Pattern Recognition Letters xxx (xxxx) xxx

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Robust multi-view continuous subspace clustering ☆

Junbo Ma [a,b], Ruili Wang [a,*], Wanting Ji [a], Jiawei Zhao [a], Ming Zong [a], Andrew Gilman [a]

[a] *School of Natural and Computational Sciences, Massey University, Auckland, New Zealand*
[b] *College of Computer, National University of Defense Technology, Changsha, China*

ARTICLE INFO

Article history:
Available online xxx

Keywords:
Multi-view learning
Clustering
Common subspace
Representation
Big data analysis

ABSTRACT

This paper proposed a novel Robust Multi-View Continuous Subspace Clustering (RMVCSC) algorithm, which can untangle heavily mixed clusters by optimizing a single continuous objective. The proposed objective uses robust estimators to automatically clip specious inter-cluster connections while maintaining convincing intra-cluster correspondences in the common representation subspace learned from multiple views. The common representation subspace can reveal the underlying cluster structure in data. RMVCSC is optimized in an alternating minimization scheme, in which the clustering result and the common representation subspace are simultaneously optimized. Since different views can describe distinct perspectives of input data, the proposed algorithm has more accurate clustering performance than conventional algorithms by exploring information among multi-view data. In other words, the proposed algorithm optimizes a novel continuous objective in the simultaneously learned common representation subspace across multiple views. By using robust redescending estimators, the proposed algorithm is not prone to stick into bad local minima even with outliers in data. This kind of robust continuous clustering methods has never been used for multi-view clustering before. Moreover, the convergence of the proposed algorithm is theoretically proved, and the experimental results show that the proposed RMVCSC can outperform several very recent proposed algorithms in terms of clustering accuracy.

1. Introduction

Clustering is one of the main approaches for data mining and statistical analysis, which is the process of partitioning a data set into different subsets according to some defined measures. In real-world applications, multi-view data are obtained naturally, since data are often collected across multiple domains or extracted by different feature extractors. Multi-view data can be described in distinctive perspectives from each view. For example, a webpage can be described according to the contents of this webpage, the webpage contents linked to this webpage and the link structures used by this webpage, while an image can be described according to its colour, texture, shapes and so on. Thus, exploring information among multiple views to create accurate multi-view clustering algorithms is beneficial for big data analysis [1,2].

Multi-view clustering is a machine learning paradigm, which aims to leverage the complementary information among multiple views to improve the clustering accuracy and generalization ability [1,2]. There are mainly two approaches to do the multi-view clustering [2]. The first one is the fusion approach, which fuses similarity measurements from multiple views to construct a graph for clustering [4]. The other one is the subspace-clustering approach, which aims to learn a common latent subspace for all the multiple views [4–7]. Since the subspace approach can reveal the underlying cluster structure in multi-view data and achieves state-of-art performance [4], multi-view subspace clustering has attracted arising attention in the past years.

Multi-view subspace clustering performs clustering on a common subspace representation of all the views simultaneously with the assumption that all the views are generated from this latent subspace. Many multi-view subspace clustering methods have been developed in recent years [8–18], such as the iteration based methods [8,9], the factorization based methods [10,11], statistical approaches [12] and spectral clustering based approaches [13,14].

Although multi-view subspace clustering has permeated into many fields and has made a great performance, there are still some limitations. Especially, since most existing multi-view subspace clustering algorithms are based on k-means or spectral clustering, the number of clusters k and the weights of different views are required to be pre-set manually. This may limit the further advancement of multi-view subspace clustering.

More recently, Shah and Koltun [19] proposed a Robust Continuous Clustering (RCC) algorithm, which does not need to know the number of clusters in advance and has the ability to achieve

☆ **Conflict of interest.** None.

* Corresponding author at: Massey University, Auckland, New Zealand.
E-mail address: ruili.wang@massey.ac.nz (R. Wang).

https://doi.org/10.1016/j.patrec.2018.12.004
0167-8655/© 2018 Published by Elsevier B.V.

high accuracy efficiently even the data is in high-dimension. RCC optimized a clear continuous objective by using standard numerical methods. Thus, RCC naturally has the ability to be integrated into a dimensionality reduction system. However, it has not been integrated into a multi-view subspace clustering system yet.

In this paper, a novel Robust Multi-View Continuous Subspace Clustering (RMVCSC) algorithm is proposed to untangle heavily mixed clusters by optimizing a single continuous objective. We use the self-expressiveness property of multi-view data, which is proposed in [20] to learn a common representation subspace across multiple views. By using robust redescending estimators, the proposed algorithm is optimized in an alternating minimization scheme, in which the clustering result and the common representation subspace are simultaneously optimized. Without the requirement of the number of clusters given in advance, RMVCSC is not prone to stick into bad local minima even with outliers in data and is insensitive to initialization.

Over the iteration of the proposed algorithm, the representatives will move and merge into several discrete clusters. This kind of robust continuous clustering approaches has never been used for multi-view clustering before. The proposed RMVCSC algorithm can outperform several very recent proposed algorithms in terms of clustering accuracy.

The rest of this paper is organized as follows: Section 2 reviews the most related works; Section 3 formulates our proposed method; Section 4 introduces the optimization process; Section 5 analyses the convergence behaviour; experimental results and conclusions are in Sections 6 and 7.

2. Related work

In this section, we review the methods in multi-view subspace clustering and continuous clustering, which are the two most relevant topics to our proposed algorithm.

2.1. Multi-view subspace clustering

Multi-view subspace clustering aims to find the shared latent subspace for all the views of the data set and obtain the segments of the data set in this subspace [21]. As this subspace is jointly learned from all the views using the self-expression property of the data set, it can represent the data set and unveil the underlying cluster structure of the data set.

