Self-Representative Manifold Concept Factorization with Adaptive Neighbors for Clustering

Sihan Ma[1], Lefei Zhang[1*], Wenbin Hu[1], Yipeng Zhang[1], Jia Wu[2], Xuelong Li[3]
[1]School of Computer, Wuhan University
[2]Department of Computing, Macquarie University
[3]Center for OPTIMAL, Xi'an Institute of Optics and Precision Mechanics, CAS
{sihanma,zhanglefei,hwb,zyp91}@whu.edu.cn, jia.wu@mq.edu.au, xuelong_li@opt.ac.cn

Abstract

Matrix Factorization based methods, e.g., the Concept Factorization (CF) and Nonnegative Matrix Factorization (NMF), have been proved to be efficient and effective for data clustering tasks. In recent years, various graph extensions of CF and NMF have been proposed to explore intrinsic geometrical structure of data for the purpose of better clustering performance. However, many methods build the affinity matrix used in the manifold structure directly based on the input data. Therefore, the clustering results are highly sensitive to the input data. To further improve the clustering performance, we propose a novel manifold concept factorization model with adaptive neighbor structure to learn a better affinity matrix and clustering indicator matrix at the same time. Technically, the proposed model constructs the affinity matrix by assigning the adaptive and optimal neighbors to each point based on the local distance of the learned new representation of the original data with itself as a dictionary. Our experimental results present superior performance over the state-of-the-art alternatives on numerous datasets.

1 Introduction

Data clustering is an important and significant research topic which has been widely studied in various applications, such as image segmentation [Shi and Malik, 2000], document analysis [Hammouda and Kamel, 2004] [Cai *et al.*, 2011], gene selection [Jiang *et al.*, 2004] and so on. It aims to partition a dataset into different groups based on similarities among data points such that the data points which share high similarities are assigned into the same group while the dissimilar data points are in different groups. In the past decades, a number of clustering algorithms have been proposed, including K-means [Hartigan and Wong, 1979], spectral clustering [von Luxburg, 2007] [Liu *et al.*, 2015], subspace clustering [Vidal, 2011] [Wang and Xu, 2016], and matrix-factorization-based algorithms [Wang and Zhang, 2013] [Hong *et al.*, 2016], etc.

Among the above methods, the matrix-factorization-(MF)-based methods have become popular in recent few years. Given a data matrix X, the basic goal of MF is to find two or more low-rank matrix factors whose product provides a good approximation to the original matrix X. For the MF-based clustering, one of the matrix factors can be considered as an ensemble of cluster prototypes that reveals the potential semantic structure, while the other matrix factor can be referred as the coefficient matrix that indicates the clustering results, which meets the psychological and physiological interpretation of part-based representation in human brain. Therefore, in real-world applications, the dimensionality of the decomposed matrix factors is set to the number of clusters, which is usually much smaller than that of the original one. This fact gives rise to compact representation of the data points, which can facilitate other learning tasks such as clustering and classification. The most representative algorithms of MF include Nonnegative Matrix Factorization (NMF) [Lee and Seung, 2001], Concept Factorization (CF) [Xu and Gong, 2004] [Cai *et al.*, 2011], Principal Component Analysis (PCA) and so on.

In particular, the NMF is different from the other matrix-factorization-based approaches since it enforces a constraint that all the elements of the matrix factors should be equal or greater than zero. One of the major advantages of NMF over other methods is that the inherent data nonnegativity is preserved by the NMF method, as a result of constraints that produce nonnegative lower rank factors. These factors can be interpreted as semantic features or patterns in the collection of data. Thus, data with common features can be viewed as a cluster [Shahnaz *et al.*, 2006]. In NMF, the clustering result can be easily obtained. Previous works have proved that NMF is superior to other MF-based methods in document clustering [Xu *et al.*, 2003] [Shahnaz *et al.*, 2006] and image clustering [Kim and Park, 2008]. However, there are two major limitations of NMF. First, it is still unclear how to optimally perform clustering tasks by NMF method on the data with negative inputs. Second, only original feature of the data points is applied to NMF for clustering so that the data to be clustered cannot be analyzed in a more compact and discriminative feature space to further improve the clustering

*Corresponding author.

performance.

The CF method is an important variant of NMF in which each cluster is regarded as a linear combination of the data points and at the same time each data point is also expressed as a linear combination of these cluster centers. Given a matrix *X*, it is decomposed into three matrices, i.e., W, V and X itself, among which the W and V are nonnegative, which satisfies that $X \approx XWV^T$. The major advantage of CF over NMF is that the clustering task can be performed by CF model in any feature space, including data with negative numbers and any transformed data spaces, e.g., the reproducing kernel Hilbert space, thus rendering the CF method a more general clustering method [Cai *et al.*, 2011] [Xu and Gong, 2004] [Pei *et al.*, 2016].

