Expert Systems with Applications 42 (2015) 1278–1286

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Feature selection and multi-kernel learning for adaptive graph regularized nonnegative matrix factorization

Jim Jing-Yan Wang [a], Jianhua Z. Huang [b], Yijun Sun [c], Xin Gao [a,*]

[a] Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia
[b] Department of Statistics, Texas A&M University, TX 77843-3143, USA
[c] University at Buffalo, The State University of New York, Buffalo, NY 14203, USA

ARTICLE INFO

Article history:
Available online 20 September 2014

Keywords:
Data representation
Nonnegative matrix factorization
Graph regularization
Feature selection
Multi-kernel learning

ABSTRACT

Nonnegative matrix factorization (NMF), a popular part-based representation technique, does not capture the intrinsic local geometric structure of the data space. Graph regularized NMF (GNMF) was recently proposed to avoid this limitation by regularizing NMF with a nearest neighbor graph constructed from the input data set. However, GNMF has two main bottlenecks. First, using the original feature space directly to construct the graph is not necessarily optimal because of the noisy and irrelevant features and nonlinear distributions of data samples. Second, one possible way to handle the nonlinear distribution of data samples is by kernel embedding. However, it is often difficult to choose the most suitable kernel. To solve these bottlenecks, we propose two novel graph-regularized NMF methods, AGNMF$_{FS}$ and AGNMF$_{MK}$, by introducing feature selection and multiple-kernel learning to the graph regularized NMF, respectively. Instead of using a fixed graph as in GNMF, the two proposed methods learn the nearest neighbor graph that is adaptive to the selected features and learned multiple kernels, respectively. For each method, we propose a unified objective function to conduct feature selection/multi-kernel learning, NMF and adaptive graph regularization simultaneously. We further develop two iterative algorithms to solve the two optimization problems. Experimental results on two challenging pattern classification tasks demonstrate that the proposed methods significantly outperform state-of-the-art data representation methods.

1. Introduction

Nonnegative matrix factorization (NMF) (Lee & Seung, 2000; Sun, Wu, Wu, Guo, & Lu, 2012; Wang, Almasri, & Gao, 2012b; Wang, Bensmail, & Gao, 2013a; Wang & Gao, 2013; Wang, Wang, & Gao, 2013c) decomposes a nonnegative data matrix as a product of two low-rank nonnegative matrices, one of them is regarded as the basis matrix, while the other one as the coding matrix, which could be used as a reduced representation of the data samples in the data matrix (Kim, Chen, Kim, Pan, & Park, 2011a). This method has become popular in recent years for data representation in various areas, such as bioinformatics (Zheng, Ng, Zhang, Shiu, & Wang, 2011) and computer vision (Cai et al., 2013). Recently, Cai, He, Han, and Huang (2011) argued that NMF fails to exploit the intrinsic local geometric structure of the data space. They improved the traditional NMF to graph regularized nonnegative matrix factorization (GNMF). The basic idea is that the data samples are drawn from a low-dimensional manifold with a local geometric structure (Orsenigo & Vercellis, 2012; Wang, Bensmail, & Gao, 2012a, 2014a; Wang, Bensmail, Yao, & Gao, 2013b; Wang, Sun, & Gao, 2014b). Thus the nearby data samples in the original data space should also have similar NMF representations. In GNMF, the geometric structure of the data space is encoded by constructing a nearest neighbor graph, and then the matrix factorization is sought by adding a graph regularization to the original NMF objective function. The key component of GNMF is the graph. In the original GNMF algorithm, the graph is constructed according to the original input feature space. The nearest neighbors of a data sample is found by comparing the Euclidean distances (Lee, Rajkumar, Lo, Wan, & Isa, 2013a; Merigó & Casanovas, 2011) between pairs of data points, while the weights of edges are also estimated in the Euclidean space, by assuming that the original features could provide a proper representation of the local structure of the data space. However, as is well known that in many pattern recognition problems, using the original feature space directly is not appropriate because of the noisy and irrelevant features and the nonlinear distribution of the samples.

* Corresponding author. Tel.: +966 12 808 0323.
E-mail address: xin.gao@kaust.edu.sa (X. Gao).

http://dx.doi.org/10.1016/j.eswa.2014.09.008
0957-4174/

To handle the noisy and irrelevant features, one may apply **feature selection** (Fakhraei, Soltanian-Zadeh, & Fotouhi, 2014; Iquebal, Pal, Ceglarek, & Tiwari, 2014; Lin, Chen, & Wu, 2014; Li, Wu, Li, & Ding, 2013b, 2013a) to assign different weights to different features, so that the data samples could be represented in a better way than using the original features. So far, the most broadly used feature selection method is proposed by Sun et al. (2012). Such an approach is able to determine feature weights from a statistics point of view to automatically discover the intrinsic features. It provides a powerful and efficient solution for feature selection in NMF. So this work has been internationally recognized by the researchers in this field. To handle the nonlinear distribution of the data samples, one could map the input data into a nonlinear feature space by kernel embedding (Cui & Soh, 2010; Yeh, Su, & Lee, 2013). However, the most suitable types and parameters of the kernels for a particular task is often unknown, and selection of the optimal kernel by exhaustive search on a pre-defined pool of kernels is usually time-consuming, and sometimes causes over-fitting. **Multi-kernel learning** (Chen, Li, Wei, Xu, & Shi, 2011; Yeh, Huang, & Lee, 2011), which seeks the optimal kernel by a weighted, linear combination of pre-defined candidate kernels, has been introduced to handle the problem of kernel selection. An, Yun, and Choi (2011), presented the Multi-Kernel NMF (NMF$_{MK}$), which learns the best convex combination of multiple kernel matrices and NMF parameters jointly. However, graph regularization was not taken into consideration in their framework. In this paper, we will incorporate feature selection and multi-kernel learning into the graph regularization NMF to obtain novel and enhanced data representation methods. In this way, we could handle the problem of noisy and irrelevant features, nonlinearly distributed data samples, graph construction, and data matrix factorization simultaneously. Compared to the methods reported in the current literature which use a fixed graph for NMF parameters learning, our method can adapt the graph to the learned feature or kernel weights, which improves the NMF by providing it with a more reliable graph.

Here, we propose two novel methods, AGNMF$_{FS}$ and AGNMF$_{MK}$, that incorporate features selection and multiple-kernel learning into graph-regularized NMF, respectively. Feature selection or multi-kernel learning will provide a new data space for the graph construction of GNMF, and at the same time, GNMF will direct feature selection or multi-kernel learning. Both AGNMF$_{FS}$ and AGNMF$_{MK}$ are formulated as constraint optimization problems, each of which has a unified objective function to optimize feature selection/multi-kernel learning and graph-regularized NMF simultaneously. Experimental results demonstrate that the two proposed methods significantly outperform state-of-the-art data representation methods.

The rest of the paper is organized as follows: We briefly review the GNMF in Section 2. We then propose the two novel algorithms, AGNMF$_{FS}$ and AGNMF$_{MK}$, in Section 3. The proposed methods are compared with other NMF learning methods on two challenging data sets for classification tasks in Section 4. Finally, the paper is concluded in Section 5 with some future works.

2. Overview of graph regularized NMF

In this section, we will briefly introduce the graph regularized NMF as background knowledge of this paper.

2.1. Nonnegative matrix factorization

Given a training set with N nonnegative data samples $\mathcal{X} = \{x_1, \ldots, x_N\} \in \mathbb{R}_+^D$ represented as a nonnegative data matrix $X = [x_1, \ldots, x_N] \in \mathbb{R}_+^{D\times N}$, where $x_n \in \mathbb{R}_+^D$ is the D-dimensional nonnegative feature vector of the nth sample, NMF aims to find two nonnegative matrices H and W whose product can well approximate the original matrix X as

$$X \approx HW, \tag{1}$$

where $H \in \mathbb{R}^{D\times R}$, and $W \in \mathbb{R}^{R\times N}$. Accordingly, each sample x_n is approximated by a linear combination of the columns of H, weighted by the components of the nth column of W, as

$$x_n \approx \sum_{r=1}^{R} h_r w_{rn} \tag{2}$$

Therefore, H can be regarded as a collection of basis vectors, while, w_n, the nth columns of W, can be regarded as the coding vector or a new representation of the nth data sample. The most commonly used cost function to solve H and W is based on the squared Euclidean distance (SED) between the two matrices:

$$\begin{aligned} O^{NMF}(H,W) &= ||X - HW||^2 \\ &= Tr(XX^\top) - 2Tr(XW^\top H^\top) + Tr(HWW^\top H^\top), \end{aligned} \tag{3}$$

where $Tr(\cdot)$ denotes the trace of a matrix.

