Pattern Recognition 45 (2012) 844–862

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A boosting approach for supervised Mahalanobis distance metric learning ☆

Chin-Chun Chang

Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung 202, Taiwan

ARTICLE INFO

Article history:
Received 6 August 2010
Received in revised form
27 July 2011
Accepted 31 July 2011
Available online 10 August 2011

Keywords:
Distance metric learning
Hypothesis margins
Boosting approaches

ABSTRACT

Determining a proper distance metric is often a crucial step for machine learning. In this paper, a boosting algorithm is proposed to learn a Mahalanobis distance metric. Similar to most boosting algorithms, the proposed algorithm improves a loss function iteratively. In particular, the loss function is defined in terms of hypothesis margins, and a metric matrix base-learner specific to the boosting framework is also proposed. Experimental results show that the proposed approach can yield effective Mahalanobis distance metrics for a variety of data sets, and demonstrate the feasibility of the proposed approach.

1. Introduction

Many supervised learning tasks need distance metrics for defining proper distance relationships among data. Although nonlinear distance metrics are more general, we aim for the Mahalanobis distance metric in this study because of two reasons. First, some nonlinear distance metrics are defined in terms of the Mahalanobis distance metric. Second, the relevance of the component of the measurement vector may be easily interpreted through the Mahalanobis distance metric.

For a comprehensive survey of distance metric learning, the reader may refer to [1]. In general, the distance metric can be global or local. The local distance metric is required to be accurate around a query point, whereas the global distance metric is desired to be proper globally. Here, we focus on the global distance metric. The eigenvector method is a popular approach to learning a global distance metric. This method is aimed at a linear transformation optimizing some criterion function, which actually induces a Mahalanobis distance metric. Some popular eigenvector methods for supervised distance metric learning are Fisher's discriminant analysis (FDA) [2], relevant component analysis (RCA) [3], heteroscedastic linear discriminant analysis (HLDA) [4], marginal Fisher analysis (MFA) [5], local Fisher's discriminant analysis (LFDA) [6], nonparametric discriminant analysis [7], average neighborhood margin maximization (ANMM) [8], and the approaches of Hastie and Tibshirani [9], and Fukunaga and Flick [10]. Many eigenvector methods, such as FDA and MFA, are under the framework of graph embedding [11]. However, some of them such as FDA and HLDA assume that the underlying sample distributions follow some models; the others may need specified target neighbors for each sample or rely on the neighbors defined by the original metric.

On the other hand, Xing et al. [12] formulated the problem of distance metric learning as a convex programming problem with equivalence and inequivalence constraints. The equivalence constraint forces the pair of semantically similar samples to be close together in the learned metric space. The inequivalence constraint makes the pair of dissimilar samples not near in that space. Based on graph embedding, Yan et al. [13] learned a distance metric by solving a semi-definite programming problem with neighborhood homogeneity constraints. The approach of Weinberger et al. [14], referred as to LMNN, uses the semi-definite programming for learning distance metrics. The objective of LMNN is to find a distance metric such that every sample is close to its pre-specified target neighbors and far away from the samples with different class labels. Jin et al. [15] also applied the semi-definite programming to learn a distance metric. Wang et al. [16] applied ANMM on metric learning in a semi-supervised setting. These approaches rely on predefined equivalence and inequivalence constraints. However, these constraints may be difficult to specify when the underlying distribution of the sample of a class exhibits multi-modal.

Local distance metric (LDM) [17] learns distance metrics by optimizing the compactness of a class and the separability among classes in a local sense. Since the metric matrix induced by LDM in fact re-weights the principal component of training samples, the metric matrix induced by LDM is not general. Neighborhood component analysis (NCA) [18] is based on stochastic nearest neighbors and learns a Mahalanobis distance metric for k NN classification. Although defining neighborhood relationships in an adaptive and local sense, NCA is not ensured to be converged. The RELIEF-based approach [19,20] learns a weight for each feature based on local and adaptive neighborhood relationships. The RELIEF-based approach has

☆This work was supported financially by National Science Council under the grant NSC 99-2221-E-019-036 and in part by NTOU-RD981-05-02-04-01.

E-mail address: cvml@mail.ntou.edu.tw

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.07.026

a nice convergency property; however, it may be inappropriate in the presence of highly correlated features.

The boosting technique [21–23] is successful in various problems of machine learning, and Adaboost [24,25] is perhaps the most famous one. Some approaches to learning distance metrics also adopt the boosting technique but most of them learn nonlinear distance metrics [26–28]. In particular, Crammer et al. [29] have used the boosting technique with the alignment loss to learn a kernel matrix in a semi-supervised setting. Their approach can produce a metric matrix in fact. The alignment loss is based on the principle that two samples which should be similar should have a large inner-product value; on the contrary, the inner-product value of two samples which should be dissimilar should be small. Since two samples having a large inner-product value are not necessary to have a short Euclidean distance between them, the alignment loss may not be suitable for inducing a Mahalanobis distance metric.

In this study, the class label of a training sample is assumed to be assigned semantically, and no target neighbors for a training sample are known beforehand. Here, the boosting technique is used to yield a Mahalanobis distance metric for defining proper distance relationships among samples. Since having the following two characteristics, the proposed approach could learn distance metrics suitable for the nearest neighbor (NN) classification.

First, the proposed approach makes use of a hypothesis margin-based loss function. This loss function is based on the nearest miss and hit of the sample, and defines an upper bound for the leave-one-out training error of the NN classification. In the literature, two types of margin, namely, the sample margin and the hypothesis margin, have been proposed [30,31]. The sample margin is referred to the distance from a sample to the discriminatory boundary. Support vector machines are based on the sample margin. The hypothesis margin is defined as the distance that the classifier can be moved in the sample space without changing the classification of the samples made by the classifier. Adaboost and the prototype-based classifier make use of the hypothesis margin. The metric matrix minimizing the proposed loss function not only enlarges the hypothesis margin but may also broaden the sample margin because the hypothesis margin is a lower bound of the sample margin [30]. In addition, whenever the proposed loss function is evaluated, the nearest neighbor classifier is invoked indeed. This mechanism is similar to the wrapper approach [32] for the feature subset selection. Therefore, the proposed loss function is suitable for learning a Mahalanobis distance metric for the NN classification.

Second, the proposed approach has a good convergency property in terms of the leave-one-out training error of the NN classification. The proposed approach learns a distance metric by combining metric matrices in a stagewise forward manner, and the proposed loss function is thus reduced iteratively. Besides, a metric matrix base-learner specific to the boosting framework is also developed. For every boosting iteration, the metric matrix to be combined is orthogonal to that learned in the previous stage. This strategy can include a variety of base metric matrices in a few boosting iterations, and make use of the whole sample space.

Besides, the proposed approach also provides a unified view for two algorithms of supervised distance metric learning, namely, MFA [5] and GI-RELIEF [33]. It turns out that MFA and GI-RELIEF learn the distance metric through optimizing hypothesis margin-based loss functions, and the difference between them is the way of defining the nearest hit and miss for a sample. Due to this unified view, we find that iterating MFA may be effective in improving the accuracy of MFA, and GI-RELIEF and an iterative version of MFA can be the base-learner of the proposed boosting approach. In [33], a connection is established between I-RELIEF [19] and GI-RELIEF. Thus, the proposed boosting framework could be a generalization of MFA, I-RELIEF, and GI-RELIEF.

The rest of this work is organized as follows. Section 2 introduces the proposed approach. Section 3 compares the proposed approach with MFA and GI-RELIEF. Section 4 shows the experimental results. The concluding remarks are at the last section.

2. The proposed approach

Suppose that the sample vector could be mapped through a linear transformation $\mathbf{L}^T$ onto a space such that in the mapped space, almost every sample and the neighbor of the sample have the same class label. Thus, the mapped sample is ideal for the NN classification. The Euclidean distance between the mapped vectors of two sample vectors $\mathbf{x}$ and $\mathbf{x}'$ can be calculated by $\|\mathbf{L}^T(\mathbf{x}-\mathbf{x}')\|_2=\sqrt{(\mathbf{x}-\mathbf{x}')^T\mathbf{M}(\mathbf{x}-\mathbf{x}')}\triangleq\|\mathbf{x}-\mathbf{x}'\|_{\mathbf{M}}$, where $\mathbf{M}=\mathbf{L}\mathbf{L}^T$. The matrix $\mathbf{M}$ is positive semi-definite and known as the *metric matrix*. For convenience, $\|\cdot\|_{\mathbf{M}}$ is called the $\mathbf{M}$-distance here. Supervised learning of an effective $\mathbf{M}$-distance is the target of this study.

In the following, a simple version of the proposed approach is introduced first. This simple version is defined in terms of the nearest hit and miss of the sample, and is mainly for guiding two more robust extensions. These two extensions are based on the loss functions defined in terms of averages of nearest hits and misses and will be discussed at last.

2.1. The loss function

Let $\mathcal{X}=\{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2),\ldots,(\mathbf{x}_N,y_N)\}$ be a set of N training samples drawn i.i.d. from an unknown distribution $\mathcal{D}$ over $\mathbb{R}^n\times\{1,\ldots,c\}$, where $\mathbf{x}_i$ denotes the measurement vector, and y_i denotes the class label of $\mathbf{x}_i$. Now, denote by $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})$ the NN classifier with the $\mathbf{M}$-distance and training set $\mathcal{X}$, which may be the prototype set for the NN classifier. Thus, $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})$ always assigns to $\mathbf{x}$ the class label of the prototype in $\mathcal{X}$ which has the shortest $\mathbf{M}$-distance to $\mathbf{x}$. Denote by $\mathcal{C}(\mathbf{x}_i)$ the set of the training vector having a class label identical to y_i. In addition, denote by $\mathcal{C}_h(\mathbf{x}_i)$ and $\mathcal{C}_m(\mathbf{x}_i)$ the sets of hits and misses of $\mathbf{x}_i$; that is, $\mathcal{C}_h(\mathbf{x}_i)=\mathcal{C}(\mathbf{x}_i)-\{\mathbf{x}_i\}$ and $\mathcal{C}_m(\mathbf{x}_i)=\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}-\mathcal{C}(\mathbf{x}_i)$. Thus, the leave-one-out error of $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$, denoted by $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$, can be defined as

$$\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})=\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_0\left(\min_{\mathbf{x}'\in\mathcal{C}_m(\mathbf{x}_i)}\|\mathbf{x}_i-\mathbf{x}'\|_{\mathbf{M}}^2-\min_{\mathbf{x}''\in\mathcal{C}_h(\mathbf{x}_i)}\|\mathbf{x}_i-\mathbf{x}''\|_{\mathbf{M}}^2\right),$$

where the θ-clipped loss function $\mathcal{L}_\theta(z)$ is defined as

$$\mathcal{L}_\theta(z)=\begin{cases}1, & z\le 0;\\ 1-\dfrac{z}{\theta}, & 0<z\le\theta;\\ 0, & z>\theta,\end{cases}$$

with $\theta\ge 0$. That is, $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$ calculates the ratio of the training sample of which the nearest hit is farther than the nearest miss with respect to the $\mathbf{M}$-distance. In addition, $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$ can be bounded above by

$$\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})\le\frac{1}{N}\sum_{i=1}^{N}\exp\left(\min_{\mathbf{x}'\in\mathcal{C}_h(\mathbf{x}_i)}\|\mathbf{x}_i-\mathbf{x}'\|_{\mathbf{M}}^2-\min_{\mathbf{x}''\in\mathcal{C}_m(\mathbf{x}_i)}\|\mathbf{x}_i-\mathbf{x}''\|_{\mathbf{M}}^2\right),\tag{1}$$

which is related to hypothesis margins for all training samples [30].

In addition, we define function $f_{\mathbf{M}}(\mathbf{x},\mathcal{S})$ as

$$f_{\mathbf{M}}(\mathbf{x},\mathcal{S})=\sum_{(i,\psi)\in\mathcal{S}}\psi\|\mathbf{x}_i-\mathbf{x}\|_{\mathbf{M}}^2,$$

where $\mathcal{S}$ is a set of pairs of the index of a sample and the averaging weight associated with the sample. Denote by $\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}}}(\mathbf{x})$ and $\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}}}(\mathbf{x})$ the sets of sample-index-and-averaging-weight pairs

for the k_h nearest hits and k_m nearest misses of $\mathbf{x}$ defined by the $\mathbf{M}$-distance, respectively; that is,

$$\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}}}(\mathbf{x})=\left\{\left(i,\frac{1}{k_h}\right)|\mathbf{x}_i\in k_h \text{ nearest hits of } \mathbf{x} \text{ defined by } \mathbf{M}\text{-distance}\right\}$$

$$\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}}}(\mathbf{x})=\left\{\left(i,\frac{1}{k_m}\right)|\mathbf{x}_i\in k_m \text{ nearest misses of } \mathbf{x} \text{ defined by } \mathbf{M}\text{-distance}\right\}.$$

Accordingly, Eq. (1) can be rewritten as

$$\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})\le\frac{1}{N}\sum_{i=1}^{N}\exp(f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i))-f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i)))\triangleq B(\mathbf{M}). \tag{2}$$

Since $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$ is discontinuous, hard to optimize, and inappropriate for yielding an effective metric matrix, $B(\mathbf{M})$ is used instead in the following sections.

2.2. The boosting framework

Denote by $\mathbf{M}_t$ the metric matrix learned in iteration t. Now, by given metric matrices $\mathbf{M}_{t-1}$ and $\mathbf{Q}_t$, $\mathbf{M}_t$ is learned by

$$\mathbf{M}_t=\mathbf{M}_{t-1}+\alpha_t\mathbf{Q}_t, \tag{3}$$

where $\alpha_t\ge 0$. Such a combination ensures that $\mathbf{M}_t$ is always positive semi-definite. Additionally, we also have

$$f_{\mathbf{M}_t}(\mathbf{x},\mathcal{S})=f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{S})+\alpha_t f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{S}).$$

Similar to most boosting algorithms, we first obtain $\mathbf{Q}_t$ by some base-learner, and then calculate α_t for combining $\mathbf{M}_{t-1}$ with $\mathbf{Q}_t$ such that $B(\mathbf{M}_t)$ is not greater than $B(\mathbf{M}_{t-1})$. To this end, an upper bound for $B(\mathbf{M}_t)$ is presented as follows.

2.2.1. An upper bound for $B(\mathbf{M}_t)$

Shown in Theorem 1 is an upper bound of the hypothesis margin for sample $\mathbf{x}$ with respect to metric matrix $\mathbf{M}_t$. This bound is expressed in terms of $\mathbf{M}_{t-1}$ and $\mathbf{Q}_t$, and is the basis for deriving an upper bound for $B(\mathbf{M}_t)$.

Theorem 1. *If $\mathbf{M}_t=\mathbf{M}_{t-1}+\alpha\mathbf{Q}_t$, in which $\mathbf{M}_{t-1}$ and $\mathbf{Q}_t$ are two metric matrices, and $\alpha\ge 0$, we have*

$$\begin{aligned}&f_{\mathbf{M}_t}(\mathbf{x},\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}))-f_{\mathbf{M}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}))\\&\quad\le f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{S}(\mathbf{x}))-f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}))\\&\qquad+\alpha(f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{S}(\mathbf{x}))-f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x})),\end{aligned}$$

where k_h and k_m are positive numbers, and $\mathcal{S}(\mathbf{x})=\{(i,\psi)|\psi=1/card(\mathcal{P}),\mathbf{x}_i\in\mathcal{P}$, where $\mathcal{P}\subseteq\mathcal{C}_h(\mathbf{x})$ and $card(\mathcal{P})\ge k_h\}$ with $card(\mathcal{P})$ denoting the cardinality of set $\mathcal{P}$.

Proof. See Appendix A.

Next, a corollary for deriving an upper bound for $B(\mathbf{M}_t)$ can be obtained as follows.

