Trace Ratio vs. Ratio Trace for Dimensionality Reduction

Huan Wang[1] Shuicheng Yan[2] Dong Xu[3] Xiaoou Tang [1,4] Thomas Huang[2]

[1] IE, Chinese University of Hong Kong, Hong Kong
hwang5@ie.cuhk.edu.hk

[2] ECE, University of Illinois at Urbana-Champaign, USA
{scyan,huang}@ifp.uiuc.edu

[3] EE, Columbia University New York, USA
dongxu@ee.columbia.edu

[4] Microsoft Research Asia Beijing, China
xitang@microsoft.com

Abstract

A large family of algorithms for dimensionality reduction end with solving a Trace Ratio *problem in the form of* $\arg\max_W Tr(W^T S_p W)/Tr(W^T S_l W)$[1]*, which is generally transformed into the corresponding* Ratio Trace *form* $\arg\max_W Tr[\ (W^T S_l W)^{-1}(W^T S_p W)\ ]$ *for obtaining a closed-form but inexact solution. In this work, an efficient iterative procedure is presented to* directly *solve the Trace Ratio problem. In each step, a* Trace Difference *problem* $\arg\max_W Tr[W^T(S_p - \lambda S_l)W]$ *is solved with* λ *being the trace ratio value computed from the previous step. Convergence of the projection matrix* W*, as well as the global optimum of the trace ratio value* λ*, are proven based on* point-to-set map *theories. In addition, this procedure is further extended for solving trace ratio problems with more general constraint* $W^T CW{=}I$ *and providing exact solutions for kernel-based subspace learning problems. Extensive experiments on faces and UCI data demonstrate the high convergence speed of the proposed solution, as well as its superiority in classification capability over corresponding solutions to the ratio trace problem.*

1. Introduction

Variations in a set of high-dimensional data, such as images, often have an underlying low-dimensional structure that compactly characterizes the changes among these observations [1][8]. To uncover this low-dimensional structure of the data, dimensionality reduction has been an active research topic in computer vision and pattern recognition.

Yan et al. [10] claimed that most traditional algorithms for dimensionality reduction can be unified within a general framework, called *Graph Embedding*. This framework derives a low-dimensional feature space which preserves the adjacency relationship between different sample pairs in addition to constraints from scale normalization or a penalty graph [10]. Within this context, many algorithms for dimensionality reduction involve a search for a transformation matrix W that maximizes a term $Tr(W^T S_p W)$ and at the same time minimizes another term $Tr(W^T S_l W)$, where matrices S_p and S_l are both positive semidefinite. The natural solution to these dual objectives is to pose a *trace ratio* optimization problem, namely $\max_W Tr(W^T S_p W)/Tr(W^T S_l W)$, which however does not have a closed-form solution. Generally, the trace ratio problem is often simplified into a more tractable one called the *ratio trace* problem: $\max_W Tr[(W^T S_l W)^{-1}(W^T S_p W)]$. The ratio trace problem can be efficiently solved with the generalized eigenvalue decomposition method [3]. However, its solution may deviate from the original objectives and suffers from the fact that it is invariant under any non-singular transformation, which may lead to uncertainty in subsequent processing such as classification and clustering.

In this work, we tackle the original trace ratio problem, and present a procedure to directly optimize the objective function $Tr(W^T S_p W)/Tr(W^T S_l W)$ by assuming that the column vectors of W are unitary and orthogonal to each other. The procedure iteratively optimizes the objective function, and the projection matrix W^n of the n-th step is obtained by solving a corresponding *trace difference* problem $\max_W Tr[W^T(S_p - \lambda^n S_l)W]$ where λ^n is the trace ratio value computed from W^{n-1}. Therefore, the sub-problem in each step can be efficiently solved with the eigenvalue decomposition method [3]. A detailed proof is provided to justify that λ^n will increase monotonically until reaching the global optimum; the convergence of the projection matrix W^n is also proven. In addition, this procedure is extended to handle the trace ratio problem with a more general constraint that $W^T CW = I$, where matrix C is positive semidefinite. This extension provides the exact solution to kernel-based subspace learning algorithms with trace ratio formulations.

The rest of the paper is organized as follows. Section 2 presents a detailed comparison of the trace ratio and ratio trace formulations for dimensionality reduction. Then,

[1] W is the desired transformation matrix; S_p, S_l and later-introduced C are constant positive semidefinite matrices.

1-4244-1180-7/07/$25.00 ©2007 IEEE

the iterative procedure to solve the trace ratio problem is introduced in Section 3, and Section 4 gives the convergence proof. The extension for more general constraints and its related works are introduced in Section 5. Experimental results are presented in Section 6, and we conclude this paper in Section 7.

2. Dimensionality Reduction Formulations: Trace Ratio vs. Ratio Trace

For a classification problem, assume that the training data are given as $\{x_i|x_i \in \mathbb{R}^m\}_{i=1}^N$, where N is the number of training samples. The corresponding class labels of the samples are denoted as $\{c_i|c_i \in \{1, ..., N_c\}\}_{i=1}^N$, where N_c is the number of classes, and the number of samples belonging to the c-th class is denoted as n_c. In practice, dimensionality reduction is in great demand owing to the fact that the effective information for classification often lies within a lower dimensional feature space.