However, the aforementioned CF, like NMF, does not inherently explore the geometric structure of the data, which is important for clustering. Therefore, some geometry-based extensions have been proposed. Cai [Cai *et al.*, 2011] incorporates a manifold regularizer with CF to address the underlying concepts which are consistent with the intrinsic manifold structure to facilitate subsequent process, such as clustering. Pei [Pei *et al.*, 2016] extends the standard CF by utilizing a sophisticated method to learn the affinity matrix by adaptively assigning the neighbors for each data point for document clustering. However, most of these works share at least one of the following drawbacks. First, the model of the affinity matrix used in the graph regularizer is predefined (e.g., the Graph Laplacian), which means the clustering results may be sensitive to the selected model. Second, the affinity matrix is constructed on the raw data to a large extent, which is unable to well reveal the global manifold structure of data.

Alternatively, inspired by the successful way to exploit the self-representation of data in which the original data is regarded as a dictionary [Guo, 2015] [Du *et al.*, 2016], we consider CF as an improved self-representation matrix factorization method with a learning-based dictionary to efficiently reveal the global structure of original data. More importantly, since the linear coefficients of CF carry clear semantic meanings, which indicate they contain plentiful information of the cluster label for each data point [Xu and Gong, 2004]. Thus, the coefficients of CF model can be extracted to construct the affinity matrix to relieve the disadvantage brought by using the raw data. In our proposed model, we construct the affinity matrix with adaptive neighbors based on the renewable coefficient matrix and learn a sparse data representation simultaneously. In detail, several aspects of our model which worth to be highlighted are as follows.

1. The proposed algorithm is a general framework that can combine the power of CF model with two kinds of graph regularizers which contain comprehensive and complementary structure information.

2. The proposed affinity matrix of the manifold structure is constructed by adaptively assigning its nearest neighbors, so that the clustering results are not sensitive to the input data matrix. At the same time, a self-representation of original data is learned to better construct the affinity matrix.

3. A graph regularizer based on the original data is added to better reveal the local structure information in the original data and learn a sparse clustering indicator matrix simultaneously. As the weight matrices of the graphs are highly sparse, an efficient multiplicative update rule is proposed to solve the proposed optimization.

The remainder of this paper is organized as follows: The derivation of our algorithm is described in section 2. After that, the optimization algorithm is proposed in section 3. The experimental results are represented in section 4, followed by the conclusion section.

2 The Proposed Method

In this section, we describe our proposed algorithm which extends the standard CF model with the consideration of two different manifold structures. In detail, the affinity matrix of first manifold structure is constructed by adaptively assigning its nearest neighbors in each iteration based on the renewable self-representation of the raw data. The other graph regularizer is based on original data to complementally reveal the local structure information of original data and learn the cluster indicator matrix simultaneously. We will describe each part of the objective function in the following subsections.

2.1 Graph Regularizer

In the CF model $X \approx XWV^T$, the j-th row of matrix V, i.e., $v_j^T = [v_{j1}, v_{j2}, \cdots, v_{jn}]$, can be regarded as a new representation of each data point with the new basis. Therefore, some constraints can be added on matrix V to exploit the geometric structure of the data, which is beneficial for the subsequent clustering task. A natural assumption needs to be mentioned that if two data points x_i, x_j are close in the intrinsic geometry of the data distribution, then their representations in the new basis should be also close to each other [Cai *et al.*, 2011] [Pei *et al.*, 2016]. This assumption plays an important role in developing various kinds of algorithms, such as spectral clustering and spectral-based dimensionality reduction algorithms.

Before defining the introduced graph regularizer, let us construct the edge weight matrix S first. Previous studies have showed that the nearest-neighbor graph on data points can effectively model the local geometric structure. Therefore, we consider a graph with n vertices where each vertex belongs to a concept. The weight matrix S is then defined as follows:

$$S_{ij} = \begin{cases} 1, & \text{if } x_i \in N_p(x_j) \text{ or } x_j \in N_p(x_i), \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

where $N_p(x_i)$ denotes the set of p-nearest neighbors of x_i. Then the graph regularizer can be defined to measure the smoothness of the low-dimensional representations on the p-nearest neighbor graph by [Belkin and Niyogi, 2002] [Belkin and Niyogi, 2003]:

$$\begin{aligned} \mathcal{R} &= \frac{1}{2}\sum_{i,j=1}^{n} \|v_i - v_j\|^2 S_{ij} \\ &= \sum_{i=1}^{n} v_i^T v_i D_{ii} - \sum_{i,j=1}^{n} v_i^T v_j S_{ij} \\ &= tr(V^T DV) - tr(V^T SV) = tr(V^T LV), \end{aligned} \tag{2}$$

where $tr(\cdot)$ denotes the trace of a matrix. $L = D - S$ is a graph Laplacian[Chung, 1997], where S is the graph weight matrix and D is a diagonal matrix whose entries are column sums of S.

