Marginal Semi-Supervised Sub-Manifold Projections with Informative Constraints for Dimensionality Reduction and Recognition

Zhao Zhang*, Mingbo Zhao, and Tommy W. S. Chow
Department of Electronic Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong
*Correspondence author: E-mail: itzzhang@ee.cityu.edu.hk

Abstract— In this work, sub-manifold projections based semi-supervised dimensionality reduction (DR) problem learning from partial constrained data is discussed. Two semi-supervised DR algorithms termed Marginal Semi-Supervised Sub-Manifold Projections (MS^3MP) and orthogonal MS^3MP (OMS^3MP) are proposed. MS^3MP in singular case is also discussed. We also present the weighted least squares view of MS^3MP. Based on specifying the types of neighborhoods with pairwise constraints (PC) and the defined manifold scatters, our methods can preserve the local properties of all points and discriminant structures embedded in the localized PC. The sub-manifolds of different classes can also be separated. In PC guided methods, exploring selecting the informative constraints is challenging and random constraint subsets significantly affect the performance of algorithms. This paper also introduces an effective technique to select the informative constraints for DR with consistent constraints. The analytic form of the projection axes can be obtained by eigen-decomposition. The connections between this work and other related work are also elaborated. The validity of the proposed constraint selection approach and DR algorithms are evaluated by benchmark problems. Extensive simulations show that our algorithms can deliver promising results over some widely used state-of-the-art semi-supervised DR techniques.

Index Terms — semi-supervised learning, marginal projections, dimensionality reduction, informative constraints, image recognition

1 Introduction

In the neural networks and machine learning communities, variable sets that contain high-dimensional attributes accompanied with a lot of redundancies are often handled in many emerging applications, so extracting the most informative attributes that hold all required information by transforming the observations in the high-dimensional space into their lower-dimensions with certain local or global characteristics of data preserved is important [3][4] [43]. Note that this issue can benefit from dimensionality reduction (DR) that is an important preprocessing step before classification and recognition. Broadly speaking, DR methods can be categorized into linear and non-linear ones. The most representative linear DR techniques are the unsupervised *Principal Component Analysis* (PCA) [5] and *Locality Preserving Projections* (LPP) [7], supervised *Linear Discriminant Analysis* (LDA) [5] and *Maximum Margin Criterion* (MMC) [6]. PCA, LDA and MMC preserve the global structures of data, whilst LPP considers the local geometry of data. One of the popular nonlinear DR approaches is neural networks [4][29], for instance *Hinton et al.* [4] used the multi-layer identity mapping neural networks for DR, but the pre-training approach to choose the best initial weights is too complicated [29]. This work mainly discusses the linear feature extraction methods to reduce the data dimensionality for feature representation.

Note that labeled data are time-consuming and expensive to obtain in reality, but unlabeled ones can often be easily achieved with low expense from the real world [1][2][41][42]. It is also worth noting that semi-supervised methods learning from partially labeled data tend to outperform the ones that use either labeled or unlabeled data [1]. So, it is great advantageous to develop semi-supervised techniques for DR in practice. Specifically, locality preserving semi-supervised DR settings have been attracting more attention, e.g., *Semi-supervised Discriminant*

Analysis (SDA) [9], *Semi-Supervised LDA* (SSLDA) [10], *SEmi-supervised Local Fisher discriminant analysis* (SELF) [2] and *Semi-Supervised MMC* (SSMMC) [10], which were recently proposed by enabling the inclusion of class labels directly. A similar work to SDA was done in [18], which performs SDA under a trace ratio criterion [17]. These methods have incorporated the locality preserving power into the discriminant frameworks, i.e., they keep the local geometry and discriminant structures of data, but only unlabeled or labeled parts in their settings are localized. Another related work is called *Semi-Supervised Sub-Manifold Preserving Embedding* (S^3MPE) [23] that aims at separating each sub-manifold of each class. Differently, local neighborhood information is incorporated into both unlabeled and labeled parts of the criterion. S^3MPE provides a soft measure on the graph weights, aided by the heat kernel [3], but finding optimal kernel width has never been easy and straightforward. Also, S^3MPE is relatively rigid in defining the neighbors of data, so it maybe inefficient for highly sparse datasets.

In addition to the class labels, pairwise must-link (ML) and cannot-link (CL) constraints can also be applied to reflect supervised information of samples. Pairwise constraints (PC) were widely used in many areas, e.g., feature selection [13][14], constrained projections focused DR problems [8][12][24][26] and constrained clustering [15]. Compared with the class labels, PC exhibits two advantages. First, PC can sometimes be achieved by using minimal human effort and can provide more supervision information for fixed labeled number. Second, class label set of labeled samples is unchangeable, whilst PC sets derived from the labeled data are flexible. That is, we can employ either partial or all available constraints. Note that the number of labeled samples is typically small in the semi-supervised settings, so it is great advantageous to apply PC to guide the semi-supervised learning. It should be noted that many previous studies indicate that different PC subsets have significant effects on the performance of algorithms and informative constraints will be useful in improving the algorithms [13][14][15]. But up to date extracting the most informative constraints for learning still remains an open problem. In almost all previous PC guided studies, random constraint subsets are selected from the pairwise constraint pool. The final results are then averaged over repeating many times selections, but large deviations over runs are often produced [14]. To mitigate this problem, Bagging Constraints Score (BCS) algorithm [12] was recently proposed by constructing individual components with different constraint subsets randomly generated from the constraint pool instead of seeking one appropriate constraint subset for optimizations. Obviously, BCS will be computationally expensive for large-scale datasets and is incapable of solving this issue thoroughly.

In this work, we also discuss the constrained projections based semi-supervised DR problems. The followings highlight the major contributions of the paper. Firstly, two effective linear marginal semi-supervised sub-manifold projection algorithms termed MS^3MP and its orthogonal version, OMS^3MP, are proposed for DR. The presented linear methods can be directly used to embed new samples. We also discuss the case such that MS^3MP in the null space of the manifold scatter. Kernelized extensions of our algorithms are also addressed. In most previous PC derived methods, all sample pairs constrained by ML and CL are equally treated without considering the local property of data. We in this paper incorporate the local geometrical information of data into the PC definition and construct the PC sets from the constrained neighborhood graph [26]. That is, local PC is used to specify whether neighbors are in the same class or different. This idea was originally introduced in the *Constrained large Margin Local Projection* (CMLP) criterion [26]. Note that CMLP is supervised, so it may be overfitted to the small number of labeled data. Motivated by [23], a total manifold scatter is defined over all training data in addition to considering the ML and CL constrained within- and between-manifolds. So, the presented algorithms are the adaptation of CMLP to the general semi-supervised cases. Secondly, we present the weighted least squares view of MS^3MP. Specifically, MS^3MP is theoretically formulated as a weighted least squares problem by constructing a specific class indictor matrix. Thirdly, a valid and efficient constraint selection technique that can extract the most informative constraint subset is presented to boost the performance of existing and our techniques with consistent constraints. In order to select the informative constraints from the constraint pool, we take the underlying class distributions within the ML and CL constraint sets into account. As a result, our methods can deliver enhanced intra-cluster compactness and separate inter-class sub-manifolds. Benchmark simulations show promising results are exhibited by our methods with only small proportion of informative constraints.

We outline the paper as follows. In Section 2, we briefly review CMLP. In Section 3, we present an informative constraint selection method. Section 4 details the proposed projection criteria mathematically. Section 5 discusses

the equivalence between weighted least squares and our proposed method. Section 6 discusses the connections between this work and the related work. We in Section 7 conduct simulations to evaluate our techniques using benchmark problems. Finally, the paper in concluded in Section 8.

2 Constrained large Margin Local Projection (CMLP)

2.1 Construction of Graph-based Pairwise Constraints

In CMLP, the *ML* and *CL* constraint sets are constructed from a neighborhood graph-induced approach [26]. For a labeled set $X_L = [x_1,...,x_l] \in \mathbb{R}^{n\times l}$ belonging to c classes, a data graph $G = (V,E)$ with l vertices $\{x_i\}_{i=1}^{l}$ can be easily achieved. Nonzero weights are put on the edges $e(x_i,x_j) \in E$ connecting vertices x_i and x_j if $x_j \in N_+^{(x_i)}$ or $x_i \in N_+^{(x_j)}$, where $x_j \in N_+^{(x_i)}$ denotes the k nearest neighbor set of the vertex x_i. Let $e(x_i,x_j)=1$, if $x_j \in N_+^{(x_i)}$ or $x_i \in N_+^{(x_j)}$, $l(x_j)=l(x_i)$; $e(x_i,x_j)=-1$, if $x_j \in N_+^{(x_i)}$ or $x_i \in N_+^{(x_j)}$, $l(x_j)\neq l(x_i)$; $e(x_i,x_j)=0$, otherwise if $x_j \notin N_+^{(x_i)}$ and $x_i \notin N_+^{(x_j)}$, where $l(x_i)$ is the label of x_i. Then, a neighborhood graph $\tilde{G}_N = (\tilde{V}_N, \tilde{E}_N)$ can be constructed by removing the edges with zero weights. $\tilde{G}_N$ is constructed to define the similarity between vertex pair, where the similarity is measured by $e_N(x_i,x_j) \in \tilde{E}_N$. Then the *ML* and *CL* constraint sets are defined as

$$ML = \left\{(x_i,x_j) \mid e_N(x_i,x_j)=1, v(x_i)\in\tilde{V}_N, v(x_j)\in\tilde{V}_N, l(x_j)=l(x_i)\right\}, \tag{1a}$$

$$CL = \left\{(x_i,x_j) \mid e_N(x_i,x_j)=-1, v(x_i)\in\tilde{V}_N, v(x_j)\in\tilde{V}_N, l(x_j)\neq l(x_i)\right\}, \tag{1b}$$

where $v(x_i)$ is vertex x_i in $\tilde{G}_N$. Based on the constrained neighborhood graph, two *ML*-graph and *CL*-graph graphs can be achieved from $\tilde{G}_N$. In particular, *ML*-graph is defined to be the graph by removing the edges with negative weights and produced isolated vertices from $\tilde{G}_N$, whilst *CL*-graph is defined to be the graph by cutting the edges with positive weights and those produced isolated vertices from $\tilde{G}_N$. To learn the marginal projections, all edge lengths in *ML*-graph are minimized, while all edge lengths in *CL*-graph are maximized. As a result, more separated embeddings of inter-class neighbors can be exhibited, and the natural clusters within each class can be preserved. Note that *ML*- and *CL*-graph can be generalized to the have the same vertex number as $\tilde{G}_N$ when each vertex has at least one intra-class neighbor and one inter-class neighbor and all available constraints are used [26].

2.2 The CMLP Algorithm

The idea of the CMLP algorithm [26] is addressed as follows. Denote by adjacency matrices $\tilde{W}^{(ML)}$ and $\tilde{W}^{(CL)}$ to reflect the proximity relations of vertices in the *ML*-graph and *CL*-graph, where matrices $\tilde{W}^{(ML)}$ and $\tilde{W}^{(CL)}$ are of dimensions $N_{ML}\times N_{ML}$ and $N_{CL}\times N_{CL}$, respectively. Denote by $X^{(ML)}$ and $X^{(CL)}$ the ML and CL constrained data matrices, where $X^{(ML)} \in \mathbb{R}^{n\times N_{ML}}$ and $X^{(CL)} \in \mathbb{R}^{n\times N_{CL}}$ consist of the vertices in *ML*-graph and *CL*-graph, N_{ML} and N_{CL} are the vertex numbers in the *ML*-graph and *CL*-graph, respectively. Let $\tilde{L}^{(ML)} = \tilde{D}^{(ML)} - \tilde{W}^{(ML)}$ and $\tilde{L}^{(CL)} = \tilde{D}^{(CL)} - \tilde{W}^{(CL)}$ denote the graph Laplacian matrices, the ML and CL constrained local within- and between-manifold scatter matrices $\widetilde{S_{ML}}$ and $\widetilde{S_{CL}}$ are defined as

$$\widetilde{S_{ML}} = \frac{1}{2}\sum_{(x_i,x_j)\in ML}(x_i-x_j)(x_i-x_j)^{\mathrm{T}}\tilde{W}_{i,j}^{(ML)} = X^{(ML)}\tilde{L}^{(ML)}X^{(ML)\mathrm{T}}, \widetilde{S_{CL}} = \frac{1}{2}\sum_{(x_i,x_j)\in CL}(x_i-x_j)(x_i-x_j)^{\mathrm{T}}\tilde{W}_{i,j}^{(CL)} = X^{(CL)}\tilde{L}^{(CL)}X^{(CL)\mathrm{T}}, \tag{2}$$

where notation $^{\mathrm{T}}$ denotes the transpose of a matrix, $\tilde{D}^{(ML)}$ and $\tilde{D}^{(CL)}$ are diagonal matrices with the entries being $\tilde{D}_{ii}^{(ML)} = \sum_j \tilde{W}_{i,j}^{(ML)}$ and $\tilde{D}_{ii}^{(CL)} = \sum_j \tilde{W}_{i,j}^{(CL)}$. Note that the bigger the value of $\tilde{D}_{ii}^{(ML)}$ (or, $\tilde{D}_{ii}^{(CL)}$) is, the more important the corresponding vertex is. The entries of the adjacency matrices $\tilde{W}^{(ML)}$ and $\tilde{W}^{(CL)}$ are weighted as

$$\begin{cases}\tilde{W}_{i,j}^{(ML)} = \tilde{W}_{j,i}^{(ML)} = 1, \textit{if data pair } (x_i,x_j)\in ML\\ \tilde{W}_{i,j}^{(ML)} = \tilde{W}_{j,i}^{(ML)} = 0, \textit{otherwise}\end{cases}, \begin{cases}\tilde{W}_{i,j}^{(CL)} = \tilde{W}_{j,i}^{(CL)} = 1, \textit{if data pair } (x_i,x_j)\in CL\\ \tilde{W}_{i,j}^{(CL)} = \tilde{W}_{j,i}^{(CL)} = 0, \textit{otherwise}\end{cases}. \tag{3}$$

That is, CMLP clearly considers the local information of data and discriminant structures embedded in the PC. In particular, CMLP aims at pushing ML constrained data pairs close and separating CL constrained pairs. Note that one usually has $N_{ML} \le l$ and $N_{CL} \le l$, especially $N_{ML} << l$ and $N_{CL} << l$ if only small proportions of constraints are used. So the matrix computational burden in computing the manifold scatter matrices $\widetilde{S}_{ML}$ and $\widetilde{S}_{CL}$ can be greatly reduced. Three are four solution schemes [26] are proposed to optimize CMLP. In this work, we consider the following two schemes, that is to compute the transformation matrix $P = [\psi_1, \psi_2, ..., \psi_d] \in \mathbb{R}^{n \times d}$ with the reduced dimension $d \le n$ by solving the following ratio trace (RT) and trace difference (TD) problems [17][18]:

$$P^* = \arg\max_{P \in \mathbb{R}^{n \times d}} Tr\left[\left(P^{\mathrm{T}} X^{(ML)} \widetilde{L}^{(ML)} X^{(ML)\mathrm{T}} P + \beta I \right)^{-1} P^{\mathrm{T}} X^{(CL)} \widetilde{L}^{(CL)} X^{(CL)\mathrm{T}} P \right], \tag{4}$$

$$P^* = \arg\max_{P \in \mathbb{R}^{n \times d}, P^{\mathrm{T}} P = I} Tr\left(P^{\mathrm{T}} X^{(CL)} \widetilde{L}^{(CL)} X^{(CL)\mathrm{T}} P - \alpha P^{\mathrm{T}} X^{(ML)} \widetilde{L}^{(ML)} X^{(ML)\mathrm{T}} P \right), \tag{5}$$

where A^{-1} is matrix inverse of A, $Tr(\bullet)$ is trace operator, α is a control parameter for balancing the tradeoff between the scatters $\widetilde{S}_{CL}$ and $\widetilde{S}_{ML}$, and β is a regularization factor. Thus the solution P can be calculated by eigen-decomposition. After P is obtained, large margin projections can be conducted. Extensive benchmark problems verified the effectiveness of CMLP for image representation, visualization and recognition.

