*Proceedings of 2012 4th International Conference on Machine Learning and Computing*
***IPCSIT vol. 25 (2012) © (2012) IACSIT Press, Singapore***

# Nearest Neighbour Classifier Accuracy Improvement Using Enhanced Particle Swarm Optimization

Hassan M. Elragal+

Electrical Engineering Department, Faculty of Engineering, University of Bahrain

**Abstract.** Feature selection/scaling process can be considered as a global optimization problem in machine learning. Feature selecting can reduce number of features, remove irrelevant data, and results in acceptable classification accuracy. Feature scaling can be used to increase separability between different classes and results in increasing classification accuracy. In this paper, a new approach is presented in which feature selection/scaling and classifier training are performed simultaneously using an enhanced version of particle swarm optimization algorithm (EPSO). A hybrid system consists of two stages is proposed. In the first stage, EPSO is used to optimize a feature weight vector that is used to select/scale input features. A nearest neighbor classifier is used to evaluate each set of feature weights. In the second stage, the selected or scaled input features are used as input to another nearest neighbour classifier. The proposed method is applied to six classification problems. Experimental results show that higher classification accuracy can be obtained using the proposed method.

**Keywords:** Feature selection, nearest neighbor classifier, classification.

## 1. Introduction

Classification algorithms seek to minimize classification error rate or maximize the classifier's ability to discriminate between classes. There are two different types of measures that are commonly used within the area of classification, accuracy and class separability. Accuracy is the primary measure for evaluation. Typical classifiers include Bayesian classifiers, maximum likelihood classifiers and minimum distance classifiers [1]. During classification, problems can arise when some of the input features are irrelevant and/or redundant. This means that, these input features do not add any new information to the description of the data structure. These issues become a concern when the input space is quite large.

Feature extraction and feature subset selection are two general approaches to input space reduction. Feature extraction is the process of extracting a set of new features from the original set of features through a mapping or transformation, Feature subset selection is an optimization problem to reduce number of irrelevant features while maintaining acceptable classification accuracy. Feature selection is of considerable importance in pattern classification, data analysis, medical data processing, machine learning and data mining applications. Most feature selection techniques are characterized either as filters, which ignore the classifier to be used, or as wrappers, which base selection directly on the classifier.

The filter approaches make use of the statistical information of the data set to carry out feature selection and is independent of the classification system. These approaches depend on the definition of a relevant measure, e.g. mutual information between outputs and inputs, similarity measure between features and class separability [2][3]. Filtering approaches require users to determine the number of features being selected, rather than providing stopping criteria. Furthermore, the classifier accuracy is not a consideration of the filtering feature selection methodologies, even though it is the ultimate goal of building classification systems.

+ Corresponding author. Tel.: + 973 3979 5284.
*E-mail address*: helragal@eng.uob.bh.

The wrapper methodology combines both the feature selection and output of the classification system into a single system. Therefore, they generally give better classification accuracy than filter approaches. The leave-one-out wrapper methodology can be applied using support vector machines (SVM) as they usually have better classification accuracy [4]. There are two major drawbacks of the wrapper approaches: high complexity and non-scalability to large inputs datasets.

Optimal feature selection methods identify the optimal feature subset by an exhaustive search of the solution space of all possible subsets [5]. For a very large number of features, exhaustive search is computationally intractable. The branch and bound method produces optimal features, but it requires a performance measure to be monotonic [6]. This condition often cannot be satisfied. Genetic algorithms were introduced for the selection of features in [7]. Tabu search (TS) metaheuristic was shown as a promising approach in [8]. A comparative study of several of the well-known optimal and sub-optimal feature selection algorithms is shown in [9].

Feature weights can be introduced to scale features based on their relative significance within the feature subset [10]. Relief algorithm is developed in [11], which assigned weight to each feature and use nearest neighbour algorithm to update the weights.

A hybrid system consists of two stages is proposed in this paper. In the first stage Enhanced Practical Swarm Optimization (EPSO) algorithm is used for either select or scale input features. Nearest neighbour classifier is used to evaluate the fitness function values needed for EPSO. In the second stage, the selected or scaled input features are used as input to another nearest neighbour classifier.

The rest of this paper is organized as follows: Section 2 describes enhanced particle swarm optimization (EPSO). Section 3 discuses a k-nearest neighbour (k-NN) classifier. Section 4 describes how input feature can be selected or scaled using EPSO. Section 5 shows other evolutionary algorithms for selection and scaling process. Section 6 shows experimental results. Concluding remarks are included in Section 7.

