CSE 291B 2/19

Nonlinear dimensionality reduction (review)

How to map inputs $\vec{x}_i \in \mathbb{R}^D$ to outputs $\vec{y}_i \in \mathbb{R}^d$ with $d \ll D$?

- Approach #1: probabilistic generative modeling
  - e.g. mixture of factor analyzers
  - strong parametric assumptions
  - latent (hidden) variables
  - parameter estimation by EM algorithm
  - potential local maxima in log-likelihood
- Approach #2: spectral methods
  - e.g. isomap, maximum variance unfolding
  - non-parametric, graph-based
  - low dimensional outputs from metric multidimensional scaling
  - tractable optimizations; no spurious local optima

Clustering

How to map inputs $\vec{x}_i \in \mathbb{R}^D$ to discrete outputs $y \in \{1, 2, \ldots, k\}$?

- Earlier in course: k-means (VQ), Gaussian mixture models
- Today: spectral clustering

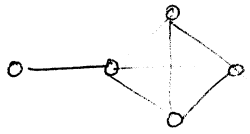
Same distinctions arise as in NLDR.
For simplicity, focus on $k=2$ binary partitioning.
(Finer clusters can be found by recursive subdivision.)

- Basic algorithm ($k=2$)

Step 1: graph construction

Define affinity matrix $W_{ij} = e^{-\frac{1}{2\sigma^2}\|\vec{x}_i - \vec{x}_j\|^2}$

Define diagonal matrix $D_{ii} = \sum_j W_{ij}$



Intuition:
- each input $\vec{x}_i$ is node in graph
- nearby inputs connected by larger weights W_{ij}
- row sums D_{ii} count "effective" # neighbors within radius σ
- "sparsify" graphs by approximating $W_{ij} = 0$ for $\|\vec{x}_i - \vec{x}_j\|^2 >> \sigma^2$

Step 2: constrained optimization over real-valued $\vec{y} \in \mathbb{R}^N$

Minimize $C(\vec{y}) = \frac{1}{2} \frac{\sum_{ij} W_{ij}(y_i - y_j)^2}{\sum_i D_{ii} y_i^2}$ subject to $\sum_i y_i D_{ii} = 0$

Numerator: favors $y_i \approx y_j$ for large W_{ij}

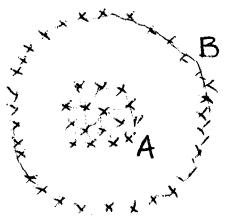
Constrairnt: prevents degenerate solution $y_i =$ constant for all i

Denominator: repels y_i from origin; more neighbors $\rightarrow$ greater repulsion

Step 3: quantization

Threshold y_i on (say) mean or median to obtain $k=2$ clusters

Example:



Suppose $W_{AB} << W_{AA}$

Then $y_i \approx \begin{cases} +1 & \text{for } i \in A \\ -1 & \text{for } i \in B \end{cases}$

will minimize $C(y)$.

But: how to minimize $C(y)$ in step #2?

Detour: generalized eigenvalue problem

Let A and B be square, symmetric matrices.
Also assume B is positive definite

- generalized Rayleigh quotient

$$Q(y) = \frac{y^T A y}{y^T B y}$$

To minimize:

$$\frac{\partial Q}{\partial y} = 0 \longrightarrow 2\left(\frac{Ay}{y^T B y}\right) - \frac{y^T A y}{(y^T B y)^2}(2By) = 0$$

$$Ay = \left(\frac{y^T A y}{y^T B y}\right) By$$

$$(B^{-1}A)y = Q(y)\, y$$

($Q(y)$: scalar)

this is an eigenvalue equation for $B^{-1}A$

Solutions of $\frac{\partial Q}{\partial y} = 0$ are eigenvectors of $B^{-1}A$.

$\min(Q(y))$ = smallest eigenvalue of $B^{-1}A$

$\operatorname{argmin} Q(y)$ = corresponding eigenvectors

- Symmetrized form: let $z = B^{\frac{1}{2}} y$

Then $Q(z) = \frac{(z^T B^{-\frac{1}{2}}) A (B^{-\frac{1}{2}} z)}{(z^T B^{-\frac{1}{2}}) B (B^{-\frac{1}{2}} z)} = \frac{z^T (B^{-\frac{1}{2}} A B^{-\frac{1}{2}}) z}{z^T z}$

To minimize: $\frac{\partial Q}{\partial z} = 0 \longrightarrow (B^{-\frac{1}{2}} A B^{-\frac{1}{2}}) z = Q(z) z$

Let z_α denote α^{th} eigenvector of $B^{-\frac{1}{2}} A B^{-\frac{1}{2}}$; ...

