Comparison of Classification Methods Based on the Type of Attributes and Sample Size

Reza Entezari-Maleki*[Corresponding author], Arash Rezaei, and Behrouz Minaei-Bidgoli
Department of Computer Engineering,
Iran University of Science & Technology (IUST), Tehran, Iran
r_entezari@comp.iust.ac.ir, arash_rezaei@comp.iust.ac.ir, b_minaei@iust.ac.ir

Abstract

In this paper, the efficacy of seven data classification methods; Decision Tree (DT), k-Nearest Neighbor (k-NN), Logistic Regression (LogR), Naïve Bayes (NB), C4.5, Support Vector Machine (SVM) and Linear Classifier (LC) with regard to the Area Under Curve (AUC) metric have been compared. The effects of parameters including size of the dataset, kind of the independent attributes, and the number of the discrete and continuous attributes have been investigated.

Based on the results, it can be concluded that in the datasets with few numbers of records, the AUC become deviated and the comparison between classifiers may not do correctly. When the number of the records and the number of the attributes in each record are increased, the results become more stable. Four classifiers DT, k-NN, SVM and C4.5 obtain higher AUC than three classifiers LogR, NB and LC. Among these four classifiers, C4.5 provides higher AUC in the most cases. As a comparison among three classifiers LogR, NB and LC, it can be said that NB provides the best AUC among them and classifiers LogR and NB have the same results, approximately.

Keywords

Classification methods, Area Under Curve metric, Sample size, Attributes types.

1. Introduction

Data mining algorithms which carry out the assigning of objects into related classes are called classifiers. Classification algorithms include two main phases; in the first phase they try to find a model for the class attribute as a function of other variables of the datasets, and in the second phase, they apply previously designed model on the new and unseen datasets for determining the related class of each record [1]. There are different methods for data classification such as Decision Trees (DT), Rule Based Methods, Logistic Regression (LogR), Linear Regression (LR), Naïve Bayes (NB), Support Vector Machine (SVM), k-Nearest Neighbor (k-NN), Artificial Neural Networks (ANN), Linear Classifier (LC) and so forth [1], [2], [3]. The comparison of the classifiers and using the most predictive classifier is very important. Each of the classification methods shows different efficacy and accuracy based on the kind of datasets [4]. In addition, there are various evaluation metrics for comparing the classification methods that each of them could be useful depending on the kind of the problem.

Receiver Operating Characteristic (ROC) curve [5]-[10] is a usual criterion for identifying the prediction power of different classification methods, and the area under this curve is one of the important evaluation metrics which can be applied for selecting the best classification method [5]-[13]. Among the other criteria for comparing the classification methods, one could mention; G-means [14], RMSE [4], [15] and Accuracy [6], [16].

In this article, using a new method, seven usual data classification methods (DT, k-NN, LogR, NB, C4.5, SVM, and LC) have been compared based on the AUC criterion. These mentioned methods have been applied on the random generated datasets which are independent from a special problem. This comparison is based on the effect of the numbers of existing discrete and continuous attributes and the size of the dataset on the AUC.

The rest of the paper is organized as follows: In section 2, previous works related to this area and the motivations of performing the new work have been presented. Section 3 provides an explanation about dataset generation and classification methods. Reporting the results of the applying classification methods on the datasets is presented in section 4. Section 5 evaluates the results and investigates the efficacy of the classifiers. Finally, section 6 concludes the paper and describes future works.

2. Related Works

Many works related to the comparison of the classification methods have been done. Any of these works has compared various classifiers with each other and with regards to the test data and evaluation criteria, has reported the results.

Efficiency criterion RMSE has been used by Kim in [4] for comparing DT, ANN and LR. In this article the effect of the kind of attributes and the size of dataset on the

classification methods have been investigated and the results have been reported. RMSE also has been used by Kumar in [15] for comparing ANN and regression. Regression and ANN have been applied on the real and simulated data and the end results have been reported. These results show that if the data include errors and real values of attributes are not available, the statistical method of regression could act better than the ANN method and its performance is much superior.

Huang et al. [6] have compared NB, DT and SVM with each other using AUC criterion. In [6], by applying mentioned methods on the real data, it is shown that the AUC criterion is better than accuracy for comparing the classification methods. Furthermore, it is shown that C4.5 implementation of DT has higher AUC compared to NB and SVM.

Song et al. [7] have compared LogR and ANN for breast cancer detection by using the experimental medical data. In [7], it is shown that LogR and ANN almost have the same efficacy, but in this situation and sensitivity of detection, using ANN compared to LogR is prior. In [9], Rudolfer et al. have compared LogR and DT, and have reported that the efficacy of the LogR and DT method is the same so they have presented a synthesis method which has higher efficacy compared to the other previous methods. Long et al. [12] have compared LogR and DT in medical application considering AUC criterion. The comparison in this work has shown that the two mentioned methods have almost the same efficacy, but in the tasted data in this article, LogR partly has more efficacy compared to DT.

