RESEARCH ARTICLE

Robust auto-weighted multi-view subspace clustering with common subspace representation matrix

Wenzhang Zhuge¹, Chenping Hou^{1*}, Yuanyuan Jiao², Jia Yue³, Hong Tao¹, Dongyun Yi¹

1 Department of Mathematics and System Science, National University of Defense Technology, Changsha, Hunan, China, **2** The College of Nine, National University of Defense Technology, Changsha, Hunan, China, **3** Key Laboratory, Taiyuan Satellite Launch Center, Taiyuan, Shanxi, China

* hcpnudet@hotmail.com

Abstract

In many computer vision and machine learning applications, the data sets distribute on certain low-dimensional subspaces. Subspace clustering is a powerful technology to find the underlying subspaces and cluster data points correctly. However, traditional subspace clustering methods can only be applied on data from one source, and how to extend these methods and enable the extensions to combine information from various data sources has become a hot area of research. Previous multi-view subspace methods aim to learn multiple subspace representation matrices simultaneously and these learning task for different views are treated equally. After obtaining representation matrices, they stack up the learned representation matrices as the common underlying subspace structure. However, for many problems, the importance of sources and the importance of features in one source both can be varied, which makes the previous approaches ineffective. In this paper, we propose a novel method called Robust Auto-weighted Multi-view Subspace Clustering (RAMSC). In our method, the weight for both the sources and features can be learned automatically via utilizing a novel trick and introducing a sparse norm. More importantly, the objective of our method is a common representation matrix which directly reflects the common underlying subspace structure. A new efficient algorithm is derived to solve the formulated objective with rigorous theoretical proof on its convergency. Extensive experimental results on five benchmark multi-view datasets well demonstrate that the proposed method consistently outperforms the state-of-the-art methods.

OPEN ACCESS

Citation: Zhuge W, Hou C, Jiao Y, Yue J, Tao H, Yi D (2017) Robust auto-weighted multi-view subspace clustering with common subspace representation matrix. PLoS ONE 12(5): e0176769. <https://doi.org/10.1371/journal.pone.0176769>

Editor: Zhao Zhang, Soochow University, CHINA

Received: January 4, 2017

Accepted: April 17, 2017

Published: May 23, 2017

Copyright: © 2017 Zhuge et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Funding: This work was supported by the National Natural Science Foundation of China (61473302, 61503396). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

1 Introduction

In many applications such as computer vision, data mining, pattern recognition and machine learning, there exists an assumption that the data points are drawn from multiple low-dimensional subspaces with each subspace corresponding to one category or class. Subspace clustering [1, 2] aims to explore the underlying subspace and cluster the data according to it.

Early subspace clustering methods can be roughly grouped into two categories: algebra based methods such as [3, 4], and statistics based methods such as [5, 6]. And recently, many methods [7–16] which belong to a new category, i.e., spectral clustering based [1] methods, have been proposed and these methods have achieved state-of-the-art performance. The core idea of spectral clustering based methods is to apply the self-representation property to compute affinities, i.e., represent every data point by a linear combination of other data points. However, these methods mostly focus on the features from single source rather than multiple ones.

In actual applications, data is often collected from diverse domains or obtained from different feature extractors, thus multi-view data are very common in many applications. For example, in computer vision, each image can be described by the color, texture, shapes and so on. In web mining, each web can be characterized by its content and link information, which are two distinct descriptions or views. In multi-lingual information retrieval, a document can be represented by several different languages. Since these different features can provide useful information from different views and these single-view subspace clustering methods have shown good performance, it is crucial to integrate these heterogeneous features to create more accurate robust multi-view subspace clustering methods.

More recently, a number of multi-view subspace clustering methods have been proposed [17–19]. The diversity-induced multi-view subspace clustering (DiMSC) was proposed in [17] to perform subspace clustering on different views simultaneously with a diverse term on the multiple representation matrices. The multi-view subspace clustering (MVSC) was introduced in [18] to perform clustering on the subspace representation of each view simultaneously with a common cluster structure. The low-rank tensor constrained multi-view subspace clustering (LT-MSC), which was proposed in [19], performs subspace clustering on different views simultaneously with a low rank tensor constraint, and the tensor is constructed by the subspace representation matrices. After obtaining subspace representation matrices, these methods use them to construct similarity matrices for different views independently and stack up these similarity matrices to a common one which represents the underlying common structures across different views. However, these methods neglect the different importance among views and the performance of their unified similarities may suffer when there is a less informative view.

In this paper, we try to solve the problem of subspace clustering for multi-view data. A novel method, named as Robust Auto-weighted Multi-view Subspace Clustering (RAMSC), has been presented. Different from the previous approaches [17–19] which treat different views equally and obtain a representation matrix in each view, our proposed method assigns a suitable weight for each view and purposes to learn a common representation matrix across different views to reflect the underlying common structure. Besides, the view weight factors can be tuned automatically and this process does not need any additional parameters. And by introducing an sparse norm, our proposed method is robust to the inaccurate features. We provide an effective algorithm to solve the proposed non-smooth minimization problem and prove that the algorithm will converge. In the algorithm, for each view, a feature weight matrix can be learned and we also proposes a new way to construct the common similarity matrix by utilizing the view weight factors and feature weight matrix. Compared to related state-of-the-art clustering methods, our proposed method consistently achieves better performance on five benchmark multi-view data sets.

The rest of this paper is organized as follows. Section 2 introduces the background and motivation of this paper. In Section 3, we propose our method RAMSC with a solving algorithm. In Section 4, we present some deep analyses about the proposed algorithm RAMSC,

including convergence behavior, computational complexity and parameter determination. Experimental results and conclusions are shown in Section 5 and Section 6, respectively.

2 Background and motivation

In this section, first we introduce some notations, then briefly review the previous subspace clustering methods to show our research motivation.

2.1 Notations

Throughout this paper, vectors and matrices are written in boldface uppercase letters and boldface lowercase letters, respectively. For a vector $\mathbf{m}$, the ℓ_2 -norm of vector $\mathbf{m}$ is denoted by $\|\mathbf{m}\|_2$. And $\mathbf{m}^{(v)}$ denotes that $\mathbf{m}$ is derived from v -th view. For a matrix $\mathbf{M}$, we denote its i -th row, j -th column and ij -th element as $\mathbf{m}_i$, $\mathbf{m}_j$ and m_{ij} respectively. The trace of matrix $\mathbf{M}$ is denoted by $\text{Tr}(\mathbf{M})$. And we denote $\mathbf{M}^{(v)}$ as a matrix $\mathbf{M}$ derived from the v -th view representation. The $\ell_{r,p}$ -norm of an matrix $\mathbf{M} \in \mathcal{R}^{d \times n}$ is defined as [20, 21]

$$\|\mathbf{M}\|_{r,p} = \left(\sum_{i=1}^d \left(\sum_{j=1}^n |m_{ij}|^r \right)^{\frac{p}{r}} \right)^{\frac{1}{p}}, r > 0, p > 0. \quad (1)$$

When $r \geq 1$ and $p \geq 1$, $\ell_{r,p}$ -norm becomes a valid norm because it satisfies the three norm conditions.

2.2 Single-view and multi-view subspace clustering

Suppose $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ is the data matrix with d -dimensional features and n data points. The subspace clustering methods based on spectral clustering mainly have the following two steps:

First, the self-representation property [7] is used to represent data matrix $\mathbf{X}$ as

$$\mathbf{X} = \mathbf{XZ} + \mathbf{E}, \quad (2)$$

where $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n] \in \mathbb{R}^{n \times n}$ is the self-representation matrix with each $\mathbf{z}_i$ being the representation of sample $\mathbf{x}_i$, and $\mathbf{E}$ is the error matrix. The nonzero elements of $\mathbf{z}_i$ correspond to points from the same subspace. And $\mathbf{Z}$ can be obtained by solving:

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \|\mathbf{X} - \mathbf{XZ}\|_l + \lambda \Omega(\mathbf{Z}), \\ \text{s.t.} \quad & \mathbf{Z} \in \mathcal{C} \end{aligned} \quad (3)$$

where $\|\cdot\|_l$ can be considered a proper norm on error matrix $\mathbf{E}$, $\Omega(\mathbf{Z})$ and $\mathcal{C}$ are the regularizer and constraint set on $\mathbf{Z}$, respectively, and $\lambda > 0$ is a balance parameter. The existing methods [7–13] distinguish each other by employing different constraints or regularizers on $\mathbf{Z}$ or $\mathbf{E}$;

Second, the obtained subspace structure $\mathbf{Z}$ is used to construct a similarity matrix $\mathbf{S}$ which encodes the pairwise similarity between data pairs by [22]

$$\mathbf{S} = \frac{|\mathbf{Z}^T| + |\mathbf{Z}|}{2}. \quad (4)$$

Afterwards, spectral clustering algorithm [23] can be used on the computed similarity matrix $\mathbf{S}$ to get the final clustering results.

