Clustering Preserving Projections for High-Dimensional Data

Weiling Cai, Honghan Zhou, Le Xu
Department of Computer Science and Technology, Nanjing Normal University, Nanjing 210097, China
caiwl@nuaa.edu.cn, caiwl@njnu.edu.cn

Abstract. In this paper, a novel clustering preserving projection method is presented to retain the cluster (or group) structures hidden in the original data. This method involves two steps: (1) the famous clustering method Fuzzy c-means is utilized to discover the cluster distribution in data; (2) such distribution is preserved by finding a linear embedding that minimizes the intra-cluster compactness in low-dimensional space. The feasibility and effectiveness of the proposed method are demonstrated on UCI dataset and USPS digital handwritten dataset.

1. Introduction
Dimensionality reduction is a traditional problem in machine learning. It has been found important in many domains such as human visual perception, text categorization, and face recognition. The goal of dimensionality reduction is to project high-dimensional data into low space so that the result performs better in further processing such as clustering, classification, indexing and searching.

The techniques for dimensionality reduction can be roughly divided into two main types [1]: (1) techniques that attempt to preserve global properties of the original data in the low dimensional representation, (2) techniques that attempt to preserve local properties of the original data in the low-dimensional representation.

The classical techniques belonging to the first type involve principle component analysis (PCA) [2] and linear discriminant analysis (LDA) [2], which are by far the most popular unsupervised and supervised linear technique, respectively. Using the kernel trick, kernel principal components analysis (KPCA) [3] and kernel discriminant analysis (KDA) [4] were developed to extract a group of new nonlinear features. In recent years, several new dimensionality reduction methods have been introduced, such as ISOMAP [5], and Multi-dimensional Scaling (MDS) [6]. All of these methods attempt to preserve solely the global properties of the data [1], but pay no attention to the neighborhood structure or cluster structure of data.

In addition, some dimensionality reduction methods aim to preserve the local properties of the data. Commonly used methods include Locality Preserving Projections (LPP) [7], Locally Linear Embedding (LLE) [8], Neighborhood Preserving Embedding (NPE) [9], and Laplacian eigenmaps [10]. In these methods, the definition of neighborhood plays an important role. Each of these methods performs well when the data belong to a single well-sampled cluster. They fail when the data are spread among multiple clusters. When data points are organized into multiple clusters, all data points in one cluster may not be able to find neighbors in other clusters and the neighborhood graph of these points does not connect to the neighborhood graph of data points in the other clusters. Therefore, these methods fail to project clustered data into a single low-dimensional coordinate system [11].

The above-mentioned techniques all neglect the cluster distribution of the original data, as a result, the cluster structures are possibly destroyed in the low-dimensional representation. Different from
these existing algorithms, in this paper, a new linear dimensionality reduction method called clustering preserving projections (CPP) is proposed to preserve the cluster (or group) structure information hidden in data. In CPP, the classical clustering analysis method Fuzzy c-means (FCM) is adopted to discover the cluster structure of data, and then a linear embedding is computed to preserve the cluster distribution to the reduced space. CPP looks forward to project the data onto a lower dimensionality in which the intra-cluster compactness will be minimized. Compared to PCA and LPP, the proposed algorithm can preserve the cluster distribution of the data in a lower dimensional space more effectively. The feasibility and effectiveness of the proposed algorithm are demonstrated.

2. Related Work

The generic problem of linear dimensionality reduction can be described as follows. Given a set $X=\{x_1, x_2, \ldots, x_N\}$ where $x_k \in \mathbb{R}^D$, find a transformation matrix $W$ that maps these $N$ points to a set of points $Y=\{y_1, y_2, \ldots, y_N\}$, $y_k \in \mathbb{R}^d$, such that $y_k$ ‘represents’ $x_k$, where $y_k=W^T x_k$.

PCA is a technique that is widely used for dimensionality reduction. It is also known as the Karhunen-Loève transform. Its main idea is to project the data along the directions of maximal variance.

Assume the projection onto a one-dimensional space, each data point is projected onto a value of $w^T x_i$ where $w \in \mathbb{R}^D$. The objective function of PCA is formulated as

$$\max \sum_{i=1}^{N} (w^T x_i - w^T \bar{x})^2 / N = w^T S w$$

subject to $w^T w = 1$ (1)

where $\bar{x}$ is the mean of the data and $S$ is the data covariance matrix. The optimal linear projection for which the variance of the projected data is maximized is defined by the $d$ eigenvectors of the data covariance matrix $S$ corresponding to the largest eigenvalues.

LPP is a linear projection algorithm that preserves the neighborhood structure of the data. It builds a graph incorporating neighborhood information of the data set. Using the notion of the Laplacian of the graph, a transformation matrix which maps the data points to a subspace is obtained.

