Cheeger Cuts and p-Spectral Clustering

Matthias Hein

Department of Computer Science, Saarland University,
Saarbrücken, Germany

Joint work with: Thomas Bühler, Markus Maier and Ulrike von Luxburg

Roadmap

Graphs, Cuts and p-Spectral Clustering

- Similarity graphs in machine learning (random geometric graphs),
- The limit of the normalized cut criterion for different graph types - why the graph construction sometimes matters more than the algorithm on top,
- p-Spectral Clustering - a generalization of spectral clustering - how to get close to the optimal Cheeger cut.

Graph-based methods

Graphs capture relations:

- web graph,
- social networks,
- protein interaction networks,
- citation networks,

$\Longrightarrow$ no “absolute” features - only relative information.

Graph-based methods in machine learning:

- semi-supervised learning,
- dimensionality reduction (LLE, Laplacian Eigenmaps, Isomap,...),
- clustering (spectral clustering).

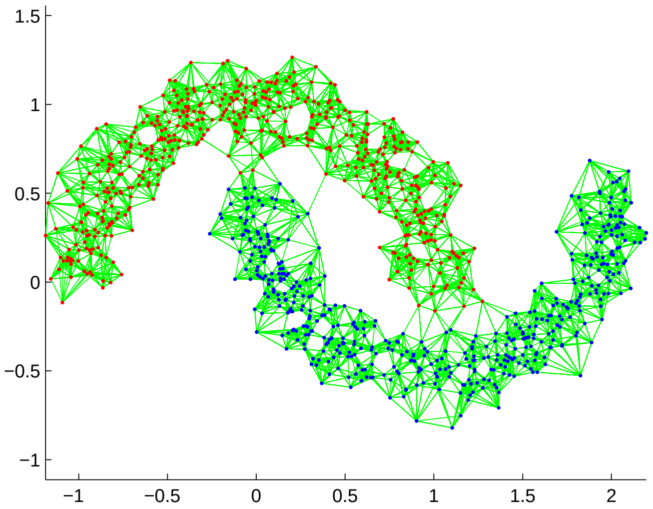
Graph-based methods

Similarity graphs in machine learning:

- data: $(X_i)_{i=1}^n$ in input space $\mathcal{X}$,
- given similarity measure: $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

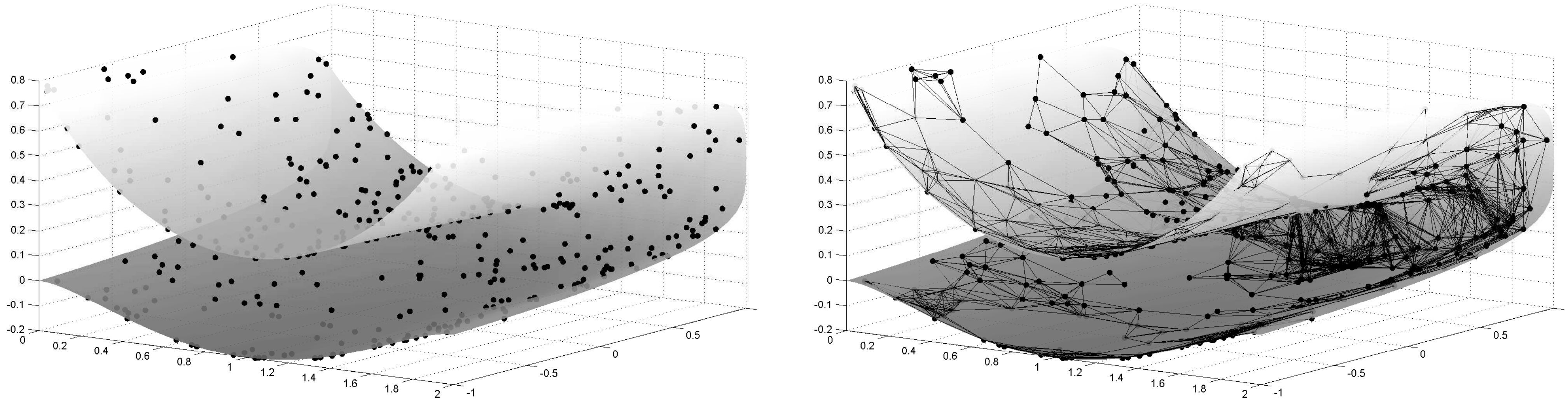
Graph construction:

- data points are vertices of the graph,
- **Idea:** connect similar points - build global structure from local structure.