A simple but effective way for dimensionality reduction is to find a matrix $W = [w_1, w_2, ..., w_d] \in \mathbb{R}^{m\times d}$ (*Rank*(W) =d, $\|w_k\|$=1, k=1, 2,. . ., d) to transform the original high-dimensional data x into a low-dimensional form $y \in \mathbb{R}^d$ (usually $d \ll m$) as

$$y = W^T x. \tag{1}$$

Many algorithms [1][8][10][11] with various motivations have been proposed to find such a W. Yan et al. [10] claimed that most dimensionality reduction algorithms can be unified into a general framework, namely graph embedding which is described as follows.

Let $G = \{X, S\}$ be an undirected weighted graph with vertex set X and similarity matrix $S \in \mathbb{R}^{N\times N}$. Each element of the real symmetric matrix S measures the similarity between a pair of vertices. The diagonal matrix D and the Laplacian matrix L of a graph G are defined as $L = D - S$ and $D_{ii} = \sum_{j\neq i} S_{ij}$, $\forall\, i$. For a specific dimensionality reduction algorithm, there may exist two graphs, the intrinsic graph $G = \{X, S\}$ and the penalty graph $G^p = \{X, S^p\}$ with $L^p = D^p - W^p$ and $D^p_{ii} = \sum_{j\neq i} S^p_{ij}$, $\forall\, i$. The intrinsic graph characterizes data properties that the algorithm favors and the penalty graph describes properties that the algorithm tries to avoid. A graph preserving criterion is imposed for these two objectives:

$$\arg\min_W \frac{\sum_{i\neq j} \|W^T x_i - W^T x_j\|^2 S_{ij}}{\sum_{i\neq j} \|W^T x_i - W^T x_j\|^2 S^p_{ij}}, \tag{2}$$

which can be further formulated in trace ratio form [10]:

$$\arg\max_W \{\frac{Tr(W^T X L^p X^T W)}{Tr(W^T X L X^T W)} = \frac{Tr(W^T S_p W)}{Tr(W^T S_l W)}\}, \tag{3}$$

where $S_p = XL^pX^T$ and $S_l = XLX^T$. Note that for the graph embedding framework in [10], the denominator of (2) can also be defined as a constraint for scale normalization, which will also result in the trace ratio optimization problem as in the latter part of (3).

Many popular dimensionality reduction algorithms, such as Linear Discriminant Analysis (LDA) [1] and the non-parametric algorithm Marginal Fisher Analysis (MFA) [10] (or similarly, Local Discriminant Embedding [2]), can be formulated in the above graph embedding framework. For example, LDA searches for a subspace that minimizes intra-class scatter and at the same time maximizes inter-class scatter:

$$W^* = \min_W \frac{\sum_{i=1}^N \|W^T x_i - W^T \bar{x}_{c_i}\|^2}{\sum_{c=1}^{N_c} n_c \|W^T \bar{x}_c - W^T \bar{x}\|^2}, \tag{4}$$

where $\bar{x}_c$ is the mean of samples belonging to the c-th class and $\bar{x}$ is the mean of all samples. From the graph embedding point of view in (3), the similarity matrices for the intrinsic and penalty graphs of LDA are defined as

$$S_{ij} = \delta_{c_i,c_j}/n_{c_i}, i \neq j, \tag{5}$$
$$S^p_{ij} = 1/N - S_{ij}, i \neq j, \tag{6}$$

where δ_{c_i,c_j}=1 if c_i=c_j, and δ_{c_i,c_j}=0 otherwise.

This optimization problem is typically nonconvex, and there does not exist a closed-form solution for the general trace ratio problem (3); hence such problems are often transformed into the simpler yet inexact ratio trace problem, which is equivalent to the determinant ratio problem [3]. For (3), the corresponding ratio trace (determinant ratio) form is

$$W^* = \arg\max_W Tr[(W^T S_l W)^{-1}(W^T S_p W)] \tag{7}$$
$$= \arg\max_W \frac{|W^T S_p W|}{|W^T S_l W|} \tag{8}$$

which can be directly solved with the generalized eigenvalue decomposition (GEVD) method:

$$S_p w_k = \tau_k S_l w_k \tag{9}$$

where τ_k is the k-th largest eigenvalue of the GEVD with the corresponding eigenvector w_k, and w_k constitutes the k-th column vector of the matrix W.

Remarks. Despite the existence of a closed-form solution for ratio trace optimization problem, the obtained solution does not necessarily best optimize the corresponding trace ratio optimization algorithm, which is the essential objective function for general dimensionality reduction. For supervised dimensionality reduction algorithms, this approximation may sacrifice the potential classification capability of the derived low-dimensional feature space, which is demonstrated later in our experiments. This motivates the need for a procedure to directly solve the trace ratio optimization problem.

3. Efficient Solution of Trace Ratio Problem

In this section, we present an efficient procedure to solve the trace ratio problem with the assumption that $W^TW = I_d$. Denote $S_t = S_p + S_l$, then the trace ratio optimization problem in (3) is equivalent to

$$W^* = \arg \max_{W^TW=I_d} \frac{Tr(W^TS_pW)}{Tr(W^TS_tW)}. \quad (10)$$

We have $0 \leq Tr(W^TS_pW)/Tr(W^TS_tW) \leq 1$, and the maximum value 1 of (10) corresponds to the maximum of (3), namely $+\infty$. Without losing generality, we instead solve for (10) in the following. Our procedure consists of two steps.

