Lecture 8: Geometric Graph Models. Factorization and SDP Approach

Radu Balan

Department of Mathematics, AMSC, CSCAMM and NWC
University of Maryland, College Park, MD

April 3, 2018

Euclidean Embeddings of Weighted Graphs
High-Level Introduction

The embedding problem is the following:

Main Problem

Given a weighted graph $G = (\mathcal{V}, W)$ *with* n *nodes, find a dimension* d *and a set of* n *points* $\{y_1, \cdots, y_n\} \subset \mathbb{R}^d$ *such that* $W_{i,j} = \varphi(\|y_i - y_j\|)$ *for some monotonically decreasing function* φ.

Euclidean Embeddings of Weighted Graphs

High-Level Introduction

The embedding problem is the following:

Main Problem

Given a weighted graph $G = (\mathcal{V}, W)$ with n nodes, find a dimension d and a set of n points $\{y_1, \cdots, y_n\} \subset \mathbb{R}^d$ such that $W_{i,j} = \varphi(\|y_i - y_j\|)$ for some monotonically decreasing function φ.

Approaches:

1. **Nearly Isometric Embeddings**. Three steps: (1) convert weights into geometric distances; (2) solve a SDP optimization problem; (3) perform a factorization (PCA) of the solution.
2. **Laplacian Eigenmaps**. Three steps: (1) Construct the symmetric normalized weighted Laplacian matrix; (2) Solve for a set of eigenvectors; (3) Perform embedding.

Distance Models for Weighted Graphs

Let $W = (W_{i,j})_{1 \leq i,j \leq n}$ be a known weight matrix.
Most frequently employed models:

1. Exponential Law: $W_{i,j} = e^{-d_{i,j}^2/\sigma^2}$.
2. Power Law: $W_{i,j} = \frac{C}{d_{i,j}^p}$.

where $d_{i,j}$ denotes the distance between points i and j.
σ, C, p are parameters.

Distance Models for Weighted Graphs

Let $W=(W_{i,j})_{1\leq i,j\leq n}$ be a known weight matrix.
Most frequently employed models:

1. Exponential Law: $W_{i,j}=e^{-d_{i,j}^2/\sigma^2}$.
2. Power Law: $W_{i,j}=\frac{C}{d_{i,j}^p}$.

where $d_{i,j}$ denotes the distance between points i and j.
σ, C, p are parameters.
Without loss of generality one can scale the coordinates (points) and absorb global constants. Hence we can assume $\sigma=1$, $C=1$.
Therefore one can compute the estimated squared-distances by:

$$d_{i,j}^2=\begin{cases}-\log(W_{i,j}) & \text{exponential law}\\ \left(\frac{1}{W_{i,j}}\right)^{2/p} & \text{power law}\end{cases}$$

Isometric Embeddings with Full Data

Problem statement and Ambiguities

Solve the geometric problem:

Main Problem

Given the set of all squared-distances $\{d_{i,j}^2;\ 1 \leq i,j \leq n\}$ *find a dimension* d *and a set of* n *points* $\{y_1, \cdots, y_n\} \subset \mathbb{R}^d$ *so that* $\|y_i - y_j\|^2 = d_{i,j}^2$, $1 \leq i,j \leq n$.

Note the set of points is unique up to rigid transformations: translations, rotations and reflections: $\mathbb{R}^d \times O(d)$. This means two sets of n points in $\mathbb{R}^d$ have the same pairwise distances if and only if one set is obtained from the other set by a combination of rigid transformations.

Isometric Embeddings with Full Data

Converting pairwise distances into the Gram matrix

Let $S = (S_{i,j})_{1\leq i,j\leq n}$ denote the $n \times n$ symmetric matrix of squared pairwise distances:

$$S_{i,j} = d_{i,j}^2 \quad , S_{i,i} = 0$$

Denote by 1 the n-vector of 1's (the Matlab $ones(n, 1)$). Let $\nu = (\|y_i\|^2)_{1\leq i\leq n}$ denote the unknown n-vector of squared-norms. Finally, let $G = (\langle y_i, y_j\rangle)_{1\leq i,j\leq n}$ denote the Gram matrix of scalar products between

