2012 International Conference on Industrial and Intelligent Information (ICIII 2012)
IPCSIT vol.31 (2012) © (2012) IACSIT Press, Singapore

Recent Trends in Data Classifications

Mohd Afizi Mohd Shukran [1], Mohammad Adib Khairuddin [2], Kamaruzaman Maskat [3]
[1,2,3] Faculty of Science and Defense Technology,
National Defense University of Malaysia Sungai Besi Camp,
57000 Kuala Lumpur, Malaysia.

Abstract. Data classification is a widely used technique in various fields, including data mining, whose goal is to classify a large set of objects into predefined classes, described by a set of attributes, using supervised learning methods. Due to the explosive growth of both business and scientific databases, extracting efficient classification rules from such databases is of major importance. However, such systems are not easy to develop because real-world databases usually contain very complex objects which make the design of similarity measures difficult. Thus, a new classification method using Swarm Intelligence is proposed. In this paper, we give a brief introduction to traditional classification techniques and comparisons between the traditional classification techniques.

Keywords: Data Mining, Classification, Machine Learning

1. Introduction to Data Classification

Data classification is one of the fundamental problems in data mining and knowledge discovery. In data classification, the goal is to learn a classifier from a given set of instances with class labels, which correctly assigns a class label to a test instance. The performance of a classifier is usually measured by its classification accuracy (the percentage of instances correctly classified). Traditional classification techniques have been extensively studied and various learning algorithms have been developed, such as Support Vector Machine (SVM), Nearest Neighbour (K-NN), Decision Trees, and Naïve Bayes. The next sections, Section 1.1 to Section1.4, will briefly describe these traditional classification techniques.

1.1. Support Vector Machine

There are many possible choices for an appropriate classifier. Among these, support vector machines (SVMs) would appear to be a good candidate because of their ability to generalise in high-dimensional spaces, without the need to add prior knowledge. The appeal of SVMs is based on their strong connection to the underlying statistical learning theory; that is, an SVM is an appropriate implementation of the structural risk minimisation method [1]. Moreover, SVM is an effective classifier that has been widely used in image classification. In intuition, a SVM constructs the Optimal Separating Hyperplane (OSH), which separates a set of positive from a set of negative samples with maximum margin. In the case where classes are not linearly separable, SVM use kernel functions to map the input space to high dimensional space. Fig. 1. shows how non-linear problems can be solved using Kernel function. Common nonlinear kernel functions are Radial Basis Functions (RBF), which can be used to do the nonlinear transform.

1.2. Nearest Neighbour (K-NN)

K-NN, as an instance based classification method, has been an effective approach to a broad range of pattern recognition and image classification. This algorithm is based on initially determining the K instances from the training samples that are closest to the pattern to be classified. The classification decision is chosen as the class to which the majority of these nearest neighbours belong. Moreover, the K-NN rule is widely used to classify the observation into the category, which only depends on a collection of correctly classified samples. The aim is to find the nearest K sample from the existing training data when a new sample appears, and classify the appeared sample according to the most similar class.

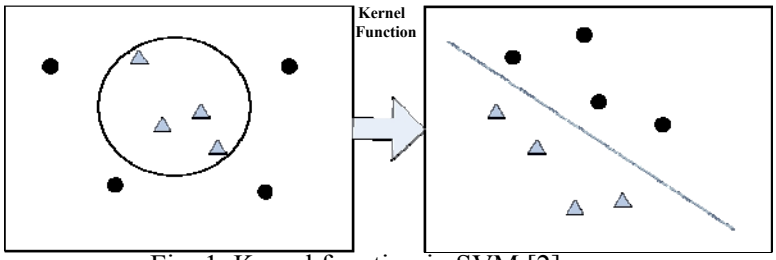

Fig. 1. Kernel function in SVM [2]

Generally, closeness is defined by Euclidean distance. Yidiz et al. [3] explained Euclidean distance precisely with a formula. Besides that, the K-NN algorithm simply retains the entire training set during learning [4]. During execution, the new input vector is compared to each instance in the training set. The class of the instance that is most similar to the new vector (using some distance function) is used as the predicted output class.

