Neurocomputing 139 (2014) 382–396

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Locally multidimensional scaling by creating neighborhoods in diffusion maps

Tomer Lancewicki *, Mayer Aladjem

Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer Sheva, Israel

ARTICLE INFO

Article history:
Received 16 January 2012
Received in revised form
18 February 2014
Accepted 24 February 2014
Communicated by S. Fiori
Available online 8 April 2014

Keywords:
Dimensionality reduction
Diffusion map
Multidimensional scaling

ABSTRACT

This paper analyzes and improves an advanced multidimensional scaling method, known as locally multidimensional scaling, which assumes that high-dimensional data lie on a low-dimensional manifold. The method preserves local distances in the manifold by using classical scaling on a set of clusters in the high-dimensional data. These clusters are called neighborhoods, and the success of the method depends on the proper selection of these neighborhoods. At present, a neighborhood set is difficult to tune, and even if done well, the method may not function properly in dealing with noisy data. Our proposal utilizes clustering in a diffusion map, and thereby improves the original method in two ways. First, neighborhood selection is easier to tune, and second, the neighborhoods chosen enable the improved method to work under noisy data conditions. Our experiments demonstrate better tuning and robustness-to-noise results compared with the original method and some other existing multidimensional scaling methods on synthetic and real data sets.

1. Introduction

Multidimensional scaling (MDS) methods [1] assume that a set of N objects is under consideration and that between each pair of objects (i,j) a *dissimilarity* measure d_{ij} exists. MDS searches for points in a low dimensional configuration in which the distances d'_{ij} between the points match in the best way possible the original dissimilarities d_{ij}. In this paper, the objects are represented by N points $X=\{x_i\}_{i=1}^N$ in a p-dimensional space $x_i \in \mathbb{R}^p$, and the goal of MDS is to find a q-dimensional set of points $Y=\{y_i\}_{i=1}^N$, $y_i \in \mathbb{R}^q$ for $q<p$, which optimizes a given criterion in order to preserve the original dissimilarities d_{ij} (i.e., distances d_{ij} between $x_i, x_j \in \mathbb{R}^p$) by means of d'_{ij}, which are the distances between points in a low dimensional space. In most applications, $q \ll p$ is used, and for visualization purposes $q=2$ or 3. Moreover, we assume that distances between pairs (y_i, y_j) are measured by a squared Euclidean norm of the vectors' difference (i.e., $y_i - y_j$), denoted as $d'^2_{ij} = \|y_i - y_j\|^2$, $i,j=1,\ldots,N$. This norm is suitable [2, page 16] for identifying clusters and other data structures by means of visual data inspection in low dimensional space. In *classical scaling*, originally formulated by Young and Mouseholder [3] and by Gower [4], the dissimilarities between the points $X=\{x_i\}_{i=1}^N$, $x_i \in \mathbb{R}^p$ are measured by $d^2_{ij} = \|x_i - x_j\|^2$. This makes it possible to find a linear mapping $y_i = Ax_i$ that preserves the distances in the sense of a minimum sum of squared errors $\sum_{ij}(d^2_{ij} - d'^2_{ij})^2$. Classical scaling is usually too restrictive in real-world applications [5, Sec. 1], which creates the need for nonlinear dimensionality reduction methods [6].

This paper studies and improves *locally multidimensional scaling* (LMDS) [5], based on the assumption that the original data points $X=\{x_i\}_{i=1}^N$, $x_i \in \mathbb{R}^p$ lie on a q-dimensional manifold. This assumption is utilized in nonlinear dimensionality reduction methods [6–14]. LMDS produces a q-dimensional set of points $Y=\{y_i\}_{i=1}^N$, $y_i \in \mathbb{R}^q$, which preserves local distances on the manifold to the maximum degree possible, using a sequential application of classical scaling. In the first stage of LMDS, a set $F=\{X_m\}_{m=1}^M$ comprising M overlapping clusters, which lie on the manifold and cover X, are created using an algorithm [5, Sec. 4]. In the terminology of LMDS, the clusters X_m are called *neighborhoods*. In the second stage of LMDS, each neighborhood is mapped by means of classical scaling $A_m : X_m \to Y'_m$, which has a projection matrix A_m, while Y'_m is known as the *patch* of X_m. In the third stage, the patches Y'_m are translated, scaled and rotated by linear transformations, which preserve to the maximum degree possible the position of the points in Y'_m, corresponding to the overlapping regions of neighborhoods X_m. Then, from the transformed Y'_m, the final low-dimensional data configuration $Y=\{y_i\}_{i=1}^N$, $y_i \in \mathbb{R}^q$ is created. The procedure for creating Y from patches Y'_m is called *alignment* and is explained in [5, Sec. 3].

