist of 214 instances of 7 different classes namely Building windows Float proceesed glass, Vehicle windows float processed glass, Buliding windows non-float processed glass , vehicle windows non-float processed glass, containers non-window glass, table ware non-window glass and headlamps non –window glass.

3.9 TEACHING ASSISTANT EVALUATION: 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. It contains 151 instances, 5 attributes namely whether or not the TA is a English Native Speaker, course instructor, course, summer or regular semester, class size and 3 classes numbered from 1-3 where 1 indicates low, 2 indicates medium and 3 indicates high.

3.10 VOTE: 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 (these three simplified to yea), voted against, paired against, and announced against (these three simplified to nay), voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known (these three simplified to an unknown disposition). **.**It has 435 instances, 16 attributes and 2 classes namely democrat and republican.

3.11 STATLOG (Australian Credit Approval): This file concerns credit card applications. All attribute names and values have been changed to meaningless symbols to protect confidentiality of the data. This dataset is interesting because there is a good mix of attributes -- continuous, nominal with small numbers of values, and nominal with larger numbers of values. There are also a few missing values. It contains 690 instances, 14 attributes and 2 classes.

4 METHODOLOGY:

a) WEKA EXPLORER:

The full form of WEKA is Waikato Environment for Knowledge Learning. Weka is a computer program that was developed by the student of the University of Waikato in New Zealand for the purpose of identifying information from raw data gathered from agricultural domains . Data preprocessing, classification, clustering, association, regression and feature selection these standard data mining tasks are supported by Weka. It is an open source application which is freely available. In Weka , datasets should be formatted to the ARFF format. Classify and Select Attribute tab in Weka Explorer is used for the classification after feature selection purpose. A large different number of classifiers are used in weka such as bayes, function, tree etc.

Steps to apply feature selection and classification techniques on data set to get results in Weka:

Step 1: Take the input dataset and open it from preprocess tab.
Step 2: Go to the Select Attribute tab and choose cfsSubsetEval as Attribute Evaluator and Genetic Search as the search method. This will perform feature selection or dimension reduction.
Step 3: Checkmark only those features which are selected by cfsSubsetEval and Genetic Search. Remove rest of the features.
Step 4: Apply the classifier algorithm on the whole data set.
Step 5: Note the accuracy given by it and time required for execution.
Step 6: Repeat steps 2, 3, 4 and 5 for different classification algorithms on different datasets.
Step 7: Compare the different accuracy provided by the dataset with different classification algorithms and identify the significant classification algorithm for particular dataset

b) ECLIPSE:

Eclipse is a Java-based open source platform that allows a software developer to create a customized development environment (IDE) from plug-in components built by Eclipse members. The WEKA library is imported in eclipse. Then packages are imported for cfsSubsetEval, genetic search and different classifiers.

Steps to apply feature selection and classification on data set through Eclipse:

Step 1: First of all weka library is imported in eclipse. This can be done by performing the following steps:

a) Create a new java project.
b) Right click on the newly created java project then perform these steps: Build Path -> Configure Build Path ->Java Build Path-> Java Libraries ->Add External Jars.
c) Browse for the Weka.jar and add it.

Step 2:

a) Import package for instance creation: import weka.core.Instances.
b) Import package for supervised attribute selection:
a) import weka.filters.supervised.attribute.AttributeSelection
b) Import package for cfsSubsetEvaluator:

c) import weka.attributeSelection.CfsSubsetEval.
d) Import package for genetic search: import weka.attributeSelection.GeneticSearch.
e) Import package for different classifiers: import weka.classifiers.bayes.NaiveBayes, import
f) weka. classifiers. functions. Multilayer Perceptron, import weka. classifiers. trees. J48, import weka. classifiers. functions. Genetic Programming, similarly we can import package for any classifier.

Step 3: Different methods (functions) are used to calculate accuracy and time complexity such as set Evaluator, set Search, set Input Format, cross Validate Model, to Summary String, set Class Index, fMeasure, precision, recall.

