Buletin Teknol. Tanaman, Bil.12, Tambahan.2 (2015): 296-300

Comparing Various Classification Algorithms by WEKA

Morteza Okhovvat, Hassan Naderi

Abstract
In knowledge discovery process classification is an important technique of data mining and widely used in various fields. The development of data-mining applications such as classification and clustering has shown the need for machine learning algorithms to be applied to large scale data. In this paper we present the comparison of different classification techniques using Waikato Environment for Knowledge Analysis or in short, WEKA. WEKA is open source software which consists of a collection of machine learning algorithms for data mining tasks. The aim of this paper is to investigate the performance of different classification methods for a set of large data. The algorithm or methods tested are K-NN, C4.5, SVM and Bayes Network algorithm. Experimental results demonstrate that in the datasets with few numbers of records, comparing classifiers with regard the AUC may be in correct while the number of the records and the number of the attributes in each record are increased, the results become more stable. Resultants also show that the most accurate algorithm based on the generated data sets is Bayes network classifier with an accuracy of 89.71%.
Key words: Data mining, Weka tools, Classification Algorithms.

Introduction

Data mining is the use of automated data analysis techniques to uncover previously undetected relationships among data items. Data mining often involves the analysis of data stored in a data warehouse. Three of the major data mining techniques are regression, classification and clustering. In this research paper we are working only with the classification because it is most important process, if we have a very large database. Classification is another data mining tasks, can be defined as learning a function that maps (classifies) a data instance into one of several predefined class labels. When classification models are constructed from rules, often they are represented as a decision list. Classification rules are of the form P - C, where P is a pattern in the training data and C is a predefined class label (target). The objective of classification is to build a model in training dataset to predict the class of future objects whose class label is not known [1]. There are two major issues in classification, which are:

- Preparing the data for classification and prediction
- Comparing classification and prediction methods

There are commonly used classifications techniques which extract relevant relationship in the data are artificial neural networks, Decision trees, Bayesian Method [2], etc. Association and classification rules are represented as *If- then* type rules. However, there are some differences between them. Association rules are generally used as descriptive tools, which give the association relationships to the specific application experts, while classification rules are used for predicting the unseen testing data. However, a major problem in association rule mining is its complexity. The result of an arbitrary association rule mining algorithm is not the set of all possible relationships, but the set of all interesting ones. That is an important issue of the mining process, but the quality of the resulting rule set is ignored. On the other hand there are approaches to investigate the discriminating power of association rules and use them according to this to solve a classification problem [3][4]. However, 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 [5]. 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 [5], [6], [7]. 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 [8]. 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 [9, 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 [9, 11]. In this article, using a new method, four common data classification methods such as k-NN, NB, C4.5and SVM have been compared based on the AUC criterion and accuracy. 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 and the accuracy of classifiers' prediction. The rest of the paper is organized as follows: In section 2, notable related works have been presented. Section 3 provides classification methods. In section 4, applied datasets generation method is described Reporting the results of the applying classification methods on the datasets is presented in section 5. Section 6 evaluates the results and investigates the efficacy of the classifiers. Finally, section 7 concludes the paper and describes future works.

Related Works

There are, however, a lot of works related to the comparison of the classification methods. These works have compared various classifiers with each other based on evaluations criteria. Beyond them, notable works are discussed here. Minaei et al. [12] have compared DT, KNN, NB, SVM and C4.5 with regard to AUC. They also investigated the effects of the continuous and discrete attribute in their work. They asserted that in the datasets with few records, the AUC measure is deviated and isn't a suitable measure to evaluate the classifiers. Efficiency criterion RMSE has been used by Kim [8] 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 for comparing ANN and regression. Huang et al. [10] have compared NB, DT and SVM with each other using AUC criterion. In [10], 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. In [15], ANN and regression was compared. To this, 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. Le Xu et al. [16] 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. [17] have compared k-NN, SVM and ANN for Thalassemia 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. [18] 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. [19] 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 are good, otherwise using the simple k-NN classifier is more advised. However, although all of the mentioned works have compared various classifiers, but as the results are not general and related to the specific problem, decision making based on these results may not be correct. Some important parameters such as kind of attributes, size of datasets and the number of continuous and discrete attributes has not been considered which may have effect on the operation of the classifies.

Data Classification Methods

The various classifiers that have been employed for this research are shortly introduced as follows:

A. SVM

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, B1 and B2. 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].

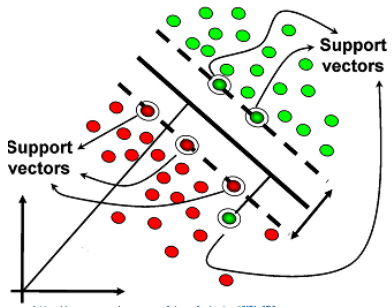

Figure 1. An example of a two-class problem with two separating hyperplanes [5]

B. Bayes Network Classifier

Bayesian networks are a powerful probabilistic representation, and their use for classification has received considerable attention. This classifier learns from training data the conditional probability of each attribute Ai given the class label C [19, 20].

