Lecture 10, 11 Oct. 11, 13 -2006

Data Visualization

STAT 442 / 890, CM 462

Lecture: Ali Ghodsi

1 Metric Multidimensional Scaling (MDS)

An alternative perspective on dimensionality reduction is offered by Multidimensional scaling (MDS). MDS is another classical approach that maps the original high dimensional space to a lower dimensional space, but does so in an attempt to preserve pairwise distances. That is MDS addresses the problem of constructing a configuration of t points in Euclidean space by using information about the distances between the t patterns.

A $t \times t$ matrix D is called a distance or affinity matrix if it is symmetric, $\mathtt{d}_{ii} = 0$, and $\mathtt{d}_{ij} > 0, \quad i \neq j$.

Given a distance matrix $D^{(X)}$, MDS attempts to find t data points $y_1, ..., y_t$ in d dimensions, such that if $d_{ij}^{(Y)}$ denotes the Euclidean distance between y_i and y_j, then D^Y is similar to $D^{(X)}$. In particular, we consider metric MDS [1], which minimizes

$$\min_Y \sum_{i=1}^{t} \sum_{i=1}^{t} (\mathtt{d}_{ij}^{(X)} - \mathtt{d}_{ij}^{(Y)})^2 \tag{1}$$

where $\mathtt{d}_{ij}^{(X)} = ||x_i - x_j||$ and $\mathtt{d}_{ij}^{(Y)} = ||y_i - y_j||$.

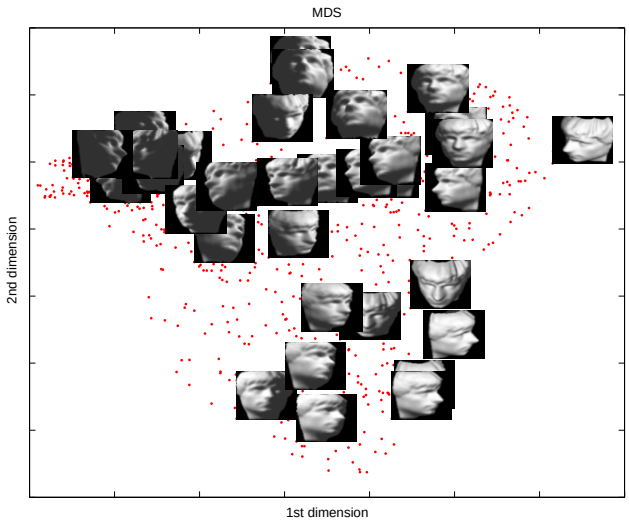

Figure 1: *MDS applied to the same data set. A two-dimensional projection is shown, with a sample of the original input images.*

The distance matrix $D^{(X)}$. can be converted to a kernel matrix K by

$$K = -\frac{1}{2} H D^{(X)} H \tag{2}$$

where $H = I - \frac{1}{t} e e^T$ and e is a column vector of all 1.

Theorem: Let D be a distance matrix and define K by (2). Then D is Euclidean if and only if K is positive semi-definite. (For detail see [1], page 397)

Since K is *p.s.d.*, it can be written as $K = X^T X$. Now (1) can be reduced to

$$\min_Y \sum_{i=1}^{t} \sum_{i=1}^{t} (x_i^T x_j - y_i^T y_i)^2$$

The norm can be converted into a trace.

$$\min_Y Tr(X^T X - Y^T Y)^2$$

By singular value decomposition X^TX and Y^TY can be decomposed as:

$$X^TX = V\Lambda V^T$$

$$Y^TY = Q\hat{\Lambda}Q^T$$

[1] Since Y^TY is *p.s.d.*, $\hat{\Lambda}$ has no negative value and therefore:

$$Y = \hat{\Lambda}^{\frac{1}{2}}Q^T. \tag{3}$$

The above definitions help rewrite the cost function as:

$$\min_{Q,\hat{\Lambda}} Tr(V\Lambda V^T - Q\hat{\Lambda}Q^T)^2$$

$$= \min_{Q,\hat{\Lambda}} Tr(\Lambda - V^TQ\hat{\Lambda}Q^TV)^2$$

Let

$$G = V^TQ \tag{4}$$

$$\min_{G,\hat{\Lambda}} Tr(\Lambda - G\hat{\Lambda}G^T)^2$$

Expand out the square.

