Regularized ℓ^1-Graph for Data Clustering

Yingzhen Yang[1]
yyang58@ifp.uiuc.edu
Zhangyang Wang[1]
zwang119@ifp.uiuc.edu
Jianchao Yang[2]
jiayang@adobe.com
Jiawei Han[1]
hanj@cs.uiuc.edu
Thomas S. Huang[1]
huang@ifp.uiuc.edu

[1] University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
[2] Adobe Research
San Jose, CA 95110, USA

Abstract

ℓ^1-Graph has been proven to be effective in data clustering, which partitions the data space by using the sparse representation of the data as the similarity measure. However, the sparse representation is performed for each datum independently without taking into account the geometric structure of the data. Motivated by ℓ^1-Graph and manifold leaning, we propose Regularized ℓ^1-Graph (Rℓ^1-Graph) for data clustering. Compared to ℓ^1-Graph, the sparse representations of Rℓ^1-Graph are regularized by the geometric information of the data. In accordance with the manifold assumption, the sparse representations vary smoothly along the geodesics of the data manifold through the graph Laplacian constructed by the sparse codes. Experimental results on various data sets demonstrate the superiority of our algorithm compared to ℓ^1-Graph and other competing clustering methods.

1 Introduction

Clustering is a common and important unsupervised data analysis method which partitions data into a set of self-similar clusters, and the clustering results always serve as indispensable input to other algorithms in machine learning and computer vision, or the clusters themselves reveal important patterns of the data.

Most clustering algorithms fall into two categories: similarity-based and model-based clustering methods. Model-based clustering methods usually statistically model the data by a mixture of parametric distributions, and the parameters of the distributions are estimated via fitting the statistical model to the data [9]. The representative model-based clustering is Gaussian Mixture Model (GMM), which assumes that the data are generated from a mixture of Gaussians and the parameters of the Gaussian distributions are estimated by Maximum Likelihood through the Expectation-Maximization algorithm [4, 9]. GMM-based clustering achieves satisfactory results and it has been broadly applied to machine learning, pattern recognition and computer vision [11, 16, 18, 22].

Although model-based clustering methods possess clear statistical interpretations, it is difficult to estimate parameters of the distributions for high dimensional data, which is the case in many real applications. In addition, the real data may not be generated from the assumed statistical models. In contrast, similarity-based clustering methods partition the data based on the similarity function and alleviate the difficult parameter estimation in case of high dimensionality. K-means [8] finds a local minima of sum of within-cluster dissimilarities, and spectral clustering [14] identifies clusters of complex shapes lying on some low dimensional manifolds. ℓ^1-graph [6, 17], which builds the graph by reconstructing each datum with all the other data, has been shown to be robust to noise and capable of producing superior results for high dimensional data, compared to K-means and spectral clustering. Compared to k-nearest-neighbor graph and ε-ball graph, ℓ^1-graph adaptively determines the neighborhood of each datum by solving sparse representation problem locally. We introduce sparse coding and ℓ^1-graph in the next subsection.

1.1 Sparse Coding and ℓ^1-Graph for Clustering

The aim of sparse coding is to represent an input vector by only a few sparse coefficients, called the sparse code, over a dictionary which is usually over-complete. It has been widely applied in machine learning and signal processing, and extensive literature has demonstrated the convincing performance of sparse code as a discriminative and robust feature representation [19]. Denote the data by $\boldsymbol{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$, where $\mathbf{x}_i$ lies in the d-dimensional Euclidean space $\mathbb{R}^d$, and let the dictionary matrix be $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_p] \in \mathbb{R}^{d\times p}$ with each $\mathbf{d}_m\,(m = 1, \ldots, p)$ being the atom or the basis vector of the dictionary. Sparse coding method searches for the linear sparse representation with respect to the dictionary $\mathbf{D}$ for each datum $\mathbf{x}_i$. Sparse coding is performed by the following convex optimization

$$\boldsymbol{\alpha}^i = \arg\min_{\boldsymbol{\alpha}^i} ||\boldsymbol{\alpha}^i||_1 \quad s.t. \quad \mathbf{x}_i = \mathbf{D}\boldsymbol{\alpha}^i \quad i = 1, \ldots, n \tag{1}$$