2.2. Graph regularized NMF

Cai et al. (2011) introduced the GNMF algorithm, by imposing the local invariance assumption (LIA) to NMF. If two data samples x_n and x_m are close in the intrinsic geometric space of the data distribution, w_n and w_m, the coding vectors of these two samples with respect to the new basis, should also be close to each other; and vice versa. They modeled the local geometric structure by a K-nearest neighbor graph $\mathcal{G}$ constructed from the data set $\mathcal{X}$. For each data sample $x_n \in \mathcal{X}$, the set of its K nearest neighbors, $\mathcal{N}_n$, in $\mathcal{X}$ is determined by the SED metric (Lee et al., 2013a) as

$$d(x_n, x_m) = ||x_n - x_m||^2 = \sum_{d=1}^{D} (x_{dn} - x_{dm})^2 = x_n^\top x_n + x_m^\top x_m - 2x_n^\top x_m \tag{4}$$

A K-nearest neighbor graph is constructed for $\mathcal{X}$. Each data sample in $\mathcal{X}$ will be a node of the graph, and each node x_n will be connected to its K nearest neighbors $\mathcal{N}_n$. We also define a weight matrix $A \in R^{N\times N}$ on the graph, with A_{nm} equal to the weight of the connection between nodes x_n and x_m. There are many choices to define the weight matrix A. Two of the most commonly used options are as follows:

Gaussian kernel weighting

$$A_{nm} = \begin{cases} exp\left(-\frac{||x_n - x_m||^2}{\sigma^2}\right), & if\ x_m \in \mathcal{N}_n, \\ 0, & otherwise. \end{cases} \tag{5}$$

Dot-product weighting

$$A_{nm} = \begin{cases} x_n^\top x_m, & if\ x_m \in \mathcal{N}_n, \\ 0, & otherwise. \end{cases} \tag{6}$$

With the weight matrix A, we can use the following graph regularization term to measure the smoothness of the low-dimensional coding vector representations in W:

$$O^G(W;A) = \frac{1}{2}\sum_{n,m=1}^{N} ||w_n - w_m||^2 A_{nm} = Tr(WDW^\top) - Tr(WAW^\top) = Tr(WLW^\top), \tag{7}$$

where D is a diagonal matrix whose entries are column sums of A, i.e., $D_{nn} = \sum_{m=1}^{N} A_{nm}$ and $L = D - A$ is the graph Laplacian matrix. By minimizing $O^G(W;A)$ with regard to W, we expect that if two data points x_n and x_m are close, i.e., A_{nm} is large, w_n and w_m are also close to each other.

Combining this geometry-based regularizer, $O^G(W;A)$, with the original NMF objective function, $O^{NMF}(H,W)$, leads to the loss function of GNMF (Cai et al., 2011):

$$\begin{aligned} O^{GNMF}(H,W;A) &= O^{NMF}(H,W) + \alpha O^G(W;A) \\ &= Tr(XX^\top) - 2Tr(XW^\top H^\top) + Tr(HWW^\top H^\top) \\ &\quad + \alpha Tr(WLW^\top), \end{aligned} \tag{8}$$

in which α is a tradeoff parameter. Thus the GNMF problem turns to a constrained minimization problem as

$$\begin{aligned} &\min_{H,W} O^{GNMF}(H,W;A), \\ &s.t.\ H \geqslant 0, \quad W \geqslant 0 \end{aligned} \tag{9}$$

where H and W can be solved in an iterative manner by optimizing and updating them alternately (Cai et al., 2011).

3. Adaptive graph regularized NMF with feature selection and multi-kernel learning

In this section, we propose two enhanced data representation methods based on GNMF by encoding feature selection and multi-kernel learning, respectively.

3.1. Adaptive graph regularized NMF with feature selection

3.1.1. Feature selection for NMF

Given an input sample x represented as a vector of D nonnegative features as $x = [x_1, \ldots, x_D]^\top \in \mathbb{R}_+^D$, feature selection tries to scale each feature to obtain a weighted feature space, parameterized by a D-dimensional nonnegative feature weight vector $\lambda = [\lambda_1, \ldots, \lambda_D]^\top \in \mathbb{R}_+^D$, where λ_d is the scaling factor for the dth feature (Sun, Todorovic, & Goodison, 2010b). We restrict its scale by $\sum_{d=1}^D \lambda_d = 1$ and $\lambda_d \geqslant 0$. Thus the scaled feature vector of x is represented as $\tilde{x} = [\lambda_1 x_1, \ldots, \lambda_D x_D]^\top = diag(\lambda)x$, where $diag(\lambda)$ is a $D \times D$ diagonal matrix with entries of λ along the main diagonal. The original data matrix and basis matrix for NMF can be represented in the scaled space as (10),

$$\tilde{X} = diag(\lambda)X \text{ and } \tilde{H} = diag(\lambda)H, \ \ s.t. \sum_{d=1}^D \lambda_d = 1, \ \ \lambda_d \geqslant 0, \ \ d = 1, \ldots, D. \tag{10}$$

By replacing the original features in X and H of NMF with the weighted features $\tilde{X}$ and $\tilde{H}$ defined in (10), we have the augmented objective function for NMF with feature selection in an enlarged parameter space

$$\begin{aligned} O^{NMF_{FS}}(H,W,\lambda) &= ||diag(\lambda)(X - HW)||^2 \\ &= Tr[diag(\lambda)^2 XX^\top] - 2Tr[diag(\lambda)^2 XW^\top H^\top] \\ &\quad + Tr[diag(\lambda)^2 HWW^\top H^\top] \end{aligned} \tag{11}$$

Here H, W and λ are all the variables to solve so that the above objective function can be minimized.

3.1.2. Graph adaptive to selected features

After the new feature space defined by feature weight vector λ is defined, the nearest neighbor graph should also be updated to be adaptive to the selected features. First, the K nearest neighbors, $\mathcal{N}_n$, of the nth data point should be re-found according to the λ-weighted SED, i.e.,

$$d^\lambda(x_n, x_m) = ||x_n - x_m||_\lambda^2 = \sum_{d=1}^D \lambda_d^2 (x_{dn} - x_{dm})^2 \tag{12}$$

The K nearest neighbors $\mathcal{N}_n$ re-found by the λ-weighted distance is denoted as $\mathcal{N}_n^\lambda$, and the graph adaptive to λ is donated as $\mathcal{G}^\lambda$. The corresponding weight matrix A^λ of $\mathcal{G}^\lambda$ should also be updated. Here we discuss how to update the Gaussian kernel weighting for adaptive graph with feature selection, which is updated as $A_{nm}^\lambda exp\left(-\frac{||x_n - x_m||_\lambda^2}{\sigma^2}\right)$ if $x_m \in \mathcal{N}_n^\lambda$, and 0 otherwise.

With the adaptive graph $\mathcal{G}^\lambda$, we can re-regularize the NMF in the selected feature space. Similar to the GNMF, we propose the adaptive graph regularization term as

$$O^{AG}(W;A^\lambda) = \frac{1}{2}\sum_{n,m=1}^N ||w_n - w_m||^2 A_{nm}^\lambda = Tr(WL^\lambda W^\top), \tag{13}$$

where $L^\lambda = D^\lambda - A^\lambda$ is the corresponding graph Laplacian. By minimizing $O^{AG}(W;A^\lambda)$, we expect that if two data points $\tilde{x}_n$ and $\tilde{x}_m$ are close with respect to the new features selected by λ, the representations w_n and w_m with respect to the selected features should also be close to each other.

