Unsupervised Feature Selection Algorithm Based on Laplace Rank Constraint and Local Structure Preservation

Yingying Meng, Qiaoyan Li, Xiaofei Yang, Xuezhen Dai

***Abstract*—Feature selection aims to select an optimal feature subset to reduce the dimension of original data, thereby solving the "dimension disaster" problem effectively. For feature selection, preserving the local manifold structure of the data is crucial, so how to learn an excellent local manifold structure has always been a research hotspot in this field. In this paper, we propose an unsupervised feature selection algorithm based on Laplace rank constraint and local structure preservation. First, we combine the locally linear embedding method with the Laplace rank constraint method to learn an outstanding similarity matrix. Secondly, the projection matrix is used to select features while preserving the similarity matrix. Furthermore, to avoid selecting redundant features, the regularization term about the redundancy of the projection matrix is used to select features with more discriminant information. In addition, the model optimization algorithm is proposed, and the model complexity is analyzed. Experiments on several public datasets show that our method can learn better manifold structural information, and select features with more discriminant information.**

***Index Terms*—Feature selection, Laplace rank constraint, local structure preservation, sparse learning, unsupervised learning**

I. INTRODUCTION

In many fields, the feature dimension of data is excessively large, which might lead to poor algorithm performance. Because many features in high-dimensional data are redundant and noisy. There are two ways to address this problem: feature extraction[1] and feature selection[2]. Unsupervised feature selection has attracted a lot of attention since labeling data is a challenging and time-consuming task.

Manifold learning[2] is a method for recovering low-dimensional manifold structures from high-dimensional data to reduce dimensionality. Nonlinear manifold learning algorithms include isometric mapping (Isomap)[3], Laplace feature map[4], locally linear embedding (LLE)[5], etc. The locally linear embedding assumes that the data is nonlinear on the total but linear on the parts. Meanwhile, the local geometric structure of the low-dimensional feature space should be consistent with the original feature space. As a result, many scholars devote themselves to preserving both local and global structures. For example, Liu et al.[6] combined both global sample similarity and local geometric data structure for feature selection. Peng et al.[7] constructed the local structure of the original data based on data point similarity. They also selected representative features to preserve the local structure. Liu et al.[8] obtained the feature weight matrix by the locally linear embedding algorithm and proposed a robust neighborhood embedding method. Considering both global and local structure preservation, Zhou et al.[9] proposed an iterative method for unsupervised feature selection.

The papers listed above improve feature selection algorithms by preserving local or global manifold structures. Furthermore, a good similarity weight matrix is very important for manifold structure learning. Based on the self-representation of data, Wang et al.[10] proposed the algorithm, which regarded the data as a dictionary for sparse coding. They also preserved the coefficient matrix in the process of dimensional reduction. The unsupervised feature selection algorithm combining adaptive manifold and embedded learning was proposed by Wu et al.[11], which learned a low dimensional embedding to preserve adaptive manifold structure. Wen et al.[12] used a loss function to capture the true structure of the data while preserving its global structure during feature selection. Wang et al.[13] proposed an unsupervised feature selection algorithm using low-rank approximation and structural learning. The Laplacian matrix rank constrained clustering algorithm was proposed by Nie et al.[14], which applies graph theory to impose rank constraints on the Laplacian matrix. Inspired by these results, we discovered a graph with rank restrictions with precisely *c*-linked components (where *c* represents the number of clusters). The linked components are utilized to determine the true similarity matrix. The similarity matrix exhibits a block diagonal structure.

Feature selection aims to reduce redundant features. Many authors have offered various approaches for dealing with feature redundancy. The pairwise dependence-based unsupervised feature selection calculates the redundant relationship between features using mutual information[15].

Manuscript received November 24, 2023; revised September 9, 2024.

This work was supported by the Natural Science Foundation of China (61976130), the Natural Science Foundation of Shaanxi Province (2022JM-053), and the Key Research and Development Projects of Shaanxi Province (2018KW-021).

Yingying Meng is a graduate student of Xi'an Polytechnic University, Xi'an City, Shaanxi Province, 710048, China (e-mail: mengying202203@163.com).

Qiaoyan Li is an associate professor at the School of Science, Xi'an Polytechnic University, Xi'an, Shaanxi Province, 710048, China. She is the corresponding author (Phone: +86-13488195310, e-mail: liqiaoyan@xpu.edu.cn).

Xiaofei Yang is an associate professor at the School of Science, Xi'an Polytechnic University, Xi'an, Shaanxi Province, 710048, China (e-mail: yangxiaofei2002@163.com).

Xuezhen Dai is an associate professor at the Public Sector, Xi'an Traffic Engineering Institute, Xi'an, Shaanxi Province, 710300, China (e-mail: 420073948@qq.com).

Furthermore, Xu et al.[16] successfully discovered redundant features from a global perspective of feature redundancy. They also suggested a feature selection approach based on orthogonal regression to minimize global redundancy. Li et al.[17] presented the unsupervised feature selection algorithm, which combines local structure preservation with redundancy minimization.

Attribute reduction is a significant study direction for rough sets. The goal is to obtain a minimum attribute subset that is invariant to the original data information under specified conditions. Different attribute reduction strategies have been proposed. The paper [18] developed a reduction procedure that preserves the distribution of decision areas in the rough set model. The paper [19] suggested a heuristic technique for attribute reduction that used similarity to preserve generalized decisions. Inspired by the rough set attribute reduction method, we propose an unsupervised feature selection algorithm based on the Laplace rank constraint and local structure preservation. The algorithm consists of two steps. Firstly, the locally linear embedding and rank constraint are merged to create a similarity matrix that includes all features. Secondly, our feature selection method is based on similarity and projection matrix. Similar to the attribute reduction idea in rough sets, our method can reveal more discriminative features by preserving the similarity matrix.

In this paper, we propose an unsupervised feature selection algorithm based on the Laplace rank constraint and local structure preservation (LRLSP).

The primary contributions of the LRLSP algorithm are as follows:

(1) To learn a similarity matrix with block diagonal structure, we combine locally linear embedding with Laplace rank constraint and attempt to preserve the local structure.

(2) To avoid selecting redundant features, we introduce a redundant regularization term into the projection matrix, guaranteeing that the learned features contain more discriminative information.

(3) The $L_{2,1}$ norm captures the sparsity of the feature projection matrix, reducing the influence of data noise on feature selection.

The rest sections of this paper are organized as follows. In the second section, some related works, including traditional linear neighborhood reconstruction and Laplace rank constraint, are briefly introduced. In the third section, we introduce the modeling and iterative algorithm for our model. In the fourth section, the experimental results on several public datasets are presented, and further relevant analyses are conducted. Finally, the conclusion is presented.

II. Related works

A. Notations

Given $X \in R^{d\times n}$ as the data matrix, where n denotes the number of samples and d denotes the dimension of features. $X_{\cdot i} \in R^{d\times 1}$ is denoted as the i-th column of X, which means i-th sample of X. Similarly, $X_{i\cdot} \in R^{1\times n}$ is defined as the i-th row of X, representing the i-th feature of X. $H \in R^{d\times m}$ is a projection matrix, where m represents the number of selected features, and $m \ll d$. $W \in R^{n\times n}$ denotes the similarity matrix of all samples. We define $I_n \in R^{n\times n}$ as an identity matrix, and $1 \in R^{n\times 1}$ as an all-one vector.

