Information-theoretic Semi-supervised Metric Learning via Entropy Regularization

Gang Niu GANG@SG.CS.TITECH.AC.JP
Department of Computer Science, Tokyo Institute of Technology, Tokyo 152-8552, Japan

Bo Dai BOHR.DAI@GMAIL.COM
Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA

Makoto Yamada YAMADA@SG.CS.TITECH.AC.JP
Masashi Sugiyama SUGI@CS.TITECH.AC.JP
Department of Computer Science, Tokyo Institute of Technology, Tokyo 152-8552, Japan

Abstract

We propose a general information-theoretic approach called SERAPH *(SEmi-supervised metRic leArning Paradigm with Hyper-sparsity)* for metric learning that does not rely upon the manifold assumption. Given the probability parameterized by a Mahalanobis distance, we maximize the entropy of that probability on labeled data and minimize it on unlabeled data following *entropy regularization*, which allows the supervised and unsupervised parts to be integrated in a natural and meaningful way. Furthermore, SERAPH is regularized by encouraging a low-rank projection induced from the metric. The optimization of SERAPH is solved efficiently and stably by an EM-like scheme with the analytical E-Step and convex M-Step. Experiments demonstrate that SERAPH compares favorably with many well-known global and local metric learning methods.

1. Introduction

A good metric for input data is a key factor for many machine learning algorithms. Classical metric learning methods fall into three types: (a) Supervised type requiring class labels (e.g., Sugiyama, 2007); (b) Supervised type requiring weak labels, i.e., $\{\pm 1\}$-valued labels that indicate the similarity/dissimilarity of data pairs (e.g., Weinberger et al., 2005; Davis et al., 2007); (c) Unsupervised type requiring no label information (e.g., Belkin & Niyogi, 2001). Types (a) and (b) have a strict limitation for real-world applications since they need lots of labels. Based on the belief that preserving the geometric structure in an unsupervised manner can be better than relying on the limited labels, semi-supervised metric learning has emerged. To the best of our knowledge, all semi-supervised extensions employ *off-the-shelf* techniques in type (c) such as principal component analysis (Yang et al., 2006; Sugiyama et al., 2010) or manifold embedding (Hoi et al., 2008; Baghshah & Shouraki, 2009; Liu et al., 2010). They can be regarded as propagating labels along an assistant metric by some unsupervised techniques and learning a target metric implicitly in a supervised manner.

However, the target and assistant metrics assume different forms, one Mahalanobis distance defined over a Euclidean space and one geodesic distance over a curved space or a Riemannian manifold. The two metrics also share slightly different goals: the target metric tries to learn a metric so that data in the same class are close and data from different classes are far apart (e.g., Fisher discriminant analysis[1] (Fisher, 1936)), and the assistant one tries to identify and preserve the intrinsic geometric structure (e.g., Laplacian eigenmaps (Belkin & Niyogi, 2001)). Simply putting them together works in practice, but the paradigm is conceptually neither natural nor unified.

In this paper, we propose a semi-supervised metric learning approach SERAPH (SEmi-supervised metRic leArning Paradigm with Hyper-sparsity) as an *information-theoretic alternative* to the manifold-based methods. Our idea is to optimize a metric by optimizing a conditional probability parameterized by that metric. Following *entropy regularization* (Grandvalet & Bengio, 2004), we maximize the entropy of that probability on labeled data, and minimize it

Appearing in *Proceedings of the 29^{th} International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

[1] Note that learning a metric is equivalent to learning a projection in the scenario of dimensionality reduction.

on unlabeled data, which can achieve the sparsity of the posterior distribution (Graça et al., 2009), i.e., the low uncertainty/entropy of unobserved weak labels. Furthermore, we employ *mixed-norm regularization* (Ying et al., 2009) to encourage the sparsity of the projection matrix, i.e., the low rank of the projection matrix induced from the metric. Unifying the posterior sparsity and the projection sparsity brings us to the *hyper-sparsity*. Thanks to this property, the metric learned by SERAPH possesses high discriminability even under a noisy environment.

Our contributions can be summarized as follows. First, we formulate the supervised metric learning problem as an instance of the generalized maximum entropy distribution estimation (Dudík & Schapire, 2006). Second, we propose a semi-supervised extension of the above estimation following entropy regularization (Grandvalet & Bengio, 2004). Notice that our extension is compatible with the manifold-based extension, which means that SERAPH could adopt an additional manifold regularization term.

2. Proposed Approach

In this section, we first formulate the model of SERAPH and then develop the EM-like algorithm to solve the model.

2.1. Notations

Suppose we have a training set $\mathcal{X} = \{x_i \mid x_i \in \mathbb{R}^m\}_{i=1}^n$ that contains n points each with m features. Let the sets of similar and dissimilar data pairs be

$$\mathcal{S} = \{(x_i, x_j) \mid x_i \text{ and } x_j \text{ are similar}\},$$
$$\mathcal{D} = \{(x_i, x_j) \mid x_i \text{ and } x_j \text{ are dissimilar}\}.$$

With some abuse of terminology, we refer to $\mathcal{S} \cup \mathcal{D}$ as the labeled data and

$$\mathcal{U} = \{(x_i, x_j) \mid i \neq j, (x_i, x_j) \notin \mathcal{S} \cup \mathcal{D}\}$$

as the unlabeled data. A weak label $y_{i,j} = 1$ is assigned to $(x_i, x_j) \in \mathcal{S}$, or $y_{i,j} = -1$ to $(x_i, x_j) \in \mathcal{D}$. We abbreviate $\sum_{(x_i,x_j)\in\mathcal{S}\cup\mathcal{D}}$, $\sum_{(x_i,x_j)\in\mathcal{U}}$ and $\sum_{y\in\{1,-1\}}$ as $\sum_{\mathcal{S}\cup\mathcal{D}}$, $\sum_{\mathcal{U}}$ and $\sum_y$. Consider learning a Mahalanobis distance metric for $x, x' \in \mathbb{R}^m$ of the form

$$d(x, x') = \|x - x'\|_A = \sqrt{(x - x')^\top A (x - x')},$$

where $^\top$ is the transpose operator and $A \in \mathbb{R}^{m\times m}$ is a symmetric and positive semi-definite matrix to be learned[2]. The probability $p^A(y \mid x, x')$ of labeling $(x, x') \in \mathbb{R}^m \times \mathbb{R}^m$ with $y = \pm 1$ is parameterized by the matrix A. When applying $p^A(y \mid x, x')$ to (x_i, x_j), it is abbreviated as $p^A_{i,j}(y)$.

[2]In this paper, A is always assumed symmetric positive semi-definite and will not be explicitly written for brevity.

2.2. Basic model

To begin with, we derive a probabilistic model to investigate the conditional probability of $y = \pm 1$ given $(x, x') \in \mathbb{R}^m \times \mathbb{R}^m$. We resort to a parametric form of $p^A(y \mid x, x')$, and will focus on it for the out-of-sample ability.

The *maximum entropy principle* (Jaynes, 1957) suggests that we should choose the probability distribution with the maximum entropy out of all distributions that match the data moments. Let[3]

$$H(p^A_{i,j}) = -\sum\nolimits_y p^A_{i,j}(y) \ln p^A_{i,j}(y)$$

be the entropy of the conditional probability $p^A_{i,j}(y)$, and

$$f(x, x', y; A) : \mathbb{R}^m \times \mathbb{R}^m \times \{+1, -1\} \mapsto \mathbb{R}$$

be a feature function that is convex with respect to A. The constrained optimization problem is

$$\begin{aligned} \max_{A, p^A_{i,j}, \xi} \quad & \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} H(p^A_{i,j}) - \frac{1}{2\gamma}\xi^2 \\ \text{s.t.} \quad & \Big|\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \mathbb{E}_{p^A_{i,j}}[f(x_i, x_j, y; A)] \\ & \qquad - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} f(x_i, x_j, y_{i,j}; A)\Big| \le \xi, \end{aligned} \tag{1}$$

where ξ is a slack variable and $\gamma > 0$ is a regularization parameter. The penalty presumes the Gaussian prior of the expected data moments from the empirical data moments, which is essentially consistent in spirit with the *generalized maximum entropy principle* (Dudík & Schapire, 2006) (see Appendix B.1).

Theorem 1. *The primal solution p^{*A} is given in terms of the dual solution (A^*, κ^*) by*

$$p^{*A}(y \mid x, x') = \frac{\exp(\kappa^* f(x, x', y; A^*))}{Z(x, x'; A^*, \kappa^*)}, \tag{2}$$

where $Z(x, x'; A, \kappa) = \sum_{y'} \exp(\kappa f(x, x', y'; A))$, and (A^*, κ^*) can be obtained by solving the dual problem

$$\begin{aligned} \min_{A,\kappa} \quad & \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln Z(x_i, x_j; A, \kappa) \\ & - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \kappa f(x_i, x_j, y_{i,j}; A) + \frac{\gamma}{2}\kappa^2. \end{aligned} \tag{3}$$

Define the regularized log-likelihood function on labeled data (i.e., on observed weak labels) as

$$\mathcal{L}_1(A, \kappa) = \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln p^A_{i,j}(y_{i,j}) - \frac{\gamma}{2}\kappa^2.$$

Then, for supervised metric learning, the regularized maximum log-likelihood estimation and the generalized maximum entropy estimation are equivalent.[4]

[3]Throughout this paper, we adopt that $0 \ln 0 = 0$.

[4]The proofs of all theorems are in Appendix A.