Global from local information

Graphs in manifold learning:



Main assumption in manifold learning: Due to strong dependencies of the features, the data is concentrated around a low-dimensional structure.

$\Longrightarrow$ Similarity graph as discrete approximation of the continuous manifold.

How should one construct the similarity graph ?

How to construct similarity graphs

Neighborhood graphs: for a dissimilarity measure $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

- **k-nearest neighbor graphs:**

  $\mathrm{kNN}(X_i)$ denotes the k nearest neighbors of X_i.

  Connect points X_i and X_j if

$$\begin{array}{lcl} X_j \in \mathrm{kNN}(X_i) & \Rightarrow & \textbf{kNN-graph (directed)} \\ X_i \in \mathrm{kNN}(X_j) \textbf{ and } X_j \in \mathrm{kNN}(X_i) & \Rightarrow & \textbf{mutual kNN-graph} \\ X_i \in \mathrm{kNN}(X_j) \textbf{ or } X_j \in \mathrm{kNN}(X_i) & \Rightarrow & \textbf{symmetric kNN-graph} \end{array}$$

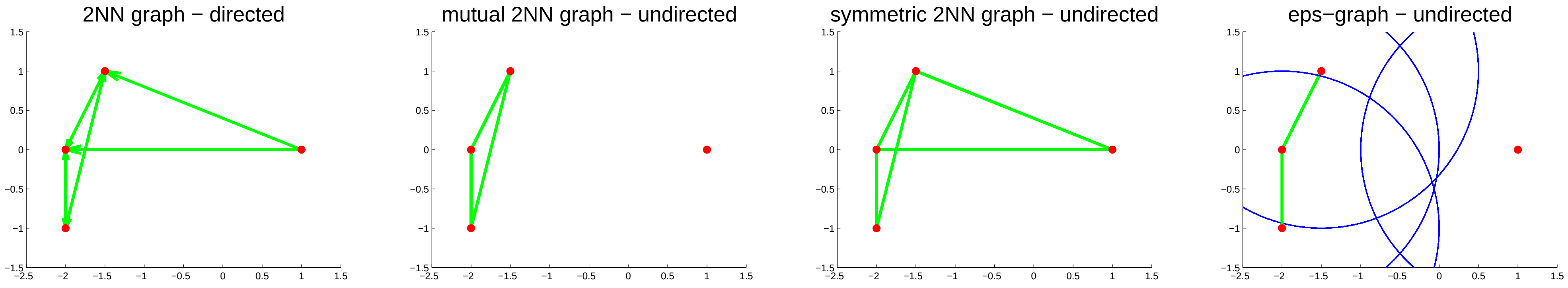
- **r-graphs:** Connect points X_i and X_j if

$$d(X_i, X_j) \le r \quad \Rightarrow \quad r\textbf{-graph (undirected)}$$

Statistical setting: $(X_i)_{i=1}^n$ is an i.i.d. sample of a probability measure P.

$\Longrightarrow$ These graphs are called **random geometric graphs**:

Illustration of different neighborhood graph types


2NN graph − directed

mutual 2NN graph − undirected

symmetric 2NN graph − undirected

eps−graph − undirected

Provocative statement:

The choice of the graph structure has at least as much influence on the learning performance as the choice of the learning algorithm on top.

Open questions in machine learning:

- Which graph type should one choose ? Are they all really the same ?
- What are the optimal parameters of the chosen graph type ?

Clustering

Definition of clustering:

Grouping of the data points $(X_i)_{i=1}^n$ such that points in each group are similar and points in different groups are dissimilar.

$\Longrightarrow$ no clear objective (different to supervised learning)

$\Longrightarrow$ clustering is ill-defined without specifying the objective !

Statistical model for clustering:

Clusters are the connected components of the levelset L_t of the density p,

$$L_t = \{x \in \mathbb{R}^d \,|\, p(x) \geq t\}.$$

Graph-based criteria for clustering:

- clusters are obtained by partitioning the similarity graph,
- no interpretation in terms of the data-generating probability measure.