In [6], the authors applied the idea of sparse coding to data clustering and subspace learning applications, and constitute the ℓ_1-graph. Given the data $\boldsymbol{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, ℓ^1-graph seeks for the robust sparse representation for the entire data by solving the ℓ_1-norm optimization problem for each data point:

$$\min_{\boldsymbol{\alpha}^i} \|\boldsymbol{\alpha}^i\|_1 \quad s.t.\ \mathbf{x}_i = \boldsymbol{X}\boldsymbol{\alpha}^i \quad i = 1, \ldots, n \tag{2}$$

where $\boldsymbol{\alpha}^i \in \mathbb{R}^{n\times 1}$, and we denote by $\boldsymbol{\alpha}$ the coefficient matrix $\boldsymbol{\alpha} = [\boldsymbol{\alpha}^1, \ldots, \boldsymbol{\alpha}^n] \in \mathbb{R}^{n\times n}$ with the element $\boldsymbol{\alpha}_{ij} = \boldsymbol{\alpha}_i^j$. To avoid trivial solution that $\boldsymbol{\alpha} = \mathbf{I}_n$ ($n \times n$ identity matrix), it is required that the diagonal elements of $\boldsymbol{\alpha}$ be zero, i.e. $\boldsymbol{\alpha}_{ii} = 0$ for $1 \le i \le n$. ℓ_1-graph features robustness to data noise and an adaptive neighborhood, specified by the non-zero entries in the sparse codes, for each datum. Let $G = (\boldsymbol{X}, \mathbf{W})$ be the ℓ^1-graph where $\boldsymbol{X}$ is the set of vertices, $\mathbf{W}$ is the graph weight matrix and $\mathbf{W}_{ij}$ indicates the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. ℓ^1-graph sets the $n \times n$ matrix $\mathbf{W}$ as

$$\mathbf{W} = (|\boldsymbol{\alpha}| + |\boldsymbol{\alpha}^T|)/2 \tag{3}$$

where $|\boldsymbol{\alpha}|$ is the matrix whose elements are the absolute values of $\boldsymbol{\alpha}$, and then feed $\mathbf{W}$ as the pairwise similarity matrix into the spectral clustering algorithm to get the clustering result. It achieves better performance than spectral clustering with pairwise similarity matrix set by Gaussian kernel which is widely used in a variety of machine learning tasks. It should be

emphasized that the pairwise similarity matrix (3) constructed by the coefficient matrix $\boldsymbol{\alpha}$ leads to the superior performance of ℓ^1-graph based clustering.

However, ℓ^1-graph performs sparse representation for each datum separately and it sacrifices the potential of the geometric structure of the data especially in case of high dimensionality. In the next section, we introduce Regularized ℓ^1-Graph, which incorporates the information of the manifold structure of the data into the construction of the sparse graph.

2 Regularized ℓ^1-Graph

High-dimensional data always lie on or close to a submanifold of low intrinsic dimension, and clustering the data according to its underlying manifold structure is important and challenging in computer vision and machine learning. While ℓ^1-graph demonstrates better performance than many traditional similarity-based clustering methods, it performs sparse representation for each datum independently without considering the geometric information and manifold structure of the entire data. In order to obtain the sparse representations that account for the geometric information and manifold structure of the data, we employ the manifold assumption [2] and propose a novel Regularized ℓ^1-Graph (Rℓ^1-Graph). The manifold assumption in this case requires that if two points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close in the intrinsic geometry of the submanifold, their corresponding sparse codes $\boldsymbol{\alpha}^i$ and $\boldsymbol{\alpha}^j$ are also expected to be similar to each other. In other words, $\boldsymbol{\alpha}$ varies smoothly along the geodesics in the intrinsic geometry (See Figure 1). Based on the spectral graph theory [7], extensive literature uses graph Laplacian to impose local smoothness of the embedding to preserve the local manifold structure [2, 10, 12, 13, 20]. Given a proper pairwise similarity matrix $\mathbf{W}$, the sparse code $\boldsymbol{\alpha}$ that captures the local geometric structure of the data in accordance with the manifold assumption should minimize the following regularization term:

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{W}_{ij}\|\boldsymbol{\alpha}^i-\boldsymbol{\alpha}^j\|_2^2=\mathrm{Tr}(\boldsymbol{\alpha}\mathbf{L}_{\mathbf{W}}\boldsymbol{\alpha}^T) \tag{4}$$

where $\mathbf{L}_{\mathbf{W}}$ is defined as

$$\mathbf{L}_{\mathbf{W}}=\frac{1}{2}(\mathbf{D}_{\mathbf{W}}+\tilde{\mathbf{D}}_{\mathbf{W}})-\mathbf{W} \tag{5}$$

wherein $\mathbf{D}_{\mathbf{W}}$ and $\tilde{\mathbf{D}}_{\mathbf{W}}$ are diagonal matrices with diagonal elements $(\mathbf{D}_{\mathbf{W}})_{ii}=\sum_{j=1}^{n}\mathbf{W}_{ij}$ and $(\tilde{\mathbf{D}}_{\mathbf{W}})_{ii}=\sum_{j=1}^{n}\mathbf{W}_{ji}$. $\mathbf{L}_{\mathbf{W}}$ is the graph Laplacian using the symmetric pairwise similarity matrix $\mathbf{W}$, and (5) also allows for nonsymmetric $\mathbf{W}$. Let $\mathbf{A}$ be a KNN adjacency matrix, and $\mathbf{A}_{ij}=1$ if and only if either $\mathbf{x}_i$ is among the K-nearest neighbors of $\mathbf{x}_j$. The KNN adjacency matrix $\mathbf{A}$ encourages local smoothness of the sparse codes in a neighborhood of each data point without considering data that are far away from each other. Motivated by the local smoothness of the embedding and the effectiveness of the pairwise similarity matrix constructed by the sparse codes (3) in clustering, we propose to use $\mathbf{W}=(\mathbf{A}\circ|\boldsymbol{\alpha}|+\mathbf{A}^T\circ|\boldsymbol{\alpha}^T|)/2$ in the regularization term (4), and $\circ$ indicates the entrywise product.

