Manifold learning, the heat equation and spectral clustering

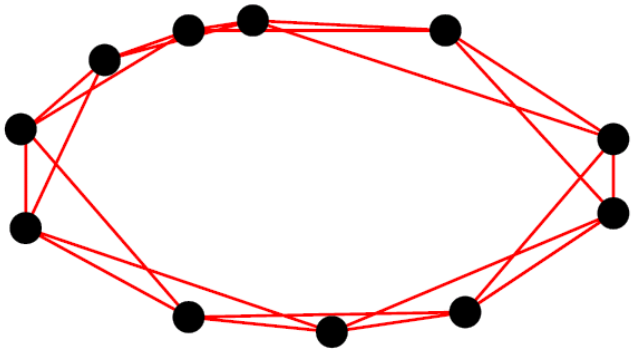
Mikhail Belkin
Dept. of Computer Science and Engineering,
Dept. of Statistics
Ohio State University

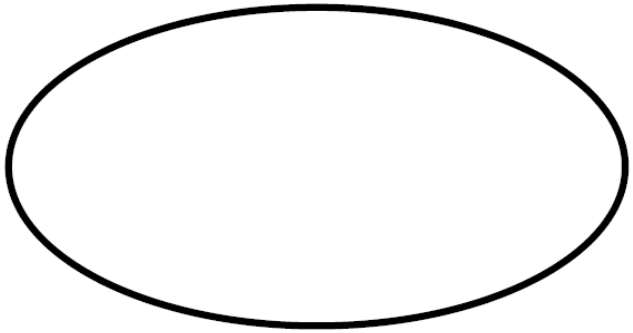
Collaborators: Partha Niyogi, Hariharan Narayanan, Jian Sun, Yusu Wang, Xueyuan Zhou

- In many domains data explicitly lies on a manifold.
- For all sources of high-dimensional data, true dimensionality is much lower than the number of features.
- Much of the data is highly nonlinear.
- In high dimension can trust local but not global distances.

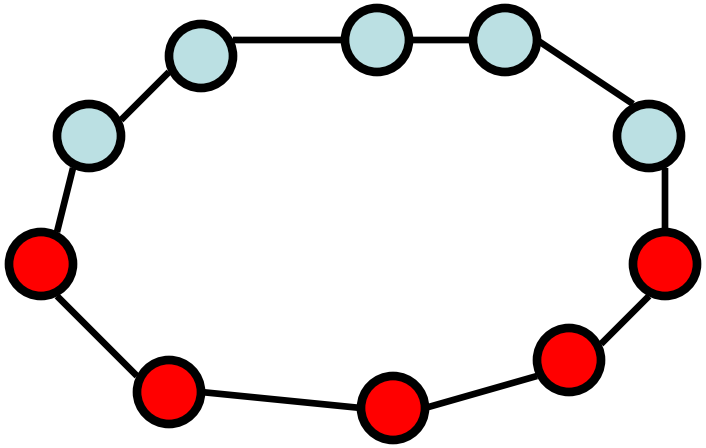
Manifolds (Riemannian manifolds with a measure + noise) provide a natural mathematical language for thinking about high-dimensional data.

How to extract manifold structure from data? Construct a graph to "represent" the underlying space. Properties of the graph should reflect properties of the manifold.

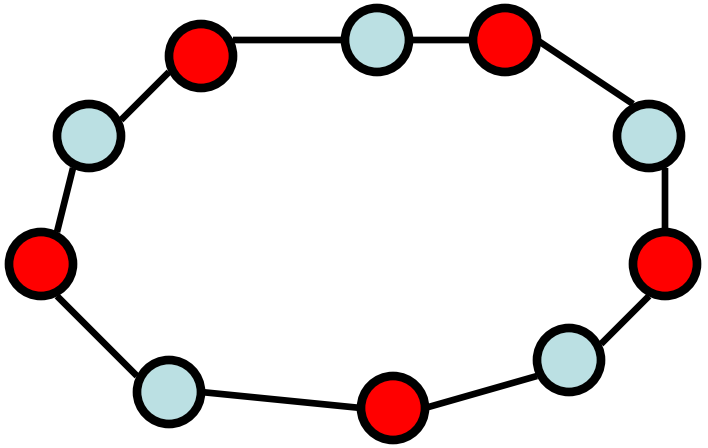




Simple intuition:



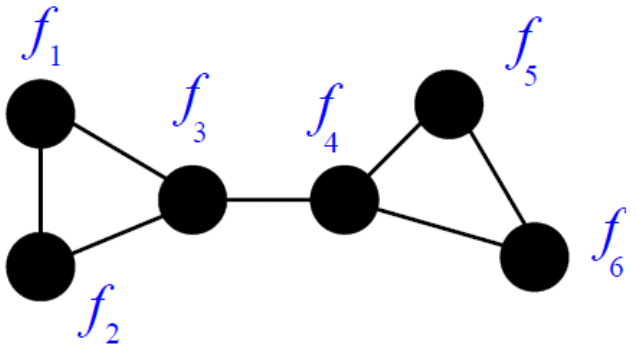
Good (plausible) classification function.



Bad (implausible) classification function.

Good functions have low "checkerboardedness".

Easy enough: graph Laplacian.



$$\mathbf{L} = \begin{pmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{pmatrix}$$

Natural smoothness functional (analogue of grad):

$$\mathcal{S}(\mathbf{f}) = (f_1-f_2)^2+(f_1-f_3)^2+(f_2-f_3)^2+(f_3-f_4)^2+(f_4-f_5)^2+(f_4-f_5)^2+(f_5-f_6)^2$$

Basic fact:

$$\mathcal{S}(\mathbf{f}) = \sum_{i\sim j}(f_i - f_j)^2 = \frac{1}{2}\,\mathbf{f}^t\,\mathbf{L}\,\mathbf{f}$$

But remember we are dealing with data.

Function $f : X \to \mathbb{R}$. Penalty at $x \in X$:

$$\frac{1}{\delta^k} \int_{\text{small } \delta} \left(f(x) - f(x+\delta)\right)^2 p(x) d\,\delta \approx \|\nabla f\|^2 p(x)$$

Total penalty – Laplace operator:

$$\int_X \|\nabla f\|^2 p(x) = \langle f, \Delta_p f \rangle_X$$

Two-class classification – conditional $P(1|x)$.

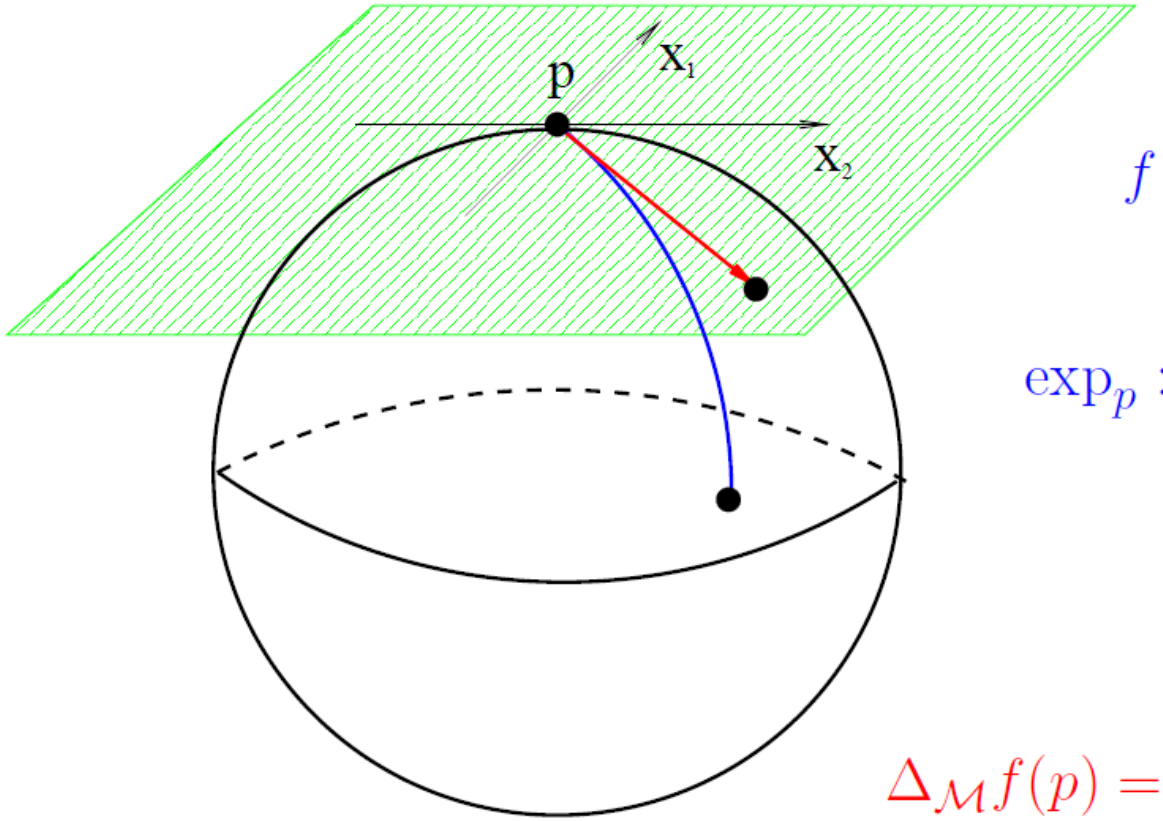
Manifold assumption: $\langle P(1|x), \Delta_p P(1|x) \rangle_X$ is small.