When $x_i$ and $x_j$ satisfy $\|x_i - x_j\|^2 < \varepsilon$ (parameter $\varepsilon \in \mathbb{R}$) or $x_i$ is among $k$ nearest neighbors of $x_j$, the weight $a_{ij}$ between $x_i$ and $x_j$ is calculated by

$$a_{ij} = e^{\frac{-\|x_i - x_j\|^2}{\varepsilon}}$$

otherwise $a_{ij}$ is set to 0. Suppose $w$ is a transformation vector, that is, $y_i^T = w^T x_i$, the objective function of LPP is defined as

$$\min \sum_{i,j} (y_i - y_j)^2 a_{ij}$$

where $a_{ij}$ will incur a heavy penalty if neighboring points $x_i$ and $x_j$ are far apart.

3. Clustering Preserving Projections

One of the central problems in machine learning and pattern recognition is to develop appropriate representations for complex data. Some classical technique such as PCA and LDA pay attention to the global structure information of the data; on the other hand, many other state-of-art algorithms such as LPP and NPE aims to preserve the local neighborhood structure of the data. Different from these existing algorithms, in this paper, a new dimensionality reduction method called clustering preserving projections (CPP) is proposed to preserve the cluster (or group) structure hidden in data.

3.1. Clustering Analysis

In this paper, Fuzzy c-means (FCM) [2] is selected for two reasons: (1) its outputs allow the sample to belong to multiple clusters with varying degrees of membership, so FCM can get much more
information from dataset than the hard clustering methods.; (2) it performs in an unsupervised manner (class labels are not used), the number of prototypes in FCM is independent of the number of classes, as a result, the obtained fuzzy partition of the data can more closely represent the underlying structures. Specifically, FCM explores the structures in data by partitioning the samples into groups (or clusters) and its objective function can be formulated as:

\[ J_{FCM}(U, V) = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \| x_i - v_k \|^2 \]

(4)

where \( c \) is the number of clusters, \( V=[v_1, v_2, \ldots, v_c] \) are the cluster centers, and the fuzzy matrix \( U=(u_{ik})_{c \times n} \) makes up of the fuzzy memberships of the each training sample \( x_i \) to each cluster \( v_j \). The parameter \( m (1 \leq m < \infty) \) is a weighting exponent on each fuzzy membership that determines the amount of fuzziness of the resulting classification. In the following experiment, the value of \( m \) is set to 2. An iterative algorithm for minimizing (4) with respect to \( u_{ik} \) and \( v_i \) can be derived as shown in (5):

\[ u_{ik} = \frac{\| x_i - v_k \|^2}{\sum_{j=1}^{c} \left( \| x_i - v_j \|^2 \right)^{(m-1)/2}} \]

\[ v_i = \frac{\sum_{k=1}^{n} u_{ik}^m x_k}{\sum_{k=1}^{n} u_{ik}^m} \]

(5)

3.2. Clustering Preservation

In fact, the objective function of FCM aims to minimize the intra-cluster compactness in the original space. Motivated by this, CPP looks forward to preserve the intra-cluster compactness in the reduced lower space. Consider a set \( X=[x_1, x_2, \ldots, x_N] \), assume its clustering partition \( U \) and clustering centers \( V \) are calculated by FCM. The goal of CPP is to find a transformation matrix \( W \) that minimizes the following objective function:

\[ J_{CPP}(W) = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \| W^T x_k - W^T v_i \|^2 \]

(6)

Here the cluster centers \( v_i \) is computed by (9) and can be further rewritten as

\[ v_i = \sum_{k=1}^{N} b_{ik} x_k = Xb_i \]

(7)

where \( b_{ik} = u_{ik}^m / \sum_{k=1}^{N} u_{ik}^m \) and \( b_i=[b_{i1}, b_{i2}, \ldots, b_{iN}]^T \). Using (7), the (6) can be translated into:

\[ J_{CPP}(W) = tr \left( \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \left( W^T x_k - W^T v_i \right) \left( W^T x_k - W^T v_i \right)^T \right) \]

\[ = tr \left( W^T \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \left( x_k - Xb_i \right) \left( x_k - Xb_i \right)^T W \right) \]

(8)

