APPLICATION OF CONIC OPTIMIZATION AND SEMIDEFINITE PROGRAMMING IN CLASSIFICATION

Abdel-Karim S.O. Hassan, Mohamed A. El-Gamal & Ahmad A.I. Ibrahim

Department of Engineering Mathematics and Physics, Cairo University, Giza, Egypt

ABSTRACT

In this paper, Conic optimization and semidefinite programming (SDP) are utilized and applied in classification problem. Two new classification algorithms are proposed and completely described. The new algorithms are; the Voting Classifier (VC) and the N-ellipsoidal Classifier (NEC). Both are built on solving a Semidefinite Quadratic Linear (SQL) optimization problem of dimension n where n is the number of features describing each pattern in the classification problem. The voting classifier updates usage of ellipsoids in separating N different classes instead of only binary classification by using a voting unit. The N-ellipsoidal classifier makes the separation by means of N separating ellipsoids each contains one of the N learning sets of the classes intended to be separated. Experiments are performed on some data sets from UCI machine learning repository. Results are compared with several well-known classification algorithms, and the proposed approaches are shown to provide more accurate and less complex classification systems with competitive error rates.

Keywords: *Conic optimization; Semidefinite programming; Pattern recognition; Classifiers; Separating ellipsoids*

1. INTRODUCTION

Conic optimization and SDP are used in many applications. They are used because of their ability to model different practical problems efficiently with a competitive complexity because of the extension of interior point methods to solve conic optimization programs. Some of these applications are; graph theory [1]; wireless communication systems [2];control theory [3];beamforming [4]; VLSI [5], cellular networks [6]; and classification and pattern recognition [7].

Classification is one of the most important tasks in machine learning and data mining. In the classification problem we have a set of objects; each is completely described by n-dimensional vector. Each entry of the vector is a numerical value representing a certain characteristic of the object and is called a feature of the object. If these objects are divided into C classes, the target will be to separate each class and to be able to decide each given vector belongs to which of these C classes. This target is accomplished through two phases:

1- The learning phase which separates patterns of well-known objects.
2- The testing phase which classifies unknown objects using the partitions obtained in the learning phase.

Many classification algorithms were introduced and have been widely studied such as;machine learning and data mining [8], neural networks [9], evolutionary algorithms [10-12], and decision trees [13].

Ellipsoids have beenused as classifiersin 2-class problems. Ellipsoid separation is suggested due to ellipsoids nature being the simplest convex hull that is able to enclose patterns of the same class which are expected to be close to each other [14].The two main challenges in choosing the separating ellipsoid are:

1- A separating ellipsoid is not unique most of the time. How will we choose one?
2- There may be no separating ellipsoid.

Many methods tried to overcomethese challenges. The separating ellipsoid is chosen according to a certain criterion, e.g.the separating ellipsoid may be the minimum volume ellipsoid. Any chosen criterionleads directly to an optimization problem.

Generally, an ellipsoid is a set of points in$R_n$described by a center $c \in R_n$ and a symmetric positive semidefinite nxn orientation matrix$E \in SR_{n\times n}$, where $SR_{n\times n}$represents the nxn symmetric matrix space. The positive semidefinite condition directs the optimization problem concerning the selection of the separating ellipsoid to semidefinite and conic optimization.

The previous work using ellipsoidal separationtargeted only 2-class problems and showed competitive results [14]. In this paper,usage of ellipsoids in relatively large classification problems is utilized by upgrading existing algorithms and creating new algorithms to be able to classify multi-class problems.

The paper is organized as follows;convex, conic, and semidefinite quadratic linear (SQL) programming problems arebriefly presented. The proposed classifiers are introduced and justified including SQL formulation of the classification problem. Finally, experimental results show the potential and effectiveness of the proposed classifiers.