(**Note**: this condition not quite strong enough for regularization, but close – more later.)

$$\Delta f = -\sum_{i=1}^{k} \frac{\partial^2 f}{\partial x_i^2}$$

Fundamental mathematical object. Heat, wave, Schroedinger equations. Fourier Analysis.

What is Laplace operator on a circle?



$$f : \mathcal{M}^k \to \mathbb{R}$$

$$\exp_p : T_p\mathcal{M}^k \to \mathcal{M}^k$$

$$\Delta_{\mathcal{M}} f(p) = -\sum_i \frac{\partial^2 f(\exp_p(x))}{\partial x_i^2}$$

Generalization of Fourier analysis.

Nice math, but how to compute from data? Answer: the heat equation.

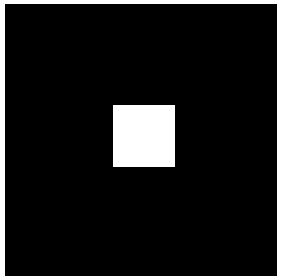
Heat equation in $\mathbb{R}^n$:

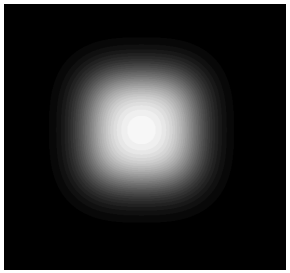
$u(x,t)$ – heat distribution at time t.
$u(x,0) = f(x)$ – initial distribution. $x \in \mathbb{R}^n, t \in \mathbb{R}$.

$$\Delta_{\mathbb{R}^n} u(x,t) = \frac{du}{dt}(x,t)$$

Solution – convolution with the heat kernel:

$$u(x,t) = (4\pi t)^{-\frac{n}{2}} \int_{\mathbb{R}^n} f(y) e^{-\frac{\|x-y\|^2}{4t}} dy$$





Functional approximation:

Taking limit as $t \to 0$ and writing the derivative:

$$\Delta_{\mathbb{R}^n} f(x) = \frac{d}{dt}\left[(4\pi t)^{-\frac{n}{2}} \int_{\mathbb{R}^n} f(y) e^{-\frac{\|x-y\|^2}{4t}} dy\right]_0$$

$$\Delta_{\mathbb{R}^n} f(x) \approx -\frac{1}{t}(4\pi t)^{-\frac{n}{2}} \left(f(x) - \int_{\mathbb{R}^n} f(y) e^{-\frac{\|x-y\|^2}{4t}} dy\right)$$