It should be emphasized that our regularization term uses the graph Laplacian constructed by the sparse codes, which exhibits superior clustering performance compared to the Laplacian regularization used by previous works [10, 12] including Laplacian regularized sparse

coding [10]. In Laplacian regularization, the pairwise similarity matrix $\mathbf{W}$ is set by the Gaussian kernel. In contrast, the pairwise similarity matrix constructed by the sparse codes enables Rℓ^1-Graph to learn the sparse codes that are optimal in both sparsely representing the data and modeling the pairwise similarity between the data. The resultant sparse codes leads to better clustering performance evidenced by our experimental results. By incorporat-

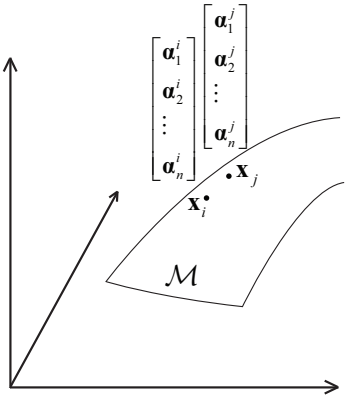

Figure 1: Illustration of the manifold assumption used in our Rℓ^1-Graph. This figure shows an example of a two-dimensional submanifold $\mathcal{M}$ in the three-dimensional ambient space. Two neighboring points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the submanifold are supposed to have similar sparse codes, i.e. $\boldsymbol{\alpha}^i = [\boldsymbol{\alpha}_1^i, \ldots, \boldsymbol{\alpha}_n^i]^T$ and $\boldsymbol{\alpha}^j = [\boldsymbol{\alpha}_1^j, \ldots, \boldsymbol{\alpha}_n^j]^T$, according to the manifold assumption.

ing the Laplacian regularizer (4) into the ℓ^1-graph scheme, we obtain the following objective function for Rℓ^1-Graph:

$$\begin{aligned} &\min_{\boldsymbol{\alpha},\mathbf{W}} \quad \sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{X}\boldsymbol{\alpha}^i\|_2^2 + \lambda\|\boldsymbol{\alpha}^i\|_1 + \gamma \mathrm{Tr}(\boldsymbol{\alpha}\mathbf{L}_{\mathbf{W}}\boldsymbol{\alpha}^T) \\ &s.t.\ \mathbf{W} = (\mathbf{A} \circ |\boldsymbol{\alpha}| + \mathbf{A}^T \circ |\boldsymbol{\alpha}^T|)/2 \quad \boldsymbol{\alpha} \in S \end{aligned} \tag{6}$$

where $S = \{\boldsymbol{\alpha} \in \mathbb{R}^{n\times n} | \boldsymbol{\alpha}_{ii} = 0, 1 \le i \le n\}$, $\lambda > 0$ is the weight controlling the sparsity of the coefficients, and $\gamma > 0$ is the weight of the regularization term.

We further reformulate the optimization problem (6) into the following optimization problem with simplified equality constraint:

$$\begin{aligned} &\min_{\boldsymbol{\alpha},\mathbf{W}} \quad \sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{X}\boldsymbol{\alpha}^i\|_2^2 + \lambda\|\boldsymbol{\alpha}^i\|_1 + \gamma \mathrm{Tr}(\boldsymbol{\alpha}\mathbf{L}_{\mathbf{A}\circ|\mathbf{W}|}\boldsymbol{\alpha}^T) \\ &s.t.\ \mathbf{W} = \boldsymbol{\alpha} \quad \boldsymbol{\alpha} \in S \end{aligned} \tag{7}$$

Note that $\mathrm{Tr}(\boldsymbol{\alpha}\mathbf{L}_{\mathbf{A}\circ|\mathbf{W}|}\boldsymbol{\alpha}^T) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \mathbf{A}_{ij}|\mathbf{W}_{ij}|\|\boldsymbol{\alpha}^i - \boldsymbol{\alpha}^j\|_2^2$. Proposition 1 establishes the equivalence between the problem (7) and problem (6).

