World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:1, No:1, 2007

A Fuzzy Classifier with Evolutionary Design of Ellipsoidal Decision Regions

Leehter Yao, Kuei-Song Weng, and Cherng-Dir Huang

***Abstract*—**A fuzzy classifier using multiple ellipsoids approximating decision regions for classification is to be designed in this paper. An algorithm called Gustafson-Kessel algorithm (GKA) with an adaptive distance norm based on covariance matrices of prototype data points is adopted to learn the ellipsoids. GKA is able to adapt the distance norm to the underlying distribution of the prototype data points except that the sizes of ellipsoids need to be determined a priori. To overcome GKA's inability to determine appropriate size of ellipsoid, the genetic algorithm (GA) is applied to learn the size of ellipsoid. With GA combined with GKA, it will be shown in this paper that the proposed method outperforms the benchmark algorithms as well as algorithms in the field.

***Keywords*—**ellipsoids, genetic algorithm, classification, fuzzy c-means (FCM).

I. Introduction

The primary task of classification is to determine the decision regions based on the prototype data points for training. As the class-labeled decision regions in the feature space are determined, the recognition of any unknown pattern depends on which decision region the feature point of this pattern falls into. Accurate and efficient learning of decision regions is thus essential to pattern recognition.

There have been numerous approaches proposed to learn the decision regions based on prototype data points. For instance, hyperboxes are employed in [1-2], polyhedrons are employed in [3-4] while ellipsoids are employed in [5-11]. In [12-16], genetic programming was employed to evolve and optimize the classifiers. Ellipsoids are found to be one of the commonly used approaches since it is easy to parameterize ellipsoids adapting with distribution and orientation of prototype data point.

The fuzzy c-means (FCM) algorithm proposed by Bezdek [17] is also a widely used and efficient clustering method for classification. Since FCM employs a Euclidean norm to measure dissimilarity, it inherently imposes a spheroid on the clusters, regardless of the actual data distribution. In [18] and [19], Gustafson and Kessel proposed a G-K algorithm (GKA) with an adaptive distance norm based on covariance matrices of cluster centers and data points. The advantage of GKA over FCM is that GKA is able to adapt the distance norm to the underlying distribution of the prototype data points. Since the distance norm employed in GKA is in the form of Mahalanobis norm, GKA essentially can be considered as utilizing ellipsoids to cluster prototype data points. The ellipsoids are able to adapt with the distribution and orientation of data point cloud in the feature space. However, GKA assumes fixed volumes of ellipsoids before iteratively calculating cluster centers. GKA is thus an effective method for the clustering of data points, but not suitable for estimating the underlying distribution of prototype data points belonging to the same cluster. In other words, the conventional GKA is effective to determine the center of each ellipsoid, yet ineffective to determine the size of ellipsoid.

In this paper, multiple ellipsoids are utilized to learn the decision region. To overcome GKA's inability to determine appropriate size of ellipsoid, the genetic algorithm (GA)[20] is applied to learn the size of ellipsoid. Since the size of ellipsoid is proportional to the determinant of norm inducing matrix of Mahalanobis norm, GA is applied to learn the determinant of each norm inducing matrix. As the determinant of norm inducing matrix is specified, GKA is able to recursively learn the ellipsoid with the specified size, or rather the specified determinant of norm inducing matrix. Multiple ellipsoids will be learned in parallel, and hence, the chromosome is encoded as the estimated determinants of all ellipsoids. Corresponding to every chromosome of GA, GKA is applied to learn the multiple ellipsoids in parallel. In order to evaluate the learning results of GKA corresponding to every estimated determinant of norm inducing matrix, the fitness function of GA is designed as the sum of misclassification errors, total volume of all ellipsoids and the overlapping volume of ellipsoids belonging to different classes. GA is applied to minimize the proposed fitness function. The misclassification error is defined as the total number of data points belonging to one class, which are included by the union of ellipsoids of the other class. Misclassification error is thus the primary index to minimize for the learning of decision regions. The reason of minimizing the total volume of ellipsoids along with misclassification errors is to prevent GKA from learning ellipsoids as large as possible to include all prototype data points in order to minimize misclassification errors. If appropriate sizes of ellipsoids are determined, the distribution of prototype data points can be approximated by ellipsoids with more accuracy. This leads to more accurate learning of decision regions. Since the volume of ellipsoid is proportional to the determinant of norm inducing matrix, the volume of ellipsoid can be directly measured by the estimated determinant of norm inducing matrix. Finally, since decision regions for different classes need to be disjoint, overlapping volume of ellipsoids belonging to different classes need to be minimized as well. An efficient method measuring overlapping volume between two ellipsoids will be proposed in this paper.