The graph regularization $\mathcal{R}$ can well reveal the local intrinsic structure contained in the original data and learn a sparse cluster indicator matrix at the same time. Although there are a few drawbacks in this manifold structure and we will solve them in the next subsection, we still incorporate it into our final objective function for the purpose of obtaining better cluster indicator matrix and the local structure information hidden in the original data.

2.2 Adaptive Neighbor Structure based on Self Representation

In the above subsection, the graph weight matrix (or affinity matrix) S is predefined, which means that it may be sensitive to the input raw data. In the literature, there are many approaches to construct the affinity matrix to solve this problem [Guo, 2015]. Recently, a new idea has been proposed to learn the affinity matrix by adaptively assigning neighbors for each data point based on the local connectivity [Nie *et al.*, 2014]. To be complete, we first introduce how to assign probabilistic neighbors in this method. For simplicity, the Euclidean distance is used as the distance measurement.

For the i-th data point x_i, the A_{ij} can be used to denote the probability of any of the data point $x_j \in [x_1, x_2, \cdots, x_n]$ (excluding itself) being connected to x_i as a neighbor. In general, a smaller distance between two points x_i and x_j indicates a larger probability A_{ij}. So the probability $A_{ij}|_{j=1}^{n}$ can be determined by solving the following problem:

$$\min_{A_i^T\mathbf{1}=1, 0\le A_i\le 1} \sum_{j=1}^{n} \|x_i - x_j\|_2^2 A_{ij}, \tag{3}$$

where $A_i \in \mathbb{R}^{n\times 1}$ is a vector with the j-th element as A_{ij}. The constraints $A_i^T\mathbf{1} = 1$ and $0 \le A_i \le 1$ are introduced to guarantee the probability property of A_i.

However, there is a problem in eq. (3) that it has a trivial solution, which means only the nearest data point can be defined as the neighbor of x_i with the probability 1, while all the other data points can not be the neighbors of x_i. Alternatively, if we solve the following problem without taking into consideration any distance information in the data:

$$\min_{A_i^T\mathbf{1}=1, 0\le A_i\le 1} \sum_{j=1}^{n} A_{ij}^2. \tag{4}$$

The ideal solution is that all the data points can be neighbors of x_i with the same probability $\frac{1}{n}$.

By integrating eq. (3) and eq. (4), we have the following optimization problem:

$$\min_{A_i^T\mathbf{1}=1, 0\le A_i\le 1} \sum_{j=1}^{n} (\|x_i - x_j\|_2^2 A_{ij} + \gamma A_{ij}^2), \tag{5}$$

where γ is a positive trade-off parameter to control the second term.

However, there is still a drawback in this method that the affinity matrix is constructed on the raw data, which is unable to well reveal the global subspace structure of data[Guo, 2015]. To solve this problem, we can naturally recall the assumption mentioned before that if two data points x_i, x_j are close in the intrinsic geometry of the data distribution, then the representations of these two data points in the new basis should also be close to each other[Cai *et al.*, 2011] [Pei *et al.*, 2016]. Therefore, we decide to employ a new representation, rather than the raw data, to construct the affinity matrix.

Meanwhile, inspired by the successful way to exploit the self-representation of data in which the original data itself is regarded as a dictionary [Guo, 2015] [Du *et al.*, 2016], we observe that CF model can be considered as an improved self-representation matrix factorization method with a learning-based dictionary to efficiently reveal the global structure of original data. More importantly, the linear coefficients of CF carry clear semantic meanings, which indicates they contain plentiful information of the cluster label for each data point[Xu and Gong, 2004]. Therefore, the CF model can be rewritten in the following form:

$$X \approx XR, s.t. R = WV^T, \tag{6}$$

where $R = WV^T$ denotes the coefficient matrix based on the dictionary of the original data matrix, which is a meaningful representation of the original data points.

Hence, by incorporating the self-representation with the adaptive neighbor structure, we have the following problem for each data point:

$$\min_{A_i^T\mathbf{1}=1, 0\le A_i\le 1} \sum_{j=1}^{n} (\|(WV^T)_i - (WV^T)_j\|_2^2 A_{ij} + \gamma A_{ij}^2). \tag{7}$$

For each data point x_i, we can use eq. (7) to assign its neighbors. Therefore, we can solve the following problem to assign the neighbors for all the data points:

$$\min_{A_i^T\mathbf{1}=1, 0\le A_i\le 1} \sum_{i=1}^{n}\sum_{j=1}^{n} (\|(WV^T)_i - (WV^T)_j\|_2^2 A_{ij} + \gamma A_{ij}^2). \tag{8}$$

With slight algebraic transformation, it gives the following formulation:

$$\min_{A_i^T\mathbf{1}=1, 0\le A_i\le 1} tr(WV^T L_A V W^T) + \gamma \|A\|_F^2, \tag{9}$$

where L_A is the Laplacian matrix of A, which is constructed in the way of $D_A - A$. The degree matrix D_A is defined as a diagonal matrix in which its i-th diagonal element is $\sum_{j=1}^{n} A_{ij}$.

