International Journal of Advanced Engineering and Management Research
Vol. 3 Issue 6; 2018

www.ijaemr.com ISSN: 2456-3676

HYBRID DIMENSIONALITY REDUCTION METHOD BASED ON SUPPORT VECTOR MACHINE AND DIFFERENTIAL EVOLUTION ALGORITHM

*Chun-Liang Lu[1] and Chih-Hsu Hsu[2]

[1,2]Department of Healthcare and Applied Informatics, Ching Kuo Institute of Management and Health, Keelung County, Taiwan.
*Corresponding author:

Abstract

Support vector machine (SVM) has achieved excellent performances in the classification of hyper spectral data and wide variety of applications. Nevertheless, how to effectively reduce the complexity features of training dataset for SVM is still a serious challenge. In this paper, an efficient scheme of differential evolution (DE)-based dimensionality reduction approach is proposed to use a simple searching criterion function, called adaptive estimated nearest neighbors (AENN), to optimize the reduction of noisy instances in a classification process for the SVM. With such an efficient criterion, DE algorithm can find a global optimal solution for the AENN and SVM kernel parameters to improve the classification accuracy. Several UCI benchmark datasets are considered to compare the proposed hybrid DE-AENN dimensionality reduction strategy with the previously published methods. Experimental results show that the proposed hybrid framework is capable of achieving better performance than other existing methods and is feasible to construct a condensed nearest neighbors of training dataset to enhance the classification accuracy for SVM.

Keywords: Differential Evolution (DE) algorithm, Adaptive Estimated Nearest Neighbors (AENN), Support Vector Machine (SVM).

Introduction

In recent years, the datasets collected for some specific domains, such as content-based image retrieval and microarray gene expression of cancer diagnosis, tend to be high-dimensional and complex to handle. The high dimensional data are composed of a large number of features while many of them are irrelevant or over-sensitivity to noise, and led to the patterns are not easily recognized and degrade the data mining performance. Consequently, dimensionality reduction techniques, as a data pre-processing task has become one of the most important steps to extract essential information from knowledge discovery in databases (KDD) [1]. Considering an efficient general feature selection method for the curse of dimensionality [2] in mining procedure is still an open problem and the cooperation among features is the most important challenge. The feature selection mechanism retains the physical interpretability property in terms of the selected features and falls into four categories: filter [3], wrapper [4], embedded [5], and ensemble [6]. The filter only employs the intrinsic properties of data, which generally not involves learning algorithm on the original features and make it have lower time complexity. Wrapper methods usually achieve better results than filters since they search for a subset of features that are best suited to the classifier and very computationally intensive. An Embedded method its feature selection performs together with training of classifier, but the computational time is smaller than wrapper. The ensemble method evaluates the importance of features by filters to produce a set of features subsets, and then generates an aggregated result from these subsets.

The support vector machine (SVM) was first proposed by Vapnik [7] along with other researchers, which maps the original data into a higher dimensional space to linearly separate the data in the nonlinear feature domain, has shown excellent performances in the classification and been widely studied and applied in many fields. The SVMs based on empirical risk minimization are supervised learning algorithms that can be used to maximize the distance of classification boundary between two classes. Due to the data sets that we process today are becoming increasingly larger, not only in terms of the number of instances, but also the dimension of features, which may degrade the efficiency of most learning algorithms, especially when there exist noisy data or irrelevant features. Meanwhile, the choice of the kernel function and how the kernel parameters are set also affect the general izability of SVM classifier. SVM feature selection and parameter setting are critical to improved classification performance by using heuristic search strategy. Differential evolution (DE) algorithm proposed by Storn and Price [8] which has the advantage of high efficiency, rapid convergence, and strong capability for global search. It is a competitive evolutionary computing technique and shows excellent global optimization ability via population search and information exchange between individuals. As a result, the essential training instances extraction, feature selection and kernel parameters setting must perform simultaneously to enhance the SVM classifier performance [9].