For multi-view data, suppose that V is the number of views and $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(V)}$ are used to denote data matrix of each view, where $\mathbf{X}^{(v)} \in \mathbb{R}^{d^{(v)} \times n}$ for $v = 1, 2, 3, \dots, V$ and $d^{(v)}$ is the v -th view dimensionality. The single-view subspace clustering methods can not be applied on multi-view data to obtain a representation matrix directly. One naive strategy is to concatenate all the features together as a new view, and then employ single view methods on the concatenated features. However, this method ignores the difference among multiple views.

The previous multi-view subspace clustering methods consider that for each single view, a subspace representation $\mathbf{Z}^{(v)} \in \mathbb{R}^{n \times n}$ should be learned. They stack up these V tasks and focus on how to explore the relationships among these V representation matrices $\mathbf{Z}^{(v)}$ so that these $\mathbf{Z}^{(v)}$ can be learned simultaneously. Two more reasonable strategies in multi-view learning are adopted by them to achieve their goal: The first one is to explore complementary information from multiple views. DiMSC [17] explores the complementarity of these representations $\mathbf{Z}^{(v)}$ by applying the Hilbert Schmidt independence criterion as a diversity term. LT-MSC [19] explores the complementary information from multiple views by regarding the subspace representation matrices $\mathbf{Z}^{(v)}$ as a tensor, then equipping the tensor with a low-rank constraint. The second strategy is to explore the consistence among multiple views. MVSC [18] explores the consistence of these representations $\mathbf{Z}^{(v)}$ by performing subspace clustering on individual modality respectively and then unifying them by a common indicator matrix.

After obtaining a representation $\mathbf{Z}^{(v)}$ for each view, all the above-mentioned multi-view subspace clustering methods construct a similarity matrix $\mathbf{S}$ by

$$\mathbf{S} = \sum_{v=1}^V \frac{|\mathbf{Z}^{(v)T}| + |\mathbf{Z}^{(v)}|}{2}. \quad (5)$$

Then they apply spectral clustering algorithm [23] on $\mathbf{S}$ to obtain clustering results.

Although these multi-view subspace clustering methods have achieved good performance, there are mainly two drawbacks of these methods which leave room to improve the clustering performance:

1. These methods treat different views equally and neglect the different importance of different views. When they learn $\mathbf{Z}^{(v)}$, each view plays the same important role. When they construct the similarity matrix $\mathbf{S}$, Eq (5) can be considered as $\mathbf{S} = \sum_{v=1}^V \mathbf{S}^{(v)}$, where $\mathbf{S}^{(v)} = (|\mathbf{Z}^{(v)T}| + |\mathbf{Z}^{(v)}|)/2$ is a graph similarity matrix constructed from v -th view. This strategy may suffer when an unreliable similarity matrix is added to.
2. Eq (5) can also be considered as $\mathbf{S} = (|\mathbf{Z}^T| + |\mathbf{Z}|)/2$, where $|\mathbf{Z}| = \sum_{v=1}^V |\mathbf{Z}^{(v)}|$ can be considered as the underlying common structures across different views. The optimizing objectives of these methods are representation matrices $\mathbf{Z}^{(v)}$, however, the final clustering results are determined by the common structure $\mathbf{Z}$ which may brings such a drawback that these $\mathbf{Z}^{(v)}$ may have good properties because of the constraints or regularized terms, but $\mathbf{Z}$ may not keep these properties.

To address these two challenges, we will introduce our proposed novel multi-view subspace clustering method in next section.

3 Formulation and solution

In this section, we will first introduce the formulation of our method, and then an alternative algorithm will be presented to solve it.

3.1 Formulation

To overcome these two drawbacks, we propose a novel robust auto-weighted multi-view subspace clustering method. Our proposed method RAMSC utilizes a reasonable way to set view weight factors automatically and learns a common subspace representation $\mathbf{Z}$ which can be directly used to construct the common similarity matrix $\mathbf{S}$ across different views. Thus one important view can have a big weight, and the constraints or regularized terms can be set on $\mathbf{Z}$ which determines the final clustering results. The objective function of RAMSC is

$$\min_{\mathbf{Z}} \sum_{v=1}^V \sqrt{\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}\|_{2,p}^p + \lambda \Omega^{(v)}(\mathbf{Z})}, \quad (6)$$

where λ is a tradeoff factor, $\|\cdot\|_{2,p}^p$ is the sparsity-inducing norm with $0 \leq p \leq 1$ and each $\Omega^{(v)}(\mathbf{Z})$ is a smooth regularized term. Denote the representation error matrix $\mathbf{E}^{(v)} \in \mathbb{R}^{d \times n}$ of the v -th view as

$$\mathbf{E}^{(v)} = \mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}. \quad (7)$$

The $\|\cdot\|_{2,p}^p$ -norm of a matrix $\mathbf{E} \in \mathbb{R}^{d \times n}$ is defined as

$$\|\mathbf{E}^{(v)}\|_{2,p}^p = \sum_{i=1}^d \left(\sum_{j=1}^n |e_{ij}^{(v)}|^2 \right)^{\frac{p}{2}} = \sum_{i=1}^d (\|\mathbf{e}_i^{(v)}\|_2)^p, \quad (8)$$

where $\mathbf{e}_i^{(v)}$ is the i -th row of $\mathbf{E}^{(v)}$ and $\|\mathbf{E}^{(v)}\|_{2,p}$ is the $\ell_{2,p}$ -norm as defined in Eq (1) with $r = 2$. $\Omega^{(v)}(\mathbf{Z})$ aims to smooth the distribution of the common representation $\mathbf{Z}$ on the v -th view. These v smooth regularized terms $\Omega^{(v)}(\mathbf{Z})$ enforce the common subspace representation matrix $\mathbf{Z}$ to meet the grouping effect. This analogous smooth regularized term is also employed by [13, 17, 24]. Specifically, in our method, each regularized term $\Omega^{(v)}(\mathbf{Z})$ for $v = 1, 2, \dots, V$ is defined as:

$$\Omega^{(v)}(\mathbf{Z}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n w_{ij}^{(v)} \|\mathbf{z}_i - \mathbf{z}_j\|_2^2 = \text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T), \quad (9)$$

$\mathbf{W}^{(v)} = (w_{ij}^{(v)})$ is the weight matrix measuring the spatial closeness of the data points on v -th view. $\mathbf{L}^{(v)} = \mathbf{D}^{(v)} - \mathbf{W}^{(v)}$ is the Laplacian matrix, in which the degree $\mathbf{D}^{(v)}$ is the diagonal matrix with $d_{ii}^{(v)} = \sum_{j=1}^n w_{ij}^{(v)}$. $\mathbf{W}^{(v)}$ can be constructed by many different ways [25–29]. To show the robustness of our method, we construct 0-1 binary weighted k -nn graphs for each view and k is set to be 5 in all experiments.

Intuitively, there is no weight factor explicitly defined in Eq (6), and all different views are treated equally. By the following analysis, it can provide a reasonable way to learn the weight factors of each view. The Lagrange function of problem (6) can be written as

$$\sum_{v=1}^V \sqrt{\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T)}. \quad (10)$$

Taking the derivative of Eq (10) with respect to $\mathbf{Z}$ and setting the derivative to zero, we have

$$\sum_{v=1}^V \alpha^{(v)} \frac{\partial (\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T))}{\partial \mathbf{Z}} = 0, \quad (11)$$

where

$$\alpha^{(v)} = \frac{1}{2\sqrt{\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T)}}. \quad (12)$$

Eq (11) can not be directly solved because $\alpha^{(v)}$ is dependent on the target variable $\mathbf{Z}$. However, if $\alpha^{(v)}$ is considered as the weight factor of the v -th view, and its value has been given or set to be stationary, Eq (11) can be considered as the solution of the following problem when these $\alpha^{(v)}$ are calculated or given:

$$\min_{\mathbf{Z}, \alpha^{(v)}} \sum_{v=1}^V \alpha^{(v)} (\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T)). \quad (13)$$

Solving the problem (13) to obtain the common representation matrix $\mathbf{Z}$ seems more reasonable. This problem can be considered as a sum of two parts with a tradeoff factor λ . The first part $\sum_{v=1}^V \alpha^{(v)} \|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}\|_{2,p}^p$ is a linear combination of the subspace representation errors on each view. Increasing $\alpha^{(v)}$ tends to reduce representation error on the v -th view. The second part is to smooth $\mathbf{Z}$ on a linear combination of Laplacian matrices with suitable weights $\alpha^{(v)}$, i.e., $\mathbf{L} = \sum_{v=1}^V \alpha^{(v)} \mathbf{L}^{(v)}$. According to [30–32], the accuracy of $\mathbf{L}$ can be higher than that of each $\mathbf{L}^{(v)}$ or the sum of them $\sum_{v=1}^V \mathbf{L}^{(v)}$.

Supposing that the common representation $\mathbf{Z}$ can be calculated from Eq (13), this $\mathbf{Z}$ can be used to update $\alpha^{(v)}$ according to Eq (12). Learning $\alpha^{(v)}$ in this way has following reasonable explanations and merits:

1. If v -th view is good, then $\|\mathbf{E}^{(v)}\|_{2,p}^p$ and $\text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T)$ should be small, and thus according to Eq (12), the learned $\alpha^{(v)}$ is large.
2. The $\ell_{2,p}$ -norm of $\mathbf{E}^{(v)}$ enforces the ℓ_p -norm along the features direction of representation error matrix $\mathbf{E}^{(v)}$, and the ℓ_2 -norm along the data points direction. Thus, when $0 \leq p \leq 1$, the effect of inaccurate features in the learning of $\alpha^{(v)}$ is reduced by the ℓ_p -norm.
3. Unlike [31–35], which depends on an extra parameter to smooth the distribution of the view weights, learning $\alpha^{(v)}$ by Eq (12) has no parameter to handle and it naturally avoids the trivial solution.