2. CONVEX, CONIC, AND SEMIDEFINITE OPTIMIZATION

The advantage of convex programs which made engineers interested in it is that a local minimum is also a global minimum. If f_0 and f are two convex functions, then the formulation of a convex optimization problem is as follows [15]

$$\min_{x\in R_n} f_0(x) \quad \text{subject to } f(x) \leq 0 \qquad (1)$$

It is important to stress that checking that a problem is convex maybe more difficult than solving the problem itself. That's why information aboutf_0 and f being both convex is needed before solving the problem. This requires some knowledge about the structure of the problem.

This leads to the usage of convex cones (conic optimization problems) which is a special type of convex sets which is closed under positive scaling.

Some of the most useful and common cones are

1- the positive semidefinite cone (S_n^+) which is a subset of the set of n × nsymmetric matrices which consists of all positive semi-definite matrices i.e.

$$M \in S_n^+ \Leftrightarrow M \succcurlyeq 0 \Leftrightarrow \lambda(M) \geq 0 \qquad (2)$$

Where $\lambda(M)$ is the set of Eigen values of M

2- The Second-order cone (SOC) also called ice-cream cone

$$SOC(n) = \{\, (t,x) \;\; such\ that \;\; t \geq \|x\| \,\} \qquad (3)$$

Conic optimization enjoys the same kind of rich duality as linear programing. In Linear optimization we have the strong duality theorem which ensures a zero duality gap in case of optimal solution and vice versa, while in conic optimization we have the week duality theorem which only ensures that a solution with a zero duality gap must be optimal[15].

The conic problem has been deeply studied in the work of Nestrov and Nemirovsky [15]. They showed that any convex program can be transformed into an equivalent conic one.Alsointerior point methods (IPMs) can solve conic programs efficiently with accuracy ϵand polynomial complexity ($\sqrt{n}\log\frac{1}{\epsilon}$) where n is the dimension of the problem.

Although solvable in polynomial time, the practical number of iteration is very high in practice which is a major drawback in the complexity theory. This can be explained easily because the bound tells about the worst case scenario which always occurs in practical problems in conic optimization. This made researchers to become interested in a special category of conic programs that can take large steps towards the optimal solution. These conic programs in interest are the ones using self-scaled cones [15]. Self-scaled cones are the homogeneous self-dual cones. A cone $C$is homogeneous if for any pair of points $x, y \in int\ C$there exists an invertible linear mapping $A : R_n \to R_n$leaving C invariant such that $A(x) = y$ and it is self-dual if $(C^*)^* = C$.

It can be shown that the second order cone is a self-scaled cone [16]. This reduces the convex program to the following Semidefinite Quadratic Linear (SQL) formulation

$$\begin{aligned} &\min_{x\in R_n} c^T x \\ &\text{subject to: } T(x) = b, \\ &\quad x \in C, \end{aligned} \qquad (4)$$

where the objective function is linear as the vectors $c, x\in R_n$, the transformation T which is T: $R_n \to R_m$ is linear, the vector $b\in R_m$, and Cis a second order self-scaled cone.

3. ELLIPSOIDS AND THE PROPOSED CLASSIFIERS

An ellipsoidε in R_n is defined by

$$\varepsilon = \{x \in R_n : (x-c)^T E (x-c) \leq 1\}, \qquad (5)$$

Where $c \in R_n$ is the center and $E \in SR_{n\times n}$ is a symmetric positive semidefinite matrix representing the orientation.

Ellipsoids have been used to separate the two classes due to the following advantages:

1- Patterns of the same class are expected to be close to each other and this characteristic leads to the ability of enclosing them in some kind of hull, possibly a ball. Ellipsoids are the generalization of balls and this makes the procedure scaling invariant.

2- Ellipsoids are considered as a generalization of using hyperplanes. This is because the setofpointslying between two parallel hyperplanes is a degenerate ellipsoid.

3- Geometrical problems involving ellipsoids can be modeled using SQL conic programs.

4- Ellipsoids are the simplest convex hull.

Ellipsoidal classifiers were implemented to classify 2-class problems using many techniques [14]. The proposed classifiers – in this paper - utilize the usage of ellipsoids in multi-class problems and this was done using two approaches;

First, an upgradeto some existing classifiers to classify multi-class problemsby designing N-stages of 2-class classifiers and then take their outputs to a voting unit and this will be called the Voting Classifier.