When considering $f(x, x', y; A)$ that should take moments about the metric information into account, we propose

$$f(x, x', y; A, \eta) = \frac{y}{2}(\|x - x'\|_A^2 - \eta), \tag{4}$$

where $\eta > 0$ is a hyperparameter used as the threshold to separate the sets $\mathcal{S}$ and $\mathcal{D}$ under the target metric $d(x, x')$. Now the probabilistic model (2) becomes

$$p^A(y \mid x, x') = \frac{1}{1 + \exp(-\kappa y(\|x - x'\|_A^2 - \eta))}. \tag{5}$$

For the optimal solution (p^{*A}, A^*, κ^*), we hope for

$$p^{*A}(y_{i,j} \mid x_i, x_j) > 1/2,\ y_{i,j}(\|x_i - x_j\|_{A^*}^2 - \eta) < 0,$$

so there must be $\kappa^* < 0$.

Although we use Eq.(4) as our feature function, other options are available. Please see Appendix C.1 for details.

2.3. Regularization

In this subsection, we extend $\mathcal{L}_1(A, \kappa)$ by entropy regularization to semi-supervised learning. Moreover, we regularize our objective by trace-norm regularization.

Our unsupervised part does not rely upon the manifold assumption and is not in the paradigm of smoothing the projected training data. In order to be integrated with the supervised part more naturally in philosophy, we follow the *minimum entropy principle* (Grandvalet & Bengio, 2004), and hence $p_{i,j}^A$ should have low entropy or uncertainty for $(x_i, x_j) \in \mathcal{U}$. Roughly speaking, the resultant discriminative models prefer peaked distributions on unlabeled data, which carries out a probabilistic *low-density separation*. Subsequently, according to Grandvalet & Bengio (2004), our optimization becomes

$$\begin{aligned}\max_{A,\kappa}\ \mathcal{L}_2(A, \kappa) = &\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln p_{i,j}^A(y_{i,j}) - \frac{\gamma}{2}\kappa^2 \\ &+ \mu \sum\nolimits_{\mathcal{U}} \sum\nolimits_y p_{i,j}^A(y) \ln p_{i,j}^A(y),\end{aligned}$$

where $\mu \geq 0$ is a regularization parameter.

In addition, we hope for the dimensionality reduction ability by encouraging a low-rank projection induced from A. This is helpful in dealing with corrupted data or data distributed intrinsically in a low-dimensional subspace. It is known that the trace is a convex relaxation of the rank for a matrix, so we revise our optimization problem into

$$\begin{aligned}\max_{A,\kappa}\ \mathcal{L}(A, \kappa) = &\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln p_{i,j}^A(y_{i,j}) - \frac{\gamma}{2}\kappa^2 \\ &+ \mu \sum\nolimits_{\mathcal{U}} \sum\nolimits_y p_{i,j}^A(y) \ln p_{i,j}^A(y) - \lambda \operatorname{tr}(A),\end{aligned} \tag{6}$$

where $\operatorname{tr}(A)$ is the trace of A, and $\lambda \geq 0$ is a regularization parameter.

Optimization (6) is the final model of SERAPH, and we say that it is equipped with the hyper-sparsity when both μ and λ are positive. SERAPH possesses standard kernel and manifold extensions. For more information, please refer to Appendix C.2 and C.3.

2.4. Algorithm

From now on we will simplify the model (6) and derive a practical algorithm. First, we eliminate κ from (6), thanks to the fact that we use a simple feature function (4) in (1).

Theorem 2. *Define the simplified optimization problem as*[5]

$$\begin{aligned}\max_{A}\ \hat{\mathcal{L}}(A) = &\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln \hat{p}_{i,j}^A(y_{i,j}) \\ &+ \mu \sum\nolimits_{\mathcal{U}} \sum\nolimits_y \hat{p}_{i,j}^A(y) \ln \hat{p}_{i,j}^A(y) - \hat{\lambda} \operatorname{tr}(A),\end{aligned} \tag{7}$$

where the simplified probabilistic model is

$$\hat{p}^A(y \mid x, x') = \frac{1}{1 + \exp(y(\|x - x'\|_A^2 - \hat{\eta}))}. \tag{8}$$

Let $\hat{A}$ and (A^*, κ^*) be the optimal solutions to (7) and (6), respectively. Then, there exist well-defined hyperparameters $\hat{\eta}$ and $\hat{\lambda}$, such that $\hat{A}$ is equivalent to A^* with respect to $d(x, x')$, and the resulting $\hat{p}^A(y \mid x, x')$ parameterized by $\hat{A}$ and $\hat{\eta}$ is identical to the original $p^A(y \mid x, x')$ parameterized by A^*, κ^* and η.

Remark 1. After the simplification, γ is dropped, η and λ are modified, but the regularization parameter μ remains the same, which means that the tradeoff between the supervised and unsupervised parts has not been affected.

Optimization (7) could be directly solved by the gradient projection method (Polyak, 1967), even though it is nonconvex. Nevertheless, we would like to pose it as an EM-like iterative scheme to access the derandomization by the initial solution, the stability for the gradient update, and the insensitivity to the step size, just to name a few of the gained algorithmic properties.

The EM-like algorithm runs as follows. In the beginning, we initialize a nonparametric probability $q(y \mid x_i, x_j)$, and then the M-Step and the E-Step get executed repeatedly until the stopping conditions are satisfied.

At the t-th E-Step, similarly to Graça et al. (2009), we have for each pair $(x_i, x_j) \in \mathcal{U}$ that

$$\min_q\ \mathrm{KL}(q \,||\, p_{i,j}^A) + \mu \mathbb{E}_q[-\ln p_{i,j}^A(y)], \tag{9}$$

where KL is the Kullback-Leibler divergence, and $p_{i,j}^A$ is parameterized by the metric $A^{(t)}$ found at the last M-Step. Optimization (9) can be solved analytically.

[5] The new functions and parameters are denoted by $\hat{}$ within this theorem for the sake of clarity.

Theorem 3. *The solution to* (9) *is given by*

$$q(y \mid x_i, x_j) = \frac{p^A_{i,j}(y) \exp(\mu \ln p^A_{i,j}(y))}{\sum_{y'} p^A_{i,j}(y') \exp(\mu \ln p^A_{i,j}(y'))}. \quad (10)$$

On the other hand, at the t-th M-Step, we find new metric $A^{(t)}$ through the probability $q(y \mid x_i, x_j)$ which is generated in the last E-Step and only defined for $(x_i, x_j) \in \mathcal{U}$:

$$\max_A \mathcal{F}(A) = \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln p^A_{i,j}(y_{i,j}) + \mu \sum\nolimits_{\mathcal{U}} \sum\nolimits_y q(y \mid x_i, x_j) \ln p^A_{i,j}(y) - \lambda \operatorname{tr}(A). \quad (11)$$

It could be solved by the gradient projection method without worry about local maxima using the calculation of $\nabla\mathcal{F}$ given by

$$\nabla\mathcal{F}(A) = -\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} y_{i,j} \left(1 - p^A_{i,j}(y_{i,j})\right) x_{i,j} - \mu \sum\nolimits_{\mathcal{U}} \sum\nolimits_y y q(y \mid x_i, x_j) \left(1 - p^A_{i,j}(y)\right) x_{i,j} - \lambda I_m,$$

where $x_{i,j} = (x_i - x_j)(x_i - x_j)^\top$, since the convexity of the feature function $f(x, x', y; A)$ with respect to A implies the convexity of the objective $\mathcal{F}(A)$.

A remarkable property of $\mathcal{F}(A)$ is that its gradient is uniformly bounded, regardless of the scale of A, i.e., the magnitude of $\operatorname{tr}(A)$.

Theorem 4. *The objective $\mathcal{F}(A)$ is Lipschitz continuous, and the best Lipschitz constant $\operatorname{Lip}_{\|\cdot\|_F}(\mathcal{F})$ with respect to the Frobenius norm $\|\cdot\|_F$ satisfies*

$$\operatorname{Lip}_{\|\cdot\|_F}(\mathcal{F}) \le (\#\mathcal{S} + \#\mathcal{D} + \mu\#\mathcal{U})(\operatorname{diam}(\mathcal{X}))^2 + \lambda m, \quad (12)$$

where $\operatorname{diam}(\mathcal{X}) = \max_{x_i, x_j \in \mathcal{X}} \|x_i - x_j\|_2$ is the diameter of $\mathcal{X}$, and $\#$ measures the cardinality of a set.

In our current implementation, the initial solution is $q(-1 \mid x_i, x_j) = 1$, which means that we treat all unlabeled pairs as dissimilar pairs. The overall asymptotic time complexity is $O(n^2 m + m^3)$ in which the stopping criteria of the M-Step and the whole EM-like iteration are ignored. Discussions about the computational complexity and the fast implementation can be found in Appendix D.

3. Discussions

In this section, we discuss the sparsity issues, namely, we can obtain the *posterior sparsity* (Graça et al., 2009) by entropy regularization and the *projection sparsity* (Ying et al., 2009) by trace-norm regularization.

By a 'sparse' posterior distribution, we mean that the uncertainty (i.e., the entropy or variance) is low. See Figure 1 as an example. Recall that supervised metric learning aims

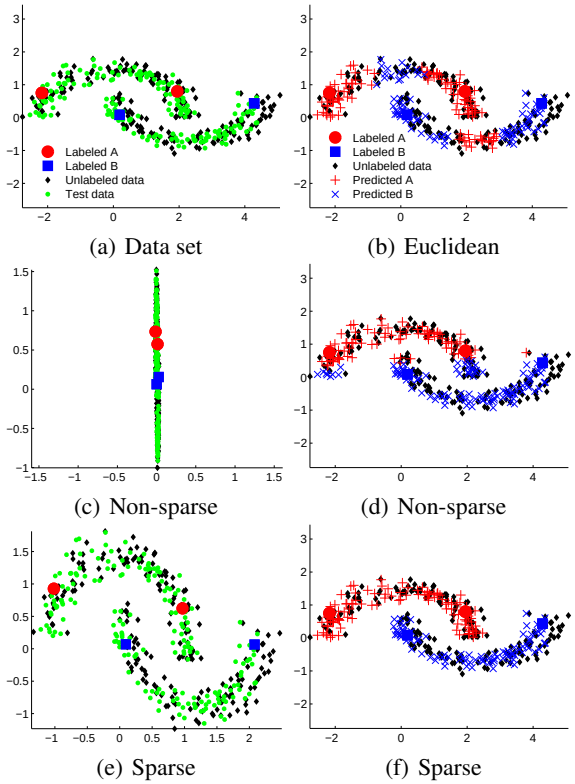

Figure 1. Sparse vs. non-sparse posterior distributions. Six weak labels are constructed according to four class labels. The left three panels show the original data and the projected data by metrics learned with/without the posterior sparsity. The right three panels exhibit one-nearest-neighbor classification results based on the Euclidean distance and two learned metrics.

at a metric under which data in the same class are close and data from different classes are far apart. This results in the metric which ignores the horizontal feature and focuses on the vertical feature. However, the vertical feature is important, and taking care of the posterior sparsity would lead to a better metric as illustrated in (e) and (f). Therefore, we prefer taking the posterior sparsity into account in addition to the aforementioned goal, and then the risk of overfitting weakly labeled data can be significantly reduced.