Currently, there are many multi-view subspace clustering algorithms have been proposed. Gao et al. [22] proposed a multi-view subspace clustering model which performs subspace clustering on each view and guarantee the consistency of the clustering structure among different views. Zhang et al. [23] performs clustering on multi-views simultaneously with a low-rank tensor constraint, which is constructed by the subspace representation matrices. Ding and Fu [24] proposed a multi-view subspace clustering algorithm via dual low-rank decompositions, which expect to find a low-dimensional view-invariant subspace for multi-view data. Fan et al. [25] proposed a localized multi-view subspace clustering model by fusing noiseless structures among views and samples. Zhuge et al. [4] proposed an auto-weighted multi-view subspace clustering algorithm based on common subspace representation matrix.

However, most of these works are based on the *k*-means algorithm and its variants.

2.2. Continuous clustering

Continuous clustering is another topic related to our proposed algorithm. The main idea of continuous clustering is to transform the clustering problem into a continuous optimization problem [3]. Lindsten et al. [26] proposed a formulation, which can relax k-means clustering to convex optimization problems. Hocking et al. [27] proposed a convex relaxation of hierarchical clustering in calculating continuous regularization. Chi and Lange [28] proposed a splitting method for solving convex clustering problem. Chi et al. [29] proposed a convex bi-clustering algorithm Convex BiclusteRing Algorithm (COBRA), which settles on a graph-based representation of both samples and features. All the methods mentioned above are regularized by convex function (l_2-norm).

In addition, Shah and Koltun [19] proposed Robust Continuous Clustering (RCC) algorithm, regularization using a non-convex function (Geman–McClure). RCC does not have the prior knowledge of the number of clusters, and it has the ability to achieve high accuracy efficiently even the data is in high-dimension. He and Moreira-Matias [3] proposed an optimization method Robust Continuous Co-Clustering (ROCCO), which formulated a co-clustering problem as a continuous non-convex optimization problem. ROCCO learns the representation regularized on both sample and feature graphs.

In this paper, a novel Robust Multi-View Continuous Subspace Clustering (RMVCSC) algorithm is proposed to untangle heavily mixed clusters by optimizing a single continuous objective. Specifically, the proposed algorithm optimizes a novel continuous objective in the simultaneously learned common representation subspace across multiple views. By using robust redescending estimators, the proposed algorithm is not prone to stick into bad local minima even with outliers in data. This kind of robust continuous clustering methods has never been used for multi-view clustering before.

3. Model formulation

In this section, we introduce the subspace representation and formulation of RMVCSC.

In multi-view clustering, the clustering results of different views should be consistent, which means the clustering assignments of all the views should be the same. As multi-view data is collected across different domains, different views may show a large divergence when learning a consensus representation. Thus, multi-view subspace representation is used in the proposed algorithm to learn a common view-invariant subspace while reducing the influence of view-variance.

Consider the problem of spectral-based subspace representation. The *i*th row, *j*th column and *ij*th element in a matrix M can be denoted as $m_{i:}$, $m_{:j}$, and m_{ij} respectively. Suppose single-view data matrix is $X=[x_{:1}, x_{:2}, \ldots, x_{:n}] \in \mathbb{R}^{d\times n}$ which includes n data points in d dimensions. If X presents multi-view data, the data matrix of X in the vth view can be denoted as $X^v \in \mathbb{R}^{dv\times n}$.

Thus, based on the self-expressiveness property, the data matrix of X in the vth view is represented as:

$$X^v = X^v Z + E^v. \tag{1}$$

where $Z=[z_{:1}, z_{:2}, \ldots, z_{:n}] \in \mathbb{R}^{n\times n}$ is the self-representation matrix, in which each $z_{:i}$ is the representative of data point $x_{:i}$, and E^v is an error matrix. The nonzero elements of $z_{:i}$ correspond to the data points from the same subspace.

Since the input data is denoted as $X^v \in \mathbb{R}^{dv\times n}$ and Z is the self-representation matrix of X^v, the new representatives are based on X to initialize. After that, all steps of RMVCSC are operated by the new representatives. Over times of iteration, the new representatives will migrate and merge into several discrete clusters.

The objective function of RMVCSC is defined as follows:

$$\Phi(Z) = \|X^v - X^v Z\|_{2,p}^p + \lambda\Omega^v(Z), \tag{2}$$

where $||\cdot||p_{2,p}$ is the sparsity-inducing norm with $0 \leqslant p \leqslant 1$; λ is a tradeoff factor, and $\Omega^v(Z)$ is a smooth regularizer on Z.

Please cite this article as: J. Ma, R. Wang and W. Ji et al., Robust multi-view continuous subspace clustering, Pattern Recognition Letters, https://doi.org/10.1016/j.patrec.2018.12.004

Given the representation error matrix $E^v \in \mathbb{R}^{dv\times n}$ of the vth view as:

$$E^v = X^v - X^v Z, \tag{3}$$

the $||\cdot||p_{2,p}$-norm of the representation error matrix E^v can be defined as:

$$\|E^v\|_{2,p}^p = \sum_{i=1}^{d^v}\left(\sum_{j=1}^{n}\left|e_{ij}^v\right|^2\right)^{\frac{p}{2}} = \sum_{i=1}^{d^v}\left(\left\|e_{i:}^v\right\|_2\right)^p, \tag{4}$$

where $e_{i:}^v$ is the ith row of E^v, and $||E^v||_{2,p}$ is a $\ell_{2,p}$-norm [30].