Second, a new proposedclassifier called the N-Ellipsoidal Classifier (NEC) which is able to solve multi-class separation problems directly without staging.

3.1 The Voting Classifier (VC)

To be able to use the 2-class pattern separation techniques in a multi-class class problem, N classifiers were designed; each separates one of the classes from all other classes by constructing two ellipsoids; one ellipsoid contains the learning vectors of the class we need to separate and the other contains all other learning vectors of all other classes. In the testing phase, the classifier that detects that the test pattern vector belongs to its ellipsoid - the classifier is nothing more than an n-dimensional ellipsoid - raises a detection flag. Of course this occurs if separation is done perfectly and the test vector is not detected to belong to more than one classifier or not detected at all by any of the N classifiers. If more than one detection flag are (no detection flag is) raised then the weighted distances of the test pattern vector from the centers of the ellipsoids that raised the detection flag (the centers of all ellipsoids) are calculated. The fault is classified to the ellipsoid whose center has the least weighted distance to the test pattern vector. Calculating the weighted distance is very simple just by substituting with the test pattern vector in the equation of the ellipsoid. The following diagram in "Fig. 1" illustrates the idea.

There are many criteria to choose the separating ellipsoid such as minimum volume, maximal separation ratio, maximum sum separation, and minimum squared sum [14].In the proposed classifiers the minimum volume and the minimum squared sum criteria are applied.

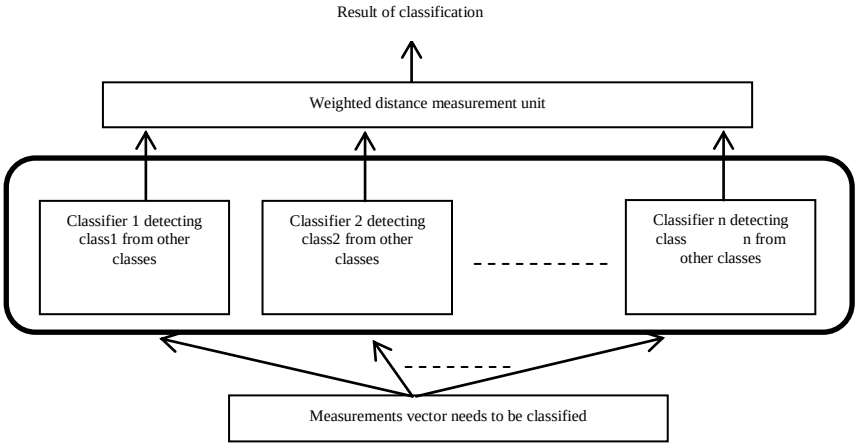


Figure 1. The voting classifier separateingN-classes

3.1.1 The Minimum Volume Ellipsoid (MINVOL)

Thiscriterion aims to get a separating ellipsoid by finding the minimum volume ellipsoid containing one of the two patterns. It's expected to make the separation with poor accuracy because this method concentrates on one of the two patterns and ignores the other class of patterns completely which means that we have no information from these ignored patterns at all.

The function boxsize(E) is introduced to measure the size of the ellipsoid as shown below and as shown in "Fig. 2".It's well known that the semi-axis of the ellipsoid described by orientation matrix E is equal to the square root of the inverse of the eigenvalues of E.

$$\text{boxsize(E)} = \sqrt{\sum_{i=1}^{n} \lambda_i(\text{E})^{-1}} = \sqrt{\sum_{i=1}^{n} \text{S}_i{}^2} \qquad (6)$$

Introducing the matrix $\begin{pmatrix} \text{E} & \text{I} \\ \text{I} & \text{T} \end{pmatrix}$ that if we guaranteed that it is positive semidefinite implies directly that[$\text{trace T} \geq \text{trace E}^{-1}$]from Schur's Compliment theorem, then we can define the following SQL-program

$$\min \text{trace } T \quad \text{s.t.} \begin{cases} a_i{}^T E\, a_i \le 1 \quad \forall i \\ \begin{pmatrix} E & I \\ I & T \end{pmatrix} \succcurlyeq 0 \\ E \succcurlyeq 0 \end{cases} \qquad (7)$$