B. Locally linear embedding

LLE is a nonlinear dimension reduction algorithm. This indicates that a sample can be linearly represented by many neighbors in the original space. As a result, the newly collected data could precisely preserve the original manifold structure while considerably resolving the "dimension disaster" problem. Based on the above idea, locally linear embedding is set up as follows:

$$\min_{W} \sum_{i=1}^{n} \left\| X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} X_{\cdot j} \right\|_2^2 \quad s.t. \sum_{j=1}^{k} W_{ji} = 1, \tag{1}$$

where $N_k(X_{\cdot i})$ means the k nearest neighbor set of $X_{\cdot i}$, and $W \in R^{n\times n}$ is the weight matrix. To avoid the columns of W being all zeros, we further constrain the column of W overall to 1.

C. Laplace rank constraints

It is obvious that $W \in R^{n\times n}$ is not a symmetry matrix in (1). Let $W = \frac{W + W^T}{2}$, then W is the symmetric matrix. And the new W is referred to as the similarity matrix. $D \in R^{n\times n}$ is a diagonal matrix whose i-th diagonal element equals the sum of i-th row in the similarity matrix W.

According to [14], the multiplicity c of the zero eigenvalue in a Laplacian matrix is the same as the number of connected components in the similarity matrix. Denote $L = D - W$. If $rank(L) = n - c$, then a similarity graph admits c connected components. As a result, each of the connected components might be regarded as a cluster.

III. The proposed framework for feature selection

In this section, based on Laplace rank constraint and local structure preservation, an unsupervised feature selection model LRLSP is proposed. Its main idea is to preserve the local structure. Concretely, the model is divided into two parts. Firstly, the locally linear embedding and rank constraint are merged to generate a similarity matrix that includes all features. Secondly, the proposed feature selection model is created utilizing the aforementioned similarity matrix. Finally, the corresponding algorithm is designed to solve the LRLSP.

A. Objective function

Firstly, based on locally linear embedding and Laplace rank constraint, the objective function of the similarity matrix is calculated as follows:

$$\min_{W} \sum_{i=1}^{n} \left\| X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} X_{\cdot j} \right\|_2^2$$

$$s.t.\sum_{j=1}^{k} W_{ji} = 1, W_{ji} \geq 0, rank(L) = n - c. \quad (2)$$

Since W is a weight used to evaluate how important $X_{\cdot j}$ is in reconstructing $X_{\cdot i}$, we impose the nonnegative constraint $W_{ji} \geq 0$. In (2), the rank constraint aims to generate a similarity matrix with a *c*-block diagonal structure. Since $rank(L) = n - c$ is a nonlinear constraint, (2) must be transformed into the following form:

$$\begin{aligned} \min_{W} \sum_{i=1}^{n} \left\| X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} X_{\cdot j} \right\|_2^2 + 2\alpha \sum_{i=1}^{c} \sigma_i(L) \\ s.t. \sum_{j=1}^{k} W_{ji} = 1, W_{ji} \geq 0, \end{aligned} \quad (3)$$

where $\sigma_i(L)$ denotes the *i*-th minimum eigenvalue of L. $\sigma_i(L) \geq 0$ indicates that L is positive semidefinite. $\sigma_i(L)$ will approach 0 when α is large enough. Thus, the optimal solution to (3) may satisfy the requirements of problem (2). In addition, according to [14], we have:

$$\sum_{i=1}^{c} \sigma_i(L) = \min_{F \in R^{n \times c}, F^T F = I_c} Tr(F^T L F). \quad (4)$$

Therefore, we can further obtain (5) as the following form:

$$\begin{aligned} \min_{W,F} \sum_{i=1}^{n} \left\| X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} X_{\cdot j} \right\|_2^2 + 2\alpha Tr(F^T L F) \\ s.t. \sum_{j=1}^{k} W_{ji} = 1, W_{ji} \geq 0, F \in R^{n \times c}, F^T F = I_c, \end{aligned} \quad (5)$$

where L is defined in section II.C. The weight matrix W in the original space can be obtained via (5). It differs from the LLE method in that it considers block diagonal structures.

In the following works, based on the weight matrix W, we introduce a projection matrix to reduce the dimensions while selecting features.

Given that the original space contains a large amount of redundant information, the outcome of feature selection may be affected. We decide to learn the feature redundancy matrix using the cosine similarity method. After the model has removed redundant features, it learns more discriminant features and performs better in actual applications.

According to the LLE assumption, when the original data $X_{\cdot i}$ is reduced from the *d* to *m* dimensions, the new data can still be described as the same linear combination of its *k* nearest neighbors. This means that the weight matrix W is preserved. Here, $H \in R^{d \times m}$ is the projection matrix of features. The feature selection model is as follows.

$$\min_{H} \sum_{i=1}^{n} \left\| H^T X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} (H^T X_{\cdot j}) \right\|_2^2 + \gamma R(H), \quad (6)$$

where γ is a regularization parameter, and $R(H)$ denotes the regularized term.

To remove redundant data, we define the redundancy matrix $Q \in R^{d \times d}$ with $Q_{ij} = \left(\frac{x_{i\cdot} x_{j\cdot}^T}{\|x_{i\cdot}\| \|x_{j\cdot}\|} \right)^2$, where $x_{i\cdot}$ and $x_{j\cdot}$ are the feature vectors after centralization. $x_{i\cdot} x_{j\cdot}^T = |x_{i\cdot}| |x_{j\cdot}| \cos\theta$, where θ is the angle between two vectors. When $\theta = 0^\circ$, the two vectors are linearly correlated, and their redundancy Q_{ij} is high. In contrast, when $\theta = 90^\circ$, the two vectors are linearly independent, with a redundancy Q_{ij} of 0. By introducing a redundant regularization term in (6), we can obtain:

$$\begin{aligned} \min_{H} \sum_{i=1}^{n} \left\| H^T X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} (H^T X_{\cdot j}) \right\|_2^2 \\ + \beta Tr(H^T Q H), \end{aligned} \quad (7)$$

where $\beta > 0$ is a parameter that controls the redundancy regularization term. Finally, we employ $L_{2,1}$ norm to guarantee the sparsity while weakening the performance of outliers for projection matrix H. Hence, the sparsity of H is helpful for feature selections. Equation (7) can be written as:

$$\begin{aligned} \min_{H} \sum_{i=1}^{n} \left\| H^T X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} (H^T X_{\cdot j}) \right\|_2^2 \\ + \beta Tr(H^T Q H) + \gamma \|H\|_{2,1}. \end{aligned} \quad (8)$$

Since the orthogonal constraint ensures that the data in the low-dimensional space is statistically irrelevant. Therefore, we add an orthogonal constraint to (8), and the LRLSP model is as follows:

$$\begin{aligned} \min_{H} \sum_{i=1}^{n} \left\| H^T X_{\cdot i} - \sum_{X_{\cdot j} \in N_k(X_{\cdot i})} W_{ji} (H^T X_{\cdot j}) \right\|_2^2 \\ + \beta Tr(H^T Q H) + \gamma \|H\|_{2,1} \\ s.t. H^T H = I_m. \end{aligned} \quad (9)$$

After solving the problem (9), the top *m*-ranked features are selected according to $\|H_{\cdot i}\|_2 \ (i = 1, 2, ..., d)$. From (5) and (9), we can observe that the low-dimension space has the same self-representation as the original space. Furthermore, some features with low redundancy can be selected by H. As a result, (9) can be used to select features and acquire additional discriminate features.

