Multiview Clustering via Adaptively Weighted Procrustes

Feiping Nie
School of Computer Science
Center for OPTIMAL
Northwestern Polytechnical University
Xi'an, China
feipingnie@gmail.com

Lai Tian
School of Computer Science
Center for OPTIMAL
Northwestern Polytechnical University
Xi'an, China
tianlai.cs@gmail.com

Xuelong Li
Center for OPTIMAL
Xi'an Institute of Optics and Precision Mechanics
Chinese Academy of Sciences
Xi'an, China
xuelong_li@opt.ac.cn

ABSTRACT

In this paper, we make a multiview extension of the spectral rotation technique raised in single view spectral clustering research. Since spectral rotation is closely related to the Procrustes Analysis for points matching, we point out that classical Procrustes Average approach can be used for multiview clustering. Besides, we show that direct applying Procrustes Average (PA) in multiview tasks may not be optimal theoretically and empirically, since it does not take the clustering capacity differences of different views into consideration. Other than that, we propose an Adaptively Weighted Procrustes (AWP) approach to overcome the aforementioned deficiency. Our new AWP weights views with their clustering capacities and forms a weighted Procrustes Average problem accordingly. The optimization algorithm to solve the new model is computational complexity analyzed and convergence guaranteed. Experiments on five real-world datasets demonstrate the effectiveness and efficiency of the new models.

CCS CONCEPTS

• **Theory of computation** → **Unsupervised learning and clustering**;

KEYWORDS

Clustering, Multiview Data, Procrustes Analysis

ACM Reference Format:
Feiping Nie, Lai Tian, and Xuelong Li. 2018. Multiview Clustering via Adaptively Weighted Procrustes. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3219819.3220049

1 INTRODUCTION

Recently, several methods to analyze multiview data have been proposed. These methods learn from data by considering the diversity of different views. We can easily obtain data with multiple views from multiple sources or from different feature subsets. For example, we can identify a person by face, fingerprint, signature or iris with information obtained from multiple sources and we can represent an image by its color or texture features, which can be seen as different feature subsets of the image. Performing clustering analysis on multiview data is an important problem. We refer the reader to [21] for a survey on multiview learning.

A direct way to integrate multiview data is to concatenate all the feature vectors into a long one and perform single view clustering algorithm on it. But this concatenation causes overfitting in the case of a small size of training sample and is not physically meaningful since each view has a specific statistical property [21]. Smarter strategies should be considered to fully exploit views of multiview data. Several works have been done on this. Many of them prefer to construct a graph for each view and then jointly utilize these graphs to learn a unified one. [9] proposed a co-regularized framework for multiview spectral clustering in which they coregularize the clustering hypotheses to make different graphs agree with each other. [3] proposed a graph-based multiview spectral learning model to integrate heterogeneous image features. [11] proposed to learn a unified graph constructed from multiview data where the Laplacian of the new graph is constrained. [20] addressed this problem by separating noise from each graph and learned a shared low-rank transition probability matrix, which will be the input to the standard Markov Chain method for clustering. Since clustering is an unsupervised learning task, it is unrealistic to tune additional hyper-parameters with cross-validation in practice. But conventional models in the literature usually have hyper-parameters to tune, which limits their applicabilities. Recently, some parameter-free models have been proposed, such as [12, 13]. But compared with conventional approaches, they may have unsatisfactory performance. Therefore, a multiview clustering model which is both parameter-free and has better performance is desirable.

A notable technique in spectral clustering literature is spectral rotation (SR) [8]. It is quite easy to see that SR is closely related to the Procrustes Analysis for shapes matching. Thus, it is straightforward to expect a multiview

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08...$15.00
https://doi.org/10.1145/3219819.3220049

extension of SR with the classical Procrustes Average (PA) [6] which is a technique to match multiple shapes. But we will show that direct applying Procrustes Average for multiview clustering may not be optimal, since PA doesn't take the clustering capacity differences of different views into consideration. To overcome this deficiency, we propose an Adaptively Weighted Procrustes (AWP) technique in Section 3 and develop an efficient optimization strategy to solve it in Section 4. Computational complexity analysis and convergence guarantee are provided in Section 5. We empirically evaluate our AWP on several real-world datasets and report the results in Section 6.

Here, we highlight the main contributions of this paper as follows

- We propose an extension of SR for multiview data with PA and show its deficiency. Furthermore, an Adaptively Weighted Procrustes technique is proposed to overcome this deficiency.
- The proposed PA and AWP are parameter-free clustering models, which makes them more applicable than other approaches in the literature.
- An efficient optimization strategy with complexity analysis and convergence guarantee to solve the AWP is provided.
- Experimental results show that both PA and AWP achieve better performance than conventional compared state-of-the-art approaches. Empirical comparisons also show the promising efficiencies of the proposed methods.

Notations: Throughout this paper, scalars, vectors and matrices are denoted by lowercase letters, boldface lowercase letters and uppercase letters, respectively; for a matrix $A = [a_{ij}] \in \mathbb{R}^{n\times n}$, A^T denotes the transpose of A, $Tr(A) = \sum_{i=1}^{n} a_{ii}$, $\|A\|_F = \sqrt{Tr(A^TA)}$; $\mathbf{1}$ denotes vector with all ones; $Ind \overset{\text{def}}{=} \{Y \in \{0,1\}^{n\times k} | Y\mathbf{1} = \mathbf{1}\}$ denotes the set of indicator matrices; $\triangle_n \overset{\text{def}}{=} \{x \in \mathbb{R}^n | x_i \geq 0, \mathbf{1}^Tx = 1\}$ is the probabilistic simplex; for multiview data, we use the superscript $x^{(i)}$ for the x of i-th view; when refer to the iterative rules, x denotes the old variable and x^+ or x_+ (depending on whether the variable already comes with a subscript) denotes the updated variable.