3 Determining the Informative Constraints for Learning

In this section, we propose an approach of extracting the informative constraints for learning the projections with consistent constraints. We take the underlying class distributions of constrained neighbors into account and select the informative constraints from the perspective of paired distance metrics. Let $d_{i,j}^{ML}$ and $d_{i,j}^{CL}$ denote the Euclidean distance between data pairs constrained by *ML* and *CL* respectively, namely

$$d_{i,j}^{ML} = d(x_i, x_j), \text{ if data pair } (x_i, x_j) \in ML, \quad d_{i,j}^{CL} = d(x_i, x_j), \text{ if data pair } (x_i, x_j) \in CL. \tag{6}$$

The guideline of our informative constraint selection approach can be described as the followings. Denote two ML and CL constrained index sets by

$$\text{ML\_}Index = \left[\left\{ \text{Ind}(v_a), \text{Ind}(v_b), d_{a,b}^{ML}, l(v_a) = l(v_b) \right\} \mid (v_a, v_b) \in ML \right] \tag{7}$$

$$\text{CL\_}Index = \left[\left\{ \text{Ind}(v_a), \text{Ind}(v_b), d_{a,b}^{CL}, l(v_a), l(v_b) \right\} \mid (v_a, v_b) \in CL \right] \tag{8}$$

where $\text{Ind}(v_a)$ and $\text{Ind}(v_b)$ are indexes of vertices v_a and v_b over the original data matrix X_L, respectively. For ML_*Index*, according to the class label $l(v_a)$ of v_a, one can easily partition the ML_*Index* set into *c* parts if available. More specifically, for each class *r*, *r* = 1, 2,…, *c*, an ML constrained index subset

$$\text{ML\_}r = \left[\left\{ \text{Ind}(v_a), \text{Ind}(v_b), d_{a,b}^{ML}, l(v_a) = r = l(v_b) \right\} \right] \tag{9}$$

can be efficiently achieved from the ML_*Index* set. We then process each ML_*r* by sorting $d_{a,b}^{ML}$ in descending order and updating the elements of ML_*r* according to the sorted $d_{a,b}^{ML}$. Similarly, the CL_*Index* set can also be divided into *c* parts when available. Specifically, according to the class label *r* of vertex v_a, an index subset

$$\text{CL\_}r = \left[\left\{ \text{Ind}(v_a), \text{Ind}(v_b), d_{a,b}^{CL}, l(v_a) = r, l(v_b) = t \ (r \ne t) \right\} \right] \tag{10}$$

can be easily achieved from the CL_*Index* set. We can analogously process each subset CL_*r* by sorting $d_{a,b}^{CL}$ in ascending order and updating the elements of CL_*r* according to the sorted $d_{a,b}^{CL}$. Then *c* sorted ML_*r* and CL_*r* subsets can be obtained. Note that we will handle each index subset ML_*r* and CL_*r* separately.

Based on the weighting methods in Eq.3, clearly for the constrained data pairs in ML_*r* and CL_*r*, the larger distance $d_{i,j}^{ML}$ is, the more priority of the corresponding ML constraint should be selected from each ML_*r* to be pushed close; on the contrary the smaller distance $d_{i,j}^{CL}$ is, the more priority of the corresponding CL constraint should be extracted from each CL_*r* to be separated. Note that constrained data pairs can be obtained according to

the stored vertex indexes. Denote by $q\%$ the selected proportion of the ML or CL constraints, $q\%$ contains the first $q\%$ constraints included in each subset ML_r or CL_r, $r = 1, 2,\ldots, c$, respectively when ML_r and CL_r are not empty. Note that the number of selected number of constraints from each ML_r or CL_r is computed by the MATLAB *ceil* function [22]. That is, if ML_r or CL_r has only a single ML or CL constraint, this constraint can be directly included in the selected constraint subset. It is also worth noting that based on the above operations, this informative constraint selection approach is computationally efficient. On the other hand, the ML constrained within-manifold and CL constrained between-manifold of similarity neighboring pairs in the training set can be effectively balanced. The validity of this method will be evaluated through simulations.

4 Marginal Semi-Supervised Sub-Manifold Projections (MS^{3}MP)

4.1 Motivation

Through effectively incorporating the local properties of data into the discriminant structures embedded in the PC, CMLP has achieved great improvements over some widely used supervised DR algorithms in real life benchmark problems, but CMLP may suffer from the following two underlying shortcomings. First, CMLP only applies the localized CL and ML constraints derived from the labeled data to learn marginal projections, but we cannot expect that there are always sufficient labeled data. So, when the number of labeled samples is limited, supervised CMLP algorithm tends to compute embedding spaces that are overfitted to the labeled data. Second, CMLP may be unstable. For datasets with some special distributions, the constructed ML-graph or CL-graph may be empty. For instance, if datasets are linearly separable and inter-class points are distributed as separate clusters, CL constraint set may be empty. In such cases, CMLP becomes ineffective. Thus these findings encourage us to make use of unlabeled samples that are often readily available from the real world to improve the criterion of CMLP. In next section, we will focus on addressing these issues to use unlabeled samples to enhance CMLP when the number of labeled data or used constraints is fewer. The following elaborates the details.

4.2 The Proposed MS^{3}MP Algorithm

This section presents a practical approach to boost the performance of CMLP. To make CMLP more general and stable, both the CL and ML constraints together with the unlabeled samples are simultaneously considered in the proposed algorithm. Denote by $X = [X_L, X_U] \in \mathbb{R}^{n\times N}$ the training set, where $X_L = [x_1,\ldots,x_l] \in \mathbb{R}^{n\times l}$ denotes a labeled set belonging to c classes and $X_U = [x_1,\ldots,x_u] \in \mathbb{R}^{n\times u}$ is an unlabeled set. That is, $l+u=N$. Denote by n_{ML} and n_{CL} the numbers of ML and CL constraints over X_L respectively, then the projection axes of MS^{3}MP can be obtained by solving the following optimization problems:

$$P^* = \arg\max_{P\in\mathbb{R}^{n\times d}} \frac{(n_{CL}/N)Tr\left(P^{\mathrm{T}}\widetilde{R_{CL}}P\right)-(\mu_1/N)Tr\left(P^{\mathrm{T}}\widetilde{S}_tP\right)}{(n_{ML}/N)Tr\left(P^{\mathrm{T}}\widetilde{R_{ML}}P\right)+(\mu_2/N)Tr\left(P^{\mathrm{T}}\widetilde{S}_tP\right)}, \tag{11}$$

$$P^* = \arg\max_{P\in\mathbb{R}^{n\times d},P^{\mathrm{T}}P=I} Tr\left(P^{\mathrm{T}}\left((n_{CL}/N)\widetilde{R_{CL}}-(n_{ML}/N)\widetilde{R_{ML}}-(\mu_3/N)\widetilde{S}_t\right)P\right), \tag{12}$$

where $\widetilde{R_{ML}}$ and $\widetilde{R_{CL}}$ denote the within-manifold and between-manifold scatters respectively, $\widetilde{S}_t = XLX^{\mathrm{T}}$ and $L = D^{-1/2}(D-W)D^{-1/2}$ is the normalized graph Laplacian [7] defined for preserving the total manifold structures of all samples including constrained and unlabeled samples, where unlabeled samples means the data that have no class labels and are not involved in any constraint. The adjacency matrix W reflects the local information of data and is defined as $W_{i,j} = W_{j,i} = 1$ if $x_j \in N_+^{(x_i)}$ or $x_i \in N_+^{(x_j)}$ when the simple-minded method [3] is used, and D is a diagonal matrix with entries being $D_{ii} = \sum_j W_{i,j}$. To balance the contributions of the local scatters $\widetilde{R_{ML}}$ and $\widetilde{R_{CL}}$, two parameters n_{CL}/N and n_{ML}/N are added. Note that parameters n_{CL}/N and n_{ML}/N are self-tunable when different proportions of constraints are employed. The parameters μ_1, μ_2 and μ_3 are set to constant 1 if without

special remarks. The intuition behind Eqs.7 and 8 is that minimizing $P^{\mathrm{T}}\widetilde{R_{ML}}P$ is equivalent to keeping the ML constrained neighbor pairs close and maximizing $P^{\mathrm{T}}\widetilde{R_{CL}}P$ is equivalent to keeping the CL constrained neighbors apart in the reduced space. At the same time, the manifold structures of the training set are preserved. It is worth noting that MS^{3}MP does not suffer from the problems of CMLP and is more general and stable. In Eqs.11 and 12, the scatter matrices $\widetilde{R_{ML}}$ and $\widetilde{R_{CL}}$ are defined as

$$\widetilde{R_{ML}} = X^{(ML)}\tilde{Q}^{(ML)}X^{(ML)\mathrm{T}}, \widetilde{R_{CL}} = X^{(CL)}\tilde{Q}^{(CL)}X^{(CL)\mathrm{T}}, \tag{13}$$

where $\tilde{Q}^{(ML)} = \left(\tilde{D}^{(ML)}\right)^{-1/2}\left(\tilde{D}^{(ML)} - \widetilde{W}^{(ML)}\right)\left(\tilde{D}^{(ML)}\right)^{-1/2}$ and $\tilde{Q}^{(CL)} = \left(\tilde{D}^{(CL)}\right)^{-1/2}\left(\tilde{D}^{(CL)} - \widetilde{W}^{(CL)}\right)\left(\tilde{D}^{(CL)}\right)^{-1/2}$ are normalized graph Laplacian, and the weight matrices $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$ are defined as

$$\begin{cases} \widetilde{W}_{i,j}^{(ML)} = \widetilde{W}_{j,i}^{(ML)} = N, if\ data\ pair\left(x_i, x_j\right) \in ML \\ \widetilde{W}_{i,j}^{(ML)} = \widetilde{W}_{j,i}^{(ML)} = 1, if\ \left(x_i, x_j\right) \notin ML, l\left(x_i\right) = l\left(x_j\right), \\ \widetilde{W}_{i,j}^{(ML)} = \widetilde{W}_{j,i}^{(ML)} = 0, otherwise \end{cases} \begin{cases} \widetilde{W}_{i,j}^{(CL)} = \widetilde{W}_{j,i}^{(CL)} = N, if\ data\ pair\left(x_i, x_j\right) \in CL \\ \widetilde{W}_{i,j}^{(CL)} = \widetilde{W}_{j,i}^{(CL)} = 1, if\ \left(x_i, x_j\right) \notin CL, l\left(x_i\right) \neq l\left(x_j\right) \\ \widetilde{W}_{i,j}^{(CL)} = \widetilde{W}_{j,i}^{(CL)} = 0, otherwise \end{cases}. \tag{14}$$

That is, more attention will be paid to the ML and CL constrained neighbors through imposing heavy weights, compared with weighting the non-neighbors of the same class and non-neighbor pairs of different classes. Note that if ML-graph or CL-graph is empty, the adjacency matrix $\widetilde{W}^{(ML)}$ or $\widetilde{W}^{(CL)}$ will be defined based on the global PC [8][12]. Note that this weighting method can further help tackle the drawbacks of CMLP when the constructed ML-graph or CL-graph is empty, as enhanced intra-class compactness and inter-class separation can be obtained even if ML- or CL-graph is empty. Moreover, based on the weighting method of Eq.14, the intrinsic multimodal structures for multimodal distributions can still be kept as long as a proper neighborhood size k is set. Also the intrinsic sub-manifolds of different classes can be effectively separated based on our presented constraint selection approach and weighting method. Because the manifold scatters $\widetilde{R_{ML}}$ and $\widetilde{S}_t$ are positive semi-definite, Eq.11 is solved by assuming that $(n_{ML}/N)\widetilde{R_{ML}} + (\mu_2/N)\tilde{S}_t$ is nonsingular. Otherwise, if P lies in the null space of the matrix $(n_{ML}/N)\widetilde{R_{ML}} + (\mu_2/N)\tilde{S}_t$, $P^{\mathrm{T}}\left((n_{ML}/N)\widetilde{R_{ML}} + (\mu_2/N)\tilde{S}_t\right)P = 0$, leading to the so-called singularity problem [17]. It is worth noting that this case can be often encountered if the null space of has larger dimensionality d' than the reduced dimension d [17]. To mitigate this issue, a regularized term βI with a small positive number β is added to the optimization problem when solving the following criterion from Eq.11 for the projection matrix:

$$P^* = \arg\max_{P\in\mathbb{R}^{n\times d}} \frac{Tr\left(P^{\mathrm{T}}\left((n_{CL}/N)\widetilde{R_{CL}} - (\mu_1/N)\widetilde{S_t}\right)P\right)}{Tr\left(P^{\mathrm{T}}\left((n_{ML}/N)\widetilde{R_{ML}} + (\mu_2/N)\widetilde{S_t}\right)P\right)}, S.t.\ P^{\mathrm{T}}\left((n_{ML}/N)\widetilde{R_{ML}} + (\mu_2/N)\widetilde{S_t}\right)P = I. \tag{15}$$

Thus the projection axes of the proposed SSMP algorithm can be obtained as eigenvectors $\{\psi_r\}_{r=1}^{d}$ according to the leading d eigenvalues $\{\lambda_r\}_{r=1}^{d}$ of the following eigen-equations:

$$\left((n_{ML}/N)\widetilde{R_{ML}} + (1/N)\widetilde{S_t} + \beta I\right)^{-1}\left((n_{CL}/N)\widetilde{R_{CL}} - (1/N)\widetilde{S_t}\right)\psi_j^* = \lambda_j^*\psi_j^*, \tag{16}$$

$$\left((n_{CL}/N)\widetilde{R_{CL}} - (n_{ML}/N)\widetilde{R_{ML}} - (1/N)\widetilde{S_t}\right)\psi_j^* = \lambda_j^*\psi_j^*. \tag{17}$$

The solution to the problem in Eq.16 corresponds to optimization criterion in Eq.15. Note that eigenvectors $\{\psi_r\}_{r=1}^{d}$ computed from Eq.17 are orthogonal with each other, thus similarity between points can be effectively kept without any change if it is based on Euclidian distance [17][18]. We refer to this scheme of obtaining orthogonal axes as orthogonal MS^{3}MP (OMS^{3}MP). Another scheme is still called MS^{3}MP. Note that the work described in this paper was partially presented in [16]. When the projection matrix $P \in \mathbb{R}^{n\times d}$ is obtained from both labeled and unlabeled samples, the output of the linear DR process can be performed as $Y = P^{\mathrm{T}}X$, where $Y \in \mathbb{R}^{d\times N}$ denotes the low-dimensional representation of the original training set X. Clearly, our methods can do induction embedding new points. And the subsequent classification or clustering tasks can then be effectively performed based on the computed low-dimensional embeddings.