Corollary 2. *If $t\ge 1$, we have*

$$\exp(\delta_{i;t})\le w_{i;t-1}\exp(-\alpha_t d_{i;t}),$$

where

$$\delta_{i;t}=f_{\mathbf{M}_t}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}_i))-f_{\mathbf{M}_t}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}_i)), \tag{4}$$

$$w_{i;t-1}=\exp(\delta_{i;t-1}), \tag{5}$$

$$d_{i;t}=\begin{cases}f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i))-f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)) & \text{if } t>1,\\ f_{\mathbf{Q}_1}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_1}}(\mathbf{x}_i))-f_{\mathbf{Q}_1}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{Q}_1}}(\mathbf{x}_i)) & \text{if } t=1\end{cases} \tag{6}$$

with $\mathbf{M}_0=\mathbf{0}$.

Proof. See Appendix B.

Accordingly, from Corollary 1, an upper bound for $B(\mathbf{M}_t)$ can be obtained as follows:

$$B(\mathbf{M}_t)\le\frac{1}{N}\sum_{i=1}^{N}w_{i;t-1}\exp(-\alpha_t d_{i;t}) \tag{7}$$

$$=\frac{1}{N}\sum_{i=1}^{N}w_{i;t-1}\exp\left(\frac{1-\frac{d_{i;t}}{\eta_t}}{2}(\eta_t\alpha_t)+\frac{1+\frac{d_{i;t}}{\eta_t}}{2}(-\eta_t\alpha_t)\right) \tag{8}$$

$$\le\frac{1}{N}\sum_{i=1}^{N}w_{i;t-1}\left(\frac{1-\frac{d_{i;t}}{\eta_t}}{2}\exp(\eta_t\alpha_t)+\frac{1+\frac{d_{i;t}}{\eta_t}}{2}\exp(-\eta_t\alpha_t)\right)\triangleq g(\alpha_t), \tag{9}$$

where $\eta_t=\max_{i=1,\ldots,N}|d_{i;t}|$, and inequality (9) is derived from Eq. (8) by using Jensen's inequality [34].

2.2.2. Calculation of α_t

With respect to given $\mathbf{Q}_t$, α_t may be determined in such a way that $g(\alpha_t)$ is minimized. Now, by taking the derivative of $g(\alpha_t)$ to zero and using the fact that $g(\alpha_t)$ is convex, an analytical formula for α_t to minimize $g(\alpha_t)$ can be as follows:

$$\alpha_t=\begin{cases}0, & \frac{1}{2\eta_t}\ln\left(\frac{1+\gamma_t}{1-\gamma_t}\right)<0,\\ \frac{1}{2\eta_t}\ln\left(\frac{1+\gamma_t}{1-\gamma_t}\right) & \text{otherwise,}\end{cases} \tag{10}$$

where

$$\gamma_t=\frac{1}{\eta_t}\sum_{i=1}^{N}w'_{i;t-1}d_{i;t} \tag{11}$$

with sample weight $w'_{i;t-1}$ defined by

$$w'_{i;t-1}=\frac{w_{i;t-1}}{\sum_{i=1}^{N}w_{i;t-1}}. \tag{12}$$

For convenience, these sample weights form a sample weight vector $\mathbf{w}'_{t-1}=[w'_{i;t-1}]$. It should be noticed that the average hypothesis margin with respect to $\mathbf{Q}_t$ with sample weight vector $\mathbf{w}'_{t-1}$ is at least $\eta_t\gamma_t$.

2.2.3. Convergency analysis

When α_t is assigned zero, $\mathbf{Q}_t$ may not be useful for improving the margin, and the boosting iteration should be stopped. Suppose that the boosting iteration is stopped after metric matrix $\mathbf{M}_T$ is yielded. Since $B(\mathbf{M}_t)=(1/N)\sum_{i=1}^{N}w_{i;t}$, $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}_T};\mathcal{X}})$ can be bounded above as follows:

$$\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}_T};\mathcal{X}})\le B(\mathbf{M}_T)\le\prod_{t=1}^{T}\sqrt{1-\gamma_t^2}$$

by substituting the second case of Eq. (10) for α_t in inequality (9) recursively. Accordingly, we have the following theorem.

Theorem 3. *The leave-one-out error $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}_T};\mathcal{X}})$ is at most $\prod_{t=1}^{T}\sqrt{1-\gamma_t^2}$.*

2.3. A metric matrix base-learner specific to the boosting framework

From Theorem 3, we can know that γ_t should be positive and $1-\gamma_t^2$ should be small in order to reduce the training loss. In other words, $\mathbf{Q}_t$ can be found through maximizing γ_t defined by Eq. (11). Since being a positive semi-definite matrix, $\mathbf{Q}_t$ can be decomposed as $\mathbf{Q}_t=\mathbf{L}_t\mathbf{L}_t^T$. Additionally, regarding η_t of Eq. (11) as a constant

for simplification, we can have

$$\gamma_t \propto \mathrm{tr}(\mathbf{L}_t^T \mathbf{X}(\mathfrak{L}_{\mathbf{A}_t} - \mathfrak{L}_{\mathbf{B}_t})\mathbf{X}^T \mathbf{L}_t),$$

where $\mathrm{tr}(.)$ is the trace of a matrix, $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_N]$ is the sample matrix, and $\mathfrak{L}_{\mathbf{A}_t}$ and $\mathfrak{L}_{\mathbf{B}_t}$ are the Laplacian matrices for weight matrices $\mathbf{A}_t$, $\mathbf{B}_t$. The Laplacian matrix $\mathfrak{L}_{\mathbf{P}}$ for matrix $\mathbf{P} = [p_{ij}]$ is defined as $\mathfrak{L}_{\mathbf{P}} = \mathbf{D}_{\mathbf{P}} - \mathbf{P}$, where $\mathbf{D}_{\mathbf{P}} = [d_{\mathbf{P};ij}]$ is a diagonal matrix with $d_{\mathbf{P};ii}$ the sum of the elements of the ith row of $\mathbf{P}$. Weight matrix $\mathbf{A}_t = [a_{ij;t}]$, which describes the nearest miss relationships among samples defined by the $\mathbf{Q}_t$-distance with sample weight vector $\mathbf{w}'_{t-1}$, is defined as follows:

$$a_{ij;t} = R(i,j;\mathbf{w}'_{t-1}, \mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\cdot)), \tag{13}$$

where the function $R(i,j;\mathbf{w}',\mathcal{S}(\cdot))$ is defined as

$$R(i,j;\mathbf{w}',\mathcal{S}(\cdot)) = \begin{cases} w'_i\psi & \text{if } (j,\psi) \in \mathcal{S}(\mathbf{x}_i) \\ 0 & \text{elsewhere} \end{cases} + \begin{cases} w'_j\psi' & \text{if } (i,\psi') \in \mathcal{S}(\mathbf{x}_j), \\ 0 & \text{elsewhere.} \end{cases} \tag{14}$$

Weight matrix $\mathbf{B}_t = [b_{ij;t}]$ is for the nearest hit relationships among samples defined by the $\mathbf{Q}_1$-distance or the $\mathbf{M}_{t-1}$-distance with sample weight vector $\mathbf{w}'_{t-1}$ as follows:

$$b_{ij;t} = \begin{cases} R(i,j;\mathbf{w}'_0, \mathcal{H}_{1;\|\cdot\|_{\mathbf{Q}_1}}(\cdot)) & \text{if } t = 1; \\ R(i,j;\mathbf{w}'_{t-1}, \mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\cdot)) & \text{if } t > 1. \end{cases} \tag{15}$$

Thus, we can obtain $\mathbf{L}_t$ through maximizing the problem P:

$$\mathrm{P}: \arg\max_{\mathbf{L}_t} \quad \mathrm{tr}(\mathbf{L}_t^T \mathbf{X}(\mathfrak{L}_{\mathbf{A}_t} - \mathfrak{L}_{\mathbf{B}_t})\mathbf{X}^T \mathbf{L}_t)$$

$$\text{subjectto} \quad \mathbf{L}_t^T \mathbf{X} \mathfrak{L}_{\mathbf{C}_t} \mathbf{X}^T \mathbf{L}_t = \mathbf{I}_0,$$

where $\mathfrak{L}_{\mathbf{C}_t}$ is the Laplacian matrix for weight matrix $\mathbf{C}_t$, and $\mathbf{I}_0$ is a diagonal matrix with diagonal elements zero or one. Weight matrix $\mathbf{C}_t = [c_{ij;t}]$ is for the nearest hit relationships among samples defined by the $\mathbf{Q}_t$-distance with sample weight vector $\mathbf{w}'_{t-1}$; that is,

$$c_{ij;t} = R(i,j;\mathbf{w}'_{t-1}, \mathcal{H}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\cdot)). \tag{16}$$

The constraint on $\mathbf{L}_t$ has two purposes: (1) to establish a proper metric system induced by $\mathbf{L}_t$; (2) to put bounds on the scale of $\mathbf{L}_t$. It should be noticed that $\mathbf{B}_t$ is equal to $\mathbf{C}_t$ if $t = 1$, and $\mathbf{B}_t$ is independent to $\mathbf{Q}_t$ if $t > 1$.

Directly solving the constrained optimization problem P is not easy because it also includes $\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i)$ and $\mathcal{H}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i)$, which are defined by the $\mathbf{Q}_t$-distance. Here, the framework of the EM algorithm [35] is adopted for calculating $\mathbf{Q}_t$ iteratively as follows.

- **The E-step**: Determine $\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i)$ and $\mathcal{H}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i)$ for every sample $\mathbf{x}_i$ with respect to $\mathbf{Q}_t$ estimated in the previous EM-iteration.
- **The M-step**: Calculate $\mathbf{L}_t$ by Theorem 4, and assign $\mathbf{L}_t\mathbf{L}_t^T$ to $\mathbf{Q}_t$. It should be noticed that $\mathbf{X}\mathfrak{L}_{\mathbf{C}_t}\mathbf{X}^T$ may need regularization because $\mathfrak{L}_{\mathbf{C}_t}$ is only positive semi-definite [36].

At last, the steps for calculating $\mathbf{Q}_t$ are summarized in Algorithm 1, which is referred as to BaseLearner.

Theorem 4. *The solution of the constrained optimization problem P with fixed $\mathfrak{L}_{\mathbf{A}_t}$, $\mathfrak{L}_{\mathbf{B}_t}$, and $\mathfrak{L}_{\mathbf{C}_t}$ can be obtained by*

$$\mathbf{L}_t = \mathbf{\Pi}\mathbf{\Lambda}, \tag{17}$$

where $\mathbf{\Pi}$ is an eigenvector matrix of the matrix pair $(\mathbf{X}(\mathfrak{L}_{\mathbf{A}_t} - \mathfrak{L}_{\mathbf{B}_t})\mathbf{X}^T, \mathbf{X}\mathfrak{L}_{\mathbf{C}_t}\mathbf{X}^T)$ with $\mathbf{\Pi}^T\mathbf{X}\mathfrak{L}_{\mathbf{C}_t}\mathbf{X}^T\mathbf{\Pi} = \mathbf{I}$, and $\mathbf{\Lambda} = [\lambda_{ij}]$ is a diagonal matrix with

$$\lambda_{ii} = \begin{cases} 1 & \textit{if the ith generalized eigenvalue is larger than } 0; \\ 0 & \textit{elsewhere.} \end{cases}$$

Proof. This theorem can be proved by the technique described in [33] for proving a similar theorem. □

Algorithm 1. The metric matrix base-learner.

```
1:  procedure BaseLearner (M_{t-1}, X, C_h, C_m, w'_{t-1})
2:    if t > 1 then
3:      Determine H_{1;||·||_{M_{t-1}}}(x_i) for every sample x_i.
4:      Calculate B_t by Eq. (15), and Ω_{B_t}.
5:    end if
6:    Q_t ← I.
7:    for l ← 1 to maximum number of iterations do
8:      For every sample x_i, determine M_{1;||·||_{Q_t}}(x_i) and
        H_{1;||·||_{Q_t}}(x_i).
9:      Calculate A_t and C_t by Eqs. (13) and (16), and then Ω_{A_t}
        and Ω_{C_t}.
10:     if t = 1 then
11:       Ω_{B_t} ← Ω_{C_t}.
12:     end if
13:     Calculate L_t by Eq. (17).
14:     Q_t ← L_t L_t^T.
15:     if relative difference between successive estimates of
        Q_t < ε then
16:       break.
17:     end if
18:   end for
19:   return Q_t.
20: end procedure
```

2.4. The boosting algorithm

Algorithm 2 presents the proposed boosting algorithm, which is called BoostMDM. Here, BaseLearner is adopted to be the metric matrix base-learner. In order to have a variety of base metric matrices in a few boosting iterations, $\mathbf{Q}_t$ will be found in the null space of $\mathbf{M}_{t-1}$. Without this strategy, $\mathbf{Q}_t$'s learned in successive boosting iterations may be highly linear correlated. Besides, BoostMDM should be stopped when one of the following three termination conditions occurs.

- First, the rank of $\mathbf{Q}_t$ is zero; in other words, BaseLearner cannot find a useful metric matrix.
- Second, γ_t is not greater than zero; that is, integrating $\mathbf{Q}_t$ may not improve the margin.
- Third, $\mathbf{M}_t$ has full rank.

Now, the main steps of BoostMDM are illustrated as follows.

- Repeat the following steps until one of the above three termination conditions occurs.
  - Line 7 calculates the sample weight vector $\mathbf{w}'_{t-1} = [w'_{i;t-1}]$ by Eq. (12).
  - Line 10 applies BaseLearner with sample weight vector $\mathbf{w}'_{t-1}$ and sample matrix $\tilde{\mathbf{X}}_{t-1}$ to yield $\tilde{\mathbf{Q}}_t$, where $\tilde{\mathbf{X}}_{t-1}$ is composed of the projections of the training samples onto the null space of $\mathbf{M}_{t-1}$.
  - Line 15 calculates the metric matrix $\mathbf{Q}_t$ by $\mathbf{Q}_t = \mathbf{E}_{t-1}\tilde{\mathbf{Q}}_t\mathbf{E}_{t-1}^T$, where the columns of $\mathbf{E}_{t-1}$ form an orthonormal basis for the null space of $\mathbf{M}_{t-1}$.

- Line 21 calculates α_t and combines $\mathbf{M}_{t-1}$ with $\mathbf{Q}_t$ to form $\mathbf{M}_t$.
- Lines 22–31 find the null space of $\mathbf{M}_t$, calculate an orthogonal basis for the null space of $\mathbf{M}_t$, and project the sample matrix onto the null space of $\mathbf{M}_t$ for calculating $\mathbf{Q}_{t+1}$.

• The last step decomposes $\mathbf{M}_T$ into $\mathbf{L}_T\mathbf{L}_T^T$ such that the columns of $\mathbf{L}_T$ are ordered by discriminatory potential.

The rationale of the last step is explained as follows. With $\mathbf{M}_T=\mathbf{\Psi\Theta\Psi}^T$ an eigen-decomposition of $\mathbf{M}_T$ and $\mathbf{\Phi}$ an orthogonal matrix, $\mathbf{L}_T=\mathbf{\Psi\Theta}^{1/2}\mathbf{\Phi}$ is a general form for $\mathbf{L}_T$ such that $\mathbf{M}_T=\mathbf{L}_T\mathbf{L}_T^T$. To order the discriminatory potential of the column of $\mathbf{L}_T$, we may define weight matrices $\mathbf{A}=[a_{ij}]$ and $\mathbf{C}=[c_{ij}]$ by

$$a_{ij}=R\left(i,j;\frac{1}{N}\mathbf{1}_{1\times N},\mathcal{M}_{1;\|\cdot\|_{\mathbf{M}_T}}(\cdot)\right) \tag{18}$$

and

$$c_{ij}=R\left(i,j;\frac{1}{N}\mathbf{1}_{1\times N},\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_T}}(\cdot)\right) \tag{19}$$

to describe the nearest miss and nearest hit relationships among the training samples defined by the $\mathbf{M}_T$-distance. Thus, the discriminatory potential of $\mathbf{x}$ may be gauged by $\mathbf{x}^T\mathbf{X}(\mathfrak{L}_\mathbf{A}-\mathfrak{L}_\mathbf{C})\mathbf{X}^T\mathbf{x}$, which is the difference between the average squared distance to the nearest miss and that to the nearest hit when the training sample is projected onto $\mathbf{x}$. Since the ith diagonal element of $\mathbf{L}_T^T\mathbf{X}(\mathfrak{L}_\mathbf{A}-\mathfrak{L}_\mathbf{C})\mathbf{X}^T\mathbf{L}_T$ is the discriminatory potential of the ith column of $\mathbf{L}_T$, $\mathbf{\Phi}$ can be determined by the eigenvector matrix of $\mathbf{\Theta}^{1/2}\mathbf{\Psi}^T\mathbf{X}(\mathfrak{L}_\mathbf{A}-\mathfrak{L}_\mathbf{C})\mathbf{X}^T\mathbf{\Psi\Theta}^{1/2}$ with corresponding eigenvalues in non-ascending order.

