Graphs in Machine Learning

Michal Valko

DeepMind Paris and Inria Lille

TA: Omar Darwiche Domingues with the help of Pierre Perrault

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

- where do the graphs come from?
  - social, information, utility, and biological networks
  - we create them from the flat data
  - random graph models

- specific applications and concepts
  - maximizing influence on a graph gossip propagation, submodularity, proof of the approximation guarantee
  - Google pagerank random surfer process, steady state vector, sparsity
  - online semi-supervised learning label propagation, backbone graph, online learning, combinatorial sparsification, stability analysis
  - Erdős number project, real-world graphs, heavy tails, small world – when did this happen?

- similarity graphs
  - different types
  - construction
  - practical considerations
This Lecture

- spectral graph theory
- Laplacians and their properties
  - symmetric and asymmetric normalization
  - random walks
- geometry of the data and the connectivity
- spectral clustering
- manifold learning with Laplacians eigenmaps
- recommendation on a bipartite graph
- resistive networks
  - recommendation score as a resistance?
  - Laplacian and resistive networks
  - resistance distance and random walks
- PS: some students have started working on their projects already
Next Class: Lab Session

- 15.10.2019 by Omar (and Pierre)
- Short written report (graded)
- All homeworks together account for 40% of the final grade

Content
- Graph Construction
- Test sensitivity to parameters: $\sigma, k, \varepsilon$
- Spectral Clustering
- Spectral Clustering vs. $k$-means
- Image Segmentation
**Similarity Graphs: $\varepsilon$ or $k$-NN?**

http://www.ml.uni-saarland.de/code/GraphDemo/DemoSpectralClustering.htm
http://www.tml.cs.uni-tuebingen.de/team/luxburg/publications/Luxburg07_tutorial.pdf
Generic Similarity Functions

Gaussian similarity function/Heat function/RBF:

\[ s_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \]

Cosine similarity function:

\[ s_{ij} = \cos(\theta) = \left(\frac{x_i^T x_j}{\|x_i\| \|x_j\|}\right) \]

Typical Kernels
Similarity Graphs

\[ G = (\mathcal{V}, \mathcal{E}) \] - with a set of nodes \( \mathcal{V} \) and a set of edges \( \mathcal{E} \)
Sources of Real Networks

- http://snap.stanford.edu/data/
- http://www-personal.umich.edu/~mejn/netdata/
- http://proj.ise.bgu.ac.il/sns/datasets.html
- http://www.cise.ufl.edu/research/sparse/matrices/
- http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm
\[ L = D - W \]

graph Laplacian

...the only matrix that matters
Graph Laplacian

\[ G = (\mathcal{V}, \mathcal{E}) \] - with a set of **nodes** \( \mathcal{V} \) and a set of **edges** \( \mathcal{E} \)

<table>
<thead>
<tr>
<th></th>
<th>A adjacency matrix</th>
<th>W weight matrix</th>
<th>D (diagonal) degree matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>( \mathbf{L} = \mathbf{D} - \mathbf{W} )</td>
<td>graph Laplacian matrix</td>
<td></td>
</tr>
</tbody>
</table>

\[
\mathbf{L} = \begin{pmatrix}
4 & -1 & 0 & -1 & -2 \\
-1 & 8 & -3 & -4 & 0 \\
0 & -3 & 5 & -2 & 0 \\
-1 & -4 & -2 & 12 & -5 \\
-2 & 0 & 0 & -5 & 7 \\
\end{pmatrix}
\]

\( \mathbf{L} \) is SDD!

demo: https://dominiks Schmidt. xyz/spectral-clustering-exp/
Properties of Graph Laplacian

**Graph function**: a vector \( f \in \mathbb{R}^N \) assigning values to nodes:

\[
f : \mathcal{V}(G) \rightarrow \mathbb{R}.
\]

\[
f^T Lf = \frac{1}{2} \sum_{i,j \leq N} w_{i,j} (f_i - f_j)^2 = S_G(f)
\]
Recap: Eigenwerte und Eigenvektoren