Manuscript received Nov. 5, 2004.

Leehter Yao is with Dept. of Electrical Engineering, National Taipei University of Technology, Taipei, Taiwan.(phone: +886-2-2751-0623; fax: +886-2-2751-3892; e-mail: ltyao@ntut.edu.tw).

K.-S. Weng and C.-D. Huang are with Dept. of Electrical Engineering, National Taipei University of Technology, Taipei, Taiwan.(e-mail: gsweng@mail.nihs.tp.edu.tw; s2408007@ntut.edu.tw).

II. FUZZY CLASSIFIER AND GUSTAFSON-KESSEL ALGORITHM

The problem of classifier design is to find an optimal mapping *f* from the feature space $E \subset R^n$ into the decision space $\Psi \equiv \{1,...,c\}$ for *c* classes, i.e., $f: E \rightarrow \Psi$. Given a set of *n*-dimensional prototypes $\boldsymbol{z_k} = [\boldsymbol{x_k}, y_k]^T \equiv [x_{k1}, x_{k2}, ..., x_{kn}, y_k]^T$, $\boldsymbol{x_k} \in E$, $y_k \in \Psi$, $k = 1...N$, the design of fuzzy classifier is to learn the decision regions based on these *N* prototypes. In this paper, the decision region for each class is composed of ellipsoids with different sizes and orientations. Let $\Omega_p \subset E$ be the decision region for the *p-th* class. Whether Ω_p is connected or disconnected, linearly separable or inseparable, α_p ellipsoids are to be utilized to approximate Ω_p, $p = 1...c$. Denote the *j-th* ellipsoid approximating Ω_p by ϕ_{pj}, then

$$\Omega_p = \bigcup_{j=1}^{\alpha_p} (\phi_{pj} \cap E). \tag{1}$$

In order to learn the ellipsoids, GKA is employed in this paper. In GKA, the learning of α_p ellipsoids corresponding to the decision region of the *p-th* class is conducted through clustering of the *p-th* class prototypes. The prototype data points belonging to different classes are clustered in parallel. As previously defined, if α_p ellipsoids are respectively assigned to cluster the prototype data points belonging to *p-th* class, totally η ellipsoids are required to learn, where

$$\eta = \sum_{p=1}^{c} \alpha_p . \tag{2}$$

In other words, the prototype data points are put into η clusters, out of which *m* classes are to be classified. For the convenience of notation, the index of ϕ_{pj} is redefined as:

$$i = (\sum_{l=1}^{p} \alpha_{l-1}) + j \tag{3}$$

where $\alpha_0 = 0$. Let $\boldsymbol{U} \in R^{\eta \times N}$ be the fuzzy partition matrix for the η clusters, the element μ_{ik} in the matrix $\boldsymbol{U}$ be the membership of the *i-th* clusters and the *k-th* prototype data points, $i = 1...\eta$, $k = 1...N$. Denote the weighting index by *m*, $m>1$, the error tolerance by ε, $\varepsilon>0$. The constraints for μ_{ik} are:

$$\mu_{ik} \in [0, 1], \quad i = 1...\eta, \; k = 1...N. \tag{4}$$

$$\sum_{i=1}^{\eta} \mu_{ik} = 1, \quad k = 1...N. \tag{5}$$

$$0 < \sum_{k=1}^{N} \mu_{ik} < N, \quad i = 1...\eta. \tag{6}$$

Suppose the coordinate of the *i-th* cluster center is defined as $\boldsymbol{v_i} \in R^{n \times 1}$, $i=1...\eta$; denote the matrix containing coordinates of all clusters by $\boldsymbol{V}$, i.e., $\boldsymbol{V} = [\boldsymbol{v_1}, \boldsymbol{v_2}, ..., \boldsymbol{v_\eta}]^T$. The distance between the *k-th* prototype data point and the *i-th* cluster center is given by the inner-product norm:

$$D_{ik}^2 = (\boldsymbol{x_k} - \boldsymbol{v_i})^T \boldsymbol{A_i} (\boldsymbol{x_k} - \boldsymbol{v_i}) \tag{7}$$

where $\boldsymbol{A_i} \in \mathrm{R}^{(n+1)\times(n+1)}$ is the norm inducing matrix defining weightings of distance with respect to different axes. From the ellipsoidal distance norm in (7), it can be considered that the each of the clusters is approximated by an ellipsoid. The decision regions are further approximated by the aggregations of suitable number of ellipsoids. The *i-th* ellipsoid $\phi_i(\cdot)$ is thus defined as

$$\phi_i(\boldsymbol{z}) = (\boldsymbol{x} - \boldsymbol{v_i})^T \boldsymbol{A_i} (\boldsymbol{x} - \boldsymbol{v_i}) = 1. \tag{8}$$

The volume of the ellipsoid in (8) is proportional to the determinant of $\boldsymbol{A_i}$. In GKA, the determinant of $\boldsymbol{A_i}$ is defined a priori. Prior to running GKA, if the determinant of $\boldsymbol{A_i}$ is given as ρ_i, $\boldsymbol{A_i}$ is constrained by

$$\det(\boldsymbol{A_i}) = \rho_i. \tag{9}$$

Since $\boldsymbol{A_i}$ is assumed to be a positive definite matrix, $\rho_i > 0$. Based on the prototype data points $\mathbf{x_k}$, $k=1...N$, GKA is to learn the fuzzy partition matrix $\boldsymbol{U}$, the coordinates of cluster centers $\boldsymbol{V}$ and the norm inducing matrix $\boldsymbol{A_i}$ by minimizing the distance norm in (7), $i = 1...\eta$ subject to the constraints in (5) and (9). For the constrained minimization, define the Lagrange multipliers ω_i, $i = 1...\eta$, and γ_k, $k = 1...N$, associated with the constraints in (5) and (9), respectively. The matrices $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{A} \equiv [\boldsymbol{A_1}, \boldsymbol{A_2}, ..., \boldsymbol{A_\eta}]$ are determined by the following equation:

$$\begin{aligned}(\boldsymbol{U},\boldsymbol{V},\boldsymbol{A}) = argmin\Big(&\sum_{i=1}^{c}\sum_{k=1}^{N} (\mu_{ik})^m \boldsymbol{D_{ik}^2} + \sum_{i=1}^{c} \eta_i (|\boldsymbol{A_i}| - \rho_i) \\ &+ \sum_{k=1}^{N} \gamma_k \Big(\sum_{i=1}^{c} \mu_{ik} - 1\Big)\Big).\end{aligned} \tag{10}$$

It is shown in [17-18] that the solutions of the optimization in (10) are as following:

$$\boldsymbol{A_i} = \left[\rho_i \, det(\boldsymbol{F_i})\right]^{1/n} \boldsymbol{F_i^{-1}}, \; i = 1...\eta, \tag{11}$$

where

$$\boldsymbol{F_i} = \frac{\sum_{k=1}^{N} (\mu_{ik})^m (\boldsymbol{x_k} - \boldsymbol{v_i})(\boldsymbol{x_k} - \boldsymbol{v_i})^T}{\sum_{k=1}^{N} (\mu_{ik})^m}.$$

