Subspace Clustering

Weiwei Feng

December 11, 2015

Abstract

Data structure analysis is an important basis of machine learning and data science, which is now widely used in computational visualization problems, e.g. facial recognition, image classification, and motion segmentation. In this project, I would like to deal with a set of small classification problems and use methods like PCA, spectral analysis, kmanifold, etc. By exploring different methods, I would try to prove both mathematically and in application that different problems need to be treated differently, and use their case specific methods.

Keywords. independent subspace, nonlinear subspace, PCA, clustering analysis, Kmanifold algorithm, KNN,SSC,LSC

1 problem background and analysis

A successful analysis on manifold data analysis is based on appropriately dealt with its data structure, and design of specific algorithm. As one of the main topics on manifold learning, subspace clustering has bright prospect in application. According to different assumptions, it is classified as Sparse Subspace Clustering(SSC) and Low-rank Subspace Clustering (LSC). First of all, the two methods both assume that the observed data satisfy

$$X = X_0 + E \in \mathbb{R}^{d\times n},$$

there are n d dimensional data points. X_0 is clean (non-noised) data, E is noise or extreme value indispensable in the data collection process. The first assumption of subspace clustering is to assume that data points can be expressed using a few vector bases. Mathematically, we can write the assumption as

$$X_0 = AZ,$$

here A is a matrix, Z is sparse matrix. In the following algorithm, we use $A = X_O$, which implies that the data is auto-representative. The difference between two methods is that SSC assumes the matrix Z is sparse, i.e. there are only a few effective representative vectors. While LSC assumes that the matrix Z is low-rank. So for SSC, we have

$$\begin{array}{ll}\text{minimize} & \frac{1}{2}\|X - XZ\|_F^2 + \lambda\|Z\|_1 \\ \text{subjet to} & \text{diag}(Z) = 0.\end{array}$$

and for LSC, we have

$$\text{minimize} \quad \|Z\|_\star + \lambda\|X - XZ\|_1.$$

The Kmanifold algorithm, which I used a lot in this project, is based on Isomap and EM algorithm. Here is the realization of Kmanifold algorithm.

Algorithm 1 Kmanifold

Require: A set of images $\{X_1, X_2, \ldots, X_n\}$ and a desired label set size, k

1: Calculate the Euclidean distance between any of the two points in the data set
2: Create a graph G with a node for each image, and an edge between pairs of neighboring images and set the edge weight to that distance
3: Compute all pairs shortest path distances on G. Let $d(i, j)$ be the length of the shortest path between node i and j.
4: Initialization: Create a $k \times n$ matrix W where $w_c i$ is the probability of X_i belonging to manifold c. W can be inititalized randomly unless domain-specific priors are available.
5: M-Step: For each class, c, supply w_c and D and use node-weighted MDS to embed the points in the desired dimension.
6: E-Step: For each point, estimate the distance to the manifolds implied by the output of Step 5 and re-weight accordingly.
7: Go to Step 5 until convergence.

2 problem solving

Here I first tried to dig into the data structure of each problem, and then applied methods which are relatively robust, easy, and computationally feasible.

2.1 Problem I

Data of Problem I are from two independent subspaces. From the geometric point of view, linear subspace are points, lines, planes,etc through the origin. Perpendicular linear subspaces imply that the two spaces are orthogonal. Hence, from the linear point of view, independent subspace are easy to differentiate. Common method like Principle Component Analysis (PCA) are suitable for solving this kind of problems, besides, PCA requires few computation. After obtaining the principle components, using simple clustering method, like K-means method, we are capable to separate data fro different subspaces.

I did PCA on the data, and projected it onto the two dimensional space. See graph 1 (a). Obviously, data are well classified to two classes, and using K-means method, I got a result presented in graph 1(b).

2.2 Problem II

For this problem, I used four different dataset which are all no longer simple independent linear classification problems. Hence here, PCA is not suitable. We analyze respectively these problems and then introduce proper algorithm to realize the clustering objective.