Proposition 1. *The solution $\boldsymbol{\alpha}^*$ to the problem (7) is also the solution to the problem (6), and vice versa.*

The proof is shown in the supplementary document. The equality constraint of the new formulation (7) removes the $|\cdot|$ operator and the transpose of the coefficient matrix $\boldsymbol{\alpha}$. (7) leads to a more tractable augmented Lagrangian function than its preliminary form (6), which facilitates the optimization algorithm shown in the next section.

3 Optimization Algorithm

We employ Alternating Direction Method of Multipliers (ADMM) [3, 5] to solve the nonconvex optimization problem (7). ADMM decomposes the original problem (7) into a sequence of tractable subproblems which can be solved efficiently. ADMM iteratively minimizes the augmented Lagrangian with respect to each primal variable, and the augmented Lagrangian function for the constrained optimization (7) is

$$\mathcal{L}(\boldsymbol{\alpha},\mathbf{W},\mathbf{Y}) = \sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{X}\boldsymbol{\alpha}^i\|_2^2 + \lambda\|\boldsymbol{\alpha}^i\|_1 + \gamma \mathrm{Tr}(\boldsymbol{\alpha}\mathbf{L}_{\mathbf{A}\circ|\mathbf{W}|}\boldsymbol{\alpha}^T) + \langle \mathbf{Y}, \mathbf{W}-\boldsymbol{\alpha}\rangle + \frac{\beta}{2}\|\mathbf{W}-\boldsymbol{\alpha}\|_F^2 \quad (8)$$

where $\langle \mathbf{A},\mathbf{B}\rangle = \mathrm{Tr}(\mathbf{A}^T\mathbf{B})$ is the Frobenius inner product, $\mathbf{Y}$ is the dual variable or the Lagrangian multiplier, and β is a pre-set small positive constant called penalty parameter.

By ADMM, the optimization of (7) consists of the following iterative optimizations:

$$\boldsymbol{\alpha}^{(k)} = \arg\min_{\boldsymbol{\alpha}} \mathcal{L}(\boldsymbol{\alpha}, \mathbf{W}^{(k-1)}, \mathbf{Y}^{(k-1)}) \quad (9)$$

$$\mathbf{W}^{(k)} = \arg\min_{\mathbf{W}} \mathcal{L}(\boldsymbol{\alpha}^{(k)}, \mathbf{W}, \mathbf{Y}^{(k-1)}) \quad (10)$$

$$\mathbf{Y}^{(k)} = \mathbf{Y}^{(k-1)} + \beta(\mathbf{W}^{(k)} - \boldsymbol{\alpha}^{(k)}) \quad (11)$$

where the superscript $k \geq 1$ is the current iteration index. From (11) we can see that the penalty parameter β is also the step size for updating the Lagrangian multiplier $\mathbf{Y}$. We explain the subproblems (9) and (10) in detail in the sequel, and we also remove the superscript k for simplicity of the presentation without confusion.

- Subproblem (9): update $\boldsymbol{\alpha}$ while fixing $\mathbf{W}$ and $\mathbf{Y}$

$$\min_{\boldsymbol{\alpha}\in S} \sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{X}\boldsymbol{\alpha}^i\|_2^2 + \lambda\|\boldsymbol{\alpha}^i\|_1 + \gamma \mathrm{Tr}(\boldsymbol{\alpha}\mathbf{L}_{\mathbf{A}\circ|\mathbf{W}|}\boldsymbol{\alpha}^T) - \langle \mathbf{K}, \boldsymbol{\alpha}\rangle + \frac{\beta}{2}\langle \boldsymbol{\alpha},\boldsymbol{\alpha}\rangle \quad (12)$$

where $\mathbf{K} = \mathbf{Y} + \beta\mathbf{W}$. We denote by $F(\boldsymbol{\alpha})$ the objective function (12), and use coordinate descent algorithm to solve problem (12). In each step of the coordinate descent, we optimize $F(\boldsymbol{\alpha}^i)$ with $\left[\boldsymbol{\alpha}^1,\ldots,\boldsymbol{\alpha}^{i-1},\boldsymbol{\alpha}^{i+1},\ldots,\boldsymbol{\alpha}^n\right]$ fixed:

$$\min_{\boldsymbol{\alpha}^i\in\mathbf{R}^n} F(\boldsymbol{\alpha}^i) = \frac{1}{2}\boldsymbol{\alpha}^{iT}\mathbf{P}_i\boldsymbol{\alpha}^i + \mathbf{b}_i^T\boldsymbol{\alpha}^i + \lambda\|\boldsymbol{\alpha}^i\|_1 \quad (13)$$
$$s.t.\ \boldsymbol{\alpha}_{ii} = 0$$