Amor et al. [13] have compared DT and NB in intrusion detection systems. This comparison has been done on KDD'99 and the gained results have expressed that the estimated predictions with NB are better than DT's predictions. Also the same comparison has been done between SVM and ANN, by Chen et al. in [11]. The reported results have shown that in considered case, SVM acts better than ANN.

Le Xu et al. [14] have compared LogR and ANN for finding the source of the error in power distribution using the G-mean criterion. According to this article, ANN has better results compared to LogR and therefore; using neural networks in this special case has been proposed. Amendolia et al. [16] have compared k-NN, SVM and ANN for talasemi detection using accuracy criterion. This test has been done for real data and results of the test show that ANN could act better than the other two methods. Karacali et al. [17] have compared SVM and k-NN methods using error rate, and finally by combining these two methods and using the power of SVM and simplicity of k-NN have gained a synthesis classifier which has the advantages of the two methods. O'Farrell et al. [18] have compared k-NN and ANN in classification of the spectral data. The results have shown that if values of data have deviation from real values, using ANN is good, otherwise using the simple k-NN classifier is more advised.

All of the articles mentioned above have compared different classifiers with each other. The problem that engage the most of these works, is gathering the experimental results from applying the classifiers on the datasets which are related to a determined problem. Therefore, decision making based on these results is not general, and it makes different judgments about the priority of one method to the others. As an example, the reported results in [6] and [13], [9] and [12], and finally [11] and [16] not only do not match but also conflict each other, and dependence of the datasets on a special problem causes these differences.

Furthermore, the most of the works which have been done in this field have ignored parameters like; size of the datasets, kind of the attributes, and the number of discrete and continuous attributes which affect on efficacy of the classifies' prediction.

3. Data Analysis and Classification Methods

In this section, we first express the approach of dataset generation and then explain applying of classifiers on the random generated datasets.

3.1 Random Dataset Generation

Linear data creation model [4] has been used for generating dataset. Class label is assumed as a linear function of a set of the discrete and continuous attributes. Class label is calculated from Eq. (1) for each record i which has n continuous attributes with symbol x and m discrete attributes with symbol c,

$$Y_i = 1 + 3 \times \sum_{j=1}^{n} x_j + 2 \times \sum_{j=1}^{m} c_j \qquad (1)$$

Where x is a continuous variable and has monotonic distribution in interval [0,1]. Variables c and Y, in Equation 1, are continuous and then using Eq. (2) which categorizes and changes to the discrete variables.

$$Y_{Discrete} = Y_{Continuous} \bmod M \qquad (2)$$

With regard to the above explanation, datasets with different sizes could be made. These datasets in addition to independency of the special problem have capability of variation in the number of discrete and continuous variables. Characteristics of the datasets which have been generated are in Tables 1 to 4.

As is shown in Tables 1 to 4, datasets DS1 to DS29 include different numbers of discrete and continuous variables. Also for investigating the effect of the size of datasets on the efficacy of the classifiers, samples with

size of 200, 500, 1000, 3000 and 5000 records have been made from datasets. These numbers have been selected for generating datasets with small, medium and large sizes.

Table 1. Properties of datasets with 3 variables

ID	Number of Continuous Variables	Number of Discrete Variables
DS_1	3	0
DS_2	2	1
DS_3	1	2
DS_4	0	3

Table 2. Properties of datasets with 5 variables

ID	Number of Continuous Variables	Number of Discrete Variables
DS_5	5	0
DS_6	4	1
DS_7	3	2
DS_8	2	3
DS_9	1	4
DS_{10}	0	5

Table 3. Properties of datasets with 7 variables

ID	Number of Continuous Variables	Number of Discrete Variables
DS_{11}	7	0
DS_{12}	6	1
DS_{13}	5	2
DS_{14}	4	3
DS_{15}	3	4
DS_{16}	2	5
DS_{17}	1	6
DS_{18}	0	7

Table 4. Properties of datasets with 10 variables

ID	Number of Continuous Variables	Number of Discrete Variables
DS_{19}	10	0
DS_{20}	9	1
DS_{21}	8	2
DS_{22}	7	3
DS_{23}	6	4
DS_{24}	5	5
DS_{25}	4	6
DS_{26}	3	7
DS_{27}	2	8
DS_{28}	1	9
DS_{29}	0	10

For applying classifiers on datasets, two distinct datasets should be used. First dataset for training (training set) and the second one for testing (test set). In this article, Cross Validation method with fold value equal to 10 has been used for training and testing phases. And it causes each of the learners to be trained by 10 stages with 90% of data and to be tested by 10 stages with 10% of data.