4.3 MS^{3}MP in Singular Case

We here discuss MS^{3}MP in singular case, that is scatter matrices $\widetilde{S}_t$ and $\widetilde{R}_{CL}$ are maximized in the null space of $(n_{ML}/N)\widetilde{R}_{ML}+(1/N)\widetilde{S}_t$. This discussion is motivated by the null space LDA (NLDA) [19]. The basic idea behind NLDA is the null space of LDA intra-class scatter S_w may contain significant discriminative information if the projection matrix of LDA inter-class scatter matrix S_b is not zero in those directions [19]. As a result, the singularity problem can be implicitly addressed. We refer to MS^{3}MP in singular case as null space MS^{3}MP. The projection matrix P of null space MS^{3}MP can be computed by solving the following optimization problem:

$$P^* = \arg \max_{P^{\mathrm{T}}\left((n_{ML}/N)\widetilde{R}_{ML}+(1/N)\widetilde{S}_t\right)P=0} (n_{CL}/N)Tr\left(P^{\mathrm{T}}\widetilde{R}_{CL}P\right)-(1/N)Tr\left(P^{\mathrm{T}}\widetilde{S}_tP\right). \tag{18}$$

Let $\widetilde{R}_{ML}=U\Sigma V^{\mathrm{T}}$ denote the singular value decomposition (SVD) of $(n_{ML}/N)\widetilde{R}_{ML}+(1/N)\widetilde{S}_t$, then U can be portioned into two parts $U_1\in\mathbb{R}^{n\times d'}$ and $U_2\in\mathbb{R}^{n\times(n-d')}$, consisting of the eigenvectors corresponding to the zero and positive eigenvalues of matrix $(n_{ML}/N)\widetilde{R}_{ML}+(1/N)\widetilde{S}_t$ respectively. By projecting scatters S_t and $\widetilde{R}_{CL}$ onto the subspace spanned by the columns of U_1, we can obtain

$$\widehat{S}_t=U_1^{\mathrm{T}}\widetilde{S}_tU_1,\ \widehat{R_{CL}}=U_1^{\mathrm{T}}\widetilde{R_{CL}}U_1,\ \widehat{R_{ML}}=U_1^{\mathrm{T}}\widetilde{R_{ML}}U_1. \tag{19}$$

With the computed U_1, the optimal projection transformation of MS^{3}MP can be addressed by $P=U_1Q$, where orthogonal matrix $Q\in\mathbb{R}^{d'\times d}$ is obtained from solving the following optimization problem:

$$Q^* = \arg\max_{Q^{\mathrm{T}}Q=I} (n_{CL}/N)Tr\left(Q^{\mathrm{T}}\widehat{R_{CL}}Q\right)-(1/N)Tr\left(Q^{\mathrm{T}}\widehat{S}_tQ\right). \tag{20}$$

Clearly, the delivered projection matrix $P=U_1Q$ by null space MS^{3}MP is orthogonal, therefore the similarity between each data point can be effectively preserved and thus stable for the subsequent classification task.

4.4 Kernelized Extensions for Non-Linear Dimensionality Reduction

This section discusses the approach of kernelizing MS^{3}MP and OMS^{3}MP, which can be addressed by the standard kernel method [31]. Let ϕ denote the mapping from $\mathbb{R}^n$ to a higher-dimensional kernel space $\mathbb{Z}^p$ ($p\gg n$). This mapping can be implicitly defined by using a kernel function. More specifically, the (*i*,*j*)-th entry of a kernel matrix K is given by $K_{ij}=K(x_i,x_j)=\phi(x_i)^{\mathrm{T}}\phi(x_j)$. A typical choice of the kernel functions is Gaussian RBF kernel $K(x_i,x_j)=\exp\left(-\|x_i-x_j\|^2/2\sigma^2\right)$. Let $\phi(X)=\left[\phi(x_1),...,\phi(x_N)\right]$, $\phi\left(X^{(CL)}\right)=\left[\phi(x_1),...,\phi\left(x_{N_{CL}}\right)\right]$ and $\phi\left(X^{(ML)}\right)=\left[\phi(x_1),...,\phi\left(x_{N_{ML}}\right)\right]$, scatters $\left(P^{\mathrm{T}}\widetilde{R}_{ML}P\right)^{\phi}$, $\left(P^{\mathrm{T}}\widetilde{R}_{CL}P\right)^{\phi}$ and $\left(P^{\mathrm{T}}\widetilde{S}_tP\right)^{\phi}$ in $\mathbb{Z}^p$ can be written as

$$\left(P^{\mathrm{T}}\widetilde{R}_{ML}P\right)^{\phi}=P^{\phi\mathrm{T}}\phi\left(X^{(ML)}\right)\widetilde{Q}^{(ML)}\phi\left(X^{(ML)}\right)^{\mathrm{T}}P^{\phi},\ \left(P^{\mathrm{T}}\widetilde{R}_{CL}P\right)^{\phi}=P^{\phi\mathrm{T}}\phi\left(X^{(CL)}\right)\widetilde{Q}^{(CL)}\phi\left(X^{(CL)}\right)^{\mathrm{T}}P^{\phi}, \tag{21}$$

$$\left(P^{\mathrm{T}}\widetilde{S}_tP\right)^{\phi}=P^{\phi\mathrm{T}}\phi(X)D^{-1/2}LD^{-1/2}\phi(X)^{\mathrm{T}}P^{\phi}\ . \tag{22}$$

Note that in kernel space $\mathbb{Z}^p$, P^{ϕ} can be expressed using the mapped data as $P^{\phi}=\phi(X)E$ if there are vectors $E=[\eta_1,\eta_2,...,\eta_d]$ available. With kernel method, the above formulations can be converted to

$$\left(P^{\mathrm{T}}\widetilde{R_{ML}}P\right)^{\phi}=E^{\mathrm{T}}K^{(ML)}\widetilde{Q}^{(ML)}K^{(ML)\mathrm{T}}E,\left(P^{\mathrm{T}}\widetilde{R_{CL}}P\right)^{\phi}=E^{\mathrm{T}}K^{(CL)}\widetilde{Q}^{(CL)}K^{(CL)\mathrm{T}}E,\left(P^{\mathrm{T}}\widetilde{S}_tP\right)^{\phi}=E^{\mathrm{T}}K_XL^{\phi}K_XE, \tag{23}$$

where $L^{\phi}=D^{-1/2}LD^{-1/2}$, kernel matrices $K^{(ML)}=\phi(X)^{\mathrm{T}}\phi\left(X^{(ML)}\right)$, $K^{(CL)}=\phi(X)^{\mathrm{T}}\phi\left(X^{(CL)}\right)$ and $K_X=\phi(X)^{\mathrm{T}}\phi(X)$ are $N\times N$ dimensional kernel matrix defined over all samples, including labeled and unlabeled data. By substituting Eq.23 into Eqs.15 and 12, we can obtain the following problems for kernelized MS^{3}MP and OMS^{3}MP:

$$\begin{aligned}&\underset{E}{Max}\ Tr\left[E^{\mathrm{T}}\left((n_{CL}/N)K^{(CL)}\widetilde{Q}^{(CL)}K^{(CL)\mathrm{T}}-(1/N)K_XL^{\phi}K_X\right)E\right]\\&S.t.\ E^{\mathrm{T}}\left((n_{ML}/N)K^{(ML)}\widetilde{Q}^{(ML)}K^{(ML)\mathrm{T}}+(1/N)K_XL^{\phi}K_X+\beta K_X\right)E=I\end{aligned}, \tag{24}$$

$$\underset{E^{\mathrm{T}}K_X E=I}{Max}\ E^{\mathrm{T}}\left(\left(n_{CL}/N\right)K^{(CL)}\widetilde{Q}^{(CL)}K^{(CL)\mathrm{T}}-\left(n_{ML}/N\right)K^{(ML)}\widetilde{Q}^{(ML)}K^{(ML)\mathrm{T}}-\left(1/N\right)K_X L^{\phi}K_X\right)E\,. \tag{25}$$

Thus the nonlinear transforming basis vectors of kernelized MS^{3}MP and OMS^{3}MP can be respectively achieved as eigenvectors $\{\eta_r\}_{r=1}^{d}$ according to the leading d eigenvalues $\{\delta_r\}_{r=1}^{d}$ of the following eigen-equations:

$$\left(\left(n_{CL}/N\right)\widetilde{R}_{CL}^{\phi}-\left(1/N\right)\widetilde{S}_t^{\phi}\right)\eta_j^*=\delta_j^*\left(\left(n_{ML}/N\right)\widetilde{R}_{ML}^{\phi}+\left(1/N\right)\widetilde{S}_t^{\phi}+\beta K_X\right)\eta_j^*, \tag{26}$$

$$\left(\left(n_{CL}/N\right)\widetilde{R}_{CL}^{\phi}-\left(n_{ML}/N\right)\widetilde{R}_{ML}^{\phi}-\left(1/N\right)\widetilde{S}_t^{\phi}\right)\eta_j^*=\delta_j^*K_X\eta_j^*, \tag{27}$$

where $\widetilde{R}_{CL}^{\phi}=K^{(CL)}\widetilde{Q}^{(CL)}K^{(CL)\mathrm{T}}$, $\widetilde{R}_{ML}^{\phi}=K^{(ML)}\widetilde{Q}^{(ML)}K^{(ML)\mathrm{T}}$ and $\widetilde{S}_t^{\phi}=K_X L^{\phi}K_X$. When the nonlinear projection matrix E is obtained, nonlinear DR can be similarly conducted. Note that kernelized extensions depend on the number of samples rather than the dimensionality. As a result, kernelized methods can improve the computational efficiency when the sample size of the dataset is higher than its input dimensionality of dataset. But the performance of the kernelized extensions heavily depends on the kernel family, including the kernel function and kernel width. This is because different kernels tend to exhibit different mappings and properties [31]. It is also noted that to date there is still no effective theoretical guarantee about selecting the optimal kernel functions and parameters. In this work, we will mainly evaluate the proposed linear DR algorithms.

5 Relationship between MS^{3}MP and Weighted Least Squares

Least squares (LS) approach [11][32] is widely applied in many areas, including regression and classification. In this section, we mainly discuss the issues such that formulating the MS^{3}MP criterion in Eq.11 as a weighted least square (WLS) problem [37]. We first give the following definition.

Definition 1. When all samples in the training set are used to construct the constrained data matrices $X^{(CL)}$, $X^{(ML)}$ and adjacency matrices $\widetilde{W}^{(CL)}$, $\widetilde{W}^{(ML)}$, we can update $\widetilde{R}_{ML}$ and $\widetilde{R}_{CL}$ as

$$\widetilde{R}_{ML}=\frac{1}{2}\sum_{i,j=1}^{N}\left(x_i-x_j\right)\left(x_i-x_j\right)^{\mathrm{T}}\widetilde{W}_{i,j}^{(ML)}=X\tilde{L}^{(ML)}X^{\mathrm{T}},\ \widetilde{R}_{CL}=\frac{1}{2}\sum_{i,j=1}^{N}\left(x_i-x_j\right)\left(x_i-x_j\right)^{\mathrm{T}}\widetilde{W}_{i,j}^{(CL)}=X\tilde{L}^{(CL)}X^{\mathrm{T}} \tag{28}$$

if non-normalized graph Laplacian $\widetilde{S}_t$, $\tilde{L}^{(ML)}$ and $\tilde{L}^{(CL)}$ are applied, where $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$ are sizes of $N\times N$. Note that we weight the same sample pairs as in Eq.14, i.e. nonzero entries of $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$ are unchangeable and the local neighborhoods preserved by $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$ keep unchanged.

Based on Definition 1, one can easily convert the optimization problem of the MS^{3}MP criterion in Eq.11 into

$$P^*=\arg\max_{P\in\mathbb{R}^{n\times d}}\frac{\frac{1}{2}\sum_{i,j}Tr\left(\left(P^{\mathrm{T}}x_i-P^{\mathrm{T}}x_j\right)\left(P^{\mathrm{T}}x_i-P^{\mathrm{T}}x_j\right)^{\mathrm{T}}\right)\left(\left(n_{CL}/N\right)\widetilde{W}_{i,j}^{(CL)}-\left(1/N\right)W_{i,j}\right)}{\frac{1}{2}\sum_{i,j}Tr\left(\left(P^{\mathrm{T}}x_i-P^{\mathrm{T}}x_j\right)\left(P^{\mathrm{T}}x_i-P^{\mathrm{T}}x_j\right)^{\mathrm{T}}\right)\left(\left(n_{ML}/N\right)\widetilde{W}_{i,j}^{(ML)}+\left(1/N\right)W_{i,j}\right)}. \tag{29}$$

Let $\Xi_{i,j}^{(CL)}=\left(n_{CL}/N\right)\widetilde{W}_{i,j}^{(CL)}-\left(1/N\right)W_{i,j}$, $\Xi_{i,j}^{(ML)}=\left(n_{ML}/N\right)\widetilde{W}_{i,j}^{(ML)}+\left(1/N\right)W_{i,j}$, $\Delta_{ii}^{(CL)}=\sum_j\Xi_{i,j}^{(CL)}$ and $\Delta_{ii}^{(ML)}=\sum_j\Xi_{i,j}^{(ML)}$, one can easily transform the above problem to

$$P^*=\arg\max_{P\in\mathbb{R}^{n\times d}}\frac{Tr\left(P^{\mathrm{T}}X\left(\Delta^{(CL)}-\Xi^{(CL)}\right)X^{\mathrm{T}}P\right)}{Tr\left(P^{\mathrm{T}}X\left(\Delta^{(ML)}-\Xi^{(ML)}\right)X^{\mathrm{T}}P\right)}=\arg\max_{P\in\mathbb{R}^{n\times d}}\frac{Tr\left(P^{\mathrm{T}}X\Omega^{(CL)}X^{\mathrm{T}}P\right)}{Tr\left(P^{\mathrm{T}}X\Omega^{(ML)}X^{\mathrm{T}}P\right)}, \tag{30}$$

where $\Omega^{(CL)}$ and $\Omega^{(ML)}$ denote the graph Laplacian over $\Xi^{(CL)}$ and $\Xi^{(ML)}$, respectively. Based on similar trick to optimize the LPP criterion in [36], one can similarly convert the above problem further to

$$P^*=\arg\max_{P\in\mathbb{R}^{n\times d}}\frac{Tr\left(P^{\mathrm{T}}X\Omega^{(CL)}X^{\mathrm{T}}P+P^{\mathrm{T}}X\Xi^{(ML)}X^{\mathrm{T}}P\right)}{Tr\left(P^{\mathrm{T}}X\Delta^{(ML)}X^{\mathrm{T}}P\right)}. \tag{31}$$

Note that this problem can be viewed as a generalized eigen-problem solved by eigen-decomposition. Next, we first analyze the above problem using eigen-decomposition and then formulate it as a WLS problem.