3.1.3. Adaptive graph regularized NMF algorithm with feature selection

To perform the feature selection together with the adaptive graph regularized NMF, we first propose the unified objective function for adaptive graph regularized NMF and feature selection for data representation, and then develop an alternately updating algorithm to estimate the basis matrix H, the coding coefficient matrix W and the feature weight matrix λ.

- **Objective function**: Combining the NMF objective function with feature selection defined in (11) with the adaptive graph-based regularizer defined in (13) leads to the objective function of our AGNMF with **f**eature **s**election — $AGNMF_{FS}$ algorithm:

$$\begin{aligned} O^{AGNMF_{FS}}(H,W,\lambda) &= O^{NMF_{FS}}(H,W,\lambda) + \alpha O^{AG}(W;A^\lambda) \\ &= Tr[diag(\lambda)^2 XX^\top] \\ &\quad - 2Tr[diag(\lambda)^2 XW^\top H^\top] \\ &\quad + Tr[diag(\lambda)^2 HWW^\top H^\top] \\ &\quad + \alpha Tr(WL^\lambda W^\top) \end{aligned} \tag{14}$$

The optimization problem (8) of GNMF can now be extended to accommodate the feature selection and adaptive graph:

$$\begin{aligned} &\min_{H,W,\lambda} O^{AGNMF_{FS}}(H,W,\lambda) \\ &s.t.\ H \geqslant 0, \quad W \geqslant 0, \quad \sum_{d=1}^D \lambda_d = C, \quad \lambda_d \geqslant 0, \quad d = 1, \ldots, D. \end{aligned} \tag{15}$$

- **Optimization**: Since direct optimization to (15) is difficult, we instead adopt an iterative, two-step strategy to alternately optimize (H,W) and λ. At each iteration, one of (H,W) and λ is optimized while the other is fixed, and then the roles of (H,W) and λ are switched. Iterations are repeated until convergence or a maximum number of iterations is reached.
  - **On optimizing** (H,W): By fixing λ and updating the adaptive graph $\mathcal{G}^\lambda$ with its corresponding Laplacian matrix L^λ according to λ, the optimization problem (15) is reduced to

$$\begin{aligned} &\min_{H,W} Tr[diag(\lambda)^2 XX^\top] - 2Tr[diag(\lambda)^2 XW^\top H^\top] + Tr[diag(\lambda)^2 HWW^\top H^\top] \\ &+ \alpha Tr(WL^\lambda W^\top) s.t.\ H \geqslant 0, \ \ W \geqslant 0. \end{aligned} \tag{16}$$

The Lagrange $\mathcal{L}$ of the above optimization problem is

$$\begin{aligned} \mathcal{L} &= Tr[diag(\lambda)^2 XX^\top] - 2Tr[diag(\lambda)^2 XW^\top H^\top] \\ &\quad + Tr[diag(\lambda)^2 HWW^\top H^\top] + \alpha Tr(WL^\lambda W^\top) \\ &\quad + Tr(\Phi H^\top) + Tr(\Psi W^\top), \end{aligned} \tag{17}$$

where $\Phi = [\phi_{dr}]$ and $\Psi = [\psi_{rn}]$ are the Lagrange multiplier matrices for constraint $H \geqslant 0$ and $W \geqslant 0$, respectively. By setting the partial derivatives of $\mathcal{L}$ with respect to H and W to zero, we have

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial H} &= -2diag(\lambda)^2 XW^\top + 2diag(\lambda)^2 HWW^\top + \Phi = 0\\ \frac{\partial \mathcal{L}}{\partial W} &= -2H^\top diag(\lambda)^2 X + 2H^\top diag(\lambda)^2 HW + 2\alpha WL^\lambda + \Psi = 0\end{aligned} \tag{18}$$

Using the KKT conditions, i.e., $\phi_{dr} h_{dr} = 0$ and $\psi_{rn} w_{rn} = 0$, we get the following equations for h_{dr} and w_{rn}:

$$\begin{aligned}&-[diag(\lambda)^2 XW^\top]_{dr} h_{dr} + [diag(\lambda)^2 HWW^\top]_{dr} h_{dr} = 0\\ &-[H^\top diag(\lambda)^2 X]_{rn} w_{rn} + [H^\top diag(\lambda)^2 HW]_{rn} w_{rn} + \alpha (WL^\lambda)_{rn} w_{rn} = 0\end{aligned} \tag{19}$$

These equations lead to the following updating rules:

$$\begin{aligned}h_{dr} &\leftarrow \frac{[diag(\lambda)^2 XW^\top]_{dr}}{[diag(\lambda)^2 HWW^\top]_{dr}} h_{dr}\\ w_{rn} &\leftarrow \frac{[H^\top diag(\lambda)^2 X + \alpha WA^\lambda]_{rn}}{[H^\top diag(\lambda)^2 HW + \alpha WD^\lambda]_{rn}} w_{rn}\end{aligned} \tag{20}$$

– **On optimizing** λ: By fixing H and W, and removing the terms irrelevant to λ, the optimization problem (15) becomes

$$\begin{aligned}&\min_{\lambda}\ Tr[diag(\lambda)^2 (XX^\top - 2XW^\top H^\top + HWW^\top H^\top)]\\ &= Tr[diag(\lambda)^2 (YY^\top)], s.t. \sum_{d=1}^{D} \lambda_{dd} = 1, \lambda_{dd} \geqslant 0, \quad d = 1, \ldots, D.\end{aligned} \tag{21}$$

where $Y = X - HW$. Here, the value of λ_d indicates the weight of the dth feature. We rewrite the objective function of (21) as follows:

$$\begin{aligned}&\min_{\lambda}\ Tr[diag(\lambda)^2 (YY^\top)] = \sum_{d=1}^{D} \lambda_d^2 \sum_{n=1}^{N} y_{dn}^2 = \sum_{d=1}^{D} \lambda_d^2 e_d,\\ &s.t. \sum_{d=1}^{D} \lambda_{dd} = 1, \quad \lambda_{dd} \geqslant 0, \quad d = 1, \ldots, D. \lambda_d \geqslant 0.\end{aligned} \tag{22}$$

where y_{dn} is the (d,n)th element of matrix Y and $e_d = \sum_{n=1}^{N} y_{dn}^2$. It could be optimized by using Theorem 1.

Theorem 1. *The closed form solution of the optimization problem in (21) is given by:*

$$\lambda_d = \frac{1/y_d}{\sum_{d=1}^{F} 1/y_d}, \quad d = 1, \ldots, D \tag{23}$$

Proof. Given the constrain of $\sum_{d=1}^{D} \lambda_d = 1$ and the Candy-Schwartz inequality, we have

$$1 = \left(\sum_{d=1}^{D} \lambda_d\right)^2 = \left(\sum_{d=1}^{D} \lambda_d \sqrt{y_d} \cdot \frac{1}{\sqrt{y_d}}\right)^2 \leqslant \left(\sum_{d=1}^{D} \lambda_d^2 y_d\right)\left(\sum_{d=1}^{D} \frac{1}{y_d}\right) \tag{24}$$

Thus we have the following inequality,

$$\sum_{d=1}^{D} \lambda_d^2 y_d \geqslant \frac{1}{\left(\sum_{d=1}^{D} \frac{1}{y_d}\right)} \tag{25}$$

and the equal sign holds if $\lambda_d \sqrt{y_d} = C \frac{1}{\sqrt{y_d}}$, or

$$\lambda_d = C \frac{1}{y_d}, \quad d = 1, \ldots, D \tag{26}$$

Moreover, since $\sum_{d=1}^{D} \lambda_d = 1$, we have $C \sum_{d=1}^{D} \frac{1}{y_d} = 1$, therefore $C = \frac{1}{\sum_{d=1}^{D} \frac{1}{y_d}}$, and the minimizer of (21) is (23). □

- **Algorithm**: The proposed iterative AGNMF algorithm with feature selection (named as AGNMF$_{FS}$) is summarized in Algorithm 1.