Consequently, all of the records which exist in dataset will affect the training and testing of the classifiers. For applying classification methods on datasets, the Orange data mining tool and its programming language (Python) [19] have been employed. The AUC criterion has been used for comparing the efficacy of the classifiers.

3.2 Data Classification Methods

Different classifiers have been employed for this research as follows;

3.2.1 Decision Tree

A decision tree (DT) is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or terminal node) holds a class label [20]. The topmost node in a tree is the root node.

During tree construction, attribute selection measures are used to select the attribute which best partitions the tuples into distinct classes. Three popular attribute selection measures are Information Gain, Gain Ratio, and Gini Index. When DTs are built, many of the branches may reflect noise or outliers in the training data. Tree pruning attempts to identify and remove such branches, with the goal of improving classification accuracy on unseen data.

3.2.2 C4.5

The C4.5 [22] is a decision tree based algorithm that uses a divide-and-conquer approach for growing the decision tree. A brief description of the method is given here. The following algorithm generates a decision tree from a set D of cases [23]:

- If D satisfies a stopping criterion, the tree for D is a leaf associated with the most frequent class in D. one reason for stopping is that D contains only cases of this class.
- Some test T with mutually exclusive outcomes T_1, T_2, T_3, … ,T_k is used to partition D into subsets D_1, D_2, D_3, …, D_k, Where D_i contains those cases that have outcome T_i. The tree for D has test T as its root with one sub tree for each outcome T_i that is constructed by applying the same procedure recursively to the cases in D_i.

C4.5 contains mechanisms for proposing three types of tests [22]:

- The “standard” test on a discrete attribute, with one outcome and branch for each possible value of that

attribute.

- A more complex test based on a discrete attribute in which the possible values are allocated to a variable number of groups with one outcome for each group rather than each value.
- If attribute *A* has continuous numeric values, a binary test with outcomes $A \leq Z$ and $A > Z$, based on comparing the value of *A* against a threshold value *Z*.

All these tests are evaluated in the same way, looking at the gain ratio arising from the division of training cases that they produce. Two modifications to C4.5 for improving use of continuous attributes are presented in [23].

3.2.3 k-Nearest Neighbor

Nearest neighbor classifiers are based on learning by analogy, that is by comparing a given test tuple with training tuples which are similar to it. The training tuples are described by *n* attributes. Each tuple represents a point in an *n*-dimensional space. In this way, all of the training tuples are stored in an *n*-dimensional pattern space. When given an unknown tuple, a k-nearest neighbor (k-NN) classifier searches the pattern space for the *k* training tuples which are closest to the unknown tuple. These *k* training tuples are the *k*-nearest neighbors of the unknown tuple [20].

"Closeness" is defined in terms of a distance metric, such as Euclidean distance. The Euclidean distance between two points or tuples $X_1=(x_{11}, x_{12},\ldots, x_{1n})$ and $X_2=(x_{21}, x_{22},\ldots, x_{2n})$ obtained from Equation 3.

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \qquad (3)$$

The basic steps of the k-NN algorithm are;

- To compute the distances between the new sample and all previous samples, have already been classified into clusters;
- To sort the distances in increasing order and select the *k* samples with the smallest distance values;
- To apply the voting principle. A new sample will be added (classified) to the largest cluster out of *k* selected samples [2].

3.2.4 Regression

Linear regression (LR) is used to model continuous-valued functions. It is widely used, owing largely to its simplicity. Generalized linear models represent the theoretical foundation on which LR can be applied to the modeling of categorical response variables. Common types of generalized linear models include logistic regression (LogR) and Poisson regression. LogR models the probability of some event occurring as a linear function of a set of predictor variables. Count data frequently exhibit a Poisson distribution and are commonly modeled using Poisson regression [20]. In this article, LogR has been used as a one of the common classification methods.

3.2.5 Naïve Bayes

Bayesian classifiers are statistical classifiers. They can predict class membership probabilities [20]. Naïve Bayes (NB) probabilistic classifiers are commonly studied in machine learning. The basic idea in NB approaches is to use the joint probabilities of words and categories to estimate the probabilities of categories given a document. The naïve part of NB methods is the assumption of word independence, i.e. the conditional probability of a word given a category is assumed to be independent from the conditional probabilities of other words given that category. This assumption makes the computation of the NB classifiers far more efficient than the exponential complexity of non-naïve Bayes approaches because it does not use word combinations as predictors [21].