$\Omega^v(Z)$ aims to smooth the distribution of the common representation Z on the vth view. The common subspace representation matrix Z will be enforced to meet the grouping effect using $\Omega^v(Z)$.

Based on the original data X and the new representatives Z, a graph is constructed automatically by using m-kNN graphs, a variant of the standard kNN graphs [31]. Compared with standard kNN graphs, all vertices in an m-kNN graph have a k-upper bound, which helps the graph not to produce vertices (hub vertices) with an extremely high degree and is more robust for utilizing.

Specifically, for $v = 1, 2, \ldots, m$, each regularized term $\Omega^v(Z)$ in our proposed algorithm is defined as:

$$\Omega^v(Z) = \frac{1}{2}\sum_{(s,t)\in\varepsilon} w_{s,t}^v \rho\left(\|z_s - z_t\|_{2,p}^p\right). \tag{5}$$

Thus, the objective function of RMVCSC can be rewritten as:

$$\Phi(Z) = \sum_{v=1}^{m}\left(\sum_{i=1}^{n}\|x_i^v - x_i^v z_i\|_{2,p}^p + \frac{\lambda}{2}\sum_{(s,t)\in\varepsilon} w_{s,t}^v\rho\left(\|z_s - z_t\|_{2,p}^p\right)\right). \tag{6}$$

where (s, t) means there is a connection between data x_s and data x_t, and ε is the edge set of this graph; weights $w_{s,t}^v$ measure the strength of each data to the pairwise terms that it exist, and λ is used to measure the proportion of each objective term to the whole; function $\rho(\cdot)$ is a penalty on the regularization terms.

Due to our algorithm is based on the duality between robust estimation and line processes [19], an auxiliary variable $h_{s,t}^v$ is introduced for each connection $(s, t) \in \varepsilon$. Thus, a joint objective over the representatives Z and the line process $H = \{h_{s,t}^v\}$ is proposed:

$$\Phi(Z,H) = \sum_{v=1}^{m}\left(\sum_{i=1}^{n}\|x_i^v - x_i^v z_i\|_{2,p}^p + \frac{\lambda}{2}\sum_{(s,t)\in\varepsilon} w_{s,t}^v \mathrm{h}_{s,t}^v\|z_s - z_t\|_{2,p}^p + \Psi\left(g_{s,t}^v\right)\right), \tag{7}$$

where $\Psi(h_{s,t}^v)$ is a penalty on ignoring a connection (s,t), i.e., when the connection is active (i.e., $h_{s,t}^v \to 1$) and $\Psi(h_{s,t}^v)$ tends to zero; when the connection is disabled (i.e., $h_{s,t}^v \to 0$) and $\Psi(h_{s,t}^v)$ tends to one. Each robust estimator $\rho(\cdot)$ has its own corresponding penalty function $\Psi(\cdot)$ so that Eqs. (6) and (7) are equivalent with respect to the representatives of X. In other words, the same set of Z will be produced by optimizing either of these two objectives. Eq. (7) is based on the iteratively reweighted least squares [32], however, it is more flexible because of the explicit variables H and the additional terms defined by these variables. Although there are many different gradient-based methods can be used to optimize Eq. (7), the iterative solution of linear least-squares systems can achieve more efficient and scalable optimization.

Although RMVCSC can accommodate different estimators within the same computational efficiency framework, our presentations and experiments are all based on a well-known estimator: Geman–McClure estimator [33],

$$\rho(y) = \frac{\mu y^2}{\mu + y^2}, \tag{8}$$

where μ is a scale parameter. The corresponding penalty function that makes Eqs. (6) and (7) equivalent with respect to the representatives is:

$$\psi(h_{s,t}) = \mu\left(\sqrt{h_{s,t}} - 1\right)^2. \tag{9}$$

4. Optimization

Based on Eq. (7) we observe that (i) when the variable H is fixed, Eq. (7) will transform into a linear least-squares problem; (ii) when the variable Z is fixed, the decoupling of individual pairwise terms and the optimal value of each connection $h_{s,t}^v$ can be calculated independently. Based on these two conditions, the objective can be optimized by updating the variable sets Z and H alternatingly. As a block coordinate descent algorithm, this alternating minimization scheme provably converges.

The first step is fixing the variable H, updating the common subspace representation Z. When the variable H is fixed, Eq. (2) can be rewritten in a matrix form and obtain a simplified expression by solving the variable Z:

$$\min_{Z}\sum_{v=1}^{m}\|X^v - X^v Z\|_{2,p}^p + \lambda\Omega^v(Z). \tag{10}$$

Intuitively, there are no weight factors explicitly defined in Eq. (10), so that all different views are treated equally. Thus, the Lagrange function of Eq. (10) can be written as:

$$\sum_{v=1}^{m}\|X^v - X^v Z\|_{2,p}^p + \lambda Tr\left(ZL^vZ^T\right). \tag{11}$$

$L^v = D^v - Q^v$ is the Laplacian matrix, in which D^v is a diagonal matrix; Q^v measures the spatial closeness of the data points on vth view. $Q^v = [q_{11}, q_{12}, \ldots, q_{nn}] \in \mathbb{R}^{n\times n}$, where $q_{st} = q_{ts} = w_{s,t}^v \cdot h_{s,t}^v$ when $w_{s,t}^v$ and $h_{s,t}^v$ is nonzero, otherwise $q_{st} = q_{ts} = 0$.