2.3 The Objective Function

Finally, by integrating the adaptive neighbor structure and manifold regularizers into the original CF model, we have the overall objective function of our proposed algorithm as follows:

$$\begin{aligned} \mathcal{O} = &\|X - XWV^T\|_2^2 + \lambda_1 tr(WV^T L_A V W^T) \\ &+ \lambda_2 tr(V^T L V) + \lambda_3 \|A\|_F^2 \\ s.t.\ &W \ge 0, V \ge 0, \forall i A_i^T\mathbf{1} = 1, 1 \ge A_i \ge 0, \end{aligned} \tag{10}$$

where $X \in \mathbb{R}^{m\times n}$ denotes the original data matrix, m is the feature dimension, and n is the number of data points. $W \in \mathbb{R}^{n\times r}$ and $V \in \mathbb{R}^{n\times r}$ are the decomposed matrix factors. $A \in \mathbb{R}^{n\times n}$ is the affinity matrix which we have learned in an adaptive approach with the new data representation. λ_1, λ_2, and λ_3 are three nonnegative regularization parameters.

3 Optimization Algorithm

The objective function in eq. (10) is not convex with both W and V. Therefore, it is difficult to reach the global optimization. However, the objective function is convex if we update the variables alternatively. Thus, we introduce an iterative algorithm which can achieve a local minimum as follows.

3.1 Update W and V with fixed A

Define $K = X^TX$, then the objective function can be reformed as follows:

$$\begin{aligned}\mathcal{O} &= tr((X - XWV^T)^T(X - XWV^T)) \\ &\quad + \lambda_1 tr(WV^TL_AVW^T) + \lambda_2 tr(V^TLV) + \lambda_3 tr(A^TA) \\ &= tr(K) - 2tr(VW^TK) + tr(VW^TKWV^T) \\ &\quad + \lambda_1 tr(WV^TL_AVW^T) + \lambda_2 tr(V^TLV) + \lambda_3 tr(A^TA).\end{aligned} \tag{11}$$

Let ψ_{jk}, ϕ_{jk} be the Lagrangian multiplier for constraints $w_{jk} \geq 0$ and $v_{jk} \geq 0$, respectively, and $\Psi = [\psi_{ik}]$, $\Phi = [\phi_{ik}]$. Since variable A is fixed here, we don't consider the Lagrangian multipliers here for the constraints $\forall i\ A_i^T\mathbf{1} = 1, A_i \geq 0$. Then the Lagrangian function $\mathcal{L}$ of $\mathcal{O}$ is:

$$\begin{aligned}\mathcal{L}(\mathcal{O}) =& tr(K) - 2tr(VW^TK) + tr(VW^TKWV^T) \\ &+ \lambda_1 tr(WV^TL_AVW^T) + \lambda_2 tr(V^TLV) \\ &+ \lambda_3 tr(A^TA) + tr(\Psi W^T) + tr(\Phi V^T).\end{aligned} \tag{12}$$

The partial derivatives of $\mathcal{L}$ with respect to W and V are:

$$\frac{\partial \mathcal{L}}{\partial W} = -2KV + 2KWV^TV + 2\lambda_1 WV^TL_AV + \Psi, \tag{13}$$

$$\frac{\partial \mathcal{L}}{\partial V} = -2KW + 2VW^TKW + 2\lambda_1 L_AVW^TW + 2\lambda_2 LV + \Phi, \tag{14}$$

Using the KKT conditions $\psi_{jk}w_{jk} = 0$ and $\phi_{jk}v_{jk} = 0$, we get the following equations for w_{jk} and v_{jk}:

$$\begin{aligned}0 =& -(KV)_{jk}w_{jk} + (KWV^TV)_{jk}w_{jk} \\ &+ \lambda_1(WV^TL_AV)_{jk}w_{jk},\end{aligned} \tag{15}$$

$$\begin{aligned}0 =& -(KW)_{jk}v_{jk} + (VW^TKW)_{jk}v_{jk} \\ &+ \lambda_1(L_AVW^TW)_{jk}v_{jk} + \lambda_2(LV)_{jk}v_{jk}.\end{aligned} \tag{16}$$

Let $L_A = D_A - A$ and $L = D - S$, then the equations can lead to the following updating rules:

$$w_{jk} \leftarrow w_{jk}\frac{(KV + \lambda_1 WV^TAV)_{jk}}{(KWV^TV + \lambda_1 WV^TD_AV)_{jk}}, \tag{17}$$

$$v_{jk} \leftarrow v_{jk}\frac{(KW + \lambda_1 AVW^TW + \lambda_2 SV)_{jk}}{(VW^TKW + \lambda_1 D_AVW^TW + \lambda_2 DV)_{jk}}. \tag{18}$$

Regarding these updating rules, we have following theorem:

Theorem 1. The objective function $\mathcal{O}$ in eq. (10) is nonincreasing under the update rules in eq. (17) and eq. (18). The objective function is invariant under these updates if and only if W and V are at a stationary point.