Although the problem (13) has a more reasonable form to learn a good common $\mathbf{Z}$, there are difficulties to solve it, which comes from the following two aspects: (1) the $\|\cdot\|_{2,p}^p$ terms are nonsmooth; (2) when $\alpha^{(v)}$ is calculated by Eq (12), $\alpha^{(v)}$ and $\mathbf{Z}$ are coupled with each other. In next subsection, we will propose an alternative algorithm to tackle them efficiently.

3.2 Optimization algorithm

To solve Eq (13), we consider the following problem to tackle the non-smooth norm problem:

$$\min_{\mathbf{Z}, \mathbf{U}^{(v)}, \alpha^{(v)}} \sum_{v=1}^V \alpha^{(v)} (\mathbf{H}^{(v)} + \lambda \text{Tr}(\mathbf{Z}\mathbf{L}^{(v)}\mathbf{Z}^T)), \quad (14)$$

where

$$\mathbf{H}^{(v)} = \text{Tr}((\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z})^T \mathbf{U}^{(v)} (\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z})). \quad (15)$$

$\mathbf{U}^{(v)} \in \mathbb{R}^{d^{(v)} \times d^{(v)}}$ is the diagonal matrix corresponding to the v -th view and the i -th entry on the

diagonal is defined as:

$$u_{ii}^{(v)} = \frac{p}{2} \|\mathbf{e}_i^{(v)}\|_2^{p-2}, \quad \forall i = 1, 2, \dots, d^{(v)}, \quad (16)$$

$u_{ii}^{(v)}$ is a subgradient of $\|\mathbf{E}^{(v)}\|_{2,p}^p$ w.r.t. $\mathbf{e}_i^{(v)}$. To avoid the situation $\mathbf{e}_i^{(v)} = 0$, which makes $u_{ii}^{(v)} = \frac{p}{2} \|0\|_2^{p-2}$ can not be calculated when $p < 2$, in practice, we replace the $\ell_{2,p}$ -norm with the regularize $\ell_{2,p}$ -norm. And it is defined as:

$$\|\mathbf{E}^{(v)}\|_{2,p} = \left(\sum_{i=1}^d \left((\mathbf{e}_i^{(v)})^T (\mathbf{e}_i^{(v)}) + \epsilon \right)^{\frac{p}{2}} \right)^{\frac{1}{p}}, \quad (17)$$

when $\epsilon \rightarrow 0$, the regularized $\|\cdot\|_{2,p}^p$ of $\mathbf{E}^{(v)}$ approximates the $\|\mathbf{E}^{(v)}\|_{2,p}^p$. Thus $u_{ii}^{(v)}$ now can be regularized as

$$u_{ii}^{(v)} = \frac{p}{2} \left((\mathbf{e}_i^{(v)})^T (\mathbf{e}_i^{(v)}) + \epsilon \right)^{\frac{p-2}{2}}. \quad (18)$$

This strategy avoids a bad situation, 0 on the denominator, and guarantees that we can repeat the following alternative steps.

- **The first step is fixing $\mathbf{U}^{(v)}$ and $\alpha^{(v)}$, updating the common subspace representation $\mathbf{Z}$.**
Differentiating the objective function J with respect to $\mathbf{Z}$ and setting it to zero

$$\mathbf{AZ} + \mathbf{ZB} + \mathbf{C} = 0, \quad (19)$$

where

$$\begin{aligned} \mathbf{A} &= \sum_{v=1}^V \alpha^{(v)} \mathbf{X}^{(v)T} \mathbf{U}^{(v)} \mathbf{X}^{(v)}, \\ \mathbf{B} &= \lambda \sum_{v=1}^V \alpha^{(v)} \mathbf{L}^{(v)}, \\ \mathbf{C} &= - \sum_{v=1}^V \alpha^{(v)} \mathbf{X}^{(v)T} \mathbf{U}^{(v)} \mathbf{X}^{(v)}. \end{aligned} \quad (20)$$

Eq (19) is a standard Sylvester equation, and according to [36], it has a unique optimal solution.

- **The second step is fixing $\alpha^{(v)}$ and $\mathbf{Z}$, updating the feature weight matrix $\mathbf{U}^{(v)}$ for each view.**

The representation error matrix $\mathbf{E}^{(v)}$ of each view is calculated by current $\mathbf{Z}$, and then each diagonal element of $\mathbf{U}^{(v)}$ is updated by Eqs (16) or (18).

- **The third step is fixing $\mathbf{Z}$ and $\mathbf{U}^{(v)}$, updating the view weight factors $\alpha^{(v)}$ for each view by Eq (12).**

By the above three steps, we alternatively update $\mathbf{Z}$, $\mathbf{U}^{(v)}$ as well as $\alpha^{(v)}$, and repeat the process iteratively. Until now, we can draw the following conclusions:

1. In the above procedures, **the alternating optimization converges**, and $\mathbf{Z}^*$ which denotes the converged value of $\mathbf{Z}$ is at least a **local optimal solution to Eq (6)**. (We will prove this conclusion in next section).
2. The second one is about initialization. Since these procedures can reach a local optimum of Eq (6), it is important to have a sensible initialization. We initialize all views with equal $\alpha^{(v)} = \frac{1}{V}$ as in previous approaches [37, 38]. And as in previous researches [21, 39], we initialize $\mathbf{U}^{(v)} = \mathbf{I}^{(v)}$ since every feature on each view has the same importance at the beginning.

After obtaining the common self-representation matrix $\mathbf{Z}^*$, the similarity matrix $\mathbf{S}_1$ can be defined as

$$\mathbf{S}_1 = \frac{|\mathbf{Z}^*| + |\mathbf{Z}^{*T}|}{2} \quad (21)$$

and use the spectral clustering algorithm to produce the final clustering results, as has been adopted by traditional single-view subspace clustering methods.

Some single-view subspace clustering methods also use other ways to construct similarity matrix [13]. In this paper, to better exploit the merit of grouping effect, we further utilize the learned view weight factors $\alpha^{(v)}$ and feature weight matrices $\mathbf{U}^{(v)}$ to define a new similarity matrix $\mathbf{S}_2$ as

$$\mathbf{S}_2 = \left(\left| \frac{\mathbf{z}_{:i}^{*T} \mathbf{z}_{:j}^*}{\|\hat{\mathbf{x}}_{:i}\|_2 \|\hat{\mathbf{x}}_{:j}\|_2} \right|^\gamma \right), \quad (22)$$

where

$$\|\hat{\mathbf{x}}_{:i}\|_2 = \sqrt{\sum_{v=1}^V \alpha^{(v)} \sum_{k=1}^{d^{(v)}} u_{kk}^{(v)} x_{k,i}^{(v)2}}. \quad (23)$$

$\hat{\mathbf{x}}_{:i} = [\sqrt{\alpha^{(1)}} \hat{\mathbf{x}}_{:i}^{(1)T}, \dots, \sqrt{\alpha^{(V)}} \hat{\mathbf{x}}_{:i}^{(V)T}]^T \in \mathbb{R}^{\sum_v d^{(v)}}$ denotes the new i -th data point which concatenates re-weighted features, and $\hat{\mathbf{x}}_{:i}^{(v)} = [\sqrt{u_{11}^{(v)}} x_{1,i}^{(v)}, \dots, \sqrt{u_{d^{(v)} d^{(v)}}^{(v)}} x_{d^{(v)},i}^{(v)}]^T \in \mathbb{R}^{d^{(v)}}$ is the re-weighted feature on the v -th view. $\gamma > 0$ is utilized to control the similarity variances. The new similarity measure can be considered as the inner product of the new common representation vectors normalized by the norms of their new features which are weighted by view weight factors $\alpha^{(v)}$ and feature weight matrices $\mathbf{U}^{(v)}$.

Based on the above analysis, we summarize the procedures of our method RAMSC in Algorithm 1.

Algorithm 1 Algorithm to solve RAMSC in Eq (6)

Input:

1. Data for V views $\{\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(V)}\}$ and $\mathbf{X}^{(v)} \in \mathbb{R}^{d^{(v)} \times n}$
2. The expected number of clusters c ,
3. The parameter λ , p , k and γ .

Initialize:

1. Initialize the feature weight matrix $\mathbf{U}^{(v)} = \mathbf{I}^{(v)}$ for each view, where $\mathbf{I}^{(v)} \in \mathbb{R}^{d^{(v)} \times d^{(v)}}$ is the identity matrix.
2. Initialize the view weight factor $\alpha^{(v)} = \frac{1}{V}$ for each view.
3. Build the 0-1 weighted k -nn graphs $\mathbf{W}^{(v)}$ and compute the corresponding Laplacian matrices $\mathbf{L}^{(v)}$ for each view.

while not converged **do**

1. Compute the common representations $\mathbf{Z}$ by solving the Sylvester eq (19).

2. Update the diagonal feature weight matrix $\mathbf{U}^{(v)}$ for each view. Its diagonal elements can be updated by Eqs (16) or (18).