Algorithm 1. AGNMF$_{FS}$ Algorithm

Input: Original data matrix X;
Input: Initial factorization matrices H^0 and W^0;
Input: Tolerance stopping criterion ξ;
Input: Maximum number of iterations, T;
Initialize the feature weight variables as $\lambda_d^0 = \frac{1}{D}$, $d = 1, \ldots, D$;
Initialize $t = 1$;
repeat
  Update the graph $\mathcal{G}^{\lambda^t}$ and its corresponding Laplacian matrix L^{λ^t} according to λ^{t-1} as introduced in Section 3.1.2;
  Update the factorization matrices H^t and W^t as in (20);
  Update the feature weights λ^t as in (22);
  $t = t + 1$;
until $O^{AGNMF_{FS}}(H^t, W^t, \lambda^t) \leqslant \xi$ or $t \geqslant T$
Output: The factorization matrices $H = H^{t-1}$, $W = W^{t-1}$ and feature weight vector $\lambda = \lambda^{t-1}$.

3.1.4. Representing test sample with AGNMF$_{FS}$

After learning (H, W) and λ via AGNMF$_{FS}$ for the training data matrix X, we can use the basis matrix H and feature weight matrix λ to infer the coding vector for a new data point. When a new test data sample $x \in \mathbb{R}_+^D$ comes in, we first connect it to its K nearest neighbors $\mathcal{N}^\lambda$ from the training set $\mathcal{X}$ which are found by using the λ-weighted SED (12), and then calculate the weight vector of x as $a^\lambda = [a_1^\lambda, \ldots, a_N^\lambda] \in \mathbb{R}^N$ where $a_n^\lambda = exp\left(-\frac{||x - x_m||_\lambda^2}{\sigma^2}\right)$, if $x_n \in \mathcal{N}^\lambda$; and 0, otherwise. Assuming the coding of the training samples are not affected by the test sample, we only need to optimize the following objective function regarding to the coding vector $w \in \mathbb{R}^R$ of the test sample:

$$\begin{aligned}&\min_{w}\ O(w)^{AGNMF_{FS}} = ||diag(\lambda)(x - Hw)||^2 + \frac{\alpha}{2} \sum_{n=1}^{N} ||w - w_n||^2 a_n^\lambda\\ &= Tr\left[diag(\lambda)^2 xx^\top\right] - 2Tr\left[diag(\lambda)^2 xw^\top H^\top\right] + Tr[diag(\lambda)^2 Hww^\top H^\top]\\ &\quad + \frac{\alpha}{2} \sum_{n=1}^{N} a_n^\lambda Tr(ww^\top) - \alpha Tr\left[w \sum_{n=1}^{N} a_n^\lambda w_n^\top\right] + \frac{\alpha}{2} \sum_{n=1}^{N} a_n^\tau Tr(w_n w_n^\top)\\ &s.t.\ w \geqslant 0.\end{aligned} \tag{27}$$

By setting the partial derivative of the Lagrange function of (27) with respect to w to zero, and using the KKT conditions, we can have the following updating rule for w:

$$w_r \leftarrow \frac{\left[H^\top diag(\lambda)^2 x + \frac{\alpha}{2} \sum_{n=1}^{N} a_n^\lambda w_n\right]_r}{\left[H^\top diag(\lambda)^2 Hw + \frac{\alpha}{2} \sum_{n=1}^{N} a_n^\lambda w\right]_r} w_r \tag{28}$$

By repeating this updating rule, we could have the optimal coding vector, w, for the test sample.

3.2. Adaptive graph regularized NMF with multiple kernel learning

3.2.1. Multiple kernel learning for NMF

Consider a nonlinear mapping $x_n \to \varphi(x_n)$ or $X \to \varphi(X) = [\varphi(x_1), \ldots, \varphi(x_N)]$, the kernel matrix $K \in \mathbb{R}^{N\times N}$ is given by $K = \varphi(X)^\top \varphi(X)$. A direct application of NMF to the feature matrix $\varphi(X)$ yields

$$\varphi(X) \approx HW \tag{29}$$

For the sake of convenience, we impose the constraint that the vectors defining H lie within the column space of $\varphi(X)$: $h_r = \sum_{n=1}^{N} f_{nr}\varphi(x_n)$ or

$$H = \varphi(X)F, \tag{30}$$

where f_{nr} is the (n,r)th element of the matrix $F \in \mathbb{R}_+^{N\times K}$. Substituting (30) to (3), we have the objective function for the kernelized version of NMF

$$\begin{aligned} O^{NMF_k}(F,W) &= ||\varphi(X) - \varphi(X)FW||^2 \\ &= Tr[\varphi(X)(I-FW)(I-FW)^\top\varphi(X)^\top] \\ &= Tr[\varphi(X)^\top\varphi(X)(I-FW)(I-FW)^\top] \\ &= Tr[K(I-FW)(I-FW)^\top] \end{aligned} \tag{31}$$

Suppose there are altogether L different kernel functions $\{K_l\}_{l=1}^{L}$ available for the NMF task in hand. Accordingly, there are L different but associated nonlinear feature spaces. In general, we do not know which kernel space should be used. An intuitive way is to use them all by concatenating all feature spaces into an augmented Hilbert space and associating each feature space with a relevance weight τ_l, where $\tau_l \geqslant 0$, $\sum_{l=1}^{L}\tau_l = 1$. We denote the kernel weights as a vector $\tau = [\tau_1, \ldots, \tau_L]^\top$. Performing the NMF in such feature space is equivalent to employing a combined kernel function for the NMF:

$$K^\tau = \sum_{l=1}^{L}\tau_l K_l, \quad s.t. \tau_l \geqslant 0, \quad \sum_{l=1}^{L}\tau_l = 1 \tag{32}$$

We substitute this relation into (31) to obtain the objective function for **Multi**ple **K**ernel-based **NMF** (NMF_{MK}):

$$O^{NMF_{MK}}(F,W,\tau) = Tr\left[\sum_{l=1}^{L}\tau_l K_l (I-FW)(I-FW)^\top\right] \tag{33}$$

3.2.2. Graph adaptation to multiple kernel learning

To update the graph $\mathcal{G}$ regarding the multiple kernel space, given a τ, the K nearest neighbors $\mathcal{N}_n^\tau$ for the GNMF algorithm will be re-found by the τ-weighted SED in the multiple kernel space, i.e.,

$$\begin{aligned} d_\tau(x_n,x_m) &= ||\varphi(x_n) - \varphi(x_m)||_\tau^2 \\ &= K^\tau(x_n,x_n) + K^\tau(x_n,x_m) - 2K^\tau(x_n,x_m) \\ &= \sum_{l=1}^{L}\tau_l[K_l(x_n,x_n) + K_l(x_n,x_m) - 2K_l(x_n,x_m)] \end{aligned} \tag{34}$$

The corresponding K nearest neighbor graph adaptive to τ is donated as $\mathcal{G}$. Here we discuss the updating of dot-product weighting for the weight matrix A^τ of the adaptive graph with multiple kernel learning, i.e, $A_{nm}^\tau = \varphi(x_n)^\top\varphi(x_m) = K^\tau(x_n,x_m) = \sum_{l=1}^{L}\tau_l K_l(x_n,x_m)$, if $x_m \in \mathcal{N}_n^\tau$; 0, otherwise.

With the graph $\mathcal{G}^\tau$ adaptive to the multiple kernel space, we then re-regularize the NMF_{MK} in the multiple kernel space. We propose the **Adap**tive **G**raph **r**egularization term as

$$O^{AG}(W;A^\tau) = \frac{1}{2}\sum_{n,m=1}^{N}||w_n - w_m||^2 A_{nm}^\tau = Tr(WL^\tau W^\top), \tag{35}$$

where $L^\tau = D^\tau - A^\tau$ is the corresponding graph Laplacian.