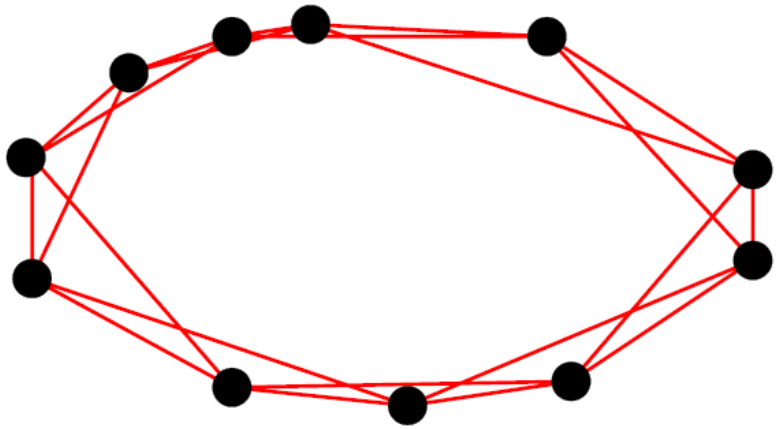
Empirical approximation:

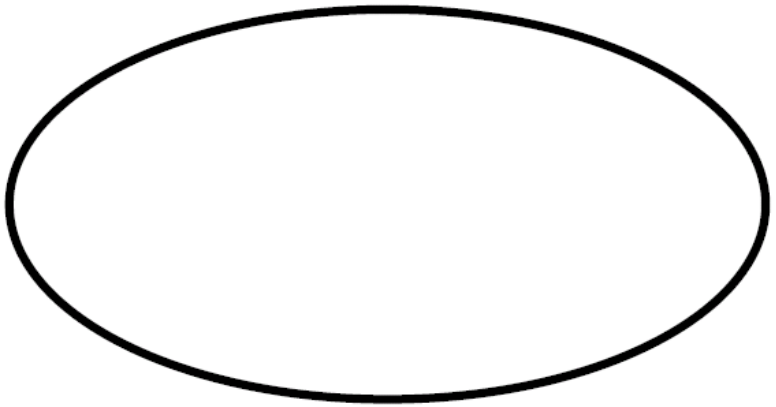
Integral can be estimated from empirical data.

$$\Delta_{\mathbb{R}^n} f(x) \approx -\frac{1}{t}(4\pi t)^{-\frac{n}{2}} \left(f(x) - \sum_{x_i} f(x_i) e^{-\frac{\|x-x_i\|^2}{4t}}\right)$$

The heat equation has a similar form on manifolds. However do not know distances and the heat kernel.

Turns out (by careful analysis using differential geometry) that these issues do not affect algorithms.





$$W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}}$$

$$Lf(x_i) = f(x_i) \sum_j e^{-\frac{\|x_i - x_j\|^2}{t}} - \sum_j f(x_j) e^{-\frac{\|x_i - x_j\|^2}{t}}$$

$$\mathbf{f}^t \mathbf{L} \mathbf{f} = 2 \sum_{i \sim j} e^{-\frac{\|x_i - x_j\|^2}{t}} (f_i - f_j)^2$$

Reconstructing eigenfunctions of Laplace-Beltrami operator from sampled data (Laplacian Eigenmaps, Belkin, Niyogi 2001).

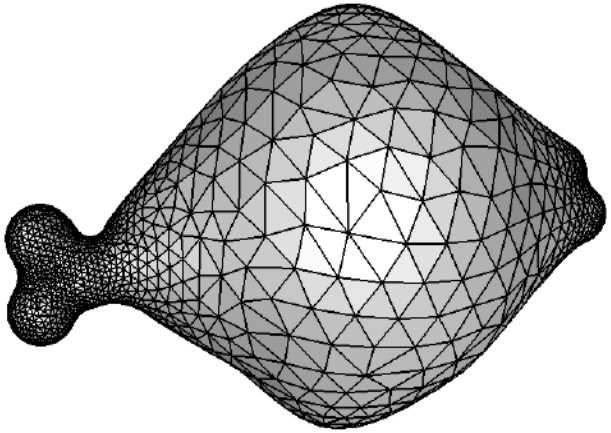
1. Construct a data-dependent weighted graph.

2. Compute the bottom eigenvectors of the Laplacian matrix.

Theoretical guarantees as data goes to infinite (using data-dependent t).

Out-of-sample extension?

Non-probabilistic data, such as meshes.



Belkin, Sun, Wang 07,09

Mesh *K*, triangle *t*.

$$L_K^t f(x_i) = \frac{1}{4\pi t^2} \sum_{t \in K} \frac{A(t)}{3} \sum_{v \in t} e^{-\frac{\|v - x_i\|^2}{4t}} (f(v) - f(x_i))$$

Surface *S*. Mesh K_ε.