In this paper, an efficient scheme of DE based dimensionality reduction approach is proposed to use a simple searching criterion function, called adaptive estimated nearest neighbors (AENN), to optimize the reduction of noisy instances and deal with the problem mentioned above for SVM. Short communications of the early stages of this work have appeared in [9]. Here we significantly extend our approach to integrate the DE algorithm with the AENN and SVM kernel parameters to efficiently select relevant instances during the SVM training process. Several UCI benchmark datasets are conducted to compare the proposed DE-AENN dimensionality reduction strategy with the previously published methods and the simulation results show the proposed framework can achieve better performance than other existing approaches in literature. The rest of the paper is organized as follows. In section 2, the related works including SVM and basic concept of DE algorithm are described. The adaptive estimated nearest neighbors scheme, particle representation of DE and the proposed hybrid DE-AENN framework for SVM classifier are illustrated in section 3. Experiment results are provided in section 4. Finally, conclusions are made in section 5.

Related Works

Support vector machine (SVM) classifier

The main concepts of SVM are to first transform input data into a higher dimensional space by means of a kernel function, and then to be trained to classify different categories of data from various disciplines. The goal of SVM is to minimize an upper bound of the generalization error and find the unique hyper plane having the maximum margin that can linearly separate the two classes of the data. For a given training dataset $\{(x_1, y_1),...,(x_m, y_m)\}$, where $x_i \in R^d$ and $y_i \in \{-1,+1\}$.

The generalized linear SVM finds an optimal separating hyper plane $f(x)=\langle w\cdot x\rangle+b$ by solving the following optimization problem [9]:

$$\underset{w,b,\xi}{\text{Minimize}} \quad \frac{1}{2}w^T w+C\sum_{i=1}^{m}\xi_i$$

$$\text{Subject to: } \begin{array}{l} y_i(< w\cdot x_i >+b)+\xi_i-1\geq 0 \\ \xi_i \geq 0 \end{array}$$

where C is a penalty parameter on the training error, and ξ_i is the non-negative slack variables. The simple kernel is the inner product function $K(x,\hat{x})=\langle x,\hat{x}\rangle$ which produces the linear decision boundaries. Nonlinear kernel function maps data points to a high-dimensional feature space as linear decision spaces. There are three commonly used kernels as follows: Linear kernel $K_L(x,\hat{x})$, Polynomial kernel $K_P(x,\hat{x})$ and the Radial basis function (RBF) kernel $K_R(x,\hat{x})$. The integer polynomial order γ in K_P and the width factor λ in K_R are hyper-parameters which are tuned to a specific classification problem.

$$K_L(x,\hat{x})=x^T\hat{x} \tag{2}$$

$$K_P(x,\hat{x})=\left(1+\langle x,\hat{x}\rangle\right)^{\gamma} \tag{3}$$

$$K_R(x,\hat{x})=e^{-\lambda\|x-\hat{x}\|^2} \tag{4}$$

Basic concept of differential evolution (DE) algorithm

DE algorithm is a population-based for solving constrained optimization problems and other complex real-world applications. The core of DE algorithm is the mutation operation, which uses a linear combination of a base vector and weighted random vector to generate a mutated vector with the better trial vector in next generation. The main steps of the DE are summarized as follows [10].

For the "*DE/rand/2/bin*" classical mutation strategy, five different vectors consisting of a base vector ($\vec{X}_{r1,G}$) and four difference vectors ($\vec{X}_{r2,G}$, $\vec{X}_{r3,G}$, $\vec{X}_{r4,G}$ and $\vec{X}_{r5,G}$) are randomly chosen from the populations. The both scale factors F_i and F_j are constants and the effective range to be usually taken from the range between 0.5 and 1. The donor vector $\vec{V}_{i,G}$ is expressed as Eq. 5.

$$\vec{V}_{i,G}=\vec{X}_{r1,G}+F_i(\vec{X}_{r2,G}-\vec{X}_{r3,G})+F_j(\vec{X}_{r4,G}-\vec{X}_{r5,G}) \tag{5}$$

, where $r1$, $r2$, $r3$, $r4$, $r5$ are random and mutually different integers, and they are also different with the vector index i, $\vec{X}_{i,G}=[x_{i,G}^1,x_{i,G}^2,\cdots,x_{i,G}^D]^T$ is the target vector, and $\vec{V}_{i,G}=[v_{i,G}^1,v_{i,G}^2,\cdots,v_{i,G}^D]^T$ is the donor vector. *G* indicates the current generation, and *D* is the dimension of the parameter. This mutation operation enables DE to explore the search space and maintain diversity. To enhance quality of solution, the crossover process creates the temporary trial vector $\vec{U}_{i,G}=[u_{i,G}^1,u_{i,G}^2,\cdots,u_{i,G}^D]^T$, $i=(1,2,\cdots,N_P)$ and is realized between each pair of target vector $\vec{X}_{i,G}$ and its corresponding donor vector $\vec{V}_{i,G}$. The scheme can be simply formulated for the binomial uniform crossover that is performed on each of the *j*-th variables whenever a randomly picked number (between 0 and 1) is less than or equal to a crossover rate C_r.