Taking the derivative of Eq. (11) with respect to Z and setting the derivative to zero, we have:

$$\sum_{v=1}^{m}\frac{\partial\left(\|X^v - X^v Z\|_{2,p}^p + \lambda Tr\left(ZL^vZ^T\right)\right)}{\partial Z} = 0. \tag{12}$$

In order to solve Eq. (12), we consider the following problem to tackle a non-smooth norm problem:

$$\min_{Z,U^v}\sum_{v=1}^{m}H^v + \lambda Tr\left(ZL^vZ^T\right), \tag{13}$$

where

$$H^v = Tr\left(\left(X^v - X^v Z\right)^T U^v\left(X^v - X^v Z\right)\right). \tag{14}$$

$U^v \in \mathbb{R}^{dv\times dv}$ is a diagonal matrix corresponding to the vth view. The ith entry on the diagonal is defined as:

$$u_{ii}^v = \frac{p}{2}\left\|e_{i:}^v\right\|_2^{p-2}, \forall i = 1, 2, \ldots, d^v. \tag{15}$$

Then differentiating the objective function with respect to Z and setting it to zero:

$$AZ + ZB + C = 0, \tag{16}$$

where

$$A = \sum_{v=1}^{m}X^{v^T}U^vX^v;$$

$$B = \lambda\sum_{v=1}^{m}L^v;$$

$$C = -\sum_{v=1}^{m}X^{v^T}U^vX^v. \tag{17}$$

Algorithm 1 RMVCSC.

I: Input: Data for m views $\{X^1, \ldots, X^m\}$ and $X^v \in \mathbb{R}^{dv \times n}$.
II: Output: Cluster assignment $\{\phi_i\}_{i=1}{}^n$.
III: Construct connectivity structure ε.
IV: Precompute $\chi^v = ||X^v||_2$, $w^v_{s,t}$, δ.
V: Initialize $h^v_{s,t} = 1$, $\mu >> \max||x_s - x_t||_2{}^2$, $\lambda = \Sigma^m(\chi^v/||L^v||_2)$.
VI: Initialize the feature weight matrix $U^v = I^v$ for each view, where $I^v \in \mathbb{R}^{dv \times dv}$ is the identity matrix.
VII: While $|\phi^t - \phi^{t-1}| < \varepsilon$ or $t <$ max-iterations do:
VIII: Compute the common representation Z by solving Eqs. (16) and (17)
IX Update the diagonal feature weight matrix U^v for each view by Eq. (15).
X: Update $h^v_{s,t}$ with Eq. (18) and $L^v = D^v - Q^v$.
XI: Every four itrations, update $\lambda = \Sigma^m(\chi^v/||L^v||_2)$ and $\mu = \max(\mu/2, \delta/2)$.
XII: End while
XIII: Construct graph $G = (V, F)$ with $f_{s,t} = 1$ if $||z_s{}^* - z_t{}^*||_2 < \delta$.
XIV: Output clusters are given by the connected components of G.

Eq. (17) is a standard Sylvester equation, which has a unique optimal solution.

As discussed in Section 3, the proposed objective function Eq. (7) is a joint objective over the representatives Z and the line process $H = \{h^v_{s,t}\}$. Thus, the second step is fixing the variable Z, then the optimal value of each connection $h^v_{s,t}$ is calculated by:

$$h^v_{s,t} = \left(\frac{\mu}{\mu + \|z_s - z_t\|_2} \right)^2. \tag{18}$$

According to the above two steps, we alternatively update Z and H, and repeat the process iteratively.

Algorithm 1 shows the whole process of RMVCSC. Note that all updates to Z and H optimize the same continuous global objective in Eq. (7). Step I and II are the input and output statements of the proposed algorithm. Step III to VI are the initialization steps, which are discussed in Section 3 above Eq. (5). Step VII to XII are the main optimization steps, which are discussed in details in Section 4. Step XIII and XIV are the output steps of the final clustering results.

5. Convergence analysis

In order to prove the convergence of our proposed algorithm and to prove the proposed algorithm can reach at least a locally optimal solution, we first introduce the following lemma [34].

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

$$a^p - \frac{p}{2}\frac{a^2}{b^{2-p}} \le b^p - \frac{p}{2}\frac{a^2}{b^{2-p}}. \tag{19}$$

The first step is fixing the variable H, updating the common subspace representation Z.

Theorem 1. *Each updated Z in Algorithm 1 will monotonically decrease the objective in Eq. (13) in each iteration.*

Proof: Denote $\tilde{Z}$as the updated Z in each iteration and $\tilde{E}^v = X^v - X^v\tilde{Z}$is the vth representation error matrix calculated by $\tilde{Z}$. According to the optimization to $\tilde{Z}$in Algorithm 1, $\tilde{Z}$ reaches the unique optimal solution of Eq. (10) when U^v are fixed. Thus,

$$\sum_{v=1}^{m}\left(Tr\left(\tilde{E}^{vT}U^v\tilde{E}^v\right) + \lambda Tr\left(\tilde{Z}L^v\tilde{Z}^T\right)\right) \le \sum_{v=1}^{m}\left(Tr\left(E^{vT}U^vE^v\right) + \lambda Tr\left(ZL^vZ^T\right)\right). \tag{20}$$

Combining weight matrix U^v which

$$u^v_{ii} = \frac{p}{2}\left\|e^v_{i:}\right\|_2^{p-2},$$

the inequation can be rewritten as:

$$\sum_{v=1}^{m}\left(\sum_{i=1}^{d^v}\frac{p}{2}\frac{\left\|\tilde{e}^v_{i:}\right\|_2^2}{\left\|e^v_{i:}\right\|_2^{2-p}} + \lambda Tr\left(\tilde{Z}L^v\tilde{Z}^T\right)\right) \le \sum_{v=1}^{m}\left(\sum_{i=1}^{d^v}\frac{p}{2}\frac{\left\|e^v_{i:}\right\|_2^2}{\left\|e^v_{i:}\right\|_2^{2-p}} + \lambda Tr\left(ZL^vZ^T\right)\right). \tag{21}$$