B. Optimization

In this section, we propose an optimization algorithm for the LRLSP. Since there are many variables in the algorithm, it is difficult to solve them together, so we use the alternative direction multipliers(ADMM) method to solve them. The algorithm solution is divided into two parts. The first step is to obtain the weight matrix with a sparse block diagonal structure by (5), and the second step is to obtain the projection matrix for feature selection by (9). The specific steps are as follows.

1) Solving the weight matrix: There are two variables in (5), and we need to simplify them further. So we need to simplify the first term of (5):

$$\begin{aligned}
&\sum_{i=1}^{n}\left\|X_{\cdot i}-\sum_{X_{\cdot j}\in N_k(X_{\cdot i})}W_{ji}X_{\cdot j}\right\|_2^2\\
&=\sum_{i=1}^{n}\left\|\sum_{X_{\cdot j}\in N_k(X_{\cdot i})}W_{ji}X_{\cdot i}-\sum_{X_{\cdot j}\in N_k(X_{\cdot i})}W_{ji}X_{\cdot j}\right\|_2^2\\
&=\sum_{i=1}^{n}\left\|\sum_{X_{\cdot j}\in N_k(X_{\cdot i})}W_{ji}(X_{\cdot i}-X_{\cdot j})\right\|_2^2\\
&=\sum_{i=1}^{n}\left\|(X^{(i)}-N^{(i)})W_{\cdot i}\right\|_2^2\\
&=\sum_{i=1}^{n}W_{\cdot i}{}^T(X^{(i)}-N^{(i)})^T(X^{(i)}-N^{(i)})W_{\cdot i},
\end{aligned}\tag{10}$$

where $X^{(i)}=[X_{\cdot i},X_{\cdot i},...,X_{\cdot i}]\in R^{d\times k}$ is the matrix containing the *i*-th sample, which copies *k* columns. The *k* nearest neighbors of the *i*-th sample represent $N^{(i)}=[X_{\cdot j_1},X_{\cdot j_2},...,X_{\cdot j_k}]\in R^{d\times k}$. For convenience, we denote $A^{(i)}=(X^{(i)}-N^{(i)})^T(X^{(i)}-N^{(i)})\in R^{k\times k}$.

Denote $B_{ji}=\sum_{j=1}^{n}\left\|f_i-f_j\right\|_2^2$ if the *j*-th sample belongs to the *k* nearest of the *i*-th sample, $B_{ji}=0$ otherwise. Thus, the (5) is equivalent to

$$\begin{aligned}
&\min_{W,F}\frac{1}{2}\sum_{i=1}^{n}W_{\cdot i}{}^TA^{(i)}W_{\cdot i}+\alpha\sum_{i=1}^{n}B_{\cdot i}{}^TW_{\cdot i}\\
&s.t.\sum_{j=1}^{k}W_{ji}=1,W_{ji}\ge 0,F\in R^{n\times c},F^TF=I_c.
\end{aligned}\tag{11}$$

Based on the aforementioned formulation, we can update W and F in an alternating method.

a) update W, fix F

Firstly, we need to solve the problem, which may be simplified as:

$$\begin{aligned}
&\min_{W}\frac{1}{2}\sum_{i=1}^{n}W_{\cdot i}{}^TA^{(i)}W_{\cdot i}+\alpha\sum_{i=1}^{n}B_{\cdot i}{}^TW_{\cdot i}\\
&s.t.\sum_{j=1}^{k}W_{ji}=1,W_{ji}\ge 0.
\end{aligned}\tag{12}$$

For the *i*-th sample, we have:

$$\begin{aligned}
&\min_{W}\frac{1}{2}W_{\cdot i}{}^TA^{(i)}W_{\cdot i}+\alpha B_{\cdot i}{}^TW_{\cdot i}\\
&s.t.\sum_{j=1}^{k}W_{ji}=1,W_{ji}\ge 0.
\end{aligned}\tag{13}$$

Here, we solve (13) by quadratic programming[20].

b) update F, fix W

We need to solve the problem, which may be summarized as:

$$\begin{aligned}
&\min_{F}Tr(F^TLF)\\
&s.t.F\in R^{n\times c},F^TF=I.
\end{aligned}\tag{14}$$

The optimal solution of F is composed of the eigenvectors corresponding to the *c* minimum eigenvalues of L. Each optimization variable of the objective function (11) is handled using alternative strategies. The proposed iterative algorithm is shown in Algorithm 1.

Algorithm 1 Optimization Algorithm for Problem (11)

Input: Data matrix $X\in R^{d\times n}$, the number of the nearest neighbor *k*

Output: The weight matrix $W\in R^{n\times n}$

1: Initialize the weight matrix $W\in R^{n\times n}$, pseudo label matrix $F\in R^{n\times c}$

2: Calculate the *k* nearest neighbor $N^{(i)}$ of the *i*-th sample

3: for $i=1$ to n do

4: update $A^{(i)}=(X^{(i)}-N^{(i)})^T(X^{(i)}-N^{(i)})\in R^{k\times k}$

5: end for

6: while not converged, do

7: for $i=1$ to n do

8: update W_{ji} by quadratic programming

9: end for

10: Calculate the eigenvectors corresponding to eigenvalues of L and update the pseudo label matrix F by (14)

11: end while

2) Solving the projection matrix: To solve H, we simplify the first term of (9).

$$\begin{aligned}
&\sum_{i=1}^{n}\left\|H^TX_{\cdot i}-\sum_{X_{\cdot j}\in N_k(X_{\cdot i})}W_{ji}H^TX_{\cdot j}\right\|_2^2\\
&=\sum_{i=1}^{n}\left\|H^TX_{i}-\sum_{j=1}^{n}W_{ji}H^TX_{\cdot j}\right\|_2^2\\
&=\sum_{i=1}^{n}\left\|H^TX_{\cdot i}-H^TXW_{\cdot i}\right\|_2^2\\
&=\left\|H^TX(I_n-W)\right\|_F^2\\
&=\left\|H^TP\right\|_F^2\\
&=Tr(H^TPP^TH),
\end{aligned}\tag{15}$$

where $P=X(I_n-W)$. According to [21],[22], we have $\left\|H\right\|_{2,1}=Tr(H^TD_HH)$, where $D_H=diag(\frac{1}{\left\|H_{\cdot i}\right\|_2+\varepsilon})$.