1. Remove the *Null Space* of S_t with Principal Components Analysis (PCA) [8]. The matrices S_p and S_l are both positive semidefinite, and the intersection of their null spaces is equal to the null space of S_t, namely, $\{x|S_tx = 0\}$. As the null space of S_t does not contain discriminating information for the training data ($x^TS_px = 0$ and $x^TS_lx = 0$), they may be removed from the solution space without sacrificing accuracy. Assume that the singular value decomposition of matrix S_t is

$$S_t = U\Lambda U^T,$$

where $\Lambda = [\lambda_1, \lambda_2, \ldots, \lambda_{m'}]$, $\lambda_k > 0, k = 1, 2, \ldots, m'$, and m' is the number of positive singular values of S_t. Then the solution is constrained to lie within the space spanned by the column vectors of U, namely, $W = UV, V \in \mathbb{R}^{m' \times d}$, and the problem defined in (10) is changed to

$$V^* = \arg \max_{V^TV=I_d} \frac{Tr(V^TS_p^uV)}{Tr(V^TS_t^uV)}. \quad (11)$$

where $S_p^u = U^TS_pU$ and $S_t^u = U^TS_tU$. Then, the denominator of the objective function (11) is always positive for non-zero V, that is, S_t^u is positive definite.

2. Iterative optimization. Here, we first introduce our iterative algorithm to solve (11). Its theoretical justifications will be presented in Section-4. In each step, we solve a trace difference problem

$$V^* = \arg \max_{V^TV=I_d} Tr[V^T(S_p^u - \lambda^nS_t^u)V],$$

where λ^n is the trace ratio value calculated from the projection matrix V^{n-1} of the previous step. The detailed procedure is listed in Algorithm 1.

Algorithm 1 . Iterative Procedure to Solve the Trace Ratio Optimization Problem

1: Initialize V^0 as an arbitrary columnly orthogonal matrix;

2: For n=1, 2, ..., N_{max}, Do

1. Compute the trace ratio value λ^n from the projection matrix V^{n-1}:

$$\lambda^n = \frac{Tr[{V^{n-1}}^TS_p^uV^{n-1}]}{Tr[{V^{n-1}}^TS_t^uV^{n-1}]}. \quad (12)$$

2. Construct the trace difference problem as

$$V^n = \arg \max_{V^TV=I_d} Tr[V^T(S_p^u - \lambda^nS_t^u)V]. \quad (13)$$

3. Solve the trace difference problem using the eigenvalue decomposition method:

$$(S_p^u - \lambda^nS_t^u)v_k^n = \tau_k^nv_k^n, \quad (14)$$

where τ_k^n is the k-th largest eigenvalue of $(S_p^u - \lambda^nS_t^u)$ with the corresponding eigenvector v_k^n.

4. Reshape the projection matrix for the sake of orthogonal transformation invariance:

   (a) Set $V^n = [v_1^n, v_2^n, \ldots, v_d^n]$, where d is the desired lower feature dimension;

   (b) Let $S_t^v = V^n(V^n)^TS_t^uV^n(V^n)^T$;

   (c) Conduct singular value decomposition as

$$S_t^v = V^n\Lambda^n{V^n}^T. \quad (15)$$

5. If $\|V^n - V^{n-1}\| < \sqrt{m'd}\,\varepsilon$ (ε is set to 10^{-4} in this work), then break.

3: Output $V = V^n$.

4. Proof of Convergency to Global Optimum

4.1. Proof of the monotonic increase of λ^n

Denote the objective function of (11) as

$$J(V) = \frac{Tr(V^TS_p^uV)}{Tr(V^TS_t^uV)}. \quad (16)$$

Then, the monotonic increase of λ^n is guaranteed by the following theorem.

Lemma-1. For Algorithm 1 to solve the trace ratio optimization problem, we must have

$$J(V^n) \geq J(V^{n-1}), \quad \text{namely} \quad \lambda^{n+1} \geq \lambda^n. \quad (17)$$

Proof. Denote $g_n(V) = Tr(V^T(S_p^u - \lambda^nS_t^u)V)$, then $g_n(V^{n-1}) = 0$. Moreover, from Algorithm 1 and the assumption that $V^TV = I_d$, we have [6]

$$\sup_{V^TV=I_d} g_n(V) = \sum_{k=1}^{d} \tau_k^n.$$

Also, $g_n(V^n) = \sum_{k=1}^{d} \tau_k^n$ from (14) and (15). Then we have

$$g_n(V^n) \geq g_n(V^{n-1}) = 0.$$

Namely, $Tr[{V^n}^T(S_p^u - \lambda^n S_t^u)V^n] \geq 0$. As the matrix S_t^u is positive definite, we obtain

$$\frac{Tr({V^n}^T S_p^u V^n)}{Tr({V^n}^T S_t^u V^n)} \geq \lambda^n,$$

that is,

$$J(V^n) \geq J(V^{n-1}), \text{ namely } \lambda^{n+1} \geq \lambda^n. \qquad \square$$

From Theorem-1, we can conclude that the trace ratio value will monotonically increase.

4.2. Proof of V^n convergence and global optimum for λ

To prove the convergence of the projection matrix V^n, we first introduce the concept of *point-to-set* mapping and some related lemmas [7]. The power set $\wp(\chi)$ of a set χ is the collection of all subsets of χ. A *point-to-set* map Ω is a function: $\chi \rightarrow \wp(\chi)$ [5]. In our iterative procedure to the trace ratio optimization problem, the map from V^{n-1} to V^n can be considered as a point-to-set map, since each V^n with any orthogonal transformation will not change the value of the objective function $J(V^n)$.