Algorithm 2. The boosting algorithm for learning a Mahalanobis distance metric.

1: **procedure** BoostMDM ($\mathbf{X}$, $\mathcal{C}$) ▷ $\mathbf{X}$ is the sample matrix.
2: $\mathbf{M}_0\leftarrow\mathbf{0}_{n\times n}$. ▷ n is the dimensionality of the sample vector.
3: $\mathbf{E}_0\leftarrow\mathbf{I}_{n\times n}$. ▷ The range of $\mathbf{E}_0$ is the null space of $\mathbf{M}_0$ initially.
4: $\tilde{\mathbf{X}}_0\leftarrow\mathbf{X}$.
5: $t\leftarrow 1$.
6: **loop**
7: Calculate sample weight vector $\mathbf{w}'_{t-1}$ by Eq. (12).
8: ▷ $\tilde{\mathbf{X}}_{t-1}$ is composed of the sample projected
9: ▷ onto the null space of $\mathbf{M}_{t-1}$.
10: Calculate $\tilde{\mathbf{Q}}_t$ by BaseLearner with $\mathbf{M}_{t-1}$, $\tilde{\mathbf{X}}_{t-1}$ and $\mathbf{w}'_{t-1}$.
11: **if** the rank of $\tilde{\mathbf{Q}}_t$ is zero **then**
12: $T\leftarrow t-1$.
13: **break**.
14: **end if**
15: $\mathbf{Q}_t\leftarrow\mathbf{E}_{t-1}\tilde{\mathbf{Q}}_t\mathbf{E}_{t-1}^T$. ▷ Form the metric matrix $\mathbf{Q}_t$.
16: Calculate γ_t by Eq. (11).
17: **if** $\gamma_t\le 0$ **then**
18: $T\leftarrow t-1$.
19: **break**.
20: **end if**
21: Calculate α_t by Eq. (10), and $\mathbf{M}_t$ by Eq. (3).
22: $\mathbf{N}_t\leftarrow\mathtt{null}(\tilde{\mathbf{Q}}_t)$. ▷ The columns of $\mathbf{N}_t$ form an orthonormal
23: ▷ basis for the null space of $\tilde{\mathbf{Q}}_t$.
24: **if** the rank of $\mathbf{N}_t=0$ **then**
25: $T\leftarrow t$.
26: **break**.
27: **end if**
28: $\mathbf{E}_t\leftarrow\mathbf{E}_{t-1}\mathbf{N}_t$. ▷ The columns of $\mathbf{E}_t$ form an orthonormal
29: ▷ basis for the null space of $\mathbf{M}_t$.
30: $\tilde{\mathbf{X}}_t\leftarrow\mathbf{N}_t^T\tilde{\mathbf{X}}_{t-1}$. ▷ Now, $\tilde{\mathbf{X}}_t$ contains the projections of the
31: ▷ samples onto the null space of $\mathbf{M}_t$.
32: $t\leftarrow t+1$.
33: **end loop**
34: Calculate $\mathbf{A}$ and $\mathbf{C}$ by Eqs. (18) and (19), and then $\mathfrak{L}_\mathbf{A}$ and $\mathfrak{L}_\mathbf{C}$.
35: $\mathbf{L}_T\leftarrow\mathbf{\Psi\Theta}^{1/2}\mathbf{\Phi}$, where $\mathbf{M}_T=\mathbf{\Psi\Theta\Psi}^T$ is the eigen-decomposition of $\mathbf{M}_T$, and $\mathbf{\Phi}$ is the eigenvector matrix of $\mathbf{\Theta}^{1/2}\mathbf{\Psi}^T\mathbf{X}(\mathfrak{L}_\mathbf{A}-\mathfrak{L}_\mathbf{C})\mathbf{X}^T\mathbf{\Psi\Theta}^{1/2}$ with eigenvalues in non-ascending order.
36: **return** $\mathbf{L}_T$.
37: **end procedure**

2.5. A bound on the generalization error

For simplicity, we consider the two-class classification problem; that is, the sample is in $\mathcal{Z}=\mathbb{R}^n\times\{-1,1\}$. Suppose that the

training set is the prototype set. Thus, the classifier to be analyzed may be expressed as

$$h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x}) = y_{\mathbf{p}} \times (\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2 - \|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2), \tag{20}$$

where $\mathbf{p}$ is the prototype having the shortest $\mathbf{M}$-distance to $\mathbf{x}$, $y_{\mathbf{p}}$ is the class label of $\mathbf{p}$, and $\mathbf{p}'$ is the prototype of class $-y_{\mathbf{p}}$ nearest to $\mathbf{x}$. In other words, the sign and magnitude of $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})$ are the predicted class label for $\mathbf{x}$, and the prediction confidence, respectively. Thus, the generalization error of $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})$ can be defined as

$$er_{\mathcal{D}}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) = \mathbb{E}_{(\mathbf{x},y)\in\mathcal{Z}}\mathcal{L}_0(y \times h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})).$$

Denoting by $\mathcal{X}^{\setminus i}$ the sample set $\mathcal{X}$ excluding $(\mathbf{x}_i,y_i)$, we may express the empirical leave-one-out error $\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$ as

$$\hat{er}_{loo}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_0(y_i \times h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^{\setminus i}}(\mathbf{x}_i)).$$

Furthermore, we also define the following θ-clipped error estimates

$$er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) = \mathbb{E}_{(\mathbf{x},y)\in\mathcal{Z}}\mathcal{L}_{\theta}(y \times h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x}));$$

$$\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{\theta}(y_i \times h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^{\setminus i}}(\mathbf{x}_i)).$$

To show a relationship between $er_{\mathcal{D}}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$ and $\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$, two assumptions are made.

- First, the spectral norm of the metric matrix is normalized to one without lost of generality.
- Second, the distribution $\mathcal{D}$ is supported by a ball of radius r.

Table 1
Descriptions of the data set and the settings for estimating the classification rate, where n, c, and N denote the dimensionality of the data, the number of classes, and the number of samples, respectively.

Data set	n	c	N	Estimation	Training	Test	Runs
Spectf	44	2	267	Holdout	201	66	50
WDBC	30	2	569	Holdout	427	142	50
Parkinsons	22	2	195	Holdout	174	48	50
Yeast	8	10	1484	Holdout	1117	367	50
Wine	13	3	178	Holdout	135	43	50
Balance	4	3	625	Holdout	469	156	50
Ionosphere	34	2	351	Holdout	264	87	50
Waveform	21	3	800	Holdout	600	200	50
Crings	3	5	500	Holdout	500	165	50
USPS	256	10	9298	Holdout	2328	6970	1
Spambase	57	2	4601	Holdout	1151	3450	1
GCM	16 063	14	190	Leave-one-out	189	1	190
COLON	2000	2	62	Leave-one-out	61	1	62
Prostate	12 000	2	102	Leave-one-out	101	1	102
AR	5120	126	1638	Holdout	1386	252	10
GT	5120	50	750	Holdout	600	150	50
YALE	5120	38	2414	Holdout	605	1809	10

In this study, we adopt a technique for analyzing the generalization error of a graphical learning algorithm of which some part is stable [37]. Here, the stable part is $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$ with given $\|\cdot\|_{\mathbf{M}}$. Define a function set $\mathcal{P}(n,l) = \{\|\mathbf{x}\|_{\mathbf{M}}^2 = \sum_{i=1}^{l}(\mathbf{u}_i^T\mathbf{x})^2\}$ for the squared $\mathbf{M}$-distance where $\mathbf{x}\in\mathbb{R}^n$ and $\mathbf{M} = \mathbf{L}\mathbf{L}^T$ with $\mathbf{L} = [\mathbf{u}_1 \dots \mathbf{u}_l]$. A theorem about an upper bound of the generalization error of $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$, which is in terms of the covering number [38] of $\mathcal{P}(n,l)$ and the classification stability [39] of the stable part, will be provided. First, the covering number of a function set, and the classification stability for a real-valued learning algorithm are defined as follows.

Definition 1. Let $\mathcal{F} = \{f(\mathbf{x}:\mathbf{u})\,|\,\mathbf{x}\in\mathbb{R}^n\}$ be a set of real-valued functions parameterized by $\mathbf{u}$. Denote by $\mathcal{V}$ a set of vectors in $\mathbb{R}^N$. Given a set of N observations $\mathcal{S} = \{\mathbf{x}_1,\dots,\mathbf{x}_N\}$, the covering number for $\mathcal{F}$ with respect to these N observations, denoted as $\mathcal{N}(\mathcal{F},\varepsilon,\mathcal{S})$, is the cardinality of the smallest $\mathcal{V}$ such that for every $\boldsymbol{\alpha}$, there exists $\mathbf{v}\in\mathcal{V}$ satisfying $\|[f(\mathbf{x}_1|\mathbf{u}),\dots,f(\mathbf{x}_N|\mathbf{u})]^T - \mathbf{v}\|_2 \le \sqrt{N}\varepsilon$. Define $\mathcal{N}(\mathcal{F},\varepsilon,N)$ as $\sup_{\mathcal{S}}\mathcal{N}(\mathcal{F},\varepsilon,\mathcal{S})$.

Definition 2. A real-valued learning algorithm which maps the training set $\mathcal{X}$ to a classifier $h_{\mathcal{X}}(\mathbf{x})$ has classification stability β if the following relationship is satisfied

$$\forall\mathcal{X}\in\mathcal{Z}^N, \forall i\in\{1,\dots,N\}, |h_{\mathcal{X}}(\mathbf{x}) - h_{\mathcal{X}^{\setminus i}}(\mathbf{x})| \le \beta,$$

where $h_{\mathcal{X}}(\mathbf{x})$ predicts the label of an arbitrary instance $\mathbf{x}$ by the sign of $h_{\mathcal{X}}(\mathbf{x})$.

The main result is as follows.

Theorem 5. *Let $\mathcal{D}$ be an unknown distribution over $\mathcal{Z}$ which is supported on a ball of radius r, and $\mathcal{X}\in\mathcal{Z}^N$ be a training set in which every element is drawn i.i.d. from $\mathcal{D}$. Suppose that $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$ with any $\mathbf{M}$-distance has classification stability β^*, where the spectral norm of $\mathbf{M}$ is one. Then, for any $\theta>0$ and $\varepsilon>0$, we have*

$$er_{\mathcal{D}}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) \le \hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) + \frac{\beta^*}{\theta} + 2\left(\frac{4N\beta^*}{\theta}+1\right) \times\sqrt{\frac{1}{2N}\left(\ln\mathcal{N}\left(\mathcal{P}(n,l),\frac{\theta\varepsilon}{16r^2},N\right)+\ln(1/\delta)\right)}$$

with probability at least $1-\delta$. In addition, if $\delta\le e^{-1}$, we also have

$$er_{\mathcal{D}}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) \le \hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) + \frac{\beta^*}{\theta} + \frac{2(4N\beta^*/\theta+1)}{\sqrt{2N}} \times\sqrt{\frac{\sqrt{2N}\left(l\left\lceil\frac{64lr^4}{\theta}\right\rceil\ln(2n+1)\right)}{4N\beta^*/\theta+1}+\ln\frac{1}{\delta}}$$

with probability at least $1-\delta$.

Proof. See Appendix C.

According to Theorem 5, we may have two knacks for learning an effective metric matrix based on the proposed approach.

Table 2
The important parameters for the proposed approach.

Algorithm	Parameter	Value	Description
BoostMDM-K	k_h	5	
	k_m	5	
BoostMDM-G	μ	$\frac{N}{\sum_{i=1}^{N}\|\mathbf{x}_i-\mathbf{x}_{i;7}\|_{\mathbf{M}}^2}$	$\mathbf{x}_{i;7}$ denotes the seventh nearest neighbor of sample vector $\mathbf{x}_i$ defined by the $\mathbf{M}$-distance.
BaseLearner	Maximum number of iterations	10	
	λ	0.1	Regularization parameter for COLON
		0.01	Regularization parameter for Prostate
		0.005	Regularization parameter for GCM
		10^{-8}	Regularization parameter for GT, YALE, and AR

- The first is improving the stability of the stable part. Two extensions based on this idea will be introduced in the next section.
- The second is minimizing the leave-one-out empirical error while keeping the rank of the metric matrix as small as possible, which can be achieved by means of the regularization technique.

2.6. Two extensions of the loss function

The nearest hit or miss of a sample defined by the original space may not be a good estimate of that defined in the mapped space. An average of the k nearest hits or misses may be more stable and better. Based on this idea, two possible extensions of $h_{\|\cdot\|_{\mathbf{M}},\mathcal{X}}$ are discussed below.

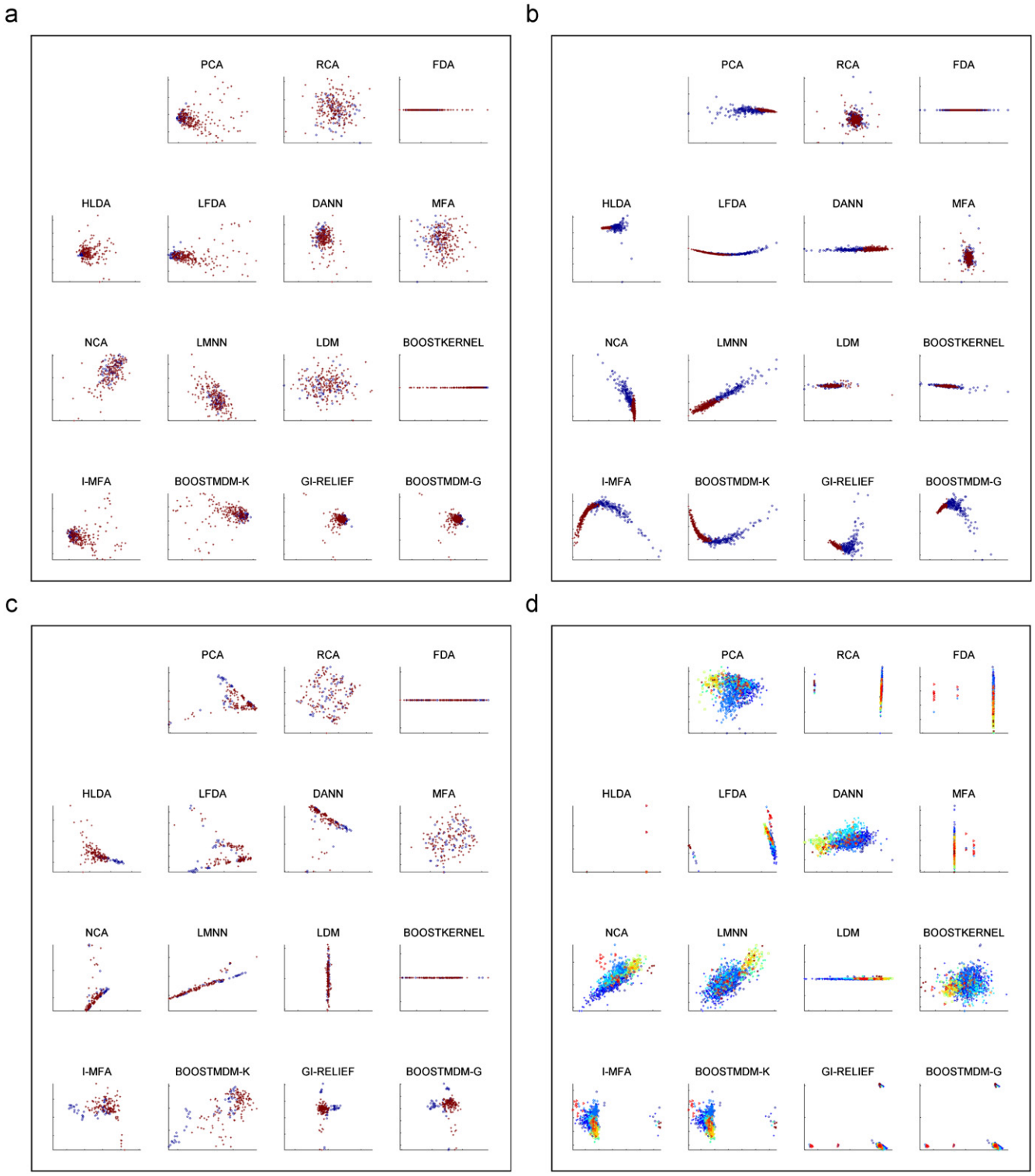

Fig. 1. 2-D visualization of lower-dimensional and small-scale data sets, and a subset of the first five classes of USPS. (a) Spectf, (b) WDBC, (c) Parkinsons, (d) Yeast, (e) Wine, (f) Balance, (g) Ionosphere, (h) Waveform, (i) Crings and (j) USPS.