3.2.3. AGNMF algorithm with multiple kernel learning

To perform the multi-kernel learning together with the adaptive graph regularized NMF, we first propose a unified object function, and then develop an alternately updating algorithm to solve it.

- **Objective function**: Combining the NMF objective function with multiple kernel defined in (33) with the adaptive graph-based regularizer defined in (35) leads to the optimization problem of our AGNMF with multi-kernel learning — $AGNMF_{MK}$:

$$\begin{aligned} \min_{F,W,\tau} O^{AGNMF_{MK}}(F,W,\tau) &= O^{NMF_{MK}}(F,W,\tau) + \alpha O^{AG}(W;A^\tau) + \beta||\tau||^2 \\ &= Tr\left[\sum_{l=1}^{L}\tau_l K_l (I-FW)(I-FW)^\top\right] + \alpha Tr(WL^\tau W^\top) + \beta||\tau||^2 \\ &= Tr\left[K^\tau(I-FW)(I-FW)^\top\right] + \alpha Tr(WL^\tau W^\top) \\ &\quad + \beta||\tau||^2, s.t.\ F \geqslant 0,\ \ W \geqslant 0,\ \ \tau \geqslant 0,\ \ \sum_{l=1}^{L}\tau_l = 1 \end{aligned} \tag{36}$$

where the regularization term $||\tau||^2$ is also introduced to prevent the parameter τ from overfitting to one kernel.

- **Optimization**: Similar to $AGNMF_{FS}$, we also adopt an iterative strategy to alternately optimize (F,W) and τ.
  - **On optimizing** (F,W): By fixing τ and updating the adaptive graph $\mathcal{G}^\tau$ and kernel matrix K^τ, the optimization problem (36) is reduced to

$$\begin{aligned} &\min_{F,W} Tr\left[K^\tau(I-FW)(I-FW)^\top\right] + \alpha Tr(WL^\tau W^\top) \\ &s.t.\ F \geqslant 0, \quad W \geqslant 0 \end{aligned} \tag{37}$$

Similar to the optimization of H and W of $AGNMF_{FS}$, we have following rules to update F and W:

$$\begin{aligned} f_{nr} &\leftarrow \frac{(K^\tau W^\top)_{nr}}{(K^\tau FWW^\top)_{nr}} f_{nr} \\ w_{rn} &\leftarrow \frac{(F^\top K^\tau + \alpha WA^\tau)_{rn}}{(F^\top K^\tau FW + \alpha WD^\tau)_{rn}} w_{rn} \end{aligned} \tag{38}$$

  - **On optimizing** τ: By fixing F and W, and removing the irrelevant terms, the optimization problem (36) becomes

$$\begin{aligned} &\min_{\tau} Tr\left[\sum_{l=1}^{L}\tau_l K_l (I-FW)(I-FW)^\top\right] + \beta||\tau||^2 = Tr\left[\sum_{l=1}^{L}\tau_l K_l ZZ^\top\right] \\ &+ \beta||\tau||^2 = \sum_{l=1}^{L}\tau_l g_l + \beta\sum_{l=1}^{L}\tau_l^2, s.t.\ \tau \geqslant 0,\ \ \sum_{l=1}^{L}\tau_l = 1 \end{aligned} \tag{39}$$

where $Z = I - FW$ and $g_l = Tr\left[K_l ZZ^\top\right]$. The optimization of (39) with respect to the feature weights τ could be solved as a standard quadratic programming (QP) problem.

- **Algorithm**: The iterative AGNMF algorithm with multiple kernel learning (named as $AGNMF_{MK}$) is summarized in Algorithm 2.

Algorithm 2. AGNMF$_{MK}$ Algorithm

Input: L base kernel matrices $K_l, l = 1, \ldots, L$;
Input: Initial factorization matrices F^0 and W^0;
Input: Tolerance stopping criterion ξ;
Input: Maximum number of iterations, T;
Initialize the kernel weight variables as $\tau_l^0 = \frac{1}{L},\ l = 1, \ldots, L$;
Initialize $t = 1$;
repeat
  Update the graph $\mathcal{G}^{\tau^t}$ and its corresponding Laplacian matrix L^{τ^t} according to τ^{t-1} as introduced in Section 3.2.2;
  Update the factorization matrices F^t and W^t as in (38);
  Update the kernel weights τ^t as in (39);
  $t = t + 1$;
until $O^{AGNMF_{MK}}(F^t, W^t, \tau^t) \leqslant \xi$ or $t \geqslant T$
Output $F = F^{t-1},\ W = W^{t-1}$ and $\tau = \tau^{t-1}$.

3.2.4. Representing test sample with AGNMF$_{MK}$

When a test sample $x \in \mathbb{R}^D$ comes in, we first connect it to its K nearest neighbors $\mathcal{N}^\tau$ from the training set $\mathcal{X}$, which is found by using the τ-weighted SED (34). Then the weight vector $a^\tau = [a_1^\tau, \ldots, a_N^\tau] \in \mathbb{R}^N$ is calculated as $a_n^\tau = K^\tau(x, x_n)$, if $x_n \in \mathcal{N}^\tau$; 0, otherwise. We need to optimize the following objective function to solve w with AGNMF$_{MK}$:

$$\begin{aligned}\min_w O(w)^{AGNMF_{MK}} &= ||\phi(x) - \phi(X)Fw||^2 + \frac{\alpha}{2}\sum_{n=1}^{N}||w - w_n||^2 a_n^\tau \\ &= Tr[K^\tau(x,x)] - 2Tr[K^\tau(X,x)w^\top F^\top] + Tr[K^\tau(X,X)Fww^\top F^\top] \\ &\quad + \frac{\alpha}{2}\sum_{n=1}^{N} a_n^\tau Tr(ww^\top) - \alpha Tr\left[w\sum_{n=1}^{N} a_n^\tau w_n^\top\right] \\ &\quad + \frac{\alpha}{2}\sum_{n=1}^{N} a_n^\tau Tr(w_n w_n^\top), s.t.\ w \geqslant 0 \end{aligned} \tag{40}$$

where $K^\tau(X, y) = [K^\tau(x_1, y), \ldots, K^\tau(x_N, y)]^\top$, and $K^\tau(X, X) = [K^\tau(x_n, x_m)] \in \mathbb{R}^{N\times N}$. By setting the partial derivative of the Lagrange function of (40) regarding w to zero and using the KKT conditions, we have the following updating rule for w

$$w_r \leftarrow \frac{\left[F^\top K^\tau(X,x) + \frac{\alpha}{2}\sum_{n=1}^{N} a_n^\tau w_n\right]_r}{\left[F^\top K^\tau(X,X)Fw + \frac{\alpha}{2}\sum_{n=1}^{N} a_n^\tau w\right]_r} w_r \tag{41}$$

4. Experiments

In this section, we apply the two proposed enhanced AGNMF algorithms to two challenging classification tasks – colon cancer diagnosis and face recognition.

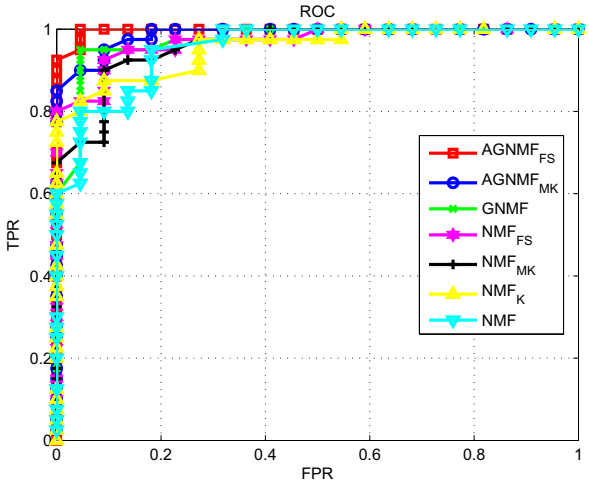
4.1. Experiment I: colon cancer diagnosis

In the first experiment, we test the proposed algorithms as data representation methods for the colon cancer diagnosis task based on the gene expression data.