5.1 Computing MS^{3}MP by Eigen-decomposition

The generalized eigen-problem according to Eq.31 can be defined as $Max_P\ Tr\left(P^{\mathrm{T}}X\Omega^{(CL)}X^{\mathrm{T}}P+P^{\mathrm{T}}X\Xi^{(ML)}X^{\mathrm{T}}P\right)$, $S.t.\ P^{\mathrm{T}}X\Delta^{(ML)}X^{\mathrm{T}}P=I$, so the projection axes of MS^{3}MP criterion are obtained as eigenvectors according to the leading d eigenvalues of matrix $\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{\dagger}\left(X\Omega^{(CL)}X^{\mathrm{T}}+X\Xi^{(ML)}X^{\mathrm{T}}\right)$, where notation † denotes pseudo-inverse. By performing SVD to the matrix $X\Delta^{(ML)}X^{\mathrm{T}}$, one can have

$$X\Delta^{(ML)}X^{\mathrm{T}}=\widetilde{U}\begin{pmatrix}\Sigma_t^2 & 0\\ 0 & 0\end{pmatrix}\widetilde{U}^{\mathrm{T}}, \tag{32}$$

where $\widetilde{U}$ is an orthogonal matrix, Σ_t^2 is a diagonal matrix. Let $\widetilde{U}=[\widetilde{U}_1,\widetilde{U}_2]$ be a partition of $\widetilde{U}$ such that $\widetilde{U}_1\in\mathbb{R}^{n\times t}$ and $\widetilde{U}_2\in\mathbb{R}^{n\times(n-t)}$, where t denotes the rank of $X\Delta^{(ML)}X^{\mathrm{T}}$ and $\widetilde{U}_2$ lies in the null space of $X\Delta^{(ML)}X^{\mathrm{T}}$, thus we have $\widetilde{U}_2^{\mathrm{T}}X\Delta^{(ML)}X^{\mathrm{T}}\widetilde{U}_2=0$. Let $\widetilde{L}_b=\Omega^{(CL)}+\Xi^{(ML)}$. Because matrix $\tilde{L}_b$ is symmetrical, it can be decomposed by performing the Cholesky Decomposition, that is $\widetilde{L}_b=\tilde{G}\tilde{G}^{\mathrm{T}}$, where $\tilde{G}$ is a lower triangular matrix. If we let $\widetilde{H}_b=X\tilde{G}$, then $X\widetilde{L}_bX^{\mathrm{T}}=\widetilde{H}_b\widetilde{H}_b^{\mathrm{T}}$. Denote $H=\Sigma_t^{-1}\widetilde{U}_1^{\mathrm{T}}\widetilde{H}_b$ and let $\Im=V\Sigma_bQ^{\mathrm{T}}$ be the *SVD* of the matrix *H*, where *V* and *Q* are orthogonal matrices, and Σ_b is a diagonal matrix with rank *q*, we can then obtain

$$\Sigma_t^{-1}\widetilde{U}_1^{\mathrm{T}}\widetilde{H}_b\widetilde{H}_b^{\mathrm{T}}\widetilde{U}_1\Sigma_t^{-1}=HH^{\mathrm{T}}=V\Sigma_b^2V^{\mathrm{T}}. \tag{33}$$

As a result, according to the formulations in Eqs.27 and 28, we can obtain the following equation:

$$\begin{aligned}&\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{\dagger}\left(X\widetilde{L}_bX^{\mathrm{T}}\right)\\&=\widetilde{U}\begin{pmatrix}\Sigma_t^{-1}\Sigma_t^{-1} & 0\\ 0 & 0\end{pmatrix}\widetilde{U}^{\mathrm{T}}\widetilde{H}_b\widetilde{H}_b^{\mathrm{T}}\widetilde{U}\begin{pmatrix}\Sigma_t^{-1}\Sigma_t & 0\\ 0 & 0\end{pmatrix}\widetilde{U}^{\mathrm{T}}\\&=\widetilde{U}\begin{pmatrix}\Sigma_t^{-1} & 0\\ 0 & 0\end{pmatrix}V\Sigma_b^2V^{\mathrm{T}}\begin{pmatrix}\Sigma_t & 0\\ 0 & 0\end{pmatrix}\widetilde{U}^{\mathrm{T}}=\widetilde{U}\begin{pmatrix}\Sigma_t^{-1}V & 0\\ 0 & I\end{pmatrix}\begin{pmatrix}\Sigma_b^2 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}V^{\mathrm{T}}\Sigma_t & 0\\ 0 & I\end{pmatrix}\widetilde{U}^{\mathrm{T}}\end{aligned}. \tag{34}$$

Then if we let $P^*_{MS^3MP}=\widetilde{U}_1\Sigma_t^{-1}V_q$, where V_q consists of the first *q* columns of the matrix *V* when only the first *q* diagonal elements of Σ_b are nonzero, we then have $\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{\dagger}\left(X\widetilde{L}_bX^{\mathrm{T}}\right)P^*_{MS^3MP}=\Sigma_b^2P^*_{MS^3MP}$, which implies that $P^*_{MS^3MP}$ will consist of the optimal transforming basis vectors of MS^{3}MP criterion.

5.2 Equivalence between WLS and MS^{3}MP

The definition of WLS is given as follows. For given a class indicator matrix Y and a diagonal matrix $\Delta^{(ML)}$ with entries $\Delta_{ii}^{(ML)}\in\mathbb{R}^+$, the objective function and solution of WLS can be described as

$$\underset{P}{Min}\,\Delta^{(ML)}\left\|Y^{\mathrm{T}}-X^{\mathrm{T}}P\right\|_F^2,\quad P^*_{WLS}=\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{\dagger}X\Delta^{(ML)}Y^{\mathrm{T}}, \tag{35}$$

where $\|\cdot\|_F$ is matrix Frobenius norm. Note that the class indicator matrix introduced in [32] can be used to define *Y*. Also, if we choose a class indicator matrix as $Y=\widetilde{G}\left(\Delta^{(ML)}\right)^{-1}$, we have $X\Delta^{(ML)}Y^{\mathrm{T}}=\widetilde{H}_b$. So the optimal solution in Eq.35 can be rewritten as $P^*_{WLS}=\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{\dagger}\widetilde{H}_b$, which can also be equivalent to

$$\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{\dagger}\widetilde{H}_b=\widetilde{U}\begin{pmatrix}\Sigma_t^{-1}\Sigma_t^{-1} & 0\\ 0 & 0\end{pmatrix}\widetilde{U}^{\mathrm{T}}\widetilde{H}_b=\widetilde{U}_1\Sigma_t^{-1}\left(\Sigma_t^{-1}\widetilde{U}_1^{\mathrm{T}}\widetilde{H}_b\right)=\widetilde{U}_1\Sigma_t^{-1}\Im=\widetilde{U}_1\Sigma_t^{-1}V\Sigma_bQ^{\mathrm{T}}=P^*_{MS^3MP}\Sigma_bQ^{\mathrm{T}}, \tag{36}$$

where Q is an orthogonal matrix, so it can be neglected if the similarity of two points is defined over Euclidean distance. As a result, the main difference between $P^*_{MS^3MP}$ and P^*_{WLS} is the diagonal matrix Σ_b. Supposing that Σ_b is an identity matrix, we have $P^*_{WLS}=P^*_{MS^3MP}$. This can only be hold when satisfying the following condition: $\mathrm{rank}\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)-\mathrm{rank}\left(X\widetilde{L}_bX^{\mathrm{T}}\right)=\mathrm{rank}\left(X\Delta^{(ML)}X^{\mathrm{T}}-X\widetilde{L}_bX^{\mathrm{T}}\right)$ [11][38]. Otherwise if this condition is not satisfied, the WLS problem can be solved by applying the following two-stage (TS) approach [35].

In the TS approach, we first solve a weighted least square problem by regressing X on Y^{T}, that is projecting the original high-dimensional dataset into the low-dimensional subspace. We can then calculate a auxiliary matrix $M \in \mathbb{R}^{d\times d}$ and its SVD. Finally, the optimal projection matrix can be obtained from the SVD of M. It is noted that the size of matrix M is small. As a result, the computational burden for calculating the SVD of matrix M is relatively low. The basic steps of performing two-stage approach can be summarized as follows. First, solve the weighted least squares problem $Min_P \Delta^{(ML)}\left\|Y^{\mathrm{T}} - X^{\mathrm{T}}P\right\|_F^2$. Second, let $\widehat{X} = P^{\mathrm{T}}X$ and calculate the auxiliary matrix as $M = \widehat{X}\Delta^{(ML)}Y^{\mathrm{T}}$. We can then calculate the SVD of M as $M = U_M\Sigma_M U_M^{\mathrm{T}}$ and set $V_M^* = U_M\Sigma_M^{-1/2}$. Finally, the optimal transformation matrix can be given by $V_{TS}^* = \Xi V_M^*$.

Next we will elaborate that the optimal projection matrix V_{TS}^* obtained by two-stage approach is equivalent to that in Eq.36. By solving the WLS problem $Min_P \Delta^{(ML)}\left\|Y^{\mathrm{T}} - X^{\mathrm{T}}P\right\|_F^2$, we have $P = \left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{-1}X\Delta^{(ML)}Y^{\mathrm{T}}$. We thus have $\widehat{X} = P^{\mathrm{T}}X = Y\Delta^{(ML)}X^{\mathrm{T}}\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{-1}X$ and the auxiliary matrix M can be given by

$$M = \widehat{X}\Delta^{(ML)}Y^{\mathrm{T}} = Y\mathrm{H}^{(ML)}X^{\mathrm{T}}\left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{-1}X\Delta^{(ML)}Y^{\mathrm{T}} = \widetilde{H}_b^{\mathrm{T}}\widetilde{U}_1\Sigma_t^{-1}\Sigma_t^{-1}\widetilde{U}_1^{\mathrm{T}}\widetilde{H}_b. \tag{37}$$

The second equation holds because $\widetilde{H}_b = X\Delta^{(ML)}Y^{\mathrm{T}}$ and $X\Delta^{(ML)}X^{\mathrm{T}} = \widetilde{U}_1\Sigma_t^2\widetilde{U}_1^{\mathrm{T}}$. Since $H = \Sigma_t^{-1}U_1^{\mathrm{T}}\widetilde{H}_b$ and its SVD is $H = P\Sigma_b Q^{\mathrm{T}}$, we have $M = H^{\mathrm{T}}H = Q\Sigma_b^2Q^{\mathrm{T}}$. The above equation indicates that $Q\Sigma_b^2Q^{\mathrm{T}}$ is the SVD of M, we thus have $V_M^* = Q\Sigma_b^{-1}$ and the optimal projection matrix using the two-stage approach can be given by

$$V_{TS}^* = \Xi V_M^* = \left(X\Delta^{(ML)}X^{\mathrm{T}}\right)^{-1}X\Delta^{(ML)}Y^{\mathrm{T}}Q\Sigma_b^{-1} = \widetilde{U}_1\Sigma_t^{-1}\left(\Sigma_t^{-1}\widetilde{U}_1^{\mathrm{T}}\widetilde{H}_b\right)Q\Sigma_b^{-1} = \widetilde{U}_1\Sigma_t^{-1}V\Sigma_b Q^{\mathrm{T}}Q\Sigma_b^{-1} = \widetilde{U}_1\Sigma_t^{-1}V \tag{38}$$

which is just equivalent to the optimal solution obtained by the direct eigen-decomposition approach in Eq.34.

6 Related Work: Connection and Discussion

In this section, we mainly discuss the important issues related to our proposed MS^{3}MP and OMS^{3}MP algorithms.

