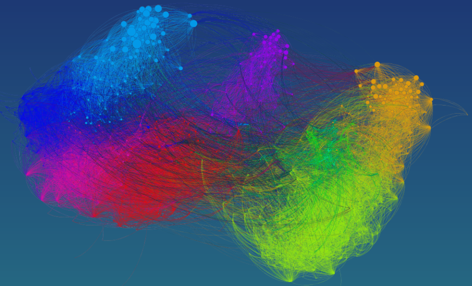
Graphs in Machine Learning

Michal Valko

Inria Lille - Nord Europe, France

TA: Daniele Calandriello

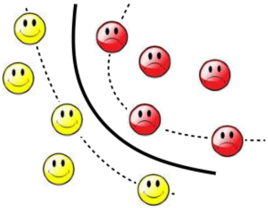
Partially based on material by: Ulrike von Luxburg, Gary Miller, Doyle & Schnell, Daniel Spielman



Previous Lecture

- similarity graphs
  - different types
  - construction
  - sources of graphs
  - practical considerations
- spectral graph theory
- Laplacians and their properties
  - symmetric and asymmetric normalization
- random walks
- recommendation on a bipartite graph
- resistive networks
  - recommendation score as a resistance?
  - Laplacian and resistive networks
  - resistance distance and random walks

Statistical Machine Learning in Paris!



https://sites.google.com/site/smileinparis/sessions-2016--17

Speaker: Isabelle Guyon - LRI (équipe TAO), UPSud

Topic: Network Reconstruction

Date: Monday, October 17, 2016

Time: 13:30 - 14:30 (this is pretty soon)

Place: Institut Henri Poincaré — salle 314

This Lecture

- geometry of the data and the connectivity
- spectral clustering
- manifold learning with Laplacians eigenmaps
- Gaussian random fields and harmonic solution
- graph-based semi-supervised learning and manifold regularization
- transductive learning
- inductive and transductive semi-supervised learning

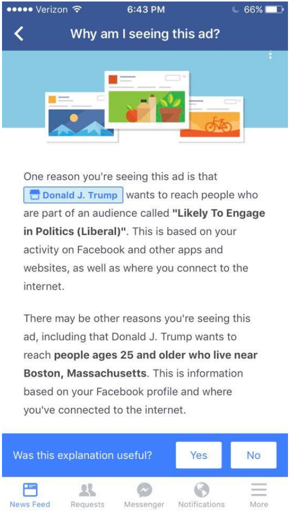
Next Class: Lab Session

- 24. 10. 2016 by Daniele Calandriello
- cca. 10h30-11h help with setup (optional), 11h-13: TD
- Salle Condorcet
- Download the image and set it up BEFORE the class
- Matlab/Octave
- Short written report (graded)
- All homeworks together account for 40% of the final grade
- Content
  - Graph Construction
  - Test sensitivity to parameters: σ, k, ε
  - Spectral Clustering
  - Spectral Clustering vs. k-means
  - Image Segmentation

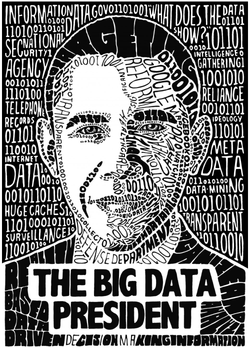
How to rule the world?

Let's make Sokovia great again!

How to rule the world?



How to rule the world: “AI” is here

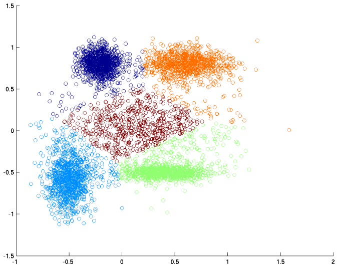


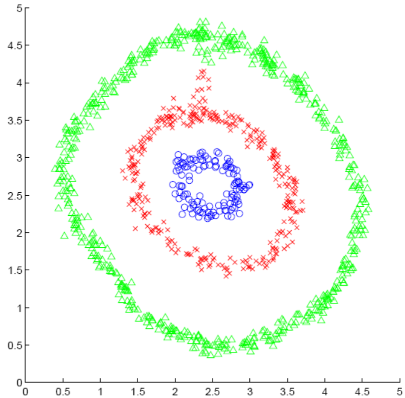
https://www.washingtonpost.com/opinions/obama-the-big-data-president/2013/06/14/1d71fe2e-d391-11e2-b05f-3ea3f0e7bb5a_story.html

https://www.technologyreview.com/s/509026/how-obamas-team-used-big-data-to-rally-voters/