where $\mathbf{P}_i = 2\boldsymbol{X}^T\boldsymbol{X} + \big(\gamma(\sum_{j\neq i}\mathbf{A}_{ij}|\mathbf{W}_{ij}| + \mathbf{A}_{ji}|\mathbf{W}_{ji}|) + \beta\big)\mathbf{I}_n$ which is a positive definite matrix, $\mathbf{b}_i = -2\boldsymbol{X}^T\mathbf{x}_i - \gamma\sum_{j\neq i}(\mathbf{A}_{ij}|\mathbf{W}_{ij}| + \mathbf{A}_{ji}|\mathbf{W}_{ji}|)\boldsymbol{\alpha}_j - \mathbf{K}^i$, $\mathbf{K}^i$ is the i-th column of $\mathbf{K}$. Problem (13) is a Lasso problem and it is also solved efficiently by ADMM, where the

resultant subproblems have closed-form solutions. We leave the details in the supplementary document of this paper. The optimal solution to (12) is obtained by iteratively solving (13) for $i = 1 \ldots n$ until convergence. We adopt a warm start technique that effectively reduces the iteration number of coordinate descent. Warm start initializes $\boldsymbol{\alpha}^{(k)}$ in the current iteration by the solution $\boldsymbol{\alpha}^{(k-1)}$ obtained in the previous iteration. In our experiments we observe that the iteration number of coordinate descent is less than 5 in most cases.

Algorithm 1 Data Clustering by Regularized ℓ^1-Graph (Rℓ^1-Graph)

Input:

The data set $\boldsymbol{X} = \{\mathbf{x}_i\}_{i=1}^n$, the number of clusters c, the parameters λ, the regularization parameter γ, the ADMM penalty parameters β and μ, the threshold ε_1, ε_2 and the maximum iteration number M.

$k = 0$, initialize the coefficient matrix, the matrix $\mathbf{W}$ and the Lagrangian multiplier as $\mathbf{0}$, i.e. $\boldsymbol{\alpha}^{(0)} = \mathbf{W}^{(0)} = \mathbf{Y}^{(0)} = \mathbf{0}$.
Begin the ADMM iterations:
while $k \leq M$ **do**
  Solve subproblems (9), (10) and (11) according to the details explained in Section 3 to obtain $\boldsymbol{\alpha}^{(k+1)}$, $\mathbf{W}^{(k+1)}$ and $\mathbf{Y}^{(k+1)}$.
  if $k \geq 1$ and ($\|\boldsymbol{\alpha}^{(k)} - \boldsymbol{\alpha}^{(k-1)}\|_F < \varepsilon_1$ and $\|\mathbf{W}^{(k)} - \boldsymbol{\alpha}^{(k)}\|_F < \varepsilon_2$) **then**
    print
  else
    $k = k + 1$.
  end if
end while
Obtain the optimal coefficient matrix $\boldsymbol{\alpha}^*$ when the ADMM algorithm converges or maximum iteration number is achieved.
Build the pairwise similarity matrix by symmetrizing $\boldsymbol{\alpha}^*$: $\mathbf{W}^* = \frac{|\boldsymbol{\alpha}^*| + |\boldsymbol{\alpha}^*|^T}{2}$, compute the corresponding normalized graph Laplacian $\mathbf{L}^* = (\mathbf{D}^*)^{-\frac{1}{2}}(\mathbf{D}^* - \mathbf{W}^*)(\mathbf{D}^*)^{-\frac{1}{2}}$, where $\mathbf{D}^*$ is a diagonal matrix with $\mathbf{D}^*_{ii} = \sum_{j=1}^{n} \mathbf{W}^*_{ij}$
Construct the matrix $\mathbf{v} = [\mathbf{v}_1, \ldots, \mathbf{v}_c] \in \mathbb{R}^{n \times c}$, where $\{\mathbf{v}_1, \ldots, \mathbf{v}_c\}$ are the c eigenvectors of $\mathbf{L}^*$ corresponding to its c smallest eigenvalues. Treat each row of $\mathbf{v}$ as a data point in $\mathbb{R}^c$, and run K-means clustering method to obtain the cluster labels for all the rows of $\mathbf{v}$.

Output: The cluster label of $\mathbf{x}_i$ is set as the cluster label of the i-th row of $\mathbf{v}$, $1 \leq i \leq n$.

- Subproblem (10): update $\mathbf{W}$ while fixing $\boldsymbol{\alpha}$ and $\mathbf{Y}$

$$\min_{\mathbf{W}} \gamma \mathrm{Tr}(\boldsymbol{\alpha} \mathbf{L}_{\mathbf{A} \circ |\mathbf{W}|} \boldsymbol{\alpha}^T) + \langle \mathbf{Y}, \mathbf{W} \rangle - \beta \langle \mathbf{W}, \boldsymbol{\alpha} \rangle + \frac{\beta}{2} \langle \mathbf{W}, \mathbf{W} \rangle$$

which is equivalent to

$$\min_{\mathbf{W}} \sum_{i,j=1}^{n} \frac{\beta}{2} (\mathbf{W}_{ij} - \frac{\beta \boldsymbol{\alpha}_{ij} - \mathbf{Y}_{ij}}{\beta})^2 + \gamma \mathbf{Q}^{\boldsymbol{\alpha}}_{ij} |\mathbf{W}_{ij}| \tag{14}$$