Therefore, we have the following minimization objective function:

$$\begin{aligned}
&\min_{H}Tr(H^T(P^TP+\gamma D_H+\beta Q)H)\\
&s.t.H^TH=I_m.
\end{aligned}\tag{16}$$

Therefore, the final objective function is as follows:

$$\begin{aligned}
&\min_{H}Tr(H^TMH)\\
&s.t.H^TH=I_m,
\end{aligned}\tag{17}$$

where $M=P^TP+\gamma D_H+\beta Q$. Similar to F, we can update H and propose an additional efficient algorithm.

Based on the above problem formulation, the proposed the LRLSP algorithm procedure is summarized in Algorithm 2.

Algorithm 2 LRLSP unsupervised feature selection algorithm

Input: Data matrix $X \in R^{d\times n}$, the number of cluster *c*, the hyper-parameters α, γ, and the number of selected features *m*

Output: Select the top *m* features according to the order

1: Initialize the projection matrix $H \in R^{d\times m}$

2: Update W by Algorithm (1) and calculate the redundancy matrix Q

3: while not converged, do

4: update $D_H = diag(\frac{1}{\|H_{\cdot i}\|_2 + \varepsilon})$

5: calculate the eigenvectors corresponding to the first *m* minimum eigenvalues of M and update the projection matrix H by (17)

6: end while

C. Time complexity analysis

When optimizing the objective function of LRLSP, the computational complexity of each parameter is as follows. The algorithm for solving $A^{(i)}$ has a time complexity of $O(dk^2)$. The F is updated by the eigenvalue decomposition and its time complexity is $O(n^3)$. The time complexity of M is $O(n^2 d)$. H has a time complexity of $O(d^3)$. So, the total computational complexity of the LRLSP algorithm is $O(d^3 + n^3 + n^2 d + dk^2)$. Here *d* is the feature number of samples, *k* is the number of the nearest neighbor, and *n* represents the number of samples.

D. Convergence of Algorithm 2

In this part, we demonstrate the convergence of the Algorithm 2. The convergence is demonstrated by ensuring that the objective function decreases under the update procedures for each variable. Equations (11) and (17) have three separate variables, so we utilize the alternate iteration approach to solve them. The optimization approach is separated into two parts: updating F and W when H is fixed, and updating H once F and W are fixed. Equation (13) is a constrained quadratic programming problem about $W_{\cdot i}$, and $A^{(i)}$ is a positive semi-definite matrix, therefore (13) can find the optimal solution. The approach for proving the monotonicity of the goal function is described in full below.

Proof: Denote the objective value in the *t*-th iteration as $\mathbb{O}(W^{(t)}, F^{(t)}, H^{(t)})$. Algorithm 1 updates W, F, H with the optimal solution in each iteration, hence for the (*t*+1)-th iteration, it must hold:

$$\mathbb{O}(W^{(t+1)}, F^{(t+1)}, H^{(t+1)}) \le \mathbb{O}(W^{(t)}, F^{(t)}, H^{(t)}). \quad (18)$$

According to (13), we have:

$$\mathbb{O}(W^{(t+1)}, F^{(t)}, H^{(t)}) \le \mathbb{O}(W^{(t)}, F^{(t)}, H^{(t)}). \quad (19)$$

According to (14), we have:

$$\begin{aligned} F^{(t+1)} &= \arg\min_{F\in R^{n\times c}, F^T F=I} Tr(F^{(t)T} L F^{(t)}) \\ &= \arg\min_{F\in R^{n\times c}, F^T F=I} \mathbb{O}(W^{(t+1)}, F, H^{(t)}). \end{aligned} \quad (20)$$

So, we have:

$$\mathrm{O}(W^{(t+1)}, F^{(t+1)}, H^{(t)}) \le \mathrm{O}(W^{(t+1)}, F^{(t)}, H^{(t)}). \quad (21)$$

According to (17), we have:

$$\begin{aligned} H^{(t+1)} &= \arg\min_{s.t. H^T H=I} Tr(H^{(t)T} M H^{(t)}) \\ &= \arg\min_{s.t. H^T H=I} \mathbb{O}(W^{(t+1)}, F^{(t+1)}, H). \end{aligned} \quad (22)$$

So, we have:

$$\mathbb{O}(W^{(t+1)}, F^{(t+1)}, H^{(t+1)}) \le \mathbb{O}(W^{(t+1)}, F^{(t+1)}, H^{(t)}). \quad (23)$$

Therefore, we can obtain:

$$\mathbb{O}(W^{(t+1)}, F^{(t+1)}, H^{(t+1)}) \le \mathbb{O}(W^{(t)}, F^{(t)}, H^{(t)}). \quad (24)$$

The above demonstrates how the objective function shrinks with each iteration. Furthermore, since functions (5) and (9) are convex in all variables, the algorithm converges.

IV. Experiments

In this section, we will assess the effectiveness of the LRLSP approach on numerous public datasets. Additionally, the proposed algorithm is compared to other advanced unsupervised feature selection methods. Furthermore, the sensitivity influence of factors is investigated in depth.

A. Datasets

In the following experiments, six publicly available datasets are used to evaluate the performance of the LRLSP method. The detailed information on the datasets is summarized in Table I.

TABLE I
DATA DESCRIPTION

Datasets	#Instances	#Features	#Classes	Data types
COIL20	1440	1024	20	Face image
ORL	400	1024	40	Face image
Yale	165	1024	15	Face image
Isolet	1560	617	26	Speech signal
warpAR10P	130	2400	10	Face image
Jaffe	213	676	10	Face image

B. Experimental setting

To verify the effectiveness of the LRLSP algorithm, we compare it with eight advanced unsupervised feature selection methods, which include Laplacian score (LS)[23], multi-cluster feature selection method (MCFS)[24], unsupervised discriminant feature selection method (UDFS)[25], nonnegative discriminant feature selection method (NDFS)[26], feature selection method via low-rank approximation and structural learning (LRSL)[13], robust neighborhood embedded unsupervised feature selection algorithm (RNE)[8], unsupervised feature selection algorithm based on feature dependency (DUFS)[15], unsupervised feature selection method based on adaptive graph learning and constraint (EGCFS)[27], and all features selected (baseline).

Clustering accuracy (ACC) and normalized mutual information (NMI) are used to verify the effectiveness of the LRLSP algorithm. They fall within the range of [0,1]. The larger the value is, the better the algorithm is.

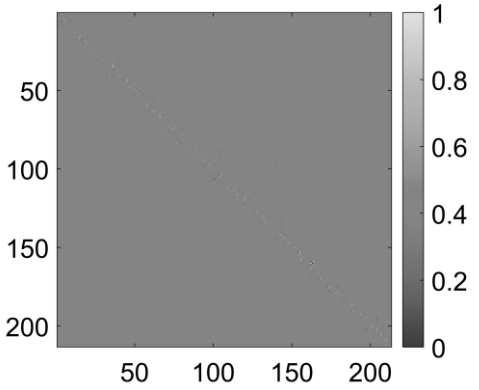
Similar to the majority of published studies, the number of *k* nearest neighbors is set to 5 in all algorithms. Except for the baseline, the number of selected features ranges $\{50, 100, ..., 300\}$ for all datasets. A grid search approach is used to fix the searching regions of β and γ as $\{1e-6, ..., 1e5, 1e6\}$. In the LRLSP algorithm, the parameter α has an initial value of 0.1, and it is adjusted adaptively during the iteration to satisfy the rank constraint. Due to the initialization sensitivity of *k*-means clustering, we randomly generate initial values 20 times as an epoch. For calculating the mean and standard deviations, the experiments need to be repeated 10 times.