2 CLUSTERING VIA PROCRUSTES

In this section, we revisit the conventional spectral clustering configuration and introduce how Procrustes can be used for the recovery of the indicator matrix from spectral embedding. Then, we extend this technique to the multiview clustering case.

2.1 Spectral Clustering and Indicators Recovery

Suppose we have n data points and i-th of them is denoted by $\mathbf{x}_i \in \mathbb{R}^d$. We want to group these data points into k clusters. The first step of spectral clustering is constructing a weighted graph $S \in \mathbb{R}^{n\times n}$ according to the pairwise similarity of data. There are variant approaches to form the similarity graph, e.g., the Heat Kernel [22],

$$s_{ij} = \exp\left(-\frac{\|x_i - x_j\|_2^2}{t}\right),$$

where t is the inverse temperature parameter, a hyper-parameter modeling the locality, to tune.

Next, we obtain the spectral embedding of original data by solving an eigenvalue decomposition problem, that is

$$\begin{aligned} &\min_F Tr(F^TLF) \\ &s.t.\ F^TF = I, \end{aligned} \tag{1}$$

where $F \in \mathbb{R}^{n\times k}$ is the spectral embedding and $L \in \mathbb{R}^{n\times n}$ is the Laplacian of the graph S. The Laplacian of a graph can be chosen in different ways, e.g., combinational $L = D - S$, normalized $L_{nor} = I - D^{-1}S$ and symmetrically normalized $L_{sym} = I - D^{-\frac{1}{2}}SD^{-\frac{1}{2}}$, where $D \in \mathbb{R}^{n\times n}$ is a diagonal matrix and $d_{ii} = \sum_{j=1}^{n} s_{ij}$. We refer the reader to the [18] for a compressive tutorial on spectral clustering. The optimal F in Eq.(1) is the eigenvectors corresponding to the k smallest eigenvalues of L.

Finally, we should recover the indicators from the spectral embedding F. Notice that every row of F, denoted by $\mathbf{f}_i \in \mathbb{R}^k$, can be viewed as an embedding of original data $\mathbf{x}_i$. The conventional approach to recover indicators is to perform k-means on the spectral embedding space and return the clustering result of k-means as final indicators.

But there is a more geometric viewpoint to recover indicators from spectral embedding, which was proposed in [8]. The idea is that, when we have the ideal similarity graph, the spectral embedding space is exactly spanned by the column vectors of the indicator matrix $Y \in Ind$. To see it, note that in the ideal similarity graph, items belonging to the same cluster are linked together and any pair of items belonging to different clusters cannot be connected by edge, which indicates that the ideal similarity graph is k-connected. With proper repermutation of data points, the adjacency matrix of the ideal graph and its Laplacian will be block diagonal matrices. Since Laplacian of a single connected component always has a constant eigenvector, e.g. it is $\mathbf{1}$ for combinational Laplacian, it is easy to see that the eigenspace corresponding to the k smallest eigenvalues of the ideal graph Laplacian is spanned by the columns of Y. Please refer to [8] for more detailed description.

Though the spectral embedding F of the ideal graph is not necessarily exact Y, the columns of F form an orthonormal basis of the span$\{Y\}$, which inspired authors of [8] to recover the indicator matrix Y by estimating the rotation of F, that is

$$\begin{aligned} &\min_{Y,R} \|Y - FR\|_F^2 \\ &s.t.\ Y \in Ind, R^TR = I, \end{aligned} \tag{2}$$

where $R \in \mathbb{R}^{k\times k}$ is a rotation matrix.

In computer vision community, Eq.(2) is called Orthogonal Procrustes Analysis [6] which can be used to estimate the rotation distance between two set of points. Its solution will be provided in Lemma 1 in Section 4.1.

2.2 Extend to Multiview Clustering

For multiview data, it is straightforward to match all v spectral embeddings, $F^{(i)} \in \mathbb{R}^{n\times k}, i = 1\ldots v$, with an overall indicator matrix $Y \in Ind$ simultaneously, that is

$$\begin{aligned} \min_{Y,\{R^{(i)}\}_v} &\sum_{i=1}^{v} \|Y - F^{(i)}R^{(i)}\|_F^2 \\ s.t. \quad & Y \in Ind, \left(R^{(i)}\right)^T R^{(i)} = I, \forall i = 1\ldots v, \end{aligned} \tag{3}$$

which is closely related to the classical Procrustes Average (PA) [6] technique in the literature. PA is frequently used for matching multiple shapes by rotating them to form a unified *average*. It is reasonable to treat v spectral embeddings $\{F^{(i)}\}_v$ as v shapes which are generated by rotating the indicator matrix Y with different rotation matrices $\{R^{(i)}\}_v$. But we will show in the next section that this direct multiview extension is not optimal and then a more reasonable approach will be proposed.

3 ADAPTIVELY WEIGHTED PROCRUSTES

In this section, we first show the drawback of simple applying Procrustes Average for multiview clustering, which is the motivation of our adaptively weighted scheme. Then we propose an Adaptively Weighted Procrustes technique for multiview clustering.

3.1 Motivation

It is quite easy to see that PA in Eq.(3) ignores the capacity differences of different views.

Specifically, if for spectral embedding $F^{(i)}$ of the i-th view, the optimization problem

$$\begin{aligned} &\min_{Y,R} \|Y - F^{(i)}R\|_F^2 \\ &s.t.\ Y \in Ind, R^T R = I, \end{aligned}$$

has a relatively larger objective value than other views, we may conclude that the i-th view has a lower clustering capacity, since it has large residual when try all possible rotation matrix R to approximate Y with $F^{(i)}R$. But in Eq.(3), the PA model for multiview clustering, this capacity differences are neglected.