$$\min_{G,\hat{\Lambda}} Tr(\Lambda^2 + G\hat{\Lambda}G^TG\hat{\Lambda}G^T - 2\Lambda G\hat{\Lambda}G^T)$$

For a fixed $\hat{\Lambda}$ we can minimize for G and the result is that

$$G = I \tag{5}$$

[1]It is known that an arbitrary matrix X can be decomposed as $X = U\Lambda V^T$ by SVD. Note that the matrix X^TX is symmetric and so we can conclude that in its SVD, $U = V$

$$\min_{\hat{\Lambda}} Tr(\Lambda^2 + \hat{\Lambda}^2 - 2\Lambda\hat{\Lambda})$$

$$= \min_{\hat{\Lambda}} Tr(\Lambda - \hat{\Lambda})^2$$

To make the two matrices Λ and $\hat{\Lambda}$ as similar as possible we can make $\hat{\Lambda}$ be the top d diagonal elements of Λ. Also 4 and 5 imply that $Q = V$. Therefore 3 can be written as:

$$Y = \hat{\Lambda}^{1/2} V^T \tag{6}$$

where V is the eigenvectors of $X^T X$ corresponding to the top d eigenvalues, and $\hat{\Lambda}$ is the top d eigenvalues of $X^T X$. Clearly the solution for MDS is identical to dual PCA. As far as Euclidean distance is concerned, MDS and PCA produce the same results. However, the distances in MDS need not be based on Euclidean distances and can represent many types of dissimilarities between objects.

2 Isomap

Similar to PCA, MDS has been recently extended to perform nonlinear dimensionality reduction. A recent approach to nonlinear dimensionality reduction based on MDS is the Isomap algorithm.

Isomap is a nonlinear generalization of classical MDS. The main idea is to perform MDS, not in the input space, but in the geodesic space of the nonlinear data manifold. The geodesic distances represent the shortest paths along the curved surface of the manifold measured as if the surface were flat. This can be approximated by a sequence of short steps between neighboring sample points. Isomap then applies MDS to the geodesic rather than straight

line distances to find a low-dimensional mapping that preserves these pairwise distances.

Like LLE, the Isomap algorithm proceeds in three steps:

1. Find the neighbors of each data point in high-dimensional data space.

2. Compute the geodesic pairwise distances between all points.

3. Embed the data via MDS so as to preserve these distances.

Again like LLE, the first step can be performed by identifying the k nearest neighbors, or by choosing all points within some fixed radius, ϵ. These neighborhood relations are represented by a graph G in which each data point is connected to its nearest neighbors, with edges of weight $d_X(i, j)$ between neighbors.

The geodesic distances $d_M(i, j)$ between all pairs of points on the manifold M are then estimated in the second step. Isomap approximates $d_M(i, j)$ as the shortest path distance $d_G(i, j)$ in the graph G. This can be done in different ways including Dijkstra's algorithm and Floyd's algorithm.

These algorithms find matrix of graph distances $D^{(\mathcal{G})}$ contains the shortest path distance between all pairs of points in G. In its final step, Isomap applies classical MDS to $D^{(\mathcal{G})}$ to generate an embedding of the data in a d-dimensional Euclidean space Y. The global minimum of the cost function is obtained by setting the coordinates of y_i to the top d eigenvectors of the inner-product matrix B obtained from $D^{(\mathcal{G})}$

References

[1] T. Cox and M. Cox. *Multidimensional Scaling.* Chapman Hall, Boca Raton, 2nd edition, 2001.