A Duality View of Spectral Methods for Dimensionality Reduction

Lin Xiao Jun Sun Stephen Boyd
Microsoft Research Stanford University

Connection II Workshop
Caltech
August 17, 2006

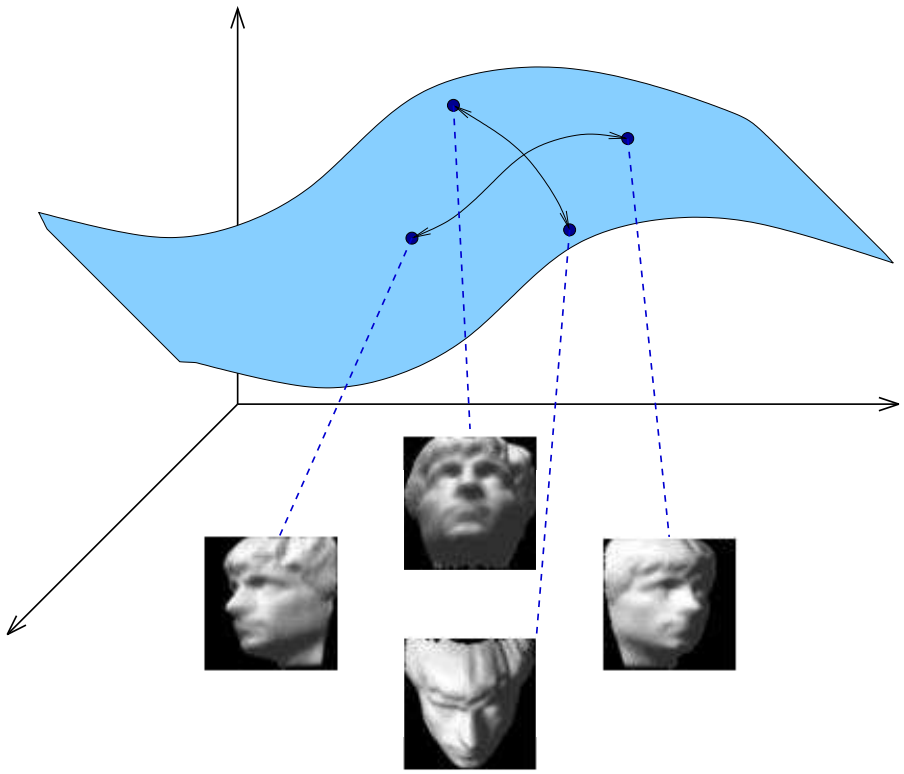
Dimensionality reduction

problem: extract low dimensional structure from high dimensional data

- example: 64×64 grayscale images, with dimension 4096
- degrees of freedom:

  left $\leftrightarrow$ right
  up $\leftrightarrow$ down

  and others, $e.g.$, lighting
- reside on 2 or 3 dimensional submanifold in $\mathbf{R}^{4096}$

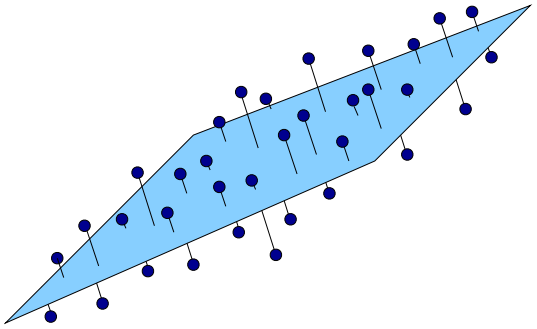


goal: discover low dimensional structure, compute faithful representations

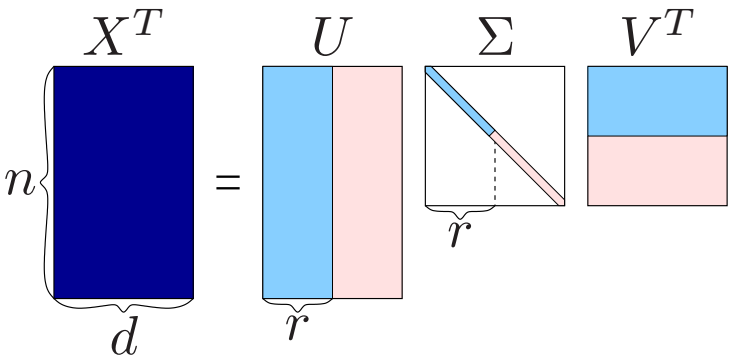
Linear versus nonlinear

- linear

$$x_1, \ldots, x_n \in \mathbf{R}^d$$

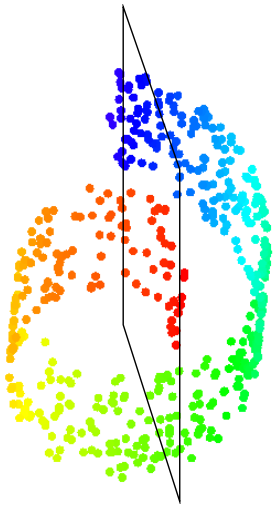


- singular value decomposition (SVD)

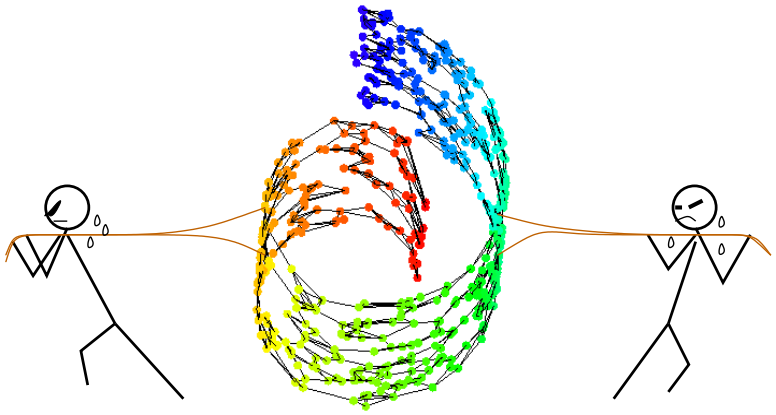


PCA: Principle Component Analysis
MDS: metric MultiDimensional Scaling

- nonlinear



- solution?