What is worse, Eq.(3) push the view which has lowest clustering capacity towards zero harder than the other views, which degrades the performance drastically. Intuitively, it is an overfitting-like behavior. To be precise, we have following theorem.

Theorem 1. *Suppose* $acc \overset{def}{=} 1 - \frac{1}{2n}\|\breve{Y} - Y\|_F^2$. *Then, we have*

$$acc \le 1 - \frac{\sqrt{2}}{4n}\left(\max_{1\le i\le v} \|\breve{Y} - F^{(i)}R^{(i)}\|_F^2 - \|Y - F^{(i)}R^{(i)}\|_F^2\right),$$

where $\breve{Y} \in Ind$ *is the* oracle *clustering indicator matrix.*

PA pushes the worst view hardest, which descents the right-hand side of inequality in Theorem 1 and causes the degradation of performance. Therefore, a weighted Procrustes scheme that can adaptively adjust the weights according to the clustering capacity of different views is desirable.

3.2 Adaptively Weighted Scheme

A trivial idea is that we substitute Eq.(3) with

$$\begin{aligned} \min_{Y,\{R^{(i)}\}_v} &\sum_{i=1}^{v} \frac{1}{p_i}\|Y - F^{(i)}R^{(i)}\|_F^2 \\ s.t. \quad & Y \in Ind, \left(R^{(i)}\right)^T R^{(i)} = I, \forall i = 1\ldots v, \end{aligned} \tag{4}$$

where the constant p_i is the optimal objective function value of the following problem:

$$p_i \overset{\text{def}}{=} \min_{Y\in Ind, R^T R = I} \|Y - F^{(i)}R\|_F. \tag{5}$$

The underlying idea of Eq.(4) is that we measure the clustering capacity with $\{p_i\}_v$ for each view individually and form a weighted PA accordingly. High clustering capacity view is given large weight in the objective function in Eq.(4).

But there is case that fails this trivial idea. Though the i-th view has a small p_i in Eq.(5), it does not desire a large weight $\frac{1}{p_i}$ if it has a large disagreement on Y with most of the other views.

To avoid disagreements, instead of evaluating capacities of views individually, we evaluate them with a unified indicator matrix Y simultaneously, which gives

$$\begin{aligned} Y_*, \{R_*^{(i)}\}_v = &\arg\min_{Y,\{R^{(i)}\}_v} \sum_{i=1}^{v} \frac{1}{p_i}\|Y - F^{(i)}R^{(i)}\|_F^2 \\ & s.t. \quad Y \in Ind, \left(R^{(i)}\right)^T R^{(i)} = I, \end{aligned} \tag{6}$$

where

$$p_i \overset{\text{def}}{=} \|Y_* - F^{(i)}R_*^{(i)}\|_F. \tag{7}$$

Problem in Eq.(6) seems weird since the constant weights $\{p_i\}_v$ depend on the optimal Y_* and $\{R_*^{(i)}\}_v$. But it is easy to see Eq.(6) is equivalent to the following problem:

$$\begin{aligned} \min_{Y,\{R^{(i)}\}_v} &\sum_{i=1}^{v} \|Y - F^{(i)}R^{(i)}\|_F \\ s.t. \quad & Y \in Ind, \left(R^{(i)}\right)^T R^{(i)} = I, \forall i = 1\ldots v, \end{aligned} \tag{8}$$

which is the proposed Adaptively Weighted Procrustes for multiview clustering.

4 OPTIMIZATION STRATEGY

To handle the objective function in Eq.(8), we make following observation:

Theorem 2. *Eq.(8) can be shown to be equivalent to following problem:*

$$\begin{aligned} &\min_{Y,\{R^{(i)}\}_v,\{p_i\}_v} \ \sum_{i=1}^{v} \frac{1}{p_i}\|Y - F^{(i)}R^{(i)}\|_F^2 \\ &s.t. \qquad Y \in Ind, \left(R^{(i)}\right)^T R^{(i)} = I, \mathbf{p} \in \triangle_v. \end{aligned} \tag{9}$$

Then, we apply an alternating minimization scheme to solve Eq.(9).

4.1 Update $R^{(i)}$ and fix others

Note that the objective function with respect to $\{R^{(i)}\}_v$ is additive and the constraints with respect to $\{R^{(i)}\}_v$ is separable. We can solve every $R^{(i)}$ individually, which is equivalent to the Orthogonal Procrustes Problem in Eq.(2). We have following closed-form solution for it.

Lemma 1. *[16] For problem*

$$\min_{R^T R = I} \|A - BR\|_F^2,$$

there is a closed-form solution of R, that is $R^* = UV^T$, where U, V is constituted by the left and right singular vectors of $B^T A$, respectively.

Thus, for $R^{(i)}$ in Eq.(9), it is updated according to

$$R_+^{(i)} := U^{(i)}\left(V^{(i)}\right)^T, \tag{10}$$

where $\left(F^{(i)}\right)^T Y = U^{(i)}\Sigma^{(i)}\left(V^{(i)}\right)^T$.

4.2 Update Y and fix others

Start from following identity:

$$\begin{aligned} &\sum_{i=1}^{v} \frac{1}{p_i}\|Y - F^{(i)}R^{(i)}\|_F^2 \\ &\overset{(a)}{=} \sum_{i=1}^{v} \frac{1}{p_i}\left(\|Y\|_F^2 + \|F^{(i)}R^{(i)}\|_F^2\right) - \sum_{i=1}^{v} \frac{2}{p_i} Tr(Y^T F^{(i)} R^{(i)}) \\ &\overset{(b)}{=} \sum_{i=1}^{v} \frac{n+k}{p_i} - 2Tr\left(Y^T\left(\sum_{i=1}^{v} \frac{F^{(i)}R^{(i)}}{p_i}\right)\right), \end{aligned} \tag{11}$$

where (a) is simply unfolding v squared norms and (b) uses facts that $Y \in Ind$, $\left(F^{(i)}\right)^T F^{(i)} = I$.