We can remove the translation ambiguity by fixing the center:

$$\sum_{i=1}^{n} y_i = 0$$

Isometric Embeddings with Full Data

Converting pairwise distances into the Gram matrix

Expand the square:

$$d_{i,j}^2 = \|y_i - y_j\|^2 = \|y_i\|^2 + \|y_j\|^2 - 2\langle y_i, y_j\rangle$$

Rewrite the system as:

$$2G = \nu \cdot 1^T + 1 \cdot \nu^T - S \quad (*)$$

The center condition reads: $G \cdot 1 = 0$, which implies:

$$0 = 2n\nu^T \cdot 1 - 1^T \cdot S \cdot 1$$

Let $\rho := \nu^T \cdot 1 = \sum_{i=1}^n \|y_i\|^2$. We obtain:

$$\rho = \frac{1}{2n} 1^T \cdot S \cdot 1 = \frac{1}{2n} \sum_{i=1}^n \sum_{j=1}^n d_{i,j}^2$$

$$\nu = \frac{1}{n}(S \cdot 1 - \rho 1) = \frac{1}{n}(S - \rho I) \cdot 1$$

that you substitute back into (*).

Isometric Embeddings with Full Data

Converting pairwise squared-distances into the Gram matrix: Algorithm

Algorithm

Input: Symmetric matrix of squared pairwise distances $S = (d_{i,j}^2)_{1 \leq i,j \leq n}$.

1. *Compute:*
$$\rho = \frac{1}{2n} 1^T \cdot S \cdot 1 = \frac{1}{2n} \sum_{i=1}^n \sum_{j=1}^n d_{i,j}^2$$
2. *Set:*
$$\nu = \frac{1}{n}(S \cdot 1 - \rho 1) = \frac{1}{n}(S - \rho I) \cdot 1$$
3. *Compute:*
$$G = \frac{1}{2}\nu \cdot 1^T + \frac{1}{2} 1 \cdot \nu^T - \frac{1}{2}S = \frac{1}{2n}(S - \rho I) 1 \cdot 1^T + \frac{1}{2n} 1 \cdot 1^T (S - \rho I) - \frac{1}{2}S.$$

Output: Symmetric Gram matrix G

Isometric Embeddings with Full Data

Factorization of the G matrix

In the absence of noise (i.e. if $S_{i,j}$ are indeed the Euclidean distances), the Gram matrix G should have rank d, the minimum dimension of the isometric embedding.
If S is noisy, then G has approximate rank d.
To find d and Y, the matrix of coordinates, perform the eigendecomposition:

$$G = Q\Lambda Q^T$$

where Λ is the diagonal matrix of eigenvalues, ordered monotonically decreasing. Choose d as the number of significant positive eigenvalues (i.e. truncate to zero the negative eigenvalues, as well as the smallest positive eigenvalues). Note G has always at least one zero eigenvalue: $rank(G) \leq n - 1$.

Isometric Embeddings with Full Data

Factorization of the G matrix

Then we obtain an approximate factorization of G (exact in the absence of noise):

$$G \approx Q_1 \Lambda_1 Q_1^T$$

where Q_1 is the $n \times d$ submatrix of Q containing the first d columns.
Set $Y = \Lambda_1^{1/2} Q_1^T$, so that $G \approx Y^T Y$.
The $d \times n$ matrix Y contains the embedding vectors $y_1, \cdots, y_n$ as columns:

$$Y = [y_1 | y_2 | \cdots | y_n] .$$

Isometric Embeddings with Full Data

Gram matrix factorization: Algorithm

Algorithm

Input: Symmetric $n \times n$ Gram matrix G.

1. *Compute the eigendecomposition of G, $G = Q\Lambda Q^T$ with diagonal of Λ sorted in a descending order;*
2. *Determine the number d of significant positive eigevalues;*
3. *Partition*

$$Q = [Q_1 \quad Q_2] \quad \text{, and} \quad \Lambda = \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{bmatrix}$$

   where Q_1 contains the first d columns of Q, and Λ_1 is the $d \times d$ diagonal matrix of significant positive eigenvalues of G.

4. *Compute:*

$$Y = \Lambda_1^{1/2} Q_1^T$$

Output: Dimension d and $d \times n$ matrix Y of vectors $Y = [y_1 | \cdots | y_n]$

Isometric Embeddings with Partial Data

Dimension estimation

Consider now the case that only a subset of the pairwise squared-distances are known, indexed by Θ. Assume that only m distances (out of $n(n-1)/2$ possible values) are known – this means the cardinal of Θ is m.