C. Experimental results and analysis

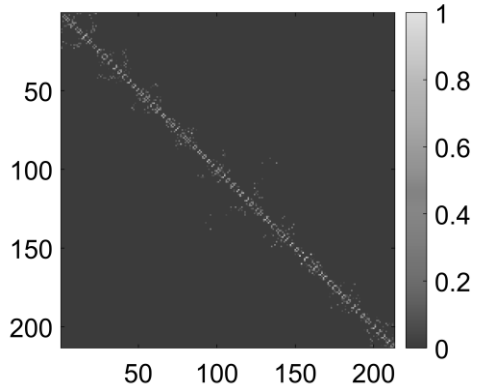
Firstly, the experimental results of ACC and NMI of all methods are shown in Table II and Table III. The ACC and corresponding standard deviations of 9 feature selection methods are displayed in Table II under the ideal dimensions. Likewise, Table III displays the NMI results for feature selection. The best results are highlighted in bold. The values in parentheses represent the number of features selected. It is easy to find that the performance of the LRLSP algorithm is superior to other advanced methods in most cases. For instance, as shown in Table I, our method achieved the highest accuracy on COIL20 datasets with the least number of selected features. This outcome highlights that excessive feature selection not only leads to redundancy but also compromises interpretability. Compared with other methods, the LRLSP algorithm improves the clustering accuracy by 6.48%, and mutual information by 4.42%. This result suggests that we may select more discriminating features. Because our model obtains superior local geometry of the data, our method is superior to RNE in terms of results. In addition, the learned similarity matrix has a block diagonal structure, which makes it possible to accurately determine the relationship between the original samples. The Jaffe similarity matrix generated by the RNE method is shown in Figure 1(a), while the Jaffe similarity matrix generated by our method is shown in Fig. 1(b).

At the same time, we can see that ACC does not increase with the increase of feature dimension. The most likely reason is that there is a lot of redundant information in the original data, and when the number of selected features reaches a certain number, adding more features will introduce a lot of redundant information. For example, in the dataset warpAR10P, when we select 250 features, the clustering accuracy of the algorithm is very high. However, as the number of features increases, the clustering accuracy of the algorithm decreases. It's indicated that many redundant features are introduced. Overall, our algorithm achieves relatively optimistic clustering accuracy when selecting a few features in most cases.

Secondly, as shown in Fig. 2, the results of ACC on different datasets are also displayed as curves. According to the graphic, the LRLSP algorithm is superior to other algorithms in most cases, especially when using the Yale datasets. However, in the Isolet datasets, the results of the LRLSP algorithm were lower than the baseline. One possible explanation is that the LRLSP algorithm selects 200 features out of 617, which means that the selected features are too few to express the data information well, resulting in poor clustering accuracy.



(a) similarity matrix obtained by RNE method



(b) similarity matrix obtained by our method

Fig. 1. Block diagonal comparison diagram of the similarity matrix

Finally, Fig. 3 shows the sensitivity of the two parameters (β and γ) when our algorithm selects 300 features. As shown in Fig. 3, our algorithm is insensitive in most cases. Unfortunately, we need to find the grid search strategy to better select the parameters. Therefore, for the datasets COIL20, Isolet, and Jaffe, the searching regions of β and γ are fixed as $\{1e-6, ..., 1e-1, 1\}$ and $\{1, ..., 1e5, 1e6\}$, respectively. Concretely, datasets COIL20 have the best performance when $\beta = 1e-4$, $\gamma = 1e1$. Datasets Isolet have the best when $\beta = 1e-4$, $\gamma = 1e6$, and Jaffe have the best when $\beta = 1e-3$, $\gamma = 1e4$. Similarly, for datasets ORL, warpAR10P, and Yale, the searching regions of β and γ are all fixed as $\{1e-6, ...,1e-1, 1\}$. Concretely, ORL datasets have the best performance when $\beta = 1e-6,\ \gamma = 1e-4$. Datasets warpAR10P have the best when $\beta = 1e-2,\ \gamma = 1e-4$, and Yale has the best when $\beta = 1e-6,\ \gamma = 1e-4$.

TABLE II
CLUSTERING ACCURACY OF DIFFERENT FEATURE SELECTION ALGORITHMS ON DIFFERENT DATASETS (ACC%±STD %)

Datasets	COIL20	Isolet	Jaffe	ORL	warpAR10P	Yale
LS	60.01±2.76(250)	58.85±1.72(300)	95.31±7.71(250)	44.90±2.38(300)	21.00±0.00(300)	38.67±1.91(100)
MCFS	62.87±1.64(200)	61.19±3.69(300)	90.94±5.13(200)	51.95±3.51(150)	22.77±2.06(250)	41.58±3.75(150)
UDFS	61.11±2.33(300)	58.29±2.21(300)	82.39±4.56(300)	47.82±2.95(300)	40.92±3.05(250)	40.06±1.77(200)
NDFS	57.73±2.17(300)	57.76±1.67(250)	81.97±1.98(250)	49.33±2.29(300)	30.77±1.54(200)	35.76±2.08(250)
LRSL	57.89±2.42(300)	61.25±2.41(300)	81.88±4.00(150)	47.55±2.06(300)	30.15±2.29(150)	35.52±2.04(200)
DUFS	56.51±3.42(300)	58.74±3.38(150)	82.91±4.63(300)	44.65±2.30(300)	34.38±6.62(250)	35.70±2.07(300)
RNE	61.47±2.28(300)	49.44±2.72(250)	87.93±3.76(300)	51.25±2.35(250)	35.62±4.44(250)	43.52±2.88(250)
EGCFS	52.15±1.40(300)	46.24±1.01(300)	88.87±4.85(250)	51.43±2.44(250)	31.85±3.25(50)	34.48±1.79(150)
Baseline	62.70±3.27(1024)	63.55±2.52(617)	88.54±5.63(676)	51.57±1.46(1024)	25.31±2.47(2400)	40.55±1.68(1024)
LRLSP	63.77±4.25(200)	61.50±3.96(200)	94.08±3.94(300)	52.20±1.83(200)	39.31±2.96(300)	50.00±2.71(150)