(14) can be solved for each $\mathbf{W}_{ij}$ separately by soft thresholding as below:

$$\mathbf{W}_{ij} = \max\{0, |\beta\boldsymbol{\alpha}_{ij} - \mathbf{Y}_{ij}| - \gamma\mathbf{Q}^{\boldsymbol{\alpha}}_{ij}\} \cdot \frac{\text{sign}(\beta\boldsymbol{\alpha}_{ij} - \mathbf{Y}_{ij})}{\beta} \quad 1 \le i, j \le n \tag{15}$$

where $\mathbf{Q}^{\boldsymbol{\alpha}}$ is a $n \times n$ matrix with elements $\mathbf{Q}^{\boldsymbol{\alpha}}_{ij} = \frac{1}{2}\mathbf{A}_{ij}\|\boldsymbol{\alpha}^i - \boldsymbol{\alpha}^j\|_2^2$, and the sign function $\text{sign}(\cdot)$ is defined as

$$\text{sign}(x) = \begin{cases} 1 & : \quad x > 0 \\ 0 & : \quad x = 0 \\ -1 & : \quad x < 0 \end{cases} \tag{16}$$

Given the initialization $\boldsymbol{\alpha}^{(0)} = \mathbf{W}^{(0)} = \mathbf{Y}^{(0)} = \mathbf{0}$, the ADMM algorithm solves three subproblems (9), (10) and (11) iteratively until convergence or the maximum iteration number is achieved. With the obtained optimal coefficient matrix α^*, we build a pairwise similarity matrix $\mathbf{W}^* = \frac{|\boldsymbol{\alpha}| + |\boldsymbol{\alpha}^T|}{2}$ and then use spectral clustering method to obtain the clustering result, as suggested in ℓ^1-Graph [6]. Algorithm 1 describes our data clustering algorithm using Rℓ^1-Graph in detail.

Suppose the maximum iteration number of ADMM is N_1, and the maximum iteration number of the coordinate descent for solving subproblem (12) is N_2, then the overall time complexity for solving the optimization Problem (7) by ADMM is $\mathcal{O}(N_1 n^{2.376} + N_1 N_2 n^2)$. We leave the details in the supplementary document. It is known that ADMM converges and achieves globally optimal solution for a class of convex problems [5]. Although our optimization problem (7) is nonconvex, we observe that ADMM for (7) always converges in less than 15 iterations for all the experiments we conduct.

4 Experimental Results

We demonstrate the performance of Rℓ^1-Graph with comparison to other competing methods in the section.

4.1 Data Set

We conduct clustering experiments on various real data sets, which are summarized in Table 1. Three data sets are image data sets, i.e the ORL face database, the Yale face database and the MNIST handwritten digits data set. The ORL face database contains facial images for 40 subjects, and each subject has 10 images. The images are taken at different times with varying lighting and facial expressions. The subjects are all in an upright, frontal position with a dark homogeneous background. The Yale face database contains 165 grayscale images of 15 individuals. The MNIST database of handwritten digits has a total number of 70000 samples ranging from 0 to 9. The digits are normalized and centered in a fixed-size image. We also choose four data sets from UCI machine learning repository [1], i.e. Heart, Breast Tissue (BT) and Breast Cancer (Breast).

4.2 Evaluation Metric

We use two measures to evaluate the performance of the clustering methods, i.e. the accuracy and the Normalized Mutual Information(NMI) [21]. Suppose the predicted label of the

Table 1: Real data sets used in experiments

	ORL	Yale	MNIST	Heart	BT	Breast
# of instances	400	168	70000	270	106	569
Dimension	1024	1024	784	13	9	30
# of classes	40	15	10	2	6	2

datum $\mathbf{x}_i$ is $\hat{y}_i$ which is produced by the clustering method, and y_i is its ground truth label. The accuracy is defined as

$$Accuracy = \frac{\mathbb{I}_{\Phi(\hat{y}_i)\neq y_i}}{n} \tag{17}$$

where $\mathbb{I}$ is the indicator function, the mapping function Φ is the best permutation mapping function obtained by the Kuhn-Munkres algorithm [15]. Based on (17), we can see that the more predicted labels match the ground truth ones, the more accuracy value is obtained.

