Fast Robust Twin Support Vector Clustering

Qiao-lin YE*, Heng-hao ZHAO and Meem Naiem

College of Information Science and Technology, Nanjing Forestry University, Nanjing, Jiangsu 210037, P. R. China

Jiangsu Key Laboratory of Image and Video Understanding for Social Safety, Nanjing University of Science and Technology, 210094, P. R. China

*Corresponding author

Keywords: Robust, Twin support vector clustering, Fast robust twin support vector clustering.

Abstract. This paper develops a fast k-plane clustering method called L1-norm Distance Minimization based Fast Robust TWSVC (FRTWSVC) by using robust L1-norm distance. To solve the resulted objective, we propose a novel iterative algorithm. Only a system of linear equations needs to be computed in each iteration. These characteristics make our methods more powerful and efficient than TWSVC. We also conduct some insightful analysis on the convergence of the proposed algorithms. Theoretical insights and effectiveness of our method are further supported by promising experimental results.

Introduction

Clustering, as one of the fundamental topics in machine learning and pattern classification, has been widely applied to various areas, such as text mining, web analysis, and bioinformatics [1]-[4]. There are many clustering techniques in the literatures. For example, [5] and [6] considered kernel-based clustering, [7] and [9] used max-margin constraint in the clustering, and [9]-[11] proposed point-based central clustering techniques, e.g., k-mean, k-median, and fuzzy c-means (FCM).

In recent years, there has been increasing interest in k-plane clustering, which changes the entity of the center from being a point to that of being a plane [12]. kPC [12], Proximal Plane Clustering (PPC) [13], and Twin Support Vector Clustering (TWSVC) [14] are three typical k-plane clustering techniques. kPC only considers the similarities among the samples in a cluster plane but ignores the dissimilarities. PPC is proposed to address the problem. In fact, PPC is based on Multisurface Proximal Support Vector Machine Classification via Generalized Eigenvalues (GEPSVM) [15], while TWSVC is an extension to Twin Support Vector Machine (TWSVM) [16], [17], [20]. Among kPC, PPC, and TWSVC, TWSVC gains the best clustering result. However, this method formulates the objective using squared L2-norm distance in a cluster plane, which could exaggerate the effect of outliers. Furthermore, TWSVC is computationally expensive, since it requires solving a series of constrained quadratic programming problems (CQPPs) to determine each of the k cluster planes. This is an open problem raised in Section “Conclusion” of [14].

In this paper a novel L1-norm distance minimization based fast robust TWSVC (FRTWSVC) method is proposed. It is well-known that L1-norm distance is more robust to outliers than the L2-norm one, since it does not magnify the effect of outliers [18], [19]. As one of the important theoretical contributions of this paper, we present a novel iterative algorithm for the derivation of the clustering planes. In each iteration of the algorithm, only a system of linear equations needs to be solved. In addition, some insightful analysis on the convergence of the proposed algorithms are conducted. Theoretical studies and extensive experimental results on several benchmark datasets verify the effectiveness and applicability of our methods.
The Proposed Method

Suppose that $\mathbf{X} = \{x_1, \ldots, x_n\} \in \mathbb{R}^{m \times n}$ is the dataset with $m$ samples of $n$ dimensions. The L2-norm of a vector is denoted by $\| \cdot \|$. Let $\text{sgn}(\cdot)$ be a sign function with $\text{sgn}(\cdot) = -1$ if $(\cdot)$ is a negative value and $\text{sgn}(\cdot) = 1$ otherwise. The primary task of clustering is to partition $\mathbf{X}$ into $k$ clusters. We represent the samples in the $i$-th cluster by $\mathbf{X}_i \in \mathbb{R}^{m \times n}$ and those in the rest clusters by $\mathbf{X} \in \mathbb{R}^{m \times n \times k}$, where $m$ denotes the number of samples in the $i$-th cluster ($i = 1, \ldots, k$). A column vector of ones of arbitrary dimension is $\mathbf{e}$. Define three augmented matrices: $z_i = [w_i^T, b_i^T] \in \mathbb{R}^{d+1}$, $G_i = \{X_i, e_i\} \in \mathbb{R}^{m \times n \times (d+1)}$, and $H_i = [\mathbf{X}_i, e_i] \in \mathbb{R}^{m \times n \times (d+1)}$.