We can rewrite $\mathcal{L}_2(A, \kappa)$ as a soft posterior regularization (PR) objective (Graça et al., 2009). Let the auxiliary feature function be $g(x, x', y) = -\ln p^A(y \mid x, x')$, then maximizing $\mathcal{L}_2(A, \kappa)$ is equivalent to

$$\max_{A,\kappa} \mathcal{L}_1(A, \kappa) - \mu \sum\nolimits_{\mathcal{U}} \mathbb{E}_{p^A_{i,j}}[g(x_i, x_j, y)]. \quad (13)$$

On the other hand, according to optimization (7) of Graça et al. (2009), the soft PR objective should take a form as

$$\max_{A,\kappa} \mathcal{L}_1(A, \kappa) - \min_q \left(\operatorname{KL}(q \,||\, p^A) + \mu \sum\nolimits_{\mathcal{U}} \xi_{i,j}\right) \quad \text{s.t. } \mathbb{E}_q[g(x_i, x_j, y)] \le \xi_{i,j}, \forall (x_i, x_j) \in \mathcal{U}, \quad (14)$$

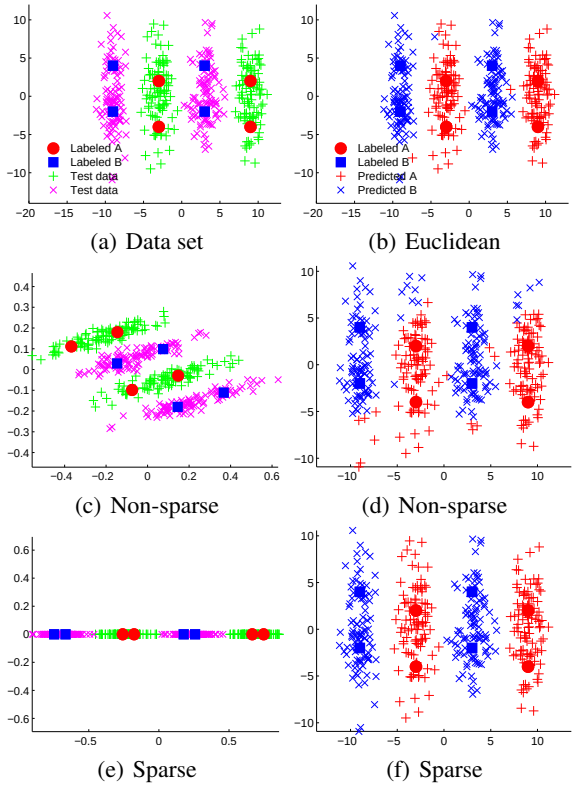

Figure 2. Sparse vs. non-sparse projections. The settings and the layout of panels are similar to Figure 1.

where $\xi_{i,j}$ are slack variables. Since q is unconstrained, we can optimize it with respect to fixed A and κ. It is easy to see that q should be p^A (restricted on $\mathcal{U}$), so the KL term is zero and the expectation term is the entropy, which implies the equivalence of optimizations (13) and (14).

Besides the above posterior sparsity, we also hope for the projection sparsity, which may guide the learned metric to better generalization performance. See Figure 2 as an example of its effectiveness, where the horizontal feature is informative and the vertical feature is useless.

The underlying technique is the mixed-norm regularization (Argyriou et al., 2006). Denote the $\ell_{(2,1)}$-norm of a symmetric matrix M as $\|M\|_{(2,1)} = \sum_{k=1}^{m}(\sum_{k'=1}^{m} M_{k,k'}^2)^{1/2}$. Similarly to Ying et al. (2009), let $P \in \mathbb{R}^{m\times m}$ be a projection, and $W = P^\top P$ be the metric induced from P. Let the i-th column of P and W be P_i and W_i. If P_i is identically zero, the i-th component of x has no contribution to $z = Px$. Since the column-wise sparsity of W and P are equivalent, we can penalize $\|W\|_{(2,1)}$ to reach the column-wise sparsity of P.

Nevertheless, this is feature selection rather than dimensionality reduction. Recall that the goal is to select a few most representative directions of input data which are not restricted to the coordinate axes. The solution is to pick an extra transformation $V \in \mathcal{O}^m$ to rotate x before the projection where $\mathcal{O}^m$ is the set of orthonormal matrices of size m, and add V to the optimization variables. Consequently, we penalize $\|W\|_{(2,1)}$, project x to $z = PVx$, and since $A = (PV)^\top(PV) = V^\top WV$, we arrive at

$$\max_{A,\kappa,W,V} \quad \mathcal{L}_2(A,\kappa) - \lambda\|W\|_{(2,1)} \tag{15}$$
$$\text{s.t.} \quad A = V^\top WV, W = W^\top, W \succeq 0, V \in \mathcal{O}^m.$$

The equivalence of optimizations (6) and (15) is guaranteed by Lemma 1 of Ying et al. (2009).

Moreover, there is another justification based on the *information maximization principle* (Gomes et al., 2010). Please see Appendix B.2 for details.

4. Related Works

Xing et al. (2002) initiated metric learning based on pairwise similarity/dissimilarity constraints by global distance metric learning (GDM). Several excellent metric learning methods have been developed in the last decade, including neighborhood component analysis (NCA; Goldberger et al., 2004), large margin nearest neighbor classification (LMNN; Weinberger et al., 2005), and information-theoretic metric learning (ITML; Davis et al., 2007).

Both ITML and SERAPH are information-theoretic, but the ideas and models are quite different. ITML defines a generative model $p^A(x) = \exp(-\frac{1}{2}\|x-\mu\|_A^2)/Z$, where μ is unknown mean value and Z is a normalizing constant. Compared with GDM, ITML regularizes the KL-divergence between $p^{A_0}(x)$ and $p^A(x)$, and transforms this term to a Log-Det regularization. By specifying $A_0 = \frac{1}{n}I_m$, it becomes the maximum entropy estimation of $p^A(x)$. Thus, it prefers the metric close to the Euclidean distance. SERAPH also follows the maximum entropy principle, but the probabilistic model $p^A(y \mid x, x')$ is discriminative.

A probabilistic GDM was designed intuitively as a baseline in the experimental part of Yang et al. (2006). It is a special case of our supervised part. In fact, SERAPH is much more general. Please refer to Section 2.2 for details.

Subsequently, local distance metric learning (LDM; Yang et al., 2006) is the pioneer of semi-supervised metric learning, which assumes that the eigenvectors of A are the principal components of training data. Hoi et al. (2008) combines manifold regularization to the min-max principle of GDM based on Belkin & Niyogi (2001), and Baghshah & Shouraki (2009) shows that Roweis & Saul (2000) is also useful for semi-supervised metric learning. Liu et al. (2010) brings the element-wise sparsity to Hoi et al. (2008).

The manifold extension described in Appendix C.3 can be attached to all metric learning methods, whereas our unsu-

Table 1. Specification of benchmark data sets.

	#classes	#features (m)	#training (n)	#test	#class labels	$\mathbb{E}\#\mathcal{S}$	$\mathbb{E}\#\mathcal{D}$	$\#\mathcal{U}$
iris	3	4	100	38	10	15.10	29.90	4905
wine	3	13	100	78	10	13.98	31.02	4905
ionosphere	2	34	100	251	20	97.50	92.50	4760
balance	3	4	100	465	10	20.38	24.62	4905
breast cancer	2	30	100	469	10	23.54	21.46	4905
diabetes	2	8	100	668	10	23.02	21.98	4905
$\text{USPS}_{1-5,20}$	5	64	100	2500	10	5	40	4905
$\text{USPS}_{1-5,40}$	5	64	200	2500	20	30	160	19710
$\text{USPS}_{1-10,20}$	10	64	200	2500	20	10	180	19710
$\text{USPS}_{1-10,40}$	10	64	400	2500	40	60	720	79020
$\text{MNIST}_{1,7}$	2	196	100	1000	4	2	4	4944
$\text{MNIST}_{3,5,8}$	3	196	150	1500	9	9	27	11139

pervised part applies to probabilistic methods only. However, any probabilistic method with an explicit expression of the posterior distribution adopts two semi-supervised extensions, while deterministic methods such as LMNN cannot benefit from entropy regularization.

Due to limited space, we leave out sparse metric learning and robust metric learning. Instead, we recommend Huang et al. (2009) and Huang et al. (2010) for the latest reviews of sparse and robust metric learning respectively.

5. Experiments

5.1. Setup

We compared SERAPH with the Euclidean distance, four famous supervised and two representative semi-supervised metric learning methods[6]: global distance metric learning (GDM; Xing et al., 2002), neighborhood component analysis (NCA; Goldberger et al., 2004), large margin nearest neighbor classification (LMNN; Weinberger et al., 2005), information-theoretic metric learning (ITML; Davis et al., 2007), local distance metric learning (LDM; Yang et al., 2006), and manifold Fisher discriminant analysis (MFDA; Baghshah & Shouraki, 2009).