$$u_{i,G}^j=\begin{cases} v_{i,G}^j & if\,(rand(0,1)\leq C_r \; or \; j=j_{rand}, \\ x_{i,G}^j & otherwise, \end{cases} \quad j=1,2,...,D \tag{6}$$

,where $u_{i,G}^{j}$, $v_{i,G}^{j}$ and $x_{i,G}^{j}$ are trial, donor and target vectors from the *i*-th vector, *j*-th dimension at *G*-th generation. C_r is a user-defined probability in the range [0, 1]. Further, C_r is an inheritance quantity control parameter for how many percentages of elements cloning from a donor vector to a trial vector. The variable j_{rand} is a randomly chosen integer in the range [1, D], which ensures that the trial vector does not duplicate the target vector. The trial vector $\vec{U}_{i,G}$ is evaluated and then enters into the selection procedure which is achieved from the target and trial vectors by comparing their fitness values to select the better individual, and the better one will enter into the next generation. In the current population, target vector is updated when the newly generated trial vector gets better fitness value than its target vector; otherwise, the target vector is retained in the population.

$$\vec{X}_{i,G+1} = \begin{cases} \vec{U}_{i,G} & if\,(f(\vec{U}_{i,G}) \le f(\vec{X}_{i,G}) \\ \vec{X}_{i,G} & otherwise. \end{cases} \tag{7}$$

The DE algorithm works through a simple cycle of these stages: (1)Basic setting of parameters for DE, (2) Initialization to generate the initial population, (3)Repeat the steps Mutation, Crossover, Selection until a stopping criterion is reached.

Methods

In this section, an adaptive estimated nearest neighbors (AENN) scheme has been proposed to optimize the reduction of noisy instances in a classification process. Second, the particle representation of DE algorithm comprised of kernel function parameters, adaptive estimated rate and features mask are illustrated. Finally, the proposed DE-AENN framework which hybridizes the instance extraction, feature selection, and parameters optimization approach with DE optimization technique has been presented.

The adaptive estimated nearest neighbors (AENN) scheme

The effectively AENN scheme that we previously published in [9] is extended to decide which instances of the training data set are extracted for support vectors. The adaptively extracted instances coefficient θ_{AENN} for data reduction is flexible to edit out noisy data for merging essential points and make the SVM less sensitive to outliners. The adaptive coefficient $\theta_{AENN} \in [0,1]$ is defined as the ratio of the number with extracted instances to be support vectors for overall training dataset. The objective of the pre processing is to select the amount of data that consist of important information to be processed in the SVM. The AENN scheme is described as follows.

Step 1: **Initialization**: Randomly generating to initialize the candidate instances set.

Step 2: **Merge**: To decide whether extracted points have been achieved the threshold θ_{AENN}. If so, terminate the process; otherwise, go to Step 3.

Step 3: **Expand**: If there are any un-merged instances, using the nearest neighbor voting to renew the merged instances for support vectors; otherwise, no more new data is extracted and go to Step 2.

Particle representation of DE algorithm

In this work, the kernel function used is based on the RBF for the SVM classifier to implement our proposed method. The RBF kernel function requires that only two parameters, C and γ should be set. Using the adaptively estimated rate for AENN and the RBF kernel for

SVM, the parameters θ_{AENN}, C, γ and features used as input attributes must be optimized simultaneously for our proposed DE-AENN hybrid strategy. The particle therefore, is comprised of four parts, θ_m, C, γ and the features. Figure 1 show the particle representation of our design [9] extended to DE optimization framework. As shown in Figure 1, the representation of particle i with dimension of $n_f + 3$, where n_f is the number of features that varies from different datasets. $x_{i,1} \sim x_{i,n_f}$ are listed the features mask, x_{i,n_f+1} indicates the parameter value θ_{AENN}, the SVM penalty parameter on the training error value C is denoted as x_{i,n_f+2} and the x_{i,n_f+3} stands for the SVM RBF kernel function parameter value γ.