Theorem:

$$\lim_{\epsilon \to 0} \sup_{K_\epsilon} \left\| \mathrm{L}_{K_\epsilon}^{h(\epsilon)} f - \Delta_S f \right\|_\infty = 0$$

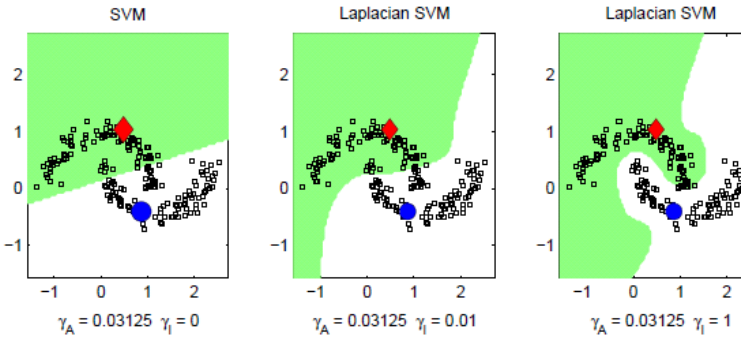
$$h(\epsilon) = \epsilon^{\frac{1}{2.5+\alpha}}$$

Can be extended to arbitrary point clouds. Idea – only need a mesh locally.

Some applications:

- Data representation/visualization.

- Semi-supervised learning.

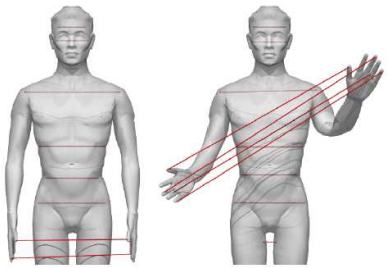


- Isometry-invariant representations.

Corazza, Andriacchi, Stanford Biomotion Lab, 05, Partiview, Surendran

Symmetry detection.

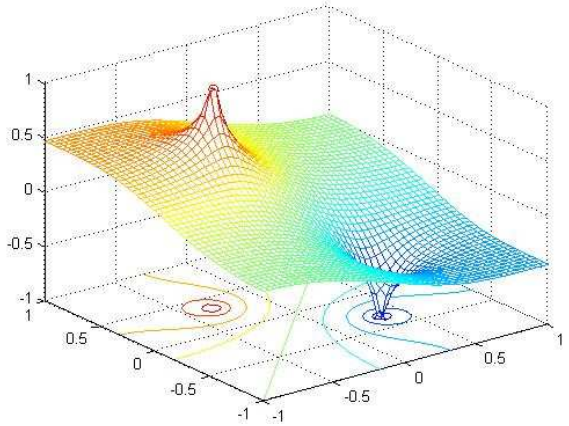
Ovsyannikov, Sun, Guibas, 2009



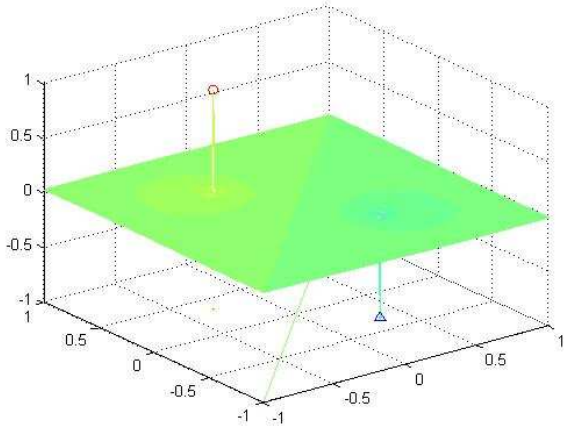
Caution about using using Laplacian for regularization

- In SSL, the more unlabeled data, the better results we expect
- However…

Less unlabeled data



More unlabeled data



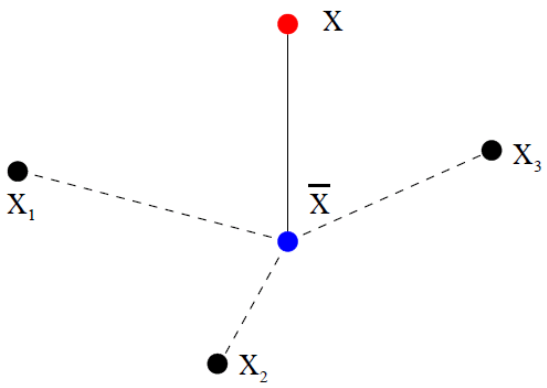
"Indicator" functions of labeled points.