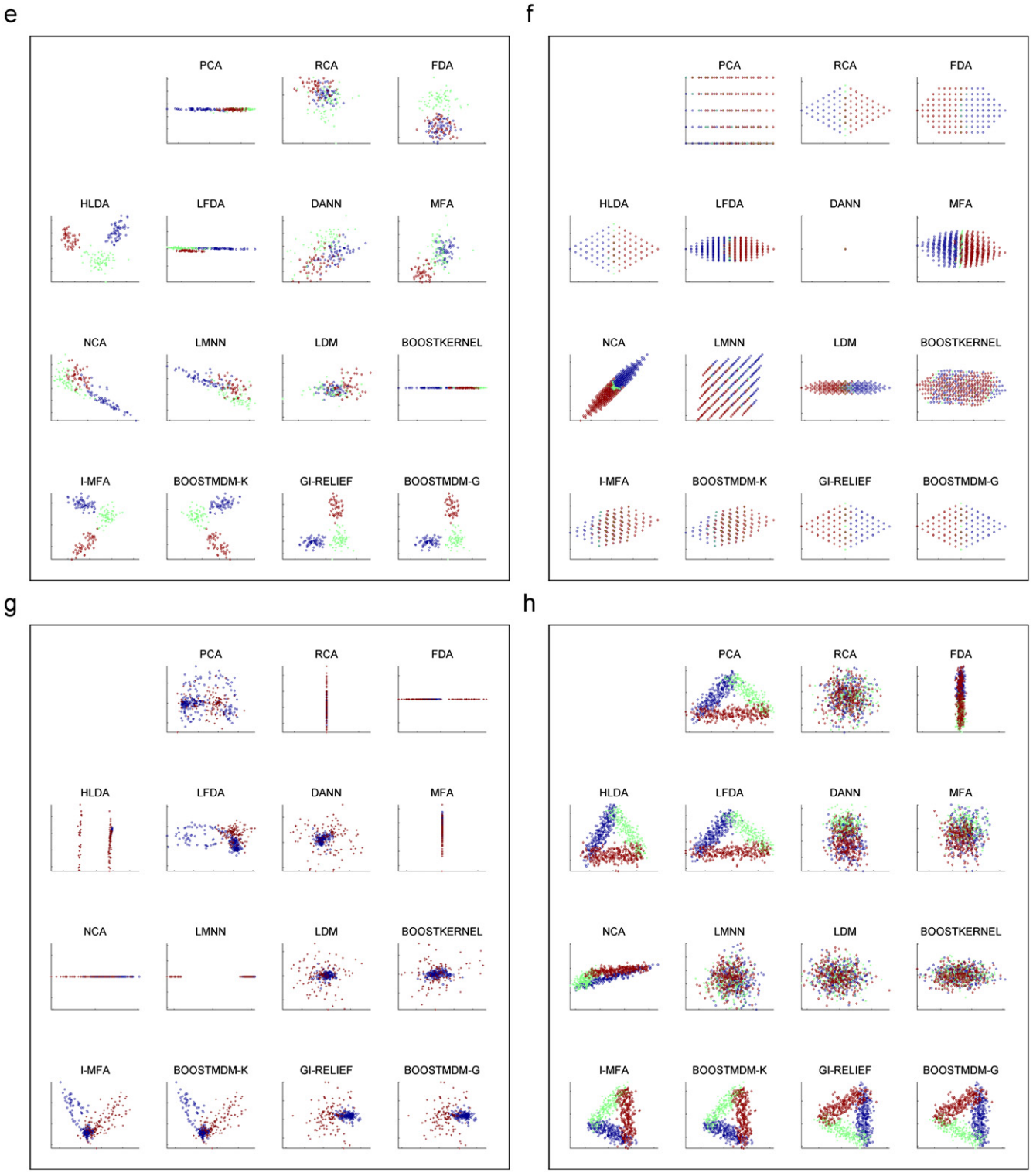

Fig. 1. (*continued*)

2.6.1. Averages of the k-nearest hits and misses

A new loss function may be defined in terms of the averages of the k_h nearest hits and k_m nearest misses as follows:

$$\hat{er}_{loo}(h^{k_h,k_m}_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_0(f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i)) > f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i)))$$

$$\leq \frac{1}{N}\sum_{i=1}^{N} \exp(f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i)) - f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i))) \triangleq B_{k_h,k_m}(\mathbf{M}).$$

Since Theorem 1 and Corollary 2 also hold for this extension, the boosting framework for $B_{k_h,k_m}(\mathbf{M})$ has a convergency property like Theorem 3, and BaseLearner and BoostMDM can be applied for this extension as well by changing $\mathcal{H}_{1;\bullet}(\mathbf{x})$ and $\mathcal{M}_{1;\bullet}(\mathbf{x})$ into $\mathcal{H}_{k_h;\bullet}(\mathbf{x})$ and $\mathcal{M}_{k_m;\bullet}(\mathbf{x})$, respectively. For convenience, this extension is called BoostMDM-K. In order not to mislead BoostMDM-K, we have found that k_m could be small but k_h could not. This is because once h_h is small, a large margin between k_m nearest misses and k_h nearest hits may result from inadequate sampling of nearest hits.

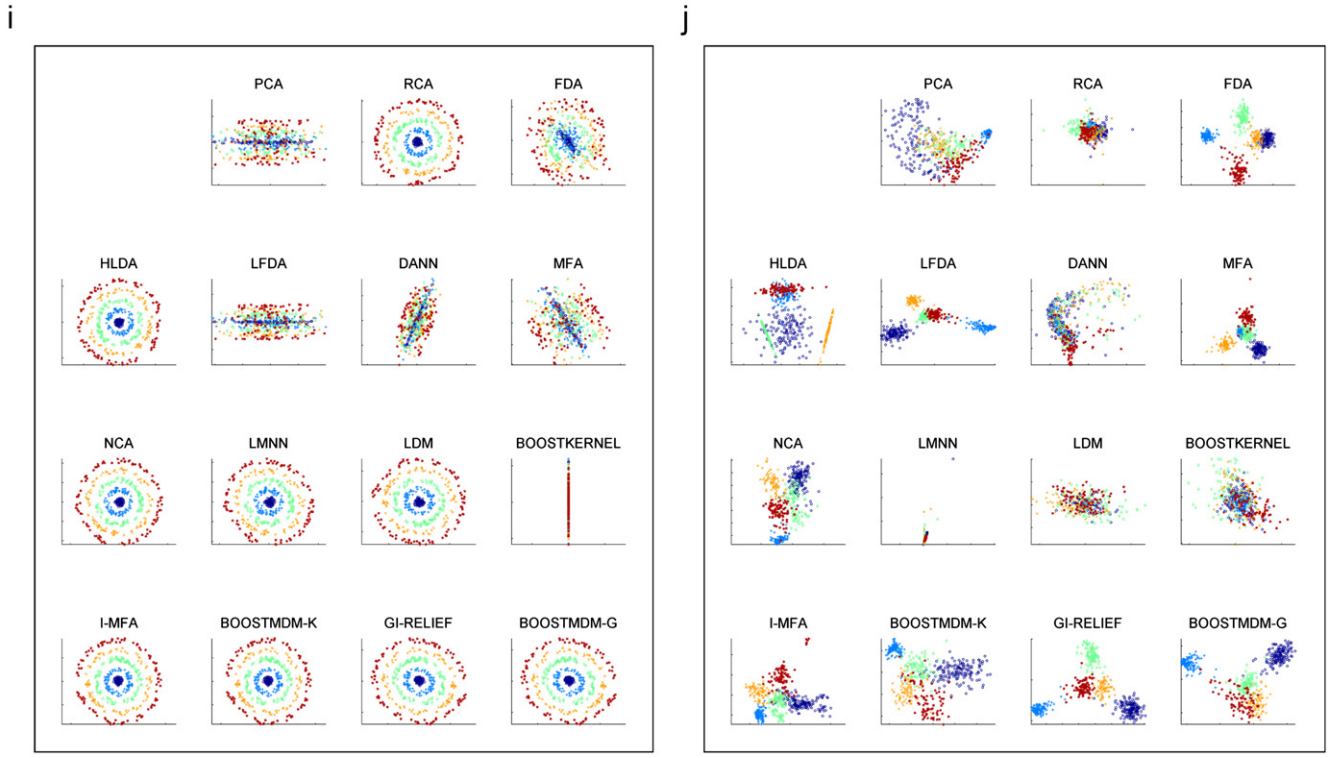

Fig. 1. (*continued*)

2.6.2. Gaussian-weighted averages of the hits and misses

Based on the stochastic nearest neighbor model described in [18], the expected squared distances from a sample $\mathbf{x}$ to its nearest hit and miss with respect to the $\mathbf{M}$-distance may be calculated by

$$\sum_{\mathbf{y}\in\mathcal{C}_h(\mathbf{x})}\frac{\exp(-\mu\|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2)}{\sum_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\exp(-\mu\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2)}\|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2$$

and

$$\sum_{\mathbf{y}\in\mathcal{C}_m(\mathbf{x})}\frac{\exp(-\mu\|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2)}{\sum_{\mathbf{z}\in\mathcal{C}_m(\mathbf{x})}\exp(-\mu\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2)}\|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2,$$

respectively, where $\mu\geq 0$. Accordingly, the second extension is defined based on the stochastic nearest neighbor model as follows:

$$\begin{aligned}\hat{er}_{loo}(h^{\mathcal{G}}_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}) &= \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_0(f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{H}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i))>f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{M}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i)))\\ &\leq \frac{1}{N}\sum_{i=1}^{N}\exp(f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{H}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i))-f_{\mathbf{M}}(\mathbf{x}_i,\mathcal{M}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}_i)))\triangleq B_{\mathcal{G}}(\mathbf{M}),\end{aligned}$$

where

$$\mathcal{H}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x})=\left\{(i,\psi)\,|\,\psi=\frac{\exp(-\mu\|\mathbf{x}_i-\mathbf{x}\|_{\mathbf{M}}^2)}{\sum_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\exp(-\mu\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2)},\mathbf{x}_i\in\mathcal{C}_h(\mathbf{x})\right\},$$

$$\mathcal{M}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x})=\left\{(i,\psi)\,|\,\psi=\frac{\exp(-\mu\|\mathbf{x}_i-\mathbf{x}\|_{\mathbf{M}}^2)}{\sum_{\mathbf{z}\in\mathcal{C}_m(\mathbf{x})}\exp(-\mu\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2)},\mathbf{x}_i\in\mathcal{C}_m(\mathbf{x})\right\}.$$

By changing $\mathcal{H}_{1;\bullet}(\mathbf{x})$ and $\mathcal{M}_{1;\bullet}(\mathbf{x})$ into $\mathcal{H}_{\mathcal{G};\bullet}(\mathbf{x})$ and $\mathcal{M}_{\mathcal{G};\bullet}(\mathbf{x})$, respectively, BaseLearner and BoostMDM can also be applied for this extension. This extension is referred as to BoostMDM-G. However, BoostMDM-G does not have a convergency property as clear as that for BoostMDM-K because this property is dependent on μ as the illustration in the following.

When μ is equal to zero, we have

$$\mathcal{H}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x})=\mathcal{H}_{\mathrm{card}(\mathcal{C}_h(\mathbf{x}));\|\cdot\|_{\mathbf{M}}}(\mathbf{x}),$$

$$\mathcal{M}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x})=\mathcal{M}_{\mathrm{card}(\mathcal{C}_m(\mathbf{x}));\|\cdot\|_{\mathbf{M}}}(\mathbf{x});$$

that is, $f_{\mathbf{M}}(\mathbf{x},\mathcal{H}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}))$ and $f_{\mathbf{M}}(\mathbf{x},\mathcal{M}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}))$ actually calculate the average squared $\mathbf{M}$-distances from $\mathbf{x}$ to the hits and misses of $\mathbf{x}$, respectively. It can be checked that Theorem 1 is satisfied here. Thus, we can conclude that the boosting framework for $B_{\mathcal{G}}(\mathbf{M})$ has a convergency property like Theorem 3 when μ is equal to zero.

On the other hand, if the distances from a sample to the others are all distinct, and μ approaches infinity, we have

$$\lim_{\mu\to\infty}\frac{\exp(-\mu\|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2)}{\sum_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\exp(-\mu\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2)}=\begin{cases}0 & \text{if } \|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2>\min\limits_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2\\ 1 & \text{if } \|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2=\min\limits_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2\end{cases}$$

for every $\mathbf{y}\in\mathcal{C}_h(\mathbf{x})$, and can obtain

$$\begin{aligned}&\lim_{\mu\to\infty}f_{\mathbf{M}}(\mathbf{x},\mathcal{H}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}))\\ &=\sum_{\mathbf{y}\in\mathcal{C}_h(\mathbf{x})}\|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2\times\begin{cases}0 & \text{if } \|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2>\min\limits_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2\\ 1 & \text{if } \|\mathbf{y}-\mathbf{x}\|_{\mathbf{M}}^2=\min\limits_{\mathbf{z}\in\mathcal{C}_h(\mathbf{x})}\|\mathbf{z}-\mathbf{x}\|_{\mathbf{M}}^2\end{cases}\\ &=f_{\mathbf{M}}(\mathbf{x},\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}}}(\mathbf{x})).\end{aligned}$$

Similarly, we also have

$$\lim_{\mu\to\infty}f_{\mathbf{M}}(\mathbf{x},\mathcal{M}_{\mathcal{G};\|\cdot\|_{\mathbf{M}}}(\mathbf{x}))=f_{\mathbf{M}}(\mathbf{x},\mathcal{M}_{1;\|\cdot\|_{\mathbf{M}}}(\mathbf{x})).$$

Accordingly, it can be verified that Theorem 1 is also true, and the boosting framework for $B_{\mathcal{G}}(\mathbf{M})$ also has a convergency property like Theorem 3 when μ approaches infinity. In summary, the boosting framework for $B_{\mathcal{G}}(\mathbf{M})$ can converge if a proper value for μ is chosen.

3. Discussions

3.1. Comparisons with related non-boosting approaches

Since more robust than BoostMDM, BoostMDM-K and BoostMDM-G are discussed here. First, the mathematical formulation of the objective function of BaseLearner is under the framework of graph embedding although the objective function of

BaseLearner for calculating the first metric matrix $\mathbf{M}_1$ is different from that for calculating the others. In fact, if BoostMDM-K and BoostMDM-G have only one boosting iteration, they are an iterative version of MFA [5], referred as to I-MFA, and the generalized iterative RELIEF (GI-RELIEF) [33], respectively. Here, I-MFA and GI-RELIEF are regarded as the non-boosting counterparts of BoostMDM-K and BoostMDM-G, respectively. The following are some weaknesses of I-MFA and GI-RELIEF.