Recently, Wang et al. [14] proposed a more powerful k-plane clustering method than kPC and PPC, called TWSVC, which is based on TWSVM [17]. With the initial cluster assignment of $\mathbf{X}$, TWSVC finds the $k$ cluster planes of (1) by solving the following problem with $i = 1, \ldots, k$

$$\min_{\mathbf{w}_i, b_i} 0.5\|\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i\|^2 + c \mathbf{e}^T \mathbf{z}_i^T \mathbf{a}_i, \text{ s.t. } \mathbf{U}_i^T (\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i) + \mathbf{z}_i \geq \mathbf{e}, \mathbf{e}_i \geq 0, \quad (1)$$

where $\mathbf{z}_i$ is a symmetric Hinge loss function. Then use (3) to update the corresponding clusters of the samples. Based on the updated clusters, we proceed to update the $k$ cluster planes of (1). The process continues till some terminate conditions are satisfied. Rewrite (1) as

$$\min_{\mathbf{w}_i, b_i} 0.5\|\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i\|^2 + c \mathbf{e}^T \mathbf{z}_i^T \mathbf{a}_i, \text{ s.t. } \mathbf{U}_i^T (\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i) + \mathbf{z}_i \geq \mathbf{e}, \mathbf{e}_i \geq 0, \quad (2)$$

where $\mathbf{U}_i = \text{diag}(\text{sign}(\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i))$ is a diagonal matrix. The constraint of the problem is nonconvex. TWSVC solves (2) using the Constrained Concave–convex Procedure (CCCP) [21], where $\mathbf{U}_i$ is viewed as a variable that depends on $\mathbf{w}_i$ and $b_i$ (or $z_i$). Specifically, compute $\mathbf{U}_i$ based on the current $z_i$ obtained in last iteration and then update $z_i$ by solving the Wolfe dual problem of (6). Suppose that $z_i^{(p)}$ is the solution of the $p$-th iteration, and $z_i^{(p+1)}$ is the one of the $(p+1)$-th iteration, where $z_i^{(p+1)} = (\mathbf{G}_i^T \mathbf{G}_i)^{-1} \mathbf{M}_i^{p} \mathbf{a}_i^{(p+1)}$. Here $\mathbf{M}_i^{p} = \mathbf{U}_i^{p} \mathbf{H}_i$, and $\mathbf{a}_i^{(p+1)}$ is the updated Lagrangian multiplier vector that is defined as

$$\mathbf{a}_i^{(p+1)} = \arg\min_{\mathbf{a}_i} 0.5 \mathbf{a}_i^T \mathbf{M}_i^{p} (\mathbf{G}_i^T \mathbf{G}_i) \mathbf{a}_i - \mathbf{e}^T \mathbf{a}_i, \text{ s.t. } 0 \leq \mathbf{a}_i \leq \mathbf{c}. \quad (3)$$

The results of [17] demonstrate the promising performance of TWSVC.

It has been well known that squared L2-norm distance measurement is not robust to outliers, which means that TWSVC may not obtain the desired solution. In literature, the L1-norm distance is usually applied to handle this problem [18], [19]. Illuminated by this, we propose a new method called FRTWSVC. Same as TWSVC, we first partition the samples of $\mathbf{X}$ into $k$ clusters, and then use

$$\min_{\mathbf{w}_i, b_i} \|\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i\|_1 + c \|\mathbf{z}_i\|_1, \text{ s.t. } \mathbf{U}_i (\mathbf{X}_i \mathbf{w}_i + \mathbf{b}_i) + \mathbf{z}_i = \mathbf{e}. \quad (4)$$

Note that our problem is also based on the efficient least squares version of TWSVM [24] for improving the computational efficiency. We call the reformulation FRTWSVC. We can rewrite (4) with the following problem

$$\min_{\mathbf{z}_i} \|\mathbf{G}_i \mathbf{z}_i\|_1 + c \|\mathbf{z}_i\|_1, \text{s.t. } \|\mathbf{H}_i \mathbf{z}_i\| + \mathbf{z}_i = \mathbf{e}. \quad (5)$$

Rewrite (5) with the following equivalent formulation

$$\min_{\mathbf{z}_i} \sum_j (\mathbf{g}_{ij}^T \mathbf{g}_{ij} / |\mathbf{g}_{ij}|) \cdot |\mathbf{z}_{ij}| + c \sum_j (|\mathbf{z}_{ij}|)^2 / |\mathbf{z}_{ij}|), \text{s.t. } \text{diag}(\text{sign}(\mathbf{H}_i \mathbf{z}_i))(\mathbf{H}_i \mathbf{z}_i) + \mathbf{z}_i = \mathbf{e}, \quad (6)$$

where $z_{ij}$ the $j$-th element of $z_i$. Let $a_{ij} = 1 / |z_{ij}|$ and construct the diagonal matrix $\mathbf{A}$, its $j$-th diagonal entry as $a_{ij}$. To the end, (6) becomes
\[
\min_{z_j} z_j^T D_j G_j z_j + c \xi_j^T A_j \hat{\xi}_j, \quad \text{s.t.} \quad F_j (H_j z_j) + \hat{\xi}_j = \mathbf{e}.
\]

We can propose the similar iterative algorithm of Algorithm 1 to obtain the solution \( z_j \). The algorithm is described in Algorithm 1.