Table 1 describes the specification of the data sets used in our experiments. The top six data sets (i.e., iris, wine, ionosphere, balance, breast cancer, and diabetes) come from the *UCI machine learning repository*[7], while the *USPS* and *MNIST* come from the homepage of the late Sam Roweis[8]. Gray-scale images of handwritten digits are downsampled to 8×8 and 14×14 pixel resolution resulting in 64- and 196-dimensional vectors for USPS and MNIST. The symbol $\text{USPS}_{1-5,20}$ means 20 training data from each of the first 5 classes, $\text{USPS}_{1-10,40}$ means 40 training data from each of all 10 classes, $\text{MNIST}_{1,7}$ means digits 1 versus 7, and so forth. Note that in the last two tasks, the dimensionality of data is greater than the size of all training data.

In our experiments, all methods were repeatedly run on 50 random samplings. For each random sampling, class labels of the first few data were revealed, and the sets $\mathcal{S}$ and $\mathcal{D}$ were constructed according to these revealed class labels. The sizes of $\mathcal{S}$, $\mathcal{D}$ and $\mathcal{U}$ were fixed for all samplings of USPS and MNIST but random for the samplings of UCI data sets. We measured the performance of the one-nearest-neighbor classifiers based on the learned metrics as well as the computation time for learning the metrics.

Four settings of SERAPH were included in our experiments (except on two artificial data sets): SERAPH_{none} stands for $\mu = \lambda = 0$, SERAPH_{post} for $\mu = \frac{\#(\mathcal{S}\cup\mathcal{D})}{\#\mathcal{U}}$ and $\lambda = 0$, SERAPH_{proj} for $\mu = 0$ and $\lambda = 1$, and SERAPH_{hyper} for $\mu = \frac{\#(\mathcal{S}\cup\mathcal{D})}{\#\mathcal{U}}$ and $\lambda = 1$. We fixed $\eta = 1$ for simplicity.

There was no cross-validation for each random sampling, otherwise the learned metrics would be highly dependent upon the final classifier, and also because of the large variance of the classification performance given the limited supervised information. The hyperparameters of other methods, e.g., the number of reduced dimensions, the number of nearest neighbors, and the percentage of principal components, were selected as the best value based on another 10 random samplings if default values or heuristics were not provided by the original authors.

5.2. Results

Figures 1 and 2 had previously displayed the visually comprehensive results of the sparsity regularization on two artificial data sets respectively. Subfigures (c) and (d) in both

[6]We downloaded the codes of all baseline methods from the official websites provided by the original authors except MFDA.

[7]http://archive.ics.uci.edu/ml/.

[8]http://cs.nyu.edu/~roweis/data.html.

Table 2. Means with standard errors of the nearest-neighbor misclassification rate (in %) on UCI, USPS and MNIST benchmarks. For each data set, the best method and comparable ones based on the unpaired t-test at the significance level 5% are highlighted in boldface.

	iris	wine	ionosphere	balance	breast cancer	diabetes
EUCLIDEAN	9.58 ± 0.73	12.93 ± 0.83	23.60 ± 0.89	27.15 ± 0.75	14.11 ± 1.07	32.94 ± 0.65
GDM	8.95 ± 0.71	11.52 ± 0.77	$\mathbf{20.82 \pm 0.82}$	22.89 ± 1.08	11.86 ± 0.83	$\mathbf{30.73 \pm 0.59}$
NCA	10.32 ± 0.83	15.03 ± 1.12	26.68 ± 0.82	32.97 ± 1.31	14.63 ± 1.09	32.95 ± 0.65
LMNN	9.81 ± 0.79	14.83 ± 0.97	22.25 ± 0.75	24.00 ± 1.34	13.86 ± 0.84	32.02 ± 0.60
ITML	$\mathbf{5.57 \pm 0.53}$	$\mathbf{8.22 \pm 0.66}$	$\mathbf{20.35 \pm 0.64}$	22.04 ± 0.80	$\mathbf{9.60 \pm 0.49}$	$\mathbf{31.21 \pm 0.73}$
LDM	7.27 ± 0.72	17.21 ± 1.41	24.54 ± 0.92	$\mathbf{21.22 \pm 0.93}$	14.85 ± 0.92	34.33 ± 0.60
MFDA	6.58 ± 0.54	11.55 ± 1.03	23.66 ± 0.91	23.61 ± 1.00	11.21 ± 0.80	31.64 ± 0.62
SERAPH$_{none}$	6.21 ± 0.48	$\mathbf{8.13 \pm 0.58}$	$\mathbf{19.70 \pm 0.43}$	$\mathbf{20.25 \pm 0.64}$	11.39 ± 0.49	$\mathbf{29.86 \pm 0.61}$
SERAPH$_{post}$	$\mathbf{4.79 \pm 0.37}$	$\mathbf{7.46 \pm 0.51}$	$\mathbf{19.64 \pm 0.45}$	$\mathbf{19.98 \pm 0.67}$	11.33 ± 0.50	$\mathbf{29.87 \pm 0.57}$
SERAPH$_{proj}$	$\mathbf{5.79 \pm 0.54}$	$\mathbf{7.39 \pm 0.50}$	$\mathbf{19.53 \pm 0.46}$	$\mathbf{20.94 \pm 0.64}$	$\mathbf{9.61 \pm 0.49}$	$\mathbf{30.43 \pm 0.65}$
SERAPH$_{hyper}$	$\mathbf{5.31 \pm 0.43}$	$\mathbf{7.38 \pm 0.49}$	$\mathbf{19.33 \pm 0.42}$	$\mathbf{20.15 \pm 0.63}$	$\mathbf{10.04 \pm 0.52}$	$\mathbf{30.02 \pm 0.63}$
	USPS$_{1-5,20}$	USPS$_{1-5,40}$	USPS$_{1-10,20}$	USPS$_{1-10,40}$	MNIST$_{1,7}$	MNIST$_{3,5,8}$
EUCLIDEAN	36.63 ± 0.80	28.43 ± 0.60	49.17 ± 0.50	39.30 ± 0.39	10.42 ± 0.67	$\mathbf{37.30 \pm 0.81}$
GDM	37.62 ± 0.77	-	-	-	-	-
NCA	37.55 ± 0.84	28.39 ± 0.60	57.01 ± 0.82	49.21 ± 0.66	10.42 ± 0.67	$\mathbf{37.75 \pm 0.92}$
LMNN	36.43 ± 0.78	28.93 ± 0.61	48.12 ± 0.57	43.68 ± 0.58	9.99 ± 0.71	$\mathbf{36.49 \pm 0.82}$
ITML	35.86 ± 0.74	27.40 ± 0.65	47.40 ± 0.60	39.44 ± 0.57	9.94 ± 0.69	40.83 ± 0.93
LDM	47.19 ± 1.51	32.52 ± 0.85	59.13 ± 0.73	43.18 ± 0.53	14.54 ± 1.41	45.53 ± 1.16
MFDA	42.52 ± 0.82	28.82 ± 0.62	52.13 ± 0.59	37.78 ± 0.50	9.35 ± 0.72	42.39 ± 0.92
SERAPH$_{none}$	36.08 ± 0.75	27.41 ± 0.60	47.29 ± 0.58	38.36 ± 0.55	9.97 ± 0.71	$\mathbf{36.44 \pm 0.84}$
SERAPH$_{post}$	35.79 ± 0.75	27.37 ± 0.60	47.12 ± 0.58	38.20 ± 0.55	10.98 ± 0.79	$\mathbf{36.45 \pm 0.84}$
SERAPH$_{proj}$	36.01 ± 0.75	$\mathbf{26.17 \pm 0.57}$	47.42 ± 0.62	35.42 ± 0.54	9.28 ± 0.72	$\mathbf{36.55 \pm 0.80}$
SERAPH$_{hyper}$	$\mathbf{32.79 \pm 0.77}$	$\mathbf{25.26 \pm 0.56}$	$\mathbf{44.89 \pm 0.58}$	$\mathbf{33.41 \pm 0.47}$	$\mathbf{7.61 \pm 0.57}$	$\mathbf{35.71 \pm 0.84}$

figures were obtained by GDM, while (e) and (f) were generated by SERAPH with $\mu = 10 \cdot \frac{\#(\mathcal{S} \cup \mathcal{D})}{\#\mathcal{U}}$, $\lambda = 0$ in Figure 1 and $\mu = 0, \lambda = 300$ in Figure 2. We can see from Figures 1 and 2 that SERAPH improved supervised global metric learning dramatically by the sparsity regularization.

The experimental results of the one-nearest-neighbor classification are reported in Table 2 (GDM was sometimes very slow and excluded from the comparison). SERAPH is fairly promising, especially with the hyper-sparsity ($\mu = \frac{\#(\mathcal{S} \cup \mathcal{D})}{\#\mathcal{U}}$ and $\lambda = 1$). It was best or tie over all tasks, and often statistically significantly better than others on UCI data sets except ITML. It was better than all other methods statistically significantly on USPS, and SERAPH$_{hyper}$ outperformed both SERAPH$_{post}$ and SERAPH$_{proj}$. Moreover, it improved the accuracy even on the ill-posed MNIST tasks, though the improvement was insignificant on MNIST$_{3,5,8}$. In a word, SERAPH can reduce the risk of overfitting weakly labeled data with the help of unlabeled data, and hence our sparsity regularization would be reasonable and practical.