Thus, when variables are fixed except Y, it is easy to see Eq.(9) is equivalent to

$$\max_{Y \in Ind} Tr\left(Y^T\left(\sum_{i=1}^{v} \frac{F^{(i)}R^{(i)}}{p_i}\right)\right), \tag{12}$$

which has a closed-form solution, that is, $\forall i = 1 \dots n$,

$$y_{ij}^+ = \begin{cases} 1 & j = \arg\max\limits_k \left[\sum_{i=1}^{v} F^{(i)}R^{(i)}/p_i\right]_k \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

Algorithm 1 Optimization Algorithm of AWP

```
Input: $\{F^{(i)} \in \mathbb{R}^{n \times k}\}_v$, initial $Y \in \mathbb{R}^{n \times k}$
  $p_i = 1/v, \forall i = 1 \dots v$
  $R^{(i)} = I, \forall i = 1 \dots v$
  repeat
    update $\{R^{(i)}\}_v$ with Eq.(10)
    update $Y$ with Eq.(13)
    update $\{p_i\}_v$ with Eq.(16)
  until converge
Output: indicator matrix $Y \in \mathbb{R}^{n \times k}$
```

4.3 Update p_i and fix others

For convenience, denote $\|Y - F^{(i)}R^{(i)}\|_F$ by ϕ_i. When all variables are fixed except $\mathbf{p}$, original problem can be written as:

$$\min_{p_i \geq 0, \mathbf{1}^T\mathbf{p}=1} \sum_{i=1}^{v} \frac{\phi_i^2}{p_i}, \tag{14}$$

which combining with Cauchy-Schwarz inequality gives:

$$\sum_{i=1}^{v} \frac{\phi_i^2}{p_i} \overset{(a)}{=} \left(\sum_{i=1}^{v} \frac{\phi_i^2}{p_i}\right)\left(\sum_{i=1}^{v} p_i\right) \overset{(b)}{\geq} \left(\sum_{i=1}^{v} \phi_i\right)^2, \tag{15}$$

where (a) holds since $\mathbf{1}^T\mathbf{p} = 1$ and the equality in (b) holds when $\sqrt{p_i} \propto \frac{\phi_i}{\sqrt{p_i}}$. Since the right-hand side of Eq.(15) is constant, the optimal $\mathbf{p}$ in Eq.(14) is given by, $\forall i = 1 \dots v$,

$$p_i^+ := \frac{\phi_i}{\sum_{i=1}^{v} \phi_i}. \tag{16}$$

To make these update rules clear, we summarize the algorithm to solve Eq.(6) in Algorithm 1.

5 THEORETICAL ANALYSIS

In this section, we provide the computational complexity analysis and convergence guarantee of Algorithm 1.

5.1 Computational Complexity

In this subsection, we consider the computational complexity of Algorithm 1. To update $\{R^{(i)}\}_v$ with Eq.(10), we need $O(nk^2v)$ for matrix multiplication and $O(k^3v)$ for singular value decomposition. Since $v \ll n$, we pay $O(nk^2v)$ to apply Eq.(10). To update Y with Eq.(13), we need $O(nk^2v)$ for the construction of $\sum_{i=1}^{v} F^{(i)}R^{(i)}/p_i$ and a simple $O(nk)$ travel for the arg min operator. To update $\{p_i\}_v$ with Eq.(16), with the pre-calculated $\{F^{(i)}R^{(i)}\}_v$ in Eq.(13), we only need $O(nkv)$ for $\{p_i\}_v$.

Overall, we need $O(nk^2t)$ to solve Eq.(8) where t is the number f iterative steps. In Section 6, we empirically show that t is usually quite small and $t < 20$ is sufficient for AWP to converge in the most cases.

It is notable that it seems that the new AWP is linear scaled with respect to n from the complexity $O(nk^2t)$. But the spectral embedding $\{F^{(i)}\}_v$ still need $O(n^3v)$ to obtain, which is the fundamental computational bottleneck of spectral algorithms and is not the main concern of this paper. Nevertheless, our AWP is still very efficient, since it does not have to perform eigenvalue decomposition inside of the iterations. Other methods in the literature, e.g., SwMC, MVSC, MLAN, usually have to perform eigenvalue decomposition in every iterative step.

5.2 Convergence Guarantee

The convergence of Algorithm 1 is obvious. To see it, note that 1) all the three update rules in Algorithm 1 do not increase the objective function value; 2) the objective function in Eq.(8) is bounded from below. Thus the convergence can be derived. The main concern of this subsection is where the Algorithm 1 will converge to.

Since the problem we concerned is a non-convex mixed-integer programming problem, not much guarantee on convergence can be expected. But we still can have something to say by considering a continual relaxation of Eq.(9).

Lemma 2. *Following problem is a continual relaxation of original Eq.*(9)*.*

$$\begin{aligned} \max_{Y,\{R^{(i)}\}_v,\{p_i\}_v} \quad & \sum_{i=1}^{v} \frac{1}{p_i} Tr\left(Y^T F^{(i)} R^{(i)}\right) \\ s.t. \quad & y_i \in \triangle_k, \mathbf{p} \in \triangle_v, \left(R^{(i)}\right)^T R^{(i)} = I, \end{aligned} \tag{17}$$

where $y_i \in \mathbb{R}^k$ is the i-th row of Y.

This relaxation has several desirable properties

- Eq.(9) and Eq.(17) share the same global optimum.
- Every local optimum of Eq.(17) is a feasible solution for Eq.(9).

and we have following guarantee:

Theorem 3. *Algorithm 1 can be viewed as a coordinate minimization scheme for Eq.*(17) *and it will converge to a stationary point of Eq.*(17)*.*

We make following remark for the relaxation Eq.(17).