3. Update the view weight factor $\alpha^{(v)}$ for each view by Eq (12).

end while

4. Compute similarity matrix by either Eqs (21) or (22).

5. Use spectral clustering algorithm to obtain c clusters.

Output: Clustering result.

4 Performance analysis

4.1 Convergence analysis

To prove that the proposed Algorithm 1 converges and it can reach at least a local optimal solution of Eq (6), we first need to introduce the following lemma [21].

Lemma 1 When $0 < p \leq 2$, for any positive number a and b , the inequality holds:

$$a^p - \frac{p}{2} \frac{a^2}{b^{2-p}} \leq b^p - \frac{p}{2} \frac{b^2}{a^{2-p}}. \quad (24)$$

Theorem 1 Each updated $\mathbf{Z}$ in Alg. 1 will monotonically decrease the objective of the problem (13) in each iteration.

Proof: Denote $\tilde{\mathbf{Z}}$ as the updated $\mathbf{Z}$ in each iteration and $\tilde{\mathbf{E}}^{(v)} = \mathbf{X}^{(v)} - \mathbf{X}^{(v)}\tilde{\mathbf{Z}}$ is the v -th representation error matrix calculated by $\tilde{\mathbf{Z}}$. According to the optimization to $\tilde{\mathbf{Z}}$ in Alg. 1, $\tilde{\mathbf{Z}}$ reaches the unique optimal solution of the problem (14) when $\alpha^{(v)}$ and $\mathbf{U}^{(v)}$ are fixed, so

$$\begin{aligned} & \sum_{v=1}^V \alpha^{(v)} (Tr(\tilde{\mathbf{E}}^{(v)T} \mathbf{U}^{(v)} \tilde{\mathbf{E}}^{(v)}) + \lambda Tr(\tilde{\mathbf{Z}} \mathbf{L}^{(v)} \tilde{\mathbf{Z}}^T)) \\ & \leq \sum_{v=1}^V \alpha^{(v)} (Tr(\mathbf{E}^{(v)T} \mathbf{U}^{(v)} \mathbf{E}^{(v)}) + \lambda Tr(\mathbf{Z} \mathbf{L}^{(v)} \mathbf{Z}^T)). \end{aligned} \quad (25)$$

Combining weight matrix $\mathbf{U}^{(v)}$ which $u_{ii}^{(v)} = \frac{p}{2} \|\mathbf{e}_i^{(v)}\|_2^{p-2}$, this inequation can be rewritten as:

$$\begin{aligned} & \sum_{v=1}^V \alpha^{(v)} \left(\sum_{i=1}^{d^{(v)}} \frac{p}{2} \frac{\|\tilde{\mathbf{e}}_i^{(v)}\|_2^2}{\|\mathbf{e}_i^{(v)}\|_2^{2-p}} + \lambda Tr(\tilde{\mathbf{Z}} \mathbf{L}^{(v)} \tilde{\mathbf{Z}}^T) \right) \\ & \leq \sum_{v=1}^V \alpha^{(v)} \left(\sum_{i=1}^{d^{(v)}} \frac{p}{2} \frac{\|\mathbf{e}_i^{(v)}\|_2^2}{\|\mathbf{e}_i^{(v)}\|_2^{2-p}} + \lambda Tr(\mathbf{Z} \mathbf{L}^{(v)} \mathbf{Z}^T) \right). \end{aligned} \quad (26)$$

Generally, $\|\mathbf{e}_i^{(v)}\|_2 > 0$ and $\|\tilde{\mathbf{e}}_i^{(v)}\|_2 > 0$, and the regularized $\ell_{2,p}$ -norm can be used to guarantee it. According to Lemma 1, we can derive

$$\|\tilde{\mathbf{e}}_i^{(v)}\|_2^p - \frac{p}{2} \frac{\|\tilde{\mathbf{e}}_i^{(v)}\|_2^2}{\|\mathbf{e}_i^{(v)}\|_2^{2-p}} \leq \|\mathbf{e}_i^{(v)}\|_2^p - \frac{p}{2} \frac{\|\mathbf{e}_i^{(v)}\|_2^2}{\|\mathbf{e}_i^{(v)}\|_2^{2-p}}. \quad (27)$$

Thus the following inequality holds

$$\begin{aligned} & \sum_{v=1}^V \alpha^{(v)} \sum_{i=1}^{d^{(v)}} \|\tilde{\mathbf{e}}_i^{(v)}\|_2^p - \sum_{v=1}^V \alpha^{(v)} \sum_{i=1}^{d^{(v)}} \frac{p}{2} \frac{\|\tilde{\mathbf{e}}_i^{(v)}\|_2^2}{\|\mathbf{e}_i^{(v)}\|_2^{2-p}} \\ & \leq \sum_{v=1}^V \alpha^{(v)} \sum_{i=1}^{d^{(v)}} \|\mathbf{e}_i^{(v)}\|_2^p - \sum_{v=1}^V \alpha^{(v)} \sum_{i=1}^{d^{(v)}} \frac{p}{2} \frac{\|\mathbf{e}_i^{(v)}\|_2^2}{\|\mathbf{e}_i^{(v)}\|_2^{2-p}}. \end{aligned} \quad (28)$$

Summing Eqs (26) and (28), we have

$$\begin{aligned} & \sum_{v=1}^V \alpha^{(v)} (\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)} \tilde{\mathbf{Z}}\|_{2,p}^p + \lambda \text{Tr}(\tilde{\mathbf{Z}} \mathbf{L}^{(v)} \tilde{\mathbf{Z}}^T)) \\ & \leq \sum_{v=1}^V \alpha^{(v)} (\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)} \mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z} \mathbf{L}^{(v)} \mathbf{Z}^T)). \end{aligned} \quad (29)$$

Thus the objective of the problem (13) has been decreased by $\tilde{\mathbf{Z}}$ in each iteration.

Theorem 2 Each updated $\mathbf{Z}$ in Alg. 1 will monotonically decrease the objective of the problem (6) in each iteration, which makes converged $\mathbf{Z}^*$ be a local optimal solution.

Proof: Denote $\mathcal{L}^{(v)}(\mathbf{Z})$ as

$$\mathcal{L}^{(v)}(\mathbf{Z}) = \sqrt{\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)} \mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z} \mathbf{L}^{(v)} \mathbf{Z}^T)}. \quad (30)$$

According to Eq (29), $\tilde{\mathbf{Z}}$ makes the objective of Eq (13) have smaller value than $\mathbf{Z}$. Combining view weight factors $\alpha^{(v)} = 1/2\mathcal{L}^{(v)}(\mathbf{Z})$, we can derive

$$\sum_{v=1}^V \frac{1}{2} \frac{(\mathcal{L}^{(v)}(\tilde{\mathbf{Z}}))^2}{\mathcal{L}^{(v)}(\mathbf{Z})} \leq \sum_{v=1}^V \frac{1}{2} \frac{(\mathcal{L}^{(v)}(\mathbf{Z}))^2}{\mathcal{L}^{(v)}(\mathbf{Z})}. \quad (31)$$

Since $\mathcal{L}^{(v)}(\tilde{\mathbf{Z}}) > 0$ and $\mathcal{L}^{(v)}(\mathbf{Z}) > 0$, according to Lemma 1, when $p = 1$, we have

$$\sum_{v=1}^V \mathcal{L}^{(v)}(\tilde{\mathbf{Z}}) - \sum_{v=1}^V \frac{1}{2} \frac{(\mathcal{L}^{(v)}(\tilde{\mathbf{Z}}))^2}{\mathcal{L}^{(v)}(\mathbf{Z})} \leq \sum_{v=1}^V \mathcal{L}^{(v)}(\mathbf{Z}) - \sum_{v=1}^V \frac{1}{2} \frac{(\mathcal{L}^{(v)}(\mathbf{Z}))^2}{\mathcal{L}^{(v)}(\mathbf{Z})}. \quad (32)$$

Summing Eqs (31) and (32), we arrive at

$$\begin{aligned} & \sum_{v=1}^V \sqrt{\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)} \tilde{\mathbf{Z}}\|_{2,p}^p + \lambda \text{Tr}(\tilde{\mathbf{Z}} \mathbf{L}^{(v)} \tilde{\mathbf{Z}}^T)} \\ & \leq \sum_{v=1}^V \sqrt{\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)} \mathbf{Z}\|_{2,p}^p + \lambda \text{Tr}(\mathbf{Z} \mathbf{L}^{(v)} \mathbf{Z}^T)}. \end{aligned} \quad (33)$$

Thus the alternating optimization will monotonically decrease the objective of the problem (6) in each iteration until it converges. In the convergence, the converged $\mathbf{Z}^*$ satisfy the Eq (11) which is the KKT condition of problem (6). Therefore, $\mathbf{Z}^*$ is at least a local optimal solution of the problem (6).