In each iteration of Algorithm 1, one needs to solve the least squares problem in step 4. Substituting the equality constraints into the objective function, the problem becomes

\[
z_j^{(p+1)} = \min_{z_j} z_j^T G_j D_j G_j z_j + c (e - F_j^{(p)} H_j z_j)^T A_j^{(p)} (e - F_j^{(p)} H_j z_j).
\]

Setting the derivative of (17) w.r.t. \( z_j \) as zero, we obtain

\[
z_j^{(p+1)} = (1/c G_j^T D_j G_j + H_j^T F_j^{(p)} A_j^{(p)} F_j^{(p)} H_j)^{-1} H_j^T F_j^{(p)} A_j^{(p)} e.
\]

Since \( F_j^{(p)} \) and \( A_j^{(p)} \) are diagonal matrices, and each of the diagonal elements of \( F_j^{(p)} \) are either 1 or -1, \( F_j^{(p)} A_j^{(p)} F_j^{(p)} = A_j^{(p)} \). Therefore, we have

\[
z_j^{(p+1)} = (1/c G_j^T D_j G_j + H_j^T A_j^{(p)} H_j)^{-1} H_j^T A_j^{(p)} e.
\]

Hence, we can compute a system of linear equations in (9).

| Algorithm 1: An efficient iterative algorithm to solve problem (4) |
|-------------------------------------------------------------|
| **Input**: Input data matrices \( X \) and \( \tilde{X} \).  |
| Initialize \( z_j^{(0)} \), set \( p = 0 \), and construct the matrices \( G_j = [X \ e] \) and \( H_j = [X \ e] \).  |
| **While** not converge **do** |
| 1. Compute \( D_j^{(p)} \) with its \( j \)-th diagonal entry \( d_{j,j}^{(p)} = 1/|g_{j,j}| \), where \( g_{j,j} \in \mathbb{R} \) denotes the \( j \)-th row of \( G_j \).  |
| 2. Compute the diagonal matrix \( F_j^{(p)} = \text{diag}(|H_j z_j^{(p)}|) \).  |
| 3. Compute the diagonal matrix \( A_j^{(p)} \) with its \( j \)-th diagonal entry as \( a_{j,j}^{(p)} = 1/|\xi_j| \), where \( \xi_j \) is the \( j \)-th element of \( \xi \).  |
| 4. Compute \( z_j^{(p+1)} \) by solving \( z_j^{(p+1)} = \arg \min_{z_j} z_j^T G_j^T D_j G_j z_j + c \xi_j^T A_j^{(p)} \xi_j \), s.t. \( F_j^{(p)} (H_j z_j) + \xi_j = e \).  |
| 5. \( p = p+1 \).  |
| **End while**  |
| **Output**: The learned \( w_j \) and \( b_j \) from \( z_j = [w_j^T \ b_j]^T \).  |

The following theorems guarantee the convergence of Algorithm 1.

**Theorem 1**: Algorithm 1 monotonically decreases the objective of problem (4) in each iteration until convergence.

**Proof**: Rewriting the problem in step 4 of Algorithm 1 by substituting the equality constraint into the objective gives

\[
z_j^{(p+1)} = \min_{z_j} 0.5 z_j^T G_j^T D_j G_j z_j + 0.5 c (e - F_j^{(p)} H_j z_j)^T A_j^{(p)} (e - F_j^{(p)} H_j z_j),
\]

which indicates

\[
0.5 z_j^{(p+1)} G_j^T D_j G_j z_j^{(p+1)} + 0.5 c (e - F_j^{(p)} H_j z_j^{(p+1)})^T A_j^{(p)} (e - F_j^{(p)} H_j z_j^{(p+1)}) \leq 0.5 z_j^{(p)} G_j^T D_j G_j z_j^{(p)} + 0.5 c (e - F_j^{(p)} H_j z_j^{(p)})^T A_j^{(p)} (e - F_j^{(p)} H_j z_j^{(p)})
\]