## 2. Enhanced Particle Swarm Optimization (EPSO)

Particle swarm optimization (PSO) is a swarm intelligence method for global optimization [12]. For classical PSO, each individual of the population adjusts its trajectory toward its own previous best position, and toward the previous best position attained by any member of its topological neighbourhood. Enhanced practical swarm optimization (EPSO) algorithm is similar to classical PSO with improved global search ability [13]. This is accomplished by introduce an updating formula for global best particle position and adding two new terms in the velocity updating formula.

- Update global best particle position $p^g(k)$ at time $k$ +1 as:

$$p^g(k+1) = [1 + \lambda . U]\, p^g(k)$$

where $U$ is a Gaussian random number with zero mean and unit variance and $\lambda$ is a convergence acceleration parameter.

- Update velocities of all particles $v^i(k)$ at time $k$ +1 as follows:

$$\begin{aligned} v^i(k+1) = w.v^i(k) + c_1.r_1.(p^i(k) - x^i(k)) + c_2.r_2.(p^g(k) - x^i(k)) \\ + c_3.r_3.(p^i_{fdr}(k) - x^i(k)) + c_4.r_4.(p^g_{fdr}(k) - x^i(k)) \end{aligned}$$

where, $r_1$, $r_2$, $r_3$ and $r_4$ are uniformly distributed random variables in [0, 1], $w$ is inertia factor, $c_1$ is self confidence factor, $c_2$ is swarm confidence factor, $c_3$ and $c_4$ are acceleration constants, $p^i_{fdr}(k)$ and $p^g_{fdr}(k)$ are local and global candidate positions that are selected by locating the individual with minimum fitness to distance ratio (FDR) over all particles in the swarm [13].

### 3. K-Nearest Neighbor Classifier (K-NN)

The K-nearest neighbor classifiers (K-NN) is a simple and powerful nonparametric classification technique [1]. Its asymptotic classification error is bounded by twice of Bayes error. In the basic one nearest neighbor rule classifier, each training sample is used as a prototype and a test sample is assigned to the class of the closest prototype [14]. The 1- nearest neighbor approach can be extended to the K nearest neighbor's method. In this case, instead of taking the class of the nearest neighbor, K (where K is an integer positive number) nearest neighbors are determined, and the class is taken as that exhibited by the majority of these neighbors. Theoretically, it can be shown that for a very large number of samples, there are advantages in using large values of K. More specifically, if K tends to infinity, the performance of the K-neighbors method approaches the Bayes rate.

## 4. Input Feature Selection/Scaling using EPSO

This section will describe the proposed classification system. A hybrid system consists of two stages. EPSO is used in the first stage for either select or scale input features. 1-NN classifier is used in the first stage to evaluate the fitness function values needed for EPSO. In the second stage, the selected or scaled input features are used as inputs to another 1-NN classifier. Fig. 1 shows a block diagram for the proposed classification system with input feature selection/scaling using EPSO.

**In the 1st stage:**

Step 1: Divide each database into three parts (training, validation and test data sets).

Step 2: Create a population of *N* particles (selecting or scaling vectors) (each particle has a length M, where M is the number of input features). In case of selecting, particle vectors are ones and zeros (one for select and zero for not select). In case of scaling, particle vectors are floating numbers.

Step 3: Apply each particle to the training, validation and test data set to create new data sets.

Step 4: Test the new validation data set to select 1-NN from new training data set and calculate the corresponding fitness value (classification accuracy).

Step 5: According to fitness value, apply EPSO algorithm and produce new population.

Step 6: Repeat steps 3 to 5 until a solution results in perfect classification of all new validation data set or until a maximum number of fitness function evaluation is reached.

Step 7: Record the best performing particle (the particle with the maximum classification accuracy)

**In the 2nd stage:**

Step 1: Recall the best performing particle from the first stage.

Step 2: Apply the best performing particle to the original training, validation and test data sets to create new data set.

Step 3: Use 1-NN classifier to blind test on the new testing data set from the new training and new validation data sets together.