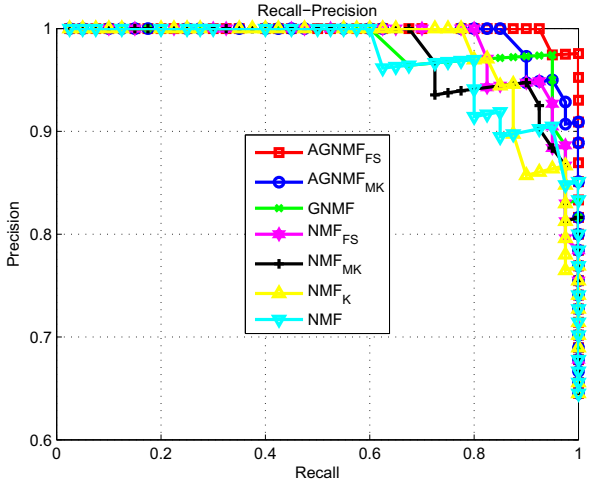
4.1.1. Colon cancer dataset and setup

Classification of colon cancer types according to the gene expression of patients' tissue samples is an important technique for colon cancer diagnosis (Zheng et al., 2011). Given the gene expression levels for genes of a sample, the aim is to identify if it is a tumor or a normal colon tissue. The gene expression data is usually nonnegative, thus NMF could be used to represent the gene expression data for classification tasks. In this experiment we will evaluate the use of the proposed algorithms as a data representation method for the colon cancer classification problem. A publicly available microarray dataset of colon tissue samples is used in this experiment (Zheng et al., 2011). The colon cancer data set contains the gene expression data of $D = 2000$ genes in $N = 62$ colon tissue samples, 22 of which are normal samples and 40 of which are tumor colon tissue samples. The colon tissue samples are defined as positive samples while the normal samples as negative ones. The gene expression data of 2000 genes of a sample are used as the original nonnegative features. The proposed AGNMF$_{FS}$ or AGNMF$_{MK}$ algorithms were applied to represent data samples in a low dimensional coding vector.

To evaluate the proposed algorithms, we performed a 5-fold cross validation on the dataset. The entire dataset was split into five folds randomly, and in each fold, there were 8 positive samples and 4–5 negative samples. Each fold was used as an independent test set in turn, while the remaining four folds were used as the training set. We first applied AGNMF$_{FS}$ and AGNMF$_{MK}$ respectively to the training set to represent all the training samples as coding vectors, and then trained a support vector machine (SVM) (Emami & Omar, 2013) to distinguish the normal and tumor colon tissue samples. When a test sample was given, we first represented



(a) ROC



(b) Recall-Precision

Fig. 1. ROC and recall-precision curves of different NMF representation methods on the colon cancer dataset.

Table 1
AUC values of different NMF representation methods on the colon cancer dataset.

Method	AUC
AGNMF$_{FS}$	0.9972
AGNMF$_{MK}$	0.9869
GNMF	0.9716
NMF$_{FS}$	0.9699
NMF$_{MK}$	0.9585
NMF$_{K}$	0.9545
NMF	0.9523

it by a coding vector, based on the basis and coding matrices, and feature selection or multi-kernel learning parameters learned using the training set, and then classified the coding vector by the trained SVM classifier. Note that all the parameters were tuned on the training set only, and the test set was not included in the parameter optimization procedure.

The classification performance is measured by the receiver operating characteristic (ROC) curves and recall-precision curves (Zhang, Xu, & Chen, 2008). The ROC curves were obtained by plotting the true positive rates (TPR) vs. the false positive rates (FPR) at various threshold settings, while recall-precision curves were obtained by plotting the recalls vs. the precisions at various threshold settings. The TPR, FPR, recall and precision are defined as follows:

$$TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN}, \\ recall = \frac{TP}{TP + FN}, \quad precision = \frac{TP}{TP + FP}, \tag{42}$$

where TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives, while FN is the number of false negatives. Moreover, the area under the ROC curve (AUC) is also used as a measure of the classification performance.

4.1.2. Experimental results

Since our algorithms combine feature selection, multi-kernel learning, graph regularization and NMF, we compared our algorithms with the following relevant methods: the original NMF (Lee & Seung, 2000), the graph-regularized NMF (GNMF) (Cai et al., 2011), the kernel NMF (NMF$_K$) (Lee, Cichocki, & Choi, 2009), the NMF with multi-kernels (NMF$_{MK}$) (An et al., 2011) and the NMF with feature selection (NMF$_{FS}$) (Das Gupta & Xiao, 2011). In total seven different methods were compared on this data set. The ROC and recall-precision curves of these methods are shown in Fig. 1, and the AUCs are given in Table 1. It can be seen that the proposed methods using feature selection or multi-kernel learning to learn an adaptive graph for regularization of NMF model consistently perform better than the GNMF method using the original data space for the graph estimation. In particular, this difference is significant when a small number of data samples with large feature dimension are available to train the NMF. Moreover, NMF$_{FS}$ outperforms NMF$_{MK}$, which implies that feature selection works better than multiple kernel learning for high dimensional data with many noisy and irrelevant features, such as the gene expression data. We also observed that both of the proposed feature selection version (AGNMF$_{FS}$) and multi-kernel version (AGNMF$_{MK}$) of graph-adaptive NMF methods are superior to their competing algorithms that only consider feature selection (NMF$_{FS}$) or multi-kernel learning (NMF$_{MK}$) without conducting graph regularization. This is consistent with the manifold assumption of the data and also shows the necessity to apply graph regularization. It is also interesting to notice that the difference between NMF$_{MK}$ and NMF$_K$ is marginal. A possible reason is that NMF$_{MK}$ does not use the l_2 norm to regularize the kernel weights whereas NMF$_{MK}$ does, thus the kernel weights overfit to one kernel in NMF$_{MK}$.

4.2. Experiment II: face recognition

In the second experiment, we test our algorithms on the face recognition task.

4.2.1. Face image dataset and setup

We used the face image dataset from Georgia Institute of Technology (GTFD) (Nefian & Hayes, 2000) in this experiment. This database contains face images of 50 individuals. Fifteen color pictures are taken for each individual in two or three sessions. Thus there are in total 750 images in the database. For each individual, the pictures of different positions, facial expressions, lighting conditions and scales are taken. The face area in each image is manually cropped. We extracted the color-based local binary pattern (LBP) (Nanni, Lumini, & Brahnam, 2012) and the Gabor wavelet coefficients (Park & Kim, 2013) as features of each face image, and concatenated them to construct the original nonnegative feature vector.

To conduct the evaluation, we randomly split the entire database into non-overlapping training and test subsets. For each individual, 10 images out of the 15 were randomly selected as the training samples, and the remaining five images were used as the test samples. Thus there were in total 500 samples in the training subset while 250 samples in the test set. To represent the samples, we first performed the NMF algorithms to the training set to obtain the basis matrix and the coding vectors of training samples, and then the test samples were coded into coding vectors using the NMF parameters learned by the training set. To classify the test samples, we first trained a hidden Markov model (HMM) (Elliott, Siu, & Fung, 2014) for each individual using the training samples,

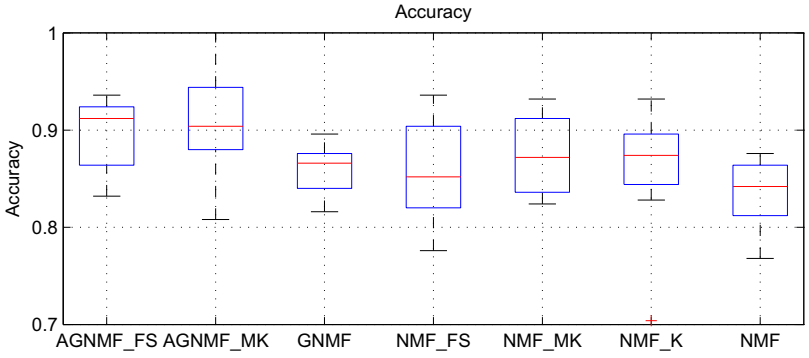


Fig. 2. Boxplots of recognition accuracies of 10 splits on the GTFD face database.

and then the test samples were fit to each of the HMM model, and classified into the one with the highest log-likelihood score. The above split process (training/test) was repeated 10 times, and the recognition accuracies over the splits were reported as the final performance.