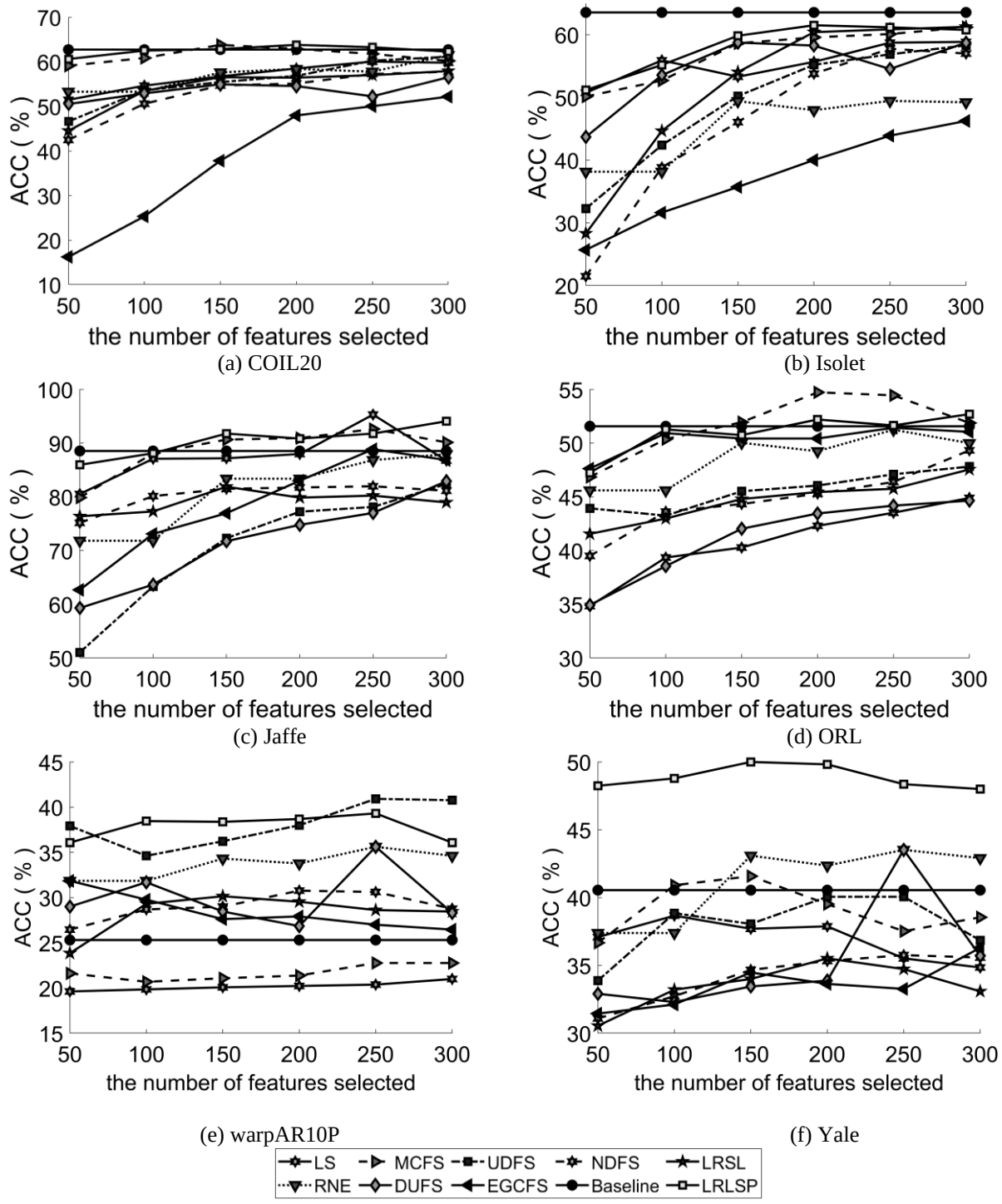
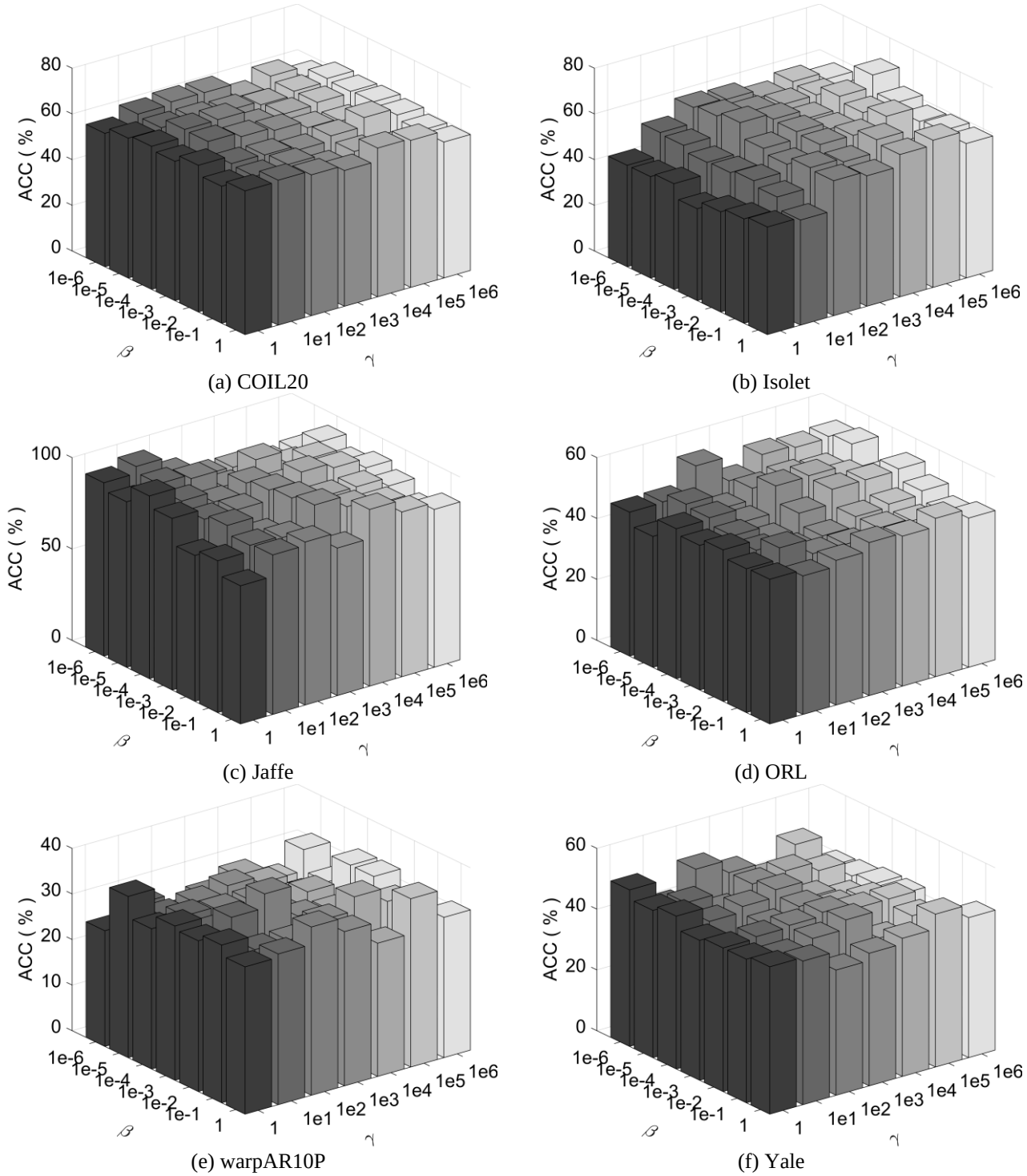