- First, although GI-RELIEF may converge to a fixed point if a proper value for μ is selected, every iteration of GI-RELIEF does not always get an improvement in the value of the objective function [33].
- Second, the convergency property of I-MFA seems not to be better than that of GI-RELIEF.
- Third, GI-RELIEF and I-MFA consider that the null space of $\mathbf{M}_1$ has no useful information, which may not be true.

On the other hand, after obtaining $\mathbf{M}_1$, BoostMDM-K and BoostMDM-G try to improve $\mathbf{M}_1$ by integrating $\mathbf{M}_1$ with other metric matrices, which are orthogonal to $\mathbf{M}_1$ and to each other. Thus, BoostMDM-K and BoostMDM-G have the following two different properties.

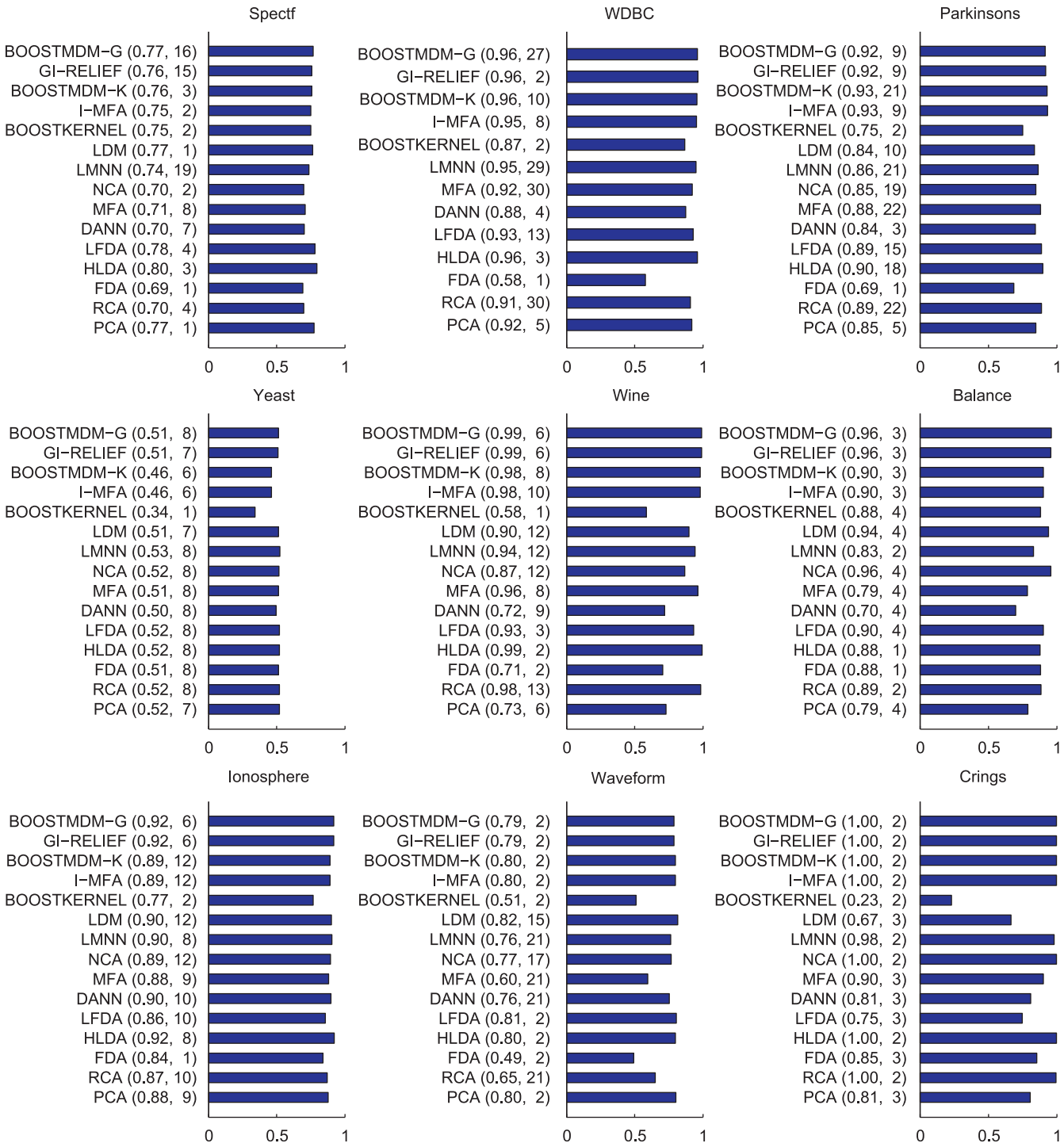

Fig. 2. Classification rates for the small-scale data set, where the numbers enclosed in the parentheses are the best average classification rate and the associated dimensionality of the mapped vector, respectively.

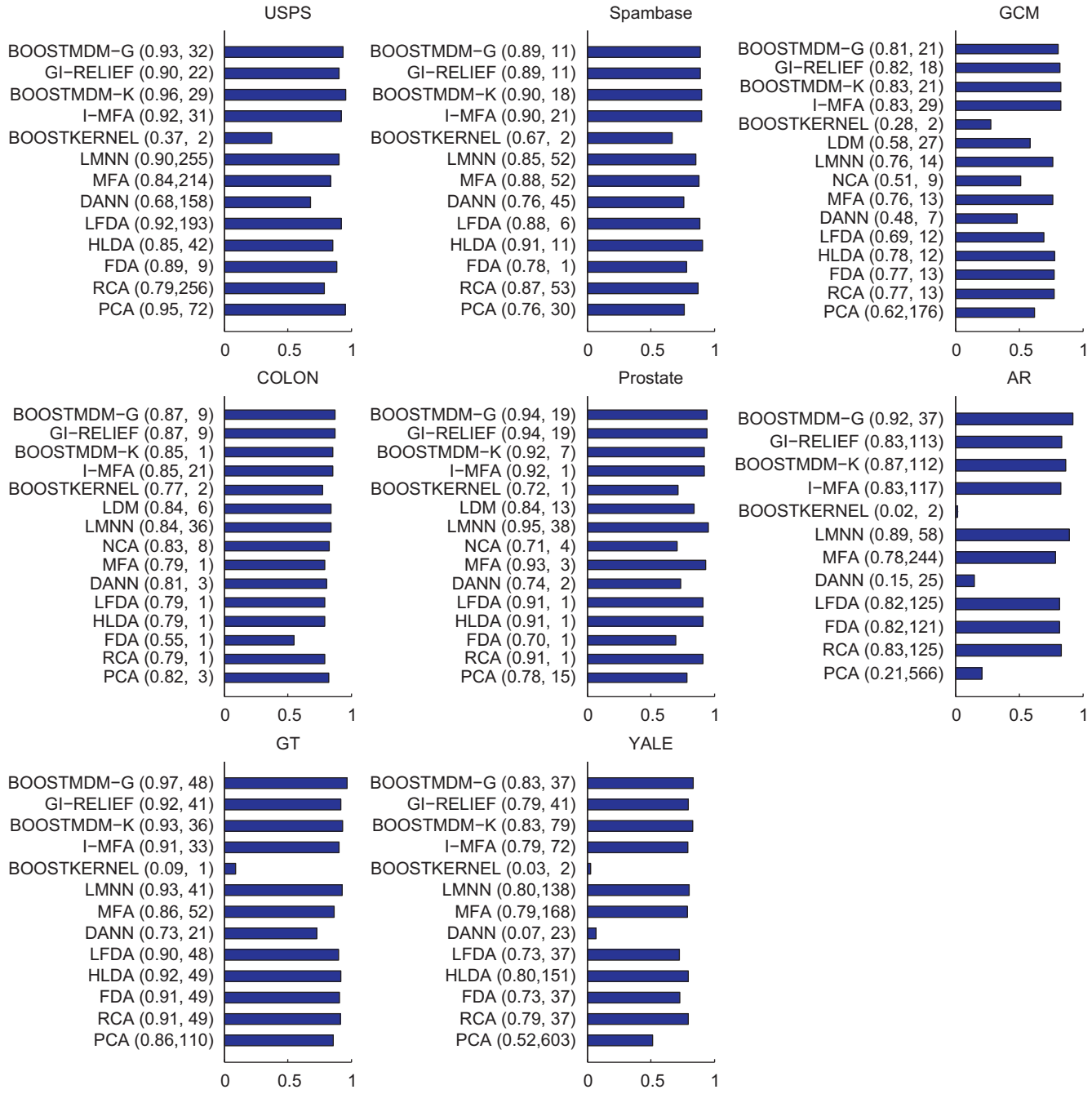

Fig. 3. Classification rates for the larger-scale data set and the higher-dimensional data set, where the numbers enclosed in the parentheses are the best average classification rate and the associated dimensionality of the mapped vector, respectively.

- First, BoostMDM-K and BoostMDM-G may converge in terms of an upper bound of the leave-one-out training error.
- Second, BoostMDM-K and BoostMDM-G make use of the information in the null space of $\mathbf{M}_1$.

3.2. Undersampled problems

When the sample vectors are higher-dimensional data, the number of training samples may be less than the dimensionality of the sample vector (i.e., $N<n$). In this circumstance, the principal component analysis (PCA) is first applied to reduce the dimensionality of the sample vector from n to $N-1$. It should be emphasized that no principal components are abandoned at this step. Denote by $\mathbf{\Omega}$ the unit eigenvector matrix for the PCA, which is an $n\times(N-1)$ matrix. Then, BoostMDM works on the reduced sample matrix, $\mathbf{\Omega}^T\mathbf{X}$. Finally, the transformation matrix corresponding to the resultant metric matrix can be formed by $\mathbf{\Omega}\mathbf{L}_T$. It should be noticed that calculating $\mathbf{L}_t$ by Theorem 4 at the M-step of BaseLearner needs proper regularization. Otherwise, the overfitting problem may occur. Here, for the undersampled problem, $\mathbf{L}_t$ is calculated by Theorem 4 with the matrix pair $(\mathbf{X}(\Omega_{\mathbf{A}_t}-\Omega_{\mathbf{B}_t})\mathbf{X}^T, \mathbf{X}\Omega_{\mathbf{C}_t}\mathbf{X}^T+\lambda\sigma_1(\mathbf{X}\Omega_{\mathbf{C}_t}\mathbf{X}^T)\mathbf{I})$, where $\lambda\sigma_1(\mathbf{X}\Omega_{\mathbf{C}_t}\mathbf{X}^T)\mathbf{I}$ is a regularization term with $\sigma_1(\mathbf{X}\Omega_{\mathbf{C}_t}\mathbf{X}^T)$ the largest singular value of $\mathbf{X}\Omega_{\mathbf{C}_t}\mathbf{X}^T$ and λ an empirically selected parameter.

4. Experimental results

Table 1 lists 17 data sets for performance evaluation. These data sets include

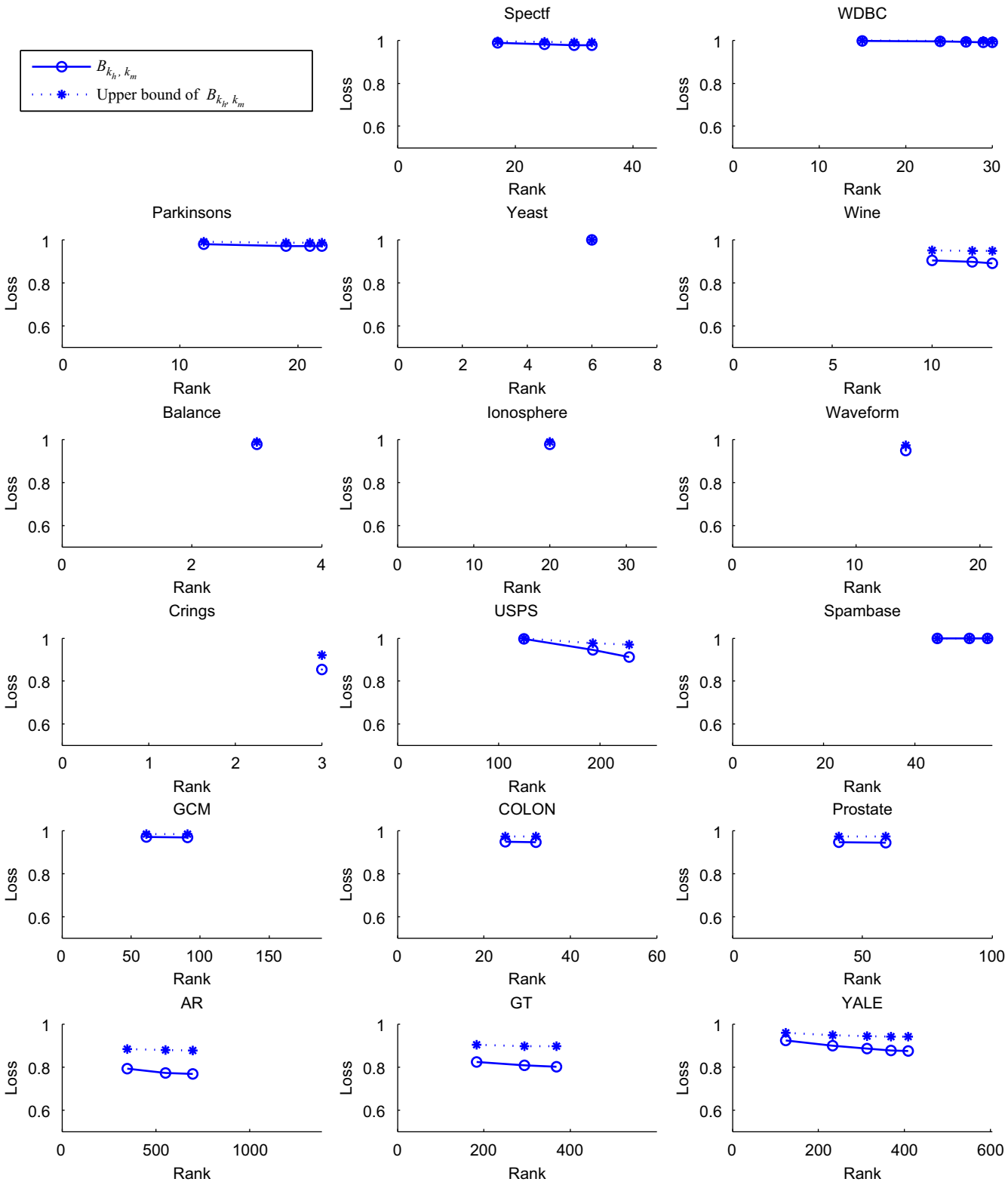

Fig. 4. Plots of $B_{k_h,k_m}(\mathbf{M}_t)$ with respect to the rank of $\mathbf{M}_t$ for BoostMDM-K.

- eight data sets from UCI machine learning repository [40]: Spectf, Wisconsin Diagnostic Breast Cancer (WDBC), Parkinsons, Yeast, Wine, Balance, Ionosphere, and Spambase;
- one data set of handwritten digit images: USPS[1];
- three gene expression data sets: COLON [41], Prostate [42] and GCM [43];
- three face data sets: GT [44], YALE [45], and AR [46];
- two synthetical data sets: Crings and Waveform.

Spectf, Breast, Parkinsons, Yeast, Wine, Balance, Ionosphere, Waveform, and Crings are lower-dimensional small-scale data sets. USPS

[1] http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/data.html.

