

Lecture 9: Laplacian Eigenmaps

Radu Balan

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

April 12, 2018

Optimization Criteria

Assume $\mathcal{G} = (\mathcal{V}, W)$ is a undirected weighted graph with n nodes and weight matrix W .

We interpret $W_{i,j}$ as the *similarity* between nodes i and j . The larger the weight the more similar the nodes, and the closer they are in a geometric graph embedding.

Thus we look for a dimension $d > 0$ and a set of points

$\{y_1, y_2, \dots, y_n\} \subset \mathbb{R}^d$ so that $d_{i,j} = \|y_i - y_j\|$'s is small for large weight $W_{i,j}$.

Optimization Criteria

Assume $\mathcal{G} = (\mathcal{V}, W)$ is a undirected weighted graph with n nodes and weight matrix W .

We interpret $W_{i,j}$ as the *similarity* between nodes i and j . The larger the weight the more similar the nodes, and the closer they are in a geometric graph embedding.

Thus we look for a dimension $d > 0$ and a set of points

$\{y_1, y_2, \dots, y_n\} \subset \mathbb{R}^d$ so that $d_{i,j} = \|y_i - y_j\|$'s is small for large weight $W_{i,j}$.

A natural optimization criterion candidate:

$$J(y_1, y_2, \dots, y_n) = \sum_{1 \leq i, j \leq n} W_{i,j} \|y_i - y_j\|^2,$$

Optimization Criteria

Lemma

$$J(y_1, y_2, \dots, y_n) = \frac{1}{2} \sum_{1 \leq i, j \leq n} W_{i,j} \|y_i - y_j\|^2$$

is convex in (y_1, \dots, y_n) .

Proof Idea: Write it as a positive semidefinite quadratic criterion:

$$J = \sum_{i=1}^n \|y_i\|^2 \sum_{j=1}^n W_{i,j} - \sum_{i,j=1}^n W_{i,j} \langle y_i, y_j \rangle$$

Let $Y = [y_1 | \dots | y_n]$ be the $d \times n$ matrix of coordinates. Let $D = \text{diag}(d_k)$, with $d_k = \sum_{i=1}^n W_{k,i}$, be a $n \times n$ diagonal matrix. A little algebra shows:

$$J(Y) = \text{trace} \{ Y(D - W)Y^T \}.$$

Optimization Criteria

Equivalent forms:

$$J(Y) = \text{trace} \{ Y(D - W)Y^T \} = \text{trace} \{ Y\Delta Y^T \} = \text{trace} \{ \Delta G \}$$

where $G = Y^T Y$ is the $n \times n$ Gram matrix. Thus: J is quadratic in Y , and positive semidefnite, hence convex.

Also: J is linear in G .

Optimization Criteria

Equivalent forms:

$$J(Y) = \text{trace} \{ Y(D - W)Y^T \} = \text{trace} \{ Y\Delta Y^T \} = \text{trace} \{ \Delta G \}$$

where $G = Y^T Y$ is the $n \times n$ Gram matrix. Thus: J is quadratic in Y , and positive semidefnite, hence convex.

Also: J is linear in G .

Question: Are there other convex functions in Y that behave similarly?

Optimization Criteria

Equivalent forms:

$$J(Y) = \text{trace} \{ Y(D - W)Y^T \} = \text{trace} \{ Y\Delta Y^T \} = \text{trace} \{ \Delta G \}$$

where $G = Y^T Y$ is the $n \times n$ Gram matrix. Thus: J is quadratic in Y , and positive semidefnite, hence convex.

Also: J is linear in G .

Question: Are there other convex functions in Y that behave similarly?

Answer: Yes! Examples:

$$J(y_1, \dots, y_n) = \sum_{1 \leq i, j \leq n} W_{i,j} \|y_i - y_j\|^2$$

Optimization Criteria

Equivalent forms:

$$J(Y) = \text{trace} \{ Y(D - W)Y^T \} = \text{trace} \{ Y\Delta Y^T \} = \text{trace} \{ \Delta G \}$$

where $G = Y^T Y$ is the $n \times n$ Gram matrix. Thus: J is quadratic in Y , and positive semidefnite, hence convex.

Also: J is linear in G .

Question: Are there other convex functions in Y that behave similarly?

Answer: Yes! Examples:

$$J(y_1, \dots, y_n) = \sum_{1 \leq i, j \leq n} W_{i,j} \|y_i - y_j\|$$

$$J(y_1, \dots, y_n) = \left(\sum_{1 \leq i, j \leq n} W_{i,j} \|y_i - y_j\|^p \right)^{1/p}, \quad p \geq 1$$

Constraints

Absent any constraint,

$$\text{minimize } \text{trace} \{ Y \Delta Y^T \}$$

has solution $Y = 0$. To avoid this trivial solution, we impose a normalization constraint.

Choices:

$$Y Y^T = I_d$$

$$Y D Y^T = I_d$$

What does this mean?

$$\sum_{k=1}^n y_k y_k^T = I_d \Rightarrow \text{Parseval frame}$$

$$\sum_{k=1}^n d_k y_k y_k^T = I_d \Rightarrow \text{Parseval weighted frame}$$

The Optimization Problem

Combining one criterion with one constraint:

$$(LE) : \begin{array}{ll} \text{minimize} & \text{trace} \left\{ Y \Delta Y^T \right\} \\ \text{subject to} & Y D Y^T = I_d \end{array}$$

called the *Laplacian Eigenmap* problem.

Alternative problem:

$$(UnLE) : \begin{array}{ll} \text{minimize} & \text{trace} \left\{ Y \Delta Y^T \right\} \\ \text{subject to} & Y Y^T = I_d \end{array}$$

called the *unnormalized Laplacian eigenmap* problem.