Normalized cut

Clustering as graph partitioning

- complement of a set $A \subset V$ is $\overline{A} = V \backslash A$,
- degree function $d : V \to \mathbb{R}$, $d_i = \sum_{j=1}^{n} w_{ij}$,
- the cut of A and $\overline{A}$,

$$\mathrm{cut}(A, \overline{A}) = \sum_{i \in A,\, j \in \overline{A}} w_{ij}.$$

- Measure of volume: $|A|$ cardinality of the set A, and $\mathrm{vol}(A) = \sum\limits_{i \in A} d_i$.

Balanced graph cut criteria

Ratio cut:
$$\mathrm{RCut}(C, \overline{C}) = \mathrm{cut}(C, \overline{C}) \left(\frac{1}{|C|} + \frac{1}{|\overline{C}|} \right),$$

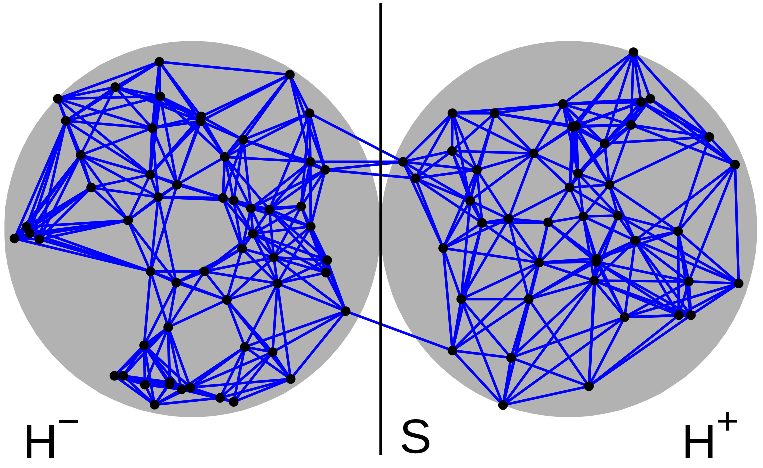
Normalized cut:
$$\mathrm{NCut}(C, \overline{C}) = \mathrm{cut}(C, \overline{C}) \left(\frac{1}{\mathrm{vol}(C)} + \frac{1}{\mathrm{vol}(\overline{C})} \right).$$

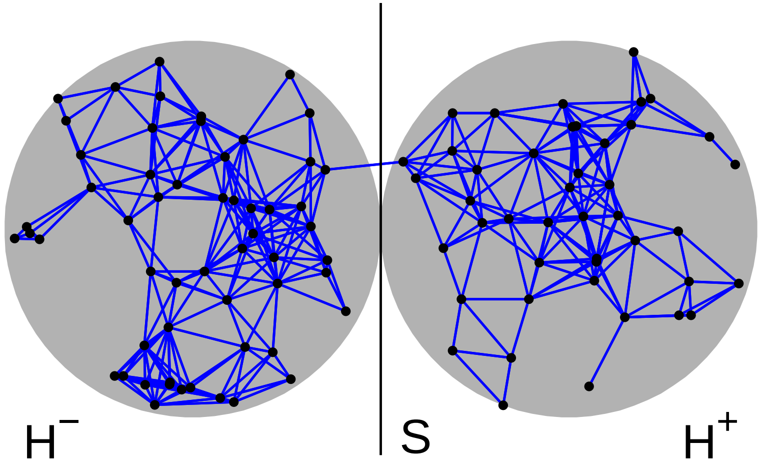
Normalized cut

Question: What is the clustering objective corresponding to the normalized cut in terms of the probability measure generating the data ? Does it depend on the graph type ?

Setting:

- $(X_i)_{i=1}^n$ sampled i.i.d. from a probability measure in $\mathbb{R}^d$ with density p,
- neighborhood graphs are unweighted,
- restrict possible cuts of the graph to cuts induced by hyperplanes in $\mathbb{R}^d$.





Left: kNN-graph with k=8, **Right:** corresponding r-graph.