In vivid contrast with SERAPH that exhibited nice generalization capability, supervised methods might learn a metric even worse than the Euclidean distance due to overfitting problems, especially NCA that optimized the leave-one-out performance based on such limited label information. The powerful LMNN did not behave satisfyingly, since it was hardly fulfilled to find a lot of neighbors belonging to the same class within labeled data. ITML was the second best method though it can only access weakly labeled data, but it became less useful for difficult tasks. On the other hand, we observed that LDM might fail when the principal components of training data were not close to the eigenvectors of the target matrix, and MFDA might fail if the amount of training data cannot recover the underlying manifold well.

An observation is that the global metric learning often outperformed the local one, if the supervised information was insufficient. This phenomenon indicates that the local metric learning tends to fit the local neighborhood information exceedingly and then suffers from overfitting problems.

Finally, we report in Figure 3 the computation time of each algorithm on each task (excluding GDM). Generally speaking, SERAPH was the second fastest method, and the fastest MFDA involves only some matrix multiplication and a single eigen-decomposition. Improvements may be expected if we program in Matlab with C/C++.

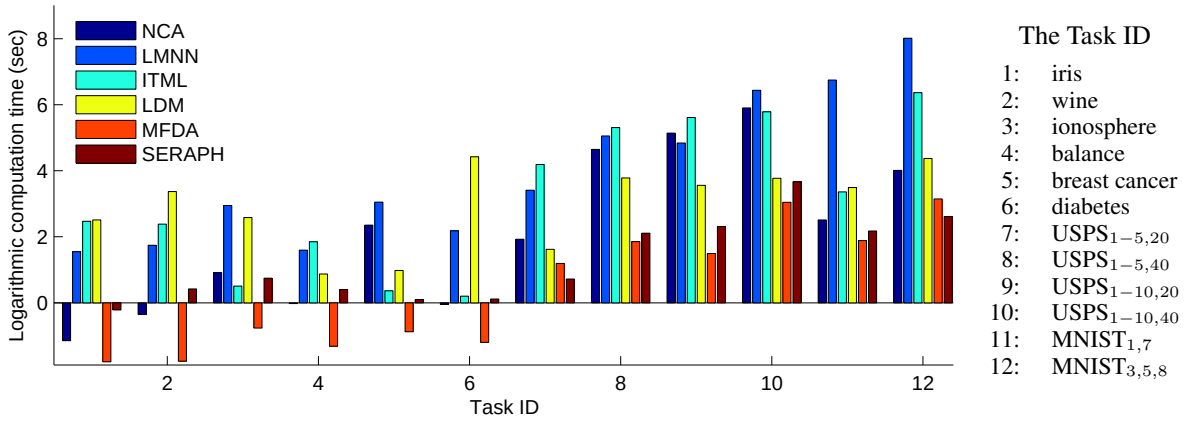

Figure 3. Computation time (per run) of different metric learning algorithms.

6. Conclusions

In this paper, we proposed an information-theoretic semi-supervised metric learning approach SERAPH as an alternative to the manifold-based methods. The generalized maximum entropy estimation for supervised metric learning was our foundation. Then a semi-supervised extension that can achieve the posterior sparsity was obtained via entropy regularization. Moreover, we enforced a trace-norm regularization that can reach the projection sparsity. The resulting optimization was solved by an EM-like scheme with several nice algorithmic properties, and the learned metric had high discriminability even under a noisy environment.

Experiments on benchmark data sets showed that SERAPH often outperformed state-of-the-art fully-/semi-supervised metric learning methods given only limited supervised information. A final note is that in our experiments the posterior and projection sparsity were demonstrated to be very helpful for high-dimensional data *if and only if* they were combined with each other, i.e., integrated into the hyper-sparsity. An in-depth study of this interaction is left as our future work.

Acknowledgments

The authors would like to thank anonymous reviewers for helpful comments. GN is supported by the MEXT scholarship No.103250, MY is supported by the JST PRESTO program, and MS is supported by the FIRST program.

References

Argyriou, A., Evgeniou, T., and Pontil, M. Multi-task feature learning. In *NIPS*, 2006.

Baghshah, M. and Shouraki, S. Semi-supervised metric learning using pairwise constraints. In *IJCAI*, 2009.

Belkin, M. and Niyogi, P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, 2001.

Bellare, K., Druck, G., and McCallum, A. Alternating projections for learning with expectation constraints. In *UAI*, 2009.

Davis, J., Kulis, B., Jain, P., Sra, S., and Dhillon, I. Information-theoretic metric learning. In *ICML*, 2007.

Dudík, M. and Schapire, R. E. Maximum entropy distribution estimation with generalized regularization. In *COLT*, 2006.

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

Goldberger, J., Roweis, S., Hinton, G., and Salakhutdinov, R. Neighbourhood components analysis. In *NIPS*, 2004.

Gomes, R., Krause, A., and Perona, P. Discriminative clustering by regularized information maximization. In *NIPS*, 2010.

Graça, J., Ganchev, K., Taskar, B., and Pereira, F. Posterior vs. parameter sparsity in latent variable models. In *NIPS*, 2009.

Grandvalet, Y. and Bengio, Y. Semi-supervised learning by entropy minimization. In *NIPS*, 2004.

Hoi, S., Liu, W., and Chang, S.-F. Semi-supervised distance metric learning for collaborative image retrieval. In *CVPR*, 2008.

Huang, K., Ying, Y., and Campbell, C. GSML: A unified framework for sparse metric learning. In *ICDM*, 2009.

Huang, K., Jin, R., Xu, Z., and Liu, C. Robust metric learning by smooth optimization. In *UAI*, 2010.

Jaynes, E. T. Information theory and statistical mechanics. *Physical Review*, 106(4):620–630, 1957.

Liu, W., Ma, S., Tao, D., Liu, J., and Liu, P. Semi-supervised sparse metric learning using alternating linearization optimization. In *KDD*, 2010.

Polyak, B. T. A general method for solving extremal problems (in Russian). *Soviet Mathematics Doklady*, 174(1):33–36, 1967.

Roweis, S. and Saul, L. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

Sugiyama, M. Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis. *Journal of Machine Learning Research*, 8:1027–1061, 2007.

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

Weinberger, K., Blitzer, J., and Saul, L. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, 2005.

Xing, E., Ng, A., Jordan, M. I., and Russell, S. Distance metric learning with application to clustering with side-information. In *NIPS*, 2002.

Yang, L., Jin, R., Sukthankar, R., and Liu, Y. An efficient algorithm for local distance metric learning. In *AAAI*, 2006.

Ying, Y., Huang, K., and Campbell, C. Sparse metric learning via smooth optimization. In *NIPS*, 2009.

Appendix: Supplementary Material

A. Proofs

A.1. Proof of Theorem 1

Proof. To simplify our notations and make the proof compact, let us denote

$$\begin{aligned} p^+_{i,j} &\triangleq p^A_{i,j}(+1), \\ p^-_{i,j} &\triangleq p^A_{i,j}(-1), \\ f^+_{i,j} &\triangleq f(x_i, x_j, +1), \\ f^-_{i,j} &\triangleq f(x_i, x_j, -1), \\ \tilde{f}_{i,j} &\triangleq f(x_i, x_j, y_{i,j}), \end{aligned}$$

respectively.

Foremost, expand optimization (1) into its complete form:

$$\begin{aligned} \max_{A, p^A_{i,j}, \xi} \; & -\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} \ln p^+_{i,j} + p^-_{i,j} \ln p^-_{i,j}) - \frac{1}{2\gamma}\xi^2 \\ \text{s.t.} \; & \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} f^+_{i,j} + p^-_{i,j} f^-_{i,j}) - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \tilde{f}_{i,j} - \xi \le 0, \\ & \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \tilde{f}_{i,j} - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} f^+_{i,j} + p^-_{i,j} f^-_{i,j}) - \xi \le 0, \\ & p^+_{i,j} + p^-_{i,j} = 1, \forall (x_i, x_j) \in \mathcal{S}\cup\mathcal{D}. \end{aligned}$$

The terms $\ln p^+_{i,j}$ and $\ln p^-_{i,j}$ in the objective function plus $p^+_{i,j} + p^-_{i,j} = 1$ in the constraints have already implied that

$$0 \le p^+_{i,j}, p^-_{i,j} \le 1.$$

By introducing dual variables $\kappa_1 \ge 0, \kappa_2 \ge 0$ for the first and second constraints, and $\delta_{i,j} \in \mathbb{R}$ for the third group of constraints, the Lagrangian is expressed as

$$\begin{aligned} & L(A, p^A_{i,j}, \xi, \kappa_1, \kappa_2, \delta_{i,j}) = \\ & \quad -\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} \ln p^+_{i,j} + p^-_{i,j} \ln p^-_{i,j}) - \frac{1}{2\gamma}\xi^2 \\ & \quad - \kappa_1 \left(\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} f^+_{i,j} + p^-_{i,j} f^-_{i,j}) - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \tilde{f}_{i,j} - \xi \right) \\ & \quad - \kappa_2 \left(\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \tilde{f}_{i,j} - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} f^+_{i,j} + p^-_{i,j} f^-_{i,j}) - \xi \right) \\ & \quad + \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \delta_{i,j} (p^+_{i,j} + p^-_{i,j} - 1). \end{aligned}$$

Differentiating the function $L(A, p^A_{i,j}, \xi, \kappa_1, \kappa_2, \delta_{i,j})$ with respect to $p^+_{i,j}$ and $p^-_{i,j}$, and equating the derivatives to zero will give us

$$\begin{aligned} \ln p^+_{i,j} &= \kappa f^+_{i,j} + \delta_{i,j} - 1, \\ \ln p^-_{i,j} &= \kappa f^-_{i,j} + \delta_{i,j} - 1, \end{aligned} \tag{16}$$

where $\kappa = \kappa_2 - \kappa_1 \in \mathbb{R}$. Note that Eq.(16) says that

$$\frac{p^+_{i,j}}{p^-_{i,j}} = \exp(\kappa f^+_{i,j} - \kappa f^-_{i,j}). \tag{17}$$

Hence Eq.(2) follows with

$$\delta_{i,j} = 1 - \ln Z^A_{i,j}. \tag{18}$$

Next, differentiating $L(A, p^A_{i,j}, \xi, \kappa_1, \kappa_2, \delta_{i,j})$ with respect to ξ and equating the derivative to zero will give us

$$\xi = \gamma(\kappa_1 + \kappa_2). \tag{19}$$

According to the Karush-Kuhn-Tucker condition about the dual complementary slackness, i.e.,

$$\begin{aligned} \kappa_1 \left(\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} f^+_{i,j} + p^-_{i,j} f^-_{i,j}) - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \tilde{f}_{i,j} - \xi \right) = 0, \\ \kappa_2 \left(\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \tilde{f}_{i,j} - \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} (p^+_{i,j} f^+_{i,j} + p^-_{i,j} f^-_{i,j}) - \xi \right) = 0, \end{aligned}$$

we know that $\kappa_1 \kappa_2 = 0$, which means

$$(\kappa_1 + \kappa_2)^2 = (\kappa_1 - \kappa_2)^2 = \kappa^2. \tag{20}$$

Substituting Eq.(16)-Eq.(20) into $L(A, p^A_{i,j}, \xi, \kappa_1, \kappa_2, \delta_{i,j})$ accomplishes dual problem (3).