6.1 Relation to Semi-Supervised Dimensionality Reduction (SSDR) [8]

A similar work to our study is SSDR, which also preserves the global covariance structure of all samples as well as the PC defined based on the labeled sample points. For two given sets of ML and CL constraints, namely $ml = \left\{\left(x_i, x_j\right) \mid x_i \in V, x_j \in V, l\left(x_i\right) = l\left(x_j\right)\right\}$ and $cl = \left\{\left(x_i, x_j\right) \mid x_i \in V, x_j \in V, l\left(x_i\right) \neq l\left(x_j\right)\right\}$, the SSDR criterion is define as

$$\underset{P\in\mathbb{R}^{n\times d}}{Max} \frac{1}{2N}\sum_{i,j}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 W_{i,j}^{(t)} + \frac{1}{2}\sum_{\left(x_i,x_j\right)\in cl}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 \widetilde{H}_{i,j}^{(cl)} - \frac{1}{2}\sum_{\left(x_i,x_j\right)\in ml}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 \widetilde{H}_{i,j}^{(ml)}, \tag{39}$$

where $\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 = Tr\left(\left(P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right)\left(P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right)^{\mathrm{T}}\right)$, $(1/2)\sum_{i,j}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 W_{i,j}^{(t)} = Tr\left(P^{\mathrm{T}}G^{(t)}P\right), G^{(t)} = X\left(D^{(t)} - W^{(t)}\right)X^{\mathrm{T}} = XL^{(t)}X^{\mathrm{T}}$ with $W_{i,j}^{(t)} = 1/N$ is the total scatter matrix [5] defined for preserving the global covariance structures of all samples, including labeled and unlabeled ones. $\widetilde{H}_{i,j}^{(cl)} = \widetilde{H}_{j,i}^{(cl)} = \alpha_1 / n_{CL}$ if $\left(x_i, x_j\right) \in cl$, and else 0. $\widetilde{H}_{i,j}^{(ml)} = \widetilde{H}_{j,i}^{(ml)} = \alpha_2 / n_{ML}$ if $\left(x_i, x_j\right) \in ml$, and 0 otherwise, where α_1 and α_2 are tuning parameters. A concise form of Eq.39 is given as

$$\underset{P\in\mathbb{R}^{n\times d}}{Max} \frac{1}{2}\sum_{i,j}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 \tilde{S}_{i,j} \quad \text{with} \quad \tilde{S}_{i,j} = \begin{cases} (1/N)W_{i,j}^{(t)} + \widetilde{H}_{i,j}^{(cl)}, \text{if pair } \left(x_i, x_j\right) \in cl \\ (1/N)W_{i,j}^{(t)} - \widetilde{H}_{i,j}^{(ml)}, \text{if pair } \left(x_i, x_j\right) \in ml \\ (1/N)W_{i,j}^{(t)}, \text{otherwise} \end{cases}. \tag{40}$$

Note that when $L = I - (1/N)ee^{\mathrm{T}}$, where e is a column vector of all ones, LPP is converted to PCA [7], that is $L^{(t)} = L$. Based on this condition and Definition 1, with similar computational formulation and weighting method as optimizing SSDR, the problem of our OMS^{3}MP algorithm can be written as

$$\underset{P\in\mathbb{R}^{n\times d}}{Max} \frac{1}{2}\sum_{i,j}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 \widetilde{M}_{i,j} \quad \text{with} \quad \widetilde{M}_{i,j} = \begin{cases} (1/N)W_{i,j} + \left(n_{CL}/N\right)\widetilde{W}_{i,j}^{(CL)}, \text{if pair } \left(x_i, x_j\right) \in CL \\ (1/N)W_{i,j} - \left(n_{ML}/N\right)\widetilde{W}_{i,j}^{(ML)}, \text{if pair } \left(x_i, x_j\right) \in ML \\ (1/N)W_{i,j}, \text{otherwise} \end{cases} \tag{41}$$

when parameter $\mu_3 = -1$ and only constrained data pairs are weighted. Then the only difference between SSDR and OMS^{3}MP lies in that matrix $\widetilde{M}$ in OMS^{3}MP can reflect the local density around each constrained data pair, while in SSDR, every pair of constrained samples is equally treated in $\tilde{S}$ and thus only the global geometrical structures are preserved. As a result, OMS^{3}MP is superior to SSDR for feature representation. It is also noted that OMS^{3}MP is a naturally localized extension of the SSDR criterion when $\alpha_1 = n_{CL}^2$ and $\alpha_2 = n_{ML}^2 / N$. In particular when every pair of constrained points is neighbors, SSDR is equivalent to our OMS^{3}MP criterion.

6.2 Relation to SSML [12]

Another related PC guided work is called *Semi-Supervised Metric Learning* (SSML), in which both the positive and negative constraints as well as the intrinsic local topological structures of data are considered. For the sets of ML and CL constraints defined in Eq.39, the linearized criterion of SSML is formulated as

$$P^* = \arg\max_{P\in\mathbb{R}^{n\times d}} \frac{\frac{1}{2}\sum_{(x_i,x_j)\in cl}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 \widetilde{H}_{i,j}^{(cl)}}{\frac{1}{2}\sum_{(x_i,x_j)\in ml}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}x_j\right\|^2 \widetilde{H}_{i,j}^{(ml)} + \frac{\alpha_3}{2}\sum_{i=1}^{N}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}\sum_{x_j\in N_+^{(x_i)}} F_{i,j}x_j\right\|^2}, \tag{42}$$

where term $\sum_{i=1}^{N}\left\|P^{\mathrm{T}}x_i - P^{\mathrm{T}}\sum_{x_j\in N_+^{(x_i)}} F_{i,j}x_j\right\|^2$ is just the criterion of *Neighborhood Preserving Embedding* (NPE) [27], which is added to preserve the local manifold structures of all samples, including labeled and unlabeled data, α_3 is a tuning parameter. Weight matrices $\widetilde{H}^{(cl)}$ and $\widetilde{H}^{(ml)}$ are similarly defined as Eq.39. Weights $F_{i,j}$ can linearly reconstruct x_i in the best possible way from its neighbors [27][40]. A very similar work has been discussed in [24], which aims at optimizing the same problem as SSML. Similar to the SSDR, SSML cannot effectively reflect the ML and CL constrained within-manifold and between-manifold, because every pair of constrained samples is equally treated. According to [39], the LPP criterion can similarly be reformulated as $Min_{P^{\mathrm{T}}\hat{X}\hat{X}^{\mathrm{T}}P=I} Tr\left(P^{\mathrm{T}}\hat{X}\hat{L}\hat{X}^{\mathrm{T}}P\right)$, where the normalized graph Laplacian $\hat{L} = I - \hat{W} = D^{-1/2}LD^{-1/2}$, $\hat{W} = D^{-1/2}WD^{-1/2}$ and $\hat{X} = XD^{1/2}$. Specifically if $U = W = (1/N)ee^{\mathrm{T}}$, we have $(I-U)^{\mathrm{T}}(I-U) = I - \hat{W}$, that is LPP is equivalent to NPE in this case. Under this case, when every pair of constrained samples is neighbors, SSML is equivalent to MS^{3}MP when $\widetilde{H}^{(cl)} = n_{CL}\widetilde{H}^{(cl)}$, $\widetilde{H}^{(ml)} = (n_{ML}/N)\widetilde{H}^{(ml)}$, $\mu_1 = 0$, $\mu_2 = N/2$ and only constrained data pairs are weighted. We also noted that when $\alpha_3 = 0$, SSML is reduced to Xiang's algorithm [30]. But Xiang's work can only reflect the global structures of data as SSDR and suffers from the same problems as CMLP.

Before addressing the following relations with another two existing works, we first give the following definition:

Definition 2. When *ML*-graph and *CL*-graph are generalized to the have the same vertex number as graph G and all available constraints are applied, one can easily generalize the scatter matrices $\widetilde{R_{ML}}$ and $\widetilde{R_{CL}}$ to

$$\widetilde{R_{ML}} = \frac{1}{2}\sum_{i,j=1}^{l}(x_i - x_j)(x_i - x_j)^{\mathrm{T}}\widetilde{W}_{i,j}^{(ML)} = X_L\tilde{L}^{(ML)}X_L{}^{\mathrm{T}},\ \widetilde{R_{CL}} = \frac{1}{2}\sum_{i,j=1}^{l}(x_i - x_j)(x_i - x_j)^{\mathrm{T}}\widetilde{W}_{i,j}^{(CL)} = X_L\tilde{L}^{(CL)}X_L{}^{\mathrm{T}} \tag{43}$$

when the non-normalized graph Laplacian $\tilde{L}^{(ML)}$ and $\tilde{L}^{(CL)}$ are applied in our methods, where the entries of $l\times l$ dimensional adjacency matrices $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$ are similarly defined as above. It is noted that the local density information preserved by the adjacency matrices $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$ are kept unchanged. Based on this definition, the following relations with the related work can be summarized.

6.3 Relation to SELF [2]

A class labels guided semi-supervised DR method is called SELF [2]. SELF smoothly bridges the Local Fisher Discriminant Analysis (LFDA) [21] and PCA so that the global structures of all points as well as local structures defined by a small number of labeled data can be controlled. The optimization problem of SELF is give by

$$P^* = \arg\max_{P\in\mathbb{R}^{n\times d}} \frac{Tr\left(P^{\mathrm{T}}\left(\alpha_4 G^{(t)} + (1-\alpha_4)S^{(lbc)}\right)P\right)}{Tr\left(P^{\mathrm{T}}\left((1-\alpha_4)S^{(lwc)} + \alpha_4 I\right)P\right)}, \tag{44}$$

where α_4 is a tunable parameter, $S^{(lbc)}$ and $S^{(lwc)}$ are local inter- and intra-class scatter matrices, and $G^{(t)}$ is similarly defined. According to [2][21], scatters $S^{(lbc)}$ and $S^{(lwc)}$ can be expressed in pairwise forms as

$$S^{(lbc)}=\frac{1}{2}\sum_{i,j=1}^{N}\left(x_i-x_j\right)\left(x_i-x_j\right)^{\mathrm{T}}B_{i,j}^{(lb)},\ S^{(lwc)}=\frac{1}{2}\sum_{i,j=1}^{N}\left(x_i-x_j\right)\left(x_i-x_j\right)^{\mathrm{T}}B_{i,j}^{(lw)}. \tag{45}$$

Let l_t be the sample number of class t in X_L, then weight matrices $B^{(lb)}$ and $B^{(lw)}$ are defined as

$$\begin{cases}B_{i,j}^{(lw)}=B_{j,i}^{(lw)}=\left(1/l_t\right)W_{i,j}, \text{if } l\left(x_i\right)=l\left(x_j\right)=t\\ B_{i,j}^{(lw)}=B_{j,i}^{(lw)}=0, \text{else if } l\left(x_i\right)\neq l\left(x_j\right)\end{cases},\ \begin{cases}B_{i,j}^{(lb)}=B_{j,i}^{(lb)}=\left(1/l-1/l_t\right)W_{i,j}, \text{if } l\left(x_i\right)=l\left(x_j\right)=t\\ B_{i,j}^{(lb)}=B_{j,i}^{(lb)}=1/l, \text{else if } l\left(x_i\right)\neq l\left(x_j\right)\end{cases}, \tag{46}$$

where W reflects the local density information of unlabeled data. When the simple-minded method is applied, $W_{i,j}=W_{j,i}=1$ if x_i and x_j are neighbors, and else 0. When Definition 2 is satisfied, our MS^{3}MP problem can be updated with the generalized $\widetilde{S_{CL}}$, $\widetilde{S_{ML}}$, $\widetilde{W}^{(ML)}$ and $\widetilde{W}^{(CL)}$. Note that both SELF and MS^{3}MP clearly consider the local density information around each data point. It is worth noting that $S^{(lbc)}$ and $S^{(lwc)}$ can be converted into scatters $\widetilde{S_{CL}}$ and $\widetilde{S_{ML}}$ respectively when the following conditions are satisfied: (1) $c=1$; (2) All data points are equally treated in W; (3) $\widetilde{W}^{(ML)}=\left(1/N\right)\widetilde{W}^{(ML)}$ and all the intra-class sample pairs are neighbors. In such cases, scatters $S^{(lbc)}$ and $\widetilde{S_{CL}}$ make no sense, but our methods can be effectively executed as SELF. Similarly when $L=I-\left(1/N\right)ee^{\mathrm{T}}$, LPP is equivalent to PCA. In particular, based on the above settings, SELF is equivalent to our MS^{3}MP algorithm when $\mu_1=-1$, $\mu_2=0$, $\beta=\alpha_4=1/N$ and $n_{ML}=N-1$. It is also noted that SELF is reduced to the original LFDA when $\alpha_4=0$ and is also able to be transformed to the PCA criterion when $\alpha_4=1$. It is also noted that when PCA is identical to LPP and non-normalized graph Laplacian are used in MS^{3}MP, the problem in Eq.11 of MS^{3}MP under the trace ratio criterion [17][18] can be considered as the out-of-sample extension of the recent Semi-Supervised Laplacian Eigenmaps [28] algorithm when $\mu_1=-1$ and $\mu_2=0$ in MS^{3}MP.

6.4 Relation to SSMMC [10]

Another related class labels directed semi-supervised DR technique is SSMMC, which is the semi-supervised extension of MMC by incorporating the local information preserving power of LPP into MMC problem. SSMMC can make use of both labeled and unlabeled samples. The optimization problem of SSMMC is give by

$$P^*=\arg\max_{P\in\mathbb{R}^{n\times d}}Tr\left(P^{\mathrm{T}}S^{(bc)}P-\alpha_5P^{\mathrm{T}}S^{(wc)}P-\alpha_6P^{\mathrm{T}}\widetilde{S}_tP\right), \tag{47}$$

where $S^{(bc)}$ and $S^{(wc)}$ are LDA intra- and inter-class scatters [5] respectively, $\widetilde{S}_t=XD^{-1/2}LD^{-1/2}X^{\mathrm{T}}$, α_5 and α_6 are tuning parameters. According to [21], $S^{(bc)}$ and $S^{(wc)}$ can be expressed using the pairwise forms as

$$S^{(bc)}=\frac{1}{2}\sum_{i,j=1}^{N}\left(x_i-x_j\right)\left(x_i-x_j\right)^{\mathrm{T}}B_{i,j}^{(b)},\ S^{(wc)}=\frac{1}{2}\sum_{i,j=1}^{N}\left(x_i-x_j\right)\left(x_i-x_j\right)^{\mathrm{T}}B_{i,j}^{(w)}, \tag{48}$$

where weight matrices $B^{(b)}$ and $B^{(w)}$ are respectively defined as follows:

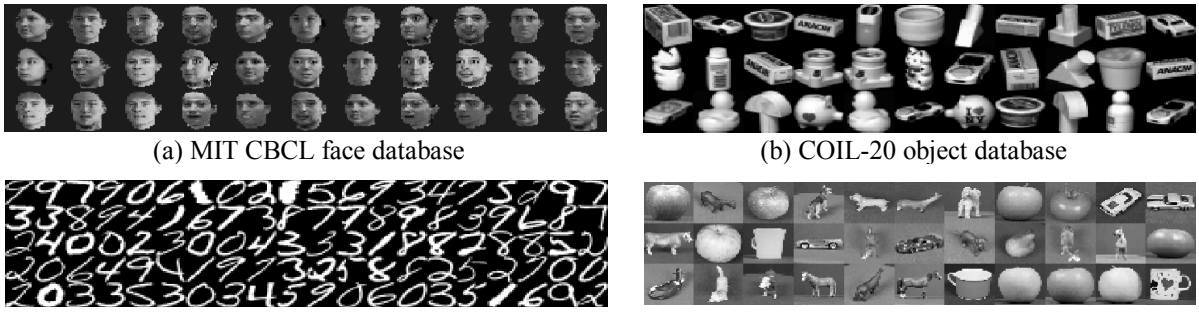
$$\begin{cases}B_{i,j}^{(w)}=B_{j,i}^{(w)}=1/l_t, \text{if } l\left(x_i\right)=l\left(x_j\right)=t\\ B_{i,j}^{(w)}=B_{j,i}^{(w)}=0, \text{else if } l\left(x_i\right)\neq l\left(x_j\right)\end{cases},\ \begin{cases}B_{i,j}^{(b)}=B_{j,i}^{(b)}=1/l-1/l_t, \text{if } l\left(x_i\right)=l\left(x_j\right)=t\\ B_{i,j}^{(b)}=B_{j,i}^{(b)}=1/l, \text{else if } l\left(x_i\right)\neq l\left(x_j\right)\end{cases}. \tag{49}$$

That is $B^{(b)}$ and $B^{(w)}$ aim at treating each pair of samples equally no matter whether their class labels are the same or not. When Definition 2 is satisfied, scatter matrices $\widetilde{S_{ML}}$ and $\widetilde{S_{CL}}$ in our methods can be generalized to Eq.43. As a result, the above two conditions are satisfied and normalized graph Laplacian are used in OMS^{3}MP, SSMMC is equivalent to OMS^{3}MP when $\alpha_5=n_{ML}/N$, $\alpha_6=1/N$, $c=1$, $\widetilde{W}^{(ML)}=\left(1/N\right)\widetilde{W}^{(ML)}$ and all intra-class point pairs are neighbors. Note that SSMMC can be easily reduced to the standard MMC when $\alpha_5=1$ and $\alpha_6=0$.