2.2.1 Problem II (a)

The difficulty of this problem is that both of the two lines do not go through the origin, therefore it's not a subspace. Use subspace clustering method, will add one more affine, and reduce the

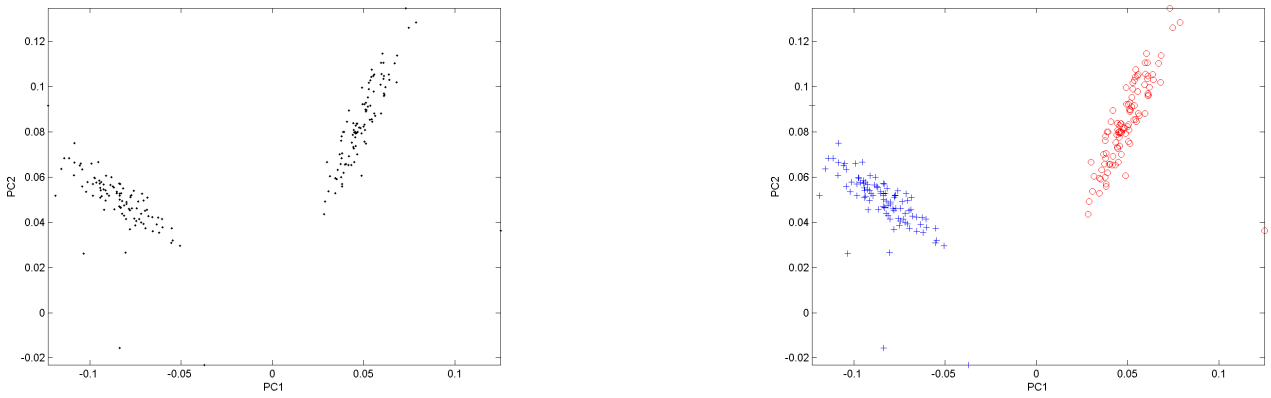

Figure 1: Problem I(1) Original data PCA analysis; (2) K-means method

algorithm efficiency. But we notice that these two lines are both one dimensional manifold, and perpendicularly intercept. Manifold clustering ?Kmanifold) is proper for this kind of problem. The basic idea is to first construct the manifold measurement using Isomap algorithm, then assume the observed data are from low-rank dimensioanl manifold plus noise, construct the probability model, finally get the result using classic EM algorithm. Particularly, due to the perpendicularity, the manifold separation is very clear, so the result of Kmanifold is good.

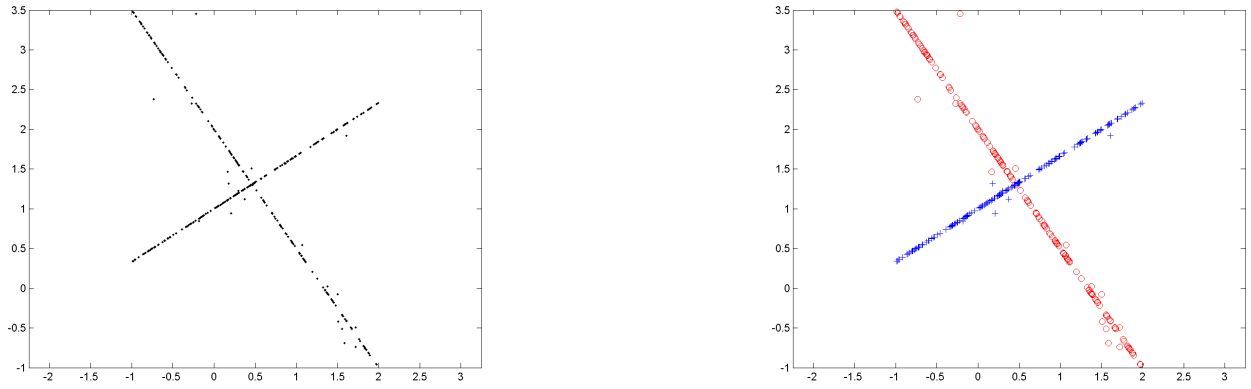

Figure 2: Problem II(a)(1) Original data; (2) Kmanifold method

2.2.2 Problem II (b)

Data for this problem are from manifolds in three dimensional spaces. It looks harder then the previous problem, but all the spaces go through origin, so it perfectly satisfies the assumption of subspace clustering. We have found that lack of independency, SSC has more robust performance than LSC. So here I used SSC.

I took regular parameter $\lambda = 0.1$, similarity matrix $W = \mid Z \mid + \mid Z' \mid$. Results are as follows.