4.2 Computational complexity and parameter determination

As seen from the procedure of RAMSC in Algorithm 1, we have solved this problem in an alternative way. The computational complexity in solving each problem is listed as follows. (1) The problem in Eq (19) can be solved by the Bartels-Stewart algorithm which has a computational complexity of $\mathcal{O}(n^3)$; (2) The problem in Eq (16) can be effectively solved by computing the 2-norm of a vector. The computational complexity is $\mathcal{O}(\sum_{v=1}^V (d^{(v)})^2)$; (3) Solving the problem in Eq (12) to update the optimal weight for each view has complexity $\mathcal{O}(n^2 \times V)$. In summary, the total computational complexity of RAMSC is $\mathcal{O}(T \times \max\{n^3, \sum_{v=1}^V (d^{(v)})^2\})$, where T is the number of iteration.

Since parameter determination is still an open problem [40, 41], we determine the parameters of our method empirically as in previous researches. As for p , it is designed to add sparsity

Table 1. Details of the multiview datasets used in our experiments (view type (dimensionality)).

View type	MSRC-v1	Caltech101-7	Digit	NBA-NASCAR	WebKB
1	LBP (256)	LBP (256)	FOU (76)	Gray (1024)	Fulltext (2949)
2	HOG (100)	PHOG (680)	FAC (216)	TFIDF (296)	Inlinks (334)
3	GIST (512)	GIST (512)	KAR (64)	-	-
4	CENTRIST (1302)	Gabor (32)	PIX (240)	-	-
5	CMT (48)	SURF (200)	ZER (47)	-	-
6	SIFT (200)	SIFT (200)	MOR (6)	-	-
Date points	210	441	2000	840	1051
Classes	7	7	10	2	2

<https://doi.org/10.1371/journal.pone.0176769.t001>

to representation error matrices $E^{(v)}$ which can alleviate the effect of inaccurate features in the learning of $\alpha^{(v)}$. Paper [43] is a timely and comprehensive survey, and a very good material to master the sparse learning field. According to it, we set $p = 1$, and this setting has been proven to be effective in most applications [20, 42]. As for k , it is the neighbor number to construct graphs $W^{(v)}$. Methods [13, 17] using similar regularized terms perform stably with different k , so we construct 5-nn graphs.

As for the parameter λ , it is very vital to the final performance since it is employed to balance the self representation accuracy and the smoothness of Z . Since there is no prior information about λ , we determine it by grid search in a heuristic way as in previous researches [13, 17, 42]. Concretely, λ is tuned from 1,2 and 5 to 60 with an incremental step 5 to get the best λ . When Eq (22) is used to construct the similarity matrix, we search it from 0.1 to 2 with an incremental step 0.2 to get the best γ .

5 Experiments

In this section, our proposed RAMSC has been evaluated on five widely used data sets, and some numerical results of its convergency behaviors and also have been shown.

5.1 Data set descriptions

To validate the effectiveness of our method, we use five multiview benchmark datasets. They are various kinds of data arisen in many real applications with different characters and commonly used in multiple view learning. They are Microsoft Research Cambridge Volume 1 (MSRC-v1) [44], Caltech101 [45], NBA-NASCAR [46], Handwritten Dutch Digit Recognition (Digit) [47] and Web Knowledge Base (WebKB) [48]. The statistics information of the five data sets is concluded in Table 1 and the detailed information about them is shown as the following

- **MSRC-v1** data set consists 240 images and is divided into 8 classes. Following [49], we select 7 classes composed of tree, building, airplane, cow, face, car, bicycle and each class has 30 images. To distinguish all of scenes, we extract 256 Local Binary Pattern (LBP), 100 Histogram of Oriented Gradient (HOG), 512 GIST, 1302 CENTRIST, 48 Color Moment (CMT) and 200 SIFT features.
- **Caltech101-7** data set is composed of 8677 objective images which belong to 101 categories. We selected 7 widely used classes, including DollaBill, Faces, Garfield, Motorbikes, Snoopy, Stop-Sign and Windsor-Chair. Following [50], the data set has totally 441 images. In order to obtain different views, we extract 256 LBP, 100 PyramidHOG (PHOG), 512 GIST, 32 Gabor texture, 200 SURF and 200 SIFT features.

- **Digit** data set contains 2,000 data points for 0 to 9 ten digit classes and each class has 200 data points. Six published features can be used for multi-view clustering: 76 Fourier coefficients of the character shapes (FOU), 216 profile correlations (FAC), 64 Karhunen-love coefficients (KAR), 240 pixel averages in 2×3 windows (PIX), 47 Zernike moment (ZER) and 6 morphological (MOR) features.
- **NBA-NASCAR** data set is collected from the sports gallery of the yahoo! website in 2008. Following [46], this data set consists 420 NBA images and 420 NASCAR images. For each image, there an attached short text describing information. To get different views, each image is normalized to have 1024 gray features, and from each text, 296 TFIDF features have been extracted.
- **WebKB** data set is a subset of web documents from four universities. This data set consists 1051 pages which are classified 2 classes: 230 Course pages and 821 Non-Course pages. Each page has 2 views: Fulltext view contains 2949 features representing the textual content on the web page, and Inlinks view consists 334 features recording that the anchor text on the hyperlinks pointing to the pages.

5.2 Experimental setup

To evaluate the performance of our method, we have compared our method with each single view counterpart. Single view methods on the concatenated features are also compared. Besides, we compare with other state-of-the-art methods, including robust multi-view K-means clustering (RMKMC) [33], pair-wised co-regularized multi-modal spectral clustering (PC-SPC) [30], centroid co-regularized multi-modal spectral clustering (CC-SPC) [30], multi-view subspace clustering (MVSC) [18] and diversity induced multi-view subspace clustering (DiMSC) [17].

- **SPC:** We employ the standard spectral clustering (SPC) [23] algorithm directly on each view, and report the results as baselines.
- **SMR:** We first run smooth representation clustering (SMR) [13] on each view features to get the subspace representations, and then run spectral clustering on such representations.
- **SPC-CON** and **SMR-CON:** We first concatenate all features together as a new single view, and then run SPC [23] and SMR [13] respectively on it.
- **RMKMC:** The robust multi-view K-means clustering method obtains the common cluster indicators across multiple views by minimizing the linear combination of the relaxed K-means on each view with learned weight factors.
- **PC-SPC:** This method enforces the corresponding point in different modality to have the same cluster membership by a pair-wised co-regularization term, which makes different views be same to each other.
- **CC-SPC:** This method is similar to PC-SPC, other than a centroid-based co-regularization term, which makes different views be same to a common one.
- **MVSC:** This method perform subspace clustering on individual modality respectively and then unify them with a common indicator matrix.
- **DiMSC:** This method learns subspace representations and employs the Hilbert-Schmidt Independence Criterion to enhance complementary information.

For fair comparison, we download the source codes of the compared methods from the authors' websites and follow their experimental settings and the parameter tuning steps in

their papers to get their best parameters. And for RAMSC, we construct 0-1 binary 5-nn graphs $\mathbf{W}^{(v)}$ for each view and the p is fixed 1 in all experiments. Thus only one parameter λ in our method needs to be tuned. We search the best λ from 1, 2 and 5 to 60 with incremental step 5. RAMSC(S_2) denotes that we use Eq (22) to construct S_2 , and the best parameter γ is searched from 0.1 to 2 with incremental step 0.2. And the experimental results are corresponding to their best parameters.

Before we do the clustering work, we first normalize each view of the multi-view data to make all the values in the range $[-1, 1]$. All the experiments are repeated 50 times independently, and the mean and standard deviation of the results are reported.

Three standard clustering evaluation metrics are utilized to measure the multi-view clustering performance, that is, **Clustering Accuracy (ACC)**, **Normalized Mutual Information (NMI)** and **Purity**.

5.3 Experimental results

The experiment results of the five datasets with three metrics are shown in Tables 2, 3, 4, 5 and 6. In terms of the clustering accuracy, we have the following observations.

1. From Tables 2, 3, 4, 5 and 6, we conclude that our proposed method outperforms the competing methods on all the benchmark datasets. And although ACC, NMI and Purity are three different evaluation metrics, they all indicate the advantages of our method. The clustering results show the effectiveness of the way to construct similarity matrix by Eq (22), and compared the way of Eq (21), it can achieve better or at least comparable performance.
2. From Tables 2, 3 and 4, it can be seen that some individual view features are more discriminative for performing clustering. And as for the comparison between single view methods

Table 2. Clustering results of different methods on MSRC-v1 data set. (mean($\pm$ std)). (On the following five result tables, two best results of each metrics are bold).