7 Simulation Results and Analysis

In this section, we conduct simulations to evaluate the effectiveness of our MS^{3}MP and OMS^{3}MP methods. The performance is compared with six semi-supervised DR algorithms, including SSDR, SSML, SDA, LapLDA [20], SSLDA and SSMMC. To avoid selecting the parameter in constructing the adjacency matrices for SDA, LapLDA [20], SSLDA, SSMMC and our methods, the simple-minded method [3] is applied. The *k*-neighborhood definition [3][7] is used to define the neighbors of each point. In the simulations, the model parameters in each algorithm are carefully chosen and the best results over tuned parameters will be reported. We perform all simulations on a PC with Intel (R) Core (TM) i5 CPU 650 @ 3.20 GHz 3.19 GHz 4G.

In this study, four benchmark problems are tested. The first one is MIT CBCL face recognition database [25]; the second one is COIL-20 database, which is available at http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php; the third one is USPS handwritten digits database [33]; the last one is ETH80 object databases [34]. As is common practice the images of the MIT CBCL, COIL-20 and ETH80 databases are resized to 32×32 pixels. Each pixel is considered as an input variable and so each image corresponds to a data point in a 1024-dimensional space. In the simulations, we randomly split each dataset into training and test set. Prior to our studies, COIL-20, USPS and ETH80 datasets are processed by PCA to reduce the dimensionality to 200 due to the computational consideration. We also randomly select samples from the training set to form the labeled and unlabeled sets. For face recognition, the one-nearest-neighbor (1NN) classifier with Euclidean metric is used. The training set is used to train a 1NN learner. The test set is then projected in the reduced output space using the DR matrix learned from the training data. Finally, the learner is used for evaluating the accuracies of the test set and the 1NN accuracy is treated as our evaluation metric. Note that unlabeled set will also be added to the test set for evaluating. The *ML* and *CL* constraints are created based on whether the class labels of neighbors included in the labeled data of training set are the same or different in our studies.



(a) MIT CBCL face database (b) COIL-20 object database

(c) USPS digit database (d) ETH80 object database

Figure 1: Some typical sample images in the tested real-world databases.

7.1 Face Representation

In this subsection, we address a simulation to examine the visual properties of the transforming matrices of our MS^{3}MP and OMS^{3}MP methods. The visual performance of our algorithms is compared with the semi-supervised SSDR, SSML, SDA, SSLDA, SSMMC and LapLDA. In this study, the benchmark real MIT CBCL face database is employed. MIT-CBCL provides us two training sets: 1. High resolution pictures, including frontal, half-profile and profile view; 2. Synthetic face images (324 images per person) rendered from 3D head models of 10 persons. The images are captured under different illuminations, poses and backgrounds. In our study, the second face set is tested. Some typical images are shown in Figure 1(a). For SSDR, SSML and our methods, our proposed constraint selection method is used and 50% constraints are applied. The *k* number is set to 9 for each NNS type method. For each semi-supervised method, we randomly select 15 images with 5 labeled from each person to learn the optimal face image subspaces. We plot the eigenfaces reconstructed by the first 9 eigenvectors obtained by each method. The eigenfaces are exhibited in Figure 2. For our proposed algorithms, we also plot the first 9 eigenvectors. As a result, faces can be mapped into the marginal semi-supervised sub-manifold subspaces spanned by the achieved

eigenvectors. Observing from the results in Figure 2, we find that the eigenvectors delivered by OMS^{3}MP, SSML, SDA, SSLDA and LapLDA, that are defined as ratio trace problems solved by generalized eigen-decomposition method, are more noisy than those of SSDR, MS^{3}MP and SSMMC that are defined as trace difference problems solved by standard eigen-decomposition method, implying that they are able to capture more detailed disciminant information on the face images.

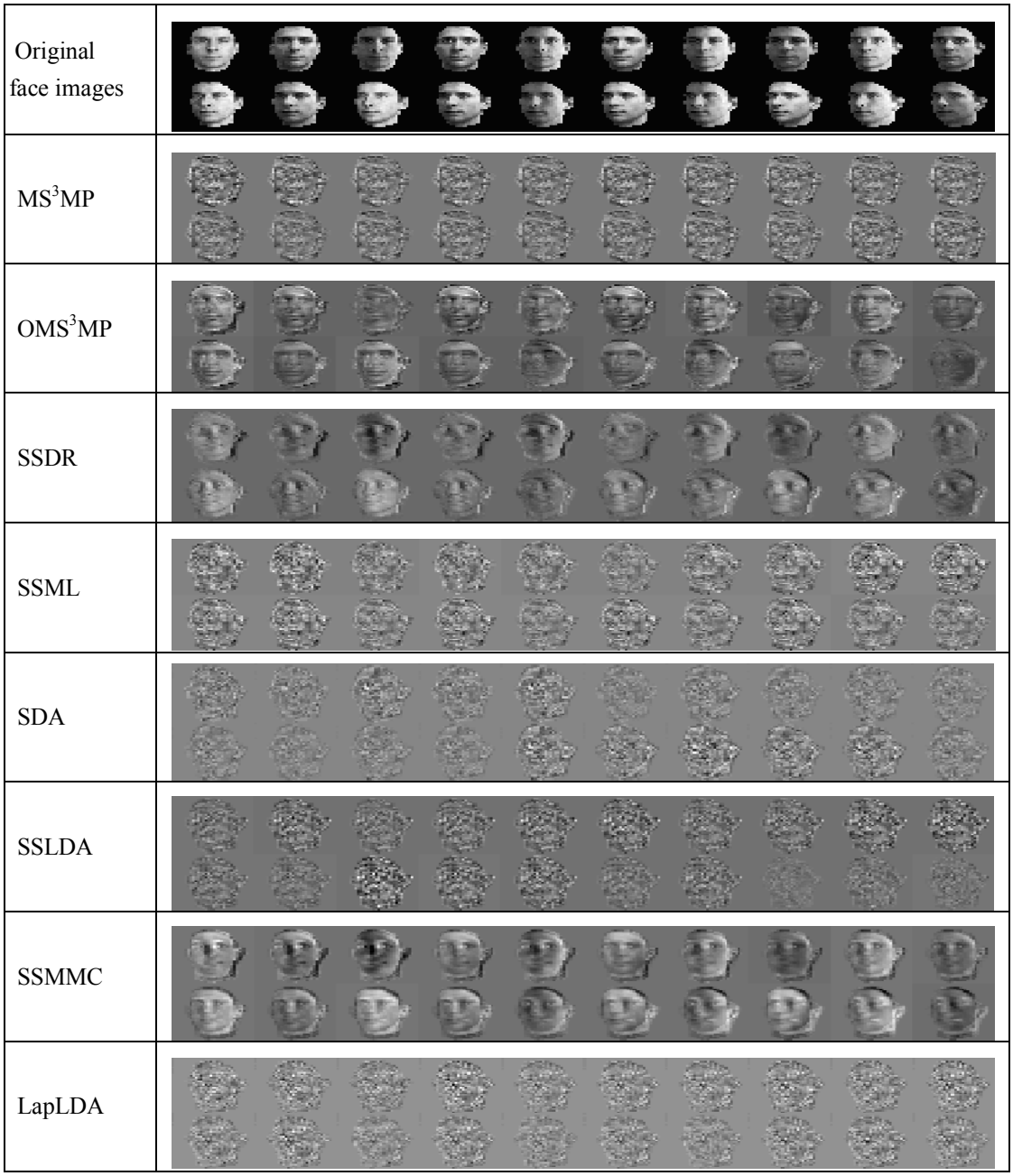


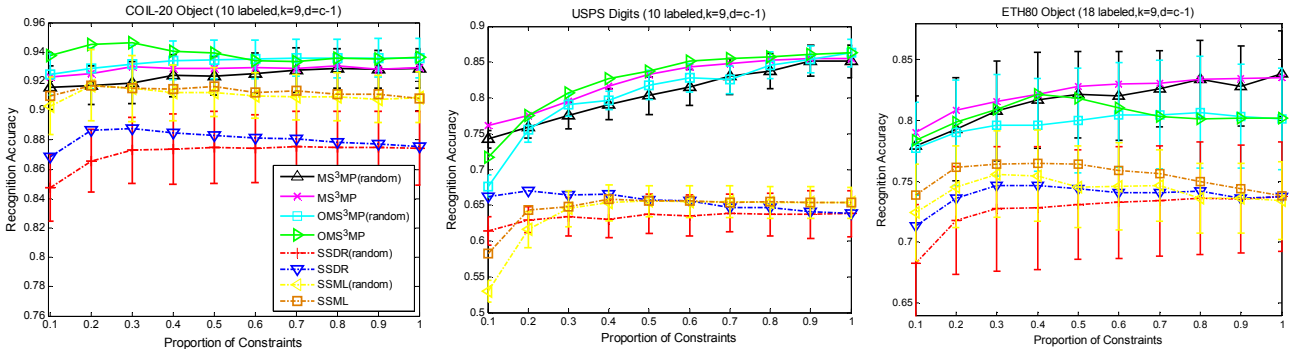
Figure 2: Visualization of the transforming matrices of MS^{3}MP, OMS^{3}MP, SSDR, SSML, SDA, SSLDA, SSMMC and LapLDA on the MIT CBCL face database.

7.2 Object Recognition on COIL-20 Database

This section tests our methods by recognizing the object images of the well-known Columbia object image library (COIL-20) database. This database is consisted of a total of 1440 gray object images with black background for 20 different subjects (72 images per object). We have shown some typical object sample images in Figure 1 (b). In this simulation, settings over different numbers of pairwise constraints, reduced dimensionalities and labeled data samples are evaluated. The k number in NNS is set to 9 in all cases.

7.2.1 Performance comparison: random vs. informative constraints

We first evaluate our informative constraint selection method by comparing with the random selected constraints. The reduced dimensionality d is set to c-1. This simulation mainly investigates whether our selected informative constraints can improve the performance of SSDR, SSML, MS^{3}MP and OMS^{3}MP. For random constraints, we average the results over 50 runs of randomly selected constraints in all simulations. In this simulation, we select 10 labeled data as well as the same number of unlabeled data per object to form the training set. The test results averaged over 10 random realizations of training/ test sets are described in Figure 3(a). In all our simulations, q% constraints means that q% ML constraints plus q% CL constraints. Based on the results of Figure 3(a), Table 1 reports the corresponding mean accuracies, cumulative running time (in seconds) and cumulative differences (C_DIFF) between the maximum and minimum accuracies of constraint proportion (from % to 100%) over repetitions. Note that the results reported in Table 1 are averaged over 10 realizations of training and test sets. We observe that: (1) The performance of SSDR, SSML, MS^{3}MP and OMS^{3}MP are boosted with the informative constraints, especially when the number of constraints is small. The result of each DR algorithm with informative constraints is very close to that averaged over random selections as the number of constraints increases to a high level. (2) Large C_DIFF and standard deviations are produced in different runs and much time is required. As we compute the result of each method using the informative constraints only once for each case, zero C_DIFF values and standard deviations as well as little time requirement are exhibited. (3) The performances of our presented methods are superior to SSDR and SSML in terms of recognition accuracies.



(a) (b) (c)

Figure 3: Recognition comparison under random and informative constraints based on the real databases.

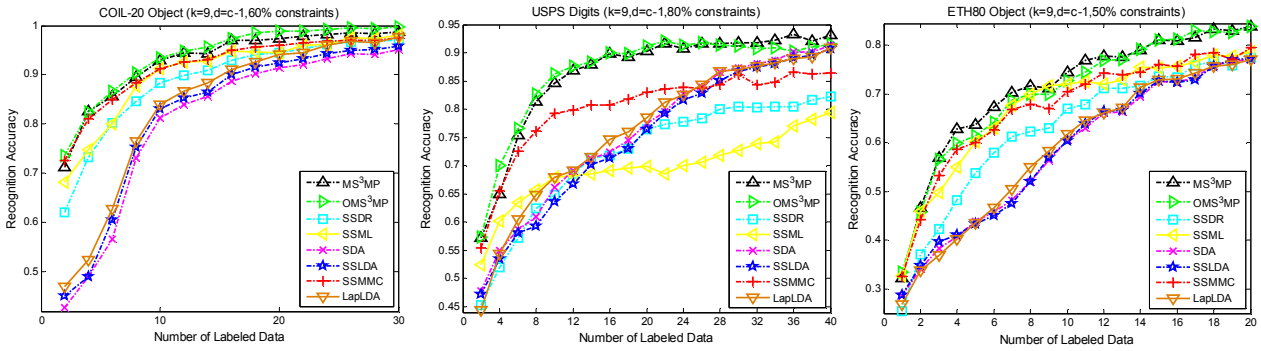
Table 1. Performance comparison over random and informative constraints based on the real databases.

Result / Method	COIL-20 (10 labeled, k=9)			USPS Digits (10 labeled, k=9)			ETH80 Object (10 labeled, k=9)		
	MEAN	C_DIFF	TIME	MEAN	C_DIFF	TIME	MEAN	C_DIFF	TIME
MS^{3}MP(random)	92.38%	13.08%	362.8092	80.56%	36.68%	146.4654	81.67%	34.09%	257.7549
MS^{3}MP	92.82%	00.00%	7.1380	82.34%	00.00%	2.8329	82.28%	00.00%	6.6697
OMS^{3}MP(random)	93.32%	08.94%	329.8419	80.51%	44.79%	117.7034	79.81%	20.00%	295.2249
OMS^{3}MP	93.94%	00.00%	6.4576	82.51%	00.00%	2.3532	80.50%	00.00%	5.8793
SSDR (random)	87.42%	12.40%	131.8985	63.34%	36.96%	121.2885	72.63%	22.27%	129.7438
SSDR	88.22%	00.00%	2.5428	65.46%	00.00%	2.6690	73.85%	00.00%	2.1509
SSML (random)	91.06%	16.06%	325.1464	63.70%	36.14%	159.2269	74.26%	40.45%	235.8547
SSML	91.30%	00.00%	6.3797	64.60%	00.00%	3.1695	75.41%	00.00%	6.2440

7.2.2 Performance evaluation under different labeled numbers

To better understand how our MS^{3}MP and OMS^{3}MP methods behave in different semi-supervised settings, this simulation tests our methods over different labeled numbers, Lab=1, 2,…, 30. Our constraint selection method is applied. In this study, the dimensionality d is set to c-1 and 60% constraints are applied in SSDR, SSML and our methods. For each Lab, we average the results over 10 realizations of training/test sets. The results are shown in

Figure 4(a) and the corresponding statistics are recorded in Table 2. Observing from the results, we can find that: (1) SDA, SSLDA and LapLDA deliver comparable results in all cases. This is because these SDA and SSLDA are all based on the idea such that by incorporating that local geometrical information into the LDA criterion. It is also noted that LapLDA is formulated under a least square framework. Thus based on the relation between LDA and multivariate linear regression with certain class indicator matrix [32], LapLDA can be equivalent to the SDA [18]. SSDR works better than SDA, SSLDA and LapLDA in most cases. (2) Our MS^{3}MP and OMS^{3}MP algorithms outperform the other DR methods. SSML obtains the comparative results with SSMMC in most cases, and both are superior to the remaining techniques. The runtime performances of our methods are comparable to SSML and SSMMC. The other methods need similar running time for object recognition.



