Dimensional reduction methods for face dataset (CSIC 5011 Final project)

ZHONG Guangzheng
22482032@life.hkbu.edu.hk

HKBU

Table of Contents

Diffusion map

Multidimensional scaling (MDS)

Isomap

Locally-linear Embedding (LLE)

Local tangent space alignment (LTSA)

2-D t-SNE

Conclusion

Diffusion map

Diffusion map

- We explore the diffusion map on the face dataset. Particularly, we construct the weight of map W_{ij} as follows

$$W_{ij} = e^{\frac{-\|x_i - x_j\|^2}{t}}. \tag{1}$$

- Then we define the degree D as

$$D = diag(\sum_j W_{ij}). \tag{2}$$

- And we obtain the diffusion matrix L by

$$L = D^{-1}W - I. \tag{3}$$

- The largest eigenvalue of the diffusion matrix L is 0, i.e., $\lambda_0 = 0$. We consider the second-largest eigenvalue λ_1, and its corresponding eigenvector v_1. We sort the faces by the values of $v_1(i), i = 1, \cdots, n$.

Result

(a) $t = 1$

(b) $t = 10$

(c) $t = 100$

Figure: Diffusion Map on Face dataset with different parameter value t.

Multidimensional scaling (MDS)

Multidimensional scaling (MDS)

In this project, we consider the metric multidimensional scaling (mMDS). The optimization process is generalized to a number of loss functions and input matrices of known distances with weights and other factors. The loss function, which is a residual sum of squares. Specifically, we optimize the following optimization problem.

$$\min_{x_1, x_2, \cdots, x_n} \sqrt{\sum_{i \neq j = 1, \ldots, N} \left(d_{ij} - \|x_i - x_j\|\right)^2} \tag{4}$$

where d_{ij} is the distance between the i^{th} data and the j^{th} data. In this project, we apply the Euclidean distance, i.e., $d_{ij} = \|z_i - z_j\|^2$, where z_i is the original data.

Result

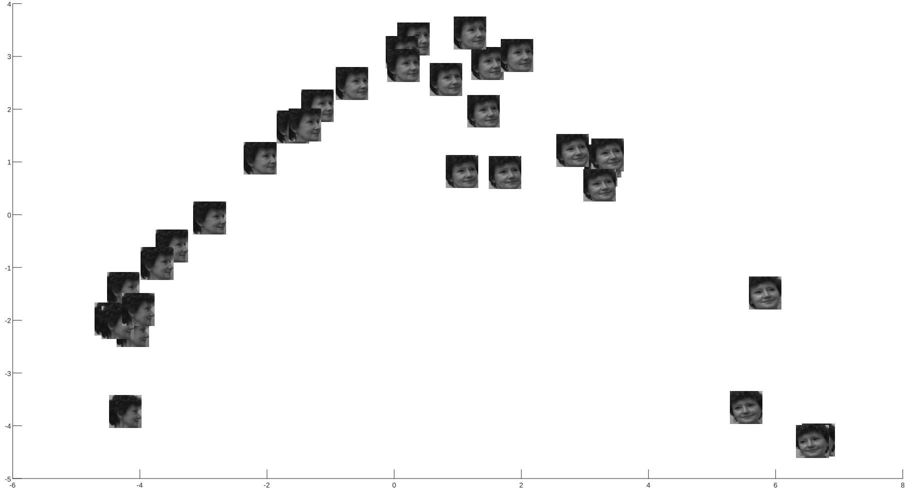

Figure: MDS on Face dataset.

Isomap

isomap

The isomap is composed with 4 steps.

1. The neighbors of each point are determined.
2. A neighborhood graph is constructed by considering K nearest neighbor.
3. The shortest path between any two data is calculated.
4. MDS method is performed, where the distance d_{ij} in (4) is the shortest path between the i^{th} data and the j^{th} data.

Result

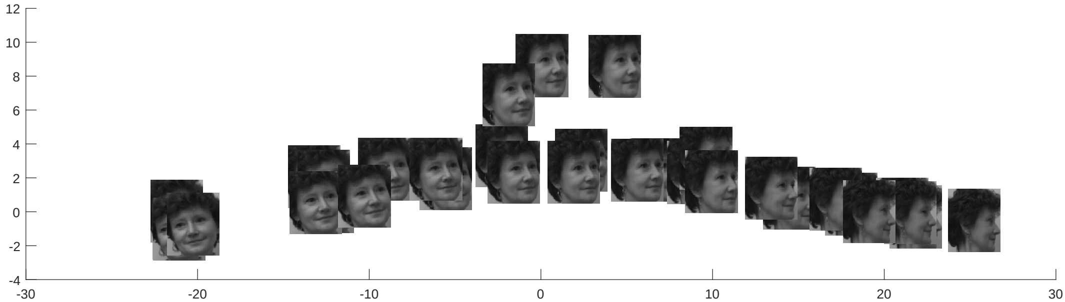

Figure: Isomap on Face dataset with $K = 5$.

Locally-linear Embedding (LLE)

Locally-linear Embedding (LLE)

In comparison to isomap, locally-linear embedding (LLE) offers a number of benefits, such as faster optimization when using sparse matrix algorithms and superior outcomes for a variety of issues.

Finding each point's closest neighbors is the main idea of LLE, which is the same as isomap. For each point, a set of weights that best describes the point as a linear combination of its neighbors is computed. The low-dimensional embedding of the points is then discovered using an eigenvector-based optimization technique, ensuring that each point is still described by the same linear combination of its neighbors.