The coordinate of each cluster center as well as the element in the partition matrix can be updated by the following equations:

$$\boldsymbol{v_i} = \frac{\sum_{k=1}^{N} (\mu_{ik})^m \boldsymbol{x_k}}{\sum_{k=1}^{N} (\mu_{ik})^m}, \tag{13}$$

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{\eta} (D_{ik}/D_{jk})^{2/(m-1)}}. \tag{14}$$

The matrix $\boldsymbol{F_i}$ in (12) is the fuzzy covariance matrix for the *i-th* cluster. Referring to (8), (11) and (12), the size of the ellipsoid for the *i-th* cluster is determined by the value of ρ_i while the geometric shape of this ellipsoid is determined by the norm inducing matrix $\boldsymbol{A_i}$.

The GKA can be summarized as following [17-18]:

Step 1: Determine the total number of clusters η ($2 \le \eta \le$ N), the exponential weight *m*, and the termination tolerance ε. Select the volume of each ellipsoid to be learned, ρ_i, $i = 1...c$. Choose an initial partition matrix $\boldsymbol{U^{(0)}}$ satisfying (1)-(3). Set the iteration index $q = 0$.

Step 2: Calculate fuzzy cluster center centers $\boldsymbol{v_i^{(q)}}$, $i = 1...\eta$, by (13) based on $\boldsymbol{U^{(q)}}$.

Step 3: Calculate the fuzzy covariance matrices $\boldsymbol{F_i^{(q)}}$, $i = 1...\eta$, by (12) based on $\boldsymbol{v_i^{(q)}}$ and $\boldsymbol{U^{(q)}}$.

Step 4: Determine the ellipsoids $\phi_i(\cdot)$ $i = 1...\eta$, by calculating the norm inducing matrices $A_i^{(q)}$ in (11) based on ρ_i and $F_i^{(q)}$.

Step 5: Calculate the distance $D_{ik}^{2(q)}$ between every prototype data and every cluster center, $i = 1...\eta$, $k = 1...N$, by (7).

Step 6: Update the partition matrix $U^{(q+1)}$ by calculating $\mu_{ik}^{(q+1)}$ in (14) based on $D_{ik}^{2(q)}$. Note that (14) can only be applied provided that $D_{ik}^{2(q)}>0$. If it happens that $D_{ik}^{2(q)} = 0$ as i = s, then $\mu_{sk}^{(q+1)} = 1$ and $\mu_{ik}^{(q+1)} = 0$, $\forall i \in [1, \eta]$ but $i \neq s$.

Step 7: Calculate $\Delta = \left\| U^{(q+1)} - U^{(q)} \right\| \equiv \max_{i,k} \left| \mu_{ik}^{(q+1)} - \mu_{ik}^{(q)} \right|$. If $\Delta > \varepsilon$, set $q = q+1$ and go to step 2; otherwise stop the algorithm.

With predetermined volume of each ellipsoid, GKA is able to learn η ellipsoids in parallel so that each ellipsoid tends to locate every cluster center and include prototype data points as many as possible. However, if the volume is not determined correctly, the ellipsoid will not include every prototype data point belonging to the same cluster no matter how well the cluster center has learned. To overcome this difficulty, a GA based learning algorithm combined with GKA is proposed to automatically learn the sizes of ellipsoids so that the decision regions are effectively approximated by suitable number of aggregated ellipsoids.

III. Evolutionary Learning of Decision Regions

As stated in the previous section, GA is applied to learn the volume of every ellipsoid ϕ_i, $i = 1...\eta$. Since the volume of ϕ_i is proportional to $\det(A_i)=\rho_i$, GA is applied to search for an optimal value of ρ_i. The searching for optimal volume is conducted in parallel for η ellipsoids. The chromosome of GA is thus implemented as a cascaded binary representations of ρ_i, $i = 1...\eta$. Within every generation, as $\rho_1, \rho_2,...,\rho_\eta$ are decoded from every chromosome, GKA is then respectively applied to learn the ellipsoid corresponding to every ρ_i, $i = 1...\eta$. The learning by the combination of GA and GKA aims to recursively search for η ellipsoids so that the classification is optimized.

