Sparse Subspace Clustering via Group Sparse Coding

Budhaditya Saha Duc Son Pham Dinh Phung Svetha Venkatesh

Abstract

Sparse subspace representation is an emerging and powerful approach for clustering of data, whose generative model is a union of subspaces. Existing sparse subspace representation methods are restricted to the single-task setting, which consequently leads to inefficient computation and sub-optimal performance. To address the current limitation, we propose a novel method that regularizes sparse subspace representation by exploiting the structural sharing between tasks and data points. The first regularizer aims at group level where we seek sparsity between groups but dense within group. The second regularizer models the interactions down to data point level via the well-known graph regularization technique. We also derive simple, provably convergent, and extremely computationally efficient algorithms for solving the proposed group formulations. We evaluate the proposed methods over a wide range of large-scale clustering problems: from challenging health care to image and text clustering benchmarks datasets and show that they outperform state-of-the-art considerably.

1 Introduction

Subspace clustering refers to the problem in which the data is modeled by a union of subspaces [21]. Such a model is a natural extension of the single subspace approach, which was used widely in previous works [3, 4]. Its applications range from image processing, compression [16, 31], to motion segmentation problems that infer the structures and movements of 3D objects from tracked points in video [15, 29, 6, 19, 25, 9]. This problem has a close connection to the well-known compressed sensing (CS) theory [5, 7], and thus its theoretical treatments have been recently made in [8, 22, 23].The previous approaches to subspace clustering include mixture of Gaussian[27], factorization [6, 19], and algebraic [28, 29], with known algorithms such as k-subspaces [15], mixture of probabilistic principal component analysis (MPPCA) [27], multi-stage learning [12], and RANSAC [10]. These methods have several major limitations, including prior knowledge of the number of subspaces and their dimensions, expensive computation exponential with the number of subspaces and the dimensions [26], and lack of robustness with respect to noise, outliers, and modeling errors [9].

Inspired by the success of compressed sensing [5, 7], a recent approach to subspace clustering using sparse [18] or low rank [21] representation as the general guiding principle has received tremendous attention. The essence of this approach is to represent each data point as a sparse linear combination of other data point via a convex formulation with ℓ_1 regularization. Such a convex problem is easily solved with an increasing availability of computationally efficient ℓ_1-regularization algorithms whose complexity is at most polynomial. Under sparse subspace assumptions, such a representation would allow one to automatically discover the number of subspaces and their dimensions. In computer vision, Elhamifar and Vidal first introduced sparse subspace clustering (SSC) to solve the motion segmentation problem [9]. SSC can discover the number of subspaces and their dimensions automatically by analyzing an affinity graph constructed out of the sparse representation for all data points. The low-rank representation method (LRR) [21], which seeks sparsity of singular values in the representation, uses the trace-norm instead to compute the representation and claims advantage over SSC. However, extension of SSC with weighted formulation [24] shows that SSC can be significantly improved by exploiting geometric relations between data points as constraints.

Nevertheless, the above SSC variants are still restricted to a single-task framework wherein the representation for each data point is found independently. Whilst this simplifies the formulation, it lacks optimality due to the existence of inherent joint structure between the representations of data points. As all data points use the same 'dictionary', some of the words in that dictionary might be outstanding and most point would refer to. This indicates that the sparse coefficients of these exemplar data points are likely to be seen significant *across* different representations for many data points. But on the other hand, the representation for each data point still needs to respect the sparse subspace assumption. Such a desirable goal is best captured by a concept known as *joint sparse* model in compressed sensing [1] or group sparse in statistics [17] and has been proved to improve the basic sparsity modeling. We note that though LRR is formulated for all data points at a time using the low-rank principle, there is no actual explicit group concept and thus the definition of tasks is vague and exploiting shared structure is impossible. Thus, extending SSC modeling to incorporate the joint sparsity concept is desirable.

We propose in this work a novel extension of SSC that aims at exploiting the shared structure between individual clustering tasks. Our first contribution is the new formu-

lation that capture joint sparsity via a ℓ_2/ℓ_1 regularization on the coefficient matrix when we solve all the tasks simultaneously. Such a regularization has been shown effective in multi-task learning, also known as group Lasso in statistics [30, 32], where it couples the individual tasks via group structure of constraint terms on the assumption that multiple tasks share a *common* sparsity structure. Though the new formulation can be transformed to a group Lasso-type format and can be naively solved with vector-based group Lasso algorithms, such an approach is extremely inefficient in large-scale clustering as the group Lasso matrix is extremely large. Here, we derive a specialized matrix-based algorithm that is extremely efficient and fast to solve the new formulation. Furthermore, we also move beyond joint sparsity modeling by extending the formulation to cater for interactions down to data point level. This is motivated from [24] which shows that the geometry between data point can be useful side information. Thus, our second contribution is the inclusion of a graph regularization that allows detail modeling of data points. We propose several choices for the construction of the graph Laplacian, including proximity of points in a k-neighborhood radial basis function (RBF), cosine, and 0/1 matrix. From the theory of elastic-net in statistics [32], the inclusion of the second-order to the existing first-order regularization will further improve stability, especially when dealing with realistic data. Our proposal can also be viewed as a much more general multi-task (matrix) version of the single-task (vector) elastic-net.

We apply the model across three real-world datasets comprising of: 1) a cohort of 1580 diabetes patients with 551 disease codes; 2) NUS-WIDE image dataset with 3411 animal images; and 3) 20-Newsgroup text dataset with 8000 documents. The diabetes data is collected over a period of 5 years, over which each code is assigned upon a hospital visit of a patient. Evaluation of such algorithms on real-world data, such as the health data is notoriously hard and thus we propose to use a special ρ-measure method. This allows ground-truth to be allocated based on degree of similarity between two points. Using this measure, we can compute the Rand-index and F-measure for a given ρ. We show how the same method can be adapted if groundtruth is present, as in the case of NUS wide and Reuters data. We show that our methods outperform the previous SSC variants and many competitive clustering methods, such as affinity propagation (AP) [11], locality-preserving projection (LPP)[14], k-means [20]. We also show tag clouds for clusters in the diabetes cohort and demonstrate how the subgroups discovered are qualitatively meaningful.