On the other hand, suppose the clusters obtained from the predicted labels $\{\hat{y}_i\}_{i=1}^n$ and the ground truth labels $\{y_i\}_{i=1}^n$ are $\hat{C}$ and C respectively. The mutual information between $\hat{C}$ and C is

$$MI(\hat{C},C) = \sum_{\hat{c}\in\hat{C},c\in C} p(\hat{c},c)\log_2\left(\frac{p(\hat{c},c)}{p(\hat{c})\,p(c)}\right) \tag{18}$$

where $p(\hat{c})$ and $p(c)$ are the probabilities that a data point belongs to the clusters $\hat{c}$ and c respectively, and $p(\hat{c},c)$ is the probability that a data point jointly belongs to clusters $\hat{c}$ and c. The normalized mutual information(NMI) is defined as follows:

$$NMI(\hat{C},C) = \frac{MI(\hat{C},C)}{\max\{H(\hat{C}),H(C)\}} \tag{19}$$

where $H(\hat{C})$ and $H(C)$ is the entropy of $\hat{C}$ and C. It can be verified that the normalized mutual information takes values in $[0,1]$. The accuracy and the normalized mutual information has been widely used for evaluate the performance of the clustering methods [6, 20, 21].

4.3 Clustering Result

We compare our algorithm to K-means (KM), Spectral Clustering (SC) and ℓ^1-Graph. Moreover, in order to demonstrate the superiority of our proposed regularization term (4) using the pairwise similarity matrix constructed by the sparse code instead of the Gaussian kernel, we derive Laplacian regularized ℓ^1-Graph (Lℓ^1-Graph). Lℓ^1-Graph is equivalent to the subproblem (12) of Rℓ^1-Graph except that $\mathbf{W}$ is set by the Gaussian kernel. For MNIST data set, we randomly select 6 digits out of the 10 digits, and then randomly choose 100 samples for each chosen digit, resulting in a subset comprsing 600 samples. We perform this process for 20 times and report the average clustering performance on the 20 runs. The clustering results on various data sets are shown in Table 2. By the manifold assumption that imposes local smoothness of the sparse codes in the data submanifold, Rℓ^1-Graph obtains better performance than ℓ^1-Graph and SMCE. Moreover, the regularization term using the sparse codes achieves better performance than that using Gaussian kernel.

Table 2: Clustering Results on Real Data Sets

Data Set	Measure	KM	SC	ℓ^1-Graph	Lℓ^1-Graph	Rℓ^1-Graph
ORL	AC	0.5333	0.4385	0.6964	0.6925	**0.7489**
	NMI	0.7317	0.6604	0.8410	0.8367	**0.8731**
Yale	AC	0.3974	0.2093	0.5339	0.5307	**0.5673**
	NMI	0.4525	0.2067	0.5731	0.5731	**0.5906**
MNIST	AC	0.6276	0.4422	0.6419	0.6425	**0.6617**
	NMI	0.5243	0.3358	0.6207	0.6156	**0.6288**
Heart	AC	0.5889	0.5519	0.6370	0.6333	**0.6407**
	NMI	0.0182	0.0032	0.0534	0.0507	**0.0573**
BT	AC	0.3396	0.4057	0.4434	0.4123	**0.5094**
	NMI	0.3265	**0.3563**	0.2762	0.2658	0.3608
Breast	AC	0.8541	0.6292	0.9016	**0.9051**	**0.9051**
	NMI	0.4223	0.0026	0.5172	**0.5249**	**0.5249**

4.4 Parameter Setting

We set $\lambda = 0.1$, $\gamma = 0.5$, and choose $K \in \{5, 15\}$ empirically throughout all the experiments in this paper. There are two parameters that influence the regularization term in Rℓ^1-Graph, namely the weight of the regularization γ and the number of nearest neighbors K of the KNN adjacency matrix. The regularization term imposes stronger smoothness constraint on the sparse codes with larger γ and K, and vice versa. We investigate how the clustering performance on ORL face database changes when varying these two parameters, and illustrate the result in Figure 2 and Figure 3 respectively. We observe that the performance of Rℓ^1-Graph is much better than other algorithms over a large range of both γ and K, revealing the robustness of our algorithm.

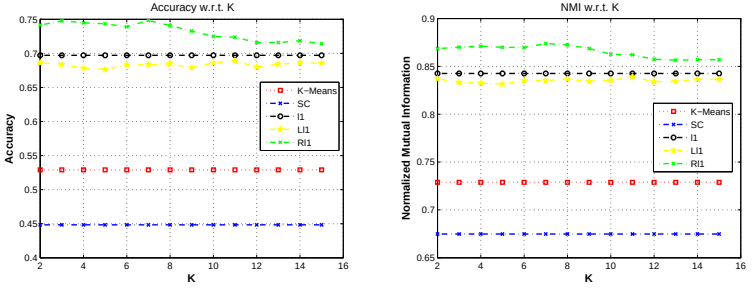

Figure 2: Clustering performance with different values of K, i.e. the number of nearest neighbors, on ORL face database when $\gamma = 0.5$. Left: Accuracy; Right: NMI