Strict Monotonicity [5]. An algorithm is a point-to-set map Ω:$\chi \rightarrow \wp(\chi)$. Given an initial point X_1, the algorithm generates a sequence of points via the rule that $X_n \in \Omega(X_{n-1})$. Suppose $J : \chi \rightarrow \mathbb{R}_+$ is a continuous, non-negative function; an algorithm is called *strictly monotonic* if 1) $Y \in \Omega(X)$ implies that $J(Y) \geq J(X)$, and 2) $Y \in \Omega(X)$ and $J(Y) = J(X)$ imply that $Y = X$.

To prove the convergence of the projection matrix V^n, we will utilize the following lemma.

Lemma-2 [7]. Assume that the algorithm Ω is strictly monotonic with respect to J and that it generates a sequence $\{X_n\}$ which lies in a compact set. If χ is normed, then $\|X_n - X_{n-1}\| \rightarrow 0$. If Ω is closed at an accumulation point $\hat{X}$, $\lim_{n\rightarrow\infty} X^n = \hat{X}$, then $\hat{X}$ is a fixed point, namely $\{\hat{X}\} = \Omega(\hat{X})$.

In our proposed iterative procedure (12-15) for the trace ratio optimization problem, let $\chi = \mathbb{R}^{m' \times d} \cap \varphi$, where the set $\varphi = \{X | \exists\, V \in \mathbb{R}^{m' \times d}$, X is the solution of (12-15) with V^{n-1}=$V\}$. Then the iterative algorithm from (12) to (15), denoted as Ω, is a point-to-set map (the set contains one point when V^n is constrained to have been obtained from (12-15)), and it generates a sequence of points via the rule that $V^n = \Omega(V^{n-1})$. This algorithm is strictly monotonic as proven below.

Lemma-3 The iterative algorithm (12-15) for the trace ratio optimization problem is strictly monotonic with respect to $J = J(V)$ as defined in (16).

Proof. It is obvious that $J(V)$ is a continuous, non-negative function. From Lemma-1, we have $J(V^n) \geq J(V^{n-1})$, hence the first condition for strict monotonicity is satisfied. For the second condition, if $Y = \Omega(X)$ and $J(Y) = J(X)$, then $\max_V Tr[V^T(S_p^u - \lambda S_t^u)V] = 0$ where $\lambda = Tr(X^T S_p^u X)/Tr(X^T S_t^u X)$; otherwise, we have $J(Y) > J(X)$. As S_t^u is positive definite, we have $\max_V \frac{Tr(V^T S_p^u V)}{Tr(V^T S_t^u V)} = \lambda$. Since both X and Y achieve the maximum of the objective function $J(V)$, they both maximize $Tr[V^T(S_p^u - \lambda S_t^u)V]$ [4]. As the maximum of $Tr[V^T(S_p^u - \lambda S_t^u)V]$ lies in the space spanned by the first d eigenvectors of the matrix $(S_p^u - \lambda S_t^u)$ [6][2], there only exists one orthogonal transform between these two matrices. Given that both X and Y are constrained in the set φ, they are orthogonal transform invariant as in (15); therefore, we have $X = Y$. Then, we can conclude that the iterative algorithm (12-15) for the trace ratio optimization problem is strictly monotonic with respect to $J = J(V)$ as defined in (16). $\square$

All possible λ^n computed from the set $\{V \mid V^T V = I_d, V \in \mathbb{R}^{m' \times d}\}$ will constitute a compact set. As described above, all members of χ are computed from (14) and (15); hence χ is compact owing to the continuity of the mapping from λ^n to V^n. A more detailed proof is omitted here.

Based on the above lemmas, we can have the following theorem on the convergence of the λ^n to the global optimum.

Theorem-1. For the iterative procedure (12-15) defined in Algorithm 1, we have $\|V^n - V^{n-1}\| \rightarrow 0$. Denote $\lim_{n\rightarrow\infty} V^n = V$, then $V \in \arg\max_V \frac{Tr(V^T S_p^u V)}{Tr(V^T S_t^u V)}$, that is, λ^n will monotonically increase and converge to the global optimum.

Proof. From Lemma-2 and Lemma-3, we can directly reach the conclusion that $\|V^n - V^{n-1}\| \rightarrow 0$. From Lemma-2, we have $J(V) = J(\Omega(V))$. According to the proof of Lemma-3, we have $V \in \arg\max \frac{Tr(V^T S_p^u V)}{Tr(V^T S_t^u V)}$.

As proven in Lemma-1, λ^n will monotonically increase, hence we can conclude that λ^n will monotonically increase and converge to the global optimum along with the corresponding projection matrix V^n. $\square$

5. Extension and Discussion

5.1. Extension to General Constraints

As mentioned previously, we have the assumption $W^T W = I$ for the trace ratio optimization problem. In practice, a more general constraint $W^T C W = I$, where C is any positive semidefinite matrix, may be imposed. Here,

[2]It is deduced with the assumption that there do not exist duplicated eigenvalues for $(S_p^u - \lambda S_t^u)$.

we take as example the kernelization of the graph embedding framework in [10], where C is a kernel matrix, to introduce how to solve the trace ratio optimization problem with general constraints.