The problem with the above program is that the first set of constraints is non-convex. The solution to this problem is done by using the homogeneous description of the orientation ellipsoid E. Homogeneous description means representing the n-dimensional ellipsoid as the projection on R_n of the intersection of the (n+1)-dimensional ellipsoid E' centered at the origin with a correctly chosen hyperplane. This formulation overcomes the difficulty caused by the extension to (n+1) dimension which is multiple possible (n+1)-dimensional ellipsoids achieving our target.

Let $E' = \begin{pmatrix} s & v^T \\ v & F \end{pmatrix}$ where s is scalar, $v \in R_n$ and $F \in SR_{n\times n}$. Also, we define (d) such that $v = -Fd$

$$(1,x)^T E' (1,x) \le 1 \leftrightarrow (x-d)^T \frac{F}{(1-\delta)} (x-d) \le 1 \qquad (8)$$

where $\delta = s - d^T F d$

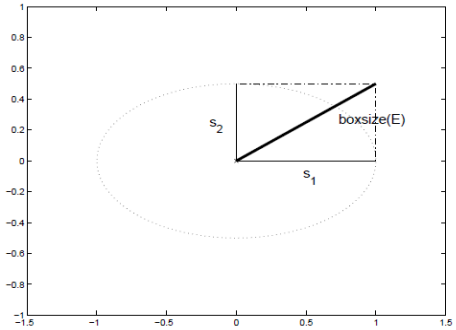


Figure 2. The boxsize function representing the volume of the ellipsoid

The above formulation means that being able to evaluate the optimized (n+1)-dimensional ellipsoid E' will enable us to get the n-dimensional ellipsoid center (d) and orientation matrix($\frac{F}{1-\delta}$)where δ is proved to be equal to zero in the optimal situation which leadsthat the orientation matrix will be (F) directly.This leads to the following conic program

$$\min \text{trace } T \quad \text{s.t.} \begin{cases} (1,a_i)^T E' (1,a_i) \le 1 \quad \forall i \\ \begin{pmatrix} E & I \\ I & T \end{pmatrix} \succcurlyeq 0 \\ E' \succcurlyeq 0 \end{cases} \qquad (9)$$

The first inequality ensures that all patterns of $a_i$are inside the homogeneous ellipse $E'$and minimizing the objective function trace (T) leads directly to minimizing the volume of the containing ellipsoid. This is because by minimizing trace (T), we minimize the upper bound of trace(E^{-1}) which leads to minimizing the boxsize(E) which models the volume of the containing ellipsoid.

The advantage of this method is that it can always find a separating ellipsoid even if patterns of the two classes are overlapping.

The disadvantages are; one of the classes is considered and the other is not which may sound rather weird for a separation problem. Besides, this method is dependent on the
coordinate system simply because the boxsize function of an ellipsoid depends on the scaling.

3.1.2 Minimum Squared Sum (MINSQUARED)

To be able to do the separation here, the separation ratio (ρ) is defined as the ratio between the sizes of two ellipsoids; one includes the patterns of class 1 and the other excludes the patterns of class 2. This method targets making most of the bj's lie outside the ellipsoid that contains the ai's. This is done by computing the separation ratio between all the ai's and each bj separately (call itρ_j) and making most of them to be as large as possible. Maximizing most of the $\rho_j's$ is suggested in this method to be done by minimizing the arithmetic mean of the inverse of the square of the separation ratios. This targets to minimize the large inverse values which correspond to the small separation ratios and then by minimizing the inverse, the small separation ratios will be maximized.