A vector \( \mathbf{v} \) is an eigenvector of matrix \( \mathbf{M} \) of eigenvalue \( \lambda \)

\[
\mathbf{M} \mathbf{v} = \lambda \mathbf{v}.
\]

If \((\lambda_1, \mathbf{v}_1)\) are \((\lambda_2, \mathbf{v}_2)\) eigenpairs for symmetric \( \mathbf{M} \) with \( \lambda_1 \neq \lambda_2 \) then \( \mathbf{v}_1 \perp \mathbf{v}_2 \), i.e., \( \mathbf{v}_1^\top \mathbf{v}_2 = 0 \).

If \((\lambda, \mathbf{v}_1), (\lambda, \mathbf{v}_2)\) are eigenpairs for \( \mathbf{M} \) then \((\lambda, \mathbf{v}_1 + \mathbf{v}_2)\) is as well.

For symmetric \( \mathbf{M} \), the multiplicity of \( \lambda \) is the dimension of the space of eigenvectors corresponding to \( \lambda \).

\( N \times N \) symmetric matrix has \( N \) eigenvalues (w/ multiplicities).
Eigenvalues, Eigenvectors, and Eigendecomposition

A vector $\mathbf{v}$ is an eigenvector of matrix $\mathbf{M}$ of eigenvalue $\lambda$

$$\mathbf{Mv} = \lambda \mathbf{v}.$$ 

Vectors $\{\mathbf{v}_i\}_i$ form an orthonormal basis with $\lambda_1 \leq \lambda_2 \leq \ldots \lambda_N$.

$$\forall i \quad \mathbf{Mv}_i = \lambda_i \mathbf{v}_i \quad \equiv \quad \mathbf{MQ} = \mathbf{Q}\Lambda$$

$\mathbf{Q}$ has eigenvectors in columns and $\Lambda$ has eigenvalues on its diagonal.

Right-multiplying $\mathbf{MQ} = \mathbf{Q}\Lambda$ by $\mathbf{Q}^T$ we get the eigendecomposition of $\mathbf{M}$:

$$\mathbf{M} = \mathbf{MQQ}^T = \mathbf{Q}\Lambda\mathbf{Q}^T = \sum_i \lambda_i \mathbf{v}_i \mathbf{v}_i^T$$
We can assume **non-negative weights**: \( w_{ij} \geq 0 \).

\( L \) is symmetric

\( L \) positive semi-definite \( \leftarrow f^T L f = \frac{1}{2} \sum_{i,j \leq N} w_{i,j} (f_i - f_j)^2 \)

Recall: If \( L f = \lambda f \) then \( \lambda \) is an **eigenvalue** (of the Laplacian).

The smallest eigenvalue of \( L \) is 0. Corresponding eigenvector: \( \mathbf{1}_N \).

All eigenvalues are non-negative reals \( 0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N \).

Self-edges do not change the value of \( L \).
Properties of Graph Laplacian

The multiplicity of eigenvalue 0 of $L$ equals to the number of connected components. The eigenspace of 0 is spanned by the components’ indicators.

Proof: If $(0, f)$ is an eigenpair then $0 = \frac{1}{2} \sum_{i,j \leq N} w_{i,j} (f_i - f_j)^2$. Therefore, $f$ is constant on each connected component. If there are $k$ components, then $L$ is $k$-block-diagonal:

$$L = \begin{bmatrix}
L_1 & & \\
& L_2 & \\
& & \ddots \\
& & & L_k
\end{bmatrix}$$

For block-diagonal matrices: the spectrum is the union of the spectra of $L_i$ (eigenvectors of $L_i$ padded with zeros elsewhere).