Fig. 2. Clustering accuracy of different algorithms when the number of features takes different values

TABLE III
NORMALIZED MUTUAL INFORMATION OF DIFFERENT FEATURE SELECTION ALGORITHMS ON DIFFERENT DATASETS (NMI ± STD%)

Datasets	COIL20	Isolet	Jaffe	ORL	warpAR10P	Yale
LS	71.88±1.34(250)	74.35±0.63(300)	91.62±6.68(150)	67.58±0.89(300)	22.09±1.72(300)	44.02±1.03(100)
MCFS	73.30±1.75(150)	75.14±1.28(300)	93.17±3.30(250)	74.76±1.55(250)	20.15±2.79(300)	47.79±2.93(150)
UDFS	72.17±1.66(300)	74.21±2.21(300)	82.13±2.98(300)	69.81±1.10(300)	47.13±2.59(250)	47.82±2.17(250)
NDFS	72.13±1.49(300)	70.98±1.49(250)	85.54±2.47(150)	71.07±1.11(300)	29.91±2.23(200)	43.48±1.64(200)
LRSL	72.51±1.50(300)	75.67±0.86(300)	82.83±1.75(250)	70.19±0.73(300)	28.06±2.90(200)	42.50±1.52(200)
DUFS	69.67±1.34(300)	74.98±1.37(150)	82.56±2.28(300)	67.83±1.17(300)	36.80±9.32(250)	43.13±1.32(300)
RNE	73.48±0.92(300)	66.83±1.12(250)	87.75±5.20(250)	72.62±1.04(250)	38.30±3.64(250)	50.71±2.84(250)
EGCFS	68.32±1.39(300)	63.83±1.32(300)	91.92±3.05(300)	73.04±1.22(250)	28.86±3.76(50)	41.53±1.37(150)
Baseline	75.56±1.43(1024)	77.70±0.77(617)	89.28±3.91(676)	72.96±0.97(1024)	24.12±1.79(2400)	46.47±2.05(1024)
LRLSP	74.84±0.70(250)	74.96±1.60(250)	94.13±2.33(300)	73.34±1.58(300)	45.62±2.48(250)	55.13±1.25(150)



(a) COIL20 (b) Isolet

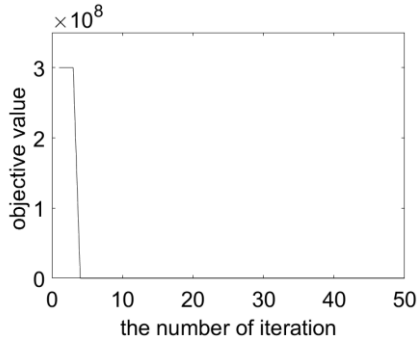
(c) Jaffe (d) ORL

(e) warpAR10P (f) Yale

Fig. 3. Clustering accuracy of proposed algorithm in different parameters value when 300 features are selected

D.Convergence analysis

In this section, we present the convergence analysis of Algorithm 2 based on the value of objective function across iterations. Fig. 4 illustrates the convergence curve of the LRLSP algorithm, where the COIL20 and Isolet datasets are displayed due to significant variations in initial objective function values. From Fig. 4, we can see that the objective function initially exhibits a sharp decline followed by a gradual decrease, ultimately stabilizing within the first six iterations. These results demonstrate that our algorithm achieves effective and rapid convergence.



(a) COIL20

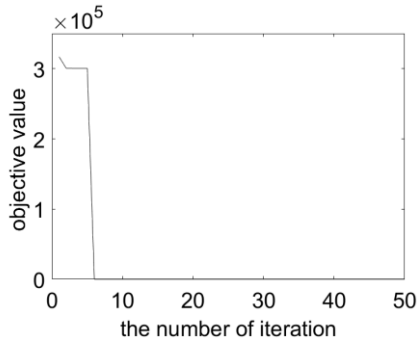


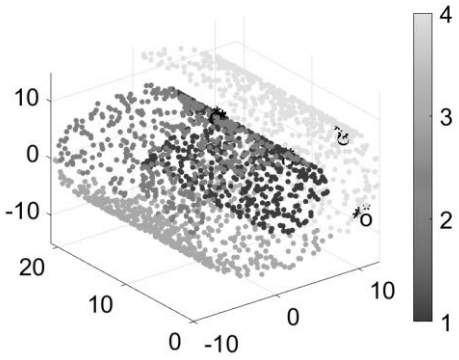
Fig. 4. Algorithm convergence curves of LRLSP

E. Visualization of experiments

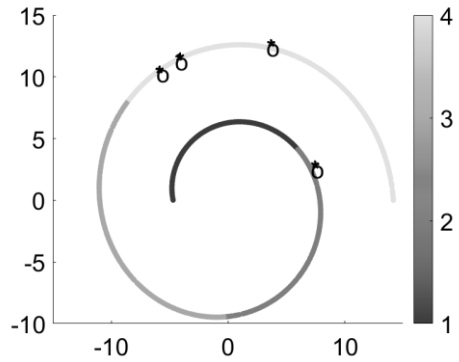
In this section, the visualization experiments on the Swiss Roll datasets are presented in Fig. 5. The primary purpose is to demonstrate that LRLSP has effectively captured a more refined local manifold structure of the data. In Fig. 5(a), we have identified four points from the original Swiss Roll datasets and highlighted them with circles. Additionally, their corresponding three nearest neighbors are marked with asterisks. By comparing both figures, it is evident that the local manifold structure of the datasets remains unchanged when projected into the low-dimensional space.

V. Conclusion

In this paper, a novel robust method LRLSP based on Laplace rank constraint and local structure preservation is proposed for unsupervised feature selection. A weight matrix with a block diagonal structure is obtained by combining locally linear embedding and Laplace rank constraint. And then based on the local manifold and the above weight matrix, the projection matrix is obtained for feature selection. Moreover, a redundancy regularization term of the projection matrix is introduced to learn the redundancy from features, to select more discriminative features. Finally, we add the $L_{2,1}$ norm constraint to the projection matrix to avoid the influence of noise and improve the robustness of the algorithm. A series of experiments demonstrate that the LRLSP algorithm effectively selects more discriminative features.



(a) Swiss Roll raw data scatter plot



(b) Swiss Roll low-dimensional spatial data scatter plot

Fig. 5. Visualization of experiments on the Swiss Roll

References

[1] Yaqing Liu and Xiaokai Yi, Rong Chen, Zhengguo Zhai, and Jingxuan Gu, “Feature Extraction Based on Information Gain and Sequential Pattern for English Question Classification,” IET Software, vol.12, no.6, pp 520-526, 2018.

[2] Yao Zhang, Yingcang Ma, and Xiaofei Yang, “Multi-Label Feature Selection Based on Logistic Regression and Manifold Learning,” Applied Intelligence, vol.52, no.8, pp9256-9273, 2022.

[3] B Tenenbaum Joshua, de Silva Vin, and C Langford John, “A Global Geometric Framework for Nonlinear Dimensionality Reduction,” Science, vol.290, no.5500, pp2319-2323, 2000.