Remark 1. *Eq.*(17) *is somewhat meaningful on its own. Notice that Eq.*(17) *can be rewritten as*

$$\max_{\mathbf{y_i}\in\triangle_k,\mathbf{p}\in\triangle_v} \sum_{i=1}^{v} \frac{1}{p_i} \left(\sum_{j=1}^{k} \cos(\theta_j^{(i)}) \right), \tag{18}$$

where $\theta_j^{(i)}$ is the j-th canonical angle *[17] of span$\{Y\}$ and span$\{F^{(i)}\}$. Since $\cos(\theta_j)$ is monotony decrease function when $0 \le \theta_i \le \frac{\pi}{2}$, Eq.*(18) *can be viewed as searching for $Y \in$ Ind that has minimal weighted* basis distance *[17] between span$\{Y\}$ and span$\{F^{(i)}\}$, which indicates that our AWP may have relation with subspace learning methods.*

6 EXPERIMENTS

In this section, we construct graphs for different views with method proposed in [14]. Since different views of data may have different scales, some multiview algorithms may fail to show their full power for clustering when scale of views are different. The graph construction approach we used naturally gives normalized graph and only needs to set one parameter which is the number of neighbors. In our experiments, the standard clustering Accuracy (ACC) , Purity and Normalized Mutual Information (NMI) metrics are used to measure the clustering performance.

6.1 Experiments Setup

In our experiments, we compare the proposed PA and AWP with several state-of-the-art methods: Multi-view Learning with Adaptive Neighbors (MLAN) [11], Multi-View Spectral Clustering (MVSC) [3], Robust Multi-view Spectral Clustering (RMSC) [20], and Self-weighted Multiview Clustering (SwMC) [13].

All the models in our experiments are repeatedly run 20 times and the mean of results are reported. The number of neighbors used in graph construction is fixed to 20 for all methods in our experiments. For the parameter-free models, i.e., SwMC, PA, and AWP, we run them as they are. For MLAN, we search the number of adaptive neighbors with $5 : 5 : 50$ (the MATLAB semicolon, we follow [7] to use this notation). For MVSC, we search the γ with $1 : 0.2 : 10$. For RMSC, we search the λ from 10^{-5} to 10^5 in the logarithmic scale.

We conduct experiments on five real-world datasets: Caltech 101 [5], Handwritten Numbers [1], NUS WIDE [4], ORL Face [15], and MSRC v1 [19]. A detailed summarization of these datasets is in Table 1.

6.2 Real-world Datasets

The experimental results of performance comparison are summarized in Table 2 and Table 3. The best results are marked in bold face. We run SR on all single views and the best results are set as the baseline. We have following observations:

- The proposed PA and AWP achieve the top two performance among the compared state-of-the-art multiview clustering models, which validates the effectiveness of utilizing the Procrustes technique for multiview clustering.
- The performance of AWP is better than PA, which validates the effectiveness of the adaptively weighted scheme proposed in Section 3.

	MSRC		
	ACC	NMI	Purity
SR(best)	0.6584	0.6136	0.6122
MLAN	0.6810	0.6299	0.7333
MVSC	0.8571	0.7385	0.7543
RMSC	0.8348	0.6265	0.7434
SwMC	0.7095	0.7744	0.8576
PA	0.8905	0.7875	0.8905
AWP	**0.8952**	**0.8021**	**0.8952**

Table 3: Clustering Performance Comparison on M-SRC

View	Cal 101	HNums	NUS	ORL	MSRC
1	GABOR(48)	FOU(76)	CH(64)	GIST(512)	CM(24)
2	WM(40)	FAC(216)	CC(44)	LBP(59)	HOG(576)
3	CENT(254)	KAR(64)	EDH(73)	HOG(864)	GIST(512)
4	HOG(1984)	PIX(240)	WAV(128)	CENT(254)	LBP(256)
5	GIST(512)	ZER(47)	BCM(155)	-	CENT(254)
6	LBP(928)	MOR(6)	BOW(500)	-	-
Size	2386	2000	2400	400	210
Classes	20	10	12	40	7

Table 1: Statistics of Five Real-world Datasets

	Caltech 101			Handwritten Numbers			NUS WIDE			ORL Face		
	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
SR(best)	0.5567	0.3842	0.6028	0.8304	0.8634	0.8792	0.2073	0.0997	0.2055	0.6746	0.7598	0.7146
MLAN	0.5339	0.4716	0.6660	0.9530	0.9190	0.9530	0.2454	0.1199	0.2604	0.7775	0.8846	0.8250
MVSC	0.6211	0.5826	0.7293	0.8825	0.8627	0.8825	0.2221	0.1069	0.2408	0.7825	0.9059	0.8275
RMSC	0.4086	0.5154	0.7347	0.7720	0.7073	0.7720	0.2238	0.1013	0.2429	0.5875	0.7752	0.6275
SwMC	0.5735	0.5108	0.7213	0.8505	0.8760	0.8895	0.2426	0.1223	0.2392	0.7650	0.8989	0.8250
PA	0.6567	0.6601	0.7850	0.9580	0.9214	0.9580	0.2717	0.1386	0.2892	0.8000	0.9115	0.8305
AWP	**0.6639**	**0.6760**	**0.7909**	**0.9725**	**0.9356**	**0.9725**	**0.2838**	**0.1445**	**0.2904**	**0.8000**	**0.9142**	**0.8350**

Table 2: Clustering Performance Comparison on Caltech101, Handwritten Numbers, NUS WIDE and ORL Face

6.3 Computation Time and Convergence

Our newly proposed AWP is not only effective but also efficient. We record the run time when conducted performance comparison experiments and report the mean of results in Table 4. We underline the results of SC and treat them as baseline. The two fastest results are marked with bold face. All the codes in the experiments are implemented in MATLAB R2016b, and run on a Windows 10 machine with 3.20 GHz i5-3470 CPU, 16 GB main memory.