(a) (b) (c)

Figure 4: Recognition accuracy under different numbers of labeled data in each class of the real databases.

Table 2. Performance comparison under different numbers of labeled data in each class of the real databases.

Result / Method	COIL-20 (10 labeled, k=9)			USPS Digits (10 labeled, k=9)			ETH80 Object (10 labeled, k=9)		
	MEAN	BEST	TIME	MEAN	BEST	TIME	MEAN	BEST	TIME
MS^{3}MP	92.96%	98.81%	3.1272	86.41%	93.26%	1.8323	71.12%	83.96%	1.4186
OMS^{3}MP	93.94%	99.80%	3.0045	86.63%	92.09%	1.6446	70.29%	84.25%	1.3114
SSDR	88.89%	97.16%	0.5562	72.10%	82.19%	0.6439	62.68%	77.42%	0.2942
SSML	90.42%	98.11%	2.7283	69.58%	79.46%	1.5767	67.16%	78.65%	0.9403
SDA	80.79%	95.11%	0.7052	76.10%	91.43%	0.8398	58.08%	77.02%	0.3695
SSLDA	82.19%	95.83%	0.7103	75.00%	90.71%	0.8484	58.26%	77.22%	0.3716
SSMMC	91.72%	97.49%	3.5811	80.26%	86.58%	1.4746	67.19%	79.60%	1.0530
LapLDA	83.81%	97.63%	0.7262	76.42%	90.93%	0.4934	58.60%	77.04%	0.2748

7.2.3 Object recognition

We then test our methods by object recognition. In this study, three setting based on *Lab*=10, 15 and 20 are tested. In our study, 60% constraints are applied in SSDR, SSML and our methods. For each *Lab*, the results are averaged over 10 realizations of training/test sets. As SDA, LapLDA, SSLDA can only extract c-1 meaningful features for c classes, we only report their results over first c-1 dimensions in all simulations. The recognition results are shown in Figure 5. For SSDR, SSML, MS^{3}MP and OMS^{3}MP, our informative constraint selection method is applied. Table 3 summarizes the results according to Figure 5. The mean time, best results and optimal image subspaces (DIM) are also included in Table 3. The following observations can be found. (1) The test result of each method varies with the increasing labeled numbers and reduced dimensionalities. (2) The mean and best accuracies of our MS^{3}MP and OMS^{3}MP methods are superior to other methods in most cases. SSML delivers comparable results to SSMMC and SSDR. The results of LapLDA are slightly better than SDA and SSLDA that are comparative in most cases. (3) Considering the running time performance, our MS^{3}MP and OMS^{3}MP are comparable to other methods, because in our methods only the vertices over the *ML*- and *CL*-graphs are used to construct the ML and CL constrained data matrices and constrained local manifold scatters. This operation can contribute to reducing the computational burden, especially when small proportions of constraints are employed.

7.3 USPS Handwritten Digits Recognition

This section tests our methods by recognizing the handwritten digits of the USPS database [33]. In this study, the popular subset containing 9298 16×16 handwritten digit images is employed. Since each pixel is considered as an input variable, each image is represented by a 256-dimensional vector. In our simulations, a sample set consisting of 300 randomly selected images from each digit are created in our studies. We show some typical sample images in Figure 1 (c). Similarly, simulation settings over different numbers of constraints, reduced dimensionalities and labeled samples are evaluated. The k number in k-neighborhood is set to 9 in all cases.

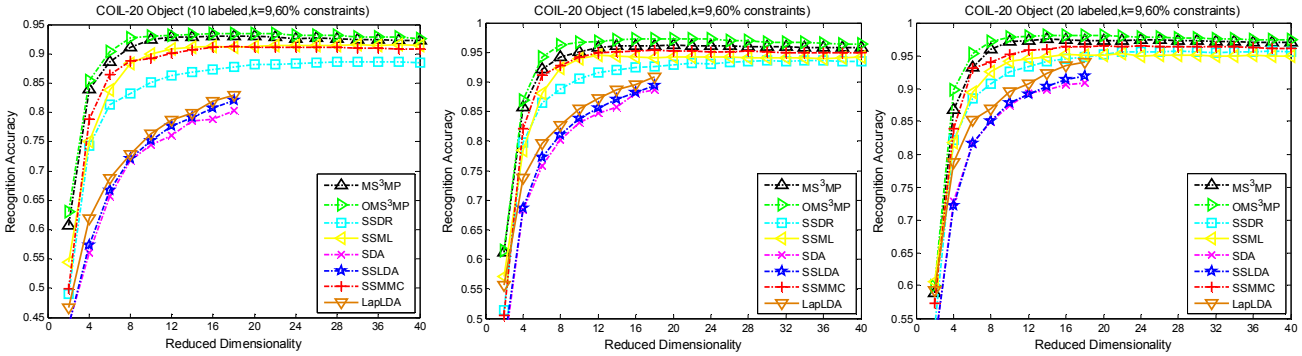


Figure 5: Recognition accuracy under different reduced dimensionalities on the COIL-20 database.

Table 3. Performance comparison under different reduced dimensionalities on the COIL-20 database.

Result / Method	COIL-20 (10 labeled, k=9)				COIL-20 (15 labeled, k=9)				COIL-20 (20 labeled, k=9)			
	MEAN	BEST	DIM	TIME	MEAN	BEST	DIM	TIME	MEAN	BEST	DIM	TIME
MS^{3}MP	90.60%	93.07%	18	1.1509	93.88%	96.24%	20	2.2958	95.01%	97.60%	14	4.1900
OMS^{3}MP	91.30%	93.46%	22	1.0340	94.87%	97.36%	22	2.1316	95.76%	98.13%	18	4.0847
SSDR	85.48%	88.70%	28	0.5145	90.43%	93.62%	30	0.7238	92.56%	95.70%	28	1.0336
SSML	88.74%	91.52%	34	0.9874	91.68%	94.50%	12	1.7229	92.73%	95.26%	22	2.8930
SDA	69.48%	80.34%	18	0.4778	77.68%	88.69%	18	0.6498	82.01%	90.88%	18	0.8002
SSLDA	70.47%	82.06%	18	0.8892	78.51%	89.41%	18	1.9451	82.30%	91.89%	18	3.2779
SSMMC	88.31%	91.23%	18	0.8626	92.46%	95.39%	18	2.0011	93.88%	96.45%	24	3.8580
LapLDA	72.23%	82.19%	18	0.2991	81.48%	90.87%	18	0.5399	85.59%	94.04%	18	0.8073

7.3.1 Performance comparison: random vs. informative constraints

This section tests the SSDR, SSML and our MS^{3}MP and OMS^{3}MP methods by using random and informative constraints. In this simulation, we select 10 labeled data as well as the same number of unlabeled samples per digit to form the training set. The results averaged over 10 random splits are given in Figure 3(b). The corresponding mean accuracies, cumulative running time and C_DIFF over repetitions are summarized in Table 1. The results in Table 1 are averaged over 10 splits. We see clearly that: (1) The figure is divided into two parts. The first part includes SSML and SSDR. Another part contains our MS^{3}MP and OMS^{3}MP methods. It is clear that the second group outperform the first group. (2) With informative constraints, the performance of each method is improved, especially when the applied constraints is fewer. With the increasing proportions of constraints, our methods tend to deliver close results to those averaged over random selections. (3) For random selections, large C_DIFF values are produced in different runs and much time are produced. Note that we only compute the result of each method with informative constraints once in each case, little time is needed.

7.3.2 Performance evaluation under different labeled numbers

To investigate how our MS^{3}MP and OMS^{3}MP algorithms perform in different semi-supervised settings on the real database, one simulation over different labeled numbers is prepared. Our constraint selection method is applied. The d value is set to c-1 and 80% constraints are employed in SSDR, SSML and our methods. For each *Lab*, we compute the averaged results over 10 realizations. The test results are exhibited in Figure 4(b) and the statistics

corresponding to Figure 4(b) are exhibited in Table 2. From Figure 4(b) and Table 2, we see that: (1) Similarly SDA, SSLDA and LapLDA deliver close results in almost all cases due to the intrinsic relationships between them. As the number of labeled samples increases to about 16, both SDA, SSLDA and LapLDA outperform SSDR and SSML. SSMMC delivers better results than SDA, SSLDA and LapLDA when the labeled number *Lab* is smaller than or equals to 26, but is worse than them after *Lab*=26. (2) In across all the labeled numbers, our MS^{3}MP and OMS^{3}MP methods outperform the other methods. For runtime performance, SSML, SSMMC and our methods are slightly slower than other methods. The running time requirements of the remaining methods are comparable.

7.3.3 Handwritten digits recognition

This subsection tests our methods by handwritten digits recognition. Three setting based on *Lab*=10, 20 and 30 are evaluated. In this simulation, 80% constraints are employed in SSDR, SSML and our methods. The informative constraints are also employed in these methods. The results are illustrated in Figure 6 and Table 4. The mean time, best results and DIM are also shown in Table 4. We have the following observations. Firstly, the performance of each method varies with the increasing numbers of labeled images and reduced dimensions. Secondly, promising results are exhibited by our proposed MS^{3}MP and OMS^{3}MP methods. The results obtained by our methods are superior to other methods. Specifically, the performance of MS^{3}MP and OMS^{3}MP goes up faster for each case as the number of reduced dimensionalities increases. Observing from the test results, for the case of *Lab*=30, the accuracies of SDA, SSLDA and LapLDA increase fast with the increasing *d* values as our methods. Thirdly, the results of SDA, SSLDA and LapLDA are similar. SSDR outperform the SSML algorithm over this real database, and both are better than SDA, SSLDA and LapLDA methods for *Lab*=10. Fourthly, the running time performance of our MS^{3}MP and OMS^{3}MP methods are comparable to the other methods.

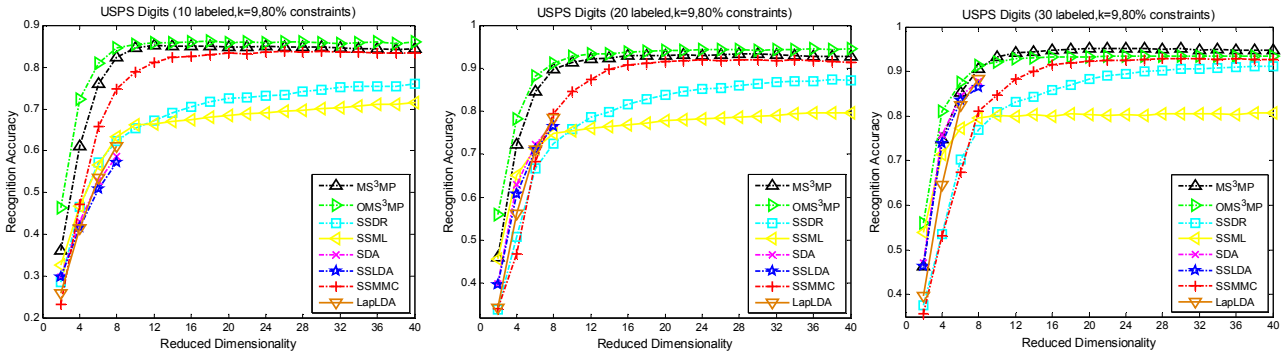


Figure 6: Recognition accuracy under different reduced dimensionalities on the USPS digits database.

Table 4. Performance comparison under different reduced dimensionalities on the USPS digits database.

Result / Method	USPS Digits (10 labeled, *k*=9)				USPS Digits (20 labeled, *k*=9)				USPS Digits (30 labeled, *k*=9)			
	MEAN	BEST	DIM	TIME	MEAN	BEST	DIM	TIME	MEAN	BEST	DIM	TIME
MS^{3}MP	81.36%	85.35%	12	0.4328	89.56%	93.42%	30	1.2099	91.47%	95.24%	26	2.6861
OMS^{3}MP	83.55%	86.21%	18	0.3599	91.60%	94.49%	40	1.0604	91.01%	93.48%	40	2.3766
SSDR	69.53%	76.55%	40	0.3678	80.21%	87.52%	40	0.7297	84.34%	91.60%	40	1.0461
SSML	66.63%	72.77%	40	0.4834	76.23%	80.11%	40	1.1921	78.84%	80.82%	40	2.4118
SDA	46.01%	58.56%	8	0.3469	63.10%	77.32%	8	0.8888	73.77%	87.65%	8	1.0232
SSLDA	44.90%	57.25%	8	0.3994	62.01%	76.47%	8	1.3275	72.50%	86.39%	8	2.1916
SSMMC	78.16%	83.77%	30	0.3374	85.46%	91.95%	24	0.9472	86.64%	92.86%	36	2.0870
LapLDA	45.50%	61.12%	8	0.1386	60.01%	78.50%	8	0.4260	68.75%	88.29%	8	0.6260

7.4 Object Recognition on ETH80 Database

This study addresses an object recognition task using the benchmark ETH80 database [34]. This database contains images of 8 big categories: *apple*, *car*, *cow*, *cup*, *dog*, *horse*, *pear* and *tomato*. In each big category, there are 10

subcategories, each of which contains 41 images from different viewpoints. Overall, the database contains 3280 images of 80 objects. In this study, the last 20 subcategories are selected for our simulations. Therefore a 20-class problem is created. We show some typical sample images from the ETH80 database in Figure 1(d).

7.4.1 Performance comparison: random vs. informative constraints

We first report the test results of SSDR, SSML, MS^{3}MP and OMS^{3}MP with random and informative constraints on this object database. The d number is set to c-1. In this study, 18 labeled samples as well as the same number of unlabeled samples from each object are randomly selected to form the training set. The results are described in Figure 3(c) and Table 1. The averaged accuracies, cumulative running time and C_DIFF over 10 realizations of training and test sets are also reported in Table 1. Similar observations are found. (1) Our selected constraints can enhance the performance of each method, especially when the employed constraints are few. Based on achieving the reported results, we also observe that repeating selecting the constraints randomly produces large C_DIFF and standard deviations. At the same time, the computational burden is heavy, compared with our method. This is as we only need to compute the result of each method with informative constraints once. (2) In this simulation, MS^{3}MP delivers comparable results to OMS^{3}MP. Similarly, both SSDR and SSML are comparative.

