Correlation learning is a technique utilized to find a common representation in cross-domain and multiview datasets. However, most existing methods are not robust enough to handle noisy data. As such, the common representation matrix learned could be influenced easily by noisy samples inherent in different instances of the data. In this paper, we propose a novel correlation learning method based on a low-rank representation, which learns a common representation between two instances of data in a latent subspace. Specifically, we begin by learning a low-rank representation matrix and an orthogonal rotation matrix to handle the noisy samples in one instance of the data so that a second instance of the data can linearly reconstruct the low-rank representation. Our method then finds a similarity matrix that approximates the common low-rank representation matrix much better such that a rank constraint on the Laplacian matrix would reveal the clustering structure explicitly without any spectral postprocessing. Extensive experimental results on ORL, Yale, Coil-20, Caltech 101-20, and UCI digits datasets demonstrate that our method has superior performance than other state-of-the-art compared methods in six evaluation metrics.

1. Introduction

Cross-domain and multiview datasets can help to improve performance accuracy in a typical clustering problem, such as face image clustering [1–5] and object clustering [6–8]. The reason is that different instances of cross-domain and multiview datasets often share a common property that is useful in the overall clustering task [9, 10]. As such, several correlation learning techniques, including canonical correlation analysis (CCA) [11–15], cotraining [15–17], non-negative matrix factorization [18, 19], and subspace clustering [20–26] were utilized over the years to find a common representation in cross-domain and multiview datasets. For example, CCA [11] learns the linear relations between two sets of correlated variables with the nonlinear type proposed in [12]. As a result, the study in [13] is built on CCA, in which CCA is used to obtain a low-dimensional subspace spanned by a set of correlated but different datasets. In addition, Qin et al. [14] utilized cluster CCA to align domains in a projected correlated subspace. Besides, cotraining [16] is another classical method that maximizes the agreement between two distinct views of data to aid the learning process by adapting the knowledge gained from one view to the second view alternatively. Therefore, considering the successes of cotraining, Kumar and Daume [17] incorporated the idea of cotraining into multiview spectral clustering by learning a common clustering matrix that agrees across different views. Furthermore, Sun and Jin [15] proposed a method named robust cotraining, which combines the principles of cotraining and CCA, where CCA is used to analyze the labels predicted by cotraining on unlabeled samples. While the clustering methods based on CCA [13, 14] and cotraining [17] are validated to be efficient, their clustering performance may degrade with noisy data, which is ubiquitous in practice [27].

Recently, subspace clustering methods [20–24] have provided a more robust solution to tackle the above-mentioned shortcomings. These methods utilize either low-
Scientific Programming

rank representation (LRR) [28, 29] or a combination of LRR
and sparse subspace clustering (SSC) [30] to minimize the
discrepancy between cross-domain and multiview datasets
in a latent subspace. Specifically, these methods utilize self-
expressiveness property to select similar samples in the
original dataset to reconstruct each other to find a low-
dimensional representation. The above is possible because
high dimensional data are assumed to have corresponding
multiple low-dimensional subspaces and the concept be-
comes finding those low-dimensional subspaces together
with their cluster members [31]. That’s why, Xia et al. [20]
combined LRR and SSC techniques to learn a low-rank
transition matrix shared across multiple views, which is used
subsequently for clustering. Similarly, Ding and Fu and
Zhang et al. [22, 24] utilized the LRR technique to pursue a
domain-invariant representation that agrees across do-
 mains. Brbic and Kopriva [23] used the combination of LRR
and SSC to find a low-dimensional representation matrix for
each view. Then, using these low-dimensional subspace
structures, Brbic and Kopriva [23] can obtain a joint sim-
ilarity matrix which balances the agreement between the
different views. However, these methods mentioned above
find the common representation matrix shared across views
by assuming that similar samples in each view reside near
each other. While this assumption is possible, it can be
problematic in a practicable case where the data samples in
each view are distributed randomly with noise. Hence, the
changes will increase for two unrelated data samples to be
selected to reconstruct one another, and this limitation can
cause the learned common representation matrix to be faulty
in a manner capable of degrading clustering performance.
Therefore, the recent work in [26], which extends [25],
attempts to resolve the above issue by allowing a common
similarity matrix to approximate different view low em-
bedding matrices. However, the state-of-the-art method
fails to consider the proximity issue mentioned above in
constructing the individual view structure. Thus, it can cause
the propagation of noise in the different low embedding
matrices.

To this end, in this paper, we propose a novel correlation
learning method, which finds a common low-rank matrix
between two different instances of data in a latent subspace.
The core idea here is that we learn this common low-rank
matrix using one instance of data in a way that a second
instance can linearly reconstruct it. First, we consider a real-
world scenario where two instances of data will have dif-
ferent dimensions. Hence, we utilize matrix factorization
to align a low-rank matrix learned in one instance to another
through an orthogonal constraint. This approach allows the
low-rank matrix to become a common one shared between
the two domains. Besides, this approach avoids the prop-
agation of noise in different low-rank matrices. Therefore,
our method finds an ideal similarity matrix that approxi-
mates the common low-rank matrix much better, with a
rank constraint on the Laplacian matrix so that it reveals the
clustering structure explicitly without spectral clustering
postprocessing. Extensive experimental results on ORL,
Yale, Coil-20, Caltech 101-20, and UCI digits datasets
demonstrate that our method has superior performance
than other state-of-the-art compared methods in six eval-
uation metrics. That is, accuracy (ACC), normalized mutual
information (NMI), adjusted rand index (AR), F-score,
precision, and recall.