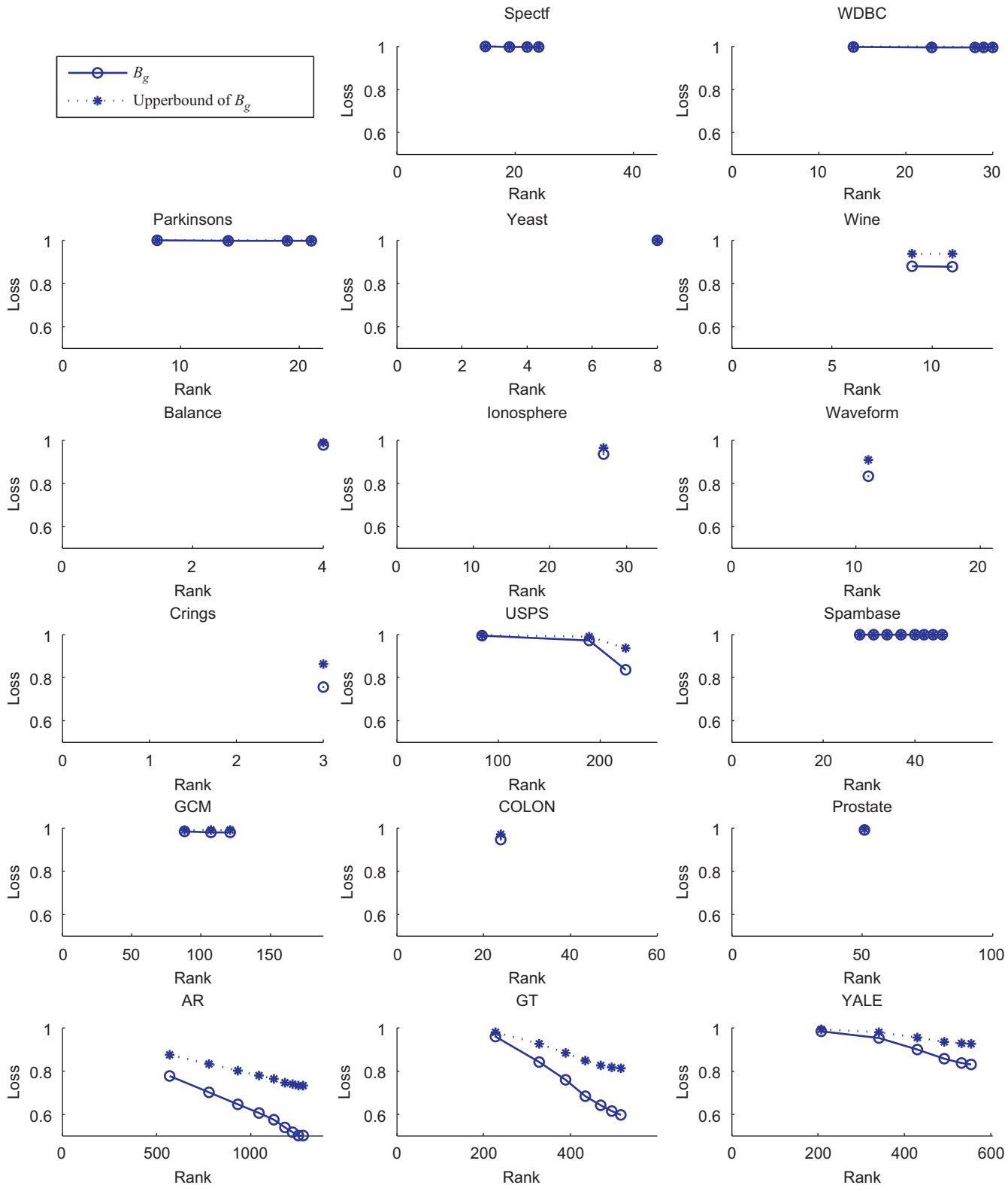

Fig. 5. Plots of $B_g(\mathbf{M}_t)$ with respect to the rank of $\mathbf{M}_t$ for BoostMDM-G.

and Spambase are larger-scale data sets. COLON, Prostate, GCM, GT, YALE, and AR are higher-dimensional data sets. The face data set consists of 64×80 grayscale face images, which are manually aligned with respect to the two eyes. The specification of the synthetical data set Waveform can be found in [47]. The synthetical data set Crings consists of three-dimensional sample vectors of five classes. The five classes have equal prior probabilities, and are distributed over five co-centric ring-shaped regions in the first two dimensions. Each ring-shaped region contains a single class. The third dimensions of the sample vectors are Gaussian noises. The magnitude of the Gaussian noise is not small so that the main principal components of the samples cover the third dimension.

The two extensions of the proposed approach, namely, BoostMDM-K and BoostMDM-G, were implemented, and

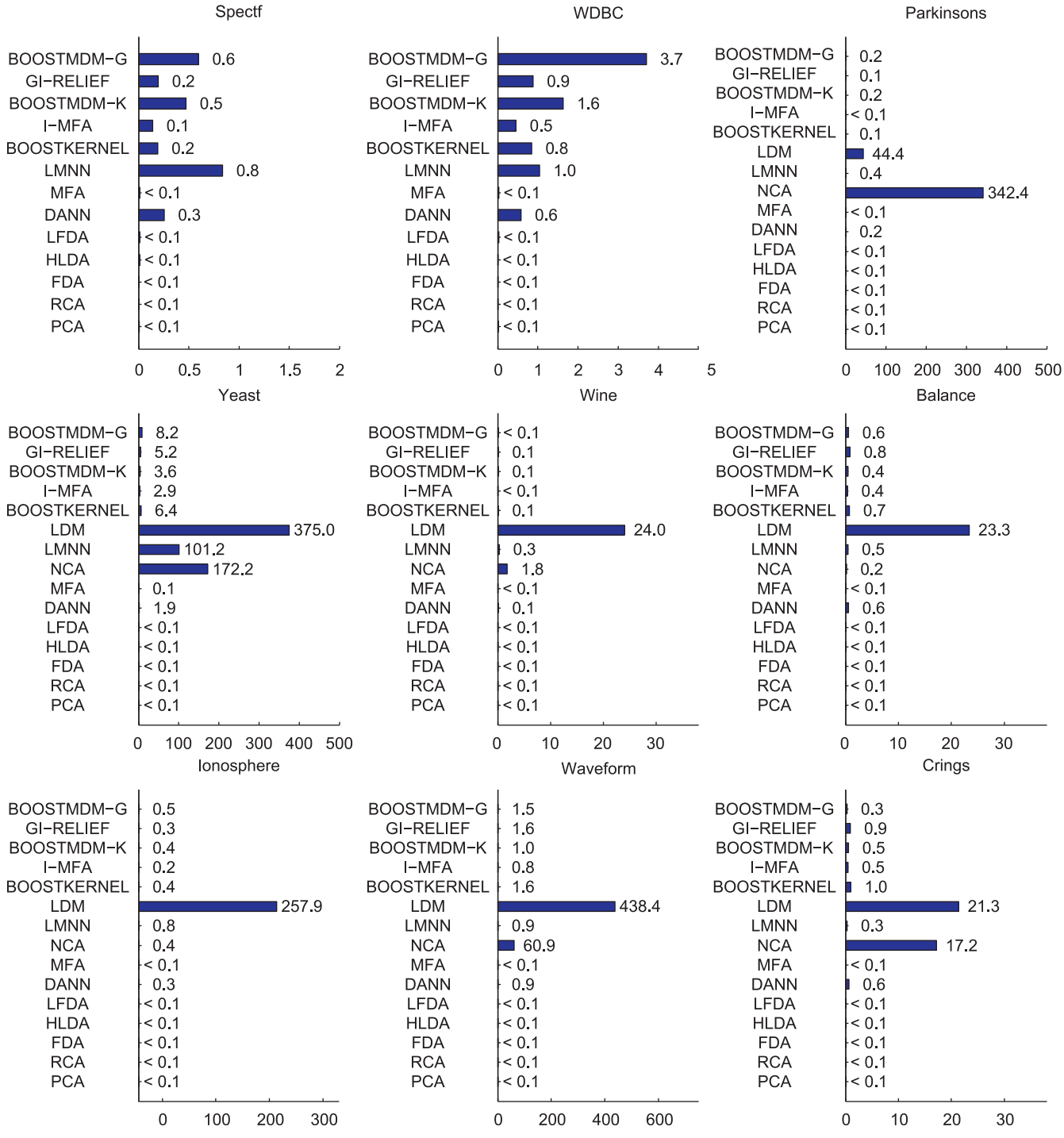

Fig. 6. Average computation time for the small-scale data set (in seconds).

important parameters for them are listed in Table 2. Thirteen algorithms for Mahalanobis metric learning were also implemented for performance comparison, which include PCA, BoostKernel with the exponential loss [29], RCA [3], DANN [9], FDA [48], HLDA [4], LDM [17],[2] LFDA [6],[3] LMNN [14],[4] NCA [18],[5] MFA [11,5], I-MFA, and GI-RELIEF [33]. It should be noticed that I-MFA and GI-RELIEF were implemented by BoostMDM-K and BoostMDM-G with only one boosting iteration, respectively.

When higher-dimensional data sets were encountered, all methods to be compared worked on the data of which dimensionality is reduced by the PCA as the step described in Section 3.2. The nearest neighbor classifier was used to evaluate the accuracy of the learned Mahalanobis distance metric. The holdout method or the leave-one-out method [49] was used to estimate the classification rate. The choice between these two methods is dependent on the scale of the data set. Table 1 also shows the experiment parameter for each data set. For each method to be tested, the components of the mapped vector were sorted by the feature importance in non-ascending

[2] Software available at http://www.cs.cmu.edu/~liuy/ldm_scripts_2.zip.
[3] http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LFDA/index.html.
[4] http://www.weinbergerweb.net/Downloads/LMNN.html.
[5] http://www.cs.berkeley.edu/~fowlkes/software/nca/.

order. The dimensionality of the mapped vector was augmented from one to the theoretical limit of the method with increments of one, and the associated classification rate was recorded over all runs of the experiment.

All algorithms were implemented in the MATLAB programming language. The experiments were conducted on a computer which has two Intel® Xeron 2.0 GHz CPUs and two gigabytes of RAM, and runs the Windows Server 2003 operating system.

4.1. Data visualization

Fig. 1 plots the mapped vector projected onto the first two dimensions for the lower-dimensional data sets, and a subset of the first five classes of USPS. It can be seen that I-MFA, BoostMDM-K, GI-RELIEF, and BoostMDM-G can capture the overall class structures for these data sets.

4.2. Nearest neighbor classification

Due to the high computational cost, NCA and LDM were not tested against WDBC, USPS, Spambase, and the three face data sets, and HLDA was not tested against AR either. Figs. 2 and 3 show the classification rate, where the numbers enclosed in the parentheses are the best average classification rate and the associated dimensionality of the mapped vector, which is defined as *the most effective dimensionality*, respectively. The experimental results are summarized as follows.

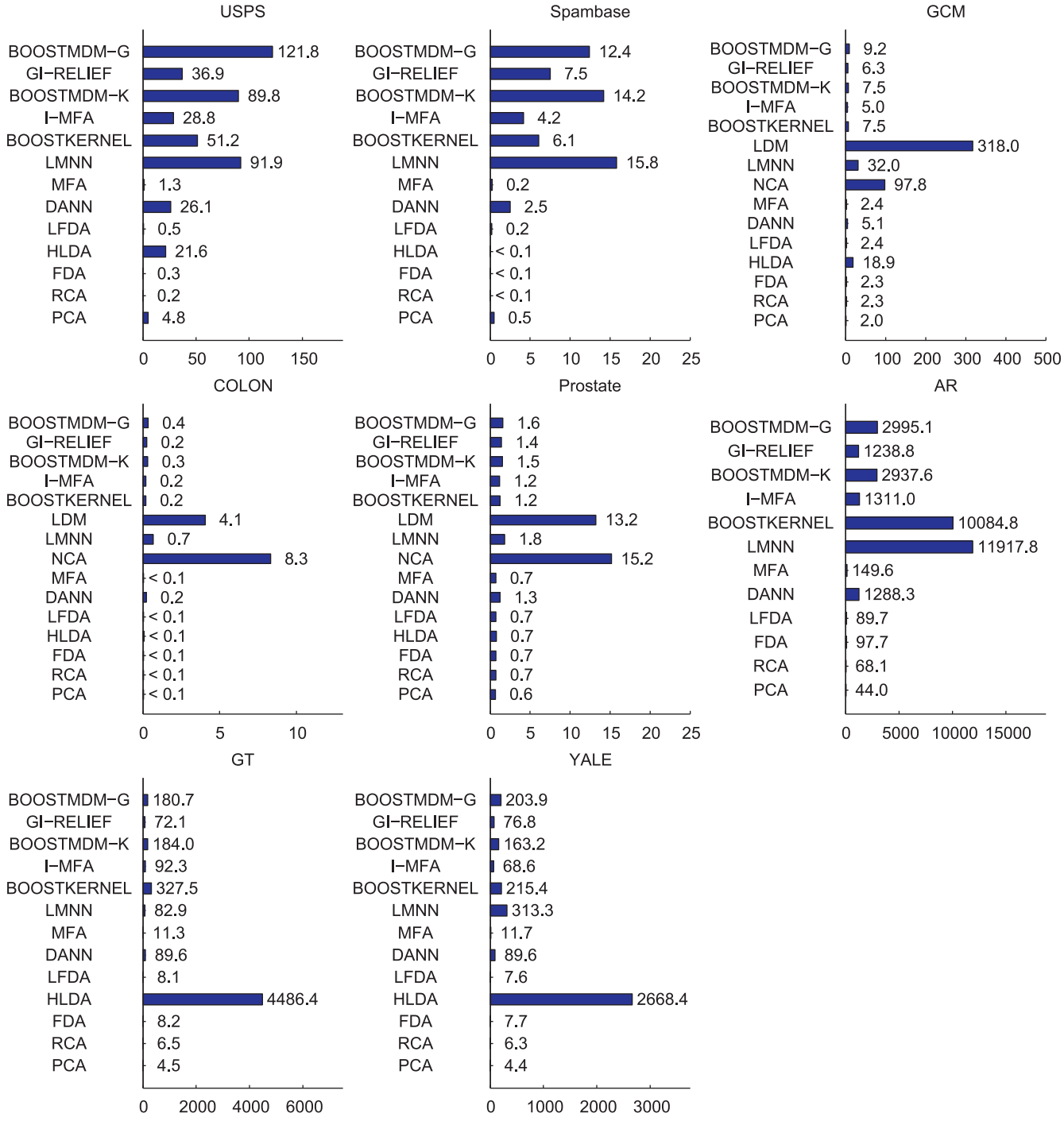

Fig. 7. Average computation time for the larger-scale data set and the higher-dimensional data set (in seconds).

- BoostMDM-G is the best or the second best on twelve data sets in terms of classification accuracy. BoostMDM-K is the best or the second best on ten data sets. Overall, among the 15 methods to be compared, BoostMDM-G is the most accurate.
- Although the rank of the metric matrix learned by the proposed boosting approach is at least as high as that learned by the non-boosting counterpart, the proposed boosting approach may have smaller values of the most effective dimensionality than the non-boosting counterpart has. This phenomenon can be observed from BoostMDM-K on data sets Wine, USPS, Spambase, GCM, COLON, AR, and YALE, and BoostMDM-G on AR and YALE. We think that this phenomenon may be owing to the last step of BoostMDM, which decomposes the learned metric matrix into the transformation matrix such that the column of the transformation matrix is ordered by discriminatory potential.
- BoostMDM-K improves I-MFA on data sets Spectf, WDBC, USPS, GCM, COLON, AR, GT, and YALE. BoostMDM-G enhances GI-RELIEF on data sets Spectf, USPS, AR, GT, and YALE but impairs GI-RELIEF on GCM a little.[6] Notice that Spectf, WDBC, and USPS are not undersampled data sets. These results can be illustrated by Figs. 4 and 5 as follows.
  - First, since some data sets have only one complete boosting iteration, BoostMDM-K and BoostMDM-G on these data sets have the same classification result as their non-boosting counterparts.
  - Second, BoostMDM-K and BoostMDM-G can improve I-MFA and GI-RELIEF if they can reduce the loss significantly after the first boosting iteration.
  - Third, since the metric matrix learned by I-MFA and GI-RELIEF for the lower-dimensional data set often has nearly full rank, BoostMDM-K and BoostMDM-G improve I-MFA and GI-RELIEF less significantly on this kind of data set.
  - Fourth, as BoostMDM-G on GCM shown, the overfitting problem may occur. Our experience shows that proper regularization for calculating the metric matrix is needed especially for the undersampled data set.
- BoostMDM-G improves GI-RELIEF less significant than BoostMDM-K does for I-MFA. Two possible reasons are (1) I-MFA is less accurate than GI-RELIEF; (2) the convergency property of BoostMDM-K is better than that of BoostMDM-G.
- I-MFA is more accurate than MFA on all test data sets except Yeast and Prostate. Thus, if the target neighbor of a sample is unknown in advance, I-MFA should be preferred more than MFA.
- From the experimental results of BoostKernel, it can be seen that the alignment loss is not always suitable for learning a Mahalanobis distance metric for the NN classification.