LLE typically does not handle non-uniform sample densities well because there is no fixed unit to prevent the weights from drifting as different regions have different sample densities.

Result

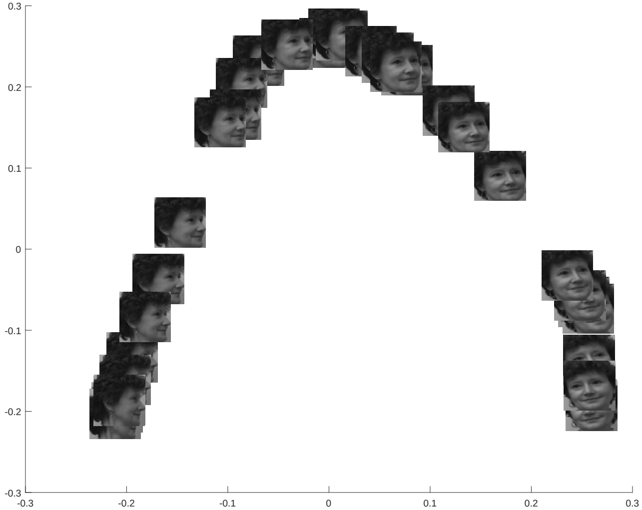

Figure: LLE on Face dataset with $K = 5$.

Local tangent space alignment (LTSA)

Local tangent space alignment (LTSA)

LTSA method can be broken down into the following steps:

1. Compute the k-nearest neighbors for each point in the high-dimensional data.
2. For each point, compute the local tangent space by computing the d-first principal components of its k-nearest neighbors.
3. Compute the weight matrix W, where W_{ij} is the weight assigned to the edge between points i and j. This weight is proportional to the squared distance between the points in the tangent space.
4. Compute the graph Laplacian $L = D - W$, where D is the diagonal matrix of row sums of W.
5. Compute the d-first eigenvectors of L, excluding the first eigenvector (which corresponds to the trivial constant solution). These eigenvectors form the new low-dimensional representation of the data.

Result

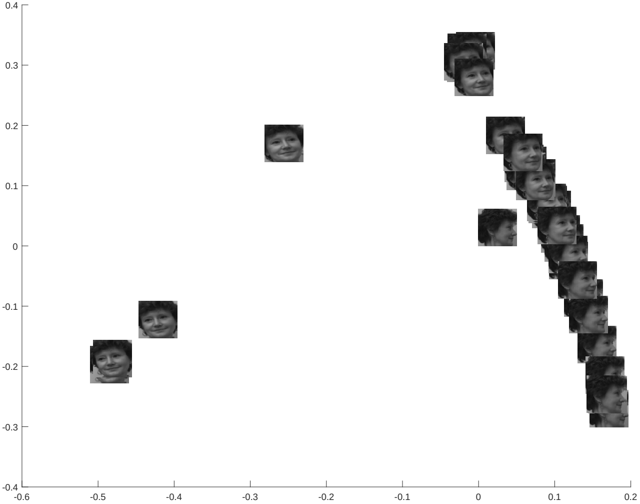

Figure: LTSA on Face dataset with $K = 5$.

2-D t-SNE

2-D t-SNE

In t-SNE, the high-dimensional data is represented by a Gaussian distribution, and the low-dimensional data is represented by a Student's t-distribution.

The Gaussian distribution is used to calculate the similarity between data points in the high-dimensional space, while the student's t-distribution is used to calculate the similarity between data points in the low-dimensional space.

The application of the student's t-distribution in the low-dimensional space helps to prevent the crowding problem, which is a common issue in other dimensionality reduction techniques such as PCA.

Result

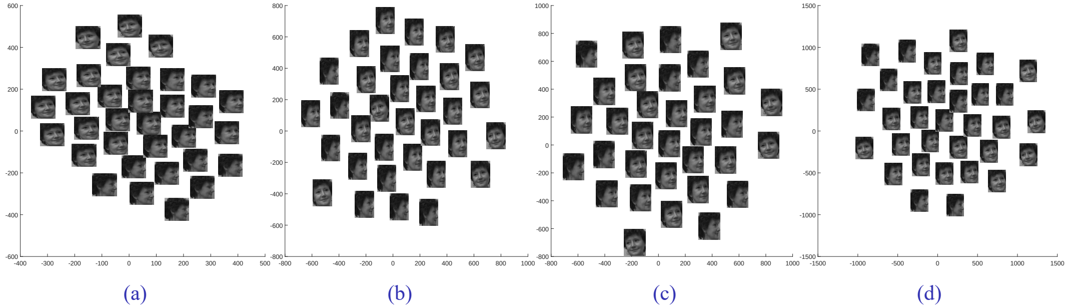

Figure: t-SNE on Face dataset.

Conclusion

Conclusion and future work

1. In this project, we investigate face dataset with dimensional reduction methods, including diffusion map, MDS, isomap, LLE, LTSA and t-SNE. The resulting mapped data are visualized and compared with each other. The analysis of each method is also presented.
2. The dimensional reduction methods that we investigate can be only conducted on existing data. In the future, a mapping function for image dimensional reduction could be considered. With a mapping function, the new data (or testing data) can be mapped into low dimension space and be classified by appropriate machine learning methods.

Thank you!