Our major contributions are summarized as follows:

(1) We propose a novel method based on LRR for
correlation learning between two instances of data.
Specifically, our method learns a common low-rank
matrix in one instance of data using matrix factor-
ization to align a second domain. This way, our
method can avoid the propagation of noise in dif-
ferent low-rank matrices that can cause a faulty
common matrix.

(2) Furthermore, our method can obtain a clustering
structure without performing spectral postproces-
sing of the low-dimensional embedding matrix. To
achieve this, we find an ideal similarity matrix that
best approximates the common low-rank matrix
such that a rank constraint on the Laplacian matrix
will reveal the clustering structure explicitly.

(3) Extensive experimental results on five benchmark
datasets demonstrate that our method has superior
performance than other state-of-the-art compared
methods in six evaluation metrics.

2. Related Work

The purpose of correlation learning in cross-domain and
multiview clustering tasks is to learn a common representation
that maximizes the agreement between different instances or
distribution of a given data to improve clustering performance
[17]. Many studies [27, 31–33] provide an extensive review of
the existing methods utilized previously to learn the correlation
between cross-domain and multiple view datasets. Among them
are two classical methods, namely, CCA [11] and cotraining
[16], which have inspired several other multiple instance-based
methods such as [13, 17]. While [13] is built on CCA principles,
[17] is based on cotraining in which a common structure is
learned in a low-dimensional subspace to reduce the dis-
agreement between distinct data views. Besides, the more recent
subspace clustering methods [22, 24, 34], built on LRR [28, 29],
provide some more robust approaches. These methods require
constructing a common similarity matrix to align the different
views or cross-domain datasets using the low-rank represen-
tation of each view where similar data samples are selected to
reconstruct themselves linearly [35]. For simplicity, assuming
we have a single view data matrix X with some noisy samples,
LRR can obtain a low-rank representation matrix using nuclear
norm regularization [36] as follows:

\[ \|U\|_* + \lambda \|E\|_{2,1}, \]

\[ \text{s.t.,} \quad X = XU + E, \tag{1} \]

where X denotes the self-dictionary, \(\|\cdot\|_*\) denotes the nuclear
norm, and \(E = X - XU\) denotes the error matrix. Upon
obtaining U in a multiview or cross-domain setting, the sub-
space clustering methods mentioned above can find a common
similarity matrix to balance the agreement between the different
Scientific Programming

U. Besides, some methods, such as [20, 23], combine both LRR and SSC [30] to learn a common similarity matrix.

Regardless, these subspace clustering methods mentioned above utilize a two-phase learning approach to obtain a clustering structure by applying spectral clustering [37] on the learned similarity matrix. In this case, the clustering performance can degrade when the similarity matrix produced in the first phase is faulty. Considering this drawback, Yang et al. [38] utilized block diagonal constraints to encourage a proper common matrix. Liang et al. [39] explored the diversity between domains to improve performance. Nonetheless, [38, 39] are equally two-phase methods. Moreover, the previous method [26] and the more recent one [40] can obtain a clustering structure directly without applying spectral clustering over the consensus similarity matrix. They both achieved that by imposing a rank constraint on the Laplacian matrix to guarantee that by imposing a rank constraint on the Laplacian matrix to ensure that all the entries of S are none negative. Let \( L_s \) is equal to the number of connected components in the Laplacian matrix because the constraint \( S \geq 0 \) normalizes \( S \) such that \( \sum_{i} s_{ij} = 1 \). Then, rank \( (L_s) = n - c \) allows \( S \) to become the clustering structure with an exact \( c \) number of connected components by avoiding spectral postprocessing of the low-dimensional embedding matrix through the following theorem [44, 45].

**Theorem 1.** If the similarity matrix \( S \) is nonnegative, the multiplicity \( C \) of the eigenvalue zero of the Laplacian matrix \( L_s \) is equal to the number of connected components in the graph that is associated with \( S \).

A detailed proof of Theorem 1 is given in preposition 2 of [45].

### 3. Proposed Method

In this section, we present our proposed low-rank correlation representation and clustering method. First, we formulate the common low-rank matrix and then incorporate clustering directly into our model.

#### 3.1. Model Formulation

Given two multi-instance datasets, X and Y of sizes \( n \times d_1 \) and \( n \times d_2 \), respectively, where \( d_1 \) and \( d_2 \) are the dimensions of the feature space and \( n \) is the number of samples, a naive approach would include learning different low-rank matrices for X and Y using equation (1) so that both matrices can be merged to obtain a common matrix. This approach, however, has a limitation. It will cause a fallible common matrix if any of the low-rank matrices is faulty. Therefore, we suppose that a common matrix can be obtained differently through one instance of the data since multiview data originate from one underlying latent subspace [41]. On the other hand, the above is tricky for two reasons. First, the dimensions of X and Y are different. Second, the redundant specific structure and common structure in X and Y are combined. To tackle this, we utilize matrix factorization [42, 43] to find a low-rank matrix \( U \) adaptively that captures the common structure in X and Y. At first, we obtain the common structure from Y as follows:

\[
\begin{align*}
\|Y - UP^T\|_F^2 + \|U\|_* \quad \text{s.t.} \quad P^T P = I, \\
\end{align*}
\]

where \( U \in \mathbb{R}^{n \times n} \) denotes a low-rank matrix and \( P \in \mathbb{R}^{d_2 \times n} \) is a factorized variable to make \( Y \approx UP^T \) possible with orthogonal constraint \( P^T P = I \). Then, \( U \) can be perceived to hold the common structure of X as well with the following model:

\[
\begin{align*}
\|U - XQ\|_F^2, \\
\text{s.t.} \quad Q^T Q = I,
\end{align*}
\]

where \( Q \in \mathbb{R}^{d_1 \times n} \) is similar to \( P \), and it is used here to align the common structure in the two domains with orthogonal constraint \( Q^T Q = I \) so that \( U \) can well capture an accurate manifold structure of X and Y through \( U \approx XQ \).