Theorem 1 guarantees the convergence of W and V in eq. (17) and eq. (18). Meanwhile, A has a closed-form solution. So the final solution will be a local optimum. The proof of Theorem 1 is omitted because of the size limitation.

3.2 Update A with fixed W and V

The update of A can be done via solving the following optimization problem:

$$\begin{aligned}&\arg\min_A \lambda_1 tr(WV^TL_AVW^T) + \lambda_3\|A\|_F^2 \\ &s.t. \forall i A_i^T\mathbf{1} = 1; 1 \geq A_i \geq 0.\end{aligned} \tag{19}$$

Note that:

$$tr(WV^TL_AVW^T) = \frac{1}{2}\sum_{i,j=1}^{n}\|(WV^T)_i - (WV^T)_j\|_2^2 A_{ij}. \tag{20}$$

Then the problem becomes:

$$\arg\min_A \frac{\lambda_1}{2}\sum_{i,j=1}^{n}\|(WV^T)_i - (WV^T)_j\|_2^2 A_{ij} + \lambda_3\|A\|_F^2. \tag{21}$$

To simplify the procedure, denote $\gamma = \frac{2\lambda_3}{\lambda_1}$. Note that the problem (21) is independent between different i, which means we can solve the following problem individually for each i. So the problem reduces to:

$$\begin{aligned}&\min_{A_i}\sum_{j=1}^{n}(\|(WV^T)_i - (WV^T)_j\|_2^2 A_{ij} + \gamma A_{ij}^2), \\ &s.t.\ A_i\mathbf{1} = 1, 1 \geq A_i \geq 0.\end{aligned} \tag{22}$$

Denote $d_{ij} = \|(WV^T)_i - (WV^T)_j\|_2^2$ and denote $d_i \in \mathbb{R}^{n\times 1}$ as a vector with the j-th element as d_{ij}, then the problem can be written in the vector form as:

$$\min_{A_i^T=1, 1\geq A_i\geq 0}\|A_i + \frac{1}{2\gamma}d_i\|_2^2. \tag{23}$$

For each i, the Lagrangian function is:

$$\mathcal{L}(A_i, \eta, \beta_i) = \frac{1}{2}\|A_i + \frac{d_i}{2\gamma_i}\|_2^2 - \eta(A_i^T\mathbf{1} - 1) - \beta_i^TA_i, \tag{24}$$

where η and $\beta_i \geq \mathbf{0}$ are the Lagrangian Multipliers.

According to the KKT conditions, it can be verified that the optimal solution A_i should be:

$$A_{ij} = (-\frac{d_{ij}}{2\gamma_i} + \eta)_+. \tag{25}$$

To have a sparse similarity A, we have:

$$\eta = \frac{1}{k} + \frac{1}{2k\gamma_i}\sum_{j=1}^{k} d_{ij}, \tag{26}$$

where k is the number of the nearest neighbors.

We could set γ_i to be:

$$\gamma_i = \frac{k}{2} d_{i,k+1} - \frac{1}{2}\sum_{j=1}^{k} d_{ij}. \tag{27}$$

The overall γ could be set to the mean of $\gamma_1, \gamma_2, \ldots, \gamma_n$, which is:

$$\gamma = \frac{1}{n}\sum_{i=1}^{n}(\frac{k}{2} d_{i,k+1} - \frac{1}{2}\sum_{j=1}^{k} d_{ij}). \tag{28}$$

3.3 Extension to Negative Data Matrices

The updating rules we introduced in the above section only take effect when the K is nonnegative. However, it is possible that the K contains negative entries. So in this section, we will present a more general algorithm to handle any case. Following [Xu and Gong, 2004], the algorithm is based on the theorem proposed by Sha [Sha *et al.*, 2007], which is stated as follows.

Theorem 2. Define the nonnegative general quadratic form as:

$$f(v) = \frac{1}{2} v^T A v + b^T v,$$

where v is an m-dimensional nonnegative vector, A is a symmetric positive definite matrix and b is an arbitrary m-dimensional vector. Let A^+ and A^- denote the nonnegative matrices with elements:

$$A^+_{ij} = \begin{cases} A_{ij}, & \text{if } A_{ij} > 0, \\ 0, & \text{otherwise.} \end{cases} \quad A^-_{ij} = \begin{cases} |A_{ij}|, & \text{if } A_{ij} < 0, \\ 0, & \text{otherwise.} \end{cases}$$

It is easy to see that $A = A^+ - A^-$. Then, we can obtain the solution v that minimizes $f(v)$ through the following iterative updating:

$$v_i \leftarrow v_i \left[\frac{-b_i + \sqrt{b_i^2 + 4(A^+ v)_i (A^- v)_i}}{2(A^+ v)_i} \right]. \tag{29}$$

The Theorem 2 can naturally be applied because the objective function $\mathcal{O}$ is a quadratic form of W (or V). We need to identify the corresponding A and b in the objective function.