The nearest neighbour algorithm has several strengths when compared to most other learning models. Firstly, it learns very quickly. Secondly, it is guaranteed to learn a consistent training set (i.e. one in which there are no instances with the same input vector and different outputs) and will not get stuck in local minima. Thirdly, it is intuitive and easy to understand, which facilitates implementation and modification. Fourthly, it provides good generalisation accuracy on many applications. For example, see [5]. However, in its basic form the nearest neighbour algorithm has several drawbacks such as its distance functions are typically inappropriate for applications with both linear and nominal attributes. Secondly, it has large storage requirements, because it stores all of the available training data in the model. Thirdly, it is slow during execution, because all of the training instances must be searched in order to classify each new input vector.

1.3. Decision Tree

A Decision Tree, more commonly known as a Classification Tree, is used to learn a classification function which predicts the value of a dependent attribute (variable), given the values of the independent (input) attributes (variables). This solves a problem known as supervised classification, because the dependent attribute and the number of classes (values) that it may have are given [6]. The attractiveness of decision trees is due to the fact that, in contrast to neural networks, decision trees represent *rules*. Rules can readily be expressed so that humans can understand them, or even be directly used in a database access language like SQL so that records falling into a particular category may be retrieved. C4.5 is also a method for approximating discrete-valued functions in which the learned function is represented by a decision tree. Learned trees can also be represented as sets of if-then rules to improve human readability. These learning methods are among the most popular of inductive inference algorithms and have been successfully applied to a broad range of tasks from learning to diagnose medical cases to learning to assess credit risk of loan applicants. C4.5 is a heuristic, one-step, look ahead (hill climbing), non-backtracking search through the space of all possible decision trees [7-8].

1.4. Bayesian Method

Bayes theorem is an effective and simple method, and for this reason it is used frequently for classifying problems [9]. In machine learning, determining the best hypothesis from some space H, given the observed training data D is often explored. Bayes theorem provides a way to calculate the posterior probability $P(h|D)$, from the prior probability $P(h)$, together with $P(D)$ and $P(D|h)$ in (3) [7]:

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)} \tag{1}$$

$P(D)$ and $P(D|h)$ denotes the prior and the posterior probability of observed training data D, respectively. In this study, the classifying process is optimized by using the expectation maximization (EM) algorithm. The expectation maximization algorithm is a method that is used to guess units which have missing data and includes maximum similarity probabilities [10]. The EM method is a repeated method and it has two stages: the Expectation stage gives expectation for the data; the Maximization stage gives expectation about mean, standard deviation or correlation when a missing data is appointed. This process continues until the change on expected values decreases to a negligible value. Bayesian methods are based on probability calculus; the expectation maximization algorithm is one of them. It is utilised for unknown values with known probability

distributions. The radial basis function is a popular function for explaining probability distributions [7]. The expression of this function is given as in:

$$y = e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2}$$

In (4), x denotes training data, μ denotes mean of data and σ refers to variance. One way to form an expectation maximization algorithm is to guess the average value of Gauss functions. If we have a sample data set which has k different classes, it means that the data set is formed from a probability distribution which is a mixture of k different normal distributions. Each sample is formed with a two-step process. At the first step, a random normal distribution is chosen from k normal distributions; at the second step, a sample data is formed according to this distribution. These steps are repeated for each point in the set (Fig. 2.).

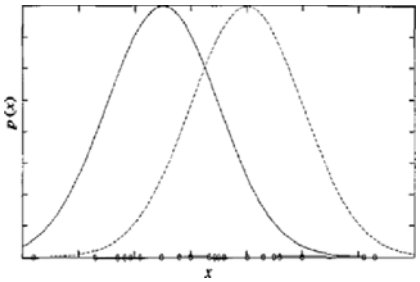

Fig. 2. The normal distributions when $k - 2$ and sample data are on the x-axis [11]

One of the most important Bayes approaches is the Naïve Bayes. The Naïve Bayes method is a method of classification applicable to categorical data, based on Bayes theorem. Careful analysis of the Bayesian classification problem has shown that there are some theoretical reasons for the apparently unreasonable efficacy of Naïve Bayes classifiers [12]. An advantage of the Naïve Bayes classifier is that it requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined, and not the entire covariance matrix.