The Experiments are conducted in a system with configuration:

Processor: Intel(R) Core (TM) i5-3230M CPU @ 2.60GHz.
RAM: 4GB
HDD: 1TB

5 PERFORMANCE MEASURES:

1. Accuracy Classification

All classification result could have an error rate and it may fail to classify correctly. So accuracy can be calculated as follows. Accuracy = (Instances Correctly Classified / Total Number of Instances)*100 %

2. Mean Absolute Error(MAE)

MAE is the average of difference between predicted and actual value in all test cases. The formula for calculating MAE is given in equation shown below:

MAE = (|a1 – c1| + |a2 – c2| + … +|an – cn|) / n

Here ‘a’ is the actual output and ‘c’ is the expected output.

3. Root Mean Squared Error(RMSE)

RMSE is used to measure differences between values predicted by a model and the values actually observed. It is calculated by taking the square root of the mean square error as shown in equation given below:

$$\surd\{((a1 - c1)^2 + \ (a2 - c2)^2 + \cdots (an - cn)^2)/n\}$$

Here ‘a’ is the actual output and c is the expected output. The mean-squared error is the commonly used measure for numeric prediction.

4. Confusion Matrix: A confusion matrix contains information about actual and predicted classifications done by a classification system.

5. Time Complexity: Time taken to execute the code of each classifier is calculated using: long b = System. Current Time Millis(); at the starting of code and long a =System. Current Time Millis(); at the ending of code. Finally printing the time taken by calculating the difference between a and b using System. out. println(a-b). The classification accuracy, time complexity, mean absolute error, root mean squared error and confusion matrices are calculated for each machine learning algorithm.

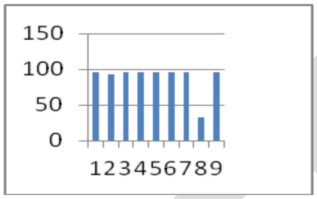
6 RESULTS

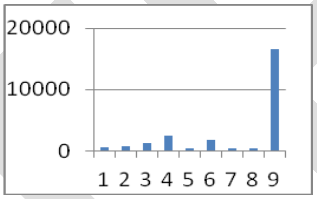
Ranking is done on the basis of accuracy and in case of tie time complexity is used to determine which algorithm performs better.

6.1 FOR IRIS DATASET

Number of features get reduced from 4 to 2 by using cfsSubsetEval and genetic search.

	MAE	RMSE	Accuracy	Time Complexity (ms)	Ranking
1. J48	0.035	0.1586	96%	503	3
2. Decision Table	0.092	0.2087	92.67%	676	8
3. Logistic	0.0324	0.1413	96%	1196	4
4.Multilayer Perceptron	0.0532	0.1559	96%	2416	6
5. NaiveBayes	0.0286	0.1386	96%	450	2
6. Random Forest	0.0366	0.1515	96%	1790	5
7. VFI	0.0623	0.1623	96.67%	406	1
8. ZeroR	0.4444	0.4714	33.33%	382	9
9. Genetic Programming	0.0311	0.1764	96.00%	16657	7



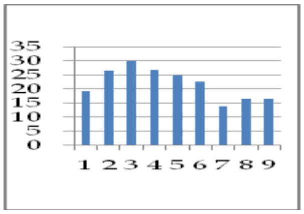


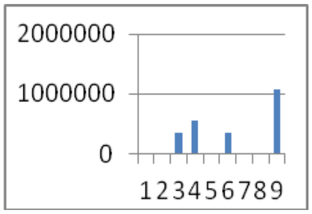
Accuracy Chart Time Complexity Chart

6.2 FOR ABALONE DATASET:

Number of features get reduced from 8 to 5 by using cfsSubsetEval and genetic search

	MAE	RMSE	Accuracy	Time Complexity (ms)	Ranking
1. J48	0.0565	0.2056	19.2%	7206	6
2. Decision Table	0.0588	0.1703	26.52%	5076	3
3.Logistic	0.0599	0.1834	30.003%	351456	1
4.Multilayer Perceptron	0.0562	0.1688	26.86%	551456	2
5.NaiveBayes	0.0558	0.1789	24.77%	2181	4
6.Random Forest	0.056	0.1759	22.6%	350904	5
7.VFI	0.0629	0.1772	13.811%	1578	9
8.ZeroR	0.0618	0.1757	16.4951%	713	8
9.Genetic Programming	0.0575	0.2398	16.519%	1076914	7