Nonlinear dimensionality reduction

- **spectral methods**
  - Isomap: (Tenenbaum, de Silva & Langford, 2000)
  - locally linear embedding (Roweis &Saul, 2000)
  - Laplacian eigenmaps (Belkin & Niyogi, 2002)
  - Hessian eigenmaps (Donoho & Grimes, 2003)
  - maximum variance unfolding (Weinberger & Saul, 2004)
  - local tangent space alignment (Zhang & Zha, 2004)
  - geodesic nullspace analysis (Brand, 2004)
  - conformal eigenmaps (Sha & Saul, 2005)
- **similar computational structure:**
  - establish k-nearest neighbor graph
  - construct a square matrix: **dense** or **sparse**
  - eigenvalue decomposition: **top of dense** or **bottom of sparse**
- **question:** what are the connections between these methods?

Outline

- brief overview of some spectral methods
- duality theory of maximum variance unfolding (MVU)
- a unified duality view of spectral methods
  - MVU and Isomap
  - MVU and locally linear embedding
  - MVU and Laplacian eigenmap
- connections to Markov chains and networked systems

Principle component analysis (PCA)

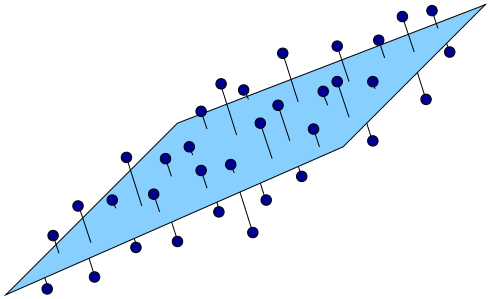
- **goal:** preserve covariance structure

$$X = [x_1 \cdots x_n],\ x_i \in \mathbf{R}^d$$

(assume $\sum x_i = 0$)

minimize $\displaystyle\sum_{i=1}^{n} \|x_i - Px_i\|^2$

P: projection matrix, rank $r < d$

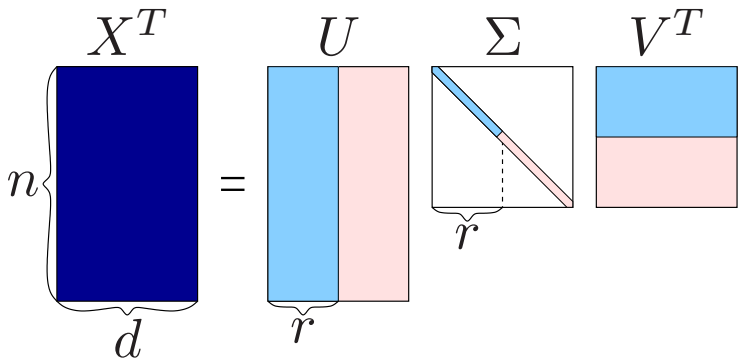


- equivalently, maximize projected variance

maximize $\displaystyle\sum_{i=1}^{n} \|Px_i\|^2 = \sum_{i=1}^{n} \|y_i\|^2 = \frac{1}{2n} \sum_{i,j=1}^{n} \|y_i - y_j\|^2$

- **solution:** SVD

$$P = V_r V_r^T, \qquad y_i = V_r^T x_i$$



Metric multidimensional scaling (MDS)

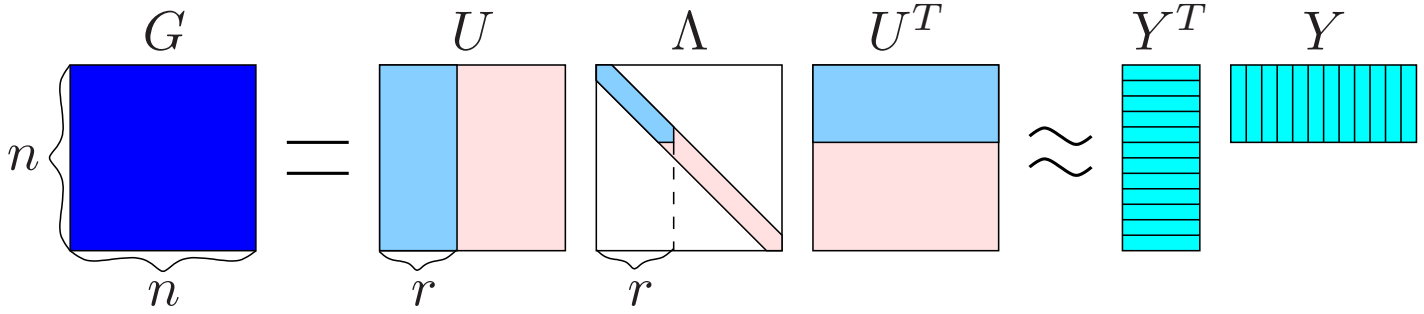
- **goal:** find $y_1, \ldots, y_n \in \mathbf{R}^r$ to faithfully preserve inner products

$$\text{minimize} \quad \textstyle\sum_{i,j}(x_i^T x_j - y_i^T y_j)^2 = \|X^T X - Y^T Y\|_F^2$$

- **solution:**
  - compute Gram matrix from pair-wise distances $D_{ij} = \|x_i - x_j\|^2$

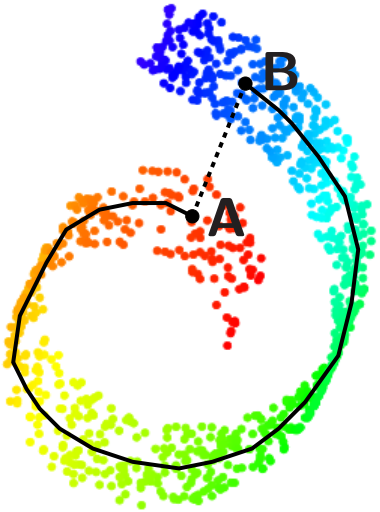
$$G = X^T X = -\tfrac{1}{2}\left(I - \tfrac{1}{n}\mathbf{1}\mathbf{1}^T\right) D \left(I - \tfrac{1}{n}\mathbf{1}\mathbf{1}^T\right)$$