Remark

Minimum number of measurements: $m \geq nd - \frac{d(d+1)}{2}$, *because:* nd *is the number of degrees of freedom (coordinates) needed to describe* n *points in* $\mathbb{R}^d$; $d(d+1)/2$ *is the the dimension of the Lie group of Euclidean transformations: translations in* $\mathbb{R}^d$ *of dimension* d *and orthogonal transformations* $O(d)$ *of dimension* $d(d-1)/2$ *(the dimension of the Lie algebra of anti-symmetric matrices).*

In the absence of noise, for sufficiently large m but less than $n(n-1)/2$, exact (i.e. isometric) embedding is possible.

Isometric Embeddings with Partial Data
Linear constraints

Given any set of vectors $\{y_1, \cdots, y_n\}$ and their associated matrix $Y = [y_1| \cdots |y_n]$ their invariant to the action of the rigid transformations (translations, rotations, and reflections) is the Gram matrix of the centered system:

$$G = (I - \frac{1}{n}1 \cdot 1^T) Y^T Y (I - \frac{1}{n}1 \cdot 1^T) =: LY^T YL \quad , \quad L = I - \frac{1}{n}1 \cdot 1^T.$$

On the other hand, the distance between points i and j can be computed by:

$$d_{i,j}^2 = \|y_i - y_j\|^2 = G_{i,i} - G_{i,j} + G_{j,j} - G_{j,i} = e_{ij}^T G e_{ij}$$

where

$$e_{ij} = \delta_i - \delta_j = [0 \cdots 0\, 1 \cdots -1\, 0 \cdots 0]^T$$

where 1 is on position i, -1 is on position j, and 0 everywhere else.

Almost Isometric Embeddings with Partial Data

The SDP Problem

Reference [10] proposes to find the matrix G by solving the following Semi-Definite Program:

$$\min_{\substack{G = G^T \geq 0 \\ G1 = 0 \\ |\langle G e_{ij}, e_{ij}\rangle - \tilde{d}_{i,j}^2| \leq \varepsilon \ , \ (i,j) \in \Theta}} trace(G)$$

where $\tilde{d}_{i,j}^2$ are noisy estimates $d_{i,j}$ and ε is the maximum noise level. The trace promotes low rank in this optimization. However, this is basically a feasibility problem: Decrease ε to the minimum value where a feasible solution exists. With probability 1 that is unique.

How to do this: Use CVX with Matlab.

Nearly Isometric Embeddings with Partial Data

Stability to Noise

[10] proves the following stability result in the case of partial measurements. Here we denote $\Theta_r = \{(i,j)\ ,\ \|y_i - y_j\| \leq r\}$ the set of all pairs of points at distance at most r.

Theorem

Let $\{y_1, \cdots, y_n\}$ be n nodes distributed uniformly at random in the hypercube $[-0.5, 0.5]^d$. Further, assume that we are given noisy measurement of all distances in Θ_r for some $r \geq 10\sqrt{d}(\log(n)/n)^{1/d}$ and the induced geometric graph of edges is connected. Let $\tilde{d}^2_{i,j} = d^2_{i,j} + \nu_{i,j}$ with $|\nu_{i,j}| \leq \varepsilon$. Then with high probability, the error distance between the estimated $\hat{Y} = [\hat{y}_1, | \cdots | \hat{y}_n]$ returned by the SDP-based algorithm and the correct coordinate matrix $Y = [y_1 | \cdots | y_n]$ is upper bounded as

$$\|L\hat{Y}^T\hat{Y}L - LY^TYL\|_1 \leq C_1(nr^d)^5\frac{\varepsilon}{r^4}.$$

Convex Sets. Convex Functions

A set $S \subset \mathbb{R}^n$ is called a *convex set* if for any points $x, y \in S$ the line segment $[x, y] := \{tx + (1-t)y\ ,\ 0 \leq t \leq 1\}$ is included in S, $[x, y] \subset S$.

A function $f : S \to \mathbb{R}$ is called *convex* if for any $x, y \in S$ and $0 \leq t \leq 1$,
$f(t\,x + (1-t)y) \leq t\,f(x) + (1-t)f(y)$.
Here S is supposed to be a convex set in $\mathbb{R}^n$.
Equivalently, f is convex if its epigraph is a convex set in $\mathbb{R}^{n+1}$. Epigraph: $\{(x, u)\ ;\ x \in S, u \geq f(x)\}$.