Limit of the normalized cut

Theorem (Maier, von Luxburg, Hein (2009))

- limit results are obtained for a fixed hyperplane S,
- $\mathrm{NCut}(S) = \mathrm{cut}(S)\left(\frac{1}{\mathrm{vol}(H^+)} + \frac{1}{\mathrm{vol}(H^-)}\right)$,
- kNN-graph ($n \to \infty$, $k/\log n \to \infty$ and $k/n \to 0$):

$$\sqrt[d]{\frac{n}{k}}\mathrm{NCut}_{n,k} \xrightarrow{a.s.} c_d^{\mathrm{kNN}} \int_S p^{1-1/d}(s)\mathrm{d}s \left(\frac{1}{\int_{H^+} p(x)\mathrm{d}x} + \frac{1}{\int_{H^-} p(x)\mathrm{d}x}\right).$$

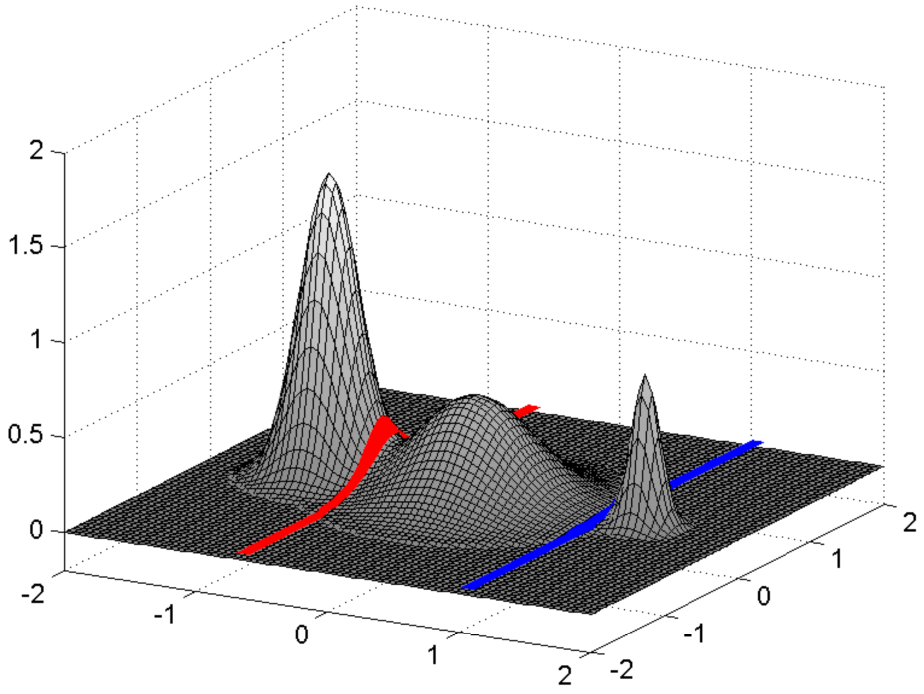
- r-graph: ($n \to \infty$, $r \to 0$ and $nr^{d+1} \to \infty$)

$$\frac{1}{r}\mathrm{NCut}_{n,r} \xrightarrow{a.s.} c_d^r \int_S p^2(s)\mathrm{d}s \left(\frac{1}{\int_{H^+} p^2(x)\mathrm{d}x} + \frac{1}{\int_{H^-} p^2(x)\mathrm{d}x}\right).$$

Limit of the normalized cut II

Does the difference matter ?