Finally, the optimization of the regularized maximum log-likelihood estimation is

$$\max_{A,\kappa} \; \mathcal{L}_1(A, \kappa).$$

By plugging the probabilistic model (2) into it we get optimization (3) exactly, which is the dual problem of the generalized maximum entropy estimation for supervised metric learning defined in optimization (1). □

A.2. Proof of Theorem 2

Proof. The proof is constructive.

As mentioned before, there must be $\kappa^* < 0$. Moreover, $\kappa^* > -\infty$ and $\mathrm{tr}(A^*) < +\infty$, since they are penalized in optimization (6). Let

$$\begin{aligned} \hat{A} &= -\kappa^* A^*, \\ \hat{\eta} &= -\kappa^* \eta, \\ \hat{\lambda} &= -\lambda/\kappa^*. \end{aligned}$$

Then $\hat{\eta}$ and $\hat{\lambda}$ are well-defined hyperparameters as finite positive real numbers, and $\hat{A}$ is a feasible solution to (7) as a finite trace symmetric positive semi-definite matrix.

Differentiate p^A and $\hat{p}^A$ with respect to A,

$$\frac{\partial p^A}{\partial A} = \kappa y p^A (1 - p^A)(x - x')(x - x')^\top, \tag{21}$$

$$\frac{\partial \hat{p}^A}{\partial A} = -y \hat{p}^A (1 - \hat{p}^A)(x - x')(x - x')^\top. \tag{22}$$

Note that from

$$\hat{p}^A(y \mid x, x'; \hat{A}, \hat{\eta}) = p^A(y \mid x, x'; A^*, \kappa^*, \eta), \tag{23}$$

we have

$$\frac{\partial \hat{\mathcal{L}}}{\partial \hat{p}^A_{i,j}}\Big|_{A=\hat{A}} = \frac{\partial \mathcal{L}}{\partial p^A_{i,j}}\Big|_{A=A^*,\kappa=\kappa^*}.$$

Thus from

$$\frac{\partial \hat{p}^A}{\partial A}\Big|_{A=\hat{A}} = -\frac{1}{\kappa^*}\frac{\partial p^A}{\partial A}\Big|_{A=A^*,\kappa=\kappa^*},$$

$\partial \operatorname{tr}(A)/\partial A = I_m$ where I_m is the identity matrix, and the KKT stationarity condition of optimization (6)

$$\frac{\partial \mathcal{L}}{\partial A}\Big|_{A=A^*,\kappa=\kappa^*} = 0_{m\times m}$$

where $0_{m\times m}$ is the zero matrix in $\mathbb{R}^{m\times m}$, we get

$$\frac{\partial \hat{\mathcal{L}}}{\partial A}\Big|_{A=\hat{A}} = -\frac{1}{\kappa^*}\frac{\partial \mathcal{L}}{\partial A}\Big|_{A=A^*,\kappa=\kappa^*} = 0_{m\times m}.$$

This implies that $\hat{A}$ is a stationary point of $\hat{\mathcal{L}}(A)$.

Similarly, we could derive

$$\frac{\partial^2 \hat{\mathcal{L}}}{\partial A^2}\Big|_{A=\hat{A}} = \left(\frac{1}{\kappa^*}\right)^2 \frac{\partial^2 \mathcal{L}}{\partial A^2}\Big|_{A=A^*,\kappa=\kappa^*}.$$

Hence, $\partial^2_A \hat{\mathcal{L}}(\hat{A}) \preceq 0$ if and only if $\partial^2_A \mathcal{L}(A^*,\kappa^*) \preceq 0$, and $\hat{A}$ is actually a maximum of $\hat{\mathcal{L}}(A)$.

Remember Eq.(23) that $\hat{p}^A(y \mid x, x'; \hat{A}, \hat{\eta})$ is identical to $p^A(y \mid x, x'; A^*, \kappa^*, \eta)$. The theorem follows. □

A.3. Proof of Theorem 3

Proof. By the techniques used in the supplementary material of Graça et al. (2009), the dual of optimization (9) should be

$$\begin{aligned} \min_{\xi_{i,j}} \quad & \ln\left(\sum\nolimits_y p^A_{i,j}(y)\exp(\xi_{i,j}\ln p^A_{i,j}(y))\right) \\ \text{s.t.} \quad & 0 \le \xi_{i,j} \le \mu, \end{aligned}$$

where $\xi_{i,j}$ is the dual variable, and the primal variable can be recovered by

$$q(y \mid x_i, x_j) = \frac{p^A_{i,j}(y)\exp(\xi_{i,j}\ln p^A_{i,j}(y))}{\sum_{y'} p^A_{i,j}(y')\exp(\xi_{i,j}\ln p^A_{i,j}(y'))}.$$

The optimal $q(y \mid x_i, x_j)$ is given by

$$q(y \mid x_i, x_j) = \frac{p^A_{i,j}(y)\exp(\mu \ln p^A_{i,j}(y))}{\sum_{y'} p^A_{i,j}(y')\exp(\mu \ln p^A_{i,j}(y'))},$$

since the objective of the dual problem is monotonically decreasing with respect to $\xi_{i,j}$.

However, we present here a shorter and direct proof to get Eq.(10) for the sake of self-containing.

As before, let us denote

$$\begin{aligned} p^+_{i,j} &\triangleq p^A_{i,j}(+1), \\ p^-_{i,j} &\triangleq p^A_{i,j}(-1), \\ q^+_{i,j} &\triangleq q(+1 \mid x_i, x_j), \\ q^-_{i,j} &\triangleq q(-1 \mid x_i, x_j), \end{aligned}$$

respectively. We expand optimization (9) to its complete form:

$$\begin{aligned} \min_{q_{i,j}} \quad & q^+_{i,j}\ln\frac{q^+_{i,j}}{p^+_{i,j}} + q^-_{i,j}\ln\frac{q^-_{i,j}}{p^-_{i,j}} - \mu q^+_{i,j}\ln p^+_{i,j} - \mu q^-_{i,j}\ln p^-_{i,j} \\ \text{s.t.} \quad & q^+_{i,j} + q^-_{i,j} = 1. \end{aligned}$$

The terms $\ln(q^+_{i,j}/p^+_{i,j})$ and $\ln(q^-_{i,j}/p^-_{i,j})$ in the objective function plus $q^+_{i,j} + q^-_{i,j} = 1$ in the constraints have already implied that

$$0 \le q^+_{i,j}, q^-_{i,j} \le 1.$$

By introducing a dual variable $\xi_{i,j}$, the Lagrangian is expressed as

$$\begin{aligned} L(q_{i,j}, \xi_{i,j}) = {} & q^+_{i,j}\ln\frac{q^+_{i,j}}{p^+_{i,j}} + q^-_{i,j}\ln\frac{q^-_{i,j}}{p^-_{i,j}} - \mu q^+_{i,j}\ln p^+_{i,j} \\ & - \mu q^-_{i,j}\ln p^-_{i,j} + \xi_{i,j}(q^+_{i,j} + q^-_{i,j} - 1). \end{aligned}$$

Differentiate the function $L(q_{i,j}, \xi_{i,j})$ with respect to $q^+_{i,j}$ and $q^-_{i,j}$, equate the derivatives to zero, and then we get

$$\begin{aligned} \ln q^+_{i,j} &= \ln p^+_{i,j} + \mu \ln p^+_{i,j} - 1 - \xi_{i,j}, \\ \ln q^-_{i,j} &= \ln p^-_{i,j} + \mu \ln p^-_{i,j} - 1 - \xi_{i,j}, \end{aligned}$$

which says that

$$\frac{q^+_{i,j}}{q^-_{i,j}} = \frac{p^+_{i,j}}{p^-_{i,j}}\exp(\mu \ln p^+_{i,j} - \mu \ln p^-_{i,j}).$$

The analytical solution defined in Eq.(10) follows after the normalization. □

A.4. Proof of Theorem 4

Proof. Obviously $\mathcal{F}(A)$ is differentiable as long as we allow unbounded derivatives. Now we prove that the derivative is uniformly bounded for fixed training set $\mathcal{X}$.

The conjugate matrix norm of the Frobenius norm is still the Frobenius norm, namely,

$$\|B\|^*_F = \max_{\|A\|_F \le 1} \operatorname{tr}(A^\top B) = \|B\|_F.$$

Then the best Lipschitz constant of $\mathcal{F}$ with respect to $\|\cdot\|_F$ can be expressed as

$$\mathrm{Lip}_{\|\cdot\|_F}(\mathcal{F}) = \sup_{A\succeq 0} \|\nabla \mathcal{F}\|_F,$$

so it is sufficient to bound $\|(\partial\mathcal{F}/\partial p^A_{i,j})(\partial p^A_{i,j}/\partial A)\|_F$ from above uniformly.