The intuition of the kernel trick is to map the data from the original input feature space to another higher dimensional Hilbert space as $\phi : x \rightarrow \mathcal{F}$, and then perform linear dimensionality reduction in this new feature space. This approach is well suited to algorithms that need only to compute the inner product of data pairs $k(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$. Assuming that the transformation matrix $W = [\phi(x_1), \phi(x_2), \cdots, \phi(x_N)]\tilde{W}$ and K is the kernel Gram matrix with $K_{ij} = \phi(x_i) \cdot \phi(x_j)$, we have the following optimization problem from (3):

$$\max_{\tilde{W}^T K \tilde{W} = I} \{\frac{Tr(\tilde{W}^T K L^p K \tilde{W})}{Tr(\tilde{W}^T K L K \tilde{W})} = \frac{Tr(\tilde{W}^T S_p^k \tilde{W})}{Tr(\tilde{W}^T S_l^k \tilde{W})}\}, \quad (18)$$

where the constraint matrices are $S_p^k = KL^pK$ and $S_l^k = KLK$.

Assume that the singular value decomposition of the kernel matrix K is $K = U_k \Lambda_k (U_k)^T$, where Λ_k is a diagonal matrix with positive diagonal elements. Let $\tilde{W} = U_k \Lambda_k^{-1/2} W_1$, then we can simplify the optimization problem (18) into

$$\max_{W_1^T W_1 = I} \frac{Tr(W_1^T S_p^1 W_1)}{Tr(W_1^T S_l^1 W_1)}, \quad (19)$$

where the constant matrix $S_p^1 = \Lambda_k^{-1/2} U_k^T S_p^k U_k \Lambda_k^{-1/2}$ and $S_l^1 = \Lambda_k^{-1/2} U_k^T S_l^k U_k \Lambda_k^{-1/2}$. Now this optimization problem is converted into the form in (3), and hence we can use the proposed Algorithm 1 to search for the global optimum.

5.2. Discussion

Trace Ratio vs. Ratio Trace: Trace ratio and ratio trace present two different formulations to the general dimensionality reduction problem. They are interlinked in the following aspects. First, for the trace ratio formulation, the objective function is invariant under any orthogonal transformation of V; while for the ratio trace formulation, it is invariant under any non-singular transformation matrix of V. Hence, the former is invariant for classification if based on Euclidean distance, while different solutions of the latter may change the similarity and thus is unstable for classification. Second, the ratio trace formulation has a closed-form solution and is more efficient compared to the trace ratio formulation. Third, for the ratio trace formulation, the column vectors of the projection matrix are not required to be orthogonal, hence it essentially puts different weights on different projection directions. Assume that the singular value decomposition of W is $W = U_w \Lambda_w V_w^T$; then, as the right orthogonal matrix V_w will not change the similarity if it is based on the Euclidean distance, the projection directions encoded by the column vectors of U_w are given different weights from the diagonal elements of Λ_w. Finally, the trace ratio formulation is the essential formulation for general dimensionality problem, which directly leads to the superiority of the solution from the trace ratio formulation over that from the ratio trace formulation.

Relationship with Guo's Work [4]: Guo et al. [4] proposed a method to solve the trace ratio problem. Our proposed algorithm is different from Guo's in many aspects. First, Guo's work proves convergence of only the trace ratio value, and does not prove the convergence of the projection matrix V; while our algorithm provides convergence proofs for both the trace ratio value and the projection matrix V. Second, Guo's work utilizes the dichotomy method to select the trace ratio value for their trace difference formulation, which commonly exhibits slow convergence of the trace ratio value. As shown in the experiment section, our proposed method converges much faster than Guo's work. Third, in our proposed algorithm, the trace ratio value increases monotonically, hence it is guaranteed that the performance improves step-by-step; while in Guo's work, the derived trace ratio value may fluctuate. Finally, our algorithm is proposed for general dimensionality reduction problems and further extended for solving kernel-based subspace learning problems formulated in trace ratio form. The work in [9] also discussed the trace ratio problem and applied the multi-scale search for pursuing the solution; hence also suffers from the same issues as Guo's work.

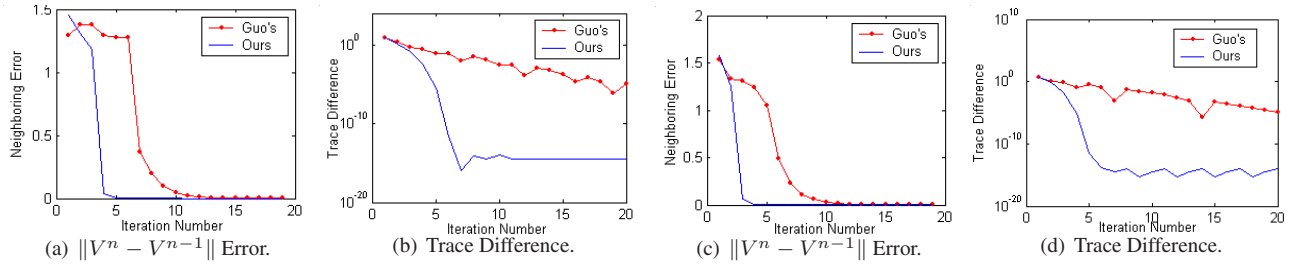
6. Experiments

In this section, our proposed **I**terative algorithm for the **T**race **R**atio (ITR) optimization problem is systematically evaluated in four aspects, taking the LDA and MFA [10] algorithms as instances of trace ratio problems. The first is the evaluation of convergence speed in comparison to Guo's work [4]; the second is visualization of the projection matrices of ITR compared to PCA and the ratio trace based LDA; the third is evaluation of the classification capability of the derived low-dimensional feature spaces from linear dimensionality reduction algorithms; and the fourth is evaluation of the classification capability of the derived low-dimensional feature space for kernel-based dimensionality reduction algorithms.