Talk of Rayid Ghaniy: https://www.youtube.com/watch?v=gDM1GuszM_U

Application of Graphs for ML: Clustering

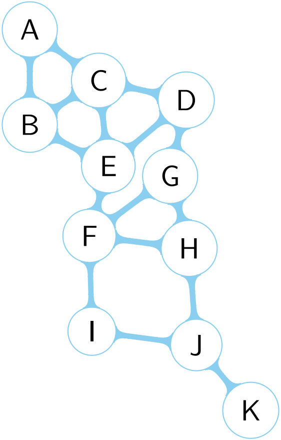




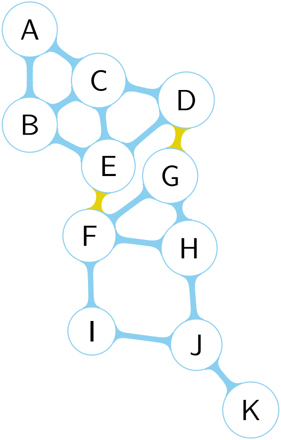
Application: Clustering - Recap

- What do we know about the **clustering** in general?
  - ill defined problem (different tasks → different paradigms)
  - "I know it when I see it"
  - inconsistent (wrt. Kleinberg's axioms)
  - number of clusters k need often be known
  - difficult to evaluate
- What do we know about **k-means**?
  - "hard" version of EM clustering
  - sensitive to initialization
  - optimizes for **compactness**
  - yet: algorithm-to-go

Spectral Clustering: Cuts on graphs

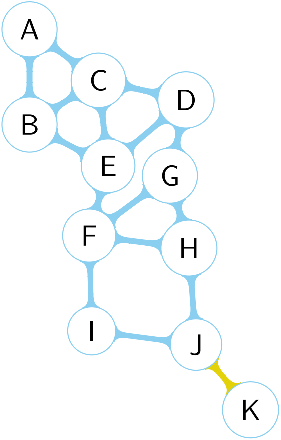


Spectral Clustering: Cuts on graphs



Defining the cut objective we get the clustering!

Spectral Clustering: Cuts on graphs



MinCut: $\mathrm{cut}(A, B) = \sum_{i \in A, j \in B} w_{ij}$

Are we done?

Can be solved efficiently, but maybe not what we want . . . .

Spectral Clustering: Balanced Cuts

Let's balance the cuts!

MinCut

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

RatioCut

$$\operatorname{RatioCut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \left(\frac{1}{|A|} + \frac{1}{|B|} \right)$$

Normalized Cut

$$\operatorname{NCut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \left(\frac{1}{\operatorname{vol}(A)} + \frac{1}{\operatorname{vol}(B)} \right)$$

Spectral Clustering: Balanced Cuts

$$\mathrm{RatioCut}(A, B) = \mathrm{cut}(A, B)\left(\frac{1}{|A|} + \frac{1}{|B|}\right)$$

$$\mathrm{NCut}(A, B) = \mathrm{cut}(A, B)\left(\frac{1}{\mathrm{vol}(A)} + \frac{1}{\mathrm{vol}(B)}\right)$$

Can we compute this? RatioCut and NCut are NP hard :(

Approximate!

Spectral Clustering: Relaxing Balanced Cuts

Relaxation for (simple) balanced cuts for 2 sets

$$\min_{A,B} \mathrm{cut}(A, B) \quad \text{s.t.} \quad |A| = |B|$$

Graph function $\mathbf{f}$ for cluster membership: $f_i = \begin{cases} 1 & \text{if } V_i \in A, \\ -1 & \text{if } V_i \in B. \end{cases}$

What it is the cut value with this definition?

$$\mathrm{cut}(A, B) = \sum_{i \in A, j \in B} w_{i,j} = \frac{1}{4} \sum_{i,j} w_{i,j}(f_i - f_j)^2 = \frac{1}{2}\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

What is the relationship with the smoothness of a graph function?

