EE 381V: Large Scale Learning **Spring 2013**

Lecture 12 — February 21

Lecturer: Caramanis & Sanghavi *Scribe: Samer Chucri*

12.1 Last Few Lectures

In the last few lectures, the following has been covered:

1. Linear Dimensionality Reduction (via PCA)
2. Dimensionality Reduction for clustering (Spectral Clustering)
3. Dimensionality reduction via Random Embedding (JohnsonLindenstrauss)

In this lecture we are going to explore dimensionality reduction from local geometry, a form of non-linear dimensionality reduction.

12.2 Non-linear dimensionality Reduction

12.2.1 High level description

Given a non-linear manifold in High Dimensional space, we want to:

1. Preserve distance between points
2. Preserve manifold distance: Whether 2 points are far or not should be determined by the manifold as shown in Figure 12.1.

Our goal is to map the points in high dimensional space to a lower dimensional space while preserving (1) & (2).

12.2.2 Problem statement

We are given the following:

1. m Points in ambient space $x_i \in \mathbb{R}^N$
2. Only specific d_{ij} are known for some pairs, where $d_{ij} = ||x_i - x_j||_2$

Note that another scenario equivalent to (2.) is where only near neighbor distances are reliable, as shown in the manifold example. In this case, we have some radius of confidence around each point, and points that fall in the sphere are considered near neighbors.

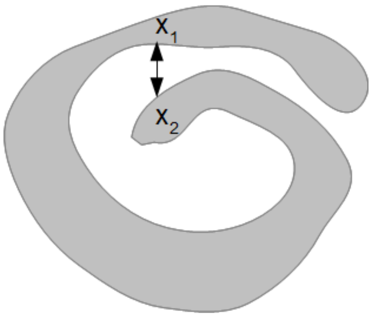

Figure 12.1. Manifold in high dimension where x_1 and x_2 should not be considered close.

Task:

Find $\hat{x}_i \in \mathbb{R}^d$, such that near neighbor distances are preserved

12.2.3 Technique 1

Multidimensional Scaling (MDS)- MAP

The main idea of MDS starts by forming the square distance matrix $D_{m\times m}$:
$(D)_{ij} = d_{ij}^2$
Remember, we are only given some of the entries since d_{ij} is not known for some pairs, or only a selected number of d_{ij}'s are reliable.

Theorem 12.1. *Suppose $x_i \in \mathbb{R}^d$ and $d_{ij}^2 = ||x_i - x_j||^2$ is known for all pairs (i, j), then D has rank at most $d + 2$*

Proof: Start by forming matrix $X_{m\times d}$ as follows:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \tag{12.1}$$

It's easy to see that D can be written as follows:

$$D = -2XX^T + a1^T + 1a^T \tag{12.2}$$

where $a_i = ||x_i||^2$
This is so since $d_{ij}^2 = ||x_i||^2 + ||x_j||^2 - 2x_i^T x_j^T$

Now we know that XX^T has rank at most d, $a1^T$ has rank 1, and $1a^T$ has rank 1.
It follows that D has rank at most $d + 2$ □

This tells us that the full matrix D is indeed low rank, so we need to find a way to fill the missing entries in D while also preserving this property.

One suggestion Fill the missing values in D with 0, and perform an SVD.
It turns out that this method is a bad idea, because typically we are only given the small entries, and SVD tries to to approximate small entries as close as possible and assign 0 for the others.

Second Approach

1. Form a graph G by representing each point x_i as a vertex i, and edge (i, j) exists if we are given d_{ij} as shown in Figure 12.2
2. Form matrix $\hat{D}$ where $\hat{D}_{ij} = D_{ij}$ if $(i, j) \in \Omega$, where Ω is the observed set. For the remaining entries $\hat{D}_{ij} =$ Graphical shortest path, if one exists.

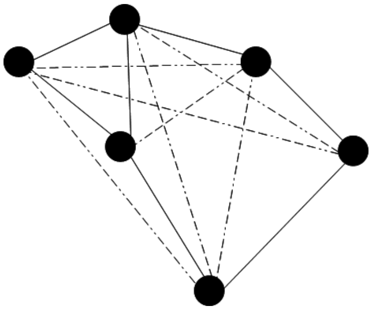

Figure 12.2. Continuous line represents a given value d_{ij} whereas a dotted line represents a missing d_{ij}

Note that we need our graph G to be connected, otherwise the best we can do is embed each connected component independently.