The more unlabeled data we have, the less stable the classifier gets, and the worse the results become.
Laplacian is not powerful enough (has to do with properties of Sobolev spaces). However can use iterated Laplacian. (Recent work with X. Zhou)

1. Construct Neighborhood Graph.

2. Let $x_1, \ldots, x_n$ be neighbors of x. Project x to the span of $x_1, \ldots, x_n$.

3. Find barycentric coordinates of $\bar{x}$.



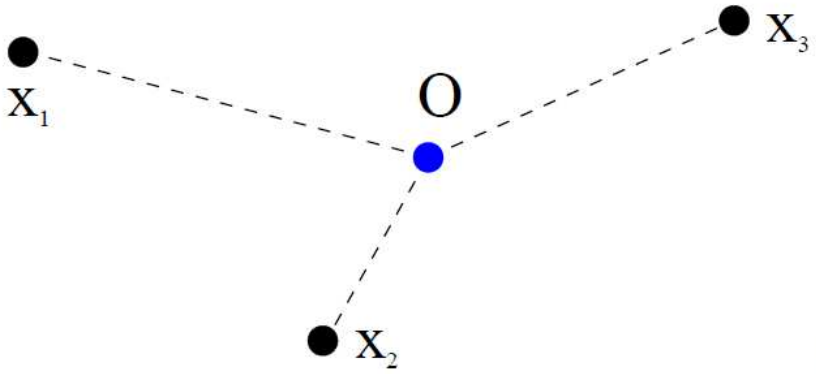
$$\bar{x} = w_1 x_1 + w_2 x_2 + w_3 x_3$$

$$w_1 + w_2 + w_3 = 1$$

Weights w_1, w_2, w_3 chosen, so that $\bar{x}$ is the center of mass.

4. Construct sparse matrix W. i th row is barycentric coordinates of $\bar{x}_i$ in the basis of its nearest neighbors.

5. Use lowest eigenvectors of $(I - W)^t (I - W)$ to embed.



$$\sum w_i x_i = 0$$

$$\sum w_i = 1$$

Hessian H. Taylor expansion :

$$f(x_i) = f(0) + x_i^t \nabla f + \frac{1}{2} x_i^t H x_i + o(\|x_i\|^2)$$

$$(I - W)f(0) = f(0) - \sum w_i f(x_i) \approx f(0) - \sum w_i f(0) - \sum_i w_i x_i^t \nabla f - \frac{1}{2} \sum_i x_i^t H x_i =$$

$$= -\frac{1}{2} \sum_i x_i^t H x_i \approx -trH = \Delta f$$

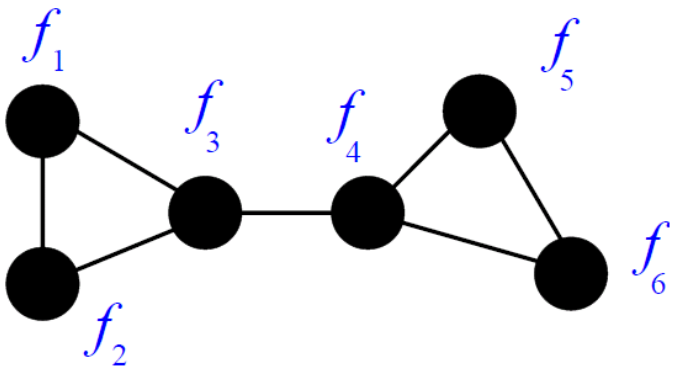
Convergence to the Laplacian in the “limit”. (But the algorithm does not actually allow that).

Embed using weighted eigenfunctions of the Laplacian:

$$x \rightarrow (e^{-\lambda_1 t}\mathbf{f}_1(x), e^{-\lambda_2 t}\mathbf{f}_2(x), \ldots)$$

Diffusion distance is (approximated by) the distance between the embedded points.