Spectral Clustering: Relaxing Balanced Cuts

$$\mathrm{cut}(A,B) = \sum_{i\in A, j\in B} w_{i,j} = \frac{1}{4}\sum_{i,j} w_{i,j}(f_i - f_j)^2 = \frac{1}{2}\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f}$$

$$|A| = |B| \implies \sum_i f_i = 0 \implies \mathbf{f} \perp \mathbf{1}_N$$

$$\|\mathbf{f}\| = \sqrt{N}$$

objective function of spectral clustering

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i = \pm 1, \quad \mathbf{f} \perp \mathbf{1}_N, \quad \|\mathbf{f}\| = \sqrt{N}$$

Still NP hard :($\rightarrow$ Relax even further!

~~$f_i = \pm 1$~~ $\rightarrow$ $f_i \in \mathbb{R}$

Spectral Clustering: Relaxing Balanced Cuts

objective function of spectral clustering

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{f} \perp \mathbf{1}_N, \quad \|\mathbf{f}\| = \sqrt{N}$$

Rayleigh-Ritz theorem

If $\lambda_1 \leq \cdots \leq \lambda_N$ are the eigenvectors of real symmetric $\mathbf{L}$ then

$$\lambda_1 = \min_{\mathbf{x} \neq 0} \frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}} = \min_{\mathbf{x}^\mathsf{T}\mathbf{x}=1} \mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}$$

$$\lambda_N = \max_{\mathbf{x} \neq 0} \frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}} = \max_{\mathbf{x}^\mathsf{T}\mathbf{x}=1} \mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}$$

$\frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}} \equiv$ **Rayleigh quotient**

How can we use it?

Spectral Clustering: Relaxing Balanced Cuts

objective function of spectral clustering

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{f} \perp \mathbf{1}_N, \quad \|\mathbf{f}\| = \sqrt{N}$$

Generalized Rayleigh-Ritz theorem (Courant-Fischer-Weyl)

If $\lambda_1 \leq \cdots \leq \lambda_N$ are the eigenvectors of real symmetric $\mathbf{L}$ and $\mathbf{v}_1, \ldots, \mathbf{v}_N$ the corresponding orthogonal eigenvalues, then for $k = 1 : N - 1$

$$\lambda_{k+1} = \min_{\mathbf{x} \neq 0, \mathbf{x} \perp \mathbf{v}_1, \ldots \mathbf{v}_k} \frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}} = \min_{\mathbf{x}^\mathsf{T}\mathbf{x}=1, \mathbf{x} \perp \mathbf{v}_1, \ldots \mathbf{v}_k} \mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}$$

$$\lambda_{N-k} = \max_{\mathbf{x} \neq 0, \mathbf{x} \perp \mathbf{v}_n, \ldots \mathbf{v}_{N-k+1}} \frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}} = \max_{\mathbf{x}^\mathsf{T}\mathbf{x}=1, \mathbf{x} \perp \mathbf{v}_N, \ldots \mathbf{v}_{N-k+1}} \mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}$$

Rayleigh-Ritz theorem: Quick and dirty proof

When we reach the extreme points?

$$\frac{\partial}{\partial \mathbf{x}}\left(\frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}}\right) = \frac{\partial}{\partial \mathbf{x}}\left(\frac{f(\mathbf{x})}{g(\mathbf{x})}\right) = 0 \iff f'(\mathbf{x})g(\mathbf{x}) = f(\mathbf{x})g'(\mathbf{x})$$

By matrix calculus (or just calculus):

$$\frac{\partial \mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{L}\mathbf{x} \quad \text{and} \quad \frac{\partial \mathbf{x}^\mathsf{T}\mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{x}$$

When $f'(\mathbf{x})g(\mathbf{x}) = f(\mathbf{x})g'(\mathbf{x})$?

$$\mathbf{L}\mathbf{x}\left(\mathbf{x}^\mathsf{T}\mathbf{x}\right) = \left(\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}\right)\mathbf{x} \iff \mathbf{L}\mathbf{x} = \frac{\mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}}\mathbf{x} \iff \mathbf{L}\mathbf{x} = \lambda\mathbf{x}$$

Conclusion: Extremes are the eigenvectors with their eigenvalues

Spectral Clustering: Relaxing Balanced Cuts

objective function of spectral clustering

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{f} \perp \mathbf{1}_N, \quad \|\mathbf{f}\| = \sqrt{N}$$

Solution: **second eigenvector** How do we get the clustering?