To make the objective function concise, (6) can be further written as
\[
J_{\text{CPP}}(W) = \text{tr} \left\{ W^T \left[ \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik}^m)^m x_i x_k^T + \sum_{i=1}^{n} \sum_{k=1}^{n} (u_{ik})^m X b_i^T X^T - 2 \sum_{i=1}^{N} \sum_{k=1}^{n} (u_{ik})^m x_i b_i^T X^T \right] W \right\} \\
= \text{tr} \left\{ \left( W^T X \right) \text{diag} \left( \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik}^m)^m \right) + B \text{diag} \left( \sum_{k=1}^{n} (u_{ik})^m \right) B^T - 2BZ^T \right\} \left( W^T X \right)^T \\
= \text{tr} \left\{ \left( W^T X \right) \text{diag} \left( \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik}^m)^m \right) - BZ^T \right\} \left( W^T X \right)^T \\
\]

where \( B = [b_1, b_2, ..., b_c] \subseteq R^{N \times c} \), \( Z = [z_1, z_2, ..., z_c] \subseteq R^{N \times c} \) and \( z_i = [u_{i1}^m, u_{i2}^m, ..., u_{iN}^m]^T \). By defining

\[
D = \text{diag} \left( \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik}^m)^m \right), S = BZ^T \text{ and } L = D - S, \\
\]

the objective function is simplified to:

\[
J_{\text{CPP}}(W) = \text{tr} \left\{ \left( W^T X \right) (D - S) X W^T \right\} \\
= \text{tr} \left\{ W^T XLX^T W \right\} \\
st : YDY^T = W^T XDXW^T = I
\]

It is interesting to observe that the final objective function of our algorithm is similar to that of LPP in their formulation. In (3) of LPP, the matrix \( A \) represents the near relationship among samples. Specifically, when the sample \( x_i \) is near to \( x_j \), the value of weight \( a_{ij} \) is small, and vice versa. Corresponding to such matrix \( A \), there is also a matrix \( S \) in our algorithm. Can we gain some insights from this matrix?

In fact, the element of \( S \) measures the possibility of samples falling into the same cluster or group. For example, suppose the cluster membership of a given sample \( x_i \) is a hard value, that is to say, the element of \( u_{i} \) can be represented as:

\[
u_{pi} = \begin{cases} 
1 & x_i \text{ belongs to the } p\text{th cluster} \\
0 & x_i \text{ does not belong to the } p\text{th cluster}
\end{cases}
\]

Then, let us consider a dataset consisting of five clusters. When the samples \( x_i \) and \( x_j \) belong to the same cluster (the 2nd cluster), their cluster memberships are \( u_{i} = u_{j} = [0 \\ 1 \\ 0 \\ 0 \\ 0] \), and then the value of \( S_{ij} \) is \( 1/\sum_{k=1}^{n} u_{pk}^m \); on the other hand, when the samples \( x_i \) and \( x_j \) belong to the different clusters (the 1st and 3rd cluster, respectively), their cluster memberships are \( u_{i} = [1 \\ 0 \\ 0 \\ 0 \\ 0] \) and \( u_{j} = [0 \\ 0 \\ 1 \\ 0 \\ 0] \), respectively, and then the value of \( S_{ij} \) is equal to 0. From this example, we can observe that the value
of $S_i$ can indeed reflect the possibility of samples belonging into the same cluster, so we name $S$ the cluster-similarity matrix.

To summarize, our projection algorithm can be described in Fig. 1.

| Step 1: Perform FCM on the original space and calculate the partition matrix $U$ according to (5). |
| Step 2: Compute the matrix $S$, $D$ and $L$ in terms of (10) and (12), respectively. |
| Step 3: Compute the eigenvectors and eigenvalues for the generalized eigenvector problem: $L'w = \lambda D'w$ |

Fig. 1 Algorithm of CPP

4. Experiments

4.1. Experiment Setup

To testify whether the presented method can effectively preserve the cluster structure of data, we use a typical clustering method K-means to evaluate the clustering accuracy on the original space and the low-dimensional space, respectively. If the clustering result obtained in low-dimensional space performances better than in original space, the effectiveness of CPP will be clearly demonstrated. To further investigate the superiority of CPP, we compare CPP with the typical global technique PCA and local technique LPP.

All above comparisons have been made on two classical datasets. Iris is a real-life dataset cited from the UCI Machine Learning Repository. This dataset consists of 150 samples (50 in each of three clusters) and each sample has four dimensions. USPS is handwritten 16x16 digits dataset. The digits 1 and 2 are used in our experiments as the two clusters. There are 1100 examples for each cluster, for a total of 2200. In our experiment, these two datasets have been projected to two dimensional spaces.

4.2. Experiment Result

Table 1 gives the clustering accuracies obtained on the original datasets and the clustering-preserving projected datasets, respectively. The clustering accuracies of 97.33% and 99.32% on the low-dimensional representations of Iris and USPS datasets are respectively higher than those of 88.00% and 98.59% on the original datasets. These results indicate that the cluster structure hidden in the data has been effectively preserved in the low-dimensional space, thus demonstrating the effectiveness of the presented dimension reduction method CPP.