Denote $\Lambda(\phi, \boldsymbol{x})$ as the function calculating the "membership value" of the prototype data $\boldsymbol{x}$ points to the geometric region ϕ. If $\lambda_p(\cdot)$ is the "class membership value" for the *p-th* class, referring to (1), it can be defined as

$$\lambda_p(\boldsymbol{x_k}) = \Lambda(\Omega_p, \boldsymbol{x_k}) = \bigvee_{j=1}^{\alpha_p} \Lambda(\phi_{pj}, \boldsymbol{x_k}) \quad (15)$$

where $\vee$ denotes the operation max(·). The membership value determined by function $\Lambda(\phi_{pj}, \boldsymbol{x_k})$ is given by

$$\Lambda(\phi_{pj}, \boldsymbol{x_k}) = \mu_{ik} \quad (16)$$

where μ_{ik} is defined as in (14) and the index i is defined as (3) which renumbers the index i based on the given class index p. Let $\xi(\boldsymbol{x_k}) \in \Psi$ be the class that $\boldsymbol{x_k}$ is classified into, then $\xi(\boldsymbol{x_k})$ is given by the class corresponding to largest class membership value, i.e.,

$$\xi(\boldsymbol{x_k}) = \underset{p=1...c}{Arg\,max}(\lambda_p(\boldsymbol{x_k})). \quad (17)$$

Define a logistic threshold function $\Gamma(\cdot)$ as following:

$$\Gamma(r) = \begin{cases} 1, \text{ if the statement } r \text{ is true;} \\ 0, \text{ if the statement } r \text{ is false.} \end{cases} \quad (18)$$

The total misclassification errors e_c is defined as

$$e_c = \sum_{k=1}^{N} \Gamma(\xi(\boldsymbol{x_k}) = p \text{ but } y_k \neq p). \quad (19)$$

To optimize the classification results, GA combined with GKA are thus applied to search for suitable volumes of ellipsoids so that the misclassification errors e_c in (19) is minimized. However, if e_c is directly used as the fitness function for GA, GA tends to find the ellipsoids as large as possible so that all the prototype data points are included by the arbitrarily large ellipsoids. To overcome this difficulty, the fitness function of GA is designed to simultaneously evaluate the sizes of generated ellipsoids as well as the total misclassification errors. Let τ be the fitness function for GA, it can be defined as following:

$$\tau = e_c + \omega \sum_{i=1}^{\eta} \rho_i, \quad (20)$$

where ω is a weighting factor and e_c is defined in (19). Within each generation of GA, as $\rho_1, \rho_2,...,\rho_\eta$ are decoded from every chromosome of GA, GKA will be applied based on the decoded $\rho_1, \rho_2,...,\rho_\eta$ to learn $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{A}$ by (10)-(14). The matrices $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{A}$ associated with every chromosome of GA are recursively calculated until GKA converges in every GA's generation.

The proposed classifier is to utilize η ellipsoids approximating decision regions for p ($<\eta$) classes. The number of ellipsoids needs to be determined prior to running GA and GKA to search for optimal parameterizations of these η ellipsoids. Gath and Geva suggested an efficient way of determining the reasonable number of ellipsoids to use in order to approximate the decision region of the *p-th* class, $p = 1...c$. With the volume of each ellipsoid being fixed, i.e. assuming that ρ_i equal to a constant, define the fuzzy hypervolume as following:

$$V_p = \sum_{i=1}^{\upsilon_p} (det(F_i))^{1/2}, \quad (21)$$

where υ_p is the number of ellipsoids for evaluation varying from 1 to the largest possible number v_p^{max}. The reasonable number of ellipsoids, α_p, is the one corresponding to relatively small V_p. In other words, if V_p varies with υ_p, α_p is where the knee point of V_p locates. Referring to (2), after $\alpha_1, \alpha_2, ..., \alpha_c$ are determined, η ellipsoids are to be learned by GA and GKA.