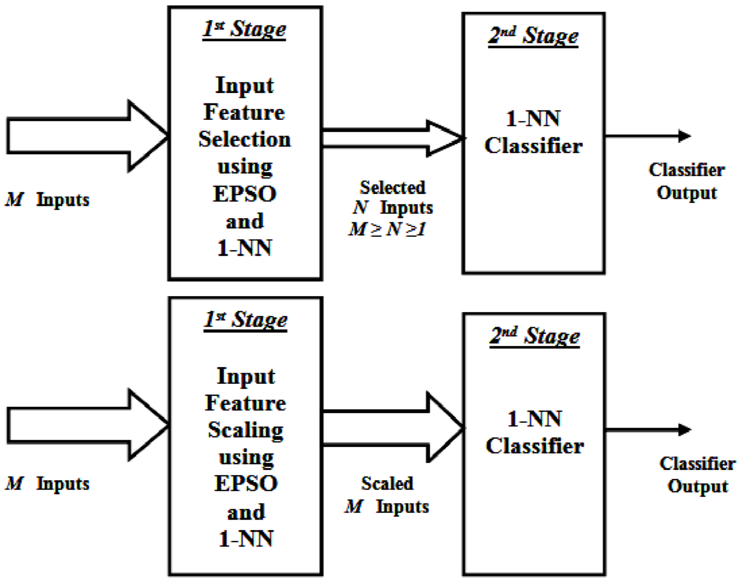


Fig 1. The proposed classification system with input feature selection/scaling using EPSO

## 5. Comparison with other Eolutionary Otimization Tchniques

The proposed hybrid system is optimized in its 1st stage with EPSO optimization technique. Two other commonly used evolutionary optimization techniques will be used to optimize the selection / scaling process in the first stage. The first one is the genetic algorithm (GA) and the second one is the differential evolving (DE) algorithm.

Genetic Algorithms (GAs) use the concept of Darwin's theory of evolution "survival of the fittest". The idea starts by creating different possible solutions to a problem. These solutions are then tested for their performance. Among all possible solutions, a fraction of the good solutions is selected. The selected solutions undergo the process of reproduction, crossover, and mutation to create a new generation. This process of production of a new generation and its evaluation is repeated until there is a convergence within a generation.

Differential Evolution (DE) is a parallel direct search heuristic approach. The initial vector population is chosen randomly. DE generates new parameter vectors by adding the weighted difference between two population vectors to a third vector (mutation). The mutated vector's parameters are then mixed with the parameters of another predetermined vector (target vector) to yield the so-called trial vector (crossover). If the trial vector yields a lower cost function value than the target vector, the trial vector replaces the target vector in the following generation (selection). The process of mutation, crossover and selection, are repeated until there is a convergence within a generation.

## 6. Experimental Results

The performance of the proposed classifiers is studied using six widely used real-world databases as [15].

| **Data Set** | Number of inputs | Number of classes | Number of data in each class |
|---|---|---|---|
| **Iris** | 4 | 3 | 50,50,50 |
| **Phoneme** | 5 | 2 | 3818,1586 |
| **Diabetes** | 8 | 2 | 268,500 |
| **Heart** | 13 | 2 | 139,164 |
| **Wine** | 13 | 3 | 59,71,48 |
| ***Radar*** | 33 | 2 | 225,126 |

Table 1 shows the overall classification accuracy for each data set. The first column is without applying selecting or scaling strategy. The next three columns are with applying selecting strategy using EPSO, GA and DE algorithms. The last three columns are with applying scaling strategy using EPSO, GA and DE algorithms.

Based on blind testing and applying selecting/scaling strategy on the six data sets, it can be seen that:

- Based on applying selecting strategy, optimization with EPSO shows consistency of getting the same classification accuracy (when number of inputs are small <10 inputs) or improved classification accuracy (when number of inputs are large) for all six data sets.
- Advantage of using selecting inputs with EPSO strategy appears clearly when number of inputs is large resulting in a simple classifier.
- Based on applying scaling strategy, optimization with EPSO shows consistency of improving classification accuracy for all six data sets irrespective of number of inputs.
- When number of inputs are small (< 10 inputs), applying scaling inputs gives higher classification accuracy than applying selecting inputs for one optimization method (EPSO).

## 7. Conclusion

In this paper a new approach is presented in which feature selection/scaling and classifier training are performed simultaneously using an enhanced version of particle swarm optimization algorithm (EPSO). A hybrid system consists of two stages is proposed. In the first stage, EPSO is used to optimize a weight vector which is used to select/scale input features. A 1- NN classifier is used to evaluate each set of feature weights. In the second stage, the selected or scaled input features are used as input to another 1-NN classifier. The proposed method is applied to six classification problems. Experimental results reveal that the proposed method resulted in a higher accuracy compared to the same classifier without selection/scaling strategy.