Method	ACC	NMI	Purity
SC(1)	0.6022($\pm$ 0.0510)	0.4887($\pm$ 0.0283)	0.6336($\pm$ 0.0382)
SC(2)	0.5755($\pm$ 0.0230)	0.4936($\pm$ 0.0262)	0.5965($\pm$ 0.0245)
SC(3)	0.6547($\pm$ 0.0369)	0.5865($\pm$ 0.0284)	0.6877($\pm$ 0.0377)
SC(4)	0.7002($\pm$ 0.0567)	0.6064($\pm$ 0.0330)	0.7096($\pm$ 0.0430)
SC(5)	0.2873($\pm$ 0.0124)	0.1448($\pm$ 0.0146)	0.3174($\pm$ 0.0122)
SC(6)	0.5489($\pm$ 0.0371)	0.4538($\pm$ 0.0311)	0.5750($\pm$ 0.0280)
SMR(1)	0.6166($\pm$ 0.0351)	0.4853($\pm$ 0.0307)	0.6388($\pm$ 0.0356)
SMR(2)	0.6184($\pm$ 0.0347)	0.4958($\pm$ 0.0242)	0.6283($\pm$ 0.0296)
SMR(3)	0.7074($\pm$ 0.0352)	0.6444($\pm$ 0.0248)	0.7490($\pm$ 0.0246)
SMR(4)	0.7596($\pm$ 0.0742)	0.7106($\pm$ 0.0373)	0.7733($\pm$ 0.0556)
SMR(5)	0.4932($\pm$ 0.0384)	0.3765($\pm$ 0.0291)	0.5101($\pm$ 0.0330)
SMR(6)	0.5538($\pm$ 0.0279)	0.4605($\pm$ 0.0259)	0.5752($\pm$ 0.0264)
SC-CON	0.5983($\pm$ 0.0337)	0.4796($\pm$ 0.0223)	0.6192($\pm$ 0.0287)
SMR-CON	0.7338($\pm$ 0.0495)	0.6920($\pm$ 0.0232)	0.7661($\pm$ 0.0326)
RMKMC	0.6501($\pm$ 0.0782)	0.5700($\pm$ 0.0537)	0.6728($\pm$ 0.0651)
PC-SPC	0.7936($\pm$ 0.0589)	0.6965($\pm$ 0.0278)	0.8029($\pm$ 0.0433)
CC-SPC	0.8368($\pm$ 0.0605)	0.7799($\pm$ 0.0306)	0.8546($\pm$ 0.0393)
MVSC	0.7444($\pm$ 0.0754)	0.7076($\pm$ 0.0507)	0.7626($\pm$ 0.0608)
DiMSC	0.7759($\pm$ 0.0462)	0.6788($\pm$ 0.0361)	0.7825($\pm$ 0.0382)
RAMSC	0.9078 ($\pm$ 0.0534)	0.8462 ($\pm$ 0.0326)	0.9125 ($\pm$ 0.0401)
RAMSC(S_2)	0.9149 ($\pm$ 0.0480)	0.8512 ($\pm$ 0.0260)	0.9184 ($\pm$ 0.0359)

<https://doi.org/10.1371/journal.pone.0176769.t002>

Table 3. Clustering results of different methods on Caltech101-7 data set. (mean($\pm$ std)).

Method	ACC	NMI	Purity
SC(1)	0.4208($\pm$ 0.0228)	0.3188($\pm$ 0.0253)	0.5233($\pm$ 0.0242)
SC(2)	0.4699($\pm$ 0.0288)	0.3839($\pm$ 0.0385)	0.5653($\pm$ 0.0299)
SC(3)	0.6116($\pm$ 0.0515)	0.5248($\pm$ 0.0490)	0.6644($\pm$ 0.0441)
SC(4)	0.5275($\pm$ 0.0342)	0.4348($\pm$ 0.0379)	0.5746($\pm$ 0.0276)
SC(5)	0.6265($\pm$ 0.0371)	0.5787($\pm$ 0.0211)	0.7031($\pm$ 0.0274)
SC(6)	0.5208($\pm$ 0.0225)	0.4453($\pm$ 0.0271)	0.5965($\pm$ 0.0218)
SMR(1)	0.3288($\pm$ 0.0197)	0.1602($\pm$ 0.0127)	0.3936($\pm$ 0.0097)
SMR(2)	0.4946($\pm$ 0.0303)	0.4645($\pm$ 0.0261)	0.6009($\pm$ 0.0303)
SMR(3)	0.6874($\pm$ 0.0336)	0.6331($\pm$ 0.0271)	0.7639($\pm$ 0.0336)
SMR(4)	0.5498($\pm$ 0.0218)	0.4558($\pm$ 0.0244)	0.5837($\pm$ 0.0221)
SMR(5)	0.5212($\pm$ 0.0587)	0.4631($\pm$ 0.0355)	0.6095($\pm$ 0.0357)
SMR(6)	0.5191($\pm$ 0.0229)	0.4090($\pm$ 0.0279)	0.6089($\pm$ 0.0274)
SC-CON	0.4303($\pm$ 0.0258)	0.3230($\pm$ 0.0284)	0.5317($\pm$ 0.0246)
SMR-CON	0.6247($\pm$ 0.0246)	0.6049($\pm$ 0.0288)	0.7095($\pm$ 0.0343)
RMKMC	0.6034($\pm$ 0.0680)	0.5488($\pm$ 0.0482)	0.6846($\pm$ 0.0541)
PC-SPC	0.6975($\pm$ 0.0499)	0.6547($\pm$ 0.0262)	0.7581($\pm$ 0.0288)
CC-SPC	0.7047($\pm$ 0.0654)	0.6879($\pm$ 0.0378)	0.7972($\pm$ 0.0389)
MVSC	0.6034($\pm$ 0.0309)	0.4766($\pm$ 0.0373)	0.6559($\pm$ 0.0314)
DiMSC	0.7312($\pm$ 0.0244)	0.6458($\pm$ 0.0179)	0.7698($\pm$ 0.0268)
RAMSC	0.7384 ($\pm$ 0.0082)	0.7276 ($\pm$ 0.0080)	0.8258 ($\pm$ 0.0115)
RAMSC(S_2)	0.7512 ($\pm$ 0.0171)	0.7388 ($\pm$ 0.0110)	0.8381 ($\pm$ 0.0166)

<https://doi.org/10.1371/journal.pone.0176769.t003>

Table 4. Clustering results on digit data set. (mean($\pm$ std)).

Method	ACC	NMI	Purity
SC(1)	0.5714($\pm$ 0.0484)	0.6061($\pm$ 0.0256)	0.6307($\pm$ 0.0347)
SC(2)	0.8329($\pm$ 0.0683)	0.8181($\pm$ 0.0337)	0.8493($\pm$ 0.0549)
SC(3)	0.6651($\pm$ 0.0438)	0.7804($\pm$ 0.0249)	0.7457($\pm$ 0.0355)
SC(4)	0.8144($\pm$ 0.0705)	0.8374($\pm$ 0.0299)	0.8476($\pm$ 0.0506)
SC(5)	0.5758($\pm$ 0.0389)	0.6273($\pm$ 0.0217)	0.6139($\pm$ 0.0356)
SC(6)	0.5978($\pm$ 0.0389)	0.6159($\pm$ 0.0217)	0.6179($\pm$ 0.0356)
SMR(1)	0.6210($\pm$ 0.0234)	0.5937($\pm$ 0.0155)	0.6374($\pm$ 0.0167)
SMR(2)	0.8555($\pm$ 0.0632)	0.8119($\pm$ 0.0308)	0.8666($\pm$ 0.0504)
SMR(3)	0.7890($\pm$ 0.0576)	0.7625($\pm$ 0.0272)	0.8007($\pm$ 0.0476)
SMR(4)	0.8697($\pm$ 0.0722)	0.8375($\pm$ 0.0302)	0.8852($\pm$ 0.0522)
SMR(5)	0.3505($\pm$ 0.0116)	0.2850($\pm$ 0.0092)	0.3573($\pm$ 0.0071)
SMR(6)	0.4628($\pm$ 0.0145)	0.4533($\pm$ 0.0141)	0.4889($\pm$ 0.0138)
SC-CON	0.8729($\pm$ 0.0698)	0.8756($\pm$ 0.0326)	0.8847($\pm$ 0.0598)
SMR-CON	0.8496($\pm$ 0.0725)	0.8344($\pm$ 0.0288)	0.8720($\pm$ 0.0501)
RMKMC	0.7853($\pm$ 0.0800)	0.8125($\pm$ 0.0384)	0.8190($\pm$ 0.0614)
PC-SPC	0.8682($\pm$ 0.0604)	0.8267($\pm$ 0.0303)	0.8759($\pm$ 0.0500)
CC-SPC	0.8768($\pm$ 0.0605)	0.8234($\pm$ 0.0338)	0.8855($\pm$ 0.0471)
MVSC	0.8242($\pm$ 0.0686)	0.8399($\pm$ 0.0355)	0.8286($\pm$ 0.0664)
DiMSC	0.8400($\pm$ 0.0569)	0.8076($\pm$ 0.0347)	0.8465($\pm$ 0.0518)
RAMSC	0.9299 ($\pm$ 0.0439)	0.8864 ($\pm$ 0.0199)	0.9343 ($\pm$ 0.0333)
RAMSC(S_2)	0.9173 ($\pm$ 0.0611)	0.8886 ($\pm$ 0.0310)	0.9229 ($\pm$ 0.0518)

<https://doi.org/10.1371/journal.pone.0176769.t004>

Table 5. Clustering results on NBA-NASCAR data set. (mean($\pm$ std)).