6.1. Dataset Preparation

In our experiments, we use six data sets. The first three are the benchmark face databases FERET, ORL, and CMU PIE[1] with high-dimensional features. For the face databases, all images are aligned by fixing the locations of the two eyes. From the FERET database, we use seventy

[1] Available at http://www.face-rec.org/databases/.



(a) $\|V^n - V^{n-1}\|$ Error. (b) Trace Difference. (c) $\|V^n - V^{n-1}\|$ Error. (d) Trace Difference.

Figure 1. $\|V^n - V^{n-1}\|$ error vs. iteration number, and the trace difference $|Tr[V^{nT}(S_p - \lambda^n S_t)V^n]|$ (implies the error between V^n and the optimum) vs. iteration number. (a-b) FERET database, and (c-d) CMU PIE database.

people with six images for each person; the images are normalized in size to 56-by-46 pixels. The ORL database contains 400 images of 40 persons, where each image is normalized in size to 56-by-46 pixels. The CMU PIE (Pose, Illumination, and Expression) database contains more than 40,000 facial images of 68 people. In our experiment, a subset of five near frontal poses (C27, C05, C29, C09 and C07) and illuminations indexed as 08 and 11 is used. Each person has ten images and all the images are normalized to 64-by-64 pixels. The other three data sets are wine, iris, and ionosphere (iono) from the UC Irvine repository[2]; this data has relatively small feature dimensions.

6.2. Convergence Speed

In this subsection, the convergence property of our proposed algorithm ITR is compared to Guo's method. The convergence property is evaluated in two aspects. One is the convergence of the projection matrix V^n, determined according to the difference of V^n and V^{n-1} ($\|V^n - V^{n-1}\|$). The other is the speed of $|Tr[V^{nT}(S_p - \lambda^n S_t)V^n]|$ converging to zero. As described in the proof of Lemma-3, the largest trace ratio value results in $Tr[V^{nT}(S_p - \lambda^n S_t)V^n] = 0$; hence this evaluation measures the convergence speed of the trace ratio value to the global optimum, and the accuracy of the projection matrix V.

The FERET and CMU PIE databases are used for these evaluations. For both ITR and Guo's method, we optimize the objective function in (11) and from the LDA algorithm. Detailed results are shown in Figure 1, from which we can see that ITR converges much faster than Guo's method. Commonly, ITR converges after about 5 iterations. Moreover, the accuracy of the trace ratio value, characterized by the value of $|Tr[V^{nT}(S_p - \lambda^n S_t)V^n]|$, from the ITR algorithm is much better than that from Guo's method.

6.3. Visualization of Projection Matrix

In this subsection, we examine the visual properties of the projection matrix W computed by our proposed ITR algorithm within LDA, and compare it to the traditional ratio trace formulation within LDA and to the PCA algorithm.

[2]Available at http://www.ics.uci.edu/ mlearn/MLRepository.html.

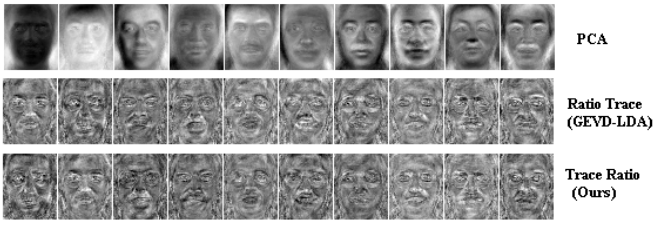


Figure 2. Visualization of the projection matrix W of PCA, ratio trace based LDA, and trace ratio based LDA (ITR) on the FERET database.

The FERET and ORL databases are used for this experiment. From the FERET database, three images of each subject are randomly selected for the computation of the projection matrices, while four images of each subject are taken from the ORL database. For the ITR algorithm, the reduced feature dimension is set to $d = 10$ for the computation of the projection matrix. For LDA related algorithms, we first conduct PCA to reduce the dimension to N-N_c as in [1], and then perform LDA. The final projection matrix is the product of the PCA projection matrix and the LDA projection matrix.

The column vectors of the projection matrix are reshaped into a matrix of the original image size. All the results are shown in Figure 2. They demonstrate that PCA vectors look similar to a face, which agrees with the motivation of PCA; while the results from LDA related algorithms are more noisy, which indicates that the most discriminating features perhaps do not possess explicit global semantics.

6.4. Classification by Linear Trace Ratio Algorithms with Orthogonal Constraints

In this subsection, we conduct classification experiments on the face databases with high feature dimensions. Our proposed ITR procedure is compared to the ratio trace solution using LDA and MFA as examples. For MFA related algorithms, the number of nearest neighbors of each sample is fixed as 4, and the number of shortest pairs from different classes is set as 40. To speed up model training, PCA is computed as a preprocessing step for LDA/MFA related algorithms. Two PCA dimensions are tested before LDA/MFA: one is N-N_c, which is equivalent to the Fisher-

Table 1. Recognition error rates (%) of PCA, PCA (N-N_c)+Ratio Trace based LDA (RLDA), PCA (N-N_c)+Trace Ratio based LDA (ITR-LDA), PCA (N-1)+Ratio Trace based LDA (RLDA2), PCA (N-1)+Trace Ratio based LDA (ITR-LDA2), PCA (N-N_c)+Ratio Trace based MFA (RMFA), PCA (N-N_c)+Trace Ratio based MFA (ITR-MFA), PCA (N-1)+Ratio Trace based (RMFA2), PCA (N-1)+Trace Ratio based MFA (ITR-MFA2), and the method without dimensionality reduction on the three face databases. Note that the boldtype numbers are those with the best classification accuracies for LDA and MFA respectively.