Several observations are made from Table 4:

- PA and AWP are faster than other compared methods except the baseline SR. The reason is that though PA and AWP are also iterative algorithms, they do not have to perform $O(n^3)$ eigenvalue decomposition inside of the iterative steps, which makes them more efficient than conventional approaches.
- PA is slightly faster than AWP. The reason is that PA does not have optimization variables $\{p_i\}_v$ and needs not the computation of Eq.(16).

Moreover, we record the value of objective function of Eq.(9) per iterative step and report the convergence curves in Figure 1. It is easy to see that Algorithm 1 converges in a few steps. Empirically, it usually needs less than 20 steps to converge.

7 CONCLUSION

In this paper, two new multiview clustering models named PA and AWP are proposed. PA is a direct application of Procrustes Average (PA) on multiview clustering task. AWP is a newly proposed Adaptively Weighted Procrustes approach for multiview clustering motivated by the deficiency of PA. Both PA and AWP are parameter-free, which makes them more applicable than conventional multiview clustering approaches. The proposed methods are validated on five real-world datasets. Experimental results show the new methods are not only effective but also efficient compared with other multiview clustering approaches in the literature.

8 ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China grant under numbers 61772427, 61751202, and 61761130079.

A PROOFS

A.1 Proof of Theorem 1

We make a lemma that will be used in the proof.

Lemma (A). *Suppose* $\sum_{i=1}^{n} a_i^2 = 1$*, we have*

$$|a_i - a_j| \leq \sqrt{2}, \forall i, j = 1 \ldots n.$$

Here comes the proof of Theorem 1.

Datasets	SR	MLAN	MVSC	RMSC	SwMC	PA	AWP
Caltech 101	1.734	34.515	66.197	346.493	228.452	**12.036**	**12.057**
Handwritten Numbers	0.924	20.122	38.719	187.697	100.071	**6.887**	**6.892**
NUS WIDE	2.055	39.668	55.574	312.770	115.521	**11.673**	**11.790**
ORL Face	0.049	0.549	0.333	2.345	1.726	**0.156**	**0.162**
MSRA	0.019	0.118	0.153	0.577	0.562	**0.053**	**0.058**

Table 4: Run Time (s) of Compared Methods on Five Datesets

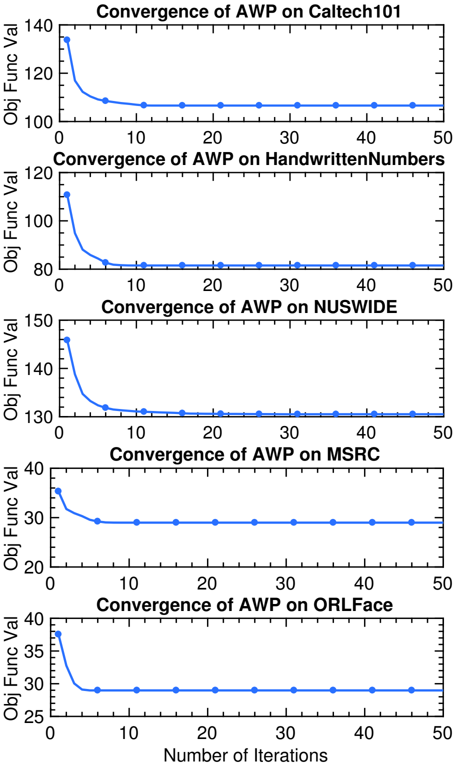

Figure 1: Convergence Curves of AWP on Caltech101, Handwritten Numbers, NUS WIDE, MSRC and ORL Face Datasets.

PROOF. Note that,

$$\begin{aligned}&\left\|\breve{Y}-F^{(i)}R^{(i)}\right\|_F^2-\left\|Y-F^{(i)}R^{(i)}\right\|_F^2\\=&\left\|\breve{Y}\right\|_F^2+\left\|F^{(i)}R^{(i)}\right\|_F^2-\|Y\|_F^2-\left\|F^{(i)}R^{(i)}\right\|_F^2\\&-2Tr\left(\left(\breve{Y}-Y\right)^TF^{(i)}R^{(i)}\right)\\\overset{(a)}{=}&2Tr\left(\left(Y-\breve{Y}\right)^TF^{(i)}R^{(i)}\right)\\\overset{(b)}{=}&2Tr\left(F^{(i)}R^{(i)}\left(Y-\breve{Y}\right)^T\right),\end{aligned} \tag{19}$$

where (a) holds by the constraint $Y\in Ind,\breve{Y}\in Ind$ and (b) holds by $Tr(AB)=Tr(BA)$.

Denote $F^{(i)}R^{(i)}$ by $H\in\mathbb{R}^{n\times k}$. Let the i-th row of H, Y and $\breve{Y}$ be $\mathbf{h}_i$, $\mathbf{y}_i$ and $\breve{\mathbf{y}}_i$, respectively. Above identity Eq.(19) can be written as

$$\begin{aligned}&2Tr\left(F^{(i)}R^{(i)}\left(Y-\breve{Y}\right)^T\right)\\=&2\sum_{i=1}^n\mathbf{h}_i^T(\mathbf{y}_i^T-\breve{\mathbf{y}}_i^T)\\\overset{(a)}{=}&2\sum_{i=1}^n\underbrace{\mathbf{h}_i^T\mathbf{y}_i^T}_{h_{ip}}-\underbrace{\mathbf{h}_i^T\breve{\mathbf{y}}_i^T}_{h_{iq}}\\=&2\sum_{\mathbf{y}_i\neq\breve{\mathbf{y}}_i}h_{ip}-h_{iq}\\\leq&2\sum_{\mathbf{y}_i\neq\breve{\mathbf{y}}_i}|h_{ip}-h_{iq}|\\\overset{(b)}{\leq}&2\sum_{\mathbf{y}_i\neq\breve{\mathbf{y}}_i}\sqrt{2}\\\overset{(c)}{=}&\sqrt{2}\|\breve{Y}-Y\|_F^2,\end{aligned} \tag{20}$$

where (a) holds since the indicator vector $\mathbf{y}_i$ $(\breve{\mathbf{y}}_i)$ only has one single non-zero element which is 1 and the index of this single non-zero element is denoted by p (q); (b) holds by the fact $\sum_{j=1}^k h_{ij}^2=1$ [10] and Lemma (A); (c) holds by the constraint $Y\in Ind,\breve{Y}\in Ind$.