4.2.2. Experimental results

Fig. 2 shows the boxplots of the classification accuracies of different methods over the 10 splits. It can be seen that the proposed $AGNMF_{MK}$ and $AGNMF_{FS}$ again consistently outperform other methods. $AGNMF_{MK}$ performs similarly as $AGNMF_{FS}$ on this dataset. This makes sense because for the computer vision problems, such as face recognition, multi-kernel-based methods have been shown to perform well (Yeh et al., 2013). In this data representation task, modeling the data as graphs gives much better representation than modeling them in the original Hilbert space that is constructed by a single kernel or multiple kernels. Thus, it is not surprising to see that the proposed $AGNMF_{MK}$ significantly and consistently outperforms NMF_{MK} and NMF_{K}, although all of them minimize the data reconstruction error. Similar reasons can be used to explain the improvement of $AGNMF_{FS}$ over NMF_{FS}. By comparing the performance between $AGNMF_{FS}$ and GNMF, and between NMF_{FS} and NMF, we can conclude that feature selection plays important roles in the classification accuracy.

5. Conclusion and future works

In this paper, we proposed two novel data representation methods that aimed to solve the issue of NMF caused by noisy and irrelevant features, and non-linear distributions of data samples. The first method conducts feature selection, graph regularization, and NMF simultaneously, whereas the second method optimizes multi-kernel learning, graph regularization, and NMF in a unified objective function. We developed two iterative optimization algorithms to optimize the two objective functions, respectively. Experimental results demonstrate that our methods significantly outperform state-of-the-art data representation methods on the colon cancer classification and face recognition tasks. The strength of the proposed algorithms lies on the fact that it does not need class label information for feature selection and multi-kernel learning. The weakness is the high computational complexity of the proposed multi-kernel learning method due to the QP problem in each iteration.

Manifold regularization based on graphs has been a popular method in NMF-related studies. However, the construction of the graph is often effected by the noisy features and the nonlinear distribution of the data. The theoretical contributions of this paper are to propose two solutions for these problems. One contribution of them is to use the feature selection to refine the data to construct a reliable graph to regularize the NMF, and also to incorporate the problem of feature selection to the problem of NMF. Another one is to incorporate the problem of multi-kernel learning to NMF and also to use it to refine the graph construction. We show that both feature selection and multi-kernel learning can be used to construct a more reliable graph for NMF, and the learning of feature and kernel weights can be learned simultaneously with NMF.

Moreover, we also provide some insightful and practical implications to feature selection and multi-kernel learning. Although the feature selection and multi-kernel learning are used to construct the graph for NMF, feature selection and multi-kernel learning problems are also solved by minimizing the objective of graph regularized NMF. This gives an insight about feature selection and multi-kernel learning: NMF and graph regularization can also be used as criteria for feature selection and multi-kernel learning even when the supervision information is missing. The problems of feature selection and multi-kernel learning, NMF and graph regularization can be unified as a single learning problem.

In the future, to extend the work proposed in this paper, we propose the following future research directions. The first direction is to parallelize the proposed algorithms in a distributed system to apply it to a big data platform (Kwon & Sim, 2013; Lee, Lee, & Sohn, 2013b; Li et al., 2011b; Wang, Jiang, & Agrawal, 2012c; Wang, Nandi, & Agrawal, 2014c; Wang, Su, & Agrawal, 2013d). The second direction is to improve the proposed algorithm to handle different data sets of different distributions for domain transfer learning problems (Al-Shedivat, Wang, Alzahrani, Huang, & Gao, 2014; Lee et al., 2013b; Meng, Lin, & Li, 2011). The third direction is to apply it to more applications, such as image watermarking (Ouhsain & Hamza, 2009), fault diagnosis(Li et al., 2011a), sensing (Sun, Hu, & Qi, 2010a, 2014; Li, Wu, & Li, 2014), malicious websites detection (Xu, Zhan, Xu, & Ye, 2014, 2013), data analysis (Luo & Brodsky, 2011; Luo, Brodsky, & Li, 2012), and protein sequence motif discovery (Kim, Chen, Kim, Pan, & Park, 2011b).

Acknowledgments

The study is in part supported by US National Science Foundation (Grant No. DBI-1322212) and a grant from King Abdullah University of Science and Technology (KAUST), Saudi Arabia. We would like to thank Dr. Qingquan Sun for sharing the code of the feature selection algorithm with us.

References

Al-Shedivat, M., Wang, J. J.- Y., Alzahrani, M., Huang, J. Z., Gao, X. (2014). Supervised transfer sparse coding. In *Twenty-eighth AAAI conference on artificial intelligence*(pp. 1665–1672).

An, S., Yun, J.- M., Choi, S., (2011). Multiple kernel nonnegative matrix factorization. In *2011 IEEE international conference on acoustics, speech, and signal processing. International conference on acoustics speech and signal processing ICASSP* (pp. 1976–1979).

Cai, Q., Yin, Y., Man, H., Cai, Q., Yin, Y., Man, H., (2013). Dspm: Dynamic structure preserving map for action recognition. In *2013 IEEE international conference on multimedia and expo (ICME)* (pp. 1–6).

Cai, D., He, X., Han, J., & Huang, T. S. (2011). Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(8), 1548–1560.

Chen, Z., Li, J., Wei, L., Xu, W., & Shi, Y. (2011). Multiple-kernel svm based multiple-task oriented data mining system for gene expression data analysis. *Expert Systems with Applications, 38*(10), 12151–12159.

Cui, S., & Soh, Y. C. (2010). Linearity indices and linearity improvement of 2-d tetralateral position-sensitive detector. *IEEE Transactions on Electron Devices, 57*(9), 2310–2316.

Das Gupta, M., & Xiao, J. (2011). Non-negative matrix factorization as a feature selection tool for maximum margin classifiers. In *2011 IEEE conference on computer vision and pattern recognition (CVPR)*. IEEE.

Elliott, R., Siu, T., & Fung, E. (2014). A double hmm approach to altman z-scores and credit ratings. *Expert Systems with Applications, 41*(4 PART 2), 1553–1560.

Emami, M., & Omar, K. (2013). A low-cost method for reliable ownership identification of medical images using svm and lagrange duality. *Expert Systems with Applications, 40*(18), 7579–7587.

Fakhraei, S., Soltanian-Zadeh, H., & Fotouhi, F. (2014). Bias and stability of single variable classifiers for feature ranking and selection. *Expert Systems with Applications, 41*(15), 6945–6958.

Iquebal, A., Pal, A., Ceglarek, D., & Tiwari, M. (2014). Enhancement of Mahalanobis–Taguchi system via rough sets based feature selection. *Expert Systems with Applications, 41*(17), 8003–8015.

Kim, W., Chen, B., Kim, J., Pan, Y., & Park, H. (2011a). Sparse nonnegative matrix factorization for protein sequence motif discovery. *Expert Systems with Applications, 38*(10), 13198–13207.

Kim, W., Chen, B., Kim, J., Pan, Y., & Park, H. (2011b). Sparse nonnegative matrix factorization for protein sequence motif discovery. *Expert Systems with Applications, 38*(10), 13198–13207.

Kwon, O., & Sim, J. (2013). Effects of data set features on the performances of classification algorithms. *Expert Systems with Applications, 40*(5), 1847–1857.

Lee, D. D., Seung, H. S., (2000). Algorithms for non-negative matrix factorization. In *NIPS* (pp. 556–562).

Lee, H., Cichocki, A., & Choi, S. (2009). Kernel nonnegative matrix factorization for spectral eeg feature extraction. *Neurocomputing, 72*(13-15), 3182–3190.

Lee, M., Lee, A., & Sohn, S. (2013b). Behavior scoring model for coalition loyalty programs by using summary variables of transaction data. *Expert Systems with Applications, 40*(5), 1564–1570.