Configure	Unsupervised		LDA Related Algorithms				MFA Related Algorithms			
FERET	w/o DR.	PCA	RLDA	ITR-LDA	RLDA2	ITR-LDA2	RMFA	ITR-MFA	RMFA2	ITR-MFA2
N3T3	17.1	16.2	8.6	7.6	7.6	**6.7**	9.5	5.3	6.2	**4.8**
N2T4	32.9	33.2	21.1	21.8	18.9	**17.1**	20.7	22.0	18.6	**16.8**
ORL	w/o DR.	PCA	RLDA	ITR-LDA	RLDA2	ITR-LDA2	RMFA	ITR-MFA	RMFA2	ITR-MFA2
N4T6	12.1	10.8	11.7	6.7	11.7	**5.8**	10.8	6.7	7.1	**5.4**
N3T7	18.6	17.5	15.7	13.2	16	**11.8**	15.4	11.8	12.1	**11.8**
N2T8	28.4	27.5	28.4	24.4	**22.2**	22.5	27.5	23.7	24.7	**22.2**
PIE	w/o DR.	PCA	RLDA	ITR-LDA	RLDA2	ITR-LDA2	RMFA	ITR-MFA	RMFA2	ITR-MFA2
N4T6	14.3	11.3	4.0	0.8	4.0	**0.5**	4.2	**0.5**	1.8	**0.5**
N3T7	19.0	15.9	7.0	2.7	4.8	**2.0**	7.3	2.5	2.9	**2.0**
N2T8	22.4	17.5	18.1	12	12.9	**6.1**	16.9	11.9	8.7	**5.7**

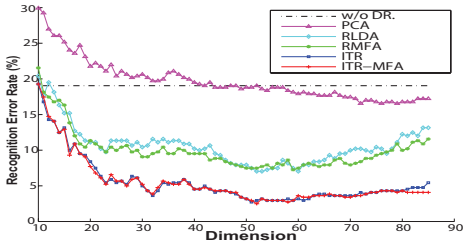


Figure 3. Recognition error rates over different dimensions. The configuration is N3T7 on the CMU PIE database. For LDA and MFA, the dimension of the preprocessing PCA step is N-Nc.

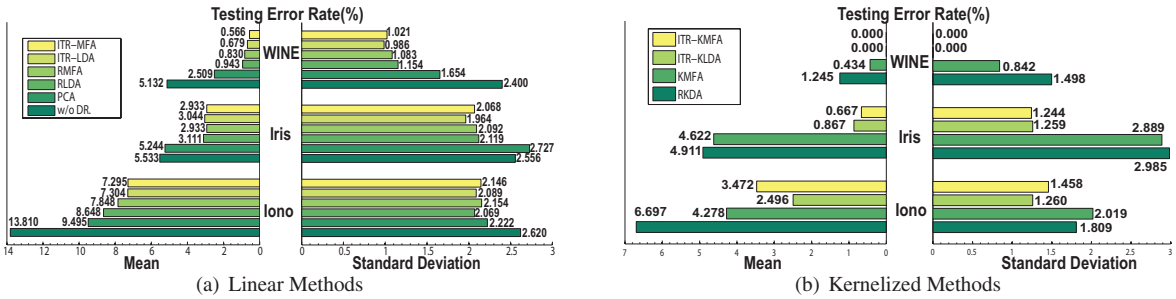
face algorithm [1] for LDA; the other is (N-1), and in this case, LDA and MFA are implemented by first transforming the objective functions into the form of Eqn. (11) for avoiding the singular value issue. With our ITR algorithm, we also test these two dimensions for PCA before formally performing ITR, and the maximum iteration number T_{max} in Algorithm 3 is set to 16 in all the experiments. For comparison, the classification result from PCA and that on the original gray-level features without dimensionality reduction are also reported as the baselines, denoted as 'PCA' and 'w/o DR.' in the tables and figures. In all experiments, the Nearest Neighbor method is used for final classification. For each database, we test various configurations of training and testing sets, where $'NxTy'$ indicates that x images of each subject are randomly selected for model training and the remaining y images of each subject are used for testing. Detailed results are listed in Table 1, and recognition error rates over different feature dimensions for the experiment $N3T7$ on the CMU PIE database are displayed in Figure 3. From the results, we can reach the conclusion that the trace ratio formulation generally outperforms the corresponding ratio trace formulation in terms of classification capability of the derived low-dimensional feature space, with only one exception for $N2T8$ of RLDA2 in the ORL database.

We also conduct classification experiments on the UCI databases with features of low dimensions. The sample data are normalized such that each feature has a standard deviation of one, and no PCA step is used for preprocessing before LDA/MFA since the feature dimension is already relatively small for these data sets. For LDA/MFA related algorithms, the best result of all possible LDA/MFA feature dimensions is reported. We randomly split each data set 100 times into training (70%) and testing (30%) sets, and the classification errors (mean and standard deviation of the testing error) are charted in Figure 4. These results again validate the superiority of the trace ratio formulation over the ratio trace formulation.