	Input features mask						θ_{AENN}	C	γ
DE particle representation	$x_{i,1}$	$x_{i,2}$	$\ldots$	$x_{i,d}$	$\ldots$	x_{i,n_f}	x_{i,n_f+1}	x_{i,n_f+2}	x_{i,n_f+3}

Figure 1: The particle representation of DE algorithm.

The proposed hybrid DE-AENN framework for SVM classifier

Figure 2 shows the system architecture of our proposed hybrid DE-AENN framework for SVM classifier. In this work, both the effective AENN scheme that we previously published, and the dimensionality reduction are successfully merged into DE algorithm to enhance the SVM classification accuracy are shown. Based on the DE particle representation and AENN optimization approach mentioned above, detailed descriptions of the hybrid DE-AENN procedure are presented as follows.

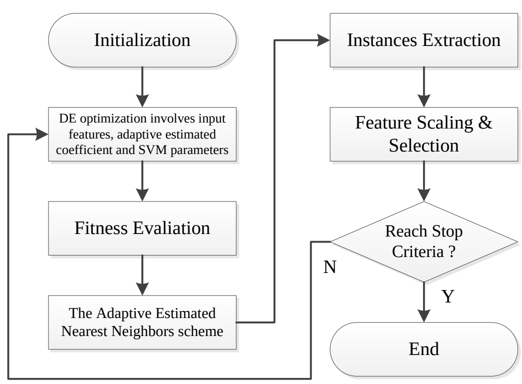

Figure 2: The system architecture of hybrid DE-AENN framework for SVM classifier.

The DE-AENN procedure: The proposed hybrid DE-AENN framework for SVM classifier.

01: Basic setting of parameters for the DE algorithm.

02: **Initialization** randomly generates initial particles comprised of the features mask, θ_{AENN}, C, and γ.

03: Evaluate the fitness value for each DE individual population.

04: **while** Termination condition is not satisfied **do**

05: **The AENN scheme phase**:

06: Generate the *n* candidate solutions, where $n \in m$ overall dataset.

07: **for** i=1 to m **do**

08: **Initialize**: Randomly generating to initialize the candidate instances set.

09: **Merge**: Check whether extracted points achieved the threshold θ_{AENN} . If so, terminate the process; otherwise, go to Line 10.

10: **Expand**: Check if any un-merged instances to renew the merged instances; otherwise, go to Line 9.

11: **end for**

12: **The DE optimization phase**:

13: **for** i=1 to N_p **do**

14: **Evaluate** the fitness for each individual $N_i \in N_p$ population.

15: **Mutation** operation.

16: **Crossover** operation.

17: **Selection** operation.

18: **end for**

19: **end while**

20: **Output** Global optimum solution to the features mask, θ_{AENN} , C , and γ for SVM.

Experiment Results

To measure the performance of the developed hybrid approach, several real benchmark data sets cited from the UCI Machine Learning Repository [9] were used: the Australian data set, the German data set, the Heart disease data set, the Iris data set, the Vehicle data set, and the Vowel data set. Table 1 summarizes the number of classes, instances, nominal attributes, numeric attributes and total attributes. The datasets considered are partitioned using the 10-fold cross validation. Each initial data set S , is randomly divided into ten disjoint sets of equal size T_1, T_2, …, T_{10}. Our proposed hybrid DE-AENN framework has been tested fairly extensively and compared with the GA based method [9] using three criteria such as the classification accuracy rate, the number of selected feature, and the non-parametric Wilcox on signed rank test. In all of the experiments, 10-fold cross validation was used to estimate the classification accuracy of each learning algorithm.

Table 1. Datasets from the UCI Machine Learning Repository.

Names	Classes	Instances	Nominal Attributes	Numeric Attributes	Total Attributes
Australian	2	690	6	8	14
German	2	1000	0	24	24
Heart	2	270	7	6	13
Iris	3	150	0	4	4
Vehicle	4	940	0	18	18
Vowel	11	990	3	10	13

Table 2. Comparison of DE-AENN strategy and GA based method on six UCI datasets.