Method	ACC	NMI	Purity
SC(1)	0.5631($\pm$ 0.0000)	0.0139($\pm$ 0.0000)	0.5631($\pm$ 0.0000)
SC(2)	0.5036($\pm$ 0.0000)	0.0003($\pm$ 0.0000)	0.5036($\pm$ 0.0000)
SMR(1)	0.6440($\pm$ 0.0000)	0.0640($\pm$ 0.0000)	0.6440($\pm$ 0.0000)
SMR(2)	0.5750($\pm$ 0.0000)	0.0289($\pm$ 0.0000)	0.5750($\pm$ 0.0000)
SC-CON	0.5631($\pm$ 0.0000)	0.0132($\pm$ 0.0000)	0.5631($\pm$ 0.0000)
SMR-CON	0.7952($\pm$ 0.0000)	0.2712($\pm$ 0.0000)	0.7952($\pm$ 0.0000)
RMKMC	0.9858($\pm$ 0.0000)	0.9005($\pm$ 0.0000)	0.9858($\pm$ 0.0000)
PC-SPC	0.7250($\pm$ 0.0000)	0.1521($\pm$ 0.0000)	0.7250($\pm$ 0.0000)
CC-SPC	0.8357($\pm$ 0.0000)	0.3555($\pm$ 0.0000)	0.8357($\pm$ 0.0000)
MVSC	0.5131($\pm$ 0.0000)	0.0060($\pm$ 0.0000)	0.5131($\pm$ 0.0000)
DiMSC	0.5476($\pm$ 0.0000)	0.0071($\pm$ 0.0000)	0.5476($\pm$ 0.0000)
RAMSC	0.9893 ($\pm$ 0.0000)	0.9146 ($\pm$ 0.0000)	0.9893 ($\pm$ 0.0000)
RAMSC(S_2)	0.9917 ($\pm$ 0.0000)	0.9316 ($\pm$ 0.0000)	0.9917 ($\pm$ 0.0000)

<https://doi.org/10.1371/journal.pone.0176769.t005>

and previous multi-view approaches, the previous multi-view clustering methods can not always achieve better performances. This may be caused by the fact that previous methods characterize the structures of each view data separately and combine them by simply addition operations, which makes the final clustering results affect by these inaccurate structures. Our approach can perform better than single view methods in most cases because our method distributes small weight factors for inaccurate views and learns a common self representation matrix $\mathbf{Z}$ which can be used to construct a common similarity matrix $\mathbf{S}$ among different views.

- Tables 5 and 6 show the robustness of our method. On NBA-NASCAR data set, all the competing methods except RMKMC can not achieve reasonable performance. It is because that RMKMC utilizes a weight factor for each view and the sparsity-inducing norm to eliminate the influence of the outliers, while the other competing methods do not consider the outliers and sparsity of the input data. Compared with RMKMC, our method learns view weight factors automatically without an additional parameter and use the $\|\cdot\|_{2,p}^p$ norm to

Table 6. Clustering results on WebKB data set. (mean($\pm$ std)).

Method	ACC	NMI	Purity
SC(1)	0.7774($\pm$ 0.0000)	0.0013($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
SC(2)	0.7755($\pm$ 0.0000)	0.0027($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
SMR(1)	0.7402($\pm$ 0.0000)	0.0573($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
SMR(2)	0.8069($\pm$ 0.0000)	0.0665($\pm$ 0.0000)	0.8069($\pm$ 0.0000)
SC-CON	0.7774($\pm$ 0.0000)	0.0013($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
SMR-CON	0.7774($\pm$ 0.0000)	0.0013($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
RMKMC	0.8049($\pm$ 0.0000)	0.1592($\pm$ 0.0000)	0.8159($\pm$ 0.0000)
PC-SPC	0.7659($\pm$ 0.0000)	0.0991($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
CC-SPC	0.5785($\pm$ 0.0000)	0.0019($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
MVSC	0.7802($\pm$ 0.0000)	0.0041($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
DiMSC	0.6147($\pm$ 0.0000)	0.0006($\pm$ 0.0000)	0.7812($\pm$ 0.0000)
RAMSC	0.9401 ($\pm$ 0.0000)	0.5689 ($\pm$ 0.0000)	0.9401 ($\pm$ 0.0000)
RAMSC(S_2)	0.9439 ($\pm$ 0.0000)	0.5889 ($\pm$ 0.0000)	0.9439 ($\pm$ 0.0000)

<https://doi.org/10.1371/journal.pone.0176769.t006>

Fig 1. Convergence behaviors of RAMSC with $\lambda = 50$ on two datasets. (A) MSRC-v1; (B) Caltech101-7.

<https://doi.org/10.1371/journal.pone.0176769.g001>

eliminate the influence of the inaccurate features. Our method has better performance on NBA-NASCAR data set and still achieve good performance on WebKB data set when all other compared methods do not work.

5.4 Convergence behavior

In order to verify the convergence of Algorithm 1, we present the numerical results of the convergence behavior on datasets MSRC-v1 and Caltech101-7.

The convergence curves are displayed in Fig 1. As shown in Fig 1, the objective values of Eq (6) are non-increasing during the iterations and converge to a fixed value. Additionally, our algorithm converges within 10 iterations which means it has fast convergence speed.

5.5 Parameter determination

As for the parameter determination problem, we conduct experiments on two data sets, i.e., NBA-NASCAR and WebKB, for evaluation. Since we fix $k = 5$ and $p = 1$, only the balance parameter λ needs to be tuned when we use Eq (21) to construct S_1 . We vary it from $\{1, 2, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60\}$ and ACC and NMI are employed as the evaluation criteria. The results are shown in Fig 2.

When we use Eq (22) to construct S_2 , there is an additional parameter γ . To show the influence of λ and γ on $\text{RAMSC}(S_2)$, we vary λ from $\{2, 10, 20, 30, 40, 50, 60\}$, and γ is varied from $\{0.1, 0.3, 0.5, 0.7, 0.9, 1.1, 1.3, 1.5, 1.7, 1.9\}$. ACC is employed as the evaluation criterion.

As we can see from the results in Figs 2 and 3, it is clear that the final clustering results of RAMSC and $\text{RAMSC}(S_2)$ are affected by different λ and the combinations of λ and γ , respectively. Besides, on the two data sets, RAMSC has different optimal λ , and $\text{RAMSC}(S_2)$ has different optimal combination of λ and γ because the two data sets have different data characteristics.

Fig 2. ACC and NMI of RAMSC with different selection of parameter λ . (A) NBA-NASCAR; (B) WebKB.

<https://doi.org/10.1371/journal.pone.0176769.g002>

Fig 3. ACC and NMI of RAMSC(S_2) with different combinations of parameters λ and γ . (A) NBA-NASCAR; (B) WebKB.

<https://doi.org/10.1371/journal.pone.0176769.g003>

6 Conclusion

In this paper, we have proposed a novel robust auto-weighted multi-view subspace clustering model, named RAMSC. This model can naturally assign suitable weights for each view and learn a common representation matrix. The common representation matrix can be used to construct a similarity matrix directly. Moreover, by imposing the structured sparsity norm, our method is robust to the inaccurate features. And the relative proof guarantees that the proposed method can converge to a local optimal solution. Experimental results on five data sets show that our proposed method enables a higher degree of accuracy than the state-of-the-art methods. However, there still remains several problems for future work:

- A series of relative methods need to be developed and systematically compared. The core idea of our method is to learn view weights automatically and find a high-quality common subspace representation matrix. Based on it, we list three possible ways to develop new relative methods. First, the smooth regularized terms of our method can be replaced by others; Second, the sparsity norm on error matrix can be considered to replace by other reasonable norms; Third, our method has no constraint, and some constraints on the common representation matrix or the error matrix can be added. According to specific applications, corresponding relative methods can be proposed.
- Another open problem lies in the selection of the parameters, especially in the balance parameter λ , which is still an unsolved problem in many learning algorithms. In this paper, we determine it empirically. Additional theoretical analysis is also needed for this topic.

Supporting information

S1 Appendix. RAMSC. A file contains matlab codes of RAMSC and the normalized datasets used in this paper.
(ZIP)

Acknowledgments

Chenping Hou was the corresponding author of this paper.

Author Contributions

Conceptualization: CH WZ.

Data curation: YJ.

Formal analysis: WZ CH HT.

Investigation: JY.

Methodology: WZ CH DY.

Project administration: DY.

Resources: YJ.

Software: WZ JY.

Supervision: DY.

Writing – original draft: WZ.

Writing – review & editing: HT YJ.