After forming $\hat{D}$, we need to ensure that $rank(\hat{D})$ is consistent with the low rank requirement that we set initially. Unfortunately in many cases, $\hat{D}$ will not be low rank because the missing distances are approximate (based on the triangle inequality).
We will present a solution to this problem, but first let's state a fact about rigid motion:

Fact 1. *X and $\hat{X}$ are connected by Rigid Motion $\Leftrightarrow P^{\perp}(XX^T - \hat{X}\hat{X}^T)P^{\perp} = 0$
where $P^{\perp} = I_{m\times m} - \frac{1}{m}11^T$*

Note: $P^{\perp}$ simply removes translation by centering the points

$$\begin{aligned} P^{\perp}X &= X - \frac{1}{m}\sum_{i=1}^{m} x_i && (12.3)\\ P^{\perp}XX^TP^{\perp} &= X_{cent}X_{cent}^T && (12.4)\end{aligned}$$

Rotation is taken care of since our algorithm depends on XX^T which takes care of any multiplication by a unitary matrix

We will restate the problem:
Given some d_{ij}, our task is to find a corresponding point $\hat{x}_i \in \mathbb{R}^d$ for every $x_i \in \mathbb{R}^N$, such that these distances are preserved.

Complete Algorithm

1. Compute $\hat{D}$ via shortest path
2. Compute $Q = -\frac{1}{2}P^{\perp}\hat{D}P^{\perp}$
3. Decompose $Q = U\Lambda U^T$
4. $\hat{X} = U_d\Lambda_d^{\frac{1}{2}}$, where U_d corresponds to top d eigenvectors

Note that $P^{\perp}DP^{\perp} = -2X_{cent}X_{cent}^T$

12.3 Analysis of Algorithm

Setup

We have m points in $[-1,1]^d$ chosen uniformly at random.

12.3.1 Model 1

In this model we assume that d_{ij} is known $\Leftrightarrow d_{ij} \leq r$.
This is known as a geometric random graph in d dimensions where two vertices i and j are connected if and only if $d_{ij} \leq r$

Fact 2. *Let* $r_0 = C_1\left(\frac{\log m}{m}\right)^{\frac{1}{d}}$
If $r > r_0$ *then the geometric random graph is connected with probability* $1 - m^{-\alpha}$, *where* α *depends on* C_1.

Lemma 1. *For* $r > r_0$, *the following sequence of inequalities holds with high probability:*

$$d_{ij}^2 \leq \hat{d}_{ij}^2 \leq d_{ij}^2(1 + \frac{c_2 r_0}{r}) \tag{12.5}$$

Remarks

1. Recall that r is our threshold of trust, and by increasing the number of samples m we end up needing a smaller r. This is so since increasing m reduces r_0, which allows us to reduce r.
2. The higher r is the better our bound is. This is intuitive since increasing r means we have more knowledge of the underlying geometry.

12.3.2 Model 2

In this model we assume that $d'_{ij} = \begin{cases} 1 & \text{if } d_{ij} \le r \\ 0 & o.w. \end{cases}$

In this case $\hat{D}$ is created by performing shortest path search on rD', where $D'_{ij} = d'_{ij}$. Lemma 1 does not apply anymore, but rather a variant of it.

Lemma 2. *For $r > r_0$, the following sequence of inequalities holds for all (i, j) with high probability:*

$$d_{ij}^2 \le \hat{d}_{ij}^2 \le d_{ij}^2(c_2 r + \frac{c_3 r_0}{r}) \tag{12.6}$$

Remark

Note that in this case increasing r is not always advantageous. By increasing r too much we lose information of closeness of points, and at the limit we lose all information about the graph.

12.4 Locally Linear Embedding (LLE)

Setup

Given a set of points $\{x_1, ..., x_m\}$ in ambient space (not only distances) and a radius of trust r, our task is to find a lower dimensional embedding $\{\hat{y}_1, ..., \hat{y}_m\}$ that preserves the local geometry (local is defined by r)

Idea

1. Represent each point as a linear combination of its near neighbors

2. Take the coefficients of the linear combination and use that to find low dimensional points

12.4.1 Algorithm

1. $\hat{W} = \underset{W}{\arg\min} \sum_{i=1}^m ||x_i - \sum_{j \in N_i} W_{ij} x_j||^2$

2. $\hat{Y} = \underset{Y s.t. \sum y_i = 0, \frac{1}{m} \sum y_i y_i = I}{\arg\min} \sum_{i=1}^m ||y_i - \sum_{j \in N_i} \hat{w}_{ij} y_j||^2$, where y_i is the ith row of $Y_{m \times d}$

In step 1, $\hat{W}$ tells us what the local geometry is, and in step 2, we try to find low dimensional points that are consistent with that local geometry.

Remarks

Finding $\hat{Y}$ reduces to the following problem:

$\hat{Y} = \underset{Y s.t.\ YY^T=I}{\arg\min}\ Y^T M T$, where $M = (I - \hat{W})(I - \hat{W})^T$
Notice that the solution to this optimization problem $\hat{Y}$, is nothing but the matrix formed by the eigenvectors corresponding to the lowest d eigenvalues of M.