3.2.6 Support Vector Machine

A support vector machine (SVM) is an algorithm that uses a nonlinear mapping to transform the original training data into a higher dimension. Within this new dimension, it searches for the linear optimal separating hyperplane. A hyperplane is a "decision boundary" separating the tuples of one class from another. With an appropriate nonlinear mapping to a sufficiently high dimension, data from two classes can always be separated by a hyperplane. The SVM finds this hyperplane using support vectors ("essential" training tuples) and margins (defined by the support vectors) [20].

The basic idea behind support vector machine is illustrated with the example shown in Figure 1. In this example the data is assumed to be linearly separable. Therefore, there exist a linear hyperplane (or decision boundary) that separates the points into two different classes. In the two-dimensional case, the hyperplane is simply a straight line. In principle, there are infinitely many hyperplanes that can separate the training data. Figure 1 shows two such hyperplanes, B_1 and B_2. Both hyperplanes can divide the training examples into their respective classes without committing any misclassification errors.

Although the training time of even the fastest SVMs can be extremely slow, they are highly accurate, owing to their ability to model complex nonlinear decision boundaries. They are much less prone to over fitting than other methods [20].

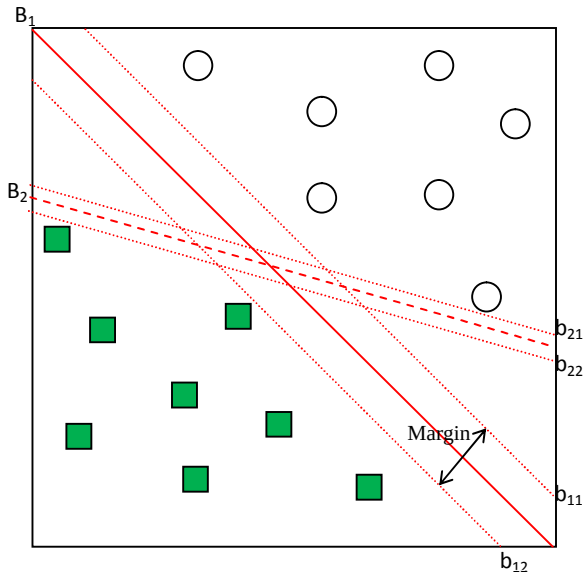

Figure 1. An example of a two-class problem with two separating hyperplanes, B_1 and B_2 [1]

3.2.7 Linear Classifier

Generalized linear models are currently the most frequently applied statistical techniques. They are used to describe the relationship between the trend of one variable and the values taken by several other variables. The relationship that fits a set of data is characterized by a prediction model called a regression equation. The most widely used form of the regression model is the general linear model formally written as Equation 4.

$$Y = a + b_1.X_1 + b_2.X_2 + b_3.X_3 + ... + b_n.X_n \quad (4)$$

Applying this equation to each of the given samples a new set of equations can be obtained i.e. Equation 5:

$$y_j = a + b_1.x_{1j} + b_2.x_{2j} + b_3.x_{3j} + ... + b_n.x_{nj} + e_j \quad (5)$$

and $j = 1,..., m$

Where ε_j's are errors of regression for each of m given samples. The linear model is called linear because the expected value of y_j is a linear function: the weighted sum of input values [2].

4. Experimental Results

The classifiers DT, k-NN, LogR, NB, C4.5, SVM and LC have been applied on datasets DS_1 to DS_{29} and with sample sizes 200, 500, 1000, 3000 and 5000 records. Because of reducing the amount of plots in this paper, the gained results of datasets have been reported in the form of Table 5, but only plots of datasets with 10 records are depicted. Table 5 shows the results of applying the classifiers on datasets DS_{19} to DS_{29}. For the sake of brevity other the results of applying classifiers on other datasets are not presented.

Because data have been generated randomly, for being sure about the truthfulness of results, some samples have been generated from each of datasets. After applying the classifiers on datasets, their averages have been calculated from the gained results.

5. Results Evaluation

In this section, the AUC of the classifiers based on the results gained from section 4 have been investigated. For the sake of brevity, only the diagrams related to DS_{19} to DS_{29} have been depicted and comparison between classifiers has been done by considering these diagrams. Other diagrams have the same results, approximately.

Figures 2 to 6 show the AUC of the classifiers for datasets with records number 200, 500, 1000, 3000 and 5000, respectively. As shown in Figure 2 in datasets with few numbers of records, the AUC become deviated, and the comparison between classifiers may not do correctly.