2 Proposed Framework

Assume that the data is collected in a matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$. In sparse coding, we seek to represent each data point $\mathbf{x}_i$ as a sparse linear combination of all data points. Denote as $\mathcal{S}_i$ the index set of the subspace (cluster) that $\mathbf{x}_i$ belongs to, then we can write the linear representation as

$$\mathbf{x}_i = \mathbf{X}\mathbf{c}_i = \sum_{j \neq i} c_{ij}\mathbf{x}_i = \sum_{i \in \mathcal{S}_i, j \neq i} c_{ij}\mathbf{x}_i + \sum_{j \notin \mathcal{S}_i} c_{ij}\mathbf{x}_i.$$

In ideal scenarios, the coefficients in the second summation of the right term would be zero and thus giving rise to sparse coefficient vector $\mathbf{c}_i$. According to CS theory, such a sparse coefficient vector $\mathbf{c}_i$ can be found by minimizing its ℓ_1-norm, and thus the original SSC formulation solves the following noisy formulation independently for all data points $i = 1, \ldots, N$:

$$\min_{\mathbf{c}_i} \ \frac{1}{2}||\mathbf{x}_i - \mathbf{X}\mathbf{c}_i||_2^2 + \lambda||\mathbf{c}_i||_1, c_{ii} = 0 \tag{2.1}$$

We now extend this to a multi-task setting. Denote as $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_N]$ the coefficient matrix, then the solutions of *all* SSC's individual tasks can be conveniently written in a matrix form as

$$\min_{\mathbf{C}} \frac{1}{2}\|\mathbf{X} - \mathbf{X}\mathbf{C}\|_F^2 + \lambda\|\mathbf{C}\|_1, \ \ , C_{ii} = 0.$$

Here, $\|\mathbf{C}\|_1 = \sum_{i=1}^N \|\mathbf{c}_i\|_1 = \sum_{i,j} |C_{ij}|$, and intutively this formulation seeks $\mathbf{C}$ such that $\mathbf{X} \approx \mathbf{X}\mathbf{C}$ and that $\mathbf{C}$ is sparse. However, note that the tasks are still independent.

Next, we seek to exploit the shared structure between the tasks. To do so, we introduce two regularizers to control the coefficient matrix $\mathbf{C}$.

2.1 Block ℓ_2/ℓ_1 Regularizer As indicated in Fig. 1, each column of $\mathbf{C}$ denotes a task, and thus each row of $\mathbf{C}$ indicates the common coefficient position between all tasks and corresponds to a word in the dictionary, which is the data matrix itself in this case. Based on numerical experience, our hypothesis is that exemplar words in the dictionary are likely to be selected for many representations, and thus the corresponding rows are likely dense. Note that all columns of $\mathbf{C}$ must still be sparse to respect the sparse modeling, which is fundamental for subspace clustering.

Such a desired property of $\mathbf{C}$ is best captured by the use of the block regularization norm: $\|\mathbf{C}\|_{\ell_2/\ell_1} = \sum_{i=1}^N \|\mathbf{r}_i\|_2$, where $\mathbf{r}_i^T$'s are the row vectors of $\mathbf{C}$. Such a block regularization has been shown effective in multi-task learning, also known as group Lasso in statistics [30, 32], where it couples the individual tasks via group structure of constraint terms on the assumption that multiple tasks share a common sparsity structure.

Inspired by the weighting scheme in [24], we also introduce additional weights between the shared structures, i.e. in between the rows of $\mathbf{C}$ to aid in the recovery of correct sparse pattern. Denote as $\mathbf{w} = [w_1, \ldots, w_N]$ the weight

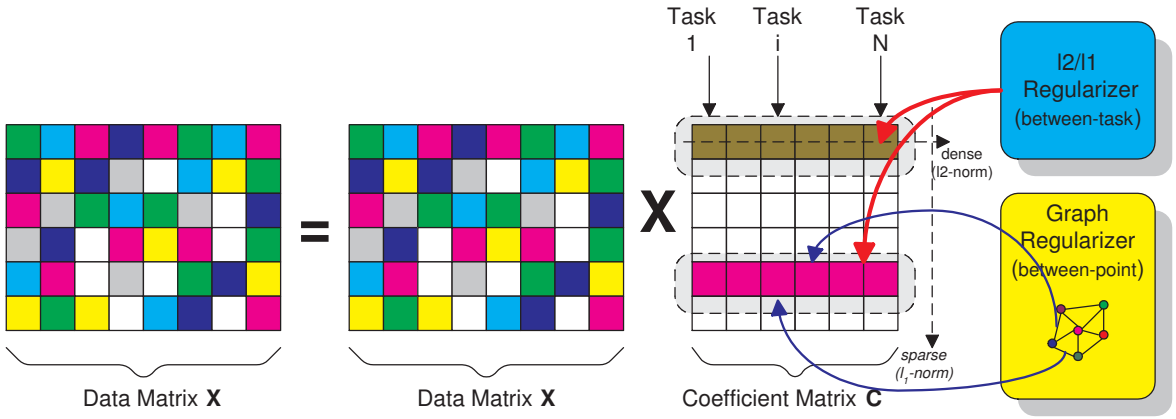


Figure 1: Proposed method

vector, then the weighted block regularizer is

$$R_{\text{shared}}(\mathbf{C}) = \lambda \|\mathbf{w} \odot \mathbf{C}^T\|_{\ell_2/\ell_1} = \lambda \sum_{i=1}^{N} \|w_i \mathbf{r}_i\|_2.$$

Here, $\odot$ denotes the elementwise product.