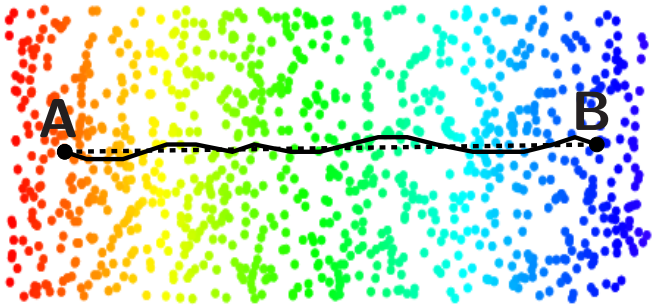
  - eigenvalue decomposition of Gram matrix



Isomap

- PCA and MDS only work for linear projection, need nonlinear extensions





- **key idea:** use geodesic instead of Euclidean distance in MDS
  - construct adjacency graph, e.g., connect k-nearest neighbors
  - estimate geodesic distance by shortest path: e.g., Djikstra's algorithm
  - use geodesic distances to compute Gram matrix G in MDS

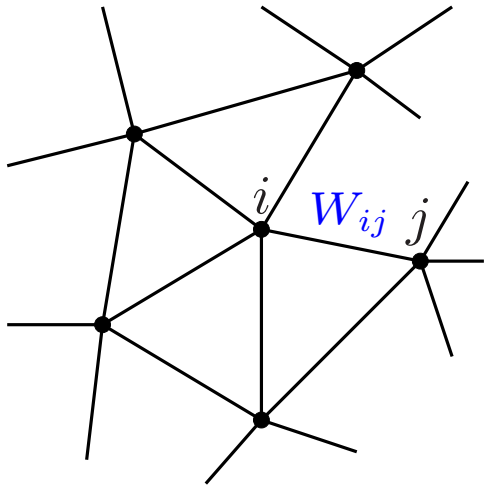
Locally linear embedding (LLE)

key idea: explore local linearity: $x_i \approx \sum_{j \in \mathcal{N}_i} W_{ij} x_j$ (say, in tangent space)

- least-square fitting to find sparse matrix W

$$\text{minimize} \quad \sum_{i=1}^{n} \left\| x_i - \sum_{j \in \mathcal{N}_i} W_{ij} x_j \right\|^2$$

$$\text{subject to} \quad \sum_{j \in \mathcal{N}_i} W_{ij} = 1, \quad i = 1, \ldots, n$$



- least-square reconstruction of $y_1, \ldots, y_n$

$$\text{minimize} \quad \sum_{i=1}^{n} \left\| y_i - \sum_{j \in \mathcal{N}_i} W_{ij} y_j \right\|^2, \qquad \text{subject to} \quad \frac{1}{n} \sum_{i=1}^{n} y_i y_i^T = I_r$$

solution: compute bottom $r+1$ eigenvectors of $(I-W)^T(I-W)$, use r of them to reconstruct y_i (discard $\mathbf{1}$ associated with $\lambda_{\min} = 0$)

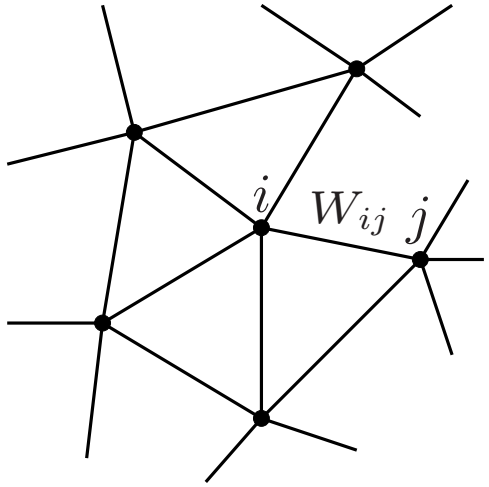
Laplacian eigenmap

key idea: map nearby inputs to nearby outputs, preserve locality on graph

- assign weights on edges $\{i,j\} \in \mathcal{E}$

$$W_{ij} = 1 \quad \text{or} \quad W_{ij} = \exp(-\beta \|x_i - x_j\|^2)$$

$$\text{Laplacian:} \quad L_{ij} = \begin{cases} -W_{ij} & \{i,j\} \in \mathcal{E} \\ \sum_{k \in \mathcal{N}_i} W_{ik} & i = j \\ 0 & \text{otherwise} \end{cases}$$



- find low dimensional representations

$$\text{minimize} \quad \sum_{\{i,j\} \in \mathcal{E}} W_{ij} \|y_i - y_j\|^2, \qquad \text{subject to} \quad \sum_{i=1}^{n} L_{ii} y_i y_i^T = I_r$$

- **solution:** construct y_i's from r bottom (generalized) eigenvectors of L

Maximum variance unfolding (MVU)

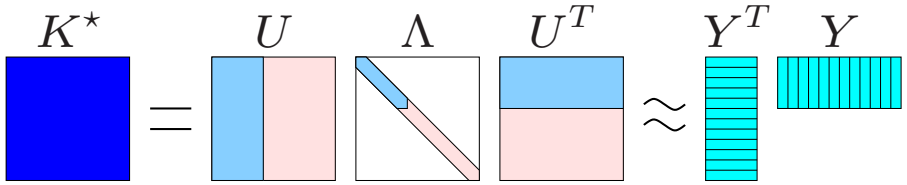
- compute k-nearest neighbor graph

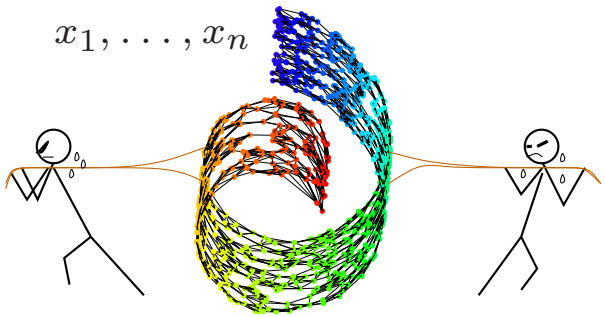
- QP formulation (**nonconvex**)