Then $z_\alpha^T z_\beta = 0$ for $\alpha \neq \beta$

Substituting $z = B^{\frac{1}{2}} y$ gives "generalized" orthogonality condition:

$y_\alpha^T B y_\beta = 0$ for $\alpha \neq \beta$ (different eigenvectors of $B^{-1}A$)

↑ not symmetric

Optimization for spectral clustering

- Cost function

$$C(y) = \frac{1}{2} \frac{\sum_{ij} W_{ij}(y_i - y_j)^2}{\sum_i D_{ii} y_i^2} = \frac{\frac{1}{2}\left[\sum_{ij} W_{ij}(y_i^2 + y_j^2) - 2\sum_{ij} W_{ij} y_i y_j\right]}{\sum_i D_{ii} y_i^2}$$

$$= \frac{y^T(D-W)y}{y^T D y}$$

generalized eigenvalue problem $A = D - W$, $B = D$

- Constraint

min $C(y)$ subject to $\sum_i y_i D_{ii} = 0$

$\sum_i y_i D_{ii} = 0 \Longrightarrow y^T D \mathbb{1} = 0$ where $\mathbb{1} = \underbrace{(1, 1, 1, \ldots, 1)}_{N \text{ ones}}^T$

- Generalized eigenvalue equation

$$\frac{\partial C}{\partial y} = 0 \longrightarrow (D - W)y = \lambda D y$$

- Unconstrained minimum of $C(y)$

Trivially: min $C(y) = 0$ by setting $y_i = y_j = 1$

From eigenvalue equation: $(D-W)\mathbb{1} = 0$ since $D_{ii} = \sum_j W_{ij}$

Hence: smallest eigenvalue $\lambda_0 = 0$ with eigenvector $y_0 = \mathbb{1}$

However: y_0 violates constraint b/c $\mathbb{1}^T D \mathbb{1} = \sum_i D_{ii} \neq 0$. ($y_0 = \mathbb{1}$)

- Constrained minimum of $C(y)$

Let y_1 denote eigenvector of $D^{-1}(D-W)$ with second smallest eigenvalue λ_1.

Also true that $\left.\frac{\partial C}{\partial y}\right|_{y_1} = 0$.

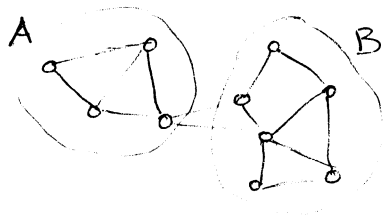
By orthogonality: $y_1^T D y_0 = 0 \longrightarrow y_1^T D \mathbb{1} = 0$ satisfies constraint.

In sum:

$$\underset{y^T D \mathbb{1} = 0}{\operatorname{argmin}} [C(y)] = y_1 \quad \text{with} \quad \min_{y^T D \mathbb{1} = 0} C(y) = \lambda_1.$$

This optimization has appeared in many different places...

Graph partitioning



$$\text{cut}(A,B) = \sum_{i \in A} \sum_{j \in B} W_{ij}$$

"cut" of weighted graph W_{ij} into partitions A and B

- Minimizing $\text{cut}(A,B)$ over partitions A and B can be done in poly-time. But results often yield highly unbalanced cuts.
- Normalized cut

$$\text{Ncut}(A,B) = \frac{\text{cut}(A,B)}{|A|} + \frac{\text{cut}(A,B)}{|B|}$$

where $|A| = \sum_{i \in A} D_{ii}$ and $|B| = \sum_{i \in B} D_{ii}$

Penalizes unbalanced cuts.

Favors $|A| \approx |B|$ with "equal mass" partitions.

However: minimizing $\text{Ncut}(A,B)$ is NP-complete.

- Eigenvalue relaxation

Let $y_i = \begin{cases} 1 & \text{if } i \in A \\ 0 & \text{if } i \in B \end{cases}$

$$\text{Ncut}(A,B) = \frac{\sum_{ij} W_{ij}\, y_i (1-y_j)}{\sum_i D_{ii}\, y_i} + \frac{\sum_{ij} W_{ij}\, y_i (1-y_j)}{\sum_i D_{ii} (1-y_i)}$$

Let $b \triangleq \dfrac{\sum_i D_{ii}(y_i)}{\sum_i D_{ii}(1-y_i)}$ ratio of $\dfrac{|A|}{|B|}$

$$z_i \triangleq y_i - b(1-y_i) \in \{1, -b\}$$

By construction: $z^T D \mathbb{1} = \sum_i D_{ii}[y_i - b(1-y_i)] = 0$.

It can be shown:

$$\min_{y \in \{0,1\}^n} Ncut(y) = \min_z \frac{z^T(D-W)z}{z^T D z} \text{ subject to } \begin{cases} z_i \in \{1, -b\} \\ z^T D \mathbb{1} = 0 \end{cases}$$

Spectral clustering relaxes $z_i \in \mathbb{R}$ to be real-valued, then thresholds to obtain a discrete solution.