2.2 Graph Regularizer The above block regularizer effective seeks sparsity at group level and promotes dense within the group. However, it cannot precisely model deep to the coefficients of $\mathbf{C}$. In many cases, such modeling can be potentially exploited to improve stability and consistency. Recall that each coefficient C_{ij} denotes the interaction between data point $\mathbf{x}_i$ and $\mathbf{x}_j$ and thus if $\mathbf{x}_j$ is 'similar' to $\mathbf{x}_i$ then it is likely to be picked in the linear representation of $\mathbf{x}_i$. To achieve this desirable goal, we propose to use the powerful graph regularization in machine learning.

Consider a graph $\mathcal{G}$ on $\mathbf{X}$ with N nodes corresponding to N data points in $\mathbf{X}$. The edges of the graph encode the similarities between the nodes, and is best captured in a similarity matrix $\mathbf{K}$. This matrix can be best obtained if there is side information. If such side information is not available, we propose to exploit within the data geometry itself. We consider the construction of the graph based on the locality of the data geometry. Denotes as $\mathcal{N}(\mathbf{x}_j)$ the nearest neighbor of $\mathbf{x}_j$ (or node j) and as $\mathbb{I}$ the indicator (0/1) function. We consider three choices for $\mathbf{K}$:

- **RBF** : $K_{ij} = \mathbb{I}_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \times \exp^{\frac{||\mathbf{x}_i - \mathbf{x}_j||_2^2}{2\sigma^2}}$
- **0-1** : $K_{ij} = \mathbb{I}_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)}$
- **Cosine**: $K_{ij} = \mathbb{I}_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x}_j)} \times \frac{\langle \mathbf{x}_i, \mathbf{x}_j \rangle}{||\mathbf{x}_i||_2 ||\mathbf{x}_j||_2}$

To effectively regularize $\mathbf{C}$, the weights for inter-cluster coefficients must be large whilst intra-cluster coefficients must be small. In graph regularization theory, such a goal is capture in the Laplacian matrix of the graph, denoted as $\mathbf{L} = \mathbf{D} - \mathbf{K}$, where $\mathbf{D}$ is a diagonal matrix with diagonal entries $D_{ii} = \sum_j K_{ij}$. The graph regularization term is then a quadratic function

$$R_{\text{graph}} = \frac{\mu}{2} \text{tr}(\mathbf{C}^T \mathbf{L} \mathbf{C}).$$

2.3 Proposed Formulation and Algorithms Thus, combining the regularization terms, we propose a new multi-task formulation, which is termed graph regularized group subspace clustering (GR-SSC).

$$\min_{\mathbf{C}} \frac{1}{2} \|\mathbf{X} - \mathbf{X}\mathbf{C}\|_F^2 + \lambda \|\mathbf{w} \odot \mathbf{C}^T\|_{\ell_2/\ell_1} + \frac{\mu}{2} \text{tr}(\mathbf{C}^T \mathbf{L} \mathbf{C}).$$

Remarks:

- Unlike the single-task case, we do not need to enforce $C_{ii} = 0$. This is due to the effectiveness of the block and graph regularizer. In principle, one can also impose such constraints, but the benefit is almost negligible at the cost of making the maths more complex.
- For $\mathbf{w} = \mathbf{1}$ and $\lambda = 0$, the above formulation leads to an ordinary ridge-regression type of problem, where the solution is exact and generally dense.
- For $\mu = 0$ and $\mathbf{w} \propto \mathbf{1}$, the above formulation leads to a group sparse version of SSC, which we call it group SSC (G-SSC), which is also a matrix version of group Lasso. This can be realized by vectorizing the matrix variable $\mathbf{C}$ and the quadratic loss term can be shown to be $(1/2)\|(\mathbf{X} \otimes \mathbf{I})\text{vec}(\mathbf{C}) - \text{vec}(\mathbf{X})\|_2^2$, which is the familiar vector form of group Lasso. Here, $\otimes$ denotes the Kronecker product, and vec denotes the vectorization operator. However, converting to vector form and using existing solvers for group Lasso would be extremely efficient, due to the matrix $(\mathbf{X} \otimes \mathbf{I})$ being extremely large.

- As there are both first-order and second-order regularizers in the formulation, our proposed method can be mathematically viewed as a matrix (multi-task) version of the well-known elastic-net [32] in the statistics literature, if one generalizes each vector entry in elastic-net to a row vector. However, the interpretation of our proposed method is far more generalized than elastic-net and treats it as a very extremely special case. Due to the general form of the Laplacian in the regularization term, it is not possible to convert the proposed formulation to a group Lasso form as in the way elastic-net can be converted to Lasso.

Next, we derive a computationally efficient algorithm to solve the proposed formulation using a powerful theory in convex optimization, known as alternative directions method of multipliers (ADMM). It is a framework that can effectively solve complex regularization problems by decoupling the complex regularization constraints from the main loss function, which is typically not group-wise or element-wise decomposable. For a background on ADMM, please see [2].

Under the ADMM framework, we introduce an additional variable $\mathbf{Z}$, derivable from $\mathbf{C}$ through constraints $\mathbf{C}-\mathbf{Z}=0$. We then consider the Lagrangian

$$\begin{aligned}\mathcal{L}(\mathbf{C},\mathbf{Z},\mathbf{Y}) &= \frac{1}{2}\|\mathbf{X}\mathbf{C}-\mathbf{X}\|_F^2+\lambda\|\mathbf{w}\odot\mathbf{Z}^T\|_{\ell_2/\ell_1}\\ &\quad+\langle\mathbf{\Lambda},\mathbf{C}-\mathbf{Z}\rangle+\frac{\rho}{2}\|\mathbf{C}-\mathbf{Z}\|_F^2\\ &\quad+\frac{\mu}{2}\mathsf{tr}(\mathbf{C}\mathbf{L}\mathbf{C}^T). \qquad (2.2)\end{aligned}$$

Here, $\mathbf{\Lambda}$ is the dual variable corresponding to the equality constraint $\mathbf{C}-\mathbf{Z}=\mathbf{0}$, $\langle\mathbf{C},\mathbf{\Lambda}\rangle$ denotes inner product, i.e., $\langle\mathbf{C},\mathbf{\Lambda}\rangle=\mathsf{tr}(\mathbf{C}^T\mathbf{\Lambda})=\sum_{ij}C_{ij}Y_{ij}$, and ρ is a parameter to improve the numerical stability of the algorithm.