Upon learning \( U \), our method finds an ideal similarity matrix \( S = \{s_1, \ldots, s_n\} \in \mathbb{R}^{n \times n} \) by making \( S \) as much as possible. As such, we formulate our model as follows:

\[
\begin{align*}
\|Y - U P^T\|_F^2 + \|U - XQ\|_F^2 + \lambda_1 \|U\|_* + \lambda_2 \|S - U\|_F^2, \\
\text{s.t.} \quad P^T P = I, Q^T Q = I, S \geq 0, S1 = 1, \text{rank}(L_s) = n - c,
\end{align*}
\]

where \( S \geq 0 \) ensures that all the entries of \( S \) are none negative. Let \( L_s = 1 - S \) denotes the normalized Laplacian matrix because the constraint \( S \geq 0 \) normalizes \( S \) such that \( \sum_{i} s_{ij} = 1 \). Then, \( \text{rank}(L_s) = n - c \) allows \( S \) to become the clustering structure with an exact \( c \) number of connected components by avoiding spectral postprocessing of the low-dimensional embedding matrix through the following theorem [44, 45].

#### 3.2. Discussion

Here, we discuss the importance of matrix \( P \) and \( Q \), constrained to be column orthogonal using \( P^T P = I \) and \( Q^T Q = I \), respectively, in our model. It can be seen that in equation (2), the low-rank matrix \( U \) is learned from \( Y \) such that it captures the common structure between \( X \) and \( Y \). This is possible because we let \( Y \) differ from \( U \) through \( P^T \) and not by itself. Then, \( Q \) is used in equation (3) to match the common structure inside \( X \) (in original \( d_2 \) dimension) to \( U \) adaptively. For this, we suppose that the relative error of the term \( \|U - XQ\|_F^2 \) should be very small for \( U \) to capture the common structure in \( X \) and \( Y \) correctly. Therefore, the proposed method can guarantee more discriminative ability in practice, since \( U \) is learned adaptively using the nuclear norm, because with this norm, we can remove the effect of noise through soft thresholding [46], which sets small eigenvalues zero to suppress the noisy data. This approach is different from the existing ones where the data samples themselves are used as dictionaries to learn low-dimensional representation for each of \( X \) and \( Y \) before a corrected space can be found.

#### 3.3. Optimization

We propose an optimization method built on augmented Lagrange multipliers [47] to solve equation (4) by iteratively updating all variables. First, we
denote the \(i\)-th smallest eigenvalue of \(L_i\) as \(\theta_i(L_i)\) to make equation (4) easy to solve. Then, referring to Proposition 1 [45], \(\theta_i(L_i) \geq 0\) because \(L_i\) is a positive semidefinite matrix. Therefore, given a large value of \(\lambda_3\), equation (4) is the same as the following:

\[
\|Y - U P^T\|^2_F + \|U - X Q\|^2_F + \lambda_1 \|U\|_* + \lambda_2 \|S - U\|^2_F \\
+ \lambda_3 \sum_{i=1}^c \theta_i(L_i),
\]

s.t. \(P^T P = I, Q^T Q = I, S \geq 0, S1 = 1\).

Furthermore, when \(\lambda_3\) is large enough, \(\sum_{i=1}^c \theta_i(L_i) = 0\), so that the constraint rank \((L_i) = n - c\) will be satisfied. According to Ky Fan’s theorem [48], we understand that \(\sum_{i=1}^c \theta_i(L_i)\) is equivalent to minimizing \(\text{Tr}(F^T L_i F)\) subject to \(F^T F = I\). This is because an optimal \(F\) would contain the eigenvectors corresponding to \(c\) smallest eigenvalue of \(L_i\). Hence, for a better understanding of Ky Fan’s theorem, Zhan et al. [26] provided a simplified version as follows.

**Theorem 2.** Eigenvalues of \(L_i\) are ordered by \(0 \leq \theta_1(L_i) \leq \theta_2(L_i) \leq \cdots \leq \theta_c(L_i)\) and the corresponding eigenvectors are \(\varphi_1, \varphi_2, \ldots, \varphi_c\). The inequality \(\sum_{i=1}^c \theta_i(L_i) \leq \sum_{i=1}^c F_i^T L_i F_i\) holds for any orthogonal vectors \(F_1, F_2, \ldots, F_c\).

Thus, we rewrite equation (5) as

\[
\|Y - U P^T\|^2_F + \|U - X Q\|^2_F + \lambda_1 \|U\|_* + \lambda_2 \|S - U\|^2_F \\
+ \lambda_3 \text{Tr}(F^T L_i F),
\]

s.t. \(P^T P = I, Q^T Q = I, S \geq 0, S1 = 1\), \(F^T F = I\),

and also, we introduce an intermediate term \(J = U\) to make equation (6) easier to solve. Then, we get

\[
\|Y - U P^T\|^2_F + \|U - X Q\|^2_F + \lambda_1 \|U\|_* + \lambda_2 \|S - U\|^2_F \\
+ \lambda_3 \text{Tr}(F^T L_i F),
\]

s.t. \(P^T P = I, Q^T Q = I, U = J, S \geq 0, S1 = 1\), \(F^T F = I\).