References

1. Vidal R. Subspace Clustering. IEEE Signal Processing Magazine. 2011; 28(2):52–68. <https://doi.org/10.1109/MSP.2010.939739>
2. Gu Q, Zhou J. Subspace maximum margin clustering. In: CIKM; 2009. p. 1337–1346.
3. Costeira J, Kanade T. A Multibody Factorization Method for Independently Moving Objects. International Journal of Computer Vision. 1998; 29(3):159–179. <https://doi.org/10.1023/A:1008000628999>

4. Vidal R, Ma Y, Sastry S. Generalized principal component analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2005; 27(12):1945–1959. <https://doi.org/10.1109/TPAMI.2005.244> PMID: 16355661
5. Fischler MA, Bolles RC. Random Sample Consensus: A Paradigm for Model Fitting with Applications To Image Analysis and Automated Cartography. *Communications of the Acm*. 1980; 24(6):381–395. <https://doi.org/10.1145/358669.358692>
6. Kuhn KW. The Hungarian Method for the assignment problem. In: *Naval research logistics quarterly*; 1955. p. 83–97. <https://doi.org/10.1002/nav.3800020109>
7. Elhamifar E, Vidal R. Sparse Subspace Clustering: Algorithm, Theory, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2013; 35(11):2765–81. <https://doi.org/10.1109/TPAMI.2013.57> PMID: 24051734
8. Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y. Robust Recovery of Subspace Structures by Low-Rank Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2013; 35(1):171–184. <https://doi.org/10.1109/TPAMI.2012.88> PMID: 22487984
9. Wang S, Yuan X, Shen J, Yao T, Yan S. Efficient Subspace Segmentation via Quadratic Programming. In: *AAAI*; 2011.
10. Luo D, Nie F, Ding C, Huang H. Multi-Subspace Representation and Discovery. *Machine Learning and Knowledge Discovery in Databases*. 2011; 6912(1):405–420. https://doi.org/10.1007/978-3-642-23783-6_26
11. Lu CY, Min H, Zhao ZQ, Zhu L, Huang DS, Yan S. Robust and Efficient Subspace Segmentation via Least Squares Regression. In: *ECCV*; 2012. p. 347–360.
12. Zhang Z, Zhao M, Chow TWS. Binary- and Multi-class Group Sparse Canonical Correlation Analysis for Feature Extraction and Classification. *IEEE Transactions on Knowledge and Data Engineering*. 2013; 25(10):2192–2205. <https://doi.org/10.1109/TKDE.2012.217>
13. Hu H, Lin Z, Feng J, Zhou J. Smooth Representation Clustering. In: *CVPR*; 2014. p. 3834–3841.
14. Nie F, Wang X, Huang H. Clustering and projected clustering with adaptive neighbors. In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; 2014. p. 977–986.
15. Gu Q, Zhou J. Learning the Shared Subspace for Multi-task Clustering and Transductive Transfer Classification. In: *ICDM*; 2009. p. 159–168.
16. Gui J, Sun Z, Jia W, Hu R, Lei Y, Ji S. Discriminant sparse neighborhood preserving embedding for face recognition. *Pattern Recognition*. 2012; 45(8):2884–2893. <https://doi.org/10.1016/j.patcog.2012.02.005>
17. Cao X, Zhang C, Fu H, Liu S, Zhang H. Diversity-induced Multi-view Subspace Clustering. In: *CVPR*; 2015. p. 586–594.
18. Gao H, Nie F, Li X, Huang H. Multi-view Subspace Clustering. In: *ICCV*; 2015. p. 4238–4246.
19. Zhang C, Fu H, Liu S, Liu G, Cao X. Low-Rank Tensor Constrained Multiview Subspace Clustering. In: *ICCV*; 2015. p. 1582–1590.
20. Nie F, Huang H, Cai X, Ding CHQ. Efficient and Robust Feature Selection via Joint $l_{2,1}$ -Norms Minimization. In: *NIPS*; 2010. p. 1813–1821.
21. Tao H, Hou C, Nie F, Jiao Y, Yi D. Effective Discriminative Feature Selection With Nontrivial Solution. *IEEE Transactions on Neural Networks and Learning Systems*. 2015; 27(4):3013–3017.
22. Zhang Z, Yan S, Zhao M. Pairwise sparsity preserving embedding for unsupervised subspace learning and classification. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*. 2013; 22(12):4640–51. <https://doi.org/10.1109/TIP.2013.2277780> PMID: 23955747
23. Ng AY, Jordan MI, Weiss Y. On Spectral Clustering: Analysis and an algorithm. *Proceedings of Advances in Neural Information Processing Systems*. 2002; 14:849–856.
24. Wu B, Liu Z, Wang S, Hu BG, Ji Q. Multi-label learning with missing labels. In: *ICPR*; 2014. p. 1964–1968.
25. He X, Niyogi P. Locality Preserving Projections (LPP). *Advances in Neural Information Processing Systems*. 2005; 45(1):186–197.
26. Nie F, Wang X, Jordan MI, Huang H. The Constrained Laplacian Rank Algorithm for Graph-Based Clustering. In: *AAAI*; 2016.
27. Zelnik-Manor L. Self-Tuning Spectral Clustering. *Advances in Neural Information Processing Systems*. 2004; 17:1601–1608.
28. Zhang Z, Chow TW, Zhao M. M-Isomap: Orthogonal Constrained Marginal Isomap for Nonlinear Dimensionality Reduction. *IEEE Transactions on Systems Man and Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man and Cybernetics Society*. 2012; 43(1):180–191.

29. Gui J, Jia W, Zhu L, Wang SL, Huang DS. Locality preserving discriminant projections for face and palmprint recognition. *Neurocomputing*. 2010; 73(13–15):2696–2707. <https://doi.org/10.1016/j.neucom.2010.04.017>
30. Kumar A, Rai P, Daum̃ĩc̃ ½ H. Co-regularized Multi-view Spectral Clustering. In: *NIPS*; 2011. p. 1413–1421.
31. Li Y, Nie F, Huang H, Huang J. Large-Scale Multi-View Spectral Clustering via Bipartite Graph. In: *AAAI*; 2015. p. 2750–2756.
32. Xia T, Tao D, Mei T, Zhang Y. Multiview Spectral Embedding. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*. 2010; 40(6):1438–46. <https://doi.org/10.1109/TSMCB.2009.2039566> PMID: 20172832
33. Cai X, Nie F, Huang H. Multi-View K-Means Clustering on Big Data. In: *IJCAI*; 2013.
34. Karasuyama M, Mamitsuka H. Multiple Graph Label Propagation by Sparse Integration. *IEEE Transactions on Neural Networks and Learning Systems*. 2012; 24(12):1999–2012. <https://doi.org/10.1109/TNNLS.2013.2271327>
35. Shen H, Tao D, Ma D. Multiview locally linear embedding for effective medical image retrieval. *Plos One*. 2013; 8(12):82409–82409. <https://doi.org/10.1371/journal.pone.0082409> PMID: 24349277
36. Bartels RH, Stewart GW. A solution of the matrix equation $AX+XB=C$. *Communications of the ACM*. 1972; 15(9):820–826. <https://doi.org/10.1145/361573.361582>
37. Nie F, Li J, Li X. Parameter-Free Auto-Weighted Multiple Graph Learning: A Framework for Multiview Clustering and Semi-supervised Classification. In: *IJCAI*; 2016.
38. Nie F, Cai G, Li X. Multi-view Clustering and Semi-supervised Classification with Adaptive Neighbours. In: *AAAI*; 2017.
39. Nie F, Huang H, Ding C. Low-rank matrix recovery via efficient Schatten p-norm minimization. In: *AAAI*; 2012. p. 655–661.
40. Gui J, Sun Z, Cheng J, Ji S. How to Estimate the Regularization Parameter for Spectral Regression Discriminant Analysis and its Kernel Version? *IEEE Transactions on Circuits and Systems for Video Technology*. 2014; 24(2):211–223. <https://doi.org/10.1109/TCSVT.2013.2273652>
41. Gu Q, Li Z, Han J. Learning a Kernel for Multi-Task Clustering. In: *AAAI*; 2011.
42. Hou C, Nie F, Li X, Yi D, Wu Y. Joint embedding learning and sparse regression: a framework for unsupervised feature selection. *IEEE Transactions on Cybernetics*. 2014; 44(6):793–804. <https://doi.org/10.1109/TCYB.2013.2272642> PMID: 23893760
43. Gui J, Sun Z, Ji S, Tao D. Feature Selection Based on Structured Sparsity: A Comprehensive Study. *IEEE Transactions on Neural Networks and Learning Systems*. 2016; p. 1–18. <https://doi.org/10.1109/TNNLS.2016.2551724> PMID: 27116754
44. Winn J, Jojic N. LOCUS: Learning Object Classes with Unsupervised Segmentation. In: *ICCV*; 2005. p. 756–763.
45. Li FF, Fergus R, Perona P. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. *Computer Vision and Image Understanding*. 2004; 106(1):178.
46. Sun S. Multi-view Laplacian Support Vector Machines. *Applied Intelligence*. 2011; 41(4):209–222. https://doi.org/10.1007/978-3-642-25856-5_16
47. Asuncion A, Newman DJ. UCI Machine Learning Repository Irvine. 2007.
48. Sindhwani V, Niyogi P, Belkin M. Beyond the point cloud: from transductive to semi-supervised learning. In: *ICML*; 2005. p. 824–831.
49. Yong J, Grauman K. Foreground Focus: Unsupervised Learning from Partially Matching Images. *International Journal of Computer Vision*. 2009; 85(2):143–166. <https://doi.org/10.1007/s11263-009-0252-y>
50. Dueck D, Frey BJ. Non-metric affinity propagation for unsupervised image categorization. In: *ICCV*; 2007. p. 1–8.