The solution may not be integral. What to do?

$$\text{cluster}_i = \begin{cases} 1 & \text{if } f_i \geq 0, \\ -1 & \text{if } f_i < 0. \end{cases}$$

Works but this heuristics is often too simple. In practice, cluster $\mathbf{f}$ using k-means to get $\{C_i\}_i$ and assign:

$$\text{cluster}_i = \begin{cases} 1 & \text{if } i \in C_1, \\ -1 & \text{if } i \in C_{-1}. \end{cases}$$

Spectral Clustering: Approximating RatioCut

Wait, but we did not care about approximating mincut!

RatioCut

$$\mathrm{RatioCut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \left(\frac{1}{|A|} + \frac{1}{|B|} \right)$$

Define graph function **f** for cluster membership of RatioCut:

$$f_i = \begin{cases} \sqrt{\frac{|B|}{|A|}} & \text{if } V_i \in A, \\ -\sqrt{\frac{|A|}{|B|}} & \text{if } V_i \in B. \end{cases}$$

$$\mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} = \frac{1}{2} \sum_{i,j} w_{i,j} (f_i - f_j)^2 = (|A| + |B|) \mathrm{RatioCut}(A, B)$$

Spectral Clustering: Approximating RatioCut

Define graph function $\mathbf{f}$ for cluster membership of RatioCut:

$$f_i = \begin{cases} \sqrt{\frac{|B|}{|A|}} & \text{if } V_i \in A, \\ -\sqrt{\frac{|A|}{|B|}} & \text{if } V_i \in B. \end{cases}$$

$$\sum_i f_i = 0$$

$$\sum_i f_i^2 = N$$

objective function of spectral clustering (same - it's magic!)

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{f} \perp \mathbf{1}_N, \quad \|\mathbf{f}\| = \sqrt{N}$$

Spectral Clustering: Approximating NCut

Normalized Cut

$$\mathrm{NCut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \left(\frac{1}{\mathrm{vol}(A)} + \frac{1}{\mathrm{vol}(B)} \right)$$

Define graph function **f** for cluster membership of NCut:

$$f_i = \begin{cases} \sqrt{\frac{\mathrm{vol}(A)}{\mathrm{vol}(B)}} & \text{if } V_i \in A, \\ -\sqrt{\frac{\mathrm{vol}(B)}{\mathrm{vol}(A)}} & \text{if } V_i \in B. \end{cases}$$

$$(\mathbf{Df})^\mathsf{T}\mathbf{1}_n = 0 \qquad \mathbf{f}^\mathsf{T}\mathbf{Df} = \mathrm{vol}(\mathcal{V}) \qquad \mathbf{f}^\mathsf{T}\mathbf{Lf} = \mathrm{vol}(\mathcal{V})\mathrm{NCut}(A, B)$$

objective function of spectral clustering (NCut)

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{Lf} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{Df} \perp \mathbf{1}_N, \quad \mathbf{f}^\mathsf{T}\mathbf{Df} = \mathrm{vol}(\mathcal{V})$$

Spectral Clustering: Approximating NCut

objective function of spectral clustering (NCut)

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{D}\mathbf{f} \perp \mathbf{1}_N, \quad \mathbf{f}^\mathsf{T}\mathbf{D}\mathbf{f} = \mathrm{vol}(\mathcal{V})$$