Identity in Eq.(20) indicates that

$$\sqrt{2}\|\breve{Y}-Y\|_F^2\geq\left\|\breve{Y}-F^{(i)}R^{(i)}\right\|_F^2-\left\|Y-F^{(i)}R^{(i)}\right\|_F^2. \tag{21}$$

Applying Inequality (21) to all v views, we have

$$\|\breve{Y}-Y\|_F^2\geq\frac{\sqrt{2}}{2}\left(\max_{1\leq i\leq v}\left\|\breve{Y}-F^{(i)}R^{(i)}\right\|_F^2-\left\|Y-F^{(i)}R^{(i)}\right\|_F^2\right). \tag{22}$$

Denote $\left(\frac{1}{n}\sum_{\mathbf{y}_i\neq\breve{\mathbf{y}}_i}1\right)$ by acc. We have

$$acc=1-\frac{1}{2n}\|\breve{Y}-Y\|_F^2.$$

Therefore, we have

$$acc\leq1-\frac{\sqrt{2}}{2n}\left(\max_{1\leq i\leq v}\left\|\breve{Y}-F^{(i)}R^{(i)}\right\|_F^2-\left\|Y-F^{(i)}R^{(i)}\right\|_F^2\right), \tag{23}$$

which completes the proof.

□

A.2 Proof of Theorem 2

Proof. Proof is an analogue of Eq.(14).

Note that,

$$\begin{aligned}&\sum_{i=1}^{v}\frac{1}{p_i}\|Y-F^{(i)}R^{(i)}\|_F^2\\ \overset{(a)}{=}&\left(\sum_{i=1}^{v}\frac{1}{p_i}\|Y-F^{(i)}R^{(i)}\|_F^2\right)\left(\sum_{i=1}^{v}p_i\right)\\ \overset{(b)}{\geq}&\left(\sum_{i=1}^{v}\|Y-F^{(i)}R^{(i)}\|_F\right)^2,\end{aligned} \tag{24}$$

where (a) holds since $\sum_{i=1}^{v}p_i=1$ and (b) holds by the Cauchy-Schwarz inequality.

Eq.(24) indicates

$$\left(\sum_{i=1}^{v}\|Y-F^{(i)}R^{(i)}\|_F\right)^2=\min_{\mathbf{p}\in\triangle_v}\sum_{i=1}^{v}\frac{1}{p_i}\|Y-F^{(i)}R^{(i)}\|_F^2. \tag{25}$$

It is easy to see

$$\begin{aligned}&\min_{Y,\{R^{(i)}\}_v}\sum_{i=1}^{v}\|Y-F^{(i)}R^{(i)}\|_F\\ \Leftrightarrow&\min_{Y,\{R^{(i)}\}_v}\left(\sum_{i=1}^{v}\|Y-F^{(i)}R^{(i)}\|_F\right)^2\\ \Leftrightarrow&\min_{Y,\{R^{(i)}\}_v,\{p_i\}_v}\sum_{i=1}^{v}\frac{1}{p_i}\|Y-F^{(i)}R^{(i)}\|_F^2,\end{aligned} \tag{26}$$

which completes the proof. □

A.3 Proof of Theorem 3

Proof. We prove by showing that every point that Algorithm 1 converges to satisfies the KKT condition [2] of problem in Eq.(17).

The Lagrangian of Eq.(17) is

$$\begin{aligned}&\mathcal{L}(\mathbf{p},\{R^{(i)}\}_v,Y)\\ =&\sum_{i=1}^{v}\frac{1}{p_i}Tr\left(Y^TF^{(i)}R^{(i)}\right)\\ &+g_1(\mathbf{p},\lambda_1,\mu_1)+g_2\left(\{R^{(i)}\}_v,\lambda_2\right)+g_3(Y,\lambda_3,\mu_3)\end{aligned} \tag{27}$$

where g_1,g_2 and g_3 are corresponding the constraints.

The derivative of the Lagrangian in Eq.(27) is

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial p_i}&=-\frac{1}{p_i^2}Tr\left(Y^TF^{(i)}R^{(i)}\right)+\frac{\partial g_1(\mathbf{p},\lambda_1,\mu_1)}{\partial p_i}\\ \frac{\partial\mathcal{L}}{\partial R^{(i)}}&=\frac{\left(F^{(i)}\right)^TY}{p_i}+\frac{\partial g_2\left(\{R^{(i)}\}_v,\lambda_2\right)}{\partial R^{(i)}}\\ \frac{\partial\mathcal{L}}{\partial\mathbf{y}_i}&=\sum_{j=1}^{v}f_i^{(j)}R^{(j)}+\frac{\partial g_3(Y,\Lambda_3)}{\partial\mathbf{y}_i}\end{aligned} \tag{28}$$