The proposed fuzzy classifier with evolutionary design of ellipsoidal decision regions can be summarized as following steps.

Step 0: Set the following GKA parameters: the exponential weight m, and the termination tolerance ε. Set the

following GA parameters: the termination fitness value tolerance ε_g and generation tolerance G_t.

Step 1: With the volume of each ellipsoid being fixed to be 1, iteratively applies GKA to classify the prototype data points for different number ellipsoids, υ_1, υ_2,..., υ_p. Based on the variations of fuzzy hypervolume in (21), determine α_1, α_2,..., α_c and consequently η.

Step 3: Initialize the gene pool encoding the estimated volumes of η ellipsoids into the chromosome of GA.

Step 4: Decode each chromosome in order to find the estimated volumes of ellipsoids, ρ_1, ρ_2,...,ρ_η. Apply GKA in the previous section to find ***(U,V,A)*** in (10) associated with each chromosome.

Step 5: Based on the matrix ***U*** associated with each chromosome, calculate fitness value of each chromosome by (15)-(20).

Step 6: Determine if the best fitness value less than or equal to ε_g or the best fitness values haven't changed for more than G_t generations. If yes, go to step 8, otherwise go to step 7.

Step 7: Pass the best chromosome into next generation and perform reproduction, crossover and mutation to generate all chromosomes in next generation. Go to step 4.

Step 8: Stop the algorithm.

IV. Computer Simulations

In this section, two computer simulations will be made to verify the performance and effectiveness of the proposed method. The gene pool of GA consists of 100 chromosomes. Each of the estimated ellipsoid volumes is represented by 16 bits. Elitist model is used for GA's reproduction operation. The mutation rate for the following two examples is set to be 0.01.

Example A: Given that the prototype data points are shown in Fig. 1 where 460 data points are classified as class 1 (represented by ○) while 97 data points are classified as class 2 (represented by •)[21], GA was employed in [21] to search for piecewise linear segments approximating the decision regions. In this example, 10% of the prototype data points (represented by △) are used to train the decision regions while the rest of 90% data points are for testing. The variations of fuzzy hypervolumes with different number of ellipsoids are shown in Fig. 2. It is obvious in Fig. 2 that the knee point of the variations is where $\upsilon_p = 6$. Therefore, 6 ellipsoids are utilized to approximate the decision regions for both class 1 and 2. The performance of the proposed method is compared with the performance by Pal's method in [21], multilayer perceptron (MLP), Bayes method and k-NN method as shown in Table I. It is shown in Table I that the proposed method outperforms the other methods list in the Table since it gives the highest recognition rate.

Table I. Comparison of recognition rate (%) by different methods

method	MLP	k-NN	Bayes	Pal's	Our method	method
Class 1	94.06	93.57	100.0	96.44	99.28	
Class 2	76.01	62.09	22.73	82.10	94.31	
overall	92.40	89.28	86.45	93.93	98.40	

Example B: To further test the proposed approach for classification, it will be tested in 3 benchmark data sets form the UCI repository of machine learning databases [22]. The first data set is the Fisher iris data set, which consists of 150 four dimensional prototype data points classified into three classes. The second data set is Pima Indian diabetes data set, which consists of 768 eight dimensional prototype data points classified into two classes. The third data set is Wisconsin breast cancer data set, which consists of 683 nine dimensional prototype data points classified into two classes. To evaluate the recognition rate, fivefold cross-validation scheme is adopted. In other words, the data set is randomly and evenly divided into five subgroups. Each subgroup's data are utilized to test the recognition rate based on the decision regions determined by the data set in the remaining four subgroups. To avoid experiment bias, the fivefold cross-validation is independently run 5 times before the results are recorded in Table II. The recognition rates obtained by the proposed method are compared with the results from the benchmark machine learning algorithms C4.5 and C4.5Rules [23] and from the method similar to genetic programming called gene expression programming (GEP) [16]. In Table II, the average recognition rate as well as its 95% confidence interval, which is proportional to the standard deviations of the recorded data, are both compared. It is shown in Table II that the proposed method outperforms the benchmark algorithms and GEP.