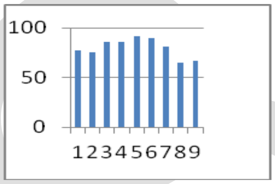
Accuracy Chart Time Complexity Chart

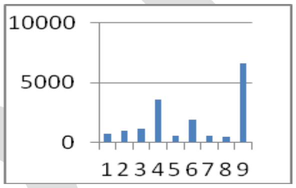
6.3 FOR LABOR DATASET:

Number of features get reduced from 16 to 7 by using cfsSubsetEva and genetic search.

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1. J48	0.2787	0.441	77.19%	742	6
2.Decision Table	0.3081	0.4061	75.4386%	983	7
3. Logistic	0.1319	0.3491	85.9649%	1171	3
4.Multilayer Perceptron	0.1427	0.344	85.9649%	3570	4
5.NaiveBayes	0.1194	0.2596	91.2281%	577	1
6.RandomForest	0.222	0.3263	89.4737%	1933	2
7.VFI	0.2784	0.3786	80.7018%	533	5
8.ZeroR	0.4574	0.4775	64.9123%	430	9
9.Genetic Programming	0.3333	0.5774	66.667%	6599	8

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1.J48	0.0148	0.089	90.3367%	1553	5
2.Decision Table	0.065	0.1583	81.4056%	4370	8
3.Logistic	0.0083	0.0805	92.9772%	78999	2
4.Multilayer Perceptron	0.0093	0.0741	93.7042%	100257	1
5.NaiveBayes	0.0108	0.0813	92.0937%	1166	4
6.Random Forest	0.0191	0.0892	92.9722%	99876	3
7.VFI	0.0725	0.1888	86.0908%	1188	6
8.ZeroR	0.0961	0.2191	13.47%	766	9
9.Genetic Programming	0.0996	0.3155	81.91%	606928	7



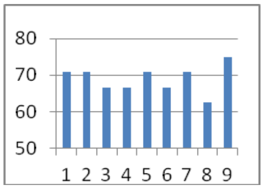


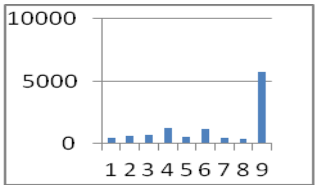
Accuracy Chart Time Complexity Chart

6.4 FOR CONTACT LENSES DATASET:

Numbers of features get reduced from 4 to 1 using cfsSubsetEval and genetic search:

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1.J48	0.2348	0.3571	70.8333%	438	2
2.Decision Table	0.3176	0.3846	70.8333%	625	5
3. Logistic	0.2348	0.3571	66.67%	716	6
4.Multilayer Perceptron	0.2425	0.3568	66.67%	1262	8
5.NaiveBayes	0.2793	0.3603	70.83%	502	4
6.RandomForest	0.2337	0.355	66.67%	1145	7
7.VFI	0.3778	0.4367	70.833%	445	3
8.ZeroR	0.3778	0.4367	62.5%	338	9
9.Genetic Programming	0.3778	0.4367	75%	5726	1

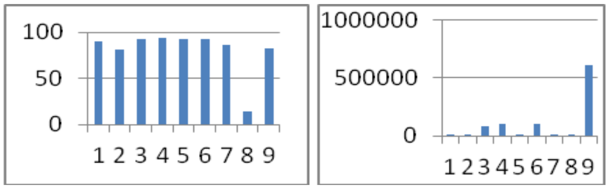




Accuracy Chart Time Complexity Chart

6.5 FOR SOYBEAN DATASET:

Number of features get reduced from 35 to 24 using cfsSubsetEval and genetic search

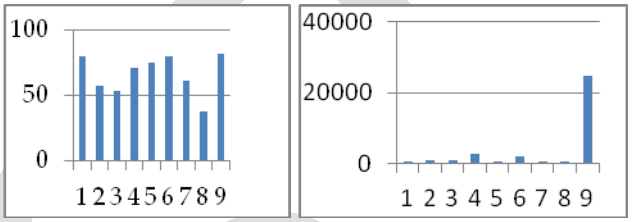


Accuracy Chart Time Complexity Chart

6.6 FOR HAYESROTH DATASET:

Number of features get reduced from 5 to 3 using cfsSubsetEval and genetic search:

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1.J48	0.134	0.2798	80.303%	819	2
2.DecisionTable	0.308	0.379	56.8182%	828	7
3.Logistic	0.2952	0.3912	53.7879%	855	8
4.Multilayer Perceptron	0.2222	0.3515	71.2121%	2948	5
5.NaiveBayes	0.2932	0.3605	75%	615	4
6.RandomForest	0.1024	0.2236	80.303%	2233	3
7.VFI	0.4017	0.4303	61.3636%	584	6
8.ZeroR	0.4335	0.4655	37.8788%	497	9
9.Genetic Programming	0.1364	0.3693	81.82%	24653	1

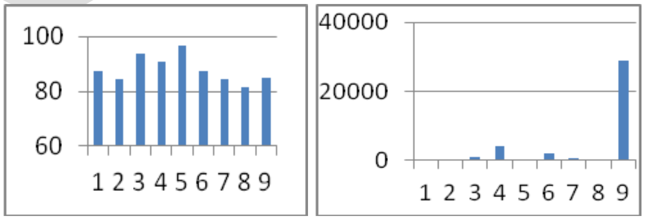


Accuracy Chart Time Complexity Chart

6.7 FOR LUNG CANCER DATASET:

Number of features get reduced from 56 to 4 by using cfsSubsetEval and genetic search

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1.J48	0.0903	0.2608	87.5%	450	4
2.Decision Table	0.2468	0.3016	84.375%	359	7
3.Logistic	0.0409	0.2005	93.75%	812	2
4.Multilayer Perceptron	0.0643	0.2155	90.625%	4137	3
5.NaiveBayes	0.0426	0.1603	96.875%	250	1
6.RandomForest	0.0792	0.2138	87.5%	2129	5
7.VFI	0.277	0.3448	84.375%	581	8
8.ZeroR	0.2285	0.3249	81.25%	250	9
9.Genetic Programming	0.2987	0.2567	85%	28739	6

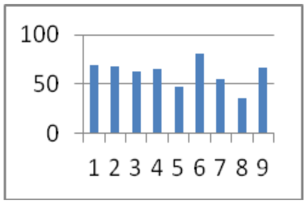


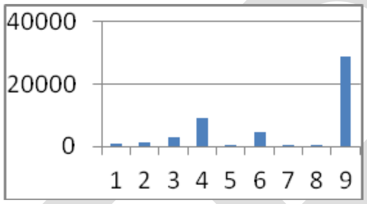
Accuracy Chart Time Complexity Chart

6.8 FOR GLASS IDENTIFICATION DATASET:

Number of features get reduced from 9 to 8 by using cfsSubsetEval and genetic search

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1. J48	0.0997	0.2822	68.6916%	859	2
2. Decision Table	0.1734	0.278	68.2243%	1265	3
3. Logistic	0.1246	0.27779	63.0841%	2953	6
4.Multilayer Perceptron	0.1186	0.274	65.8879%	9048	5
5. NaiveBayes	0.1544	0.3387	47.6636%	547	8
6.RandomForest	0.0971	0.2058	80.3738%	4484	1
7. VFI	0.2053	0.3113	54.6729%	500	7
8.ZeroR	0.2118	0.3245	35.514%	406	9
9.Genetic Programming	0.0948	0.3079	66.824%	28955	4



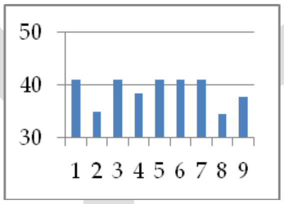


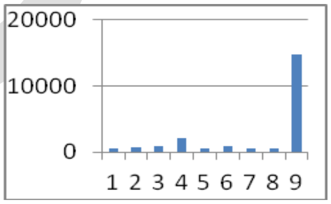
Accuracy Chart Time Complexity Chart

6.9 FOR TEACHING ASSISTANT EVALUATION:

Number of features get reduced from 5 to 1 by using cfsSubsetEval and genetic search