Fixing V, the part b can be attained by taking the first order derivative with respect to W at $W = 0$:

$$\left.\frac{\partial \mathcal{O}}{\partial w_{jk}}\right|_{W=0} = -2(KV)_{jk}. \tag{30}$$

The part A for the quadratic form $\mathcal{O}(W)$ can be obtained by taking the second order derivative with respect to W:

$$\frac{\partial^2 \mathcal{O}}{\partial w_{jk} \partial w_{il}} = 2(K)_{ji}(V^T V)_{lk} + 2\lambda_1 \delta_{ji}(V^T L_A V)_{lk}, \tag{31}$$

where

$$\delta_{ji} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases} \quad \delta_{lk} = \begin{cases} 1, & \text{if } l = k, \\ 0, & \text{otherwise.} \end{cases}$$

Let K^+ and K^- denote the nonnegative matrices with elements:

$$K^+_{ij} = \begin{cases} K_{ij}, & \text{if } K_{ij} > 0, \\ 0, & \text{otherwise.} \end{cases} \quad K^-_{ij} = \begin{cases} |K_{ij}|, & \text{if } K_{ij} < 0, \\ 0, & \text{otherwise.} \end{cases}$$

where we have $K = K^+ - K^-$. Substituting A and b_i using eq. (30) and eq. (31), we get the multiplicative updating equation for each element w_{jk} of W:

$$w_{jk} \leftarrow w_{jk} \frac{(KV)_{jk} + \sqrt{(KV)^2_{jk} + 4P^+_{jk} P^-_{jk}}}{2P^+_{jk}}, \tag{32}$$

where $P^+ = K^+ W V^T V + \lambda_1 W V^T D_A V$ and $P^- = K^- W V^T V + \lambda_1 W V^T A V$.

Similarly, we can get the updating equation for each element v_{jk} in V as following:

$$v_{jk} \leftarrow v_{jk} \frac{(KW)_{jk} + \sqrt{(KW)^2_{jk} + 4Q^+_{jk} Q^-_{jk}}}{2Q^+_{jk}}, \tag{33}$$

where $Q^+ = V W^T K^+ W + \lambda_1 D_A V W^T W + \lambda_2 D V$ and $Q^- = V W^T K^- W + \lambda_1 A V W^T W + \lambda_2 S V$.

4 Experimental Results

In this section, we evaluate the performance of our proposed method on some datasets to show the effectiveness of our algorithm. We compare our method with some exist algorithms including the K-means, CF [Xu and Gong, 2004], NMF [Lee and Seung, 2001], SMCE [Elhamifar and Vidal, 2011], SSC [Elhamifar and Vidal, 2013], and Normalized Cuts (Ncut) [Shi and Malik, 2000]. For all the clustering methods, the number of clusters is known as input.

Datasets. There are in total six datasets used in our experiments, all from the UCI Machine Learning Repository. Table 2 summarizes the characteristics of the datasets used in our experiments. All the datasets are nonnegative.

Evaluation Metrics. Following [Xu *et al.*, 2003] [Huang *et al.*, 2014], we adopt three widely used evaluation metrics to quantitatively measure the clustering performance of our algorithm, i.e. Clustering Accuracy, Normalized Mutual Information and Purity.

Parameters Setting. To compare these methods fairly, we run them with some selected parameter combinations and report the best result for comparison.

For K-means, NMF, CF and Ncut, we run them for 10 times and calculate both of the mean and standard deviation. The experimental results of SMCE, SSC and our method are relatively stable. So there is no need to average them. Thus, they are only run for 1 time. During the experiments, we set the cluster number and dimension of reduced data representation equal to the number of ground truth classes for all datasets and methods. For Ncut and our method, we construct the predefined Laplacian matrices by 0-1 weight based on the Euclidean distances between each data point. Also, the neighbor size of these fixed Laplacian matrices is set to be 5 for simplicity. For the Laplacian matrix in the adaptive