Can we apply Rayleigh-Ritz now? Define $\mathbf{w} = \mathbf{D}^{1/2}\mathbf{f}$

objective function of spectral clustering (NCut)

$$\min_{\mathbf{w}} \mathbf{w}^\mathsf{T}\mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}\mathbf{w} \quad \text{s.t.} \quad w_i \in \mathbb{R}, \mathbf{w} \perp \mathbf{D}^{1/2}\mathbf{1}_N, \|\mathbf{w}\|^2 = \mathrm{vol}(\mathcal{V})$$

objective function of spectral clustering (NCut)

$$\min_{\mathbf{w}} \mathbf{w}^\mathsf{T}\mathbf{L}_{\mathrm{sym}}\mathbf{w} \quad \text{s.t.} \quad w_i \in \mathbb{R}, \quad \mathbf{w} \perp \mathbf{v}_{1,\mathbf{L}_{\mathrm{sym}}}, \quad \|\mathbf{w}\|^2 = \mathrm{vol}(\mathcal{V})$$

Spectral Clustering: Approximating NCut

objective function of spectral clustering (NCut)

$$\min_{\mathbf{w}} \mathbf{w}^{\mathsf{T}} \mathbf{L}_{\mathrm{sym}} \mathbf{w} \quad \text{s.t.} \quad w_i \in \mathbb{R}, \quad \mathbf{w} \perp \mathbf{v}_{1,\mathbf{L}_{\mathrm{sym}}}, \quad \|\mathbf{w}\| = \mathrm{vol}(\mathcal{V})$$

Solution by Rayleigh-Ritz? $\mathbf{w} = \mathbf{v}_{2,\mathbf{L}_{\mathrm{sym}}}$ $\mathbf{f} = \mathbf{D}^{-1/2}\mathbf{w}$

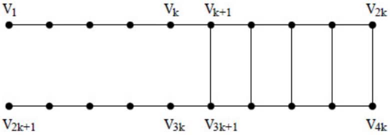
$\mathbf{f}$ is a the second eigenvector of $\mathbf{L}_{\mathrm{rw}}$!

tl;dr: Get the second eigenvector of $\mathbf{L}/\mathbf{L}_{\mathrm{rw}}$ for RatioCut/NCut.

Spectral Clustering: Approximation

These are all approximations. How bad can they be?

Example: cockroach graphs

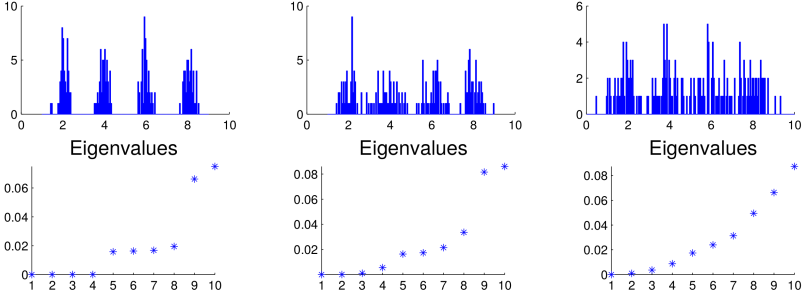


No efficient approximation exist. Other relaxations possible.

https://www.cs.cmu.edu/~glmiller/Publications/Papers/GuMi95.pdf

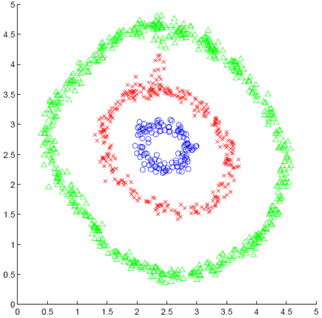
Spectral Clustering: 1D Example

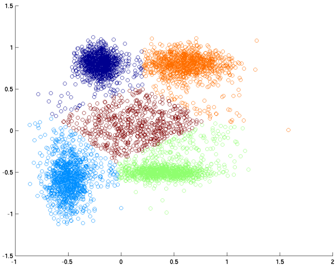
Elbow rule/EigenGap heuristic for number of clusters