For $L_i (0, 1_{|V_i|})$ is an eigenpair, hence the claim.
Smoothness of the Function and Laplacian

- \( f = (f_1, \ldots, f_N)^T \): graph function
- Let \( L = Q\Lambda Q^T \) be the eigendecomposition of the Laplacian.
  - Diagonal matrix \( \Lambda \) whose diagonal entries are eigenvalues of \( L \).
  - Columns of \( Q \) are eigenvectors of \( L \).
  - Columns of \( Q \) form a basis.
- \( \alpha \): Unique vector such that \( Q\alpha = f \)
  - Note: \( Q^Tf = \alpha \)

Smoothness of a graph function \( S_G(f) \)

\[
S_G(f) = f^TLf = f^TQ\Lambda Q^Tf = \alpha^T\Lambda\alpha = \|\alpha\|_\Lambda^2 = \sum_{i=1}^{N} \lambda_i \alpha_i^2
\]

Smoothness and regularization: Small value of

(a) \( S_G(f) \)  (b) \( \Lambda \) norm of \( \alpha^* \)  (c) \( \alpha_i^* \) for large \( \lambda_i \)
Smoothness of the Function and Laplacian

\[ S_G(f) = f^T L f = f^T Q \Lambda Q^T f = \alpha^T \Lambda \alpha = \|\alpha\|_\Lambda^2 = \sum_{i=1}^{N} \lambda_i \alpha_i^2 \]

Eigenvectors are graph functions too!

**What is the smoothness of an eigenvector?**

Spectral coordinates of eigenvector \( v_k \): \( Q^T v_k = e_k \)

\[ S_G(v_k) = v_k^T L v_k = v_k^T Q \Lambda Q^T v_k = e_k^T \Lambda e_k = \|e_k\|_\Lambda^2 = \sum_{i=1}^{N} \lambda_i (e_k)_i^2 = \lambda_k \]

The smoothness of \( k \)-th eigenvector is the \( k \)-th eigenvalue.
Laplacian of the Complete Graph $K_N$

What is the eigenspectrum of $L_{K_N}$?

From before: we know that $(0, 1_N)$ is an eigenpair.

If $v \neq 0_N$ and $v \perp 1_N \implies \sum_i v_i = 0$. To get the other eigenvalues, we compute $(L_{K_N}v)_1$ and divide by $v_1$ (wlog $v_1 \neq 0$).

$$(L_{K_N}v)_1 = (N - 1)v_1 - \sum_{i=2}^{N} v_i = Nv_1.$$

What are the remaining eigenvalues/vectors?
Normalized Laplacians

\[
L_{un} = D - W
\]
\[
L_{sym} = D^{-1/2} LD^{-1/2} = I - D^{-1/2} WD^{-1/2}
\]
\[
L_{rw} = D^{-1} L = I - D^{-1} W
\]

\[
f^T L_{sym} f = \frac{1}{2} \sum_{i,j \leq N} w_{i,j} \left( \frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2
\]

\((\lambda, u)\) is an eigenpair for \(L_{rw}\) iff \((\lambda, D^{1/2} u)\) is an eigenpair for \(L_{sym}\)
**Normalized Laplacians**

**\( L_{\text{sym}} \) and \( L_{\text{rw}} \) are PSD with non-negative real eigenvalues**

\[
0 = \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \cdots \leq \lambda_N
\]

(\( \lambda, u \)) is an eigenpair for \( L_{\text{rw}} \) iff (\( \lambda, u \)) solve the generalized eigenproblem \( Lu = \lambda Du \).

(\( 0, 1_N \)) is an eigenpair for \( L_{\text{rw}} \).

(\( 0, D^{1/2}1_N \)) is an eigenpair for \( L_{\text{sym}} \).

Multiplicity of eigenvalue 0 of \( L_{\text{rw}} \) or \( L_{\text{sym}} \) equals to the number of connected components.