	MAE	RMSE	Accuracy	Time Complexity (ms)	Ranking
1.J48	0.4284	0.4644	41.0596%	442	1
2.Decision Table	0.4415	0.4736	35.0093%	775	7
3.Logistic	0.4284	0.4644	41.0596%	835	3
4.Multilayer Perceptron	0.4289	0.4665	38.4106%	2049	5
5.NaiveBayes	0.4242	0.4656	41.0596%	442	1
6.RandomForest	0.429	0.4649	41.0596%	882	4
7.VFI	0.4418	0.4691	41.0596%	518	2
8.ZeroR	0.4444	0.4714	34.4371%	448	8
9.Genetic Programming	0.415	0.6442	37.7483%	14767	6



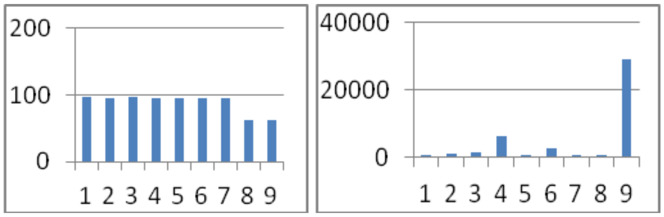


Accuracy Chart Time Complexity Chart

6.10 FOR VOTE DATASET:

Numbers of features get reduced from 16 to 4 using cfsSubsetEval and genetic search

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1.J48	0.0687	0.1794	96.092%	725	1
2.Decision Table	0.0829	0.2016	95.6332%	1099	6
3.Logistic	0.0601	0.1778	96.092%	1439	2
4.Multilayer Perceptron	0.0543	0.1771	95.8621%	6106	5
5.NaiveBayes	0.0575	0.1768	96.02%	590	3
6.RandomForest	0.06	0.1729	95.1724%	2582	7
7.VFI	0.2195	0.2473	95.8621%	600	4
8.ZeroR	0.4742	0.4869	61.3793%	416	8
9.Genetic Programming	0.3862	0.6215	61.3793%	28937	9

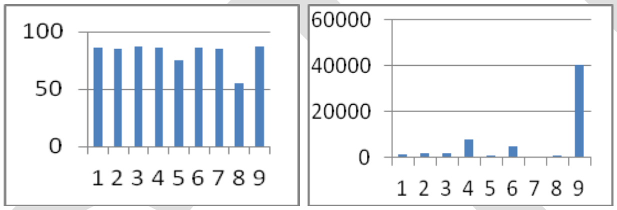


Accuracy Chart Time Complexity Chart

6.11 FOR STATLOG DATASET:

Numbers of features get reduced from 14 to 7 using cfsSubsetEval and genetic search

	MAE	RMSE	Accuracy	Time Complexity(ms)	Ranking
1.J48	0.198	0.341	85.7971%	1391	4
2.Decision Table	0.234	0.338	85.07225%	1892	7
3.Logistic	0.1945	0.3143	86.6667%	1687	2
4.Multilayer Perceptron	0.1863	0.3302	85.7971%	8158	5
5.NaiveBayes	0.2398	0.4524	75.3623%	833	8
6.RandomForest	0.1838	0.3033	86.087%	5108	3
7.VFI	0.4749	0.4749	85.3623%	398	6
8.ZeroR	0.494	0.494	55.5072%	724	9
9.Genetic Programming	0.129	0.3591	87.1014%	40305	1