A function $f : S \to \mathbb{R}$ is called *strictly convex* if for any $x \neq y \in S$ and $0 < t < 1$, $f(t\,x + (1-t)y) < t\,f(x) + (1-t)f(y)$.

Convex Optimization Problems

The general form of a convex optimization problem:

$$\min_{x \in S} f(x)$$

where S is a closed convex set, and f is a convex function on S.
Properties:

1. Any local minimum is a global minimum. The set of minimizers is a convex subset of S.
2. If f is strictly convex, then the minimizer is unique: there is only one local minimizer.

In general S is defined by equality and inequality constraints:
$S = \{g_i(x) \leq 0 \ , \ 1 \leq i \leq p\} \cap \{h_j(x) = 0 \ , \ 1 \leq j \leq m\}$. Typically h_j are required to be affine: $h_j(x) = a^T x + b$.

Convex Programs

The hiarchy of convex optimization problems:

1. Linear Programs: Linear criterion with linear constraints
2. Quadratic Programs: Quadratic Criterion with Linear Constraints; Quadratically Constrained Quadratic Problems (QCQP); Second-Order Cone Program (SOCP)
3. Semi-Definite Programs(SDP)

Typical SDP:

$$\min_{X = X^T \geq 0} \quad trace(XA)$$
$$trace(XB_k) = y_k \ , \ 1 \leq k \leq p$$
$$trace(XC_j) \leq z_j \ , \ 1 \leq j \leq m$$

CVX

Matlab package

Downloadable from: http://cvxr.com/cvx/ . Follows "Disciplined" Convex Programming – à la Boyd [2].

```
m = 20; n = 10; p = 4;
A = randn(m,n); b = randn(m,1);
C = randn(p,n); d = randn(p,1); e = rand;
cvx_begin
    variable x(n)
    minimize( norm( A * x - b, 2 ) )
    subject to
        C * x == d
        norm( x, Inf ) <= e
cvx_end
```

$$\min \|Ax - b\|$$
$$Cx = d$$
$$\|x\|_\infty \le e$$

CVX

SDP Example

```
cvx_begin sdp

    variable X(n,n) semidefinite;
    minimize trace(X);
    subject to
    X*ones(n,1) == zeros(n,1);
    abs(trace(E1*X)-d1)<=epsx;
    abs(trace(E2*X)-d2)<=epsx;

cvx_end
```

$$\begin{array}{ll} \text{minimize} & trace(X) \\ \text{subject to} & X = X^T \geq 0 \\ & X \cdot 1^T = 0 \\ & |trace(E_1 X) - d_1| \leq \varepsilon \\ & |trace(E_2 X) - d_2| \leq \varepsilon \end{array}$$

References

- B. Bollobás, **Graph Theory. An Introductory Course**, Springer-Verlag 1979. **99**(25), 15879–15882 (2002).
- S. Boyd, L. Vandenberghe, **Convex Optimization**, available online at: http://stanford.edu/ boyd/cvxbook/
- F. Chung, **Spectral Graph Theory**, AMS 1997.
- F. Chung, L. Lu, The average distances in random graphs with given expected degrees, Proc. Nat.Acad.Sci. 2002.
- F. Chung, L. Lu, V. Vu, The spectra of random graphs with Given Expected Degrees, Internet Math. **1**(3), 257–275 (2004).
- R. Diestel, **Graph Theory**, 3rd Edition, Springer-Verlag 2005.
- P. Erdös, A. Rényi, On The Evolution of Random Graphs

- G. Grimmett, **Probability on Graphs. Random Processes on Graphs and Lattices**, Cambridge Press 2010.
- C. Hoffman, M. Kahle, E. Paquette, Spectral Gap of Random Graphs and Applications to Random Topology, arXiv: 1201.0425 [math.CO] 17 Sept. 2014.
- [10]A. Javanmard, A. Montanari, Localization from Incomplete Noisy Distance Measurements, arXiv:1103.1417, Nov. 2012; also ISIT 2011.
- J. Leskovec, J. Kleinberg, C. Faloutsos, Graph Evolution: Densification and Shrinking Diameters, ACM Trans. on Knowl.Disc.Data, **1**(1) 2007.