Name	GA based method [9]		DE-AENN strategy		*p*-value
	Avg_A_{cc}	Avg_n_f	Avg_A_{cc}	Avg_n_f	
Australian	90.3±1.42	4.3±0.82	91.2±1.31	4.0±0.63	0.025*
German	87.4±0.97	12.6±0.84	88.7±0.65	12.2±0.76	0.021*
Heart	96.1±2.32	5.0±1.15	96.9±2.15	5.0±0.13	0.007*
Iris	100±0	1±0	100±0	1±0	1.0
Vehicle	85.9±1.85	8.6±0.97	87.6±2.23	8.2±1.30	0.006*
Vowel	99.2±0.75	7.1±0.74	99.7±0.56	7.0±0.63	0.009*

Table 2 shows the summary results for the average classification accuracy rate of the DE-AENN hybrid framework and GA based method on six UCI datasets. In Table 2, the classification accuracy rate is represented as the form of 'average ± standard deviation'. To highlight the advantage, we used the non-parametric Wilcox on signed rank test for all of the datasets. It is observed that the p-values of DE-AENN versus GA based method [9] are smaller than the statistical significance level of 0.05 except Iris dataset. The performance of both PSO and GA optimization techniques in terms of the average classification accuracy rate Avg_A_{cc} and the average number of selected features Avg_n_f is compared. In Table 2, the DE-AENN exhibits slightly higher classification accuracy and fewer selected features than GA based method. Therefore, the performance of our developed DE-AENN is relatively better than existing method and can enhance the classification accuracy for the SVM.

Conclusions

In this paper, the effectively hybrid DE-AENN method is proposed to optimize the reduction of noisy instances in a classification process for the SVM. The unified framework can characterize the essential instances, feature selection and SVM parameters via the DE optimization algorithm. Several UCI benchmark datasets are conducted to validate the effectiveness of the proposed method and the simulation results revealed that the proposed hybrid framework can achieve better performance than existing method in literature. Concentrate on the investigation of more large scale dataset as well as other heuristic search strategy may be interesting future work.

References

Zhen Guo, Zhongfei Mark Zhang, Shenghuo Zhu, Yun Chi and Yihong Gong, "A Two-Level Topic Model Towards Knowledge Discovery from Citation Networks," IEEE Transactions on Knowledge and Data Engineering, vol. 26, pp. 780-794, 2014.

Narges Armanfard, James P. Reilly, and Majid Komeili,"Logistic Localized Modeling of the Sample Space for Feature Selection and Classification,"IEEE Transactions on Neural Networks and Learning Systems, Vol. 29, pp. 1396-1413, 2018.

Mohammed A. Ambusaidi, Xiangjian He, Priyadarsi Nanda and Zhiyuan Tan, "Building an Intrusion Detection System Using a Filter-Based Feature Selection Algorithm," IEEE Transactions on Computers, vol. 65, pp. 2986-2998, 2016.

Swati Jadhav, Hongmei He and Karl Jenkins, "Information gain directed genetic algorithm wrapper feature selection for credit rating," Applied Soft Computing, vol. 69, pp. 541-553, 2018.

Jun Zhao, Long Chen, Wit old Pedrycz and Wei Wang, "Variational Inference-Based Automatic Relevance Determination Kernel for Embedded Feature Selection of Noisy Industrial Data," IEEE Transactions on Industrial Electronics, vol. 66, pp. 416-428, 2018.

H. Güney and H. Öztoprak, "Microarray-based cancer diagnosis: repeated cross-validation-based ensemble feature selection," International Journal of Engineering and Technology Innovation, vol. 54, pp. 272-274, 2018.

V.N. Vapnik, Statistical Learning Theory, 1st ed. New York: Wiley, 1998.

R. Storn and K. Price, "Differential Evolution–A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," Journal of Global Optimization, vol. 11, pp. 341-359, 1997.

C.L. Lu, I.F. Chung and T.C. Lin, "The Hybrid Dynamic Prototype Construction and Parameter Optimization with Genetic Algorithm for Support Vector Machine," International Journal of Engineering and Technology Innovation, vol. 5, pp. 220-232, 2015.

Chun-Liang Lu, Shih-Yuan Chiu, Chih-Hsu Hsu and Shi-Jim Yen, "Enhanced Differential Evolution Based on Adaptive Mutation and Wrapper Local Search Strategies for Global Optimization Problems", Journal of Applied Research and Technology, Vol. 12, No. 6, pp.1131-1143, 2014.