Proof: As for \( L \).
Laplacian and Random Walks on Undirected Graphs

- stochastic process: vertex-to-vertex jumping
- transition probability $v_i \rightarrow v_j$ is $p_{ij} = w_{ij}/d_i$
  - $d_i \overset{\text{def}}{=} \sum_j w_{ij}$
- transition matrix $P = (p_{ij})_{ij} = D^{-1}W$ (notice $L_{rw} = I - P$)
- if $G$ is connected and non-bipartite $\rightarrow$ unique stationary distribution $\pi = (\pi_1, \pi_2, \pi_3, \ldots, \pi_N)$ where $\pi_i = d_i/\text{vol}(V)$
  - $\text{vol}(G) = \text{vol}(V) = \text{vol}(W) \overset{\text{def}}{=} \sum_i d_i = \sum_{i,j} w_{ij}$
- $\pi = \frac{1^T W}{\text{vol}(W)}$ verifies $\pi P = \pi$ as:

$$\pi P = \frac{1^T WP}{\text{vol}(W)} = \frac{1^T DP}{\text{vol}(W)} = \frac{1^T DD^{-1}W}{\text{vol}(W)} = \frac{1^T W}{\text{vol}(W)} = \pi$$

What’s the difference from the PageRank™?
cut\((A, B)\) = \frac{1}{2} f^T L f

spectral clustering
...with connectivity beyond compactness
How to rule the world?

Let’s make France great again!
How to rule the world?

One reason you're seeing this ad is that Donald J. Trump wants to reach people who are part of an audience called "Likely To Engage in Politics (Liberal)". This is based on your activity on Facebook and other apps and websites, as well as where you connect to the internet.

There may be other reasons you're seeing this ad, including that Donald J. Trump wants to reach people ages 25 and older who live near Boston, Massachusetts. This is information based on your Facebook profile and where you've connected to the internet.
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


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

\[ \text{MinCut: } \text{cut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \]

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

Are we done?
### Spectral Clustering: Balanced Cuts

Let’s balance the cuts!

| MinCut          | \[
| \text{cut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \]

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

| Normalized Cut  | \[
| \text{NCut}(A, B) = \sum_{i \in A, j \in B} w_{ij} \left( \frac{1}{\text{vol}(A)} + \frac{1}{\text{vol}(B)} \right) \]
Spectral Clustering: Balanced Cuts

\[
\text{RatioCut}(A, B) = \text{cut}(A, B) \left( \frac{1}{|A|} + \frac{1}{|B|} \right)
\]

\[
\text{Ncut}(A, B) = \text{cut}(A, B) \left( \frac{1}{\text{vol}(A)} + \frac{1}{\text{vol}(B)} \right)
\]

Easily generalizable to \( k \geq 2 \)

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

Approximate!
**Spectral Clustering: Relaxing Balanced Cuts**

Relaxation for (simple) balanced cuts for 2 sets

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

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

What is the cut value with this definition?

\[
\text{cut}(A, B) = \sum_{i \in A, j \in B} w_{i,j} = \frac{1}{4} \sum_{i,j} w_{i,j} (f_i - f_j)^2 = \frac{1}{2} f^T L f
\]

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

---

Michal Valko – Graphs in Machine Learning
DeepMind - 31/48
Spectral Clustering: Relaxing Balanced Cuts

\[ \text{cut}(A, B) = \sum_{i \in A, j \in B} w_{i,j} = \frac{1}{4} \sum_{i,j} w_{i,j}(f_i - f_j)^2 = \frac{1}{2} f^T L f \]

\[ |A| = |B| \implies \sum_i f_i = 0 \implies f \perp 1_N \]

\[ \|f\| = \sqrt{N} \]

**objective function of spectral clustering**

\[ \min_f f^T L f \quad \text{s.t.} \quad f_i = \pm 1, \quad f \perp 1_N, \quad \|f\| = \sqrt{N} \]

Still NP hard :(

\[ f_i = \pm 1 \quad \rightarrow \quad f_i \in \mathbb{R} \quad \text{Relax even further!} \]
Spectral Clustering: Relaxing Balanced Cuts