[4] Belkin Mikhail and Niyogi Partha, “Laplacian Eigenmaps for Dimensionality Reduction and Data Representation,” Neural Computation, vol.15, no.6, pp1373-1396, 2003.

[5] Sam T Roweis and Lawrence K Saul, “Nonlinear Dimensionality Reduction by Locally Linear Embedding,” Science, vol.290, no.5500, pp2323-2326, 2000.

[6] Xinwang Liu, Lei Wang, Jian Zhang, Jianping Yin, and Huan Liu, “Global and Local Structure Preservation for Feature Selection,” IEEE Transactions on Neural Networks and Learning Systems, vol.25, no.6, pp1083-1095, 2014.

[7] Yu Peng, Yong Xu, Datong Liu, and Junbao Li, “Locality Structure Preserving Based Feature Selection for Prognostics,” Intelligent Data Analysis, vol.19, no.3, pp659-682, 2015.

[8] Yanfang Liu, Dongyi Ye, Wenbin Li, Huihui Wang, and Yang Gao, “Robust Neighborhood Embedding for Unsupervised Feature Selection,” Knowledge-Based Systems, vol.193, pp105462, 2020.

[9] Nan Zhou, Yangyang Xu, Hong Cheng, Jun Fang, and Witold Pedrycz, “Global and Local Structure Preserving Sparse Subspace Learning: An Iterative Approach to Unsupervised Feature Selection,” Pattern Recognition, vol.53, pp87-101, 2016.

[10] Shiping Wang and William Zhu, "Sparse Graph Embedding Unsupervised Feature Selection," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol.48, no.3, pp329-341, 2018.
[11] Jiansheng Wu, Mengxiao Song, Weidong Min, Jianhuang Lai, and Weishi Zheng, "Joint Adaptive Manifold and Embedding Learning for Unsupervised Feature Selection," Pattern Recognition, vol.112, pp107742, 2021.
[12] Guoqiu Wen, Yonghua Zhu, Mengmeng Zhan, and Malong Tan, "Sparse Low-Rank and Graph Structure Learning for Supervised Feature Selection," Neural Processing Letters, vol.52, no.3, pp1793-1809, 2020.
[13] Shiping Wang and Han Wang, "Unsupervised Feature Selection Via Low-Rank Approximation and Structure Learning," Knowledge-Based Systems, vol.124, pp70-79, 2017.
[14] Feiping Nie, Xiaoqian Wang, Michael Jordan, and Heng Huang, "The Constrained Laplacian Rank Algorithm for Graph-Based Clustering," in Proc. of the 13th AAAI Conference on Artificial Intelligence, USA, Feb, vol.30, no.1, pp1969-1976, 2016.
[15] Hyunki Lim and DaeWon Kim, "Pairwise Dependence-Based Unsupervised Feature Selection," Pattern Recognition, vol.111, pp107663, 2021.
[16] Xueyuan Xu, Xia Wu, Fulin Wei, Wei Zhong, and Feiping Nie, "A General Framework for Feature Selection under Orthogonal Regression with Global Redundancy Minimization," IEEE Transactions on Knowledge and Data Engineering, vol.34, no.11, pp5056-5069, 2022.
[17] Hao Li, Yongli Wang, Yanchao Li, Peng Hu, and Ruxin Zhao, "Joint Local Structure Preservation and Redundancy Minimization for Unsupervised Feature Selection," Applied Intelligence, vol.50, no.12, pp4394-4411, 2020.
[18] Wang Guoyin Ma, Xi'ao, Hong Yu, Tianrui Li, "Decision Region Distribution Preservation Reduction in Decision-Theoretic Rough Set Model," Information Sciences, vol.278, pp614-640, 2014.
[19] Nan Zhang, Xueyi Gao, and Tianyou Yu, "Heuristic Approaches to Attribute Reduction for Generalized Decision Preservation," Applied Sciences, vol.9, no.14, pp2841, 2019.
[20] Santiago Gonzalez Zerbo, Alejandra Maestripieripieri, and Francisco Martinez Peria, "Linear Pencils and Quadratic Programming Problems with A Quadratic Constraint," Linear Algebra and Its Applications, vol.665, pp12-35, 2023.
[21] Feiping Nie, Heng Huang, Xiao Cai, and Chris Ding, "Efficient and Robust Feature Selection via Joint L2, 1-Norms Minimization," in International Conference on Neural Information Processing Systems, vol.23, 2010.
[22] Yao Zhang and Yingcang Ma, "Nonnegative Multi-Label Feature Selection with Dynamic Graph Constraints," Knowledge-Based Systems, vol.238, pp107924, 2022.
[23] Xiaofei He, Deng Cai, and Partha Niyogi, "Laplacian Score for Feature Selection," in Proc. of the 18th International Conference on Neural Information Processing Systems, Canada, pp507-514, 2005.
[24] Deng Cai, Chiyuan Zhang, and Xiaofei He, "Unsupervised Feature Selection for Multi-Cluster Data," in Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, USA, pp333-342, 2010.
[25] Yi Yang, Hengtao Shen, Zhigang Ma, Zi Huang, and Xiaofang Zhou, "L2,1-Norm Regularized Discriminative Feature Selection for Unsupervised Learning," in Proc. of the 22nd International Joint Conference on Artificial Intelligence, Spain, pp1589-1594, 2011.
[26] Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, and Hanqing Lu, "Unsupervised Feature Selection Using Nonnegative Spectral Analysis," in Proc. of the 26th AAAI Conference on Artificial Intelligence, Canada, pp1026-1032, 2012.
[27] Rui Zhang, Yunxing Zhang, and Xuelong Li, "Unsupervised Feature Selection via Adaptive Graph Learning and Constraint," IEEE Transactions on Neural Networks and Learning Systems, vol.33, no.3, pp1355-1362, 2022.

Yingying Meng acquired a B.S. degree in applied mathematics from Shangluo University, China, in 2020, and an M.S. degree in mathematics from Xi'an Polytechnic University, China, in 2023. Her main research interests include machine learning and so on.

Qiaoyan Li acquired her B.S. Degree in computational mathematics from Northwest University, China, in 2000. Her M.S. degree in applied mathematics from Xi'an Polytechnic University, China, in 2007. Currently, she is an associate professor at the School of Science of Xi'an Polytechnic University. Her main research interests include statistic learning, fuzzy logic, and neutrosophic set theory.

Xiaofei Yang acquired a B.S. degree in applied mathematics from Luoyang Normal University, China, in 2006, an M.S. degree in pure mathematics from Shaanxi Normal University, China, in 2009, and a Ph.D. degree in pure mathematics from Shaanxi Normal University, China, in 2012. His research interests include data mining, image processing, and fuzzy mathematics.

Xuezhen Dai acquired her B.S. degree in mathematics and applied mathematics from Xi'an Polytechnic University in 2009, and her M.S. degree in applied mathematics from Xi'an Polytechnic University in 2012. Currently, she is an associate professor at Xi'an Traffic Engineering Institute. Her main research interests include statistic learning, fuzzy logic, and intelligent computing.