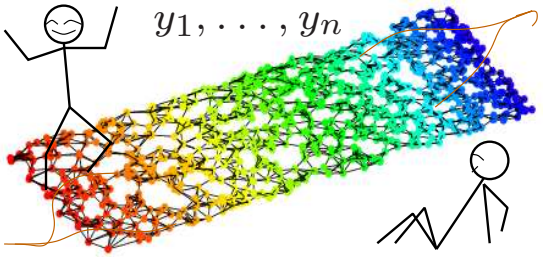
$$\begin{array}{ll} \text{max.} & \sum_{i=1}^n \|y_i\|^2 \\ \text{s. t.} & \|y_i - y_j\| = \|x_i - x_j\|, \quad \{i,j\} \in \mathcal{E} \\ & \sum_{i=1}^n y_i = 0 \end{array}$$

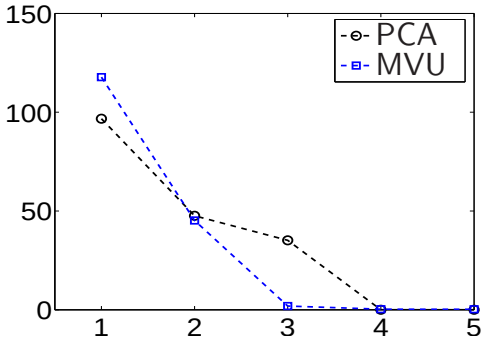
- SDP formulation (**convex**): let $K_{ij} = y_i^T y_j$

$$\begin{array}{ll} \text{max.} & \mathbf{Tr}\, K \\ \text{s. t.} & K_{ii} + K_{jj} - 2K_{ij} = D_{ij}, \quad \{i,j\} \in \mathcal{E} \\ & K = K^T \succeq 0, \quad \mathbf{1}^T K \mathbf{1} = 0 \end{array}$$



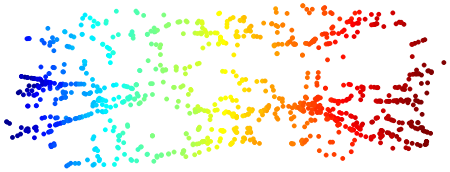




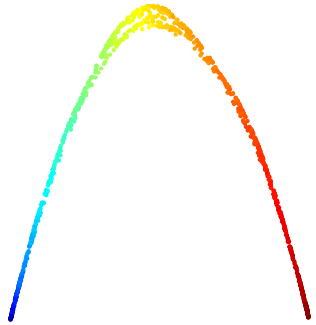


Swissroll example

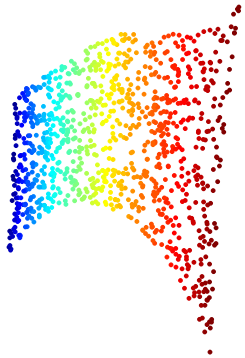
$n = 1000,\ k = 10$



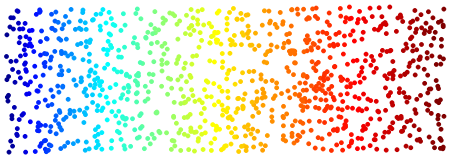
Isomap



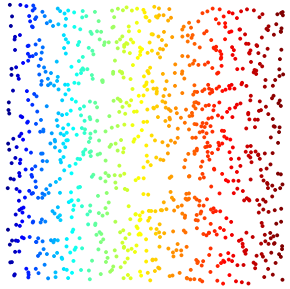
Laplacian eigenmap



LLE



MVU primal



MVU dual

Quest of unified views

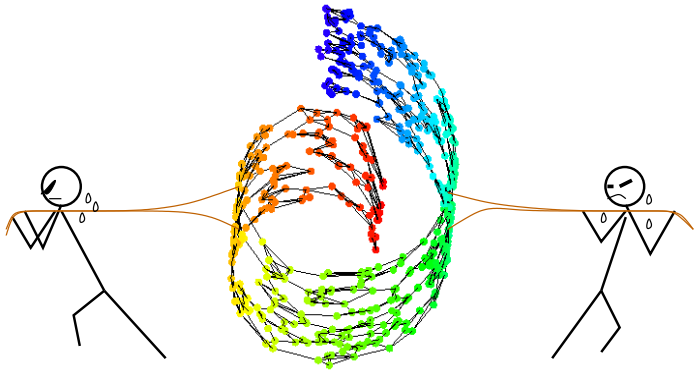
myth: these different methods are capable of producing similar results

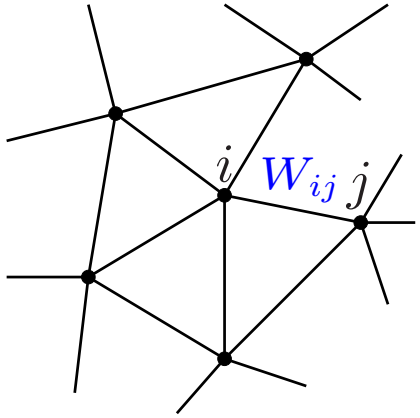
- MDS, Isomap, MVU
  - try to preserve **global** pairwise (geodesic) distances
  - use **top** eigenvectors of **dense** matrices
  - can estimate dimensionality from number of significant eigenvalues
- LLE, Laplacian eigenmap
  - try to preserve **local** geometric relationships
  - use **bottom** eigenvectors of **sparse** matrices
  - cannot estimate dimensionality from gap of eigenvalues

toward unified views

- each as an instance of kernel PCA (Ham, Lee, Mika, & Schölkopf, 2004)
- semidefinite programming duality theory (Xiao, Sun, & Boyd, 2006)