We note that if we are only interested in only one variable $\mathbf{C}$ or $\mathbf{Z}$ then we can always write

$$\langle\mathbf{\Lambda},\mathbf{C}-\mathbf{Z}\rangle+\frac{\rho}{2}\|\mathbf{C}-\mathbf{Z}\|_F^2=\frac{\rho}{2}\|\mathbf{C}+\frac{\mathbf{\Lambda}}{\rho}-\mathbf{Z}\|_F^2+\text{const.}$$

This suggests that we can normalize the dual variable $\mathbf{U}=\mathbf{\Lambda}/\rho$. Then, under the ADMM framework, we solve the problem by using the following update [2, p.15]:

$$\begin{aligned}\mathbf{C}^{k+1} &= \arg\min_{\mathbf{C}}\frac{1}{2}\|\mathbf{X}\mathbf{C}-\mathbf{X}\|_F^2\\ &\quad+\frac{\rho}{2}\|\mathbf{C}+\mathbf{U}^k-\mathbf{Z}^k\|_F^2+\frac{\mu}{2}\mathsf{tr}(\mathbf{C}\mathbf{L}\mathbf{C}^T) \qquad (2.3)\\ \mathbf{Z}^{k+1} &= \arg\min_{\mathbf{Z}}\lambda\|\mathbf{w}\odot\mathbf{Z}^T\|_{\ell_2/\ell_1}\\ &\quad+\frac{\rho}{2}\|\mathbf{C}^{k+1}+\mathbf{U}^k-\mathbf{Z}^k\|_F^2, \qquad (2.4)\\ \mathbf{U}^{k+1} &= \mathbf{U}^k+\mathbf{C}^{k+1}-\mathbf{Z}^{k+1}. \qquad (2.5)\end{aligned}$$

We next show that the update steps for $\mathbf{C}$ and $\mathbf{Z}$ are exact. Indeed, for the update step of $\mathbf{C}$, the objective function is convex, and thus making matrix derivative equal zero yields

$$\mathbf{X}^T\mathbf{X}\mathbf{C}-\mathbf{X}^T\mathbf{X}+\rho(\mathbf{C}+\mathbf{U}^k-\mathbf{Z}^k)+\mu\mathbf{L}\mathbf{C}=\mathbf{0}$$

Here, we have exploited the fact that $\mathbf{L}$ is symmetric. This yields the analytical solution

$$\begin{aligned}\mathbf{C}^{k+1}=(\mathbf{X}^T\mathbf{X}+\rho\mathbf{I}+\mu\mathbf{L})^{-1}(\mathbf{X}^T\mathbf{X}\\ +\rho(\mathbf{Z}^k-\mathbf{U}^k)) \qquad (2.6)\end{aligned}$$

We note that both $(\mathbf{X}^T\mathbf{X}+\rho\mathbf{I}+\mu\mathbf{L})^{-1}$ and $\mathbf{X}^T\mathbf{X}$ are independent of the iterations, and thus they can be computed and cached in the memory to improve computational efficiency.

Next, we derive the update step for $\mathbf{Z}$ by solving (2.4). To simplify the notation, denote as $\mathbf{V}=\mathbf{C}^{k+1}+\mathbf{U}^k$ then it is equivalent to solving

$$\mathbf{Z}^{k+1}=\min_{\mathbf{Z}}\lambda\|\mathbf{w}\odot\mathbf{Z}^T\|_{\ell_2/\ell_1}+\frac{\rho}{2}\|\mathbf{V}-\mathbf{Z}\|_F^2.$$

With a slightly abuse of notation, denote as $\mathbf{z}_i^T$'s and $\mathbf{v}_i^T$'s the row vectors of $\mathbf{Z}$ and $\mathbf{V}$ respectively. Then, it is easily recognized that the problem is row-wise decomposable as follows

$$\mathbf{Z}^{k+1} = \min_{\mathbf{Z}}\left\{\sum_i\lambda w_i\|\mathbf{z}_i\|_2+\frac{\rho}{2}\|\mathbf{v}_i-\mathbf{z}_i\|_2^2\right\}.$$

Thus, each row of $\mathbf{Z}^{k+1}$ is the solution of the following problem (we drop the subscript for generalization)

$$\mathbf{z}^{k+1} = \arg\min_{\mathbf{z}}\left\{\lambda w\|\mathbf{z}\|+\frac{\rho}{2}\|\mathbf{v}-\mathbf{z}\|_2^2\right\}. \qquad (2.7)$$

LEMMA 2.1. *Let* $\mathbf{e}=\frac{\mathbf{v}}{\|\mathbf{v}\|_2}$ *be the direction of* $\mathbf{v}$*, and let* $\kappa=\|\mathbf{v}\|_2$*. Then the solution of (2.7) has the form* $\mathbf{z}=\eta\mathbf{e}$ *where* $\eta\geq 0$ *is the minimizer of*

$$(\kappa-\eta)^2+(2\lambda w/\rho)\eta, \qquad (2.8)$$

which is known as the soft-thresholding shrinkage operator, i.e., $\eta=\max(\kappa-\lambda w/\rho,0)$.