The Lagrangian of equation (7) can be obtained as follows:

\[
L = \|Y - U P^T\|^2_F + \|U - X Q\|^2_F + \lambda_1 \|U\|_* + \lambda_2 \|S - U\|^2_F \\
+ \lambda_3 \text{Tr}(F^T L_i F) + M_1,
\]

\[
U - J + \frac{\mu}{2}\|U - J\|^2_F,
\]

where \(M_1\) is the Lagrange multiplier. Hence, we divide equation (8) into several subproblems and then update all subproblems by fixing the others in the following order.

**3.3.1. J Subproblem.**

\[
\min_j \frac{\lambda_1}{\mu} \|U\|_* + \frac{1}{2}\|\theta - (U + M_1/\mu)\|^2_F.
\]

We solve equation (9) by denoting the singular value decomposition (SVD) [46] of \(S^\mu [M] = T \mathcal{H}_\mu [\Sigma V^T]\) to obtain \(J\):

\[
J = S^\mu [\Sigma] (U + \frac{M_1}{\mu}),
\]

where \(\mathcal{H}_\mu [\Sigma] = \min (0, \Sigma + \mu, \max (0, \Sigma - \mu))\).

**3.3.2. P Subproblem.**

\[
\min_p \|Y - U P^T\|^2_F, \\
\text{s.t.} \quad P^T P = I.
\]

The orthogonal Procrustes problem in equation (11) is difficult to solve because the feasible set \(P\) satisfying \(P^T P = I\) is not convex. However, luckily, SVD [46] provides a method to solve this problem to find a unique solution. Therefore, we solve \(P\) by svd \((U^T Y)\) as follows:

\[
[U, \Sigma, V] = \text{svd}(U^T Y),
\]

\[
P = VU^T.
\]

**3.3.3. Q Subproblem.**

\[
\min_q \|U - X Q\|^2_F, \\
\text{s.t.} \quad Q^T Q = I.
\]

The Q subproblem can be solved the same way as P subproblem.

**3.3.4. U Subproblem.**

\[
\min_u \|Y - U P^T\|^2_F + U - X Q^2_F + \lambda_2 S - U^2_F + M_1, \\
U - J + \frac{\mu}{2}\|U - J\|^2_F.
\]

Setting the derivatives \((\partial / \partial U) = 0\), we get

\[
U = (2 Y P + 2 X Q + 2 \lambda_2 S - M_1 + \mu) / (2 + 2 \lambda_2 + \mu) I + P^T P)^{-1}.
\]

**3.3.5. S Subproblem.**

\[
\min_s \lambda_2 \|S - U\|^2_F + \lambda_3 \text{Tr}(F^T L_i F), \\
\text{s.t.} \quad S \geq 0, S1 = 1, F^T F = I.
\]
Since $F$ in equation (16) contains the eigenvectors corresponding to $c$ smallest eigenvalue of $L_s$, we rewrite equation (16) as

$$
\min_{s_j} \left( \lambda_2 \|f_i - f_j\|_2^2 - \lambda_2 u_{ij} \right) s_j + \lambda_2 s_j^T s_j,
$$

s.t. $s_j \geq 0, 1^T s_j = 1.$

Then, denoting $\lambda_2 \|f_i - f_j\|_2^2 - \lambda_2 u_{ij}$ as $p_{ij}$, equation (17) can be rewritten as

$$
\min_{s_j} \lambda_2 \|s_j^2 + \frac{p_{ij}}{2\lambda_2}\|_2^2,
$$

s.t. $s_j \geq 0, 1^T s_j = 1.$

Similar to [26], the optimal solution $s_j$ ist

$$
s_j = \left(-\frac{p_{ij}}{2\lambda_2 + \eta 1}\right)_+.
$$

### 3.4 Complexity Analysis

The computational complexity of our proposed method illustrated in Algorithm 1 is determined mainly by nuclear norm calculation in equation (10), matrix inverse and multiplication in equation (13), and Euclidean projection onto the simplex space in equation (19). The cost of equation (8) is $O(n^3)$, while the inverse of an $n \times n$ matrix in equation (15) consumes $O(n^3)$. The time complexity for matrix multiplication is $O(n^3)$. However, since there are several multiplications in equation (15), the overall multiplication cost will be $(k + 1) O(n^3)$, which is not ignorable when the number of data samples is large. Moreover, the Euclidean projection onto the simplex space in equation (19) takes $O(n)$.

### 4. Experiments

In this section, we conduct an extensive experiment to demonstrate effectiveness of our proposed method on five benchmark datasets described in Section 4.1. Accordingly, we compare our method with seven state-of-the-art methods, namely, MLRSSC [23], RMSL [22], MCGC [26], DiMSC [21], MVGL [25], GMC [40], and SM2SC [38].

#### 4.1. Datasets

We perform experiments on ORL and Yale, Coil-20 and Caltech 101-20, and UCI digits datasets to demonstrate the superiority of our method on face image clustering, object image clustering, and handwritten digit, respectively. For each dataset, we select two types of features to represent $X$ and $Y$. Specifically, we extract LBP together with Gabor features for ORL, Yale, Coil-20, and Caltech 101-20, and we select Fourier coefficients of the character shape (FOU) and profile correlation (FAC) features for UCI digits. Therefore, the following is a detailed description of each dataset, while Table 1 and Figure 1 provide a summary and pictures of the datasets, respectively.