Generally, $\|e^v_{i:}\|_2 > 0$and $\|\tilde{e}^v_{i:}\|_2 > 0$, the regularised $l_{2,p}$-norm can be used to guarantee it. According to Lemma 1, we can derive

$$\left\|\tilde{e}^v_{i:}\right\|_2^p - \frac{p}{2}\frac{\left\|\tilde{e}^v_{i:}\right\|_2^2}{\left\|e^v_{i:}\right\|_2^{2-p}} \le \left\|e^v_{i:}\right\|_2^p - \frac{p}{2}\frac{\left\|e^v_{i:}\right\|_2^2}{\left\|e^v_{i:}\right\|_2^{2-p}}. \tag{22}$$

Thus, the following inequality holds

$$\sum_{v=1}^{m}\sum_{i=1}^{d^v}\left\|\tilde{e}^v_{i:}\right\|_2^p - \sum_{v=1}^{m}\sum_{i=1}^{d^v}\frac{p}{2}\frac{\left\|\tilde{e}^v_{i:}\right\|_2^2}{\left\|e^v_{i:}\right\|_2^{2-p}} \le \sum_{v=1}^{m}\sum_{i=1}^{d^v}\left\|e^v_{i:}\right\|_2^p - \sum_{v=1}^{m}\sum_{i=1}^{d^v}\frac{p}{2}\frac{\left\|e^v_{i:}\right\|_2^2}{\left\|e^v_{i:}\right\|_2^{2-p}}. \tag{23}$$

Summing Eqs. (20) and (22), we have

$$\sum_{v=1}^{m}\left(\left\|X^v - X^v\tilde{Z}\right\|_{2,p}^p + \lambda Tr\left(\tilde{Z}L^v\tilde{Z}^T\right)\right) \le \sum_{v=1}^{m}\left(\|X^v - X^vZ\|_{2,p}^p + \lambda Tr\left(ZL^vZ^T\right)\right). \tag{24}$$

Thus, each updated Z will monotonically decrease $\Phi(Z, H)$ in each iteration, which means the following inequality holds:

$$\min \Phi\left(\tilde{Z}, H\right) \le \min \Phi(Z, H).$$

The second step is fixing the variable Z, updating the variable H.

Since Eq. (18) is the optimal solution of H, this step will have a unique optimal solution H decreasing our objective function $\Phi(Z, H)$. Moreover, the second-order partial derivatives of $\Phi(Z, H)$ with respect to H is greater than zero. Thus, it is obvious that each updated H will monotonically decrease $\Phi(Z, H)$ in each iteration, which means the following inequality holds:

$$\min \Phi\left(\tilde{Z}, \tilde{H}\right) \le \min \Phi\left(\tilde{Z}, H\right).$$

To sum up, the alternately updated Z and H in Algorithm 1 can monotonically decrease the objective function in the iteration process.

$$\min \Phi\left(\tilde{Z}, \tilde{H}\right) \le \min \Phi\left(\tilde{Z}, H\right) \le \min \Phi(Z, H).$$

Please cite this article as: J. Ma, R. Wang and W. Ji et al., Robust multi-view continuous subspace clustering, Pattern Recognition Letters, https://doi.org/10.1016/j.patrec.2018.12.004

Table 1
Details of the multi-view datasets (view type (dimensionality)).

View type	WebKB	Caltech101-7	Digit
1	Fulltext (2949)	LBP (256)	FOU(76)
2	Inlinks (334)	PHOG (680)	FAC(216)
3	–	GIST (512)	KAR(64)
4	–	Gabor (32)	PIX(240)
5	–	SURF (200)	ZER(47)
6	–	SIFT (200)	MOR(6)
Data points	1051	441	2000
Classes	2	7	10

Table 2
Clustering results of different methods on the Cal-tech101-7 dataset (mean(±std)).

Approach	ACC	NMI	Purity
RMKMC	0.6034 (±0.0680)	0.5488 (±0.0482)	0.6846 (±0.0541)
PC-SPC	0.6975 (±0.0499)	0.6547 (±0.0262)	0.7581 (±0.0288)
CC-SPC	0.7047 (±0.0654)	0.6879 (±0.0378)	0.7972 (±0.0389)
MVSC	0.6034 (±0.0309)	0.4766 (±0.0373)	0.6559 (±0.0314)
DiMSC	0.7312 (±0.0244)	0.6458 (±0.0179)	0.7698 (±0.0268)
RAMSC	0.7384 (±0.0082)	0.7276 (±0.0080)	0.8258 (±0.0115)
RMVCSC	0.7360 (±0.0078)	0.7631 (±0.0075)	0.8912 (±0.0112)

Table 3
Clustering results of different methods on WebKB dataset (mean(±std)).

Approach	ACC	NMI	Purity
RMKMC	0.8049 (±0.0000)	0.1592 (±0.0000)	0.8159 (±0.0000)
PC-SPC	0.7659 (±0.0000)	0.0991 (±0.0000)	0.7812 (±0.0000)
CC-SPC	0.5785 (±0.0000)	0.0019 (±0.0000)	0.7812 (±0.0000)
MVSC	0.7802 (±0.0000)	0.0041 (±0.0000)	0.7812 (±0.0000)
DiMSC	0.6147 (±0.0000)	0.0006 (±0.0000)	0.7812 (±0.0000)
RAMSC	0.9401 (±0.0000)	0.5689 (±0.0000)	0.9401 (±0.0000)
RMVCSC	0.9402 (±0.0000)	0.5694 (±0.0000)	0.9420 (±0.0000)

6. Experiments

In this section, we evaluate the RMVCSC algorithm and several reference algorithms on three widely used datasets. Experimental results show their convergence behaviour.