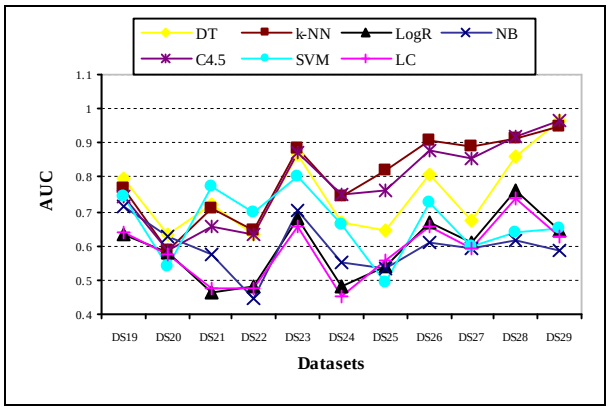

Figure 2. The AUC of datasets with 200 records

Since most of the data mining tasks encounter with large amount of data and huge datasets, it is better that the datasets with a large number of the records have been investigated. With drawing plots related to datasets with 500, 1000, 3000 and 5000 records the results become more stable, gradually. As shown in Figure 4 the plots related to datasets with 1000 records are roughly similar to the plots related to datasets with 3000 records as well as datasets with 3000 and 5000 records.

For comparing the classifiers and investigating the impact of the type of attributes, we define a metric C as follows:

C = (No. of continuous attributes) / (No. of total attributes) (6)

Table 5. AUC values for dataset DS_{19} to DS_{29}

ID	*Sample size*	*DT*	*k-NN*	*LogR*	*NB*	*C4.5*	*SVM*	*LC*
	200	0.7941	0.7683	0.6328	0.7126	0.7452	0.7448	0.6408
	500	0.8914	0.8347	0.6117	0.6650	0.8743	0.8589	0.6075
DS_{19}	***1000***	0.9141	0.8451	0.5702	0.6607	0.9176	0.9078	0.5690
	3000	0.9865	0.9220	0.5800	0.6520	0.9917	0.9841	0.5780
	5000	0.9942	0.9453	0.5844	0.6499	0.9956	0.9927	0.5830
	200	0.6343	0.5848	0.5828	0.6264	0.5872	0.5418	0.5727
	500	0.7984	0.7565	0.6318	0.6359	0.8034	0.7442	0.6236
DS_{20}	***1000***	0.9139	0.8208	0.5989	0.6371	0.9099	0.8609	0.5958
	3000	0.9892	0.9297	0.6241	0.6574	0.9917	0.9720	0.6211
	5000	0.9972	0.9579	0.6268	0.6517	0.9984	0.9860	0.6214
	200	0.7206	0.7074	0.4667	0.5737	0.6547	0.7728	0.4743
	500	0.8836	0.8273	0.5468	0.5947	0.9114	0.9569	0.5095
DS_{21}	***1000***	0.9684	0.8859	0.5646	0.6074	0.9731	0.9810	0.5496
	3000	0.9978	0.9668	0.5745	0.5955	0.9984	0.9973	0.5492
	5000	0.9965	0.9924	0.5642	0.5987	0.9963	0.9951	0.5467
	200	0.6307	0.6473	0.4789	0.4473	0.6334	0.6997	0.4731
	500	0.8869	0.8286	0.5477	0.5600	0.9079	0.9109	0.5478
DS_{22}	***1000***	0.9545	0.9221	0.6012	0.6366	0.9574	0.9577	0.6016
	3000	0.9929	0.9785	0.5559	0.5906	0.9906	0.9897	0.5495
	5000	0.9823	0.9722	0.5393	0.5743	0.9802	0.9666	0.5176
	200	0.8670	0.8829	0.6771	0.7060	0.8698	0.8044	0.6577
	500	0.9004	0.8938	0.6249	0.6747	0.8812	0.8143	0.5798
DS_{23}	***1000***	0.9389	0.9060	0.5612	0.6347	0.9337	0.8465	0.5272
	3000	0.9907	0.9709	0.6082	0.6353	0.9913	0.9386	0.5916
	5000	0.9977	0.9840	0.5931	0.6325	0.9981	0.9766	0.5598
	200	0.6700	0.7458	0.4791	0.5507	0.7521	0.6616	0.4500
	500	0.8269	0.8759	0.6047	0.6170	0.8745	0.7433	0.5935
DS_{24}	***1000***	0.9366	0.9294	0.6207	0.6629	0.9393	0.8467	0.6151
	3000	0.9950	0.9729	0.6117	0.6518	0.9964	0.9020	0.5993
	5000	0.9975	0.9859	0.6038	0.6539	0.9980	0.9634	0.5942
	200	0.6422	0.8172	0.5387	0.5318	0.7597	0.4948	0.5561
	500	0.8419	0.8468	0.4988	0.5636	0.8996	0.8714	0.5154
DS_{25}	***1000***	0.9840	0.9132	0.5546	0.6172	0.9826	0.9898	0.5438
	3000	0.9947	0.9838	0.5689	0.6008	0.9941	0.9940	0.5468
	5000	0.9944	0.9978	0.5627	0.6086	0.9952	0.9937	0.5482
	200	0.8069	0.9063	0.6679	0.6127	0.8760	0.7238	0.6564
	500	0.8901	0.9387	0.6326	0.6434	0.9312	0.7754	0.6080
DS_{26}	***1000***	0.9599	0.9584	0.6082	0.6083	0.9755	0.8081	0.5935
	3000	0.9946	0.9864	0.5608	0.6038	0.9956	0.9018	0.5321
	5000	0.9977	0.9945	0.5684	0.6187	0.9979	0.9675	0.5512
	200	0.6746	0.8919	0.6081	0.5949	0.8567	0.5961	0.5945
	500	0.8766	0.9436	0.4900	0.5789	0.9511	0.6865	0.4917
DS_{27}	***1000***	0.9587	0.9726	0.5094	0.5822	0.9811	0.7169	0.5142
	3000	0.9981	0.9960	0.5722	0.6094	0.9981	0.9120	0.5733
	5000	0.9992	0.9986	0.5303	0.6017	0.9992	0.9634	0.5331
	200	0.8622	0.9147	0.7635	0.6148	0.9190	0.6382	0.7384
	500	0.9185	0.9651	0.4849	0.5163	0.9639	0.6309	0.4896
DS_{28}	***1000***	0.9712	0.9782	0.5642	0.5773	0.9788	0.7655	0.5701
	3000	0.9982	0.9969	0.5339	0.5551	0.9990	0.8861	0.5260
	5000	0.9986	0.9983	0.5430	0.5620	0.9984	0.9347	0.5381
	200	0.9669	0.9508	0.6425	0.5886	0.9661	0.6524	0.6278
	500	0.9949	0.9828	0.5558	0.5492	0.9949	0.7645	0.5570
DS_{29}	***1000***	0.9975	0.9945	0.5278	0.5141	0.9975	0.7676	0.4999
	3000	0.9985	0.9987	0.5640	0.5364	0.9983	0.8647	0.5646
	5000	0.9983	0.9993	0.5417	0.5253	0.9983	0.8796	0.5314