Overall, BoostMDM-K and BoostMDM-G are effective in learning metric matrices for the 17 test data sets. These results justify the feasibility of the proposed boosting algorithm to learn a metric matrix for the NN classification. Figs. 6 and 7 also show that BoostMDM-K and BoostMDM-G have acceptable computation time. However, we found that the two parameters k_h and k_m for BoostMDM-K are critical on the two synthetical data sets, namely, Waveform and Crings. Thus, BoostMDM-G is suggested in terms of accuracy and stability.

5. Conclusion

In this paper, a boosting algorithm (BoostMDM) for supervised learning of Mahalanobis distance metrics has been proposed. This algorithm learns a metric matrix through minimizing a hypothesis margin-based loss function. In addition, a metric matrix base-learner (BaseLearner) specific to BoostMDM has also been developed. Based on the theoretical framework of BoostMDM, two more robust extensions, namely, BoostMDM-K and BoostMDM-G, have been introduced. We have also indicated that the metric matrix base-learners for BoostMDM-K and BoostMDM-G are closely related to two non-boosting approaches: I-MFA and GI-RELIEF, respectively. The experimental results show that BoostMDM-K and BoostMDM-G can yield Mahalanobis distance metrics effective in the nearest neighbor classification for a variety of data sets. Since making use of the whole sample space and having a better convergency property in terms of the leave-one-out training error, BoostMDM-K and BoostMDM-G may improve the accuracy of I-MFA and GI-RELIEF. However, the proposed boosting algorithm lacks a mechanism to deal with noisy data, which is worthy of further investigation.

Appendix A. Proof for Theorem 1

According to the definitions of $\mathcal{S}$ and $\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})$, $f_{\mathbf{M}_t}(\mathbf{x},\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}))$ calculates the average squared $\mathbf{M}_t$-distance from $\mathbf{x}$ to its k_h nearest hits which are defined by the $\mathbf{M}_t$-distance, whereas $f_{\mathbf{M}_t}(\mathbf{x},\mathcal{S}(\mathbf{x}))$ calculates the average squared $\mathbf{M}_t$-distance from $\mathbf{x}$ to at least k_h hits. Thus, we have $f_{\mathbf{M}_t}(\mathbf{x},\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) \le f_{\mathbf{M}_t}(\mathbf{x},\mathcal{S}(\mathbf{x}))$. In addition, due to $f_{\mathbf{M}_t}(\mathbf{x},\mathcal{S}(\mathbf{x})) = f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{S}(\mathbf{x})) + \alpha f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{S}(\mathbf{x}))$, we can obtain

$$f_{\mathbf{M}_t}(\mathbf{x},\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) \le f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{S}(\mathbf{x})) + \alpha f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{S}(\mathbf{x})). \tag{A.1}$$

Furthermore, we can decompose $f_{\mathbf{M}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}))$ as

$$f_{\mathbf{M}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) = f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) + \alpha f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}))$$

and obtain

$$\begin{aligned} &f_{\mathbf{M}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) \\ &\quad \ge f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x})) + \alpha f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x})), \end{aligned} \tag{A.2}$$

by inequalities (A.3) and (A.4) as follows:

$$f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) \ge f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x})), \tag{A.3}$$

$$f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) \ge f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x})). \tag{A.4}$$

Inequalities (A.3) and (A.4) are established based on the fact that $\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x})$ and $\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x})$ are composed of the k_m nearest misses of $\mathbf{x}$ defined in terms of metric matrices $\mathbf{M}_{t-1}$ and $\mathbf{Q}_t$, respectively. Subtracting the left-hand side of inequality (A.2) from that of inequality (A.1), we can obtain

$$\begin{aligned} &f_{\mathbf{M}_t}(\mathbf{x},\mathcal{H}_{k_h;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) - f_{\mathbf{M}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x})) \le f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{S}(\mathbf{x})) + \alpha f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{S}(\mathbf{x})) \\ &\quad - (f_{\mathbf{M}_{t-1}}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x})) + \alpha f_{\mathbf{Q}_t}(\mathbf{x},\mathcal{M}_{k_m;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}))) \end{aligned} \tag{A.5}$$

and complete this proof by rearranging the right-hand side of inequality (A.5).

Appendix B. A proof for Corollary 2

In case $t > 1$, we may express $\delta_{i;t}$ as

$$\begin{aligned} \delta_{i,t} &= f_{\mathbf{M}_t}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}_i)) - f_{\mathbf{M}_t}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{M}_t}}(\mathbf{x}_i)) \\ &\le f_{\mathbf{M}_{t-1}}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)) - f_{\mathbf{M}_{t-1}}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)) \\ &\quad + \alpha_t(f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)) - f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i))) \\ &= \delta_{i;t-1} + \alpha_t(f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)) - f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i))) \end{aligned} \tag{B.1}$$

by Theorem 1 with $\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)$ selected for $\mathcal{S}(\mathbf{x}_i)$. Considering the exponential functions of the both sides of inequality (B.1), we have

$$\begin{aligned} \exp(\delta_{i;t}) &\le \exp(\delta_{i;t-1}) \times \exp(\alpha_t(f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{H}_{1;\|\cdot\|_{\mathbf{M}_{t-1}}}(\mathbf{x}_i)) \\ &\quad - f_{\mathbf{Q}_t}(\mathbf{x}_i,\mathcal{M}_{1;\|\cdot\|_{\mathbf{Q}_t}}(\mathbf{x}_i)))) = w_{i;t-1}\exp(-\alpha_t d_{i;t}). \end{aligned}$$

[6] Since the regularization parameter λ for the higher-dimensional data set was selected more properly, GI-RELIEF implemented here is more accurate than that reported in [33].

In case $t=1$, we may select $\mathcal{H}_{1;\|\cdot\|_{\mathbf{Q}_1}}(\mathbf{x}_i)$ for $\mathcal{S}(\mathbf{x}_i)$ of Theorem 1, and obtain $\exp(\delta_{i;1})=w_{i;0}\exp(-\alpha_1 d_{i;1})$ because of $\mathbf{M}_0=\mathbf{0}$. Accordingly, we have proven this corollary.

Appendix C. A proof for Theorem 5

First, the classification stability of the stable part, $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$ with given $\|\cdot\|_{\mathbf{M}}$, is analyzed. It turns out that the classification stability of the stable part is bounded above by $4r^2$ due to Lemma 6 and Corollary 7. Since the bound shown in Corollary 7 is independent to $\|\cdot\|_{\mathbf{M}}$, we may select a number β^* from the range between 0 and $4r^2$ for the classification stability of $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$ with respect to any $\mathbf{M}$-distance. Thus, $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$ with given $\mathbf{M}$-distance has uniform stability β^*/θ with respect to the loss function $\mathcal{L}_\theta$ [39].

Lemma 6. *Suppose that $\mathcal{X}$ consists of N training samples. For any $i\in\{1,\ldots,N\}$, we have*

$$\begin{aligned}&\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|\\&\quad\le\max\{\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2,\|\mathbf{x}-\mathbf{q}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2\},\end{aligned}$$

where $\mathbf{p}$ and $\mathbf{p}'$ are the nearest and the second nearest prototype of class $+1$ to $\mathbf{x}$ defined by the $\mathbf{M}$-distance, respectively, and $\mathbf{q}$ and $\mathbf{q}'$ are similarly defined for the prototype of class -1.

Proof. Denote by $(\mathbf{p},y)\in\mathcal{X}$ the training sample which has the shortest $\mathbf{M}$-distance to $\mathbf{x}$. Among all training samples of class y, let $\mathbf{p}'$ denote the one which has the second shortest $\mathbf{M}$-distance to $\mathbf{x}$. Let $\mathbf{q}$ and $\mathbf{q}'$ be the analogies of $\mathbf{p}$ and $\mathbf{p}'$ for class $-y$. Let $\mathbf{r}$ be the sample in $\mathcal{X}^i$ nearest to $\mathbf{x}$. Then, this lemma can be proved by considering the following four cases.

Case 1. If neither $\mathbf{p}$ nor $\mathbf{q}$ is the ith sample in $\mathcal{X}$, $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})$ is equal to $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})$; that is, $\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|=0$.

Case 2. If $\mathbf{q}$ is the ith sample in $\mathcal{X}$, $\mathbf{r}$ must be $\mathbf{p}$ because the sample in $\mathcal{X}$ nearest to $\mathbf{x}$ is not removed. Thus, we have

$$\begin{aligned}&\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|=\left|y(\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2)\right.\\&\quad\left.-y(\|\mathbf{x}-\mathbf{q}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2)\right|=\|\mathbf{x}-\mathbf{q}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2.\end{aligned}$$

Case 3. If $\mathbf{p}$ is the ith sample, we have to consider the following two cases.

Case 3.1. If the class label of $\mathbf{r}$ is y, we have that $\mathbf{r}$ is $\mathbf{p}'$ and obtain

$$\begin{aligned}&\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|=\left|y(\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2)\right.\\&\quad\left.-y(\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2)\right|=\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2.\end{aligned}$$

Case 3.2. If the label of $\mathbf{r}$ is $-y$, $\mathbf{r}$ must be $\mathbf{q}$. Thus, we have

$$\begin{aligned}&\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|=\left|y(\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2)\right.\\&\quad\left.-(-y)(\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2)\right|=\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2.\end{aligned}$$

Accordingly, we can conclude for any $i\in\{1,\ldots,N\}$,

$$\begin{aligned}&\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|\\&\quad\le\max\{\|\mathbf{x}-\mathbf{p}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{p}\|_{\mathbf{M}}^2,\|\mathbf{x}-\mathbf{q}'\|_{\mathbf{M}}^2-\|\mathbf{x}-\mathbf{q}\|_{\mathbf{M}}^2\}\end{aligned}$$

and have completed the proof for this lemma. □

Corollary 7. *The classification stability of $h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}$ with given $\|\cdot\|_{\mathbf{M}}$ is bounded above by $4r^2$.*

Proof. Combining Lemma 6 with the two assumptions made beforehand: the metric matrix $\mathbf{M}$ has the spectral norm of one, and the distribution $\mathcal{D}$ is supported by a ball of radius r, we have that $\left|h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}^i}(\mathbf{x})\right|$ is not greater than $4r^2$. □

The next step is to estimate the covering number of $\mathcal{P}(n,l)$. Lemma 8 shows that $\mathcal{L}_\theta$ is Lipschitz continuous with respect to $\|\cdot\|_{\mathbf{M}}$. Corollary 10 presents a covering number of $\mathcal{P}(n,l)$ based on Theorem 9 [38], which is about the covering number for a set of linear functions.

Lemma 8. *Suppose that $\mathcal{D}$ is supported by a ball of radius r. Then, for any $(\mathbf{x},y)\in\mathcal{Z}$, we have*

$$\begin{aligned}&\left|\mathcal{L}_\theta(y\times h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x}))-\mathcal{L}_\theta(y\times h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x}))\right|\\&\quad\le\frac{8r^2}{\theta}\sup_{\|\mathbf{y}\|_2=1}\left|\|\mathbf{y}\|_{\mathbf{M}_1}^2-\|\mathbf{y}\|_{\mathbf{M}_2}^2\right|=\frac{8r^2}{\theta}\|\mathbf{M}_1-\mathbf{M}_2\|_s,\end{aligned}$$

where $\|\cdot\|_s$ denotes the spectral norm.

Proof. Since $\mathcal{L}_\theta$ is θ^{-1}-Lipschitz, we have

$$\begin{aligned}&\left|\mathcal{L}_\theta(y\times h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x}))-\mathcal{L}_\theta(y\times h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x}))\right|\\&\quad\le\frac{1}{\theta}\left|h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x}))-h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x}))\right|.\end{aligned}$$

Denote by $\mathbf{p}_i$ the prototype nearest to $\mathbf{x}$ with respect to the $\mathbf{M}_i$-distance, and $y_{\mathbf{p}_i}$ the label of $\mathbf{p}_i$. Denote by $\mathbf{p}_i'$ the prototype of class $-y_{\mathbf{p}_i}$ nearest to $\mathbf{x}$ with respect to the $\mathbf{M}_i$-distance. Thus, we have

$$\begin{aligned}&\frac{1}{\theta}\left|h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x}))-h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x}))\right|=\frac{1}{\theta}\left|\|y_{\mathbf{p}_1}(\|\mathbf{x}-\mathbf{p}_1'\|_{\mathbf{M}_1}^2-\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_1}^2)\right.\\&\quad\left.-y_{\mathbf{p}_2}(\|\mathbf{x}-\mathbf{p}_2'\|_{\mathbf{M}_2}^2-\|\mathbf{x}-\mathbf{p}_2\|_{\mathbf{M}_2}^2)\right|.\end{aligned}$$

Next, this lemma may be proven by considering the relationship between $y_{\mathbf{p}_1}$ and $y_{\mathbf{p}_2}$.

- In case $y_{\mathbf{p}_1}$ is equal to $y_{\mathbf{p}_2}$, we may assume $h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x})$ is not less than $h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x})$ without lost of generality. Thus, we have

$$\begin{aligned}&\frac{1}{\theta}\left|h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x}))-h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x}))\right|\\&\quad=\frac{1}{\theta}((\|\mathbf{x}-\mathbf{p}_1'\|_{\mathbf{M}_1}^2-\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_1}^2)-(\|\mathbf{x}-\mathbf{p}_2'\|_{\mathbf{M}_2}^2-\|\mathbf{x}-\mathbf{p}_2\|_{\mathbf{M}_2}^2))\\&\quad\le\frac{1}{\theta}((\|\mathbf{x}-\mathbf{p}_2'\|_{\mathbf{M}_1}^2-\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_1}^2)-(\|\mathbf{x}-\mathbf{p}_2'\|_{\mathbf{M}_2}^2-\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_2}^2))\\&\quad\le\frac{1}{\theta}\sup_{\|\mathbf{y}\|_2=1}\left|\|\mathbf{y}\|_{\mathbf{M}_1}^2-\|\mathbf{y}\|_{\mathbf{M}_2}^2\right|(\|\mathbf{x}-\mathbf{p}_1\|_2^2+\|\mathbf{x}-\mathbf{p}_2'\|_2^2)\\&\quad\le\frac{8r^2}{\theta}\sup_{\|\mathbf{y}\|_2=1}\left|\|\mathbf{y}\|_{\mathbf{M}_1}^2-\|\mathbf{y}\|_{\mathbf{M}_2}^2\right|\\&\quad=\frac{8r^2}{\theta}\|\mathbf{M}_1-\mathbf{M}_2\|_s,\end{aligned}$$

  because $\mathcal{D}$ is supported by a ball of radius r.
- In case $y_{\mathbf{p}_1}$ is not equal to $y_{\mathbf{p}_2}$, we similarly have

$$\begin{aligned}&\frac{1}{\theta}\left|h_{\|\cdot\|_{\mathbf{M}_1};\mathcal{X}}(\mathbf{x})-h_{\|\cdot\|_{\mathbf{M}_2};\mathcal{X}}(\mathbf{x})\right|\\&\quad=\frac{1}{\theta}((\|\mathbf{x}-\mathbf{p}_1'\|_{\mathbf{M}_1}^2-\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_1}^2)+(\|\mathbf{x}-\mathbf{p}_2'\|_{\mathbf{M}_2}^2-\|\mathbf{x}-\mathbf{p}_2\|_{\mathbf{M}_2}^2))\\&\quad\le\frac{1}{\theta}((\|\mathbf{x}-\mathbf{p}_2\|_{\mathbf{M}_1}^2-\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_1}^2)+(\|\mathbf{x}-\mathbf{p}_1\|_{\mathbf{M}_2}^2-\|\mathbf{x}-\mathbf{p}_2\|_{\mathbf{M}_2}^2))\\&\quad\le\frac{8r^2}{\theta}\sup_{\|\mathbf{y}\|_2=1}\left|\|\mathbf{y}\|_{\mathbf{M}_1}^2-\|\mathbf{y}\|_{\mathbf{M}_2}^2\right|.\\&\quad=\frac{8r^2}{\theta}\|\mathbf{M}_1-\mathbf{M}_2\|_s.\quad\square\end{aligned}$$