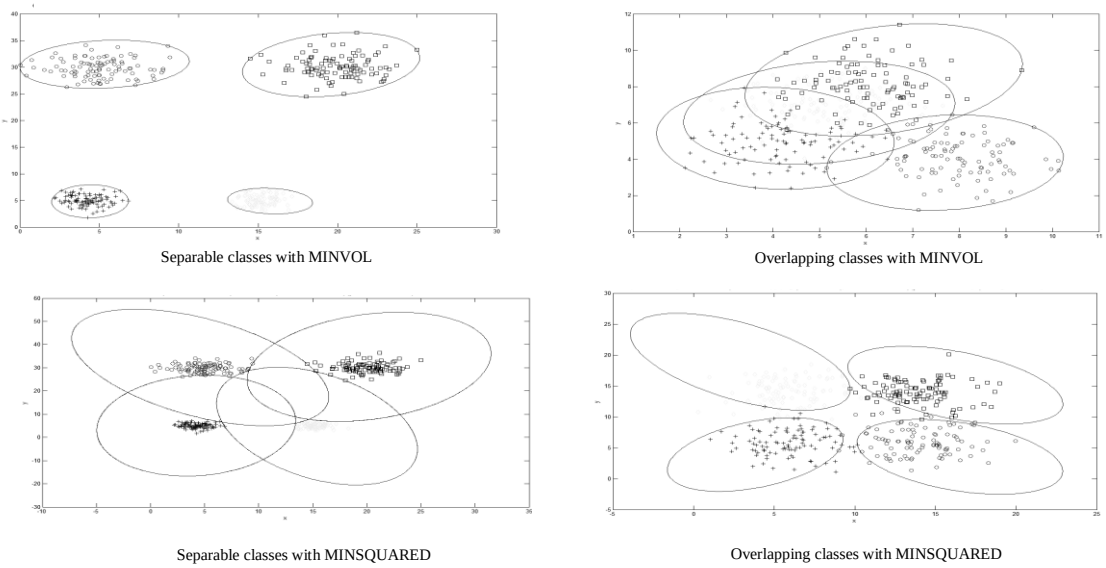
Now, the SQL formulation of this method using homogeneous description of the orientation ellipsoid E (where $k_j = \rho_j{}^{-2}$) is

$$\min \sum_j {k_i}^2 \quad \text{s.t.} \quad \begin{cases} (1,a_i)^T E' (1,a_i) = k_j & \forall i \\ (1,b_j)^T E' (1,b_j) \geq 1 & \forall j \\ E' \succcurlyeq 0 \end{cases} \qquad (10)$$

3.2 The N-ellipsoidal Classifier (NEC)

In this approach, instead of getting one separating ellipsoid to separate between two classes only, N ellipsoids are constructed; each contains the vectors of one of the classes only in the learning phase. In the testing phase, any test pattern (test vector) will be classified by evaluating the weighted distances for this test vector from the center of each of the N constructed ellipsoids. The test pattern is then decided to belong to the ellipsoid whose center is nearest.

In "Fig. 3", an illustrative synthetic example shows the NEC working over a classification problem that contains four classes. The classes of the problem are constructed in two different ways; separable and overlapping. Minimum volume and minimum squared sum criteria are chosenin constructing the N separating ellipsoids. The minimum volume criterion is chosen because it is always able to give information about the separating ellipsoids even if the patterns were not separable. And the minimum squared sum was chosen because it is expected to provide the best results as it targets maximum possible separation between the two classes.



Separable classes with MINVOL

Overlapping classes with MINVOL

Separable classes with MINSQUARED

Overlapping classes with MINSQUARED

Figure 3. The synthetic example illustrating the NEC

4. EXPERIMENTAL RESULTS

The proposed classifiers are applied to three datasets from the UCI machine repository [17]. Table I gives a brief description of these data sets.

The performance measure is the error rate. The error rate is calculated by repeating the classification process 10 times. Each time different patterns are randomly chosen from data to construct the learning and the testing set. The average error of these trials is computed.

The new proposed classifiers - Voting Classifier and NEC - are simulated using Matlab R2007b software and the SQL problem is solved using Sedumi 1.1 solver under Yalmip [18].

All the experiments are performed on a 2.67GHz Intel Core 2 Duo CPU with 4.00 GB main memory, running Microsoft Windows 7.