Based on plots shown in Figure 3 to Figure 6 the following results can be induced:

DT, k-NN, C4.5 and SVM in all datasets have higher AUC than LogR, NB and LC. As C4.5 is a general implementation of DT, these two classifiers provide a similar behavior with respect to AUC in large datasets. But in small datasets (200 or 500 records) C4.5 has a higher AUC than DT.

k-NN provides high AUC in all cases. When C is larger than 0.5 (DS19 to DS23), this classifier has lower AUC than the opposite case (C<0.5). So when number of continuous attributes decreases in DS24 to DS29, k-NN acts better with respect to AUC.

Based on gained results, and focusing on Figure 4 to Figure 6, one can observe that the SVM provides less AUC than DT and C4.5. When the value of C is larger than 0.5, SVM is better than k-NN, but when C is gradually decreased, k-NN acts better than the other one.

Three classifiers including logR, NB and LC has lower AUC when they are compared with the other classifiers. NB is better than LC and logR when their AUC are compared. NB has higher AUC in the cases of C equal to 1 or 0.5 than the case of C equal to 0.

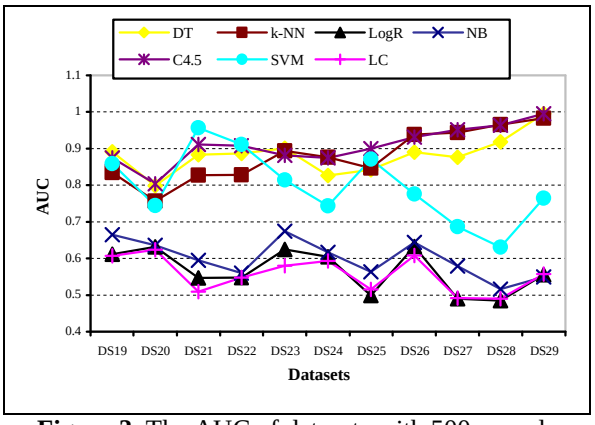

Figure 3. The AUC of datasets with 500 records

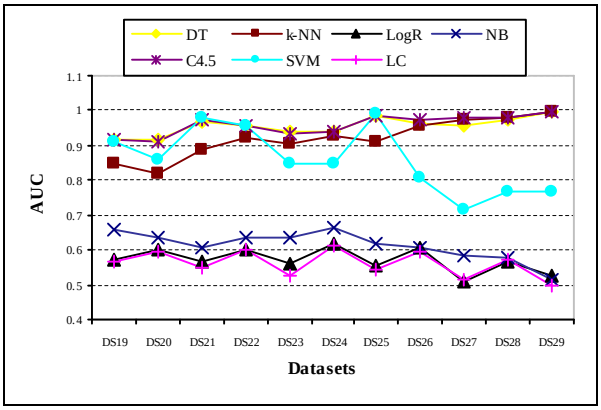

Figure 4. The AUC of datasets with 1000 records

LogR and LC have a similar behavior and approximately provide the same AUC when are applied to the identical datasets. In the comparison between these two classifiers, it can be said that the AUC of the LogR is better than the AUC of the LC in almost all datasets.