We check the update rules of Algorithm 1, i.e., Eq.(10), Eq.(13), and Eq.(16), which are aimed to solve following problems

$$\min_{\left(R^{(i)}\right)^TR^{(i)}=I}\|Y-F^{(i)}R^{(i)}\|_F^2\Leftrightarrow\max_{\left(R^{(i)}\right)^TR^{(i)}=I}Tr\left(Y^TF^{(i)}R^{(i)}\right) \tag{29}$$

$$\begin{aligned}&\max_{Y\in Ind}Tr\left(Y^T\left(\sum_{i=1}^{v}\frac{F^{(i)}R^{(i)}}{p_i}\right)\right)\\ \Leftrightarrow&\max_{\mathbf{y}_i\in\triangle_k}Tr\left(Y^T\left(\sum_{i=1}^{v}\frac{F^{(i)}R^{(i)}}{p_i}\right)\right)\end{aligned} \tag{30}$$

$$\min_{\mathbf{p}\in\triangle_v}\sum_{i=1}^{v}\frac{1}{p_i}Tr\left(Y^TF^{(i)}R^{(i)}\right) \tag{31}$$

Since Eq.(10), Eq.(13), and Eq.(16) solve Eq.(29), Eq.(30) and Eq.(31), respectively. They satisfied the KKT conditions of Eq.(29), Eq.(30) and Eq.(31).

Write down the KKT conditions of Eq.(29), Eq.(30) and Eq.(31) and check that they are equivalent to The KKT condition of Eq.(27), which completes the proof. □

A.4 Proofs of Technical Lemmas

A.4.1 Proof of Lemma 2.

Proof. Using the identity in Eq.(11), we have Eq.(9) is equivalent to

$$\max_{Y\in Ind,\{R^{(i)}\}_v,\{p_i\}_v}\sum_{i=1}^{v}\frac{1}{p_i}Tr\left(Y^TF^{(i)}R^{(i)}\right) \tag{32}$$

The relaxation is that we substitute the constraint $Y\in Ind$ by $Y\in\{X\in[0,1]^{n\times k}|X\mathbf{1}=\mathbf{1}\}$. You should check the definition of Ind in Notations to see why we call this relaxation a continual relaxation.

Note that, the problem of Eq.(32) is a linear programming problem with respect to Y and its optimum is always a vertex of the probabilistic simplex. □

A.4.2 Proof of Lemma (A).

Proof. Note that,

$$\begin{aligned}(a_i-a_j)^2=&a_i^2+a_j^2-2a_ia_j\\ =&2a_i^2+2a_j^2-(a_i+a_j)^2\\ \leq&2(a_i^2+a_j^2)\\ \leq&2\end{aligned} \tag{33}$$

Therefore,

$$|a_i-a_j|\leq\sqrt{2},\forall i,j=1\dots n$$

which completes the proof. □

REFERENCES

[1] Arthur Asuncion and David Newman. 2007. UCI machine learning repository. (2007).

[2] Stephen Boyd and Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge university press.

[3] Xiao Cai, Feiping Nie, Heng Huang, and Farhad Kamangar. 2011. Heterogeneous image feature integration via multi-modal spectral clustering. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 1977–1984.

[4] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. 2009. NUS-WIDE: a real-world web image database from National University of Singapore. In *Proceedings of the ACM international conference on image and video retrieval*. ACM, 48.

[5] Li Fei-Fei, Rob Fergus, and Pietro Perona. 2007. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer vision and Image understanding* 106, 1 (2007), 59–70.

[6] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*. Vol. 1. Springer series in statistics New York.

[7] Gene H Golub and Charles F Van Loan. 2012. *Matrix computations*. Vol. 3. JHU Press.

[8] Jin Huang, Feiping Nie, and Heng Huang. 2013. Spectral Rotation versus K-Means in Spectral Clustering.. In *AAAI*.

[9] Abhishek Kumar, Piyush Rai, and Hal Daume. 2011. Co-regularized multi-view spectral clustering. In *Advances in neural information processing systems*. 1413–1421.

[10] Andrew Y Ng, Michael I Jordan, and Yair Weiss. 2002. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*. 849–856.

[11] Feiping Nie, Guohao Cai, and Xuelong Li. 2017. Multi-View Clustering and Semi-Supervised Classification with Adaptive Neighbours.. In *AAAI*. 2408–2414.

[12] Feiping Nie, Jing Li, and Xuelong Li. 2016. Parameter-Free Auto-Weighted Multiple Graph Learning: A Framework for Multiview Clustering and Semi-Supervised Classification.. In *IJCAI*. 1881–1887.

[13] Feiping Nie, Jing Li, and Xuelong Li. 2017. Self-weighted multiview clustering with multiple graphs. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. 2564–2570.

[14] Feiping Nie, Xiaoqian Wang, Michael I Jordan, and Heng Huang. 2016. The Constrained Laplacian Rank Algorithm for Graph-Based Clustering. In *Thirtieth AAAI Conference on Artificial Intelligence*. Citeseer.

[15] Ferdinando S Samaria and Andy C Harter. 1994. Parameterisation of a stochastic model for human face identification. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*. IEEE, 138–142.

[16] Peter H. Schonemann. 1966. A generalized solution of the orthogonal procrustes problem. *Psychometrika* 31, 1 (1966), 1–10.

[17] GW Stewart and Ji-Guang Sun. 1990. Matrix Perturbation Theory (Computer Science and Scientific Computing). (1990).

[18] Ulrike Von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and computing* 17, 4 (2007), 395–416.

[19] John Winn and Nebojsa Jojic. 2005. Locus: Learning object classes with unsupervised segmentation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, Vol. 1. IEEE, 756–763.

[20] Rongkai Xia, Yan Pan, Lei Du, and Jian Yin. 2014. Robust Multi-View Spectral Clustering via Low-Rank and Sparse Decomposition.. In *AAAI*. 2149–2155.

[21] Chang Xu, Dacheng Tao, and Chao Xu. 2013. A survey on multi-view learning. *arXiv preprint arXiv:1304.5634* (2013).

[22] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, Vol. 3. 912–919.