MVU duality theory





- **primal problem** $\longleftrightarrow$ **dual problem**

$$\begin{array}{ll} \text{max.} & \mathbf{Tr}\, K \\ \text{s. t.} & K = K^T \succeq 0, \quad \mathbf{1}^T K \mathbf{1} = 0 \\ & K_{ii} + K_{jj} - 2K_{ij} = D_{ij}, \quad \{i,j\} \in \mathcal{E} \end{array}$$

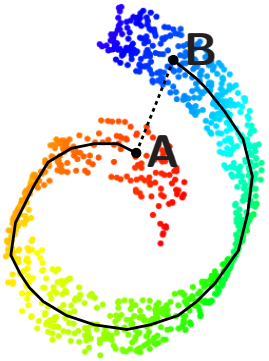
$$\begin{array}{ll} \text{min.} & \displaystyle\sum_{\{i,j\} \in \mathcal{N}_i} D_{ij} W_{ij} \\ \text{s. t.} & \lambda_2(L) \geq 1 \end{array}$$

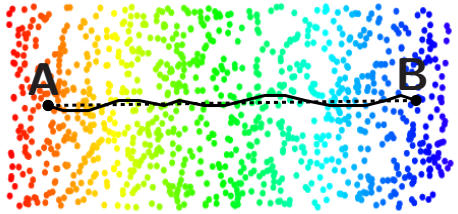
- **optimality conditions:** primal-dual feasibility, and complementarity

$$L^\star K^\star = K^\star \quad \Longrightarrow \quad \left\{ \begin{array}{l} \text{top of dense } K^\star \;=\; \text{bottom of sparse } L^\star \\ r \;\leq\; \mathbf{rank}\; K^\star \;\leq\; \text{multiplicity of } \lambda_2(L^\star) \end{array} \right.$$

Connection between MVU and Isomap

- Isomap: use geodesic instead of Euclidean distance in MDS





- Interpretation: try to construct optimal solution to MVU directly

$$\text{maximize} \quad \sum_{i=1}^{n} \|y_i\|^2 = \frac{1}{2n} \sum_{i,j=1}^{n} \|y_i - y_j\|^2 \leq \frac{1}{2n} \sum_{i,j=1}^{n} \text{geod}(i,j)^2$$

$$\text{subject to} \quad \sum_{i=1}^{n} y_i = 0, \qquad \|y_i - y_j\| = \|x_i - x_j\|, \quad \{i,j\} \in \mathcal{E}$$

- if data manifold isometric to convex subset of Euclidean space, then Isomap and MVU give same result in limit (increase sampling density)

Connection between MVU and LLE

- key idea of LLE: locally linear approximation: $x_i \approx \sum_{j \in \mathcal{N}_i} W_{ij} x_j$
  - least-square fitting to find sparse matrix W

    minimize $\sum_{i=1}^{n} \left\| x_i - \sum_{j \in \mathcal{N}_i} W_{ij} x_j \right\|^2$, subject to $\sum_{j \in \mathcal{N}_i} W_{ij} = 1, \quad i = 1, \ldots, n$
  - least-square reconstruction of $y_1, \ldots, y_n \in \mathbf{R}^r$

    minimize $\sum_{i=1}^{n} \left\| y_i - \sum_{j \in \mathcal{N}_i} W_{ij} y_j \right\|^2$, subject to $\frac{1}{n} \sum_{i=1}^{n} y_i y_i^T = I_r$
- interpretation from MVU optimality conditions. Let $K^\star = Y^T Y$

$$L^\star K^\star = K^\star \quad \Longrightarrow \quad L^\star Y^T = Y^T \quad \Longrightarrow \quad (L^\star_{ii} - 1) y_i = \sum_{i \in \mathcal{N}_i} W^\star_{ij} y_j$$

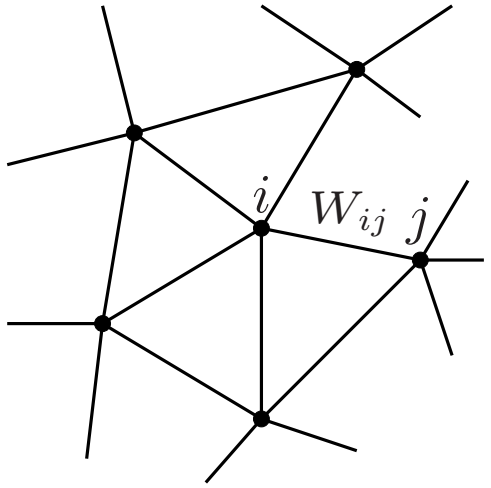
- however, computationally very different . . . needs further investigation

Connection between MVU and Laplacian eigenmap

- consider dual MVU problem

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{\{i,j\}\in\mathcal{E}} W_{ij}\|x_i - x_j\|^2 \\ \text{subject to} & \lambda_2(L) \geq 1 \end{array}$$

$$L_{ij} = \begin{cases} -W_{ij} & \{i,j\} \in \mathcal{E} \\ \sum_{k\in\mathcal{N}_i} W_{ik} & i = j \\ 0 & \text{otherwise} \end{cases}$$



- Laplacian eigenmap simply use feasible solutions to dual MVU problem

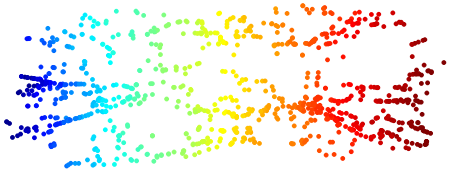
$$W_{ij} = 1 \quad \text{or} \quad W_{ij} = \exp(-\beta\|x_i - x_j\|^2) \qquad \text{(satisfy } \lambda_2(L) \geq 1 \text{ by scaling)}$$

- construct $y_1, \ldots, y_n \in \mathbf{R}^r$ from r bottom eigenvectors of L

$$\text{minimize} \quad \sum_{\{i,j\}\in\mathcal{E}} W_{ij}\|y_i - y_j\|^2, \qquad \text{subject to} \quad \sum_{i=1}^{n} y_i y_i^T = I_r$$