1. ORL: this face dataset contains 400 images of 40 distinct subjects. Each subject represents different images taken under various conditions such as time, light, and facial expressions. Note that the dimensions of LBP and Gabor features extracted from this dataset are 3304 and 6750, respectively.
2. Yale: this dataset contains 165 gray-scale images of 15 individuals with 11 images per individual. The variations of this dataset include the images taken, with center light, with glasses, under happy mood, with left light, without glasses, etc. Here, the feature dimension of LBP and Gabor is the same as that extracted for ORL above.
3. COIL-20: this dataset is from the Columbia Object Image Library, and there are 1440 images of 20 different classes. Each class contains 72 images. The LBP and Gabor features of this dataset have 944 and 4096 dimensions, respectively.
4. Caltech 101-20: his image dataset has 101 categories of images. We select the widely used twenty classes to obtain 2368 images each for 48-D Gabor and 928-D LBP features.
5. UCI digits: this dataset denotes handwritten digits of 0 to 9 from the UCI machine learning repository. It is composed of 2000 data samples. The two features extracted are FOU and FAC with 76 and 216 dimensions, respectively.

#### 4.2. Experimental Setting

We fine-tuned the parameters for each method with strict compliance with the experimental settings in the respective literature. For our proposed method, two parameters $\lambda_1$ and $\lambda_2$ need tuning. Therefore, we utilize a grid search to find the best $\lambda_1$ and $\lambda_2$ from $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$. Then, we perform all experiments ten times and report the mean performance to guarantee fairness.

#### 4.3. Evaluation Metrics

To evaluate our method, we utilize six standard evaluation metrics, i.e., accuracy (ACC), normalized mutual information (NMI), adjusted rand index (AR), F-score, precision, and recall. These metrics capture a different aspect of the performance to demonstrate the superiority of our method over state-of-the-art compared methods. For example, ACC measures the percentage of correctly clustered data samples in the learned clustering structure compared with the ground truth labels, whereas NMI is a theoretic validation measure, which relies on the amount of statistical information shared by random variables.

#### 4.4. Experimental Results

4.4.1. Image Clustering

In this section, we present the image clustering performance results on all five benchmark datasets using the six evaluation metrics described earlier.
Table 2 displays the performance results on ORL and Yale datasets. Specifically, on the ORL dataset, our method has better performance results than the seven state-of-the-art compared methods with 81.39%, 91.32%, 71.13%, and 83.10% in ACC, NMI, precision, and recall, respectively. Surprisingly, DiMSC outperforms the more recent methods, such as MCGC, MVGL, and MLRSSC, with 78.87% in ACC and 91.19% in NMI. The above may result from the approach utilized by DiMSC to learn the common representation matrix where it introduced a diversity regularizer to obtain a representation matrix for each view that enhanced the diversity of the different views. Besides, one can observe that GMC and SM2SC, which are more recent methods, have better performances than other compared methods in most evaluation metrics. In particular, SM2SC has the best performance of 78.36% in AR, which is just 1% higher than that obtained by our proposed method. Nonetheless, our proposed method has the best performance on the Yale dataset in all evaluation metrics, with 74.48% in ACC, 71.96% in NMI, 50.11% in AR, 51.84% in F-score, 50.42% in precision, and 59.18% in the recall metrics.

Table 3 illustrates the clustering performance on Coil-20 and Caltech 101-20 object datasets. Furthermore, our method has the best performance in all evaluation metrics,

**Algorithm 1**: Algorithm for our proposed method.

**Figure 1**: Example pictures of four benchmark datasets. (a) ORL, (b) Yale, (c) COIL-20, and (d) Caltech 101-20.
except for the Coil-20 dataset, where SM²SC has the best performance in ACC by a small margin of 0.29%.

Table 4 shows that our proposed method outperforms other state-of-the-art methods in five evaluation metrics such as NMI and AR, where it has 86.76% and 78.42%, respectively.

Overall, our method outperforms GMC, SM²SC, and MCGC only slightly, especially on object datasets, and it is not surprising because, for instance, SM²SC employed a block diagonal regularization to enforce a proper common structure. Similarly, GMC and MCGC utilized a rank constraint to find a unified clustering structure to balance the agreement between the different views by avoiding k-means spectral postprocessing of the low-dimensional embedding matrix as well. Notwithstanding, our proposed method outperforms other state-of-the-art methods by a wide margin in all evaluation metrics. Furthermore, we provide more intuitiveness of our approach by visualizing the clustering structure learned on three datasets in Figure 2. Accordingly, we can see clearly that our proposed method has a block diagonal clustering structure on all three datasets with an exact $c$ number of connected components. Therefore, we conclude that our common low-rank learning approach is very efficient.

### 4.4.2. Image Recognition with 0%, 10%, and 20% Levels of Corruptions

In this section, we present experimental results obtained for image recognition with respect to accuracy evaluation metric. In this experiment, we study the robustness against noise corruption of each algorithm by gradually injecting 10% and 20% noise into the five benchmark datasets. To perform this experiment, we keep the same parameter settings described in Section 4.2, while the K nearest neighbor (KNN) classifier is applied to evaluate the classification accuracy of each algorithm. It is easily noticeable through Tables 5–9 that all algorithms had a degrading performance with an increment in the noise level. Specifically, our proposed method had degraded performance of only about 3–5% on all datasets with a 10% level of noise corruption. However, all algorithms had a significant