Table II. Comparison of recognition rate (%)

data set	C4.5	C4.5Rules	GEP	Our method
Iris	93.9±8.1	94.6±8.2	95.3±4.6	98.7±3.5
Pima Indian	74.8±4.7	75.4±4.3	69.7±3.8	76.6±1.9
Breast Cancer	94.7±1.5	95.6±1.6	96.2±1.8	96.76±0.74

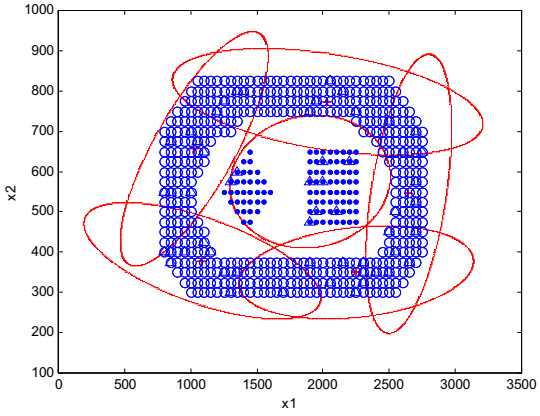

Fig. 1 Prototype data points for training and testing

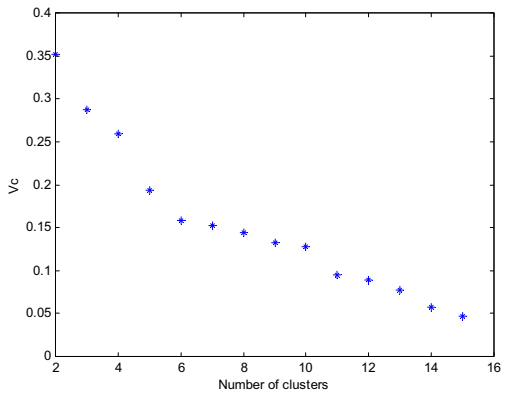

Fig. 2. Variations of fuzzy hypervolumes with different number of ellipsoids.

V. Conclusions

This paper presents an efficient method of learning decision regions for classification via evolutionary optimization. The decision regions are approximated by suitable number of ellipsoids. Although GA is a notorious time consuming optimization approach, this paper combines GA with GKA so that the ellipsoid learning efforts are reduced to simply the sizes of ellipsoids. The learning efficiency can thus be greatly improved. The reasonable number of ellipsoids to be used for classification can also be determined by calculating fuzzy hypervolumes.

Although not be further used in this paper, the norm inducing matrix of every estimated ellipsoid also contain important information about the data set clustered by the ellipsoid. The ellipsoid's orientation in *n*-dimensional space is actually determined by the eigenvectors of norm inducting matrix. This can be explored for further investigations.

References

[1] P. K. Simpson, "Fuzzy min-max neural networks-Part 1: Classification," *IEEE trans. Neural Networks,* vol. 3, pp. 776-786, Sept. 1992.

[2] S. Abe and M.-S. Lan, "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification," *IEEE Trans. Fuzzy Syst.,* vol. 3, pp. 18-28, Jan. 1995.

[3] F. Uebele, S. Abe and M.-S. Lan, "A neural network-based fuzzy classifier," *IEEE Trans. Syst., Man, Cybern.,* vol. 25, pp. 353-361, Mar. 1995.