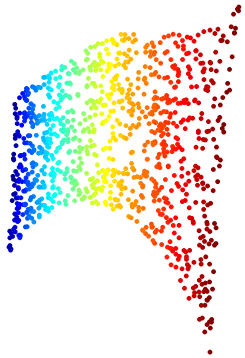
Swissroll example

$n = 1000,\ k = 10$

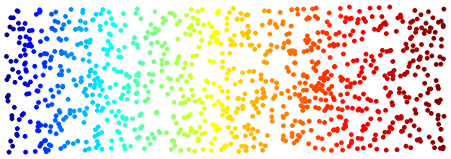


Isomap

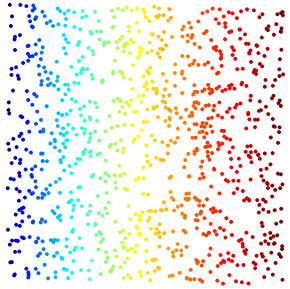
Laplacian eigenmap



LLE



MVU primal

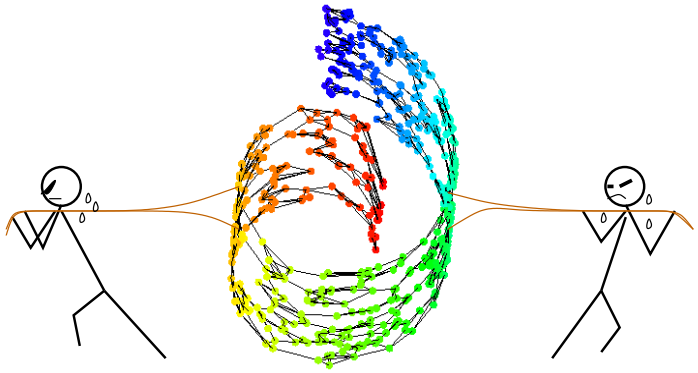


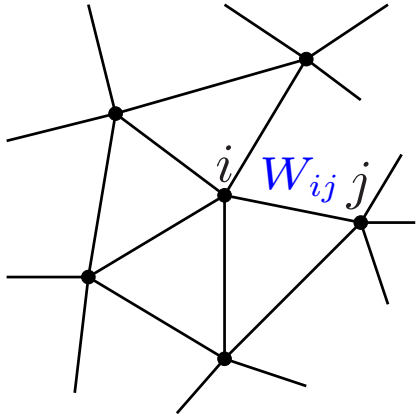
MVU dual

A unified duality view

- connections between spectral methods
  - Isomap: construct (approximate) solution to primal MVU problem
  - LLE: motivation interpreted from MVU optimality conditions
  - Laplacian eigenmap: feasible dual MVU solutions, can be optimized
- key insights from MVU optimality condition $L^\star K^\star = K^\star$
  - top eigenspace of dense $K^\star$ = bottom eigenspace of sparse $L^\star$
  - embedding dimension $r \leq$ rank of $K^\star \leq$ multiplicity of $\lambda_2(L^\star)$
  - explains different capability of estimating dimensionality
- further connections to Markov chains and network design

MVU duality theory





- **primal problem** $\longleftrightarrow$ **dual problem**

$$\begin{aligned} \text{max.} \quad & \mathbf{Tr}\, K \\ \text{s. t.} \quad & K = K^T \succeq 0, \quad \mathbf{1}^T K \mathbf{1} = 0 \\ & K_{ii} + K_{jj} - 2K_{ij} = D_{ij}, \quad \{i,j\} \in \mathcal{E} \end{aligned}$$

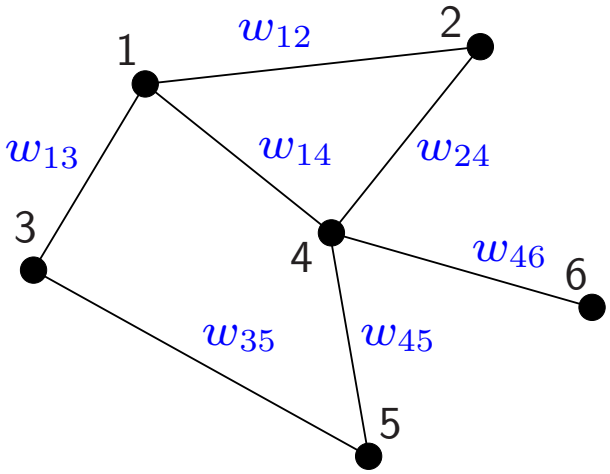
$$\begin{aligned} \text{min.} \quad & \sum_{\{i,j\} \in \mathcal{N}_i} D_{ij} W_{ij} \\ \text{s. t.} \quad & \lambda_2(L) \geq 1 \end{aligned}$$

- an equivalent dual problem (solutions related by simple scaling)

$$\begin{aligned} \text{maximize} \quad & \lambda_2(L) \\ \text{subject to} \quad & \sum_{\{i,j\} \in \mathcal{N}_i} D_{ij} W_{ij} \leq 1 \end{aligned}$$