Lee, L., Rajkumar, R., Lo, L., Wan, C., & Isa, D. (2013a). Oil and gas pipeline failure prediction system using long range ultrasonic transducers and Euclidean-support vector machines classification approach. *Expert Systems with Applications, 40*(6), 1925–1934.

Lin, C.-H., Chen, H.-Y., & Wu, Y.-S. (2014). Study of image retrieval and classification based on adaptive features using genetic algorithm feature selection. *Expert Systems with Applications, 41*(15), 6611–6621.

Li, H., Wu, G.-Q., Hu, X.-G., Zhang, J., Li, L., & Wu, X. (2011b). K-means clustering with bagging and mapreduce. In *2011 44th Hawaii international conference on system sciences (HICSS)* (pp. 1–8). IEEE.

Li, H., Wu, X., & Li, Z. (2014). Online learning with mobile sensor data for user recognition. In *The 29th symposium on applied computing* (pp. 64–70). ACM.

Li, H., Wu, X., Li, Z., & Ding, W. (2013a). Group feature selection with feature streams. In *2013 IEEE 13th international conference on data mining (ICDM)* (pp. 1109–1114). IEEE.

Li, H., Wu, X., Li, Z., & Ding, W. (2013b). Online group feature selection from feature streams. In *Twenty-seventh AAAI conference on artificial intelligence* (pp. 1627–1628). AAAI.

Li, B., Zhang, P.-L., Tian, H., Mi, S.-S., Liu, D.-S., & Ren, G.-Q. (2011a). A new feature extraction and selection scheme for hybrid fault diagnosis of gearbox. *Expert Systems with Applications, 38*(8), 10000–10009.

Luo, J., Brodsky, A., (2011). An em-based multi-step piecewise surface regression learning algorithm. In *The seventh international conference on data mining (WORLDCOMP DMIN 11)* (pp. 286–292). Las Vegas, Nevada.

Luo, J., Brodsky, A., & Li, Y. (2012). An em-based ensemble learning algorithm on piecewise surface regression problem. *International Journal of Applied Mathematics and Statistics, 28*(4), 59–74.

Meng, J., Lin, H., & Li, Y. (2011). Knowledge transfer based on feature representation mapping for text classification. *Expert Systems with Applications, 38*(8), 10562–10567.

Merigó, J., & Casanovas, M. (2011). Induced aggregation operators in the euclidean distance and its application in financial decision making. *Expert Systems with Applications, 38*(6), 7603–7608.

Nanni, L., Lumini, A., & Brahnam, S. (2012). Survey on lbp based texture descriptors for image classification. *Expert Systems with Applications, 39*(3), 3634–3641.

Nefian, A., Hayes, M. H. I. (2000). Maximum likelihood training of the embedded hmm for face detection and recognition. In *Proceedings 2000 international conference on image processing* (pp. 33–6). Vol. 1.

Orsenigo, C., & Vercellis, C. (2012). Kernel ridge regression for out-of-sample mapping in supervised manifold learning. *Expert Systems with Applications, 39*(9), 7757–7762.

Ouhsain, M., & Hamza, A. (2009). Image watermarking scheme using nonnegative matrix factorization and wavelet transform. *Expert Systems with Applications, 36*(2 PART 1), 2123–2129.

Park, J.-G., & Kim, K.-J. (2013). Design of a visual perception model with edge-adaptive Gabor filter and support vector machine for traffic sign detection. *Expert Systems with Applications, 40*(9), 3679–3687.

Sun, Q., Hu, F., & Hao, Q. (2014). Mobile target scenario recognition via low-cost pyroelectric sensing system: Toward a context-enhanced accurate identification. *IEEE Transactions on Systems, Man, and Cybernetics. Systems, 44*(3), 375–384.

Sun, Q., Hu, F., & Qi, H. (2010a). Context awareness emergence for distributed binary pyroelectric sensors. In *2010 IEEE conference on multisensor fusion and integration for intelligent systems (MFI)* (pp. 162–167). IEEE.

Sun, Y., Todorovic, S., & Goodison, S. (2010b). Local-learning-based feature selection for high-dimensional data analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32*(9), 1610–1626.

Sun, Q., Wu, P., Wu, Y., Guo, M., & Lu, J. (2012). Unsupervised multi-level non-negative matrix factorization model: Binary data case. *Journal of Information Security, 3*, 245.

Wang, J.-Y., Almasri, I., & Gao, X. (2012b). Adaptive graph regularized nonnegative matrix factorization via feature selection. In *2012 21st International conference on pattern recognition (ICPR)* (pp. 963–966). IEEE.

Wang, J. J.-Y., Bensmail, H., & Gao, X. (2012a). Multiple graph regularized protein domain ranking. *BMC Bioinformatics, 13*(1), 307.

Wang, J. J.-Y., Bensmail, H., & Gao, X. (2013a). Multiple graph regularized nonnegative matrix factorization. *Pattern Recognition, 46*(10), 2840–2847.

Wang, J. J.-Y., Bensmail, H., & Gao, X. (2014a). Feature selection and multi-kernel learning for sparse representation on a manifold. *Neural Networks, 51*(0), 9–16.

Wang, J. J.-Y., Bensmail, H., Yao, N., & Gao, X. (2013b). Discriminative sparse coding on multi-manifolds. *Knowledge-Based Systems, 54*, 199–206.

Wang, J. J.-Y., & Gao, X. (2013). Beyond cross-domain learning: Multiple domain nonnegative matrix factorization. *Engineering Applications of Artificial Intelligence, 28*(0), 181–189.

Wang, Y., Jiang, W., & Agrawal, G. (2012c). SciMATE: A novel mapreduce-like framework for multiple scientific data formats. In *2012 12th IEEE/ACM international symposium on cluster cloud and grid computing (CCGrid)* (pp. 443–450). IEEE.

Wang, Y., Nandi, A., & Agrawal, G. (2014c). SAGA: Array storage as a DB with support for structural aggregations. In *Proceedings of the 26th international conference on scientific and statistical database management* (pp. 9). ACM.

Wang, Y., Su, Y., & Agrawal, G. (2013d). Supporting a light-weight data management layer over HDF5. In *2013 13th IEEE/ACM international symposium on cluster, cloud and grid computing (CCGrid)* (pp. 335–342). IEEE.

Wang, J. J.-Y., Sun, Y., & Gao, X. (2014b). Sparse structure regularized ranking. *Multimedia Tools and Applications*, 1–20.

Wang, J. J.-Y., Wang, X., & Gao, X. (2013c). Non-negative matrix factorization by maximizing correntropy for cancer clustering. *BMC Bioinformatics, 14*(1), 107.

Xu, L., Zhan, Z., Xu, S., Ye, K. (2014). An evasion and counter-evasion study in malicious websites detection. In *2014 IEEE conference on communications and network security (CNS) (IEEE CNS 2014)*. San Francisco, USA.

Xu, L., Zhan, Z., Xu, S., & Ye, K. (2013). Cross-layer detection of malicious websites. In *Proceedings of the third ACM conference on data and application security and privacy* (pp. 141–152). ACM.

Yeh, C.-Y., Huang, C.-W., & Lee, S.-J. (2011). A multiple-kernel support vector regression approach for stock market price forecasting. *Expert Systems with Applications, 38*(3), 2177–2186.

Yeh, C.-Y., Su, W.-P., & Lee, S.-J. (2013). An efficient multiple-kernel learning for pattern classification. *Expert Systems with Applications, 40*(9), 3491–3499.

Zhang, T., Xu, D., & Chen, J. (2008). Application-oriented purely semantic precision and recall for ontology mapping evaluation. *Knowledge-Based Systems, 21*(8), 794–799.

Zheng, C.-H., Ng, T.-Y., Zhang, L., Shiu, C.-K., & Wang, H.-Q. (2011). Tumor classification based on non-negative matrix factorization using gene expression data. *IEEE Transactions on Nanobioscience, 10*(2), 86–93.