(11)

According to step 3 of Algorithm 1, we can define \( \xi_j^{(p)} = e - F_j^{(p)} H_j z_j^{(p)} \) and \( \xi_j^{(p)} = e - F_j^{(p)} H_j z_j^{(p)} \). The former is the update of the latter. Applying the definitions of \( D_j^{(p)} \) and \( A_j^{(p)} \) in step 1 and step 3, problem (10) can be rewritten as the following one by decoupling the computation for each row for \( D_j^{(p)} \) and \( A_j^{(p)} \)

\[
0.5 \left( \sum_{j=1}^n \frac{(g_{j,j} z_j^{(p)})^2}{|g_{j,j}|^2} + c \sum_{j=1}^n \frac{(\xi_j^{(p)})^2}{|\xi_j|} \right) \leq 0.5 \left( \sum_{j=1}^n \frac{(g_{j,j} z_j^{(p)})^2}{|g_{j,j}|^2} + c \sum_{j=1}^n \frac{(\xi_j^{(p)})^2}{|\xi_j|} \right)
\]

(12)

For each \( j \), we have \( (|g_{j,j} z_j^{(p)}| - |g_{j,j} z_j^{(p)}|)^2 + (g_{j,j} z_j^{(p)})^2 - 2 |g_{j,j} z_j^{(p)}| |g_{j,j} z_j^{(p)}| \geq 0 \), which leads to

\[
\frac{(g_{j,j} z_j^{(p)})^2}{|g_{j,j} z_j^{(p)}|} + |g_{j,j} z_j^{(p)}| \geq 0 \Rightarrow 2 |g_{j,j} z_j^{(p)}| - (g_{j,j} z_j^{(p)})^2 \leq 2 |g_{j,j} z_j^{(p)}| - (g_{j,j} z_j^{(p)})^2
\]

(13)

Using the inequality and adding it to (12) gives
\[ \|G, z^{(s+1)}\| + 0.5c \sum_{j} \frac{(\xi_{i,j}^{(s+1)})^2}{|\xi_{i,j}^{(s)}} \leq \|G, z^{(s)}\| + 0.5c \sum_{j} \frac{(\xi_{i,j}^{(s)})^2}{|\xi_{i,j}^{(s)}} \]  

With the similar proof of (22), it is easy to conclude that

\[ c \sum_{j} \left( |\xi_{i,j}^{(s+1)}| - 0.5 \frac{(\xi_{i,j}^{(s+1)})^2}{|\xi_{i,j}^{(s)}} \right) \leq c \sum_{j} \left( |\xi_{i,j}^{(s)}| - 0.5 \frac{(\xi_{i,j}^{(s)})^2}{|\xi_{i,j}^{(s)}} \right) \]

Combining (14) and (15) leads to

\[ \|G, z^{(s+1)}\| + c\|\xi^{(s+1)}\| \leq \|G, z^{(s)}\| + c\|\xi^{(s)}\|. \]  

Algorithm 1 converges, since the problem in (4) has a lower bound 0. In such case, the equality in (16) holds. Therefore, the objective value of problem (4) decreases in each iteration till the algorithm converges.

**Experiments**

To evaluate the performance of the proposed FRTWSVC, we conduct experiments on several benchmark datasets. To measure the clustering performance, we use the metric accuracy, which is defined in [14]. Following [14], the initial cluster labels of each method are selected using the effective EEG-based initialization. For a linear case, PPC, TWSVC, and FRTWSVC have two common parameters \( c \) and \( P \) (neighborhood size in the EEG-based initialization). The parameter \( c \) is selected from the values \( \{2, 4, \ldots, 4\} \), while \( P \) is selected from the values \( \{1, 2, \ldots, 5\} \) as in [14]. In using TWSVC and FRTWSVC, the initial cluster plane is set as the solution of PPC.