Recall that the partial derivative of the simplified p^A with respect to A was given by Eq.(22) as

$$\frac{\partial p^A}{\partial A} = -y p^A (1-p^A)(x-x')(x-x')^\top.$$

On the other hand,

$$\frac{\partial \mathcal{F}}{\partial p^A_{i,j}} = \begin{cases} \dfrac{1}{p^A_{i,j}(y_{i,j})} & \text{if } (x_i,x_j)\in\mathcal{S}\cup\mathcal{D} \\ \dfrac{\mu q(y\mid x_i,x_j)}{p^A_{i,j}(y)} & \text{if } (x_i,x_j)\in\mathcal{U}, y\in\{1,-1\}. \end{cases}$$

Hence when $(x_i,x_j)\in\mathcal{S}\cup\mathcal{D}$,

$$\begin{aligned} &\left\| \frac{\partial\mathcal{F}}{\partial p^A_{i,j}} \cdot \frac{\partial p^A_{i,j}}{\partial A} \right\|_F \\ =&\left\| -y_{i,j}(1-p^A_{i,j}(y_{i,j}))(x_i-x_j)(x_i-x_j)^\top \right\|_F \\ \le&\left\| (x_i-x_j)(x_i-x_j)^\top \right\|_F \\ =&\|x_i-x_j\|_2^2 \\ \le&(\mathrm{diam}(\mathcal{X}))^2, \end{aligned}$$

where we use the fact that

$$\begin{aligned} \left\|zz^\top\right\|_F^2 &= \sum_{i,j=1}^m (z_i z_j)^2 \\ &= \left(\sum_{i=1}^m z_i^2\right)\left(\sum_{j=1}^m z_j^2\right) \\ &= \|z\|_2^4. \end{aligned}$$

When $(x_i,x_j)\in\mathcal{U}$, for fixed y we have

$$\left\| \frac{\partial\mathcal{F}}{\partial p^A_{i,j}} \cdot \frac{\partial p^A_{i,j}}{\partial A} \right\|_F \le \mu q(y\mid x_i,x_j)(\mathrm{diam}(\mathcal{X}))^2,$$

and thus

$$\sum\nolimits_y \left\| \frac{\partial\mathcal{F}}{\partial p^A_{i,j}} \cdot \frac{\partial p^A_{i,j}}{\partial A} \right\|_F \le \mu(\mathrm{diam}(\mathcal{X}))^2.$$

As a result, there exists a finite $\mathrm{Lip}_{\|\cdot\|_F}(\mathcal{F})$. The inequality (12) is obtained by applying the triangle inequality of the Frobenius norm. □

B. Additional Justifications

B.1. Generalized maximum entropy principle

The regularization term $-\gamma\kappa^2/2$ is necessary, otherwise we will have $\kappa^* = -\infty$ for the optimal solution (A^*,κ^*). In other words, the optimization degenerates, and the learned metric may easily overfit weakly labeled training data. The reason for this phenomenon is the single point prior of the expected data moments from the empirical data moments. According to the generalized maximum entropy principle (Dudík & Schapire, 2006), an ℓ_2-regularization on the dual variable reflects some Gaussian prior of the expected data moments from the empirical data moments rather than the single point prior of the maximum entropy principle (Jaynes, 1957), where no regularization applies to the dual variable.

The potential function underlies the generalized maximum entropy distribution estimation. By the potential function and the slack variable, we can obtain the same dual problem for ℓ_2-regularization but different dual problems for ℓ_1-regularization on the gap of expected data moments and empirical data moments.

Let the potential function $U_f(\cdot)$ and its target value u_f be

$$\begin{aligned} U_f(x) &= \frac{1}{2\gamma}(x-u_f)^2, \\ u_f &= \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} f(x_i,x_j,y_{i,j}). \end{aligned}$$

Redefine optimization (1) as an equivalent form

$$\max_{A,p^A_{i,j}} \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} H(p^A_{i,j}) - U_f\left(\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \mathbb{E}_{p^A_{i,j}}[f(x_i,x_j,y)]\right),$$

where the equivalence is due to *Fenchel's Duality Theorem* of Dudík & Schapire (2006) and the fact that the conjugate of $U_f(x)$ is $U_f^*(\kappa) = \gamma\kappa^2/2$. Subsequently,

$$\begin{aligned} \max_{A,\kappa}\ \mathcal{L}_2(A,\kappa) = &\sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln p^A_{i,j}(y_{i,j}) - U_f^*(-\kappa) \\ &- \mu U_g\left(\sum\nolimits_{\mathcal{U}} \mathbb{E}_{p^A_{i,j}}[g(x_i,x_j,y)]\right) \end{aligned}$$

is a problem with two potential functions $U_f(\cdot)$ and $U_g(\cdot)$ under the framework of Bellare et al. (2009), and hence SERAPH can be viewed as a semi-supervised maximum entropy estimation equipped with the projection sparsity.

B.2. Information maximization principle

The framework of regularized information maximization (Gomes et al., 2010) follows the information maximization principle, and advocates the preference for maximizing the mutual information between data and labels as well as the necessity of regularization on model parameters.

Given the supervised part of SERAPH, the regularized information maximization framework would suggest

$$\max_{A,\kappa} \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} \ln p^A_{i,j}(y_{i,j}) + \mu I(y;\mathcal{U}) - \frac{\gamma}{2}\kappa^2 - \lambda \operatorname{tr}(A),$$

where $I(y;\mathcal{U})$ is the mutual information between unlabeled data and unobserved weak labels:

$$I(y;\mathcal{U}) = \sum\nolimits_{\mathcal{U}} \sum\nolimits_{y} p(x_i, x_j, y) \ln\left(\frac{p(x_i, x_j, y)}{p(x_i)p(x_j)p(y)}\right).$$

By decomposing $I(y;\mathcal{U})$, it could be rewritten as

$$\max_{A,\kappa} \mathcal{L}(A,\kappa) + \mu H(\hat{p}(y)),$$

where $\hat{p}(y)$ is a simple estimate of the prior $p(y)$ defined as

$$p(y) = \iint_{\mathbb{R}^m\times\mathbb{R}^m} p^A(y \mid x, x')p(x)p(x')\mathrm{d}x\mathrm{d}x'.$$

The entropy of $p(y)$ encourages a balanced prior distribution of y under the metric $d(x, x')$. However, the number of similar and dissimilar pairs (i.e., $y = 1$ and $y = -1$) are inherently imbalanced in all metric learning problem settings. Therefore we simply drop $H(\hat{p}(y))$ and attain (6).

C. Extensions

C.1. Generality

Although we use Eq.(4) as our feature function, other options are available. In fact, optimization (1) is very general such that it could be applied to other problem settings. Suppose temporarily that the input domain is $\mathscr{X}$, the output range is $\mathscr{Y}$, and the training data is

$$(\mathcal{X}, \mathcal{Y}) = \{(x_i, y_i) \mid x_i \in \mathscr{X}, y_i \in \mathscr{Y}\}_{i=1}^n.$$

Subsequently, there is a general feature function

$$\begin{aligned} f(x, y, x', y') = \\ -(d^2_X(x, x') - \eta_X)(d^2_Y(y, y') - \eta_Y) \end{aligned} \tag{24}$$

defined on $(\mathscr{X}\times\mathscr{Y})^2$, where d_X and d_Y are two metrics for $\mathscr{X}$ and $\mathscr{Y}$, and $\eta_X, \eta_Y > 0$ are separating thresholds. As long as the range $\mathscr{Y}$ is finite, the problem can be solved by SERAPH. For instance, when $\mathscr{Y} = \{1, \dots, c\}$ for some positive integer c, the global distance metric feature function (4) can be derived from (24) by

$$\begin{aligned} d_X(x, x') &= \|x - x'\|_A, \\ d_Y(y, y') &= 1 - \delta(y, y'), \\ \eta_X &= \eta, \\ \eta_Y &= \frac{1}{2}, \end{aligned}$$

where $\delta(\cdot,\cdot)$ is the Kronecker delta function. The local distance metric feature function can be given by

$$d_X(x, x') = \|x - x'\|_A / \|x - x'\|_2$$

as well. When considering multi-label classification where $\mathscr{Y} = \{0, 1\}^c$, one can simply replace the above $d_Y(y, y')$ with

$$\delta(y, y') = 1 - \langle y, y'\rangle / (\|y\|_2\|y'\|_2).$$

C.2. Kernel extension

The kernel extension is straightforward. Suppose that we have a kernel function $k : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}$, and a basis $\mathcal{B} = \{\bar{x}_i \mid \bar{x}_i \in \mathbb{R}^m\}_{i=1}^b$ where most often but not necessarily $\mathcal{B} \subseteq \mathcal{X}$. Let the empirical kernel map be

$$\begin{aligned} \phi : \mathbb{R}^m &\mapsto \mathbb{R}^b \\ x &\mapsto (k(x, \bar{x}_1), \dots, k(x, \bar{x}_b))^\top. \end{aligned}$$

Under this scenario, we learn a Mahalanobis distance metric for $\phi(x), \phi(x') \in \mathbb{R}^b$ of the form

$$d(x, x') = \sqrt{(\phi(x) - \phi(x'))^\top A(\phi(x) - \phi(x'))},$$

where $A \in \mathbb{R}^{b\times b}$ is a symmetric positive semi-definite matrix to be learned.