2. Conclusion

When investigating the best parameter settings for the SVM, the linear kernel was found to be the best choice, which is consistent with previous work [13]. With respect to the best feature space size, SVM exhibited generally good performance for small or medium sizes, which surprises us as SVM is commonly said to best perform in very large feature spaces. In terms of overall performance, the K-NN classifier, the Naïve Bayes and the SVM perform similarly if suitable parameter settings are used. These results are in agreement with a study [14] showing that the set-up parameter influence more the performance than the individual choice of a particular learning technique. Therefore, one should keep considering the K-NN classifier and the Naïve Bayes classifier as possible options because they are fast, simple and well understood. Regarding SVM, perhaps it can handle better complex classification tasks, but it remains to be seen how we can identify them. Moreover, it is costly to train SVM. Results depend on the evaluation methodology and we have focused here on binary classification tasks. New experiments should be carried out to explain why the Naïve Bayes behave so well on one against one classification tasks in contrast to its behaviour on one against all tasks. We are also interested to understand more precisely SVM behaviour as it exhibited an uncommon performance pattern shaped as a wave when the size of the feature space increases. Finally, to recommend a classifier with suitable parameter settings, a way to characterize classification tasks should be investigated, eventually via the use of a meta-learning strategy.

3. References

[1] Vapnik, V., *The nature of Statistical Learning Theory*, ed. Springer-Verlag. 2000, New York USA: Springer-Verlag.

[2] Collobert, R. and S. Bengio, *SVMTorch: Support vector machines for large regression problems.* Machine Learning, 2001. **1**: p. 143-160.

[3] Yildiz, T., Yildirim, S., Altilar, D. T., *Spam filtering with parallelized KNN algorithm*. 2008: Akademik Bilisim.

[4] Cover, T.M. and P.E. Hart, *Nearest Neighbour Pattern Classification.* Institute of Electrical and Electronics Engineers Transactions on Information Theory, 1967. **13**(1): p. 21-27.

[5] Fogarty, T.C., *First Nearest Neighbor Classification on Frey and Slate's Letter Recognition Problem.* Machine Learning, 1992. **9**(4): p. 387-388.

[6] Osei-Bryson, K.-M., *Evaluation of decision trees: a multi-criteria approach.* Computers & Operations Research, 2004. **31**(11): p. 1933-1945.

[7] Mitchell, M.T., *Machine learning*. 1997, Singapore: McGraw-Hill.

[8] Quinlan, J.R., *Induction of decision trees.* Machine Learning, 1986. **1**: p. 81-106.

[9] Gungor, T., *Developing dynamic and adaptive methods for Turkish spam messages filtering*, in *Technical report 04A101*. 2004, Bogazici University Research Fund.

[10] Friedman, N., *The Bayesian structural EM algorithm*, in *Proceedings of the 14th conference on uncertainty in artificial intelligence (UAI '98)*. 1998. p. 129-138.

[11] John, G.H. and P. Langley, *Estimating continuous distributions in Bayesian classifiers*, in *Proceedings of the 11th conference on uncertainty in artificial intelligence*. 1995, Norgan Kaufman: Sam Mateo. p. 338-345.

[12] Zhang, H. *The optimality of Naives Bayes*. in *17th International Florida Artificial Intelligence Research Society*. 2004. Florida, USA.

[13] Dong, J.X., Krzyzak, A., Suen, C. Y., *Fast SVM training algorithm with decomposition on very large data sets.* IEEE Transaction on Pattern Anal. Machine Intelligence, 2005.

[14] Gao, D., Zhou, J., Xia, L., *Svm-based detection of moving vehicles for automatic traffic monitoring*, in *IEEE Intelligent Transportation Systems Conference Proceeding*. 2001. p. 745-749.