6. Conclusion and Future Works

In this article, the efficacy of DT, k-NN, LogR, NB, C4.5, SVM and LC has been investigated and the AUC of these four methods is compared to each other with respect to different circumstances of attributes deployment. In this comparison, the size of datasets, types of the attributes and the number of discrete and continuous attributes have been considered.

The above analysis shows that for small datasets, in all cases deviation of AUC is such high that the comparison between classifiers may not do correctly. For larger datasets results become more stable and the comparisons can be done. DT and C4.5 as an implementation of that, show an efficient performance in all datasets. Two classifiers including k-NN and

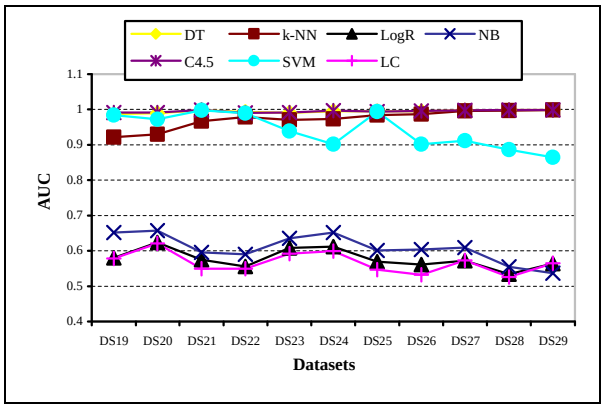

Figure 5. The AUC of datasets with 3000 records

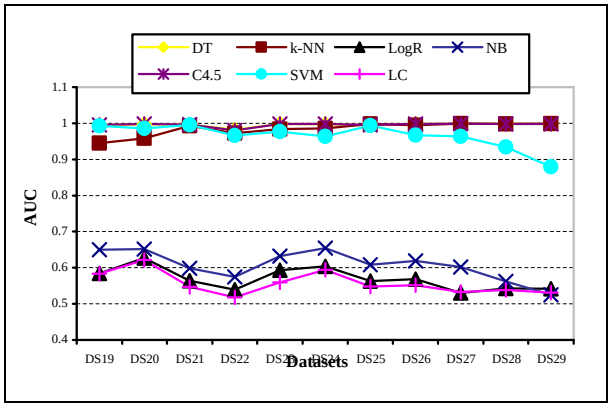

Figure 6. The AUC of datasets with 5000 records

SVM are classifiers with high efficacy in the current work. When the value of C is larger than 0.5, SVM has higher AUC than k-NN, and in the opposite situation, as the value of C is lower than 0.5, the AUC is higher.

Three classifiers including LogR, NB and LC have lower efficacy than the other ones. As a comparison between these classifiers it can be said that NB provides the best AUC between them and similar results for LogR and NB have been gathered.

Following the proposed process, we can choose the best classifier according to data type and continuous or discrete attributes existing in datasets. Therefore, one can use the above conclusion and choose the most efficient classifier with respect to his/her own dataset characteristics.

Lastly, it should be mentioned that the current research is based on simulation data and the generated datasets that are not dependent to a special problem. Consequently, the above results can be extended to a wide range of problems and these datasets are suitable for comparing the mentioned methods. In future studies, it is possible to compare the efficacy of other classifiers by using the current method. Furthermore, using other evaluation criteria and applying new classifiers on datasets with more variables could be as open problems in this field.

7. Acknowledgment

The authors want to express their gratitude to the Iranian National Elite Foundation for their support of this paper. This work is partially supported by Noor data mining research group as well.

8. References

[1] P-N. Tan, M. Steinbach, V. Kumar, “Introduction to Data Mining,” Addison-Wesley Publishing, 2006.

[2] M. Kantardzic, “Data Mining: Concepts, Models, Methods, and Algorithms,” John Wiley & Sons Publishing, 2003.

[3] I.H. Witten, E. Frank, “Data Mining: Practical Machine Learning Tools and Techniques,” Morgan Kaufmann Publishing, Second Edition, 2005.

[4] Y.S. Kim, “Comparision of the decision tree, artificial neural network, and linear regression methods based on the number and types of independent variables and sample size,” Journal of Expert Systems with Application, Elsevier, 2008, pp. 1227-1234.

[5] A. Fadlalla, “An experimental investigation of the impact of aggregation on the performance of data mining with logistic regression,” Journal of Information & Management, Elsevier, 2005, pp. 695-707.