| Datasets | Methods | ACC | NMI | AR | F-score | Precision | Recall |
|----------|---------|-----|-----|----|---------|-----------|--------|
| ORL      | MLRSSC  | 71.95 | 87.22 | 64.18 | 65.06 | 60.25 | 70.81 |
|          | RMSL    | 70.16 | 84.45 | 63.19 | 65.78 | 58.57 | 65.32 |
|          | MCGC    | 77.25 | 85.62 | 72.27 | 73.69 | 70.36 | 79.39 |
|          | DiMSC   | 78.87 | 91.19 | 73.55 | 74.19 | 69.48 | 79.61 |
|          | MVGL    | 74.50 | 88.63 | 74.34 | 76.14 | 69.09 | 82.06 |
|          | GMC     | 79.35 | 90.57 | 76.89 | 71.58 | 71.05 | 81.98 |
|          | SM²SC   | 80.56 | 91.08 | **78.36** | 75.36 | 70.82 | 82.76 |
|          | Ours    | **81.39** | **91.32** | 77.36 | 75.73 | **71.13** | **83.10** |

| Yale     | MLRSSC  | 64.31 | 68.18 | 47.59 | 50.94 | 48.68 | 53.46 |
|          | RMSL    | 65.43 | 70.21 | 47.28 | 48.35 | 45.14 | 55.41 |
|          | MCGC    | 68.06 | 68.37 | 45.88 | 49.63 | 43.62 | 57.58 |
|          | DiMSC   | 66.06 | 67.34 | 47.57 | 50.87 | 49.36 | 52.48 |
|          | MVGL    | 67.67 | 70.78 | 46.27 | 49.83 | 45.79 | 54.67 |
|          | GMC     | 74.05 | 66.28 | 47.58 | 50.66 | 48.28 | 57.52 |
|          | SM²SC   | 74.26 | 70.43 | 49.23 | 51.19 | 49.53 | 58.31 |
|          | Ours    | **74.48** | **71.96** | **50.11** | **51.84** | **50.42** | **59.18** |

| Datasets | Methods | ACC | NMI | AR | F-score | Precision | Recall |
|----------|---------|-----|-----|----|---------|-----------|--------|
| COIL-20  | MLRSSC  | 83.35 | 83.25 | 69.05 | 70.36 | 68.27 | 73.18 |
|          | RMSL    | 80.79 | 83.04 | 62.41 | 70.94 | 69.14 | 72.36 |
|          | MCGC    | 88.21 | 83.78 | 69.90 | 70.84 | 65.53 | 77.08 |
|          | DiMSC   | 85.23 | 82.71 | 69.86 | 71.37 | 70.30 | 72.48 |
|          | MVGL    | 85.21 | 95.92 | 83.25 | 82.17 | 75.64 | 94.88 |
|          | GMC     | 87.05 | 93.72 | 83.69 | 84.03 | 77.37 | 91.26 |
|          | SM²SC   | **89.42** | 95.71 | 83.76 | 84.21 | 78.30 | 94.81 |
|          | Ours    | 89.13 | **96.54** | **84.03** | **86.49** | **78.53** | **95.10** |

| Caltech 101-20 | MLRSSC | 55.43 | 51.91 | 46.01 | 42.06 | 83.76 | 68.86 |
|                | RMSL   | 57.94 | 52.15 | 44.67 | 41.16 | 81.32 | 64.35 |
|                | MCGC   | 64.51 | 59.47 | 46.57 | 48.06 | 88.94 | 72.72 |
|                | DiMSC  | 58.12 | 47.77 | 40.59 | 47.61 | 86.47 | 67.42 |
|                | MVGL   | 62.95 | 58.81 | 45.96 | 46.79 | 87.25 | 72.13 |
|                | GMC    | 65.23 | 60.52 | 42.80 | 46.51 | 89.15 | 69.51 |
|                | SM²SC  | 66.04 | 60.49 | 47.21 | 49.33 | 89.30 | 72.29 |
|                | Ours   | **66.42** | **61.14** | **48.75** | **49.76** | **89.44** | **73.05** |
reduction in performance with a 20% level of noise corruption. Yet, we can observe that our proposed method is generally more robust against noise than the other compared methods.

4.5. Parameter Sensitivity. Figure 3 demonstrates the clustering performance of our method with respect to NMI and ACC by varying $\lambda_1$ and $\lambda_2$ on Yale, COIL-20, and UCI digits datasets. First of all, we should note that $\lambda_3$...
Table 9: Variation of classification accuracy with different corruption rates on the UCI digits dataset.

| Method   | rates | 0%    | 10%   | 20%   |
|----------|-------|-------|-------|-------|
| MLRSSC   |       | 83.33 | 77.52 | 58.27 |
| RMSL     |       | 81.79 | 77.15 | 60.84 |
| MCGC     |       | 84.31 | 79.67 | 61.15 |
| DiMSC    |       | 86.58 | 82.34 | 64.88 |
| MVGL     |       | 86.23 | 81.55 | 64.51 |
| GMC      |       | 87.05 | 83.14 | 65.28 |
| SM²SC    |       | **88.34** | 82.98 | 66.87 |
| Ours     |       | 87.11 | **84.57** | **68.32** |

Scientific Programming 9
does not need turning like the other parameters because we find its best value heuristically like [26] to accelerate the process. Therefore, we initialize $\lambda_3 = 1$ and automatically set it as $\lambda_3 = \lambda_3 + 2$ or $\lambda_3 = (\lambda_3/2)$ in each iteration when the number of connected components is smaller or larger than $c$, respectively. As a result, we can see from Figure 3 that our method is more sensitive to $\lambda_1$ on both NMI and ACC because it controls the learning of the common low-rank matrix $U$. Hence, our proposed method is certain to have a good performance by finding a suitable value for $\lambda_1$. On the other hand, our method is only slightly sensitive to $\lambda_3$, which means that $\lambda_3$ is not as important as $\lambda_1$ to our model. Still, these clustering results show that our proposed method is relatively stable regardless of the dataset.