The significance of this result is that it converts a multidimensional optimization problem (2.7) to an univariate optimization problem (2.8). This result can be proved by simple geometrical arguments. Indeed, denote $\mathbf{z}^*$ as the solution of (2.7), then we consider all points $\mathbf{z}$ such that $\|\mathbf{v}-\mathbf{z}\|_2=\|\mathbf{v}-\mathbf{z}^*\|=R$. It turns out that these points are lying on the ball with center at $\mathbf{v}$ and radius R. Among these points, only the point that satisfies $\mathbf{z}=\eta\mathbf{e}$, i.e. intersection of the ball and the vector $\mathbf{v}$, will have minimum ℓ_2 norm, which minimizes the second term in (2.7), then (2.8) follows immediately.

In summary, to solve the proposed formulation, we iterate through (2.6) for $\mathbf{C}$, then (2.7) for each row of $\mathbf{Z}$,

and finally (2.5) for $\mathbf{U}$. The initial values of $\mathbf{C}$ and $\mathbf{U}$ can be set to $\mathbf{0}$ and the stopping criterion is when the primal residual matrix $\mathbf{R}^k = \mathbf{C}^k - \mathbf{Z}^k$ and the dual residual matrix $\mathbf{S}^{k+1} = \rho(\mathbf{Z}^{k+1} - \mathbf{Z}^{k+1})$ are sufficiently small. Algorithm 1 presents the overall steps in proposed framework.

Final spectral clustering: Once the coefficient matrix $\mathbf{C}$ is obtained, the next step is to do final clustering. This step involves constructing a balanced affinity graph $\bar{\mathbf{C}}$ where $\bar{\mathbf{C}} = (\mathbf{C} + \mathbf{C}^T)/2$, followed by computing the Laplacian of $\bar{\mathbf{C}}$ as $\mathbf{L_C} = \mathbf{I} - \mathbf{D}^{-1/2}\bar{\mathbf{C}}\mathbf{D}^{-1/2}$ where $\mathbf{I}$ is an identity matrix of appropriate dimension. $\mathbf{D}$ is a diagonal matrix where $\mathbf{D}_{ii} = \sum_{j=1}^{N} \bar{c}_{ij}$ The smallest eigenvalues of $\mathbf{L_C}$ is used to estimate number of subspaces and the corresponding data points are obtained using k-means[20] algorithm. For detail see [24, 9].

Algorithm 1 Graph Regularized Group Sparse Subspace Clustering(GR-SSC)

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{D \times N}$ and ρ and ϵ.
Output: ID_K = K Cluster Index.
Initialize: $k = 0$, set $\mathbf{U}^k = \mathbf{0}$, $\mathbf{C}^k = \mathbf{0}$ and $\mathbf{Z}^k = \mathbf{0}$.

- Compute $\mathbf{B} = \mathbf{X}^T\mathbf{X}$.
- Compute $\mathbf{A} = (\mathbf{B} + \rho\mathbf{I} + \mu\mathbf{L})^{-1}$.

1. **Sparse Recovery:**

   Do until stopping criteria is met

   - Compute $\mathbf{C}$: $\mathbf{C}^{k+1}$= $\mathbf{A}(\mathbf{B} + \rho(\mathbf{Z}^k - \mathbf{U}^k))$.
   - Compute $\mathbf{V}$: $\mathbf{V} = \mathbf{C}^{k+1} + \mathbf{U}^k$.
   - Compute $\mathbf{Z}$: $\{\mathbf{z}_i = \eta_i \mathbf{e}_i\}\ \forall i \in [N]$.
   - Compute $\mathbf{U}$: $\mathbf{U}^{k+1} = \mathbf{U}^k + \mathbf{C}^{k+1} - \mathbf{Z}^{k+1}$.
   - Stopping criteria: $||\mathbf{R}^{k+1}||_F^2, \|\mathbf{S}^{k+1}\|_F^2 \leq \epsilon$.

   end

2. **Spectral Clustering:**

- Compute ID_Kfollowing ***Final spectral clustering***.

3 Experiments

3.1 Datasets We validate our approach on three real-world datasets which are diabetes data, image data in NUS-WIDE and collection of newsgroup documents in 20 Newsgroup dataset.

The diabetes data is collected from patients having diabetes recorded over a period of five years from 2007 to 2011[1] and has diagnosis records from 9878 patients.

Codes	Description of Codes
E1172	Type 2 diabetes mellitus
I10	Essential (primary) hypertension
Z9222	Long-term use history of other medicament, insulin
R63Z	Chemotherapy

Table 1: Examples of diagnostic codes

Each patient has been diagnosed several times over the period of five years and assigned unique diagnosis code(s). An example of a record for a patient over time might be (`E1172, I10, E1172, Z9222`). Table 1 shows the description of some diagnostic codes. Patients may be assigned similar code more than once over time. We remove records without codes, patients diagonalized less than thrice and also duplicate codes. This results in 1580 diabetes patients with 551 unique codes. We construct a code-patient matrix, where codes are used as features and each patient is an observation, analogous to word-document matrix for text data analysis.

NUS-WIDE[2] dataset is a large collection of Flicker images and we have selected a subset of 3411 images involving 13 animals[13] (see Figure 2). This dataset also provides 6 different low-level features, namely 64-D color histogram, 144-D color correalogram, 73-D edge direction histogram, 128-D wavelet texture, 225-D block wise color moments and 500-D SIFT descriptors along with their groundthruth. In our experiment we ignore SIFT descriptors and use only other low-level features. NUS-WIDE is most challenging for

The 20-Newsgroup dataset [3] contains 20,000 documents partitioned over 20 groups. We randomly extracted 8 different groups form 8 possible permutations of all possible sets and conducted 10 test on each subset. We construct a word-document matrix from randomly selected groups where rows correspond to the vocabulary in the corpus and columns correspond to the documents. The size of the dictionary is 30311 and number of documents is 8000.

3.2 Evaluation Metric As no ground-truth is available for latent groups, it is impossible to measure the clustering performance by standard evaluation metrics. Thus, we evaluate the performance using a novel ρ-measure method as follows:

1. Each data point $\mathbf{x}_i \in \mathbb{R}^N$ is mapped to a binary vector $\bar{\mathbf{x}}_i$ where $\bar{x}_{ij} = \mathbb{I}_{x_{ij} \neq 0}$.