Spectral Clustering: Understanding

Compactness vs. **Connectivity**

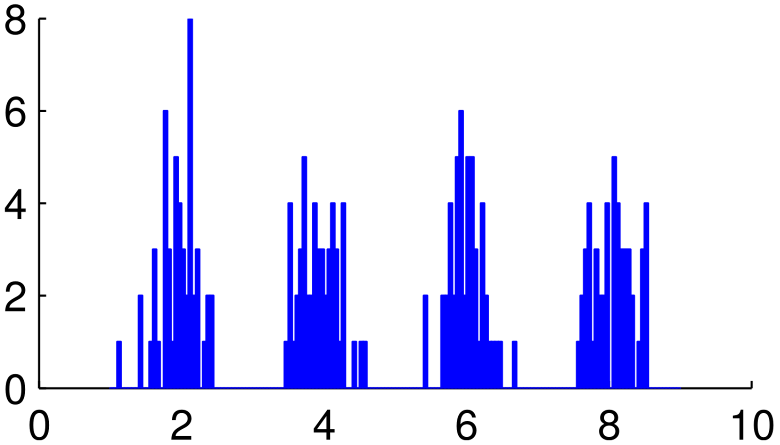




For which kind of data we can use one vs. the other?

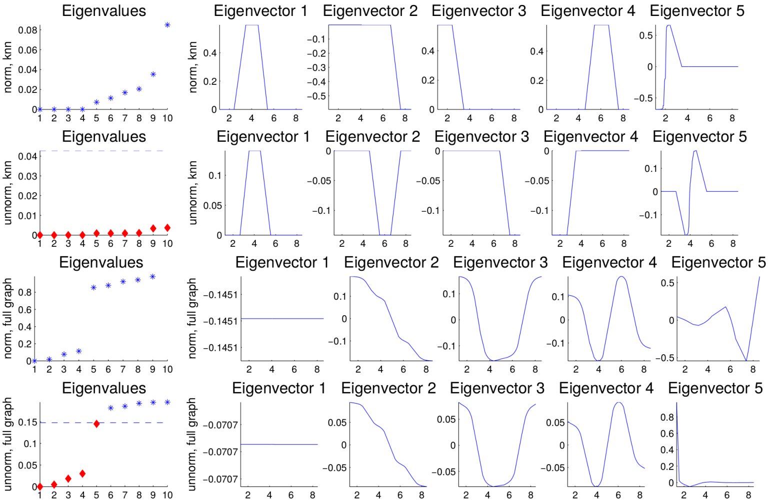
Any disadvantages of spectral clustering?

Spectral Clustering: 1D Example - Histogram



http://www.informatik.uni-hamburg.de/ML/contents/people/luxburg/publications/Luxburg07_tutorial.pdf

Spectral Clustering: 1D Example - Eigenvectors



Spectral Clustering: Bibliography

- M. Meila et al. "A random walks view of spectral segmentation". In: *International Conference on Artificial Intelligence and Statistics* (2001)
- $\mathbf{L}_{\mathrm{sym}}$ Andrew Y Ng, Michael I Jordan, and Yair Weiss. "On spectral clustering: Analysis and an algorithm". In: *Neural Information Processing Systems*. 2001
- $\mathbf{L}_{\mathrm{rm}}$ J Shi and J Malik. "Normalized Cuts and Image Segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000), pp. 888–905
- Things can go wrong with the relaxation: Daniel A. Spielman and Shang H. Teng. "Spectral partitioning works: Planar graphs and finite element meshes". In: *Linear Algebra and Its Applications* 421 (2007), pp. 284–305

Manifold Learning: Recap

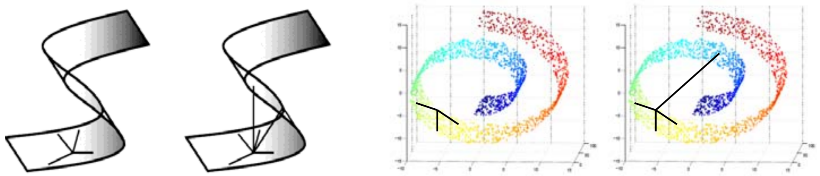
problem: definition reduction/manifold learning

Given $\{\mathbf{x}_i\}_{i=1}^N$ from $\mathbb{R}^d$ find $\{\mathbf{y}_i\}_{i=1}^N$ in $\mathbb{R}^m$, where $m \ll d$.