Closely related to random walks on graphs.

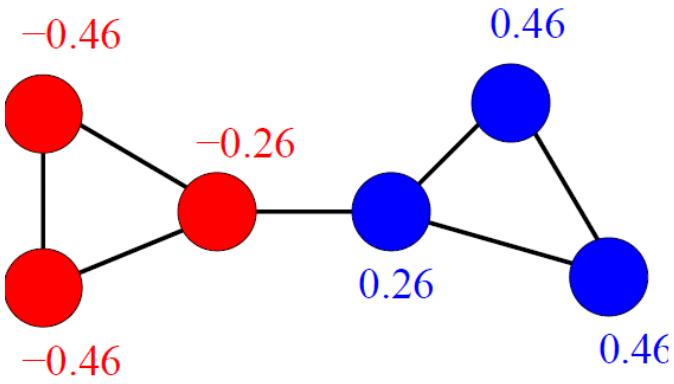


$$\mathbf{L} = \begin{pmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{pmatrix}$$

$$\operatorname*{argmin}_{S} \sum_{i \in S,\ j \in V-S} w_{ij} = \operatorname*{argmin}_{f_i \in \{-1,1\}} \sum_{i \sim j} (f_i - f_j)^2 = \frac{1}{8} \operatorname*{argmin}_{f_i \in \{-1,1\}} \mathbf{f}^t \mathbf{L}\, \mathbf{f}$$

Relaxation gives eigenvectors.

$$\mathbf{L}v = \lambda v$$



$$\mathbf{L} = \begin{pmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{pmatrix}$$

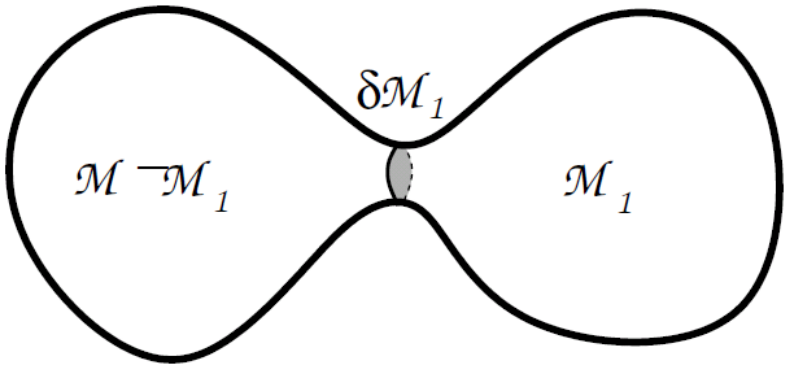
Unnormalized clustering:

$$L\mathbf{e_1} = \lambda_1 \mathbf{e_1} \qquad \mathbf{e_1} = [-0.46, -0.46, -0.26, 0.26, 0.46, 0.46]$$

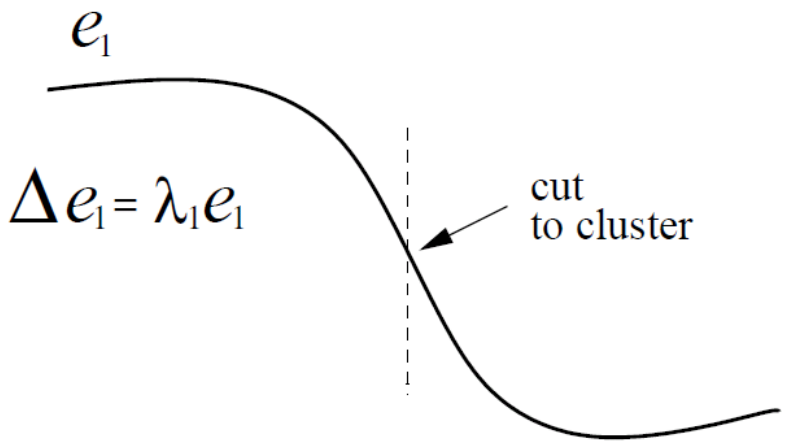
Normalized clustering:

$$L\mathbf{e_1} = \lambda_1 D\mathbf{e_1} \qquad \mathbf{e_1} = [-0.31, -0.31, -0.18, 0.18, 0.31, 0.31]$$

Laplacian eigenfunction as a relaxation of the isoperimetric problem.