[1] This dataset has been obtained from a large regional hospital in Australia. Ethics approval obtained through University and regional hospital – Number 12/82}.

[2] http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm

[3] http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html

Figure 2: NUS-WIDE Animal Datasets

2. Compute relative similarity metric $s(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j)$

$$s(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j) = \frac{\sum(\bar{\mathbf{x}}_i \odot \bar{\mathbf{x}}_j)}{\sum_{k=1}^{N} \bar{\mathbf{x}}_{ik} + \sum_{k=1}^{N} \bar{\mathbf{x}}_{jk} - \sum(\bar{\mathbf{x}}_i \odot \bar{\mathbf{x}}_j)}$$

3. Construct a ground-truth matrix $\mathbf{G}_\rho \in \mathbb{R}^{N \times N}$ with element $g_{ij} = \mathbb{I}_{s(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j) \geq \rho}$ Note that, if ground-truth is available and $\ell(i)$ denotes the true label of i^{th} observation, then we compute $g_{ij} = \mathbb{I}_{\ell(i)=\ell(j)}$.

4. Construct a cluster membership matrix $\mathbf{V}$ with element $v_{ij} = \mathbb{I}_{ID_K(i)=ID_K(j)}$.

Next, we compute *Precision* (P), *Recall* (R) and *F-measure* (F):

(3.9)
$$P = \frac{TP}{TP + FP}, \; R = \frac{TP}{TP + FN}, \; F = \frac{2 \times P \times R}{P + R}$$

Here, true positive (TP) is scored when two similar data points in the ground-truth are grouped together in the obtained results, a true negative (TN) is when two dissimilar data points are grouped separately, a false positive (FP) is when two dissimilar data points are grouped together and a false negative (FN) is when two similar data points are grouped separately. The rand index (RI) is defined as

$$RI = \frac{TP + TN}{TP + FP + FN + TN}$$

where high RI and F indicates the better accuracy.

3.3 Results and Comparison In pre-processing, the dimension of each data point is reduced by projecting it onto a lower dimensional subspace. The projection matrix is found by extracting the basis vectors corresponding to the principal singular values of the full data matrix. We compare our proposed framework against state-of-art sparse subspace clustering methods SSC [9], weighted SSC (W-SSC) [24], low-rank representation (LRR) [21] and also with benchmark methods like affinity propagation (AP) [11], LPP [14] and k-means [20]. Although the original LRR [21] only uses original dimension, we also examine LRR method on dimensionally reduced data (LRR-RD) as well as full dimensional data (LRR-FD).

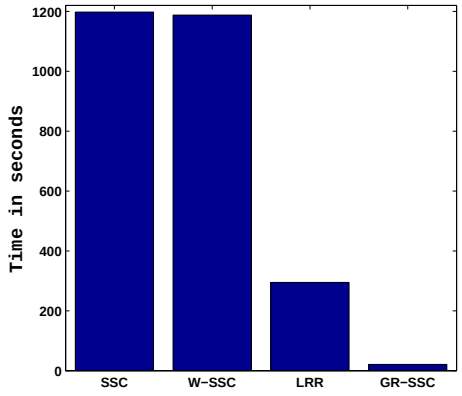
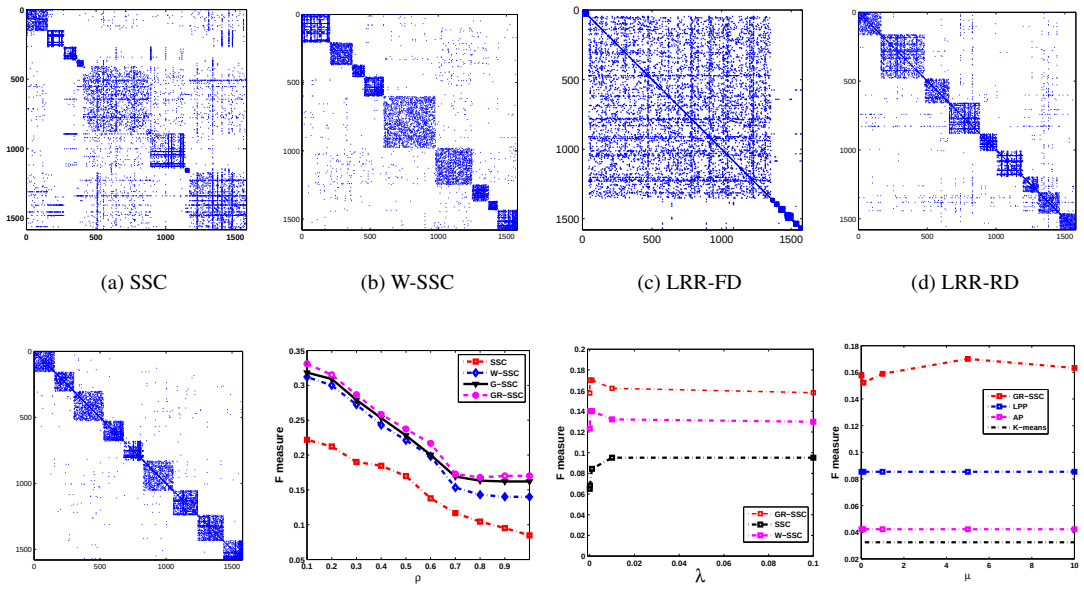


Figure 3: Computational time for $\mathbf{C}$ (NUS-WIDE Dataset)