Table 1 Comparison of clustering accuracies obtained on original space and reduced space

| Dataset | Kmeans | CPP+ Kmeans |
|---------|--------|-------------|
| Iris    | 88.00% | 97.33%      |
| USPS    | 98.59% | 99.32%      |

To further validate the effectiveness of the presented method, we utilize PCA, LPP and CPP to project Iris and USPS datasets to the two-dimensional spaces, and then adopt K-means to quantificationally measure the correctness of retained cluster structure information in the projected 2D spaces. Table 2 lists the performances of clustering preserving achieved by PCA, LPP and CPP, respectively. From this table, it can be observed that CPP can achieve much better projecting results than PCA and LPP in preserving the cluster structure. The underlying reason is that PCA and LPP all pay no attention to the cluster distribution of data, so the group information is unlikely to be preserved; in contrast, CPP can successfully retain the local cluster distribution of data.

Table 2 Comparison of clustering accuracies obtained by PCA, LPP and CPP

| Dataset | PCA+ Kmeans | LPP+ Kmeans | CPP+ Kmeans |
|---------|-------------|-------------|-------------|
| Iris    | 88.67%      | 86.67%      | 97.33%      |
| USPS    | 98.14%      | 50.27%      | 99.32%      |
Fig. 2 shows a 2D projection of Iris dataset achieved by PCA, LPP and CPP, respectively. The dataset Iris has three clusters: the class Iris Setosa is well separated from the other two clusters; the class Iris Versicolor and Iris Virginica are not easily separated from each other. In these figures, the characters ‘+’, ‘.’ and ‘*’ represent the data samples from three different clusters, respectively. The character ‘O’ denotes a misclassified sample. In Fig.2(a) and (b), many samples falling into the overlap area between cluster Versicolor and Virginica have been misclustered, these results indicating that PCA and CPP do not care about the cluster structure of data; in contrast, in Fig. 2(c), only four samples have been misclustered, so it is clear that CPP can effectively retain the cluster structure in the processing of dimension reduction.

5. Conclusion
In this paper, we propose a clustering preserving projection method for dimension reduction. This method can retain the cluster (or group) structures hidden in the original data. This method employs Fuzzy c-means to discover the cluster structure in data, and then calculates a linear embedding to minimize the intra-cluster compactness in low-dimensional space. The experiments on UCI dataset and USPS digital handwritten dataset demonstrate the feasibility and effectiveness of the proposed method.

References
[1] Van Der Maaten, E. O. Postma, H. J. Van Den Herik, Dimensionality Reduction: A Comparative Review, Machine Learning, 2009.
[2] A Shin, et al. A dimension-reduction based multilayer perception method for supporting the medical decision making. Pattern Recognition Letters, 2020, 131:15-22.
[3] R. Duda, P. Hart, D. Stork, Pattern Classification, 2nd ed. John Wiley & Sons, NY, 2001.
[4] B. Scholkopf, A. Smola, and K. Muller, Kernel Principal Component Analysis, Advances in Kernel Methods-Support Vector Learning, B. Scholkopf, C. Burges, and A. Smola, Ed. Cambridge, MA: MIT Press, pp. 327-352, 1999.
[5] Z Xu, J Liu, X Luo, et al. Cross-version defect prediction via hybrid active learning with kernel principal component analysis. IEEE International Conference on Software Analysis. IEEE, 2018.
[6] S. Yan, D. Xu, B. Zhang et al., Graph Embedding and Extensions: A General Framework for Dimensionality Reduction, IEEE Trans. on Pattern Anal. Mach. Intell., vol. 29, no. 1, pp. 40-51, 2007.
[7] S. Zhong, W. Xiao, W. Tian. Time series semi-supervised classification based on Locality Preserving Projections. Intelligent Computer and Applications, 2019.
[8] Y. Zhang et al. Robust locally linear embedding algorithm for machinery fault diagnosis[J]. Neurocomputing, 2018.
[9] W. Wang, V. Aggarwal, S. Aeron. Tensor Train Neighborhood Preserving Embedding[J]. IEEE Transactions on Signal Processing, 2018:1-1.
[10] M. Belkin and P. Niyogi, “Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering,” Advances in Neural Information Processing Systems 14, Vancouver, British Columbia, Canada, 2002.

[11] L. Yang, Distance-Preserving Projection of High-Dimensional Data for Nonlinear Dimensionality Reduction, IEEE Trans. on Pattern Anal. Mach. Intell., vol. 26, no. 9, pp. 1243-1246, 2004.