Continuous-time Markov chains

- assign transition rate $w_{ij} \geq 0$ on each edge



- weighted Laplacian matrix L

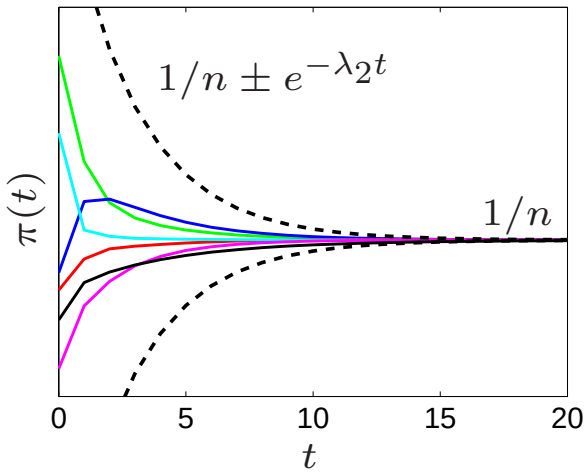
$$L_{ij} = \begin{cases} -w_{ij} & \{i,j\} \in \mathcal{E} \\ 0 & \{i,j\} \notin \mathcal{E} \\ \sum_k w_{ik} & i = j \end{cases}$$

eigenvalues: $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$

- probability distribution $\pi(t)$ satisfy

$$\frac{d\pi(t)}{dt} = -L\pi(t)$$

convergence: $\sup_{\pi(0)} \|\pi(t) - \mathbf{1}/n\| \leq c\, e^{-\lambda_2 t}$

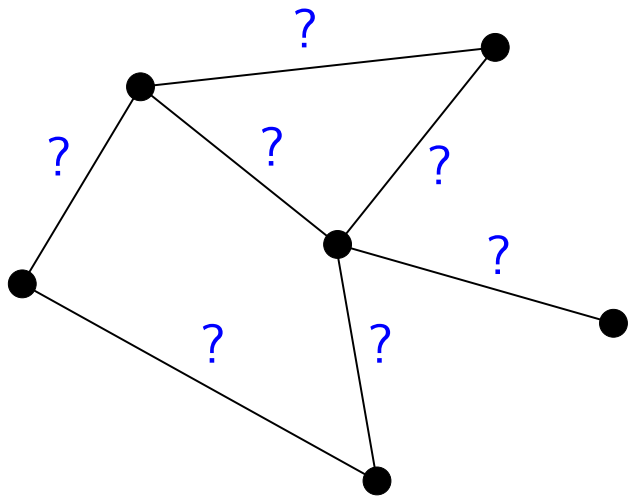


Continuous-time FMMC problem

maximize $\lambda_2(L)$

subject to $\sum_{\{i,j\}\in\mathcal{E}} d_{ij}^2\, w_{ij} \leq 1$

$$L_{ij} = \begin{cases} -w_{ij} & \{i,j\} \in \mathcal{E} \\ 0 & \{i,j\} \notin \mathcal{E} \\ \sum_k w_{ik} & i = j \end{cases}$$

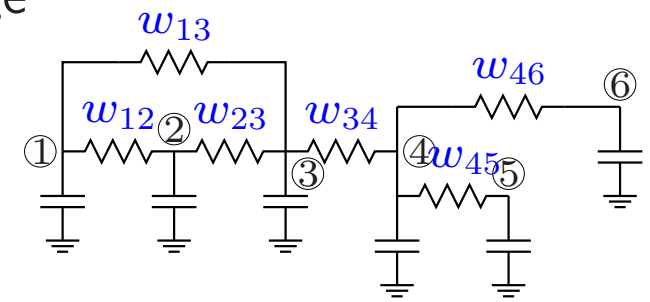


- objective function nonlinear, nondifferentiable, but concave
- total rate constraints, weighted by edge lengths d_{ij}^2
- discrete-time version: fastest random walk on a graph

More connections

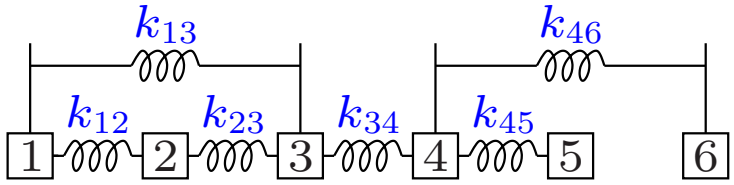
- fastest equilibration of electrical charge

$$\frac{dq(t)}{dt} = -Lq(t)$$



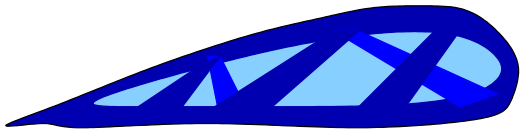
- maximize lowest natural frequency of mechanical systems

$$\frac{d^2x(t)}{dt^2} = -Lx(t)$$



- optimal design of inhomogeneities of physical medium

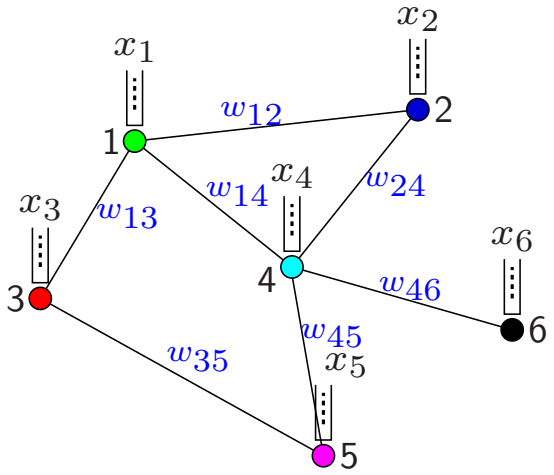
$$\nabla \cdot (k \nabla u) = 0$$



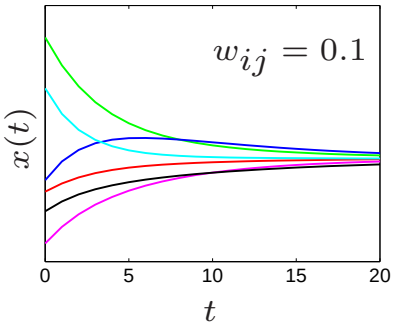
Connection to distributed computing

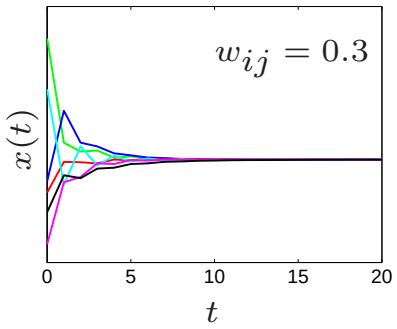
- **example: load balancing**
  - initial queue lengths $x_i(0)$
  - every server to process $\frac{1}{n}\sum x_i(0)$
  - distributed algorithm