Subsequently, we assume that $\mathcal{B} = \mathcal{X}$. Let K be the kernel matrix and $k_1, \dots, k_n$ be its columns, then for any $x_i, x_j \in \mathcal{X}$,

$$d(x_i, x_j) = \sqrt{(k_i - k_j)^\top A(k_i - k_j)}.$$

All the components of SERAPH remain the same by replacing x_i with the corresponding column k_i of the kernel matrix. The resultant $d(x, x')$ will be highly non-linear with respect to $x, x' \in \mathbb{R}^m$. There is a similar kernel extension based on the kernel PCA map defined as

$$\begin{aligned} \phi : \mathbb{R}^m &\mapsto \mathbb{R}^n \\ x &\mapsto K^{-1/2}(k(x, x_1), \dots, k(x, x_n))^\top. \end{aligned}$$

C.3. Manifold extension

The manifold extension is also straightforward.

Without loss of generality, we adopt the kernel matrix K as the adjacency matrix of the underlying similarity graph. Let $D = \operatorname{diag}(d_1, \dots, d_n)$ be the degree matrix such that $d_i = \sum_{j=1}^n K_{i,j}$, then the *unnormalized graph Laplacian* is given by

$$L = D - K.$$

Let $P \in \mathbb{R}^{m'\times m}$ be the projection associated with A such that $A = P^\top P$, $X = (x_1, \dots, x_n)^\top \in \mathbb{R}^{n\times m}$ be the matrix

form of $\mathcal{X}$, and $Z = (z_1, \ldots, z_n)^\top \in \mathbb{R}^{n \times m'}$ be the projected data such that $z_i = Px_i$. According to the manifold assumption, we should also minimize

$$\begin{aligned}\mathcal{M}(A) &= \operatorname{tr}\left(Z^\top L Z\right) \\ &= \operatorname{tr}\left(X^\top L X A\right).\end{aligned}$$

More specifically, the closeness of x_i and x_j is measured by the kernel function whereas the closeness of z_i and z_j is measured by the Euclidean distance, and the assumption that z_i and z_j should be close if x_i and x_j are close is interpreted as $\|z_i - z_j\|_2^2$ should be penalized more for larger $K_{i,j}$ than smaller $K_{i,j}$. Consequently, we have

$$\begin{aligned}\mathcal{M}(A) &= \frac{1}{2}\sum_{i,j=1}^{n} K_{i,j}\|z_i - z_j\|_2^2 \\ &= \sum_{i,j=1}^{n} K_{i,j}\left(z_i^\top z_i - z_i^\top z_j\right) \\ &= \sum_{i=1}^{n} d_i z_i^\top z_i - \sum_{i,j=1}^{n} K_{i,j} z_i^\top z_j \\ &= \operatorname{tr}\left(DZZ^\top\right) - \operatorname{tr}\left(KZZ^\top\right) \\ &= \operatorname{tr}\left(LZZ^\top\right) \\ &= \operatorname{tr}\left(LXP^\top PX^\top\right) \\ &= \operatorname{tr}\left(X^\top L X A\right).\end{aligned}$$

Note that $\mathcal{M}(A)$ is again linear with respect to A, and it will not affect the convexity and the Lipschitz continuity of the M-Step. Let $\omega \geq 0$ be a regularization parameter for $\mathcal{M}(A)$, then the optimization problem becomes

$$\max_A \ \mathcal{L}(A) - \omega\mathcal{M}(A),$$

and at each M-Step, we solve

$$\max_A \ \mathcal{F}(A) - \omega\mathcal{M}(A).$$

The gradient of $\mathcal{F}(A) - \omega\mathcal{M}(A)$ is given by

$$\begin{aligned}\nabla\mathcal{F}(A) - \omega\nabla\mathcal{M}(A) = \\ &- \sum\nolimits_{\mathcal{S}\cup\mathcal{D}} y_{i,j}\left(1 - p_{i,j}^A(y_{i,j})\right)x_{i,j} \\ &- \mu\sum\nolimits_{\mathcal{U}}\sum\nolimits_{y} yq(y \mid x_i, x_j)\left(1 - p_{i,j}^A(y)\right)x_{i,j} \\ &- \lambda I_m - \omega X^\top L X,\end{aligned}$$

where $x_{i,j} = (x_i - x_j)(x_i - x_j)^\top$.

D. Computational Issues

We employ a heuristic strategy for step sizes of the M-Step: At the k-th iteration of the gradient projection method, we set initially

$$s_k = \frac{m}{10\sqrt{k}\|\nabla\mathcal{F}\|_F},$$

where $\|\cdot\|_F$ is the Frobenius norm. Let π be an operator that projects a symmetric matrix to the cone of symmetric positive semi-definite matrices, which includes eigendecomposing a matrix and recovering it from the positive eigenvalues and the eigenvectors associated with those positive eigenvalues. Then we first try this gradient projection update

$$A_{k+1} = \pi(A_k + s_k\nabla\mathcal{F})$$

and keep it if A_{k+1} improves A_k, otherwise we decrease s_k by half, i.e.,

$$s_k \leftarrow s_k/2,$$

and try again, since we are maximizing a concave objective function. The maximum number of such trails is set to be 20. What is more, the maximum number $k_{\max}$ of inner gradient projection iterations and the maximum number $t_{\max}$ of outer EM iterations are set to be 10. We stop before reaching the maximum iteration numbers if either the solutions have been converged for both inner and outer iterations, or we fail to further improve the objective function $\mathcal{L}$ after the last M-Step.

Consider the computational complexity of the M-Step first. The complexity of the gradient part is $O(n^2m)$, the complexity of the projection part is $O(m^3)$, and thus each inner iteration takes $O(n^2m + m^3)$ time[9]. Let ϵ' be a stopping criterion of the M-Step such that $\mathcal{F}(A)$ has to increase at least ϵ', then the asymptotic time complexity of each M-Step will be

$$O\left(\frac{n^2m + m^3}{\epsilon'}\right).$$

Secondly, it is easy to see that each E-Step consumes the time of order $O(n^2)$. Thirdly, let ϵ be a stopping criterion of the whole algorithm, the total number of outer iterations is then $O(1/\epsilon)$. Therefore, the overall asymptotic time complexity is

$$O\left(\frac{n^2m + m^3}{\epsilon\epsilon'}\right).$$

Note that we ignore the maximum iteration numbers used in our implementation when discussing the asymptotic time complexities.

In practice, the main computational bottleneck is how to compute $\nabla\mathcal{F}(A)$ in Matlab without inefficient double FOR loops, as well as computing $\mathcal{L}(A)$, $\mathcal{F}(A)$ and $q(y \mid x_i, x_j)$ for all $(x_i, x_j) \in \mathcal{U}$ without double FOR loops. Fortunately, there are such methods. Without loss of generality, we describe the efficient method in Algorithm 1 to compute $\nabla\mathcal{F}(A)$. We observed that in our experiments Algorithm 1

[9] For a well-posed training set, there should be $n > m$.

Algorithm 1 Efficient computation of $\nabla\mathcal{F}(A)$

Input: the current solution A,
$X \in \mathbb{R}^{n\times m}$ that is the design matrix of $\mathcal{X}$,
$S \in \mathbb{R}^{n\times n}$ such that $S(i,j) = 1$ if $(x_i, x_j) \in \mathcal{S}$ and $S(i,j) = 0$ if $(x_i, x_j) \notin \mathcal{S}$,
$D \in \mathbb{R}^{n\times n}$ such that $D(i,j) = 1$ if $(x_i, x_j) \in \mathcal{D}$ and $D(i,j) = 0$ if $(x_i, x_j) \notin \mathcal{D}$,
$Q \in \mathbb{R}^{n\times n}$ such that $Q(i,j) = q(+1 \mid x_i, x_j)$ for $(x_i, x_j) \in \mathcal{U}$
Output: $\nabla\mathcal{F}(A)$

1: Compute all pairwise Mahalanobis distances by

$$\bar{x} \leftarrow \mathrm{diag}(XAX^\top), M \leftarrow \mathrm{repmat}(\bar{x}, 1, n) + \mathrm{repmat}(\bar{x}^\top, n, 1) - 2XAX^\top.$$

% Now, $M(i,j) = (x_i - x_j)^\top A(x_i - x_j)$.

2: Compute

$$P \leftarrow 1./(1 + \exp(M - \eta)),$$

where $./$ and $\exp$ are the element-wise matrix division and exponential function.
% Then, $P(i,j) = p^A(+1 \mid x_i, x_j)$.

3: Let $C \in \mathbb{R}^{n\times n}$ that will store all the coefficients of $(x_i - x_j)(x_i - x_j)^\top$.
Initialize it as $C \leftarrow 0_{n\times n}$.

4: Let $O \leftarrow 1_{n\times n}$, and subsequently

$$C_S \leftarrow P_S - O_S, C_D \leftarrow P_D,$$

where the subscripts S and D mean that the matrix operations are done only for the entries corresponding to $S(i,j) = 1$ or $D(i,j) = 1$.

5: Get the matrix form of $\mathcal{U}$ by

$$U \leftarrow O - S - D - I_n,$$

and compute

$$C_U \leftarrow \mu(Q_U .* (P_U - O_U) + (O_U - Q_U) .* P_U)$$

where $.*$ is the element-wise matrix multiplication.

6: Finally,

$$\nabla\mathcal{F}(A) \leftarrow X^\top(\mathrm{repmat}(\mathrm{sum}(C, 2), 1, m) .* X) - X^\top CX - \lambda I_m.$$

% Note that $\nabla\mathcal{F}(A) \neq 2X^\top(\mathrm{repmat}(\mathrm{sum}(C, 2), 1, m) .* X) - 2X^\top CX - \lambda I_m$,
% since we consider one pair (x_i, x_j) twice in the algorithm by $C(i,j)$ and $C(j,i)$.

was at least twenty times faster than the naive implementation of this subroutine using double FOR loops. SERAPH was in general the second fastest algorithm in our experiment. The fastest MFDA involved solving a linear system in the locally linear embedding and an eigen-decomposition, while it has the same computational complexity with SERAPH.