Tables 1 shows the clustering performance and computing time of kPC, PPC, TWSVC, and FRTWSVC. From Tables I, we observe first that FRTWSVC yield better accuracy than other methods. Secondly, the computational advantage of our FRTWSVC over TWSVC is extremely obvious. Anyway, our results likewise show that FRTWSVC is better than TWSVC when directly coping with the original high dimensional data. Finally, we can observe that TWSVC outperforms kPC and PPC, which is consistent to [14]. To illustrate the robustness of each method, we corrupt the training set using a noise matrix \( N \), whose element is i.i.d. standard Gaussian variables. Given the training set \( X \), each method learns on the corrupted training set \( X + \psi N \), where \( \psi = \kappa \|X\|_F \|N\| \) is a given noise factor. In the experiment, \( \kappa \) is set as 0.2. Tables 2 reports the clustering performance and computing time. As can be seen, on most datasets the performance of each method is more or less impaired by the corruption. Even so, our d FRTWSVC significantly outperforms other methods in most cases. In computational cost, TWSVC is inferior to FRTWSVC. Considering both accuracy and efficiency, FRTWSVC is the best choice among all the compared methods.
Table 1. Clustering Performance and Computing Time of kPC, PPC, TWSVC, And FRTWSVC.

| Data set m:n | kPC Acc./Time | PPC Acc./Time | TWSVC Acc./Time | FRTWSVC Acc./Time |
|--------------|---------------|---------------|-----------------|-------------------|
| Leaf 340×15  | 89.78/0.031   | 67.89/0.209   | 94.56/183.8     | 93.97/84.94       |
| Dematoloy 366×34 | 60.50/0.015 | 60.50/0.082   | 70.13/93.47     | 70.82/14.57       |
| Ecoli 336×7   | 42.55/0.006   | 77.89/0.050   | 77.89/39.19     | 79.91/2.590       |
| Wine 178×13   | 57.93/0.003   | 73.17/0.006   | 73.65/3.832     | 92.84/1.472       |
| Haberm 306×3  | 55.86/0.003   | 60.95/0.007   | 61.26/1.215     | 61.26/0.294       |
| Iris 150×4    | 67.54/0.003   | 83.68/0.004   | 91.24/0.904     | 95.75/0.550       |
| Glass 214×9   | 68.70/0.0112  | 65.71/0.037   | 65.56/11.088    | 63.75/1.687       |
| Vowel 528×10  | 83.91/0.033   | 83.32/0.071   | 83.37/949.1     | 84.25/61.17       |
| Zoo 101×16    | 55.39/0.001   | 81.56/0.049   | 88.91/2.288     | 88.59/2.262       |
| Mush 8124×22  | 89.40/15.74   | 89.49/2.202   | 89.45/754.4     | 89.49/320.1       |

Table 2. Clustering Performance and Computing times of kPC, PPC, TWSVC, And FRTWSVC on the Noisy Data Sets.

| Data set m:n | kPC Acc./Time | PPC Acc./Time | TWSVC Acc./Time | FRTWSVC Acc./Time |
|--------------|---------------|---------------|-----------------|-------------------|
| Leaf 340×15  | 93.79/0.0237  | 79.87/0.2297  | 94.46/91.09     | 91.27/58.49       |
| Userknow 403×5 | 58.09/0.0161 | 65.30/0.0161  | 67.30/37.12     | 68.20/1.021       |
| Teachinga 151×5 | 50.38/0.0028 | 54.84/0.0093  | 52.49/2.360     | 55.77/0.291       |
| Cleveland 297×13 | 50.31/0.0183 | 63.14/0.0110  | 59.89/1.688     | 59.89/0.209       |
| Dermatoloy 366×34 | 70.21/0.0628 | 62.58/0.1736  | 70.14/84.35     | 70.25/8.963       |
| Haberman 306×3  | 49.89/0.0043  | 60.95/0.0053  | 61.57/1.735     | 60.95/0.173       |
| Iris 150×4    | 64.54/0.0032  | 81.68/0.0073  | 77.77/0.891     | 84.15/0.1172      |
| Glass 214×9   | 66.43/0.0069  | 68.08/0.0305  | 64.85/12.61     | 66.96/1.683       |
| Vowel 528×10  | 83.11/0.0377  | 80.32/0.0377  | 82.77/817.3     | 83.91/32.24       |
| Zoo 101×16    | 79.86/0.0067  | 78.30/0.0544  | 84.79/0.632     | 89.57/0.632       |
| Mush 8124×22  | 50.27/10.169  | 88.88/2.026   | 89.47/7215.8    | 89.49/311.89      |

Summary
We presented a new k-plane clustering method, called FRTWSVC. The objective of FRTWSVC was formulated using robust L1-norm distance. The resulted objective was solved by a newly-designed iterative algorithm. In each round of optimization of the algorithm, we solved only a system of linear equations. We presented some insightful analysis on the convergence of the algorithm. Furthermore, The experimental results on benchmark datasets indicated first that both methods yield better
accuracy than existing k-pane clustering methods, and second that FRTWSVC runs by far faster than
TWSVC.