- What do we know about the **dimensionality reduction**
  - representation/visualization (2D or 3D)
  - an old example: globe to a map
  - often assuming $\mathcal{M} \subset \mathbb{R}^d$
  - feature extraction
  - linear vs. nonlinear dimensionality reduction
- What do we know about linear vs. nonlinear methods?
  - linear: ICA, PCA, SVD, ...
  - nonlinear often preserve only **local** distances

Manifold Learning: Linear vs. Non-linear

Manifold Learning: Preserving (just) local distances



$$d(\mathbf{y}_i, \mathbf{y}_j) = d(\mathbf{x}_i, \mathbf{x}_j) \quad \text{only if} \quad d(\mathbf{x}_i, \mathbf{x}_j) \quad \text{is small}$$

$$\min \sum_{ij} w_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2$$

Looks familiar?

Manifold Learning: Laplacian Eigenmaps

Step 1: Solve generalized eigenproblem:

$$\mathbf{Lf} = \lambda \mathbf{Df}$$

Step 2: Assign m new coordinates:

$$\mathbf{x}_i \mapsto \left(f_2\left(i\right), \ldots, f_{m+1}\left(i\right)\right)$$

Note$_1$: we need to get $m+1$ smallest eigenvectors
Note$_2$: $\mathbf{f}_1$ is useless

`http://web.cse.ohio-state.edu/~mbelkin/papers/LEM_NC_03.pdf`

Manifold Learning: Laplacian Eigenmaps to 1D

Laplacian Eigenmaps 1D objective

$$\min_{\mathbf{f}} \mathbf{f}^\mathsf{T}\mathbf{L}\mathbf{f} \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad \mathbf{f}^\mathsf{T}\mathbf{D}\mathbf{1} = 0, \quad \mathbf{f}^\mathsf{T}\mathbf{D}\mathbf{f} = \mathbf{1}$$

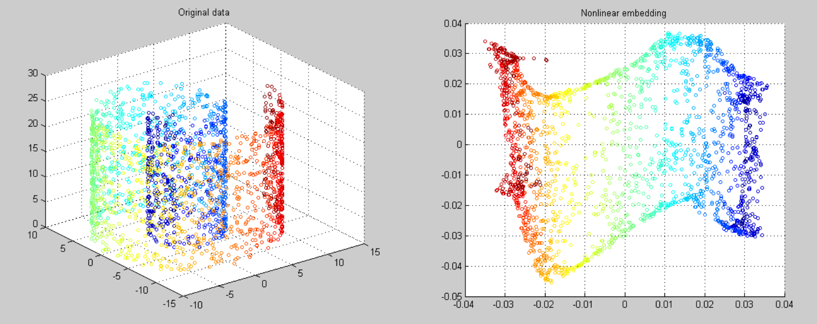
The meaning of the constraints is similar as for spectral clustering:

$\mathbf{f}^\mathsf{T}\mathbf{D}\mathbf{f} = \mathbf{1}$ is for scaling

$\mathbf{f}^\mathsf{T}\mathbf{D}\mathbf{1} = 0$ is to not get $\mathbf{v}_1$

What is the solution?

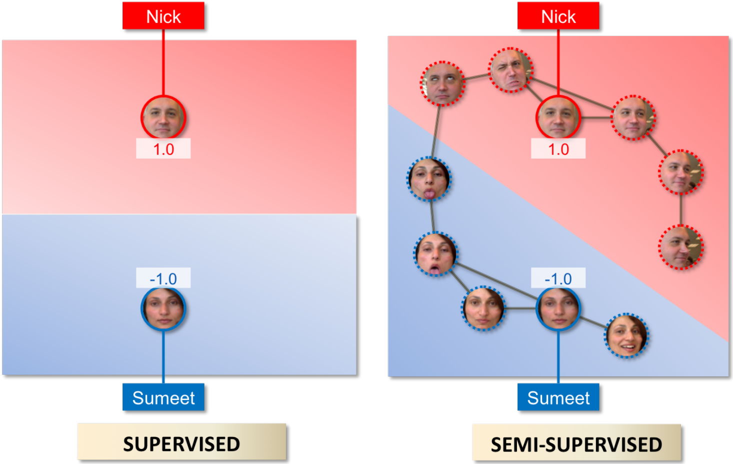
Manifold Learning: Example



`http://www.mathworks.com/matlabcentral/fileexchange/`

`36141-laplacian-eigenmap-~-diffusion-map-~-manifold-learning`

Semi-supervised learning: How is it possible?