Table II-VI summarizes the classification results.Moreover, the classification results are compared with a number of prominent classification techniques, namely; Classification Based on Multiple Class-Association Rules (CMAR) [19], Classification Based on Associations (CBA) [20], C4.5 [21], K- Nearest Neighbor (KNN), Support Vector Machine (SVM), classificationby Linearity Assumption (LA) [22], Interpretable Simulated Annealing based Fuzzy classification (ISAF), and k-nearest neighborhood [23] which were implemented by their authors.

TABLE I. UCI BENCH MARK EXAMPLES

Data Set	**Data Structure**		
	No. of Classes	***No. of attributes***	***No. of patterns***
Fisher Iris	3	4	150
Wine Types	3	13	178
Glass Ident.	7	10	214

TABLE II. RESULTS OF THE FISHER IRIS CLASSIFICATION PROBLEM

CLASSIFIER TYPE	**THE VOTING CLASSIFIER**				**THE VOTING CLASSIFIER**			
Separating Ellipsoid	***MINVOL Ellipsoid***		***MINSQUARED Ellipsoid***		***MINVOL Ellipsoid***		***MINSQUARED Ellipsoid***	
Learning Set	20% learning	50% learning	20% learning	50% learning	20% learning	50% learning	20% learning	50% learning
Learn Error	0%	0%	0%	0%	0%	0%	0%	0%
Test Error	2.5%	6.7%	2.4%	5.3%	2.8%	7.8%	2.5%	6.2%

TABLE III. RESULTS OF THE WINE TYPESCLASSIFICATION PROBLEM

CLASSIFIER TYPE	**THE VOTING CLASSIFIER**		**THE N-ELLIPSOIDAL CLASSIFIER**	
Separating Ellipsoid Criteria	***MINVOL Ellipsoid***	***MINSQUARED Ellipsoid***	***MINVOL Ellipsoid***	***MINSQUARED Ellipsoid***
Learning Set Ratio	20% learning	20% learning	20% learning	20% learning
Learn Error Rate	0%	0%	0%	0%
Test Error Rate	32.4%	8.4%	38%	4.3%

TABLE IV. RESULTS OF THEGLASS IDENTIFICATION PROBLEM

CLASSIFIER TYPE	**THE VOTING CLASSIFIER**		**THE N-ELLIPSOIDAL CLASSIFIER**	
Separating Ellipsoid Criteria	***MINVOL Ellipsoid***	***MINSQUARED Ellipsoid***	***MINVOL Ellipsoid***	***MINSQUARED Ellipsoid***
Learning Set Ratio	20% learning	20% learning	20% learning	20% learning
Learn Error Rate	0%	0%	0%	0%
Test Error Rate	35.2%	23.4%	32.7%	20.9%

TABLE V. ACCURACY COMPARISON BETWEEN VC, NEC, AND OTHER CLASSIFICATION ALGORITHMS

Data Set	**Comparison of Accuracy (%)**								
	KNN	***SVM***	***C4.5 1993***	***CBA 1998***	***CMAR 2001***	***LA 2009***	***ISAF 2010***	***Voting Classifier***	***NEC***
Fisher Iris	96	92	95.3	94.7	94	96	-	***97.6***	97.5
Wine	82.2	74.7	92.7	95	95	85.5	***97.2***	91.6	95.7
Glass	68.4	69.2	68.7	73.9	70.1	69.2	66.4	76.6	***79.1***

5. CONCLUSIONS

Two classifiers are proposed. They are based on constructing ellipsoids that effectively separate different classes. Experimental results indicate the potential of the introduced classifiers. They outperform most of the existing prominent classifiers in terms of error rates, accuracy, and complexity of the classifier.

6. REFERENCES

[1] De Klerk E., "Aspects of Semidefinite Programming", Kluwer Academic Publisher, 2002.

[2] Ma W.-K., Davidson T. N., Wong K. M., Luo Z.-Q, and Ching P.-C., "Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA", IEEE Transactions on Signal Processing, vol. 50, no. 4, pp. 912-922, Apr. 2002.