[6] J. Huang, J. Lu, C.X. Ling, “Comparing Naïve Bayes, Decision Trees, and SVM with AUC and Accuracy,” Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[7] W.H. Chen, S.H. Hsu, H.P. Shen, “Application of SVM and ANN for intrusion detection,” Journal of computers & operations research, Vol. 32, Elsevier, 2005, pp. 2617-2634.

[8] J.H. Song, S.S. Venkatesh, E. A. Conant, P. H. Arger, C. M. Sehgal, “Comparative Analysis of Logistic Regression and Artificial Neural Network for Computer-Aided Diagnosis of Breast Messes,” Journal of Academic Radiology, Vol. 12, No. 4, 2005.

[9] W.J. Long, J.L. Griffith, H.P. Selker, R.B. D’Agostino, “A Comparison of logistic regression to decision-tree induction in a medical domain,” Journal of Computers and Biomedical Research, Vol. 26, 1993, pp. 74-97.

[10] A.P. Bradley, “The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithms,” Journal of Pattern Recognition, Vol. 30, No. 7, 1997, pp. 1145-1159.

[11] S.M. Rudolfer, “A Comparison of Logistic Regression to Decision Tree induction in the Diagnosis of Carpal Tunnel Syndrome,” Journal of Computers and Biomedical Research, Vol. 32, 1999, pp. 391-414.

[12] K.O. Hajian-Tilaki, J.A. Hanley, “Comparison of Three Methods for Estimating the Standard Error of the Area under the Curve in ROC Analysis of Quantitative Data,” Journal of Academic Radiology, Vol. 9, No. 11, 2002.

[13] N.B. Amor, S. Benferhat, Z. Elouedi, “Naïve Bayes vs Decision Trees in Intrusion Detection Systems,” ACM Symposium on Applied Computing, Cyprus, 2004.

[14] L. Xu, M-Y. Chow, X.Z. Gao, “Comparisons of Logistic Regression and Artificial Neural Network on Power Distribution Systems Fault Cause Identification,” IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications, Finland, 2005.

[15] U.A. Kumar, “Comparison of neural networks and regression analysis: A new insight,” Journal of Expert Systems with Applications, Vol. 29, 2005, pp. 424-430.

[16] S.R. Amendolia, G. Cossu, M.L. Ganadu, B. Golosio, G.L. Masala, G.M. Mura, “A comparative study of K-Nearest Neighbour, Support Vector Machine and Multi-Layer Perceptron for Thalassemia screening,” Journal of Chemometrics and Intelligent Laboratory Systems, Vol. 69, 2003, pp. 13– 20.

[17] B. Karacali, R. Ramanath, W.E. Snyder, “A comparative analysis of structural risk minimization by support vecto machines and nearest neighbor rule,” Journal of Pattern Recognition Letters, Vol. 25, 2004, pp. 63-71.

[18] M. O’Farrell, E. Lewis, C. Flanagan, W. Lyons, N. Jackman, “Comparison of k-NN and neural network methods in the classification of spectral data from an optical fiber-based sensors system used for quality control in the food industry,” Journal of Sensors and Actuators B, pp. 354-362, 2005.

[19] Orange See: http://www.ailab.si/orange/

[20] J. Han, M. Kamber, “Data Mining: Concepts and Techniques,” Elsevier, Second Edition, 2006.

[21] Y. Yang, X. Liu, “A re-examination of text categorization methods,” Annual ACM Conference on Research and Development in Information Retrieval, USA, 1999, pp. 42-49.

[22] J. R. Quinlan, “C4.5: Programs for Machine Learning”, Morgan Kaufmann Series in Machine Learning, 1993.
[23] J. R. Quinlan, “Improved Use of Continuous Attributes in C4.5”, Journal of Artificial Intelligence Research, volume 4, pp. 77-90, 1996.

Reza Entezari-Maleki received the B.Sc. and M.Sc. degree in computer engineering (software) (2007 and 2009, respectively) from department of computer engineering, Iran University of Science and Technology (IUST). He is also a member of Iranian National Elite Foundation.
His research interests include grid computing, task scheduling algorithms, modelling and performance/dependability analysis within grid environments, and data mining.

Arash Rezaei attained his B.Sc. in software engineering from Razi University. He is now a full time student in master degree of software engineering at Iran University of Science and Technology (IUST). His interest mainly spans around virtualization, dependable systems, data mining, Performance modelling and analysis.

Behrouz Minaei-Bidgoli obtained his Ph.D. degree from computer science & engineering department of Michigan State University, USA, in the field of data mining. Now, he is assistant professor in the department of computer engineering, Iran University of Science and Technology (IUST).
He is also leader of data and text mining research group in Noor computer research center, developing large scale NLP and text mining projects for Persian/Arabic language.