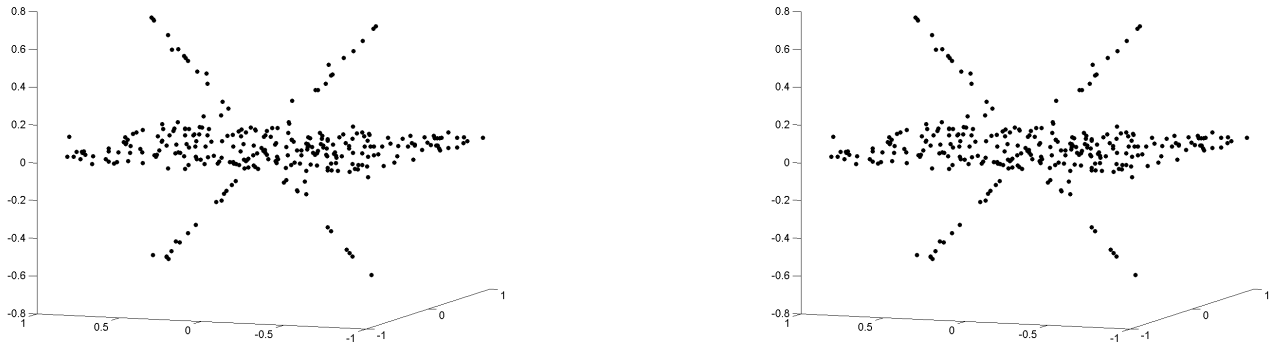

Figure 3: Problem II(b)(1) Original data; (2) SSC method

2.2.3 Problem II (c)

We see that the two parabolas are nonlinear, one of the two not passes the origin, and the two do not intersect. This implies two points: 1. This problem doesn't satisfy the subspace clustering assumption. 2. The two one dimensional manifold do not intersect, so Kmanifold doesn't apply. Considering these two insights, we first use KNN to construct an affinity matrix, then use spectrum clustering method.

I took K= 10. One thing to notice is that K cannot be too large, otherwise we violate the fact that the two lines do not intersect. Results are as follows.

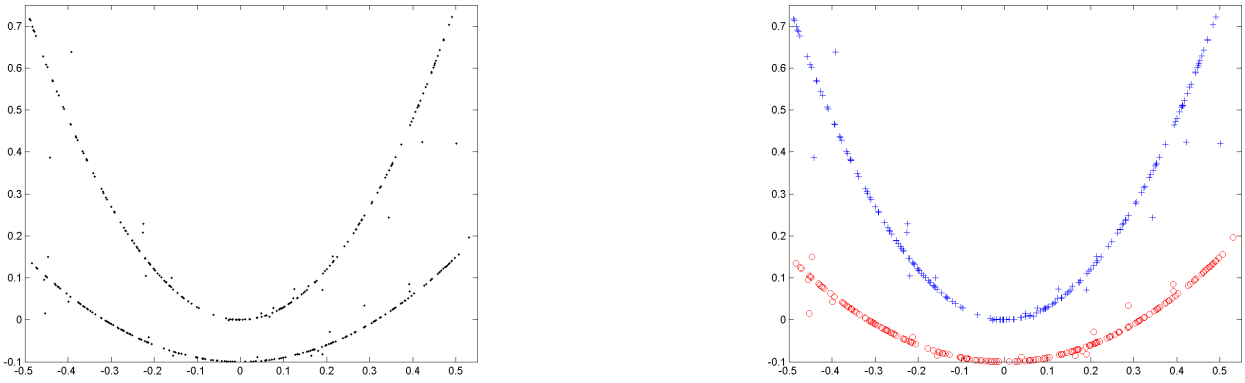

Figure 4: Problem II(c)(1) Original data; (2) LSC method

3 Conclusion

From the previous parts, we can see that kmanifold, with the realization of SSC and LSC, can be applied to 2D and 3D data separation and classification. With smaller data set and simple structure, methods like PCA and KNN are both sufficient for solving our problems. However, with

bigger data set and previously unknown data structure, kmanifold is an efficient way of dealing with these data sets. Taking into consideration of the curse of dimensionality, LSC is more robust in multi-dimensional data problems. (This interesting observation can be compared to QDA and LDA in their application with different data sets).

References

[1] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(2):218–233, 2003.

[2] R. Vidal. Subspace clustering. IEEE Signal Processing Magazine, 28(2):52–68, 2011.

[3] J. Shi and J. Malik, Normalized cuts and image segmentation. IEEE Transactions Pattern Analysis Machine Intelligence, 22(8):888–905, 2000.

[4] 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, 35(1):171–184, 2013.

[5] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(11):2765–2781, 2013.