[4] J. A. Dickerson and B. Kosdo, "Fuzzy function approximation with ellipsoidal rules," *IEEE Trans. Syst., Man, Cybern.,* pt. B, vol. 26, pp. 542-560, Aug. 1996.

[5] R. H. Dav and R. Krishnapuram, "Robust Clustering Methods: A Unified View," *IEEE Trans. Fuzzy Systems,* vol. 5, no. 2, pp. 270-293, May 1997.

[6] S. Abe and R. Thawonmas, "A fuzzy classifier with ellipsoidal regions," *IEEE Trans. Fuzzy Systems,* vol.5, no. 3, pp. 358-368, Aug. 1997.

[7] S. Abe, "Dynamic cluster generation for a fuzzy classifier with ellipsoidal regions," *IEEE Trans. Syst., Man, Cybern.,* pt. B, vol. 28, no 6, pp. 869-876, Dec. 1998.

[8] S. Abe, R. Thawonmas and M. Kayama, "A Fuzzy Classifier with Ellipsoidal Regions for Diagnosis Problems," *IEEE Trans. Syst., Man, Cybern.,* pt. C, vol. 29, no. 1, pp. 140-149, Feb. 1999.

[9] L. Yao, "Nonparametric learning of decision regions via the genetic algorithm," *IEEE Trans. System, Man, and Cybernetics,* Vol. 26, No. 2, pp. 313-321, April 1996.

[10] S. Wu, M. J. Er and Y. Gao, "A fast approach for automatic generation of fuzzy rules by generalized dynamic fuzzy neural networks," *IEEE Trans. Fuzzy Systems,* vol. 9, no. 4, pp. 578-594, Aug. 2001.

[11] G. D. Tattersall and K. Yi, "Packed hyper-ellipsoid classifiers," *Electronics Letters,* vol. 30, no. 5, pp. 427-428, March 1994.

[12] J. K. Kishore, L. M. Patnaik, V. Mani, and V. K. Agrawal, "Application of genetic programming for multicategory pattern classification," *IEEE Trans. Evolutionary Computation,* vol. 4, no. 3, pp. 242-258, Sep. 2000.

[13] Kushchu, "Genetic programming and evolutionary generalization," *IEEE Trans. Evolutionary Computation,* vol. 6, no. 5, pp. 431-442, Oct. 2002.

[14] M. M. Rizki, M. a. Zmuda, and L. a. Tamburino, "Evolving pattern recognition systems," *IEEE Trans. Evolutionary Computation,* vol. 6, no. 6, pp. 594-609, Dec. 2002.

[15] D. P. Muni, N. R. Pal, and J. Das, "A novel approach to design classifiers using genetic programming," *IEEE Trans. Evolutionary Computation,* vol. 8, no. 2, pp. 183-196, April 2004.

[16] C. Zhou, W. Xiao, T. M. Tirpak, and P. C. Nelson, "Evolving accurate and compact classification rules with gene expression programming," *IEEE Trans. Evolutionary Computation,* vol. 7, no. 6, pp. 519-531, Dec. 2003.

[17] J. Bezdek, *Pattern Recognition with Fuzzy Objective Function,* Plenum Press, New York, 1981.

[18] D. E. Gustafson and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," *in Proc. IEEE Conf. Decision Contr.,* San Diego, CA, 1979, pp. 761-766.

[19] R. Babuška, *Fuzzy modeling for control*, Kluwer Academic Publishers: Massachusetts, 1998.

[20] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning,* Reading, MA: Addison-Wesley, 1989.

[21] S. K. Pal, S. Bandyopadhyay, and C. A. Murthy, "Genetic algorithms for generation of class boundaries," *IEEE Trans. Syst., Man, Cybern.,* vol. 28, pp. 816-828, Dec. 1998.

[22] C. Blake, E. Keogh, and C. J. Merz, *UCI Repository of Machine Learning Databases,* University of California at Irvine, Dept. Inform. Comput. Sci., CA., available at http://www.ics.uci.edu/~mlearn/MLRepository.html