Datasets	Metric	K-means	CF	NMF	SMCE	SSC	Ncut	ours
Scale	AC	52.38 ± 3.99	49.57 ± 6.79	52.22 ± 8.70	46.72	51.52	63.62 ± 0.77	**70.08**
	NMI	12.54 ± 7.00	8.96 ± 7.13	14.62 ± 10.52	2.08	12.97	22.61 ± 0.85	**22.76**
	purity	66.67 ± 4.39	63.62 ± 6.77	65.50 ± 10.34	47.68	68.96	72.80 ± 0.51	**75.20**
Zoo	AC	74.95 ± 4.50	60.00 ± 5.85	77.72 ± 6.28	46.53	47.52	45.74 ± 3.39	**87.13**
	NMI	75.44 ± 5.28	62.04 ± 1.92	70.63 ± 4.69	42.15	56.81	34.91 ± 2.05	**83.97**
	purity	83.37 ± 2.13	78.91 ± 1.55	80.79 ± 3.34	64.36	70.30	58.22 ± 2.37	**87.13**
Wine	AC	92.70 ± 0.00	38.15 ± 2.24	69.44 ± 14.25	44.94	94.94	50.45 ± 3.91	**96.07**
	NMI	77.03 ± 1.47	1.13 ± 0.72	49.54 ± 13.92	5.48	81.89	16.62 ± 3.36	**86.86**
	purity	92.70 ± 0.00	41.35 ± 1.78	70.39 ± 13.24	44.94	94.94	51.46 ± 3.09	**96.07**
Iris	AC	87.60 ± 19.12	60.20 ± 10.63	75.07 ± 5.87	64.00	96.67	36.80 ± 0.28	**97.33**
	NMI	83.45 ± 13.45	36.61 ± 19.37	58.35 ± 6.09	32.19	88.46	1.23 ± 0.10	**91.35**
	purity	90.67 ± 12.65	61.40 ± 9.83	75.07 ± 5.87	64.00	96.67	38.00 ± 0.00	**97.33**
Chess	AC	51.86 ± 2.34	50.58 ± 0.35	50.57 ± 0.15	52.25	51.35	51.81 ± 0.00	**54.10**
	NMI	0.23 ± 0.61	0.01 ± 0.01	5.26e-3 ± 3.20e-3	9.83e-3	0.31	5.50e-3 ± 0.00	**2.03**
	purity	52.85 ± 1.99	52.22 ± 0.00	52.22 ± 0.00	52.25	52.22	52.22 ± 0.00	**54.10**
Vote	AC	79.77 ± 0.00	59.26 ± 7.36	80.69 ± 0.77	56.55	63.68	55.86 ± 0.00	**82.99**
	NMI	28.96 ± 0.15	4.26 ± 5.69	29.80 ± 1.17	7.95	3.34	0.36 ± 0.00	**31.73**
	purity	79.77 ± 0.00	63.63 ± 3.98	80.69 ± 0.77	61.38	63.68	61.38 ± 0.00	**82.99**

Table 1: Clustering results ((mean± standard deviation)%) of different algorithms on six datasets

Dataset	Number of Samples	Dimensions	Classes
scale	625	4	3
zoo	101	16	7
wine	178	13	3
iris	150	4	3
chess	3196	36	2
vote	435	16	2

Table 2: Description of datasets

neighbor structure in our model, the neighbor size is set by searching in the range of $\{c-1, c, c+1, c+2, c+3, c+4\}$, where c is the cluster number. For our method, the regularization parameters λ_1 and λ_2 are set by searching from $\{10^{-5}, 10^{-4}, \cdots, 10^{4}, 10^{5}\}$. Since the regularization parameter λ_3 can update automatically in each iteration based on the value of λ_1 and γ, we just initialize it empirically. We have also initialized the W and V by PCAN [Nie *et al.*, 2014]. For SSC and SMCE, we use the codes provided by their authors with the recommended parameters to achieve a good performance. Note that there is no parameter selection required for the K-means, NMF and CF, since the number of clusters is given.

Clustering Results. Table 1 shows the clustering results on six datasets in terms of accuracy, NMI and purity, respectively. In the table, we can see that regardless of the dataset, our proposed model always achieve the best clustering performance in terms of all the measurements. In addition, we can find some other detailed points:

1. For three matrix-factorization-based methods (NMF, CF, our model), our model achieves a significant improvement over both CF and NMF, which confirms that our model has a better ability to capture the intrinsic geometrical structure of the original data by considering a self-representative manifold regularizer with an adaptive neighbor structure.

2. Although some methods like SMCE have built a similarity graph to explore the data manifold, our model construct a better similarity matrix based on self-representation and employ an adaptive way to assign neighbors so that achieves better performance.

3. The clustering performance of our method on datasets which are of high dimensionality, such as Chess, Vote, is extremely better than other algorithms, especially in the measurement of NMI, which indicates our model can effectively handle the clustering problem of high dimensional data comparing with other methods. In these tables, we can observe that most methods produce low NMI, less than 1%, on the data set Chess. One possible reason is that the data in Chess is both high dimensional and diverse, which renders it difficult for clustering. Nevertheless, our method still has a relative higher NMI, 2.03% comparing with other methods.

For datasets containing categorical attributes, such as Zoo and Vote, our method also presents a great advantage over other methods in all the three measurements. One possible reason may be that the two kinds of manifold structures in our algorithm can better extract comprehensive information of the original data, no matter it is numerical, categorical or mixed.