Classification is then done by applying Bayes rule to compute the probability of C given the particular instances of A_1…..A_n and then predicting the class with the highest posterior probability. The goal of classification is to correctly predict the value of a designated discrete class variable given a vector of predictors or attributes [19]. In

particular, the naive Bayes classifier is a Bayesian network where the class has no parents and each attribute has the class as its sole parent [19, 20].

C. K- Nearest Neighbor

Nearest neighbors algorithm is considered as statistical learning algorithms and it is extremely simple to implement and leaves itself open to a wide variety of variations. In brief, the training portion of nearest-neighbor does little more than store the data points presented to it. When asked to make a prediction about an unknown point, the nearest neighbor classifier finds the closest training-point to the unknown point and predicts the category of that training point accordingly to some distance metric [22]. The distance metric used in nearest neighbor methods for numerical attributes can be simple Euclidean distance.

D. C4.5

The C4.5 [23] 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 $T_1, T_2, T_3, \ldots, T_k$is used to partition *D* into subsets $D_1, D_2, D_3, \ldots, D_k$, 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 *Di*. C4.5 contains mechanisms for proposing three types of tests [23]:
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≤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 [24].

Dataset

Linear data creation model [13] 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.

$$Y_i = 1 + 3*\sum_{k=1}^{n} x_k + 2 *\sum_{k=1}^{m} c_j \quad (1)$$

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.

$$Y_{Disc} = Y_{Cont} \bmod M \quad (2)$$

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. In order to apply 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. It causes each of the learners to be trained with 80% of data and to be tested with 20% 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 Weka data mining tool and its programming language [29] have been employed. The AUC criterion and accuracy have been used for comparing the efficacy of the classifiers.

Table 1: Properties of datasets

Data Set	Size	Meta Attribute	Nominal Attribute	Binary Attribute	Numeral Attribute	Number of Categorize
1	1800	0	4	0	3	4
2	500	0	0	0	9	5
3	100	1	1	15	1	7
4	6300	3	2	5	20	40

Experimental Results

To gauge and investigate the performance on the selected classification methods, we use the same experiment procedure as suggested by WEKA. The 75% data is used for training and the remaining is for testing purposes.
In WEKA, all data is considered as instances and features in the data are known as attributes. The simulation results are partitioned into several sub items for easier analysis and evaluation. On the first part, correctly and incorrectly classified instances will be partitioned in numeric and percentage value and subsequently Kappa statistic, mean absolute error and root mean squared error will be in numeric value only. We also show the relative absolute error and root relative squared error in percentage for references and evaluation. The results of the simulation are shown in Tables 2 and 3 below. Table 2 mainly summarizes the result based on AUC-Curve. Meanwhile, Table 3 shows the result based on accuracy. Figure 2 is the graphical representations of the simulation result. Based on the above Figures 1, 2 and Table 1, we can clearly see that the highest accuracy is 89.71% and the lowest is 84.57%. The other algorithm yields an average accuracy of around 85%. In fact, the highest accuracy belongs to the Bayes network classifier, followed by Radial basis function with a percentage of 87.43%. Nearest neighbor bottom the chart with percentage around 84%. The total time required to build the model is also a crucial parameter in comparing the classification algorithm. From Figure 2, we can observe the differences of errors resultant from the training of the selected algorithms. This experiment implies very commonly used indicator which are mean of absolute errors and root mean squared errors. Alternatively, the relative errors are

also used. Since, we have two readings on the errors, taking the average value will be wise. It is discovered that the highest error is found in single rule conjunctive rule learner with an average score of around 0.3 where the rest of the algorithm ranging averagely around 0.2-0.28. An algorithm which has a lower error rate will be preferred as it has more powerful classification capability and ability in terms of medical and bioinformatics fields.

Table 2: AUC values for dataset

Data Set	K-NN	SVM	C4.5	NB
1	91.31	98.80	89.17	97.75
2	89.10	89.80	85.10	91.71
3	99.61	99.51	83.03	97.55
4	73.10	75.02	64.70	68.08

Table 3: Accuracy of classifiers' prediction

Data Set	K-NN	SVM	C4.5	NB
1	66.15	68.16	66.64	67.45
	66.3	66.1	63.69	67.00
	64.00	55.7	59.53	64.6
	56.5	46.2	49.9	49.76
2	93.22	93.2	91.39	91.39
	93.01	94.1	93.23	93.11
	92.43	93.72	91.73	89.97
	89.8	85.00	86.1	89.02
3	90.64	92.56	93.47	91.43
	91.91	88.00	91.32	88.35
	97	67.01	87.97	82.43
	90.01	39.09	71.96	65.74
4	80.18	95.76	91.62	84.71
	81.01	95.01	89.39	79.9
	81.9	91.2	84.41	83.1
	78.90	87.45	76.2	79.91