6.5. Classification by Kernel Trace Ratio algorithms with General Constraints

We also evaluate the effectiveness of the ITR procedure for solving the trace ratio problem with general constraints. Kernel-based LDA and MFA are used as examples for the trace ratio and ratio trace formulations. In all the experiments, the Gaussian Kernel $exp\{-\|x-y\|^2/\delta^2\}$ is used, and parameter δ is set as $\delta = 2^{(n-10)/2.5}\delta_0$, $n = 0, 1, \ldots, 20$, where δ_0 is the standard derivation of the training data set. The reported result is the best one among the 21 configurations. The three face databases and the three UCI databases are used for experiments. Detailed results are listed in Table 2 and Figure 4. They show that for both kernel-based Discriminant Analysis and kernel-based MFA algorithms, the trace ratio based solutions are consistently superior to the ratio trace based solutions.

Discussion: Over the past few decades, many algorithms have been proposed for dimensionality reduction [12][13]; even for just the PCA+LDA/MFA paradigm, numerous different procedures have been proposed on how to select the PCA dimension, and there are two parameters in MFA that



(a) Linear Methods (b) Kernelized Methods

Figure 4. Testing classification errors on three UCI databases for both linear and kernel-based algorithms. Results are obtained from 100 realizations of randomly generated 70/30 splits of data.

Table 2. Recognition error rates (%) of Ratio Trace + Kernel Discriminant Analysis (RKDA), Trace Ratio + Kernel Discriminant Analysis (ITR-KDA), Ratio Trace + Kernel MFA (RKMFA), and Trace Ratio + Kernel MFA (ITR-KMFA) on the three face databases.

FERET	RKDA	ITR-KDA	KMFA	ITR-KMFA
N3T3	8.0	6.3	8.0	**4.8**
N2T4	26.4	**17.5**	26.4	**17.5**
ORL	RKDA	ITR-KDA	KMFA	ITR-KMFA
N4T6	6.7	6.7	6.7	**5.0**
N3T7	14.3	**12.1**	14.3	**12.1**
N2T8	26.6	22.2	26.6	**21.6**
PIE	RKDA	ITR-KDA	KMFA	ITR-KMFA
N4T6	2.4	1.3	2.4	**1.1**
N3T7	3.4	**2.7**	3.4	**2.7**
N2T8	11.9	7.7	11.9	**6.9**

can be tuned for better performance. In this work, we did not try to evaluate all the algorithms to determine which one is best; instead, we claim that for each algorithm, solutions based on the trace ratio formulation are better than those from the corresponding ratio trace versions. For the PCA+LDA/MFA paradigm, when the PCA dimension is fixed, this benefit of the trace ratio formulation can also be gained in the LDA/MFA step.

7. Conclusion

In this paper, an efficient iterative procedure (ITR) has been proposed to directly solve the trace ratio optimization problem. The convergence of the projection matrix and the global optimality of the trace ratio value were proven. ITR truly provides the optimal solution for the joint objectives of most popular dimensionality reduction algorithms.The superiority of solutions from ITR over those from the ratio trace formulation has been extensively verified by large number of experiments on various data sets.

8. Acknowledgement

The work described in this paper was funded in part by the U.S. Government VACE program and in part by grants from the Research Grants Council of the Hong Kong Special Administrative Region.

References

[1] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997. 1, 2, 6, 7

[2] H. Chen, H. Chang and T. Liu, *Local discriminant embedding and its variants*, Proc. IEEE Computer Vision and Pattern Recognition, vol. 2, pp. 846–852, 2005. 2

[3] K. Fukunaga. Introduction to statistical pattern recognition. *Academic Press, second edition*, 1991. 1, 2

[4] Y. Guo, S. Li, J. Yang, T. Shu, and L. Wu. A generalized Foley-Sammon transform based on generalized fisher discriminant criterion and its application to face recognition. *Pattern Recognition Letter*, 24:147–158, 2003. 4, 5

[5] W. Hogan. Point-to-set maps in mathematical programming. *SIAM Rev.*, 15(3):591–603, 1973. 4

[6] R. Horn and C. Johnson. Matrix analysis. *Cambridge University Press, New York*, 1985. 3, 4

[7] R. Meyer. Sufficient conditions for the convergence of monotonic mathematical programming algorithms. *J. Comp. Sys. Sci.*, 12:108–121, 1976. 4

[8] M. Turk and A. Pentland. Face recognition using eigenfaces. *Proceedings of Conference on Computer Vision and Pattern Recognition*, pages 586–591, 1991. 1, 2, 3

[9] S. Yan and X. Tang. Trace Quotient Problems Revisited. *Proceedings of the European Conference on Computer Vision*, v. 2, pp. 232-244, 2006. 5

[10] S. Yan, D. Xu, B. Zhang, and H. Zhang. Graph embedding: A general framework for dimensionality reduction. *Proceedings of Conference on Computer Vision and Pattern Recognition*, pp. 830–837, 2005. 1, 2, 5

[11] X. Wang and X. Tang. Dual-space linear discriminant analysis for face recognition. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 564-569, June 2004. 2

[12] X. Wang and X. Tang. A unified framework for subspace face recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1222-1228, Sep. 2004. 7

[13] X. Wang and X. Tang. Random sampling for subspace face recognition. *International Journal of Computer Vision*, Vol. 70, No. 1, pp. 91-104, Oct. 2006. 7