Theorem 9 (*Zhang [38]*). *Define $\mathcal{F}=\{\mathbf{u}^T\mathbf{x}:\|\mathbf{u}\|_2\le 1,\mathbf{x}\in\mathbb{R}^n,\|\mathbf{x}\|_2\le 2r\}$. Then, we have $\log_2\mathcal{N}(\mathcal{F},\varepsilon,N)\le\lceil 4r^2/\varepsilon^2\rceil\log_2(2n+1)$.*

Corollary 10.

$$\ln\mathcal{N}(\mathcal{P}(n,l),\varepsilon,N)\le l\left\lceil\frac{4lr^2}{\varepsilon}\right\rceil\ln(2n+1)-l\ln 2-\ln l!$$

Proof. Since the spectral norm of a metric matrix $\mathbf{M}=\mathbf{L}\mathbf{L}^T$ is one, the ℓ^2-norm of a column of $\mathbf{L}$ is at most one. We may construct an ε-cover of $\mathcal{P}(n,l)$ from a $\sqrt{\varepsilon/l}$-cover of $\mathcal{F}$. Since the sign of $\mathbf{u}_i^T\mathbf{x}$ and the order of the l columns of $\mathbf{L}$ are irrelevant to the squared $\mathbf{M}$-distance, we have

$$\mathcal{N}(\mathcal{P}(n,l),\varepsilon,N)\le\frac{(\frac{1}{2}\mathcal{N}(\mathcal{F},\sqrt{\varepsilon/l},N))^l}{l!}\le\frac{(\frac{1}{2}(2n+1)^{\lceil 4lr^2/\varepsilon\rceil})^l}{l!}$$

and can complete the proof of this corollary by taking the natural logarithm for the both sides of the above inequality. □

Now, Theorem 5 can be proven as follows. Due to Lemma 8, we have

$$\sup_{\|\cdot\|_{\mathbf{M}}\in\mathcal{P}(n,l)} er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$$
$$\le\sup_{\|\cdot\|_{\mathbf{M}}\in\theta\varepsilon/16r^2-\mathcal{P}(n,l)} er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})+\varepsilon.$$

It then follows that

$$P_{\mathcal{X}}\left(\sup_{\|\cdot\|_{\mathbf{M}}\in\mathcal{P}(n,l)} er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>t\right)$$
$$\le P_{\mathcal{X}}\left(\sup_{\|\cdot\|_{\mathbf{M}}\in\theta\varepsilon/16r^2-\mathcal{P}(n,l)} er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>t-\varepsilon\right)$$
$$\le\mathcal{N}\left(\mathcal{P}(n,l),\frac{\theta\varepsilon}{16r^2},N\right)\sup_{\|\cdot\|_{\mathbf{M}}\in\theta\varepsilon/16r^2-\mathcal{P}(n,l)} P_{\mathcal{X}}(er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>t-\varepsilon)$$
$$\le\mathcal{N}\left(\mathcal{P}(n,l),\frac{\theta\varepsilon}{16r^2},N\right)\sup_{\|\cdot\|_{\mathbf{M}}\in\mathcal{P}(n,l)} P_{\mathcal{X}}(er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>t-\varepsilon),$$

where $\theta\varepsilon/16r^2-\mathcal{P}(n,l)$ denotes a $\theta\varepsilon/16r^2$-cover of $\mathcal{P}(n,l)$. Since $0\le\mathcal{L}_\theta\le1$, we may utilize Theorem 12 in [39]:

$$P_{\mathcal{X}}\left(er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>\varepsilon+\frac{\beta^*}{\theta}\right)\le\exp\left(-\frac{2N\varepsilon^2}{(4N\beta^*/\theta+1)^2}\right)$$

and obtain

$$P_{\mathcal{X}}\left(\sup_{\|\cdot\|_{\mathbf{M}}\in\mathcal{P}(n,l)} er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>2\varepsilon+\frac{\beta^*}{\theta}\right)$$
$$\le\mathcal{N}\left(\mathcal{P}(n,l),\frac{\theta\varepsilon}{16r^2},N\right)\exp\left(-\frac{2N\varepsilon^2}{(4N\beta^*/\theta+1)^2}\right)\tag{C.1}$$

by assigning $2\varepsilon+\beta^*/\theta$ to t. Thus, a PAC style generation error bound can be obtained by

$$er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})\le\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})+\frac{\beta^*}{\theta}$$
$$+2\left(\frac{4N\beta^*}{\theta}+1\right)\sqrt{\frac{1}{2N}\left(\ln\mathcal{N}\left(\mathcal{P}(n,l),\frac{\theta\varepsilon}{16r^2},N\right)+\ln(1/\delta)\right)}.$$

Since $er_{\mathcal{D}}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$ is not greater than $er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})$, the proof of the first part of Theorem 5 has been completed.

To prove the second part, we may expand inequality (C.1) as

$$P_{\mathcal{X}}\left(\sup_{\|\cdot\|_{\mathbf{M}}\in\mathcal{P}(n,l)} er_{\mathcal{D}}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})-\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})>2\varepsilon+\frac{\beta^*}{\theta}\right)$$
$$\le\frac{(\frac{1}{2}(2n+1)^{\lceil 64lr^4/(\theta\varepsilon)\rceil})^l}{l!}\exp\left(-\frac{2N\varepsilon^2}{(4N\beta^*/\theta+1)^2}\right)$$
$$\le((2n+1)^{\lceil 64lr^4/(\theta\varepsilon)\rceil})^l\exp\left(-\frac{2N\varepsilon^2}{(4N\beta^*/\theta+1)^2}\right)=\delta.$$

Thus, we have

$$\ln\delta\approx\frac{l\left\lceil\frac{64lr^4}{\theta}\right\rceil\ln(2n+1)}{\varepsilon}-\frac{2N\varepsilon^2}{(4N\beta^*/\theta+1)^2}$$

or equivalently

$$\varepsilon^2\approx\frac{(4N\beta^*/\theta+1)^2}{2N}\ln\frac{1}{\delta}+\frac{(4N\beta^*/\theta+1)^2}{2N}\times\frac{l\left\lceil\frac{64lr^4}{\theta}\right\rceil\ln(2n+1)}{\varepsilon}.\tag{C.2}$$

In addition, if $\delta<e^{-1}$, we can obtain $\varepsilon\ge\sqrt{(4N\beta^*/\theta+1)^2/2N}$, then substitute this result for ε in the right-hand side of Eq. (C.2), and get an upper bound for ε as follows:

$$\varepsilon\le\sqrt{\frac{4N\beta^*/\theta+1}{\sqrt{2N}}\times\left(l\left\lceil\frac{64lr^4}{\theta}\right\rceil\ln(2n+1)\right)+\frac{(4N\beta^*/\theta+1)^2}{2N}\ln\frac{1}{\delta}}.$$

Finally, by using this upper bound for ε, a generalization error bound can be obtained as follows

$$er_{\mathcal{D}}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})\le\hat{er}_{loo}^{\theta}(h_{\|\cdot\|_{\mathbf{M}};\mathcal{X}})+\frac{\beta^*}{\theta}$$
$$+\frac{2(4N\beta^*/\theta+1)}{\sqrt{2N}}\sqrt{\frac{\sqrt{2N}\left(l\left\lceil\frac{64lr^4}{\theta}\right\rceil\ln(2n+1)\right)}{4N\beta^*/\theta+1}+\ln\frac{1}{\delta}}.$$

Thus, the proof of the second part has been completed. □

References

[1] L. Yang, R. Jin, Distance Metric Learning: A Comprehensive Survey, Technical Report 24, Department of Computer Science and Engineering, Michigan State University, 2006.

[2] R.A. Fisher, The use of multiple measurements in taxonomic problems, Annals of Eugenics 7 (1936) 179–188.

[3] A. Bar-Hillel, T. Hertz, N. Shental, D. Weinshall, Learning a Mahalanobis metric from equivalence constraints, Journal of Machine Learning Research 6 (2005) 937–965.

[4] M. Loog, R.P.W. Duin, Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (6) (2004) 732–739.

[5] D. Xu, S. Yan, D. Tao, S. Lin, H.J. Zhang, Marginal Fisher analysis and its variants for human gait recognition and content-based image retrieval, IEEE Transactions on Image Processing 16 (11) (2007) 2811–2821.

[6] M. Sugiyama, Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis, Journal of Machine Learning Research 8 (2007) 1027–1061.

[7] K. Fukunaga, J.M. Mantock, Nonparametric discriminant analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 5 (1983) 671–678.

[8] F. Wang, C. Zhang, Feature extraction by maximizing the average neighborhood margin, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 1–8.

[9] T. Hastie, R. Tibshirani, Discriminant adaptive nearest neighbor classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (6) (1996) 607–616.

[10] K. Fukunaga, T.E. Flick, An optimal global nearest neighbor metric, IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6 (3) (1984) 314–318.

[11] S. Yan, D. Xu, B. Zhang, H.J. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1) (2007) 40–51.

[12] E.P. Xing, A.Y. Ng, M.I. Jordan, S. Russell, Distance metric learning with application to clustering with side-information, in: S.T.S. Becker, K. Obermayer (Eds.), Advances in Neural Information Processing Systems, vol. 15, MIT Press, Cambridge, MA, 2003, pp. 505–512.

[13] S. Yan, J. Liu, X. Tang, T.S. Huang, A parameter-free framework for general supervised subspace learning, IEEE Transactions on Information Forensics and Security 2 (1) (2007) 69–76.

[14] K.Q. Weinberger, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, Journal of Machine Learning Research 10 (2009) 207–244.

[15] R. Jin, S. Wang, Regularized distance metric learning: theory and algorithm, Advance in Neural Information Processing Systems, vol. NIPS 23, 2009.

[16] F. Wang, S. Chen, T. Li, C. Zhang, Semi-supervised metric learning by maximizing constraint margin, in: Proceedings of ACM 17th Conference on Information and Knowledge Management (CIKM), 2008, pp. 1457–1458.

[17] L. Yang, R. Jin, R. Sukthankar, Y. Liu, An efficient algorithm for local distance metric learning, in: Proceedings of the Twenty-first National Conference on Artificial Intelligence, 2006.

[18] J. Goldberger, S. Roweis, G. Hinton, R. Salakhutdinov, Neighbourhood components analysis, in: L.K. Saul, Y. Weiss, L. Bottou (Eds.), Advances in Neural Information Processing Systems, vol. 17, MIT Press, Cambridge, MA, 2005, pp. 513–520.
[19] Y. Sun, Iterative RELIEF for feature weighting: algorithms, theories and applications, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (6) (2007) 1035–1051.
[20] Y. Sun, S. Todorovic, S. Goodison, Local-learning-based feature selection for high-dimensional data analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2010) 1610–1626.
[21] J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: a statistical view of boosting, Annals of Statistics 28 (2000) 337–407.
[22] R. Meir, G. Rätsch, An introduction to boosting and leveraging, in: Advanced Lectures on Machine Learning, Lecture Notes in Computer Science, Springer, 2003, pp. 119–184.
[23] R.E. Schapire, The boosting approach to machine learning: an overview, in: Workshop on Nonlinear Estimation and Classification, MSRI, 2002.
[24] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, Journal of Computer and System Science 55 (1) (1997) 119–139.
[25] R.E. Schapire, Y. Singer, Improved boosting algorithms using confidence-rated predictions, Machine Learning 37 (3) (1999) 297–336.
[26] T. Hertz, A. Bar-Hillel, D. Weinshall, Boosting margin based distance functions for clustering, in: Proceedings of the Twenty-First International Conference on Machine Learning, 2004, pp. 393–400.
[27] T. Hertz, A. Bar-Hillel, D. Weinshall, Learning a kernel function for classification with small training samples, in: ICML '06: Proceedings of the 23rd International Conference on Machine Learning, ACM, New York, NY, USA, 2006, pp. 401–408.
[28] L. Yang, R. Jin, L. Mummert, R. Sukthankar, A. Goode, B. Zheng, S.C.H. Hoi, M. Satyanarayanan, A boosting framework for visuality-preserving distance metric learning and its application to medical image retrieval, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (1) (2010) 30–44.
[29] K. Crammer, J. Keshet, Y. Singer, Kernel design using boosting, Advances in Neural Information Processing Systems, vol. 15, MIT Press, 2003, pp. 537–544.
[30] K. Crammer, R. Gilad-bachrach, A. Navot, N. Tishby, Margin analysis of the LVQ algorithm, Advances in Neural Information Processing Systems, vol. 14, MIT Press, 2002, pp. 462–469.
[31] R. Gilad-Bachrach, A. Navot, N. Tishby, Margin based feature selection—theory and algorithms, in: International Conference on Machine Learning (ICML), ACM Press, 2004, pp. 43–50.
[32] R. Kohavi, G. John, Wrappers for feature subset selection, Artificial Intelligence 97 (1997) 273–324.
[33] C.C. Chang, Generalized iterative RELIEF for supervised distance metric learning, Pattern Recognition 43 (2010) 2971–2981.
[34] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, Inc., New York, 1991.
[35] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society Series B 39 (1) (1977) 1–38.
[36] F.R.K. Chung, Spectral Graph Theory, American Mathematical Society, 1997.
[37] D. Hush, C. Scovel, I. Steinwart, Stability of unstable learning algorithm, Machine Learning 67 (2007) 197–206.
[38] T. Zhang, Covering number bounds of certain regularized linear function classes, Journal of Machine Learning Research 2 (2002) 527–550.
[39] O. Bousquet, A. Elisseeff, Stability and generalization, Journal of Machine Learning Research 2 (2002) 499–526.
[40] A. Asuncion, D. Newman, UCI Machine Learning Repository, 〈http://www.ics.uci.edu/∼mlearn/MLRepository.html〉, 2007.
[41] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine, Broad patterns of gene expression revealed by clustering of tumor and normal colon tissues probed by oligonucleotide arrays, Proceedings of National Academy of Science USA 96 (12) (1999) 6745–6750.
[42] D. Singh, P.G. Febbo, K. Ross, D.G. Jackson, J. Manola, C. Ladd, P. Tamayo, A.A. Renshaw, A.V. D'Amico, J.P. Richie, E.S. Lander, M. Loda, P.W. Kantoff, T.R. Golub, W.R. Sellers, Gene expression correlates of clinical prostate cancer behavior, Cancer Cell 1 (2002) 203–209.
[43] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J. Mesirov, T. Poggio, W. Gerald, M. Loda, E. Lander, R. Golub, Multiclass cancer diagnosis using tumor gene expression signatures, Proceedings of National Academy of Science USA 98 (26) (2001) 15149–15154.
[44] Georgia Tech Face Database, 〈http://www.anefian.com/face_reco.htm〉.
[45] K. Lee, J. Ho, D. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (5) (2005) 684–698.
[46] A.M. Martinez, R. Benavente, The AR Face Database, Technical Report 24, CVC, June 1998.
[47] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, 2001.
[48] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second ed., John Wiley & Sons, Inc., 2001.
[49] K. Fukunaga, Introduction to Statistical Pattern Recognition, second ed., Academic Press, Boston, 1990.

Chin-Chun Chang received the B.S. degree and the M.S. degree in computer science in 1989 and 1991, respectively, and the Ph.D. degree in computer science in 2000, all from National Chiao Tung University, Hsinchu, Taiwan.

From 2001 to 2002, he was a faculty of the Department of Computer Science and Engineering, Tatung University, Taipei, Taiwan. In 2002, he joined the Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, Taiwan, where he is currently an Assistant Professor. His research interests include computer vision, machine learning, and pattern recognition.