4.6. Convergence Analysis. Although the convergence of the inexact augmented Lagrange multiplier (ALM) method with more than three subproblems is still not easy to theoretically prove [29], we compute the relative error of the term $\|\lambda_2 S - U\|_F^2$ to demonstrate the convergence behavior of our proposed method. To illustrate this better, Figure 4 shows the errors at different iterations. It is easy to see that on all three benchmark datasets, our method converges within 150 iterations.

5. Conclusion

We proposed a novel method based on LRR, which learns a common low-rank representation matrix shared by two multi-instance datasets to improve the clustering performance accuracy. Specifically, our proposed method obtained this common low-rank representation matrix using only one instance of the data so that a second instance of the data can linearly reconstruct it. Thus, utilizing this common low-rank representation, our method obtained an ideal similarity matrix, which explicitly revealed the clustering structure without any spectral postprocessing. This approach is different from existing state-of-the-art methods, where a projection matrix is learned for each instance of the data to obtain a common structure. Extensive experiments on five benchmark datasets demonstrate our method’s superiority over seven state-of-the-art compared methods in all six evaluation metrics. In future work, we would extend our ideas to deep multiview learning.
Data Availability

The data used in this study are available in the following websites: (1) http://cam-orl.co.uk/facedatabase.html. (2) http://vision.ucsd.edu/content/yale-face-database. (3) https://www.cs.columbia.edu/CAVE/software/softlib/colt-20.php. (4) http://www.vision.caltech.edu/Image_Datasets/Caltech 101/. (5) https://archive.ics.uci.edu/ml/datasets/Optical+Recognition+of+Handwritten+Digits.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded in part by the National Key Research and Development Program of China, Grant no. 2020YFC1511800.

References

[1] B. Raytchev and H. Murase, “Unsupervised recognition of multi-view face sequences based on pairwise clustering with attraction and repulsion,” Computer Vision and Image Understanding, vol. 91, no. 1-2, pp. 22–52, 2003.

[2] P. Huang, Y. Wang, and M. Shao, “A new method for multi-view face clustering in video sequence,” in Proceedings of the 2008 IEEE International Conference on Data Mining Workshops, pp. 869–873, IEEE, Pisa, Italy, 2008.

[3] P. Qian, Y. Jiang, Z. Deng et al., “Cluster prototypes and fuzzy memberships jointly leveraged cross-domain maximum entropy clustering,” IEEE Transactions on Cybernetics, vol. 46, no. 1, pp. 181–193, 2015.

[4] P. Qian, S. Sun, Y. Jiang et al., “Cross-domain, soft-partition clustering with diversity measure and knowledge reference,” Pattern Recognition, vol. 50, pp. 155–177, 2016.

[5] M. Shao, D. Kit, and Y. Fu, “Generalized transfer subspace learning through low-rank constraint,” International Journal of Computer Vision, vol. 109, no. 1-2, pp. 74–93, 2014.

[6] S. Ando and E. Suzuki, “Unsupervised cross-domain learning by interaction information co-clustering,” in Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pp. 13–22, IEEE, Pisa, Italy, 2008.

[7] X. Zhang, H. Gao, G. Li et al., “Multi-view clustering based on graph-regularized nonnegative matrix factorization for object recognition,” Information Sciences, vol. 432, pp. 463–478, 2018.

[8] C. Wang, M. Pelillo, and K. Siddiqi, “Dominant set clustering and pooling for multi-view 3d object recognition,” 2019, http://arxiv.org/abs/1906.01592.

[9] S. Sun, “A survey of multi-view machine learning,” Neural Computing and Applications, vol. 23, no. 7-8, pp. 2031–2038, 2013.

[10] C. Xu, D. Tao, and C. Xu, “Multi-view learning with incomplete views,” IEEE T. Image Process.vol. 24, no. 12, 2015.

[11] H. Hotelling, “Relations between two sets of variates,” Biometrika, vol. 28, no. 3, pp. 321–377, 1936.

[12] S. Akaho, “A kernel method for canonical correlation analysis,” 2006, http://arxiv.org/abs/0609071.

[13] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, “Multi-view clustering via canonical correlation analysis,” in Proceedings of the 26th Annual International Conference on Machine Learning, pp. 129–136, Montreal, Canada, 2009.

[14] Y. Qin, L. Bruzzone, B. Li, and Y. Ye, “Cross-domain collaborative learning via cluster canonical correlation analysis and random walker for hyperspectral image classification,” IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 6, pp. 3952–3966, 2019.

[15] S. Sun and F. Jin, “Robust co-training,” International Journal of Pattern Recognition and Artificial Intelligence, vol. 25, no. 7, pp. 1113–1126, 2011.

[16] A. Blum and T. Mitchell, “Combining labeled and unlabeled data with co-training,” in Proceedings of the 11th Annual Conference on Computational Learning Theory, pp. 92–100, Madison, MI, USA, 1998.

[17] A. Kumar and H. Daumé, “A co-training approach for multi-view spectral clustering,” in Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 393–400, Lille, France, 2011.

[18] Z. Akata, C. Thurau, and C. Bauckhage, Non-negative Matrix Factorization in Multimodality Data for Segmentation and Label Prediction, 2011.

[19] L. Jialu, “Multi-view clustering via joint nonnegative matrix factorization,” in Proceedings of the ICDM, pp. 252–260, Beijing, China, 2013.