6.1. Dataset descriptions

There are three multi-view benchmark datasets, which are commonly used for multi-view learning, have been used to validate the effectiveness of RMVCSC. They are Caltech101 [35], Handwritten Dutch Digit Recognition (Digit) [36] and Web Knowledge Base (WebKB) [37]. The statistics information of these five datasets is concluded in Table 1.

All the experiments are following the 5-fold cross-validation scheme. Each dataset is randomly split into five subsets equally. Each clustering methods is tested on a selected subset and trained on the rest of the subsets. The final results are reported as the average of these 5 clustering results.

Caltech101-7 dataset is composed of 8677 objective images, which belong to 101 categories. We selected seven widely used classes, including DollaBill, Faces, Garfield, Motorbikes, Snoopy, Stop-Sign and Windsor-Chair. Following Tzortzis and Likas [38], which 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.

WebKB dataset is a subset of web documents from four universities. This dataset consists 1051 pages, which are classified 2 classes: 230 Course pages and 821 Non-Course pages. Each page has two 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.

Digit dataset contains 2000 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.

6.2. Experimental setup

In order to evaluate the performance of RMVCSC, we compared RMVCSC with several state-of-art approaches, which includes robust multi-view *k*-means clustering (RMKMC) [40], pairwised co-regularized multi-modal spectral clustering (PC-SPC) [41], centroid co-regularized multi-modal spectral clustering (CC-SPC) [42], multi-view subspace clustering (MVSC) [22], diversity induced multi-view subspace clustering (DiMSC) [43], and auto-weighted multi-view subspace clustering (RAMSC) [4].

RMKMC: RMKMC obtains 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: PC-SPC 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 the same as each other.

CC-SPC: Similar to PC-SPC, CC-SPC makes different views be the same as a common one based on a centroid-based co-regularization term.

MVSC: MVSC performs subspace clustering on individual modality respectively and then unify them with a common indicator matrix.

DiMSC: DiMSC learns subspace representations and employs the Hilbert-Schmidt Independence Criterion to enhance complementary information.

RAMSC: RAMSC is an auto-weighted multi-view subspace clustering algorithm based on common subspace representation matrix.

The source codes of these reference approaches are downloaded from the Internet. The best performances of these approaches are achieved according to these reference papers' setting. All the experiments are written in Matlab R2017a and processed on an HP Elite Desk 800 workstation with Intel i7-4790 CPU and 16GB RAM.

We normalize each view of the multi-view data firstly. All values of these input data will be in the range $[-1, 1]$ before clustering. The threshold δ is set to be the mean of the lengths of the shortest 1% of the edges in the edge set ε. The parameter μ is initially set to $\mu = 3r^2$, where r is the maximal edge length in the edge set ε. All experiments are repeated 5 times independently. We reported the mean and standard deviation of them as experimental results.

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

6.3. Experimental results

The experimental results on three datasets with three metrics are shown in Tables 2–4 respectively. The final representation produced by RMVCSC on Digit dataset is shown in Fig. 1. The convergence behaviours of RMVCSC on three datasets are illustrated in Fig. 2. In terms of clustering accuracy, we have the following conclusions.

From Tables 2–4, RMVCSC outperforms the reference approaches on all benchmark datasets. According to three different

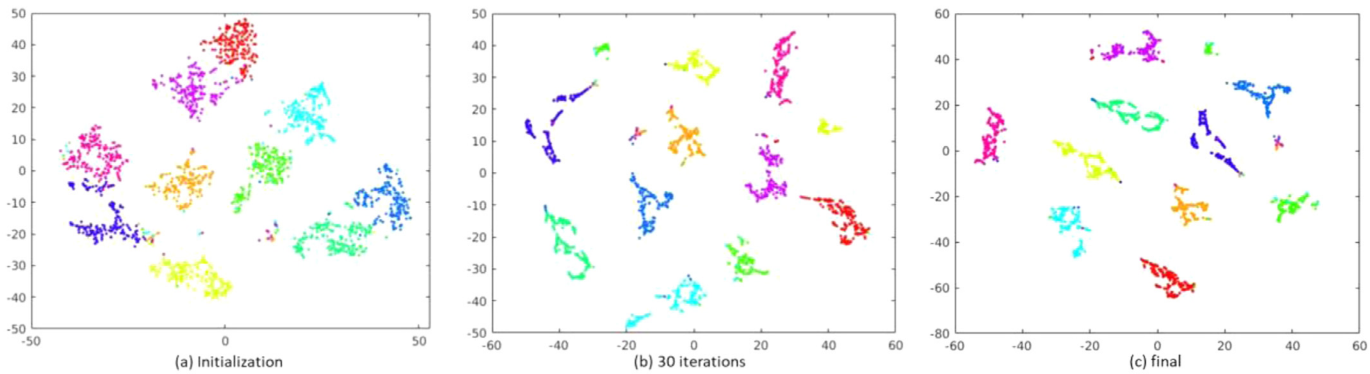

Fig. 1. The common representations produced by RMVCSC on Digit dataset. Each data point is labeled with its ground truth label. In the final output, RMVCSC identifies 10 large clusters with more than 100 instances and outliers. We use t-SNE [39] to visualize the representations discovered by the algorithm.