Table 1 Classification accuracy of a 1-NN classifier with selecting and scaling strategy

| | | Selection Strategy | | | Scaling Strategy | | |
|---|---|---|---|---|---|---|---|
| | **Original** | **EPSO** | **GA** | **DE** | **EPSO** | **GA** | **DE** |
| **Iris** | 96.19<br>4 Inputs | 96.19<br>4 Inputs | 96.19<br>4 Inputs | 96.19<br>4 Inputs | 100 | 100 | 100 |
| **Phoneme** | 84.95<br>5 Inputs | 84.95<br>5 Inputs | 84.95<br>5 Inputs | 84.95<br>5 Inputs | 85.95 | 85.60 | 86.08 |
| **Diabetes** | 63.74<br>8 Inputs | 64.21<br>5 Inputs | 64.21<br>5 Inputs | 64.21<br>5 Inputs | 64.27 | 61.50 | 63.44 |
| **Heart** | 60.30<br>13 Inputs | 75.80<br>7 Inputs | 75.80<br>7 Inputs | 75.80<br>7 Inputs | 73.21 | 66.40 | 69.45 |
| **Wine** | 77.28<br>13 Inputs | 97.17<br>7 Inputs | 92.87<br>4 Inputs | 92.87<br>4 Inputs | 98.92 | 97.85 | 98.92 |
| **Radar** | 84.32<br>33 Inputs | 87.78<br>7 Inputs | 83.72<br>12 Inputs | 83.01<br>14 Inputs | 88.98 | 85.03 | 81.69 |

## 8. Reference


[1] R. O. Duda and P. E. Hart. Pattern Classification and Scene Analysis. New York: Wiley-Interscience Publication, 1973.

[2] N. Kwak & C-H Choi. Input Feature Selection by Mutual Information Based on Parzen Window, IEEE Trans. On PAMI, vol. 24, no. 12, 2002, PP 1667–1671.

[3] P. Mitra, C. A. Murthy & S. K. Pal, Unsupervised Feature Selection Using Feature Similarity. IEEE Trans. on PAMI, vol. 24, 2002, PP 301–312.

[4] J. Bi, K. P. Bennett, M. Embrechts, C. M. Breneman & M. Song. Dimensionality Reduction via Sparse Support Vector Machines, Journal of Machine Learning Research, vol. 3, 2003, PP 1229–1243.

[5] M. R. Garey and D. S. Johnson. Computers and Intractability, A Guide to the Theory of NP-Completeness. San Francisco: W.H. Freeman and Company, 1979.

[6] H. Liu, H. Motoda, and M. Dash, “A monotonic measure for optimal feature selection,” in European Conference on Machine Learning, 1998, PP 101-106.

[7] M.L. Raymer, W.F. Punch, , E.D. Goodman, L.A. Kuhn, and A. K. Jain. Dimensionality Reduction Using Genetic Algorithms. IEEE Trans. Evolutionary Computation, vol. 4, no. 2, July 2000, PP 164-171.

[8] H. Zhang and G. Sun. Feature Selection Using Tabu Search Method. Pattern Recognition, vol. 35, issue 3, Mar. 2002, PP 701-711.

[9] M. Kudo and J. Sklansky. Comparison of algorithms that select features for pattern classifiers. Pattern Recognition, vol. 33, issue 1, Jan. 2000, PP 25-41.

[10] W. Duch, K. Grudzinski. Search and global minimization in similarity-based methods. Int. Joint Conf. on Neural Networks. 1999 paper no. 742 Washington.

[11] Caruana, R. Freitag, D.: Greedy Attribute Selection. Proc. 11th Int. Conf. on Machine Learning, 1994, PP 26–28.

[12] J. Kennedy and R. Eberhart. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Piscataway, NJ, 1995.PP 1942–1948.

[13] M. A.-A. Mangoud and H. M. Elragal. Antenna Array Pattern Synthesis and Wide Null Control Using Enhanced Particle Swarm Optimization. Progress In Electromagnetic Research B, PIERB 17, 2009, PP, 1-14.

[14] B. V. Dasarathy. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, 1991.

[15] P.M. Murphy, and D.W. Aha, “UCI Repository of Machine Learning Databases,” technical report, Department of Information and Computer Science, University of California, Irvine, Calif., 1994.