Initially, we set regularization parameter λ to 0.001, and the penalty parameter μ to 5 and ρ to 0.9. Table 2 presents the experimental results obtained for all datasets on which proposed method **GR-SSC outperforms all variants of SSC and state-of-the-arts benchmark methods** as well. Interestingly, G-SSC (which is GR-SSC without graph regularizer) also performs better than the state-of-art methods. On diabetes dataset, the F-measure for GR-SSC improves by a margin of $\mathbf{88}\%$, $\mathbf{21}\%$, $\mathbf{466}\%$ and $\mathbf{13}\%$ with respect to SSC, W-SSC, LRR-FD and LRR-RD respectively. Note that, there is also a large improvement in performance for LRR-RD against existing LRR-FD. Similarly, Rand index is also improved by $\mathbf{53}\%$, $\mathbf{3.5}\%$, $\mathbf{183}\%$, $\mathbf{10}\%$ against SSC, W-SSC, LRR-FD and LRR-RD respectively. Figures 4a-4e show the sparse representation matrix $\mathbf{C}$ of different subspace clustering methods. In ideal case, the nonzero entries at off-the-block diagonal locations in $\mathbf{C}$ must be zero. But due to noise and numerical properties of the data, this nonzero entries eventually contributes to the missclassification of the data points. If we visually compare the sparse representation matrices SSC (s) and GR-SSC, the number of nonzero entries at off-the-block diagonal location are lesser for GR-SSC. Hence, the betterment in performance of GR-SSC is consistent with F and *rand index* measure. F - measure is



(a) SSC (b) W-SSC (c) LRR-FD (d) LRR-RD

(e) GR-SSC (f) (g) (h)

Figure 4: SSC plots : Figure 4a - 4e and Clustering performances for different value of ρ, λ and μ: Figure 4f - 4h.

also improved by $\mathbf{325}\%$, $\mathbf{112}\%$, $\mathbf{466}\%$ while *rand index* is by $\mathbf{92.5}\%$, $\mathbf{16.7}\%$ and $\mathbf{107}\%$ against benchmark methods AP, LPP and k-means respectively. For NUS-WIDE dataset, F - measure is bettered by $\mathbf{42}\%$, $\mathbf{26}\%$, $\mathbf{20}\%$, $\mathbf{25}\%$,$\mathbf{337}\%$ and *rand index* $\mathbf{12}\%$,$\mathbf{10}\%$,$\mathbf{7}\%$,$\mathbf{6}\%$,$\mathbf{65}\%$ against SSC, LRR-RD, AP, LPP and k-means respectively. On 20 Newsgroup dataset, the performance is improved by $\mathbf{60}\%$, $\mathbf{11}\%$, $\mathbf{95}\%$, $\mathbf{26}\%$,$\mathbf{96}\%$ (F - measure) and $\mathbf{41}\%$, $\mathbf{10}\%$, $\mathbf{88}\%$, $\mathbf{19}\%$,$\mathbf{108}\%$ (*rand index*) against SSC, LRR-RD, AP, LPP and k-means respectively (see Table 2). Figure 5 shows the tag clouds of the diagnostic codes where clouds are corresponding to the each cluster in Figure 4e. As expected the clouds are qualitatively different in terms of grouping of similar disease within diabetes: for example tobacco disorder, tobacco addiction, cancer treatment, Hypertension etc. Most importantly, Type 1 and Type 2 diabetes are clearly identified.

Influence of weighting schemes: Table 2c include the performance of several weighting schemes and it is observed that cosine measure has better performance for Diabetics and NUS-WIDE data, whereas RBF is best for 20 Newsgroups among other choices. Regardless of which weighting scheme used, the proposed method always outperforms others.

Selection of Parameters: The two important parameters for model selection are λ and μ respectively. We investigate the influence of λ and μ by fixing one of them while varying other. Figure 4g and 4h computed on Diabetes data where we found that the performance of GR-SSC is significantly better than other algorithms over a wide range of λ and μ. We also vary ρ from 0.1 to 1 and results are presented in Figure 4f . As expected, F is high for small values of ρ and F is low when ρ is increasing.

Computational cost Fig. 3 shows the computational cost of the compared sparse subspace methods on NUS-WIDE data. Our proposed method GR-SSC is approximately $\mathbf{14}$ and $\mathbf{57}$ times faster than the state-of-the-art sparse subspace methods.

4 Conclusion

We have presented a novel method for clustering of data modeled as generated from a union of subspaces. The method is formulated in a sparse subspace representation and extends previous sparse subspace clustering method considerably in that we frame it in a multi-task setting. Our key contributions are: 1) the introduction of two regularizers that control at both group level (block regularization) and coefficient level (graph regularization); and 2) an ADMM algorithm that is computationally efficient and provably convergent. The proposed formulation is general and treats many existing methods as special cases. On challenging and realistic datasets, we demonstrate that it significantly outperforms both SSC variants and other popular and competitive approaches in data clustering.

References

[1] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Trans. Info. Theory*, 56(4):1982–2001, 2010.

Datasets	Diabetics Data		NUS-WIDE Data		20 Newsgroup	
Methods	F measure	Rand Index	F measure	Rand Index	F measure	Rand Index
SSC	0.0951	0.5817	0.1332	0.8001	0.5512	0.6812
W-SSC	0.1401	0.8652	0.1502	0.858	0.6231	0.7714
LRR-FD	0.0356	0.3193	0.0487	0.4875	0.3215	0.4555
LRR-RD	0.1504	0.8009	0.1501	0.8229	0.8001	0.8711
G-SSC	**0.1620**	**0.8802**	**0.1667**	**0.8644**	**0.8314**	**0.9311**
GR-SSC	**0.1701**	**0.8934**	**0.1895**	**0.8989**	**0.8816**	**0.9602**

(a) Performance analysis against Sparse subspace clustering methods

Datasets	Diabetics Data		NUS-WIDE Data		20 Newsgroup	
Methods	F measure	Rand Index	F measure	Rand Index	F measure	Rand Index
AP	0.0423	0.4639	0.1589	0.8405	0.4511	0.5122
LPP	0.0854	0.7654	0.1507	0.8510	0.7022	0.8122
k-means	0.0325	0.4312	0.0433	0.5449	0.4469	0.4689
GR-SSC	**0.1701**	**0.8934**	**0.1895**	**0.8989**	**0.8816**	**0.9602**

(b) Performance analysis against benchmark methods

Datasets	Diabetics Data		NUS-WIDE Data		20 Newsgroup	
Weighting Schemes	G-SSC	GR-SSC	G-SSC	GR-SSC	G-SSC	GR-SSC
RBF Kernel	0.1589	0.1695	0.1650	0.1850	**0.8314**	**0.8816**
0-1 matrix	0.1601	0.1684	0.1601	0.1799	0.8280	0.8684
Cosine measure	**0.1620**	**0.1701**	**0.1667**	**0.1895**	0.8281	0.8799

(c) F measure analysis using different weighting schemes

Table 2: Experimental results

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. *Foundations and Trends in Machine Learning*, volume 3, chapter Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, pages 1–122. 2011.