5 Conclusion

In this paper, we propose a novel graph regularized concept factorization model with adaptive neighbor structure for data clustering. Different from other graph regularized clustering methods, our model constructs the affinity matrix by adaptively assigning closest neighbors for each data point. The manifold regularization term based on a learned affinity matrix can better consider the intrinsic geometric structure of data. Besides, the distances between each point used to build the affinity matrix are calculated based on a sparse self-representation of the original data. Therefore, the clustering result is not sensitive to the input data. An efficient multiplicative update rule is proposed to optimize the problem. Experimental results on several benchmark datasets show that our algorithm outperforms many state-of-art clustering methods. In the further research, we may extend our current algorithm to be implemented more effectively to address the task of big data clustering.

Acknowledgments

This paper is supported in part by the National Natural Science Foundation of China under grants U1536204, 61771349, 61711530239, and in part by the MQNS Grant No. 9201701203 and the MQ Enterprise Partnership Scheme Pilot Res Grant No. 9201701455.

References

[Belkin and Niyogi, 2002] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Proceedings of Advances in Neural Information Processing Systems*, pages 585–591, 2002.

[Belkin and Niyogi, 2003] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

[Cai *et al.*, 2011] D. Cai, X. He, and J. Han. Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering*, 23(6):902–913, June 2011.

[Chung, 1997] Fan RK Chung. *Spectral graph theory*. Number 92. American Mathematical Soc., 1997.

[Du *et al.*, 2016] S. Du, W. Wang, and Y. Ma. Graph regularized compact self-representative decomposition for image representation. In *Proceedings of Chinese Control and Decision Conference*, pages 3955–3959, May 2016.

[Elhamifar and Vidal, 2011] Ehsan Elhamifar and René Vidal. Sparse manifold clustering and embedding. In *Proceedings of Advances in Neural Information Processing Systems*, pages 55–63, 2011.

[Elhamifar and Vidal, 2013] Ehsan Elhamifar and Rene Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.

[Guo, 2015] Xiaojie Guo. Robust subspace segmentation by simultaneously learning data representations and their affinity matrix. In *Proceedings of International Conference on Artificial Intelligence*, pages 3547–3553, 2015.

[Hammouda and Kamel, 2004] K. M. Hammouda and M. S. Kamel. Efficient phrase-based document indexing for web document clustering. *IEEE Transactions on Knowledge and Data Engineering*, 16(10):1279–1296, Oct 2004.

[Hartigan and Wong, 1979] J.A. Hartigan and M.A. Wong. A k-means clustering algorithm: Algorithm as 136. 28:100–108, 01 1979.

[Hong *et al.*, 2016] Seunghoon Hong, Jonghyun Choi, Jan Feyereisl, Bohyung Han, and Larry S Davis. Joint image clustering and labeling by matrix factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1411–1424, 2016.

[Huang *et al.*, 2014] Jin Huang, Feiping Nie, Heng Huang, and Chris Ding. Robust manifold nonnegative matrix factorization. *ACM Transactions on Knowledge Discovery from Data*, 8(3):11, 2014.

[Jiang *et al.*, 2004] Daxin Jiang, Chun Tang, and Aidong Zhang. Cluster analysis for gene expression data: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 16(11):1370–1386, Nov 2004.

[Kim and Park, 2008] Jingu Kim and Haesun Park. Sparse nonnegative matrix factorization for clustering. Technical report, Georgia Institute of Technology, 2008.

[Lee and Seung, 2001] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proceedings of Advances in Neural Information Processing Systems*, pages 556–562, 2001.

[Liu *et al.*, 2015] Hongfu Liu, Tongliang Liu, Junjie Wu, Dacheng Tao, and Yun Fu. Spectral ensemble clustering. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 715–724. ACM, 2015.

[Nie *et al.*, 2014] Feiping Nie, Xiaoqian Wang, and Heng Huang. Clustering and projected clustering with adaptive neighbors. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 977–986. ACM, 2014.

[Pei *et al.*, 2016] Xiaobing Pei, Chuanbo Chen, and Weihua Gong. Concept factorization with adaptive neighbors for document clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 2016.

[Sha *et al.*, 2007] Fei Sha, Yuanqing Lin, Lawrence K Saul, and Daniel D Lee. Multiplicative updates for nonnegative quadratic programming. *Neural Computation*, 19(8):2004–2031, 2007.

[Shahnaz *et al.*, 2006] Farial Shahnaz, Michael W Berry, V Paul Pauca, and Robert J Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, 2006.

[Shi and Malik, 2000] Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, Aug 2000.

[Vidal, 2011] R. Vidal. Subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, March 2011.

[von Luxburg, 2007] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, Dec 2007.

[Wang and Xu, 2016] Yu-Xiang Wang and Huan Xu. Noisy sparse subspace clustering. *Journal of Machine Learning Research*, 17(12):1–41, 2016.

[Wang and Zhang, 2013] Y. X. Wang and Y. J. Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, June 2013.

[Xu and Gong, 2004] Wei Xu and Yihong Gong. Document clustering by concept factorization. In *Proceedings of Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 202–209, 2004.

[Xu *et al.*, 2003] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 267–273, 2003.