As in the original LMDS paper, we present it on a synthetic *two-dimensional* (2D) manifold in *three-dimensional* (3D) space. This is created using a non-linear transformation (which we refer to as

* Corresponding author.
E-mail address: tomerlance@gmail.com (T. Lancewicki).

http://dx.doi.org/10.1016/j.neucom.2014.02.019
0925-2312/

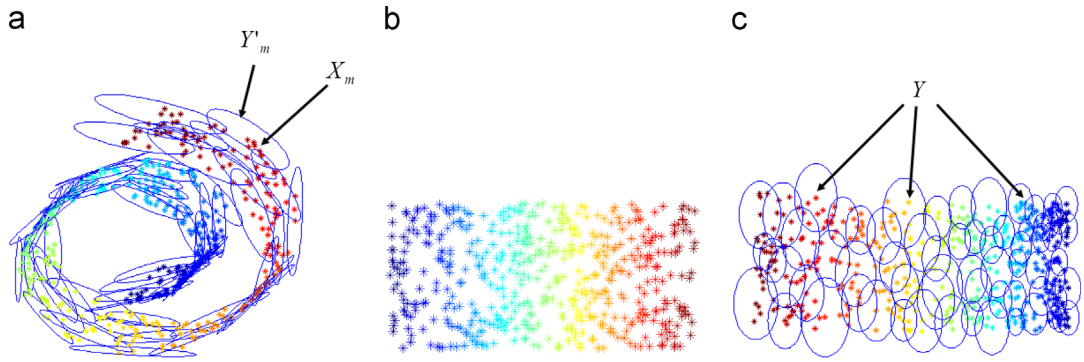

Fig. 1. LMDS used on swiss-roll points. (a) Schematic presentation of patches Y'_m, marked by ellipses, which are spanned on the row vectors of local classical scaling projection matrices A_m, (b) scattered points in the plane prior to rolling and (c) the final configuration $Y=\{y_i\}_{i=1}^N$, created by using the LMDS alignment procedure.

"rolling") of a plane into 3D space. In Fig. 1(a), we present uniformly distributed points on a 2D manifold. These points are generated using a procedure explained in [5, Sec. 5]. In this paper, we refer to them as *swiss-roll* points. Fig. 1(b) presents these points in the plane prior to rolling. A schematic illustration of the patches Y'_m obtained by LMDS is shown in Fig. 1(a). The points in patches Y'_m lie within planes, which are spanned on the row vectors of the local classical scaling projection matrices A_m. In Fig. 1(a), these planes are marked by ellipses. The final data configuration $Y=\{y_i\}_{i=1}^N$ obtained by the LMDS alignment procedure is shown in Fig. 1(c), which is similar to the embedded points in the plane prior to rolling (Fig. 1(b)). This demonstrates the efficiency of LMDS for discovering the manifold in the example described above.

A key step in the LMDS method is the proper creation of overlapping neighborhoods $F=\{X_m\}_{m=1}^M$. In this paper, we propose an algorithm for obtaining $F=\{X_m\}_{m=1}^M$, which improves LMDS in two ways: neighborhood selection is easier to tune, and the neighborhoods chosen enable the improved method to function under noisy data conditions.

The rest of this paper is organized as follows. Section 2 explains the need for improving LMDS. In Section 3, we propose an algorithm for creating neighborhoods $F=\{X_m\}_{m=1}^M$ by performing clustering in a *diffusion map* (DM) to obtain $F=\{X_m\}_{m=1}^M$, which improves LMDS under conditions of noisy data. Section 4 presents experiments on synthetic and real data sets. The results confirm the improvement obtained by utilizing our LMDS relative to the original LMDS [5] as well as some other dimensionality reduction methods, namely, ISOMAP [7], MVU [9], LTSA [11] and CB-LMDS [13].