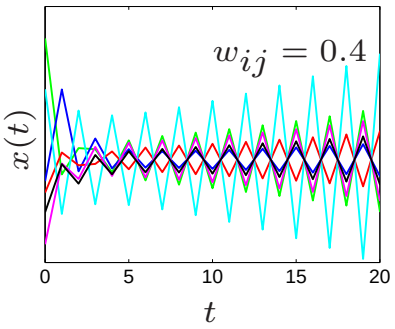
$$x_i(t+1) = x_i(t) + \sum_{j\in\mathcal{N}_i} w_{ij}\Big(x_j(t) - x_i(t)\Big)$$



- design **local parameters** w_{ij} to achieve **global performance**
  - guarantee convergence; obtain fastest convergence
  - robustness to unreliable links, noises, topology changes





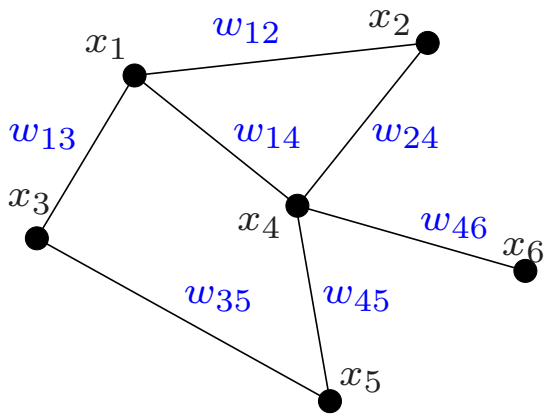


Connection to distributed computing

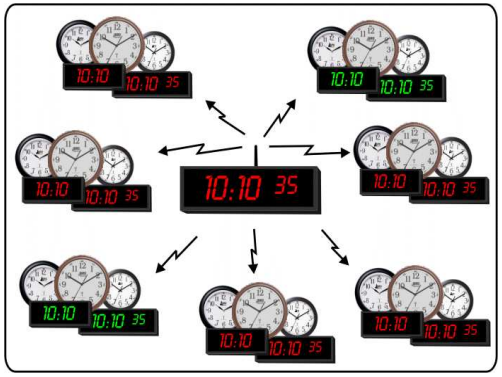
- distributed average consensus

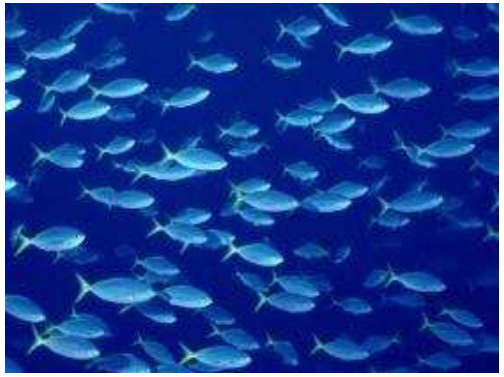
$$x_i(t+1) = x_i(t) + \sum_{j \in \mathcal{N}_i} w_{ij}\Big(x_j(t) - x_i(t)\Big)$$

  - convergence conditions, robustness
  - optimal design for fastest convergence



- distributed coordination, synchronization, and flocking

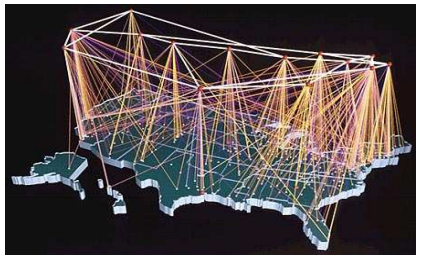




Tsitsiklis (1984), Jadbabaie, Lin & Morse (2003), Olfati-Saber & Murray (2004), Moreau (2005), Xiao, Boyd & Lall (2005), etc.

Duality as unifying tool

ubiquitous networks



NSFNET, Donna Cox and Robert Patterson

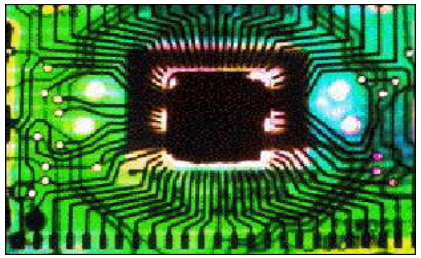
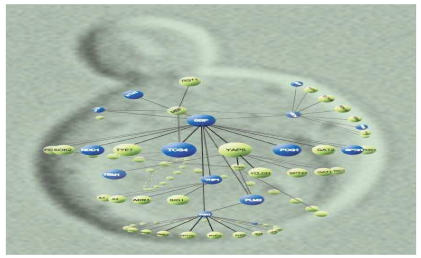
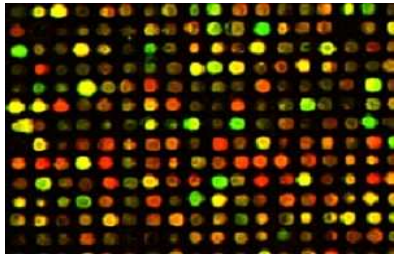
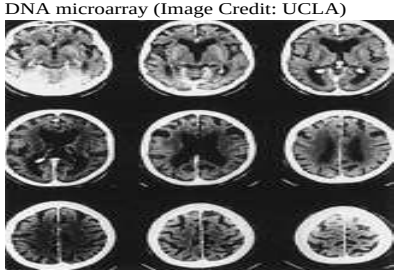


Image Credit: Lawrence Berkeley National Lab

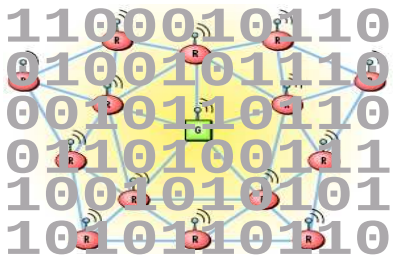




DNA microarray (Image Credit: UCLA)



CT Scan of brain (Credit: University of Iowa)



data, data, data!