5 Conclusion

We propose Regularized ℓ^1-Graph for data clustering in this paper. Complying to the manifold assumption, Rℓ^1-Graph encourages the sparse representations of the data to vary smoothly along the geodesics of the intrinsic data submanifold using the graph Laplacian constructed by the sparse codes. Rℓ^1-Graph achieves better performance than Laplacian Regularized ℓ^1-

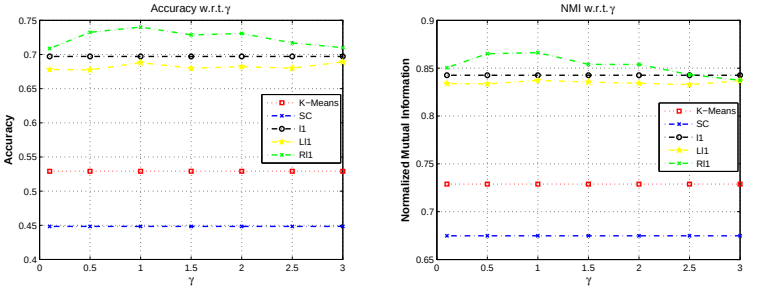

Figure 3: Clustering performance with different values of γ, i.e. the weight of the regularization term, on ORL face database when $K = 5$. Left: Accuracy; Right: NMI

Graph where the graph Laplacian is constructed by Gaussian kernel. Experimental results on real data sets shows the effectiveness of our algorithm.

Acknowledgements

This research is supported in part by ONR Grant N00014-12-1-0122; and in part by National Science Foundation Grant DBI 10-62351.

References

[1] D.J. Newman A. Asuncion. UCI machine learning repository, 2007.

[2] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[3] Dimitri P. Bertsekas and Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 2nd edition, September 1999.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[5] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011. ISSN 1935-8237.

[6] Bin Cheng, Jianchao Yang, Shuicheng Yan, Yun Fu, and Thomas S. Huang. Learning with l1-graph for image analysis. *IEEE Transactions on Image Processing*, 19(4):858–866, 2010.

[7] F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

[8] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000.

[9] Chris Fraley and Adrian E. Raftery. Model-Based Clustering, Discriminant Analysis, and Density Estimation. *Journal of the American Statistical Association*, 97(458):611–631, June 2002. ISSN 0162-1459.

[10] Shenghua Gao, Ivor Wai-Hung Tsang, and Liang-Tien Chia. Laplacian sparse coding, hypergraph laplacian sparse coding, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):92–104, 2013.

[11] Xiaofei He, Deng Cai, Yuanlong Shao, Hujun Bao, and Jiawei Han. Laplacian regularized gaussian mixture model for data clustering. *Knowledge and Data Engineering, IEEE Transactions on*, 23(9):1406–1418, Sept 2011. ISSN 1041-4347.

[12] Xiaofei He, Deng Cai, Yuanlong Shao, Hujun Bao, and Jiawei Han. Laplacian regularized gaussian mixture model for data clustering. *Knowledge and Data Engineering, IEEE Transactions on*, 23(9):1406–1418, Sept 2011. ISSN 1041-4347.

[13] Jialu Liu, Deng Cai, and Xiaofei He. Gaussian mixture model with local consistency. In *AAAI*, 2010.

[14] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, pages 849–856, 2001.

[15] D. Plummer and L. Lovász. *Matching Theory*. North-Holland Mathematics Studies. Elsevier Science, 1986.

[16] Daniel Povey, Lukas Burget, Mohit Agarwal, Pinar Akyazi, Kai Feng, Arnab Ghoshal, Ondrej Glembek, Nagendra K. Goel, Martin Karafiát, Ariya Rastrow, Richard C. Rose, Petr Schwarz, and Samuel Thomas. Subspace gaussian mixture models for speech recognition. In *ICASSP*, pages 4330–4333, 2010.

[17] Shuicheng Yan and Huan Wang. Semi-supervised learning by sparse representation. In *SDM*, pages 792–801, 2009.

[18] Shuicheng Yan, Xi Zhou, Ming Liu, M. Hasegawa-Johnson, and T.S. Huang. Regression from patch-kernel. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008.

[19] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas S. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pages 1794–1801, 2009.

[20] Miao Zheng, Jiajun Bu, Chun Chen, Can Wang, Lijun Zhang, Guang Qiu, and Deng Cai. Graph regularized sparse coding for image representation. *IEEE Transactions on Image Processing*, 20(5):1327–1336, 2011.

[21] Xin Zheng, Deng Cai, Xiaofei He, Wei-Ying Ma, and Xueyin Lin. Locality preserving clustering for image database. In *Proceedings of the 12th Annual ACM International Conference on Multimedia*, MULTIMEDIA '04, pages 885–891, New York, NY, USA, 2004. ACM.

[22] Xi Zhou, Na Cui, Zhen Li, Feng Liang, and Thomas S. Huang. Hierarchical gaussianization for image classification. In *ICCV*, pages 1971–1977, 2009.