[3] Pozo F., Pujol G., Rodellar J., "Nonlinear control of uncertain systems via semidefinite programming", Proceedings - IEEE International Symposium on Intelligent Control and 13th Mediterranean Conference on Control and Automation, v 1, p 382-386, 2005.

[4] Bengtsson M. and Ottersten B., "Optimal and suboptimal transmit beamforming", chapter 18 in Handbook of Antennas in Wireless Communications, Godara L. C., Ed. Boca Raton, FL: CRC Press, Aug. 2001.

[5] Liu B. , and Tan S. X. D. , "Minimum decoupling capacitor insertion in VLSI power/ground supply networks by semidefinite and linear programs", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, v 15, n 11, p 1284-1287, November 2007.

[6] Cheung K. W., Ma W. K., So H. C., "Accurate approximation algorithm for TOA-based maximum likelihood mobile location using semidefinite programming", Proceedings - IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), v 2, p II145-II148, 2004.

[7] Shen C., Li H., and Brooks M. J., "Supervised dimensionality reduction via sequential semidefinite programming", Pattern Recognition, v 41, n 12, p 3644-3652, December 2008.

[8] J. Han and M. Kamber, "Data Mining: Concepts andTechniques,"Morgan Kaufmann, 2006.

[9] G.P. Zhang, "Neural networks for classification: Asurvey," IEEE Trans. on Systems Man and CyberneticsPart C –Applications and Reviews 30 (4), 2000, pp.451-461.

[10] O. Cordon, F. Gomide, F. Herrera, F. Hoffmann and L. Magdalena, "Ten years of genetic fuzzy systems:current framework and new trends,"Fuzzy Sets andSystems, Volume 141, Issue 1, 2004, pp. 5-31.

[11] H. Ishibuchi and Y. Nojima, "Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning," InternationalJournal of Approximate Reasoning, Volume 44, Issue 1,2007, pp. 4-31.

[12] H. Ishibuchi and T. Nakashima, "Effect of Rule Weights in Fuzzy Rule-Based Classification Systems," IEEETrans. on Fuzzy Systems, volume 9, no. 4, 2001, pp.

[13] K. Wang, S. Zhou, and Y. He. Growing decision tree on support-less association rules. In KDD'00, Boston, MA,Aug. 2000.

[14] Fr Glineur. "Pattern separation via ellipsoids and conic optimization". Thesis.

[15] Y.E. Nesterov and A. S. Nemirovsky. "Interior-point polynomial methods in convex programming". SIAM studies in Applied Mathematics, SIAM Publications, Philadelphia, 1994.

[16] Etienne DE KLERK. "Interior Point methods for Semidefinite Programming". Text book.

[17] C.J. Merz and P.M. Murphy. "UCI respository of machine learning databases". Unoversity of California, Irvine, Dept. of Information and comp. science http://www.ics.uci.edu/~mlearn/MLReporsitory.html.

[18] SeDuMi. A Matlab toolbox for optimization over symmetric cones. Sedumi.ie.lehigh.edu.

[19] Wenmin Li, Jiawei Han, Jian Pei. "CMAR: Accurate and Efficient Classification Based on MultipleClass-Association Rules".inData Mining, 2001. ICDM 2001,San Jose, CA, Dec.2001.

[20] B. Liu, W. Hsu, and Y. Ma. "Integrating classification and association rule mining". In KDD'98, New York, NY, Aug. 1998.

[21] J. R. Quinlan. "C4.5": Programsfor Machine Learning. Morgan Kaufmann, 1993.

[22] Majumdar, A.; Bhattacharya, A. "Classification by Linearity Assumption". In Seventh International Conference on Advances in Pattern Recognition 2009, ICAPR '09 Kolkata, Feb. 2009.

[23] Chung-Ru Dong, Patrick P. K. Chan, Wing W. Y. NG, Daniel S. Yeung."2-Stage instance selection algorithm for KNN based on nearest unlike neighbors".the Ninth International Conference on Machine Learning and Cybernetics, Qingdao, 11-14 July 2010.