This is how children learn! hypothesis

Semi-supervised learning (SSL)

SSL problem: definition

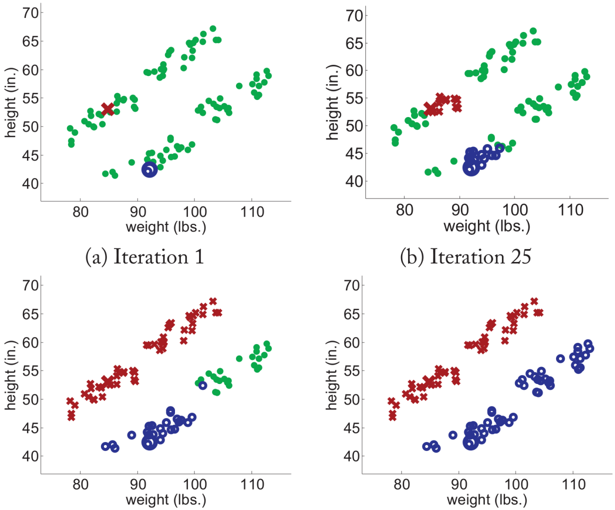
Given $\{\mathbf{x}_i\}_{i=1}^{N}$ from $\mathbb{R}^d$ and $\{y_i\}_{i=1}^{n_l}$, with $n_l \ll N$, find $\{y_i\}_{i=n_l+1}^{n}$ (**transductive**) or find f predicting y well beyond that (**inductive**).

Some facts about SSL

- assumes that the unlabeled data is useful
- works with data geometry assumptions
  - cluster assumption — low-density separation
  - manifold assumption
  - smoothness assumptions, generative models, ...
- now it helps now, now it does not (sic)
  - provable cases when it helps
- inductive or transductive/out-of-sample extension

http://olivier.chapelle.cc/ssl-book/discussion.pdf

SSL: Self-Training



(a) Iteration 1 (b) Iteration 25

(c) Iteration 74 (d) Final labeling of all instances

SSL: Overview: Self-Training

SSL: Self-Training

Input: $\mathcal{L} = \{\mathbf{x}_i, y_i\}_{i=1}^{n_l}$ and $\mathcal{U} = \{\mathbf{x}_i\}_{i=n_l+1}^{N}$

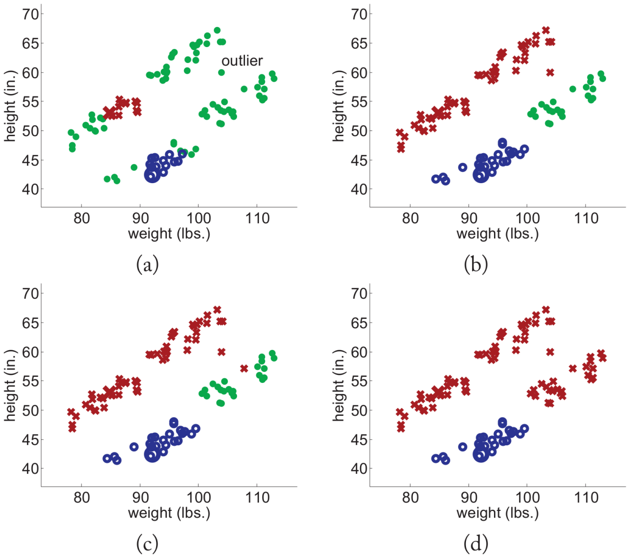
Repeat:

- train f using $\mathcal{L}$
- apply f to (some) $\mathcal{U}$ and add them to $\mathcal{L}$

What are the properties of self-training?

- its a wrapper method
- heavily depends on the the internal classifier
- some theory exist for specific classifiers
- nobody uses it anymore
- errors propagate (unless the clusters are well separated)

SSL: Self-Training: Bad Case



Michal Valko

michal.valko@inria.fr

ENS Paris-Saclay, MVA 2016/2017

SequeL team, Inria Lille — Nord Europe

https://team.inria.fr/sequel/