Accuracy Chart Time Complexity Chart

7 CONCLUSIONS AND DISCUSSIONS

Thus in this paper we have compared the performance of various classifiers. Eleven data sets from benchmark data set (UCI) are used for experimentation. Numbers of cross-folds in each case are 10. In terms of overall performance that is if we consider Accuracy, Time Complexity, MAE and RMSE, **MLP, NaiveBayes, RandomForest, J48, Genetic Programming perform comparatively better than others in case of all datasets.** According to the rankings, for iris VFI performed best, for abalone Logistic performed best, for labor and lung cancer NaiveBayes performed best, for contact-lense, hayesroth and statlog Genetic Programming(GP) performed best, for Soybean MLP performed best, for glass identification test RandomForest performed best, for vote J48 performed best, for teaching assistant evaluation NaiveBayes and J48 both performed best. ZeroR performed worst in almost all the cases. As this work is much concerned on GP, it can be concluded from results section that accuracy given by GP is appreciable in almost all datasets except abalone, labor and teaching assistant evaluation. In case of contact-lense, hayesroth and statlog datasets accuracy given by GP is the highest. The performance of GP decreases by a small amount as the no of instances increases because GP is an iterative process. As the number of instances increase number of iterations also increase. This is the case with abalone dataset. For the datasets containing missing values for attributes performance of GP decreases. Time complexity charts for different datasets show similarity. The height of the bar for GP is highest in every chart. In every case, time complexity of GP is maximum. This is because GP is an iterative process, the number of iterations are the same as the number of generations. The most commonly used representation schema in GP is tree structure such as parse trees and decision trees. This also increases the time taken by GP for classification. After GP, MLP took maximum time followed by Random Forest and others. In general it is found that the performance of classification techniques varies with different data sets. Factors that affect the classifier's performance are 1. Data set, 2. Number of instances and attributes, 3. Type of attributes, 4. System configuration.

8 FUTURE WORK

Our future work will be to implement the combination of classification techniques to improve the performance.

REFERENCES

[1] Clustering using firefly algorithm: Performance study: J.Senthilnath, S.N. Omkar, V.Mani .

[2] A survey on the Application Of Genetic Programming to Classification: Pedro G. Espejo, Sebastian Ventura, and Francisco Herrera.

[3] Application Of Genetic Programming for Multicategory Pattern Classification : J.K. Kishore, L.M. Patnaik, V.Mani, and V.K. Agarwal.

[4] Black Hole : A new heuristic optimization approach for data clustering.

[5] Comparison of different classification techniques using different datasets: V.vaithiyanathan, K.Rajeswari, Rahul Pitale.

[6] Classification of multivariate datasets without missing values using memory based classifier- An effective evaluation: C.Lakshmi Devasena.

[7] John R. Koza: Genetic programming on the programming of computers by means of natural selection.

[8] Applied Evolutionary Algorithms In java: Robert Ghanea Hercock.

[9] Discovering Interesting Classification rules with Genetic Programming: De Falco, A. Della Cioppa, E.Tarantino.

[10] Feature Selection and classification In Genetic Programming : Application to Haptic-Based Biometric data.

[11] Genetic Programming for classificationv learning problems: Thomas loveard.

[12] S.N. Omkar, Manoj kumar M, Dheevatsa Mudigere, Dipti Muley: Urban Satellite Image Classification using Biologically Inspired Techniques

[13] Crop Classification using Biologically-inspired Technique with High Resolution Satellite Image: S. N. Omkar . J. Senthilnath . Dheevatsa Mudigere . M. Manoj Kumar

[14] Crop stage classification of Hyperspectral data using unsupervised techniques: J.Senthilnath, S.N. Omkar, V.Mani, Nitin Karnval and Shreyas P.B .

[15] Hierarchical clustering Algorithm for Land Cover Mapping Using Satellite Images:J.Senthilnath, S.N. Omkar, V.Mani .

[16] Crop Type Classification Based On Clonal Selection Algorithm for High Resolution Sattelite Image: J.Senthilnath, Nitin Karnval, D Sai Teja.

[17] Hierarchical Artificial Immune System For Crop Stage Classification : J. Senthilnath, Nitin Karnval.

[18] Improving the accuracy of Land Use and Land Cover classification of LandSat Data Using Post Classification Enhancement : By Ramita Manandhar and Tiko Ancev.

[19] Satellite Image Processing for land use and land cover Mapping: Ashoka Vanjare, S.N. Omkar, J.Senthilnath.

[20] Classification of Remote Sensing Image Areas Using Feature and Latent Drichlet Allocation Ms Chandrakala, Mrs R. Amsaveni.

[21] Impact of Accuracy, Spatial Availability, and Revisit Time Of Satellite-Derived Surface Soil Moisture in a Multiscale Ensemble Data Assimilation System: Ming Pan and Eric F.Wood.