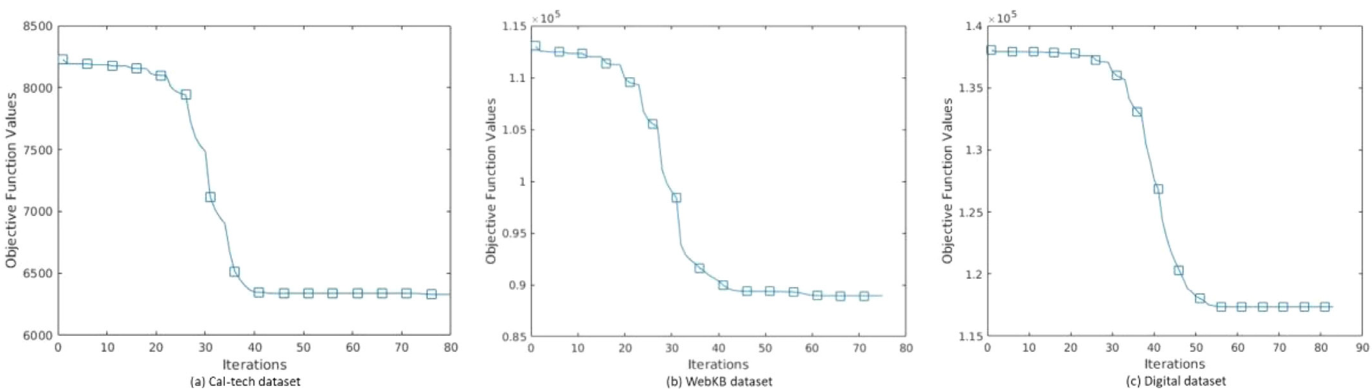

Fig. 2. Convergence behaviours of RMVCSC on three datasets. (a) Cal-tech101-7 dataset; (b) WebKB dataset; (c) Digit dataset.

Table 4
Clustering results of different methods on Digit dataset (mean(± std)).

Approach	ACC	NMI	Purity
RMKMC	0.7853 (±0.0800)	0.8125 (±0.0384)	0.8190 (±0.0614)
PC-SPC	0.8682 (±0.0604)	0.8267 (±0.0303)	0.8759 (±0.0500)
CC-SPC	0.8768 (±0.0605)	0.8234 (±0.0338)	0.8855 (±0.0471)
MVSC	0.8242 (±0.0686)	0.8399 (±0.0355)	0.8286 (±0.0664)
DiMSC	0.8400 (±0.0569)	0.8076 (±0.0347)	0.8465 (±0.0518)
RAMSC	0.9299 (±0.0439)	0.8864 (±0.0199)	0.9343 (±0.0333)
RMVCSC	0.9312 (±0.0245)	0.8867 (±0.0123)	0.9962 (±0.0323)

evaluation metrics: ACC, NMI, and Purity, our proposed algorithm can achieve a better or at least comparable performance.

As shown in Tables 2–4, the previous multi-view clustering methods cannot always achieve better performances. This may be because the previous methods characterize the structure of each view data separately and combine them with naïve addition operations, which makes the final clustering result affected by these inaccurate structures. RMVCSC can perform better results in most cases due to our proposed algorithm assigns small weight factors to the inaccurate view and learns a common self-expressiveness matrix Z among different views.

Tables 2 and 3 also show the robustness of RMVCSC. Our proposed algorithm learns view weight factors without an extra parameter and uses the $\ell_{2,p}$-norm to eliminate the effects of inaccurate functions.

7. Conclusion

In this paper, we have proposed a novel robust multi-view continuous subspace clustering algorithm, named RMVCSC. Compared with recently proposed multi-view subspace clustering algorithms, it can achieve higher clustering accuracy across multiple views and is more robust for utilizing. We use the self-expressiveness to learn a common representation subspace across multiple views. By using robust redescending estimators, the proposed algorithm is optimized in an alternating minimization scheme, in which the clustering result and the common representation subspace are simultaneously optimized. Moreover, the convergence of the proposed algorithm is theoretically proved rigorously. Compared with several very recently algorithms, RMVCSC has more accurate clustering performance without pre-setting the number of clusters.

Acknowledgement

This work was partially supported by the Marsden Fund, Royal Society, New Zealand.

References

[1] G. Chao, S. Sun, and J. Bi. A survey on multi-view clustering. 2017, arXiv:1712.06246.
[2] C. Xu, D. Tao, C. Xu, Multi-view self-paced learning for clustering, in: IJCAI, 2015, pp. 3974–3980.
[3] X. He, and L. Moreira-Matias. Robust continuous co-clustering. 2018, arXiv:1802.05036.
[4] W. Zhuge, C. Hou, Y. Jiao, J. Yue, H. Tao, D. Yi, Robust auto-weighted multi-view subspace clustering with common subspace representation matrix, PloS One 12 (5) (2017) e0176769.
[5] C. Zhang, Q. Hu, H. Fu, P. Zhu, X. Cao, Latent multi-view subspace clustering, in: CVPR, 2017, pp. 4279–4287.
[6] X. Wang, Z. Lei, X. Guo, C. Zhang, H. Shi, S. Li, Multi-view subspace clustering with intactness-aware similarity, Pattern Recognit. (2018).
[7] Y. Fan, J. Liang, R. He, B. Hu, and S. Lyu. Robust localized multi-view subspace clustering. 2017, arXiv:1705.07777.
[8] J. Ho, M. Yang, J. Lim, K. Lee, D. Kriegman, Clustering appearances of objects under varying illumination conditions, in: CVPR, 2003, p. 11.
[9] P. Tseng, Nearest q-flat to m points, J. Optim. Theory Appl. 105 (1) (2000) 249–252.