[3] E.J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(3), 2011.

[4] E.J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[5] E.J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE TIP*, 52(2):489–509, 2006.

[6] J.P. Costeira and T. Kanade. A multibody factorization method for independently moving objects. *IJCV*, 29(3):159–179, 1998.

[7] D.L. Donoho. Compressed sensing. *IEEE Trans. Info. Theory*, 52(4):1289–1306, 2006.

[8] Y.C. Eldar and M. Mishali. Robust recovery of signals from a structured union of subspaces. *IEEE Trans. Info. Theory*, 55(11):5302–5316, 2009.

[9] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Proc. CVPR*, pages 2790–2797, 2009.

[10] M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[11] B.J. Frey and D. Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.

[12] A. Gruber and Y. Weiss. Multibody factorization with uncertainty and missing data using the EM algorithm. In *Proc. CVPR*, 2004.

[13] S Gupta, D Phung, and S Venkatesh. A bayesian nonparametric joint factor model for learning shared and individual subspaces from multiple data sources. In *Proc. SDM*, pages 200–211, 2012.

[14] X. He, D. Cai, H. Liu, and W.Y. Ma. Locality preserving indexing for document representation. In *Proc. ACM SIGIR*, pages 96–103, 2004.

[15] J. Ho, M.H. Yang, J. Lim, K.C. Lee, and D. Kriegman. Clustering appearances of objects under varying illumination conditions. In *Proc. CVPR*, 2003.

[16] W. Hong, J. Wright, K. Huang, and Y. Ma. A multiscale hybrid linear model for lossy image representation. In *Proc. ICCV*, pages 764–771, 2005.

[17] J. Huang and T. Zhang. The benefit of group sparsity. *The Annals of Statistics*, 38(4):1978–2004, 2010.

[18] J Ye Jun Liu, S Ji. Mining sparse representations: Formulations, algorithms and applications. *Tutorial in SDM*, 2010.

[19] K. Kanatani. Motion segmentation by subspace separation and model selection. In *Proc. ICCV*, volume 2, pages 586–591, 2001.

(a) Cluster 1: Diabetes with Psychiatric Disorders

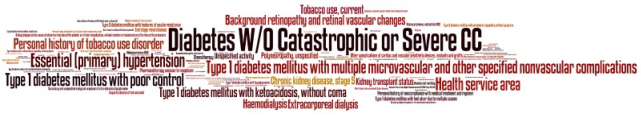
(b) Cluster 2: Type2 diabetes with Tobacco addiction

(c) Cluster 3: Diabetes with Hypertensions and Cancer treatment

(d) Cluster 4: Type 2 Diabetes with Hypertension

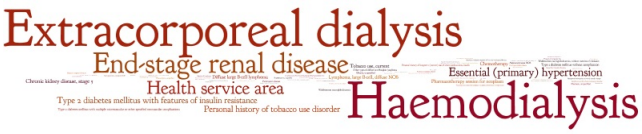
(e) Cluster 5: Type 2 diabetes with vascular complications and mix of other disease

(f) Cluster 6: Tobacco Disorder



(g) Cluster 7: Type 1 diabetes with Ketoacdosis

(h) Cluster 8: Type 2 daibetes with Hypertension, Tobacco disorder and Chest pain



(i) Cluster 9: Diabetic Nephropathy

Figure 5: Diagnosis Cloud

[20] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, and A.Y. Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE PAMI*, 24(7):881–892, 2002.

[21] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *Proc. ICML*, 2010.

[22] Y.M. Lu and M.N. Do. A theory for sampling signals from a union of subspaces. *IEEE Trans. Sig. Process.*, 56(6):2334–2345, 2008.

[23] B. Nasihatkon and R. Hartley. Graph connectivity in sparse subspace clustering. In *Proc. CVPR*, 2011.

[24] Duc-Son Pham, B Saha, dinh Phung, and Svetha Venkatesh. Improved subspace clustering via exploitation of spatial constraints. In *Proc. CVPR*. IEEE, 2012.

[25] S.R. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories. *IEEE PAMI*, pages 1832–1845, 2009.

[26] S.R. Rao, A.Y. Yang, S.S. Sastry, and Y. Ma. Robust algebraic segmentation of mixed rigid-body and planar motions from two views. *IJCV*, 88(3):425–446, 2010.

[27] M.E. Tipping and C.M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482, 1999.

[28] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca). *IEEE PAMI*, 27(12):1945–1959, 2005.

[29] R. Vidal, R. Tron, and R. Hartley. Multiframe motion segmentation with missing data using power factorization and GPCA. *IJCV*, 79(1):85–105, 2008.

[30] J. Vogt and V. Roth. A complete analysis of the l_1, p group-lasso. *arXiv preprint arXiv:1206.4632*, 2012.

[31] A.Y. Yang, J. Wright, Y. Ma, and S.S. Sastry. Unsupervised segmentation of natural images via lossy data compression. *In proc. CVIU*, 110(2):212–225, 2008.

[32] H. Zou and T. Hastie. Regression shrinkage and selection via the elastic net, with applications to microarrays. *Journal of the Royal Statistical Society: Series B. v67*, pages 301–320, 2003.