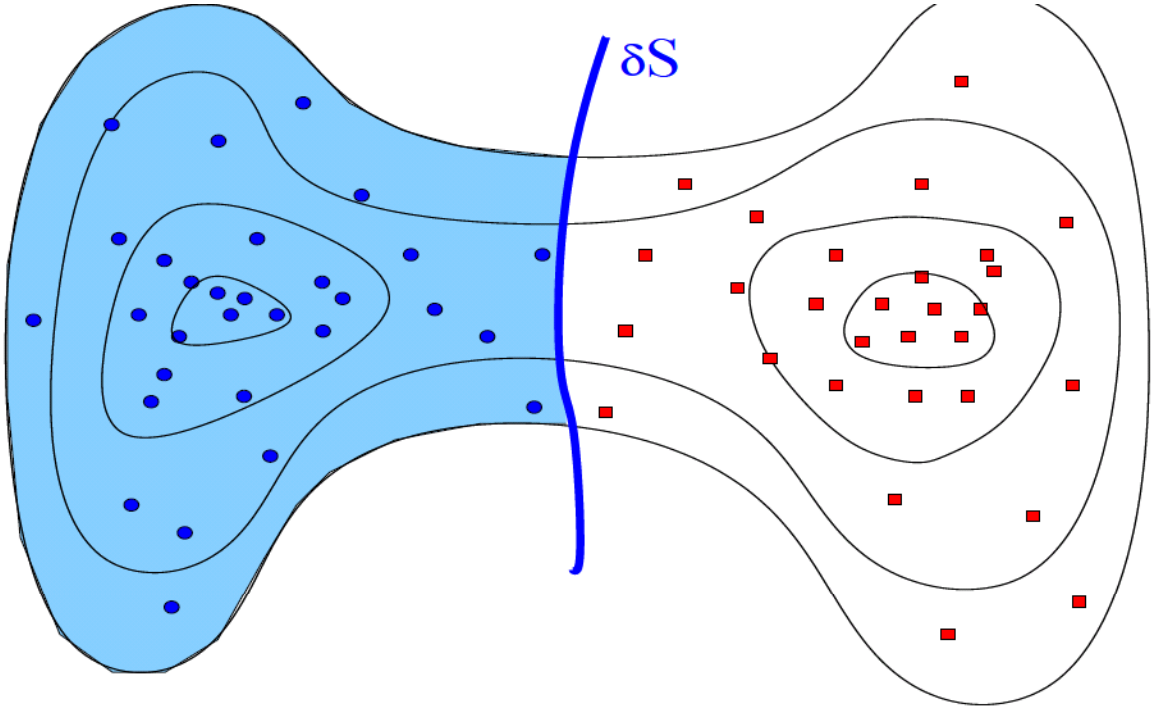


$$h = \inf \frac{\mathrm{vol}^{n-1}(\delta\mathcal{M}_1)}{\min\left(\mathrm{vol}^n(\mathcal{M}_1), \mathrm{vol}^n(\mathcal{M}-\mathcal{M}_1)\right)}$$



$$0 = \lambda_0 \leq \lambda_1 \leq \lambda_2 \leq \dots$$

$$h \leq \frac{\sqrt{\lambda_1}}{2}$$ [Cheeger]



$$\sum_{i \in \text{blue}} \sum_{j \in \text{red}} \frac{w_{ij}}{\sqrt{d_j d_j}}$$

$$w_{ij} = e^{-\frac{\|x_i - x_j\|^2}{4t}}$$

$$d_i = \sum_j w_{ij}$$

Theorem:

$$\text{vol}(\delta S) \approx \frac{2}{N} \frac{1}{(4\pi t)^{n/2}} \sqrt{\frac{\pi}{t}}\, \mathbf{1}_S^t\, L\, \mathbf{1}_S$$

L is the normalized graph Laplacian and $\mathbf{1}_S$ is the indicator vector of points in S. (Narayanan Belkin Niyogi, 06)

Laplacian and the heat equation are a key bridge between data, algorithms and classical mathematics.

Data analysis --- Differential Geometry --- Differential Equations --- Functional Analysis --- Numerical methods --- Algorithms