[20] R. Xia, Y. Pan, L. Du, and J. Yin, “Robust multi-view spectral clustering via low-rank and sparse decomposition,” in Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, pp. 2149–2155, Palo Alto, PA, USA, 2014.

[21] X. Cao, C. Zhang, H. Fu, S. Liu, and H. Zhang, “Diversity-induced multi-view subspace clustering,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern, San Francisco, CA, USA, 2015.

[22] Z. Ding and Y. Fu, “Robust multi-view subspace learning through dual low-rank decompositions,” in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, pp. 1181–1187, Phoenix, AR, USA, 2016.

[23] M. Brbic and I. Kopriva, “Multi-view low-rank sparse subspace clustering,” Pattern Recognition, vol. 73, pp. 247–258, 2018.

[24] L. Zhang, J. Fu, S. Wang, D. Zhang, Z. Dong, and C. Chen, “Guide subspace learning for unsupervised domain adaptation,” IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 9, pp. 3374–3388.

[25] K. Zhan, C. Zhang, J. Guan, and J. Wang, “Graph learning for multiview clustering,” IEEE Transactions on Cybernetics, vol. 48, no. 10, pp. 2887–2895, 2018.

[26] K. Zhan, F. Nie, J. Wang, and Y. Yang, “Multiview consensus graph clustering,” IEEE Transactions on Image Processing, vol. 28, no. 3, pp. 1261–1270, 2019.

[27] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, “Canonical correlation analysis: an overview with application to learning methods,” Neural Computation, vol. 16, no. 12, pp. 2639–2664, 2004.

[28] G. Liu, Z. Lin, and Y. Yu, “Robust subspace segmentation by low rank representation,” in Proceedings of the 27th International Conference on Machine Learning, pp. 663–670, Haifa, Israel, 2010.

[29] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, “Robust recovery of subspace structures by low-rank representation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 1, pp. 171–184, 2013.

[30] E. Elhamifar and R. Vidal, “Sparse subspace clustering: algorithm, theory, and applications,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 11, pp. 2765–2781, 2013.
[31] L. Parsons, E. Haque, and H. Liu, “Subspace clustering for high dimensional data,” Acm Sigkdd Explorations Newsletter, vol. 6, no. 1, pp. 90–105, 2004.

[32] W. Wang and Z. Zhou, “A new analysis of co-training,” in Proceedings of the 27th International Conference on Machine Learning, pp. 1135–1142, Haifa, Israel, 2010.

[33] Y. Yang and H. Wang, “Multi-view clustering: a survey,” Big Data Mining and Analytics, vol. 1, no. 2, pp. 83–107, 2018.

[34] Y. Wang, L. Wu, X. Lin, and J. Gao, “Multiview spectral clustering via structured low-rank matrix factorization,” IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 10, pp. 4833–4843, 2018.

[35] R. Vidal and P. Favaro, “Low rank subspace clustering (LRSC),” Pattern Recognition Letters, vol. 43, pp. 47–61, 2014.

[36] D. Zhang, Y. Hu, J. Ye, X. Li, and X. He, “Matrix completion by truncated nuclear norm regularization,” in Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2192–2199, IEEE, Providence, RI, USA, June 2012.

[37] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On spectral clustering: analysis and an algorithm,” in Proceedings of the Advances in Neural Information Processing Systems, pp. 849–856, Lake Tahoe, NV, USA, 2002.

[38] Z. Yang, Q. Xu, W. Zhang, X. Cao, and Q. Huang, “Split multiplicative multi-view subspace clustering,” IEEE Transactions on Image Processing, vol. 28, no. 10, pp. 5147–5160, 2019.

[39] Y. Liang, D. Huang, and C. D. Wang, “Consistency meets inconsistency: a unified graph learning framework for multi-view clustering,” in Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), pp. 1204–1209, IEEE, Beijing, China, 2019.

[40] H. Wang, Y. Yang, B. Liu, and GMC. “,” “GMC: graph-based multi-view clustering,” IEEE Transactions on Knowledge and Data Engineering, vol. 32, no. 6, pp. 1116–1129, 2020.

[41] C. Zhang, Q. Hu, H. Fu, P. Zhu, and X. Cao, “Latent multi-view subspace clustering,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4279–4287, Seattle, WA, USA, 2017.

[42] Y. Zheng, G. Liu, S. Sugimoto, S. Yan, and M. Okutomi, “Practical low-rank matrix approximation under robust l 1-norm,” in Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1410–1417, Providence, RI, USA, 2012.

[43] Z. Lai, D. Mo, W. K Wong, Y. Xu, D. Miao, and D. Zhang, “Robust discriminant regression for feature extraction,” IEEE Transactions on Cybernetics, vol. 48, no. 8, pp. 2472–2484, 2017.

[44] F. R. Chung, Spectral Graph Theory, Vol. 92, American Mathematical Society, Providence, RI, USA, 1997.

[45] U. von Luxburg, “A tutorial on spectral clustering,” Statistics and Computing, vol. 17, no. 4, pp. 395–416, 2007.

[46] J.-F. Cai, E. J. Cand`es, and Z. Shen, “A singular value thresholding algorithm for matrix completion,” SIAM Journal on Optimization, vol. 20, no. 4, p. 1956, 2010.

[47] B. Recht, M. Fazel, and P. A. Parrilo, “Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization,” SIAM Review, vol. 52, no. 3, pp. 471–501, 2010.

[48] K. Fan, “On a theorem of weyl concerning eigenvalues of linear transformations: II,” National Academy of Sciences, vol. 36, no. 1, pp. 31–35, 1950.