[10] J. Costeira, T. Kanade, A multibody factorization method for independently moving objects, Int. J. Comput. Vis. 29 (3) (1998) 159–179.
[11] K. Kanatani, Motion segmentation by subspace separation and model selection, in: ICCV, 2001, pp. 586–591.
[12] M. Tipping, C.M. Bishop, Mixtures of probabilistic principal component analyzers, Neural Comput. 11 (2) (1999) 443–482.
[13] J. Yan, M. Pollefeys, A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate, in: ECCV, 2006, pp. 94–106.
[14] A. Goh, R. Vidal, Segmenting motions of different types by unsupervised manifold clustering, in: CVPR, 2007, pp. 1–6.
[15] W. Zheng, X. Zhu, G. Wen, Y. Zhu, H. Yu, J. Gan, Unsupervised feature selection by self-paced learning regularization, Pattern Recognit. Lett. (2018). https://doi.org/10.1016/j.patrec.2018.06.029.
[16] X. Zhu, S. Zhang, Y. Li, J. Zhang, L. Yang, Y. Fang, Low-rank sparse subspace for spectral clustering, IEEE Trans. Knowl. Data Eng. (2018). https://doi.org/10.1109/TKDE.2018.2858782.
[17] X. Zhu, S. Zhang, R. Hu, Y. Zhu, Local and global structure preservation for robust unsupervised spectral feature selection, IEEE Trans. Knowl. Data Eng. 30 (3) (2018) 517–529.
[18] W. Zheng, X. Zhu, Y. Zhu, R. Hu, C. Lei, Dynamic graph learning for spectral feature selection, Multimedia Tools Appl. (2017) 1–17.
[19] S. Shah, V. Koltun, Robust continuous clustering, Proc. Natl. Acad. Sci. 114 (37) (2017) 9814–9819.
[20] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.
[21] M. Brbić, I. Kopriva, Multi-view low-rank sparse subspace clustering, Pattern Recognit. 73 (2018) 247–258.
[22] H. Gao, F. Nie, X. Li, H. Huang, Multi-view subspace clustering, in: ICCV, 2015, pp. 4238–4246.
[23] C. Zhang, H. Fu, S. Liu, G. Liu, X. Cao, Low-rank tensor constrained multiview subspace clustering, in: ICCV, 2015, pp. 1582–1590.
[24] Z. Ding, Y. Fu, Robust multi-view subspace learning through dual low-rank decompositions, AAAI (2016) 1181–1187.
[25] Y. Fan, J. Liang, R. He, B. Hu, and S. Lyu. Robust localized multi-view subspace clustering. 2017, arXiv:1705.07777.
[26] F. Lindsten, H. Ohlsson, L. Ljung, Just Relax and Come clustering!: A convexification of K-Means Clustering, Linköping University Electronic Press, 2011.
[27] T. Hocking, A. Joulin, F. Bach, J. Vert, Clusterpath an algorithm for clustering using convex fusion penalties, in: ICML, 2011, pp. 1–16.
[28] E. Chi, K. Lange, Splitting methods for convex clustering, J. Computat. Graph. Stat. 24 (4) (2015) 994–1013.
[29] E. Chi, G. Allen, R. Baraniuk, Convex biclustering, Biometrics 73 (1) (2017) 10–19.
[30] F. Nie, H. Huang, X. Cai, C. Ding, Efficient and robust feature selection via joint $\ell 2$, 1-norms minimization, in: NIPS, 2010, pp. 1813–1821.
[31] K. Ozaki, M. Shimbo, M. Komachi, Y. Matsumoto, Using the mutual k-nearest neighbor graphs for semi-supervised classification of natural language data, in: CNLL, 2011, pp. 154–162.
[32] P. Green, Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives, J. R. Stat. Soc. Series B (Methodological) (1984) 149–192.
[33] S. Geman, D. McClure, Statistical methods for tomographic image reconstruction, Bull. Int. Stat. Inst. 4 (1987) 5–21.
[34] H. Tao, C. Hou, F. Nie, Y. Jiao, D. Yi, Effective discriminative feature selection with nontrivial solution, IEEE Trans. Neural Netw. Learn. Syst. 27 (4) (2016) 796–808.
[35] F. Li, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories, Comput. Vis. Image Understand. 106 (1) (2007) 59–70.
[36] A. Asuncion, D. Newman, UCI Machine Learning Repository, 2007.
[37] V. Sindhwani, P. Niyogi, M. Belkin, Beyond the point cloud: from transductive to semi-supervised learning, in: ICML, 2005, pp. 824–831.
[38] G. Tzortzis, A. Likas, Multiple view clustering using a weighted combination of exemplar-based mixture models, IEEE Trans. Neural Netw. 21 (12) (2010) 1925–1938.
[39] L. Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.
[40] X. Cai, F. Nie, H. Huang, Multi-view k-means clustering on big data, in: IJCAI, 2013, pp. 2598–2604.
[41] A. Kumar, P. Rai, H. Daume, Co-regularized multi-view spectral clustering, in: NIPS, 2011, pp. 1413–1421.
[42] Y. Lee, K. Grauman, Foreground focus: unsupervised learning from partially matching images, Int. J. Comput. Vis. 85 (2) (2009) 143–166.
[43] X. Cao, C. Zhang, H. Fu, S. Liu, H. Zhang, Diversity-induced multi-view subspace clustering, in: CVPR, 2015, pp. 586–594.

Please cite this article as: J. Ma, R. Wang and W. Ji et al., Robust multi-view continuous subspace clustering, Pattern Recognition Letters, https://doi.org/10.1016/j.patrec.2018.12.004