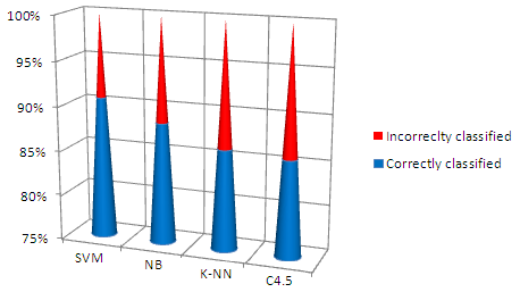

Figure2: Classification Results

Conclusion

As a conclusion, we have met our objective which is to evaluate and investigate selected classification algorithms based on Weka. The best algorithm based on the generated data sets is Bayes network classifier with an accuracy of 89.71% and the total time taken to build the model is at 0.19 seconds. Bayes network classifier has the lowest average error at 0.2140 compared to others.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. C4.5 as an implementation of that, show an efficient performance in all datasets. Two classifiers including k-NN and 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. From the results, it can be said that NB provides the best AUC between them. Ultimately, 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. As a future work, it is possible to compare the efficacy of other classifiers by using the current method and also using other evaluation criteria and applying classifiers on real datasets could be as open problems in this field.

References

1. Tom Johnsten and Vijay V. Raghavan, "Impact of Decision- Region Based Classification Mining Algorithms On Database Security",supported in part by a grant from the U.S. Department of Energy.
2. Hui Yin1, 2, Fengjuan Cheng2, Chunjie Zhou1, "An Efficient SFL Based Classification Rule Mining Algorithm", Proceedings of 2008 IEEE International Symposium on IT in Medicine and Education, pp. 969 – 972.
3. Xianneng Li, Shingo Mabu, Huiyu Zhou, Kaoru Shimada and Kotaro Hirasawa, "Analysis of Various Interestingness Measures in Classification Rule Mining for Traffic Prediction", SICE Annual Conference 2010 ,August 18-21, 2010, pp. 1969 – 1974.
4. Bing Liu, Wynne Hsu, and Yiming Ma, "Integrating classification and association rule mining, "Knowledge Discovery and Data Mining Integrating, pages 80-86, 1998.
5. Imre, P. et al. "Big Data Optimization at SAS," Edinburgh, SAS Institute, 2013.
6. M. Kantardzic, "Data Mining: Concepts, Models, Methods, and Algorithms," John Wiley & Sons Publishing, 2003.
7. I.H. Witten, E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques," Morgan Kaufmann Publishing, Second Edition, 2005.
8. Y.S. Kim, "Comparison 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.
9. 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.
10.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.
11. N.B. Amor, S. Benferhat, Z. Elouedi, "Naïve Bayes Vs Decision Trees in Intrusion Detection Systems," ACM Symposium on Applied Computing, Cyprus, 2004.
12.R. Entezari-Maleki, A. Rezaei, and B. Minaei-Bidgoli, "Comparison of Classification Methods Based on the Type of Attributes and Sample Size," Journal of Convergence Information Technology (JCIT), Vol. 4, No. 3, pp. 94-102, 2009.
13. 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.
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. S.R. Amendolia, G. Cossu, M.L. Ganadu, B. Golosio, G.L. Masala, G.M. Mura, "A comparative study of KNearestNeighbour, Support Vector Machine and MultiLayer Perceptron for Thalassemia screening," Journal of Chemometrics and Intelligent Laboratory Systems, Vol. 69, 2003, pp. 13– 20.
16. 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.
17. 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.
18. J. Han, M. Kamber, "Data Mining: Concepts and Techniques," Elsevier, Second Edition, 2006.
19. Bouckaert, R.R. (1994). Properties of Bayesian network Learning Algorithms. In R. Lopex De Mantaras& D. Poole (Eds.), In Press of Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence (pp. 102-109). San Francisco, CA.
20.Buntine, W. (1991). Theory refinement on Bayesian networks. In B.D. D'Ambrosio, P. Smets, & P.P. Bonissone (Eds.), In Press of Proceedings of the Seventh Annual Conference on Uncertainty Artificial Intelligent (pp. 52-60). San Francisco, CA.
21.Daniel Grossman and Pedro Domingos).Learning Bayesian Network Classifiers by Maximizing Conditional Likelihood. In Press of Proceedings of the 21st International Conference on Machine Learning, Banff, Canada.
22.T. Darrell and P. Indyk and G. Shakhnarovich (2006). Nearest Neighbor Methods in Learning and Vision: Theory and Practice. MIT Press.
23.J. R. Quinlan, "C4.5: Programs for Machine Learning", Morgan Kaufmann Series in Machine Learning, 1993.
24.J. R. Quinlan, "Improved Use of Continuous Attributes in C4.5", Journal of Artificial Intelligence Research, volume 4, pp. 77-90, 1996.
25.WEKA at http://www.cs.waikato.ac.nz/~ml/weka.

Morteza Okhovvat, Iran University of Science and Technology, School of Computer Engineering, Tehran, Iran.
Young Researchers and Elite Club, Sari Branch, Islamic Azad University, Sari, Iran
E-mail: okhovat@comp.iust.ac.ir
Hassan Nader, Iran University of Science and Technology, School of Computer Engineering, Tehran, Iran.
E-mail: naderi@iust.ac.ir