2. The motivation for improving LMDS

In this section, we first describe the functioning of the original algorithm [5, Sec. 4] currently in use for obtaining overlapping neighborhoods $F=\{X_m\}_{m=1}^M$. We then highlight some of the algorithm's drawbacks , which provide motivation for the need to improve it.

2.1. The algorithm in the original LMDS method used for creating neighborhoods $F=\{X_m\}_{m=1}^M$

The original LMDS algorithm [5, Sec. 4] first produces N neighborhoods $\tilde{F}=\{\tilde{X}_i\}_{i=1}^N$ for each point in the original data set $X=\{x_i\}_{i=1}^N$, $x_i\in\mathbb{R}^p$, where $\tilde{X}_i$ comprises K points[1] with the smallest Euclidean distances to x_i (K *is a user supplied parameter of LMDS*). It then creates a set of overlapping clusters $F=\{X_m\}_{m=1}^M$ for $M<N$ by iteratively choosing elements from $\tilde{F}=\{\tilde{X}_i\}_{i=1}^N$. In the initial iteration, the first neighborhood, denoted as X_1, is randomly selected. Then, other neighborhoods X_m, $m=2,3,\ldots,M$, are sequentially selected from $\tilde{F}$. The algorithm maintains the overlap of the current X_m and previous F at[2] $\alpha\times 100\%$ $(0<\alpha<1)$ (where α *is another user supplied parameter of LMDS*) and terminates when F covers all points in X. In Fig. 2, three iterations of the algorithm are presented for points in 2D space using the setting $K=12$ and $\alpha=0.25$. This means that $\tilde{X}_i$ has twelve points and an overlap of three points $(12\times 0.25=3)$ between the current X_m and the previous F. Fig. 2(a) presents the first neighborhood X_1, while Fig. 2(b) presents the second X_2, which overlaps X_1 at $\alpha\times 100\%$ (i.e., in the figure with three common points). There may be several X_2 candidates with the same degree of overlap. In this case, the algorithm selects one of them randomly. Fig. 2(c) shows the third iteration, in which X_3 is added. As previously, X_3 is selected randomly from several X_3 candidates.

It should be noted that the tuning parameters α and K must be chosen carefully. Trials with various α and K can be conducted, followed by a visual examination of the final low-dimensional data configuration Y for $q=2$ or 3. When $q>3$, other research should be conducted, such as an assessment of the accuracy of clustering and/or evaluation of classification algorithms. When there are large α values (i.e., high overlapping between current X_m and previous F), the algorithm may terminate before covering all data points in X. For small α (i.e., those barely overlapping between current X_m and F), the LMDS alignment procedure may prove incapable of functioning, because of the need to resolve an ill-posed optimization problem (explained in [5, page 443, paragraph 3]). The parameter K, which is the number of points in $\tilde{X}_i$, has a considerable impact on the final low-dimensional configuration Y. For K that are too large or small, the local geometry of the manifold can be lost.

2.2. Drawbacks of the original LMDS algorithm for creating neighborhoods $F=\{X_m\}_{m=1}^M$

In this section, we explain two LMDS drawbacks. First, in order to find a suitable low-dimensional configuration Y, the user must conduct trials on a large grid of K and α values, in addition to different first neighborhoods X_1. Second, LMDS performs poorly under conditions of noisy data.

[1] In the general description of the current LMDS, neighborhoods $\tilde{X}_i$ are allowed to be of different sizes. In this paper, an identical number K of points in $\tilde{X}_i$ is used in order to remain consistent with the original paper's experiments [5, Sec. 5].

[2] For the general case, entailing neighborhoods $\tilde{X}_i$ of different sizes, the algorithm [5, Sec. 4] chooses a candidate neighborhood that overlaps the previous F at a minimum degree of $\alpha\times 100\%$ and contains the maximum number of uncovered data points.