References
[1] M. Berry, “Survey of text mining I: clustering, classification, and retrieval,” vol.1, Berry, Michael
W., Ed. Springer, Berlin, 2004.
[2] A. Hotho, A. Nrnberger, and G. Paab, “A brief survey of text mining,” Journal for Computational
Linguistics and Language Technology, vol. 20, no. 1, pp. 19–62, 2005.
[3] S. Bandyopadhyay, A. Mukhopadhyay, and U. Maulik, “An improved algorithm for clustering
gene expression data,” Bioinformatics, vol. 23, no. 21, pp. 2859–2865, 2007.
[4] A. Jain and R. Dubes, “Algorithms for Clustering Data,” Englewood Cliffs, NJ: Prentice-Hall,
1988.
[5] H. Jia, Y.M. Cheung and J. Liu, "Cooperative and Penalized Competitive Learning with
Application to Kernel-based Clustering,” Pattern Recognition, vol. 47, no. 9, pp. 3060-3069, 2014.
[6] Y. F. Li, I. W. Tsang, J. T. Kwok, and Z. H. Zhou. "Tighter and convex maximum margin
clustering,” in Proc. Int. Conf. Artif. Intell and Stats. Clearwater Beach, Flori, 2009, pp. 344-351.
[7] H. Zeng and Y.M. Cheung, "Semi-supervised Maximum Margin Clustering with Pairwise
Constraints,” IEEE Trans. Knowl. Data Eng., vol. 24, no. 5, pp. 926-939, 2012.
[8] L. L. Xu, N. James, L. Bryce, and S. Dale, "Maximum margin clustering,” in Proc. Int. Conf. Adv.
Neural Inf. Process. Syst., Vancouver, British Columbia, Canada, 2004, pp. 1537-1544.
[9] M. Anderberg, “Cluster analysis for applications,” Academic Press, New York, 1973.
[10]X. Huang, Y. Ye, and H. Zhang, “Extensions of k-means-type algorithms: A new clustering
framework by integrating intracluster compactness and intercluster separation,” IEEE Trans. Neural
Netw. Learn. Syst., vol. 25, no. 8, pp. 1433–1446, Aug. 2014.
[11]P. Bradley and O. Mangasarian, “Clustering via concave minimization,” in Proc. Int. Conf. Adv.
Neural Inf. Proces. Syst., Denver, Colorado, USA, 1997, pp. 368–374.
[12]P. S. Bradley and O. L. Mangasarian, “k-plane clustering,” J. Global Optim., vol. 16, no. 1, pp.
23–32, 2000.
[13]Y. H. Shao, L. Bai, Z. Wang, X. Y. Hua, et al., “Proximal plane clustering via eigenvalues,”
Procedia Computer Science, vol. 17, pp. 41–47, May 2013.
[14]Z. Wang, Y. Shao, L. Bai, and N. Deng, “Twin Support Vector Machine for Clustering,” IEEE
Trans. Neural Netw. Learn. Syst., vol. 16, no. 26, pp. 2583–2588, Oct. 2015.
[15]O. L. Mangasarian and E. W. Wild, “Multisurface proximal support vector machine classification
via generalized eigenvalues,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 1, pp. 69–74, Jan.
2006.
[16]Y. H. Shao, C.H. Zhang, X. B. Wang, et al., “Improvements on twin support vector machines,”
IEEE Trans. Neural Netw., vol. 22, no. 6, pp. 962–968, Jun. 2011.
[17]Jayadeva, R. Khemchandani, and S. Chandra, “Twin support vector machines for pattern
classification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 5, pp. 905–910, May 2007.
[18]Q. L. Ye, J. Yang, F. Liu, et al, “L1-norm Distance Linear Discriminant Analysis Based on An
Effective Iterative Algorithm,”IEEE Transactions on Circuits and Systems for Video Technology,
vol. pp, no. 99, pp. 1-1, 2016.
[19] F. J. Zhong, Discriminant, “Locality Preserving Discriminants Based on L1-Norm Maximization,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 11, pp. 2065-2074, 2014.

[20] Q. L. Ye, N. Ye, and T.M. Yin, “Enhanced Multi-weight Vector Projection Support Vector Machine,” Pattern Recognition Letters, vol. 42, pp. 91-100, 2014.

[21] Q. L. Ye, C. X. Zhao, H. F. Zhang, and X. B. Chen, “Recursive Concave–convex Fisher Linear Discriminant with Applications to Face, Handwritten Digit and Terrain Recognition,” Pattern Recognition, vol. 45, no. 1, pp. 54–65, 2012.