The optimization problem

How to solve the Laplacian eigenmap problem:

$$(LE) : \begin{array}{ll} \text{minimize} & \text{trace} \left\{ Y \Delta Y^T \right\} \\ \text{subject to} & Y D Y^T = I_d \end{array}$$

First note the problem is not convex, because of the equality constraint. How to make it convex? How to solve?

1. First absorb the scaling D into the solution:

$$\tilde{Y} = Y D^{1/2}$$

Problem becomes:

$$\begin{array}{ll} \text{minimize} & \text{trace} \left\{ \tilde{Y} D^{-1/2} \Delta D^{-1/2} \tilde{Y}^T \right\} = \text{trace} \left\{ \tilde{Y} \tilde{\Delta} \tilde{Y}^T \right\} \\ \text{subject to} & \tilde{Y} \tilde{Y}^T = I_d \end{array}$$

The optimization problem

2. Consider the optimization problem for P :

$$\begin{aligned} & \text{minimize} && \text{trace} \left\{ \tilde{\Delta} P \right\} \\ & \text{subject to} && P = P^T \geq 0 \\ & && P \leq I_n \\ & && \text{trace}(P) = d \end{aligned}$$

Proposition

Claims:

- A. *The above optimization problem is a convex SDP.*
- B. *At optimum: $P = \tilde{Y}^T \tilde{Y}$.*

Eigenproblem

The optimum solutions of the (LE) and (UnLE) problems are given by appropriate eigenvectors:

$$\begin{array}{ll} \text{minimize} & \text{trace} \left\{ \tilde{Y} \tilde{\Delta} \tilde{Y}^T \right\} \\ \text{subject to} & \tilde{Y} \tilde{Y}^T = I_d \end{array}$$

Solution:

$$\tilde{Y} = \begin{bmatrix} e_1^T \\ \vdots \\ e_d^T \end{bmatrix}, \quad \tilde{\Delta} e_k = \lambda_k e_k$$

where $0 = \lambda_1 \leq \dots \leq \lambda_d$ are the smallest d eigenvalues, and $\|e_k\| = 1$ are the normalized eigenvectors.

Generalized Eigenproblem

$$(LE) : \begin{array}{ll} \text{minimize} & \text{trace} \left\{ Y \Delta Y^T \right\} \\ \text{subject to} & Y D Y^T = I_d \end{array} \Rightarrow Y = \tilde{Y} D^{-1/2}$$

the rows of \tilde{Y} are eigenvectors of the normalized Laplacian $\tilde{\Delta} e_k = \lambda_k e_k$.
Let f_k be the (transpose) rows of Y :

$$Y = \begin{bmatrix} f_1^T \\ \vdots \\ f_d^T \end{bmatrix}, \quad f_k = D^{-1/2} e_k$$

Thus: $\tilde{\Delta} D^{1/2} f_k = \lambda_k D^{1/2} f_k$, or: $D^{1/2} \tilde{\Delta} D^{1/2} f_k = \lambda_k D f_k$, or:

$$\Delta f_k = \lambda_k D f_k$$

This is called *generalized eigenproblem* associated to (Δ, D) .

Eigenproblem

Consider the unnormalized Laplacian eigenmap problem:

$$(UnLE) : \begin{array}{ll} \text{minimize} & \text{trace} \{ Y \Delta Y^T \} \\ \text{subject to} & Y Y^T = I_d \end{array} .$$

The solution Y^{unLE} is the $d \times n$ matrix whose rows are eigenvectors of the unnormalized Laplacian $\Delta = D - W$, $\Delta g_k = \mu_k g_k$, $\|g_k\| = 1$, $0 = \mu_1 \leq \dots \leq \mu_d$, and

$$Y^{unLE} = \begin{bmatrix} g_1^T \\ \vdots \\ g_d^T \end{bmatrix} .$$

What eigenspace to choose?

In most implementations one skips the eigenvectors associated to 0 eigenvalue. Why? In the unnormalized case, $g_1 = \frac{1}{\sqrt{n}}[1, 1, \dots, 1]^T$, hence no new information.

In your class projects, skip the bottom eigenvector. This fact is explicitly stated in the problem.

Laplacian Eigenmaps Embedding

Algorithm

Algorithm (Laplacian Eigenmaps)

Input: Weight matrix W , target dimension d

- 1 Construct the diagonal matrix $D = \text{diag}(D_{ii})_{1 \leq i \leq n}$, where $D_{ii} = \sum_{k=1}^n W_{i,k}$.
- 2 Construct the normalized Laplacian $\tilde{\Delta} = I - D^{-1/2}WD^{-1/2}$.
- 3 Compute the bottom $d + 1$ eigenvectors e_1, \dots, e_{d+1} , $\tilde{\Delta}e_k = \lambda_k e_k$, $0 = \lambda_1 \cdots \lambda_{d+1}$.

Laplacian Eigenmaps Embedding

Algorithm-cont's

Algorithm (Laplacian Eigenmaps - cont'd)

- 4 Construct the $d \times n$ matrix Y ,

$$Y = \begin{bmatrix} e_2 \\ \vdots \\ e_{d+1} \end{bmatrix} D^{-1/2}$$

- 5 The new geometric graph is obtained by converting the columns of Y into n d -dimensional vectors:

$$\left[y_1 \mid \cdots \mid y_n \right] = Y$$








Output: Set of points $\{y_1, y_2, \dots, y_n\} \subset \mathbb{R}^d$.

Example

see:

<http://www.math.umd.edu/~rvbalan/TEACHING/AMSC663Fall2010/PROJECTS/P5/index.html>

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. Erdős, 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.



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.