Objective function of spectral clustering

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

Rayleigh-Ritz theorem

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

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

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

$$\frac{x^T L x}{x^T x} \equiv \text{Rayleigh quotient}$$

How can we use it?
Spectral Clustering: Relaxing Balanced Cuts

Objective function of spectral clustering

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

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

If \(\lambda_1 \leq \cdots \leq \lambda_N\) are the eigenvectors of real symmetric \(L\) and \(v_1, \ldots, v_N\) the corresponding orthogonal eigenvalues, then for \(k = 1 : N - 1\)

\[
\lambda_{k+1} = \min_{x \neq 0, x \perp v_1, \ldots, v_k} \frac{x^{T}Lx}{x^{T}x} = \min_{x^{T}x=1, x \perp v_1, \ldots, v_k} x^{T}Lx
\]

\[
\lambda_{N-k} = \max_{x \neq 0, x \perp v_n, \ldots, v_{N-k+1}} \frac{x^{T}Lx}{x^{T}x} = \max_{x^{T}x=1, x \perp v_N, \ldots, v_{N-k+1}} x^{T}Lx
\]
Rayleigh-Ritz theorem: Quick and dirty proof

When we reach the extreme points?

\[ \frac{\partial}{\partial x} \left( x^T L x \right) = \frac{\partial}{\partial x} \left( \frac{f(x)}{g(x)} \right) = 0 \iff f'(x)g(x) = f(x)g'(x) \]

By matrix calculus (or just calculus):

\[ \frac{\partial x^T L x}{\partial x} = 2Lx \quad \text{and} \quad \frac{\partial x^T x}{\partial x} = 2x \]

When \( f'(x)g(x) = f(x)g'(x) \)?

\[ Lx \left( x^T x \right) = \left( x^T L x \right) x \iff Lx = \frac{x^T L x}{x^T x} x \iff Lx = \lambda x \]

Conclusion: Extremes are the eigenvectors with their eigenvalues
Spectral Clustering: Relaxing Balanced Cuts

**Objective function of spectral clustering**

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

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

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

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

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

$$f^T L f = \frac{1}{2} \sum_{i, j} w_{i, j} (f_i - f_j)^2 = (|A| + |B|) \text{RatioCut}(A, B)$$
Spectral Clustering: Approximating RatioCut

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

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

$$\sum_i f_i = 0$$

$$\sum_i f_i^2 = N$$

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

$$\min_f f^T L f \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad f \perp 1_N, \quad \|f\| = \sqrt{N}$$
Spectral Clustering: Approximating NCut

Normalized Cut

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

Define graph function \( f \) for cluster membership of NCut:

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

\( (Df)^T 1_n = 0 \quad f^T Df = \text{vol}(V) \quad f^T Lf = \text{vol}(V) \text{NCut}(A, B) \)

objective function of spectral clustering (NCut)

\[ \min_f f^T Lf \quad \text{s.t.} \quad f_i \in \mathbb{R}, \quad Df \perp 1_N, \quad f^T Df = \text{vol}(V) \]
Spectral Clustering: Approximating NCut

Minimizing the objective function of spectral clustering (Ncut):

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

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

Minimizing the objective function of spectral clustering (Ncut):

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

Minimizing the objective function of spectral clustering (Ncut):

\[
\min w^T L_{\text{sym}} w \quad \text{s.t.} \quad w_i \in \mathbb{R}, \quad w \perp v_{1, L_{\text{sym}}}, \quad ||w||^2 = \text{vol}(\mathcal{V})
\]
Spectral Clustering: Approximating NCut

Objective function of spectral clustering (NCut)

$$\min_w w^T L_{sym} w \quad \text{s.t.} \quad w_i \in \mathbb{R}, \quad w \perp v_1, L_{sym}, \quad \|w\| = \text{vol}(\mathcal{V})$$

Solution by Rayleigh-Ritz?

$$w = v_2, L_{sym} \quad f = D^{-1/2}w$$

**f** is a the second eigenvector of **L**<sub>RW</sub> !

**tl;dr:** Get the second eigenvector of **L**/<sub>RW</sub> for RatioCut/NCut.

demo: https://dominikschmidt.xyz/spectral-clustering-exp/
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

Eigenvectors

Eigenvalues

Eigenvalues

Eigenvalues

Eigenvalues

Eigenvalues

Eigenvalues

Eigenvalues

Eigenvalues

Eigenvalues
Spectral Clustering: Bibliography


Next class on Tuesday, October 22th at 13:30!
Michal Valko
contact via Piazza