7.4.2 Performance evaluation under different labeled numbers

We also prepare a simulation over different labeled numbers (from 1 to 20) to better understand how our methods behave over this real dataset. Our constraint selection method is applied in SSDR, SSML and our methods. In this simulation, d value is set to c-1 and 50% constraints are applied. For each labeled number, we average the results over 10 random splits. Figure 4(c) shows the simulation results. The corresponding statistics are reported in Table 2. From the obtained results, we can conclude that: (1) The performance of each algorithm increases with the increasing labeled numbers in each class. (2) The delivered results of SDA, SSLDA and LapLDA are close with each other and are inferior to other methods. SSMMC obtains the very competitive results with the SSML method. MS^{3}MP and OMS^{3}MP perform the best in all across labeled numbers in terms of accuracy, followed by SSMMC (SSML) and SSDR, respectively. Considering the running time performance, our methods are comparable to the SSML and SSMMC methods, and both are slightly slower than the remaining methods.

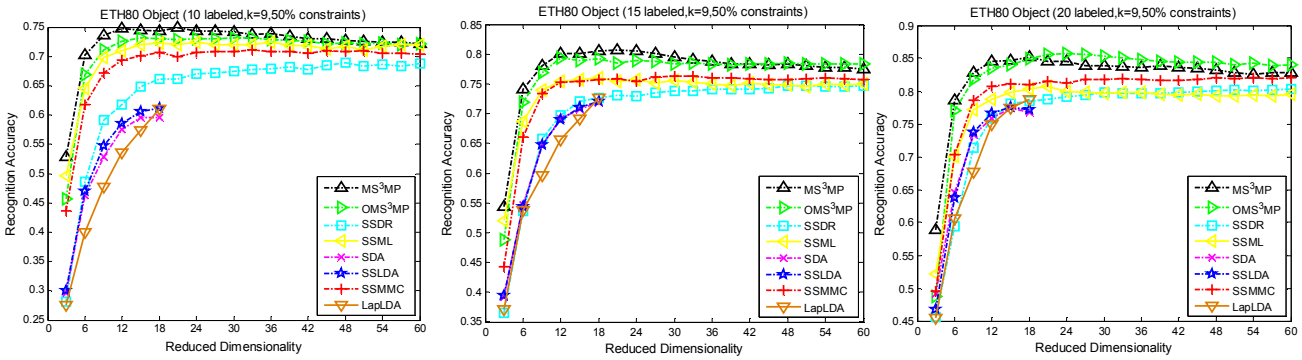


Figure 7: Recognition accuracy under different reduced dimensionalities on the ETH80 object database.

Table 5. Performance comparison under different reduced dimensionalities on the ETH80 object database.

Result / Method	ETH80 Object (10 labeled, k=9)				ETH80 Object (15 labeled, k=9)				ETH80 Object (20 labeled, k=9)			
	MEAN	BEST	DIM	TIME	MEAN	BEST	DIM	TIME	MEAN	BEST	DIM	TIME
MS^{3}MP	0.7238	0.7495	21	0.9539	0.7752	0.8071	21	2.2310	0.8223	0.8517	18	4.6313
OMS^{3}MP	0.7095	0.7315	30	0.8552	0.7667	0.7931	12	2.1841	0.8230	0.8571	24	4.3868
SSDR	0.6397	0.6890	48	0.4039	0.7037	0.7469	48	0.6680	0.7629	0.8029	60	0.8719
SSML	0.7028	0.7243	36	0.8160	0.7366	0.7596	18	1.7284	0.7767	0.8077	21	2.7999
SDA	0.5091	0.5965	15	0.3788	0.6175	0.7192	18	0.6108	0.6952	0.7748	15	0.8318
SSLDA	0.5202	0.6116	18	0.7591	0.6179	0.7206	18	1.8741	0.6930	0.7751	15	3.8085
SSMMC	0.6864	0.7116	33	0.6959	0.7366	0.7634	22	1.9203	0.7931	0.8204	30	3.5826
LapLDA	0.4788	0.6106	18	0.2313	0.5966	0.7259	18	0.4698	0.6748	0.786	18	0.8309

7.4.3 Object recognition

This subsection evaluates our methods by object recognition task by using the ETH80 dataset. In our study, three cases over labeled number *Lab*=10, 15 and 20 are evaluated. We test SSDR, SSML, MS^{3}MP and OMS^{3}MP with 50% constraints selected by our method for the simulations. The results are described in Figure 7 and Table 5. The mean running time, best records and DIM are also shown in Table 4. By observing from Figure 7 and Table 5, we find that: (1) The performance of each method vary with the increasing labeled numbers and *d* values. And the performance superiority of the algorithms is clear. (2) Based on incorporating the local geometrical information into the PC as well as the defined manifold scatters, the performances of MS^{3}MP and OMS^{3}MP are comparable and both are superior to the other methods. In particular, promising results can be obtained by our method with small *d* values. The performance of SDA and SSLDA are slightly better than LapLDA in all cases, and both are worse than other methods in most cases. SSMMC outperforms the SSDR in all settings. SSML performs better than SSDR and SSMMC for the case of *Lab*=10, and is comparable with them for the latter two settings. We also experimentally observe that larger *d* values cause the embeddings of our methods to degrade to some extent. (3) Similar observations can be found from Table 5. That is, the running time performance of our methods are similar to SSLDA, SSMMC and SSML. SSDR, SDA and LapLDA are slightly faster compared with other methods.

8 Concluding Remarks

In this paper, we have discussed the marginal semi-supervised sub-manifold projection problems. Two effective pairwise constrained marginal semi-supervised sub-manifold projection algorithms named MS^{3}MP and OMS^{3}MP are addressed for subsequent linear dimensionality reduction and classification. The learnt linear projections can do induction representing new points, thus out-of-sample problem is effectively solved. Kernelized MS^{3}MP and OMS^{3}MP are also elaborated with the standard kernel method. Considering that selecting the most informative constraints for constrained problems is challenging and important, we also introduce a constraint selection method to address the unstable issue caused by the random constraints as appeared in virtually all previous studies. In our method, instead of repeating randomly selecting the constraint subsets, only a single appropriate subset with "good" constraints is extracted. We experimentally observe that the performance of some existing PC guided semi-supervised methods can be boosted with our constraint selection method, especially when small proportions of constraints are used. We also describe the mathematical comparison between this study and the related studies. Several popular semi-supervised algorithms can be embedded into our framework as special cases. Based on the informative constraints and efficiently using the partially constrained data, our methods deliver satisfactory results for image representation and recognition.

Images are intrinsically matrices or second-order tensors. So, an interesting future work will lie in theoretically extending our approaches to handle the images in matrix form directly. It is also noted that neural networks have emerged as an important tool for classification. But most real data have high-dimensional attributes, so exploring whether one can enhance the scalability of algorithms and improve the classification results at the same time via pre-processing the original inputs by the proposed semi-supervised algorithms is worth studying.

References

[1] Chapelle, O., Schölkopf, B., Zien, A., eds. *Semi-Supervised Learning*. MIT Press, Cambridge, 2006.

[2] Sugiyama, M., Idé, T., Nakajima, S., Sese, J. Semi-supervised local Fisher discriminant analysis for dimensionality reduction. *Machine Learning*, 2010, 78(1-2):35-61.

[3] Belkin, M., Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 2003, 15(6):1373-1396.

[4] Hinton, G. E., Salakhutdinov R. R. Reducing the dimensionality of data with neural networks. *Science*, 2006, 313(5786):504-507.

[5]Martinez, A. M., Kak, A. C. PCA versus LDA. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2001, 23(2):228-233.

[6]Li, H., Jiang, T., Zhang, K. Efficient and Robust Feature Extraction by Maximum Margin Criterion. *IEEE Trans. on Neural Networks*, 2006, 17(1):157-165.

[7] He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H. Face Recognition Using Laplacianfaces. *IEEE Trans. on Patten Analysis and Machine Intelligence*, 2005, 27(3):228-340.

[8] Zhang, D. Q., Zhou, Z. H., Chen, S.C. Semi-supervised dimensionality reduction. In: *Proceedings of SDM 2007*, Minneapolis, Minnesota, USA.

[9] Cai, D., He, X. F., Han, J. Semi-supervised discriminant analysis. In: *Proceedings of ICCV 2007*, Rio de Janeiro, Brazil, pp.1-7.

[10] Song, Y. Q., Nie, F. P. Zhang, C. S., Xiang, S. M. A unified framework for semi-supervised dimensionality reduction. *Pattern Recognition*, 2008, 41(9):2789-2799.

[11] Sun, L., Ji, S. W., Ye, J. P. Canonical Correlation Analysis for Multi-Label Classification: A Least Squares Formulation, Extensions and Analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2011, 33(1): 194-200.

[12] Baghshah, M. S., Shouraki, S. B. Semi-Supervised Metric Learning Using Pairwise Constraints. In: *Proceedings of IJCAI*, 2009, pp.1217-1222.

[13] Zhang, D. Q., Chen, S. C., Zhou, Z. H. Constraint score: A new filter method for feature selection with pairwise constraints. *Pattern Recognition*, 2008, 41(5):1440-1451.

[14] Sun, D., Zhang, D. Q. Bagging Constraint Score for feature selection with pairwise constraints. *Pattern Recognition*, 2010, 43:2106-2118.

[15] Xing, E.P. Ng, A.Y., Jordan, M.I., Russell, S. Distance metric learning with application to clustering with side-information. In: *Proceedings of NIPS 15*, MIT Press, Cambridge, MA, 2003, pp.505-512.

[16] Zhang, Z., Zhao, M. B., Chow, T. W. S. Extracting the Informative Constraints for Semi-Supervised Marginal Projections in Multimodal Dimensionality Reduction. In: *Proceedings of IEEE International Joint Conference on Neural Networks*, Brisbane, Australia, 2012.

[17] Jia, Y., Nie, F., Zhang, C. Trace ratio problem revisited. *IEEE Trans. Neural Network*, 2009, 20(4): 729-735.

[18] Zhao, M. B., Zhang, Z., Chow, W. S. Trace Ratio Criterion based Generalized Discriminative Learning for Semi-Supervised Dimensionality Reduction. *Pattern Recognition*, 2012, 45(4):1482-1499.

[19] Chen, L., Liao, H., Ko, M., Lin, J., Yu, G. A new LDA-based face recognition system which can solve the small sample size problem. *Patter Recognition*, 2000, 33(10):1713-1726.

[20] Chen, J., Ye, J., Li, Q. Integrating global and local structures: a least squares framework for dimensionality reduction. In: *Proceedings of CVPR 2007*.

[21] Sugiyama, M. Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *Journal of Machine Learning Research*, 2007, 8:1027-1061.

[22] MATLAB, User's Guide, The MathWorks, Inc., 1994-2001, http://www.mathworks.com.

[23] Song, Y. Q., Nie, F. P., Zhang, C. S. Semi-supervised sub-manifold discriminant analysis. *Pattern Recognition Letters*, 2008, 29:1806-1813.

[24] Baghshah M. S., Shouraki, S. B. Non-linear metric learning using pairwise similarity and dissimilarity constraints and the geometrical structure of data. *Pattern Recognition*, 2010, 43:2982-2992.

[25] Weyrauch, B., Huang, J., Heisele, B., Blanz, V. Component-based Face Recognition with 3D Morphable Models. In: *Proceedings of the 1st IEEE Workshop on Face Processing in Video*, Washington, D.C., 2004.

[26] Zhang, Z., Zhao, M. B., and Chow, T. W. S. Constrained Large Margin Local Projection Algorithms and Extensions for Multimodal Dimensionality Reduction, *Pattern Recognition*, 2012, 45(12):4466-4493.

[27] He, X., Cai, D., Yan, S., Zhang, H. Neighborhood Preserving Embedding, In: *Proceedings of ICCV 2005*, Beijing, China, pp.1208-1213.

[28] Zhang, Z., Chow, T. W. S., and Zhao, M.B. Trace Ratio Optimization Based Semi-Supervised Nonlinear Dimensionality Reduction for Marginal Manifold Visualization. *IEEE Transactions on Knowledge and Data Engineering*, March 2012. Doi.ieeecomputersociety.org/10.1109/TKDE.2012.47.

[29] Mohammad N. T. Dimensionality Reduction Using Neural Networks. In: *Proceedings of the Artificial Neural Networks In Engineering*, Conference, St. Louis, MO, 2007.

[30] Xiang, S. M., Nie, F. P., Zhang, C. S. Learning a Mahalanobis distance metric for data clustering and classification. *Pattern Recognition*, 2008, 41(12):3600-3612.

[31] Schölkopf, B., Smola, A. *Learning with Kernels*, Cambridge, MA, MIT Press, 2002, pp.25-55.

[32] Ye, J. P. Least square linear discriminant analysis. In: *Proceedings of ICML 2007*.

[33] Hull, J. A database for handwritten text recognition research. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1994, 16(5):550-554.

[34] Leibe, B., Schiele, B. Analyzing appearance and contour based methods for object categorization. In: *Proceedings of CVPR 2003*, pp.409-415.

[35] Sun, L., Ceran, B., Ye, J. P. A Scalable Two-Stage Approach for a Class of Dimensionality Reduction Techniques. In: *Proceedings of ACM KDD 2010*, pp.313-322.

[36] Xua,Y., Zhong, A, N., Yang, J., Zhang, D. LPP solution schemes for use with face recognition. *Pattern Recognition*, 2010, 43(12):4165-4176.

[37] Strutz, T. Data Fitting and Uncertainty: A practical introduction to weighted least squares and beyond, Vieweg+ Teubner Verlag, 2010.

[38] Sun, L., Ji, S. W., Ye, J. P. A least squares formulation for canonical correlation analysis. In: *Proceedings of ICML 2008*, Helsinki, Finland, pp.1024-1031.

[39] Kokiopoulou1, E., Chen, J., Saad, Y. Trace optimization and eigenproblems in dimension reduction methods. *Numerical Linear Algebra with Applications*, 2011, 18:565-602.

[40] Roweis, S., Saul, L. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 2000, 290 (5500):2323-2326.

[41] Qi, Z. Q., Tian, Y. J., Shi Y. Laplacian twin support vector machine for semi-supervised classification, *Neural Networks*, 2012(35):46-53.

[42] Chen, H., Li, L. Q., Peng, J. T. Semi-supervised learning based on high density region estimation. *Neural Networks*, 2010(23):812-818.

[43] Venna, J., Kaski, S. Local multidimensional scaling. *Neural Networks*, 2006, (19):889-899.