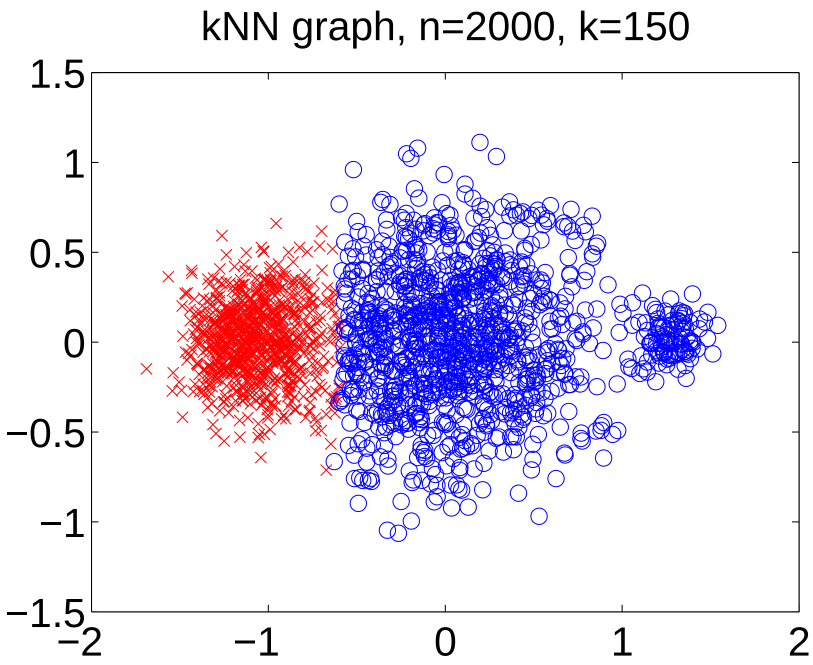
- Density is a mixture of three Gaussians in $\mathbb{R}^2$,
- Out of symmetry reasons the optimal (hyperplane) cut should be orthogonal to the axis connecting the means,
- **Red:** optimal cut for kNN-graph, **Blue:** optimal cut for r-graph

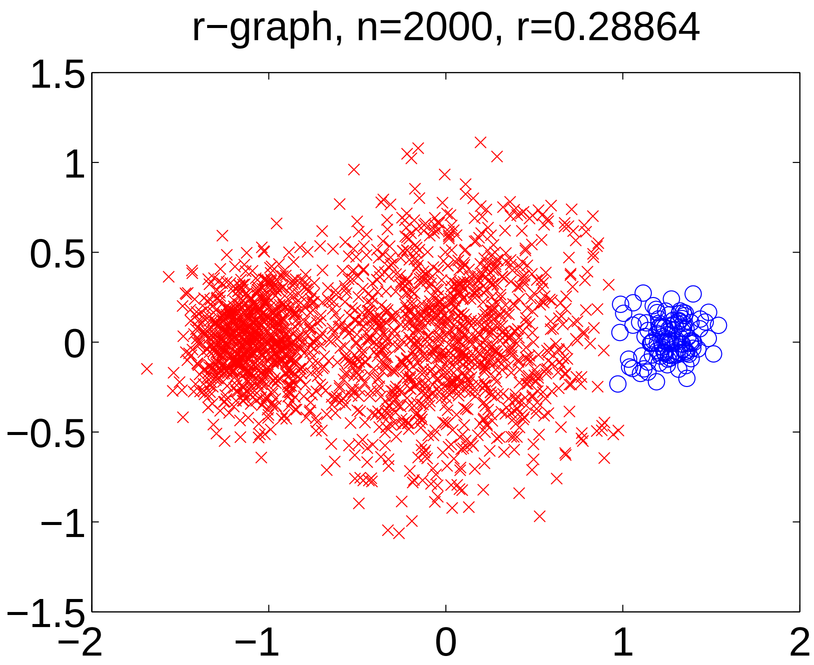


Limit of the normalized cut III

Do we see the difference in practice ?

- Finding the optimal normalized cut is NP-hard,
- In practice one uses spectral clustering (relaxation of normalized cut),
- Result of spectral clustering for the density of the last slide:





Radius of r-graph is chosen such that results are comparable.

Discussion

- Examples of differences also in higher dimensions - also results of spectral clustering is different.

  But: optimal cut is not at predicted place (boundary effects in high-dimensions).

- Limits of Ratio and Cheeger cut can also be derived.

- At the moment result holds only for unweighted graphs but can be extended to weighted graphs.

  $\Longrightarrow$ allows for the construction of clustering criteria with different influence of the density.

Discussion

- Examples of differences also in higher dimensions - also results of spectral clustering is different.

  But: optimal cut is not at predicted place (boundary effects in high-dimensions).

- Limits of Ratio and Cheeger cut can also be derived.

- At the moment result holds only for unweighted graphs but can be extended to weighted graphs.

  $\Longrightarrow$ allows for the construction of clustering criteria with different influence of the density.

Question for the rest of the talk

Is standard spectral clustering the best approximation to the normalized cut ?

The graph Laplacian

Notation: D diagonal degree matrix, W weight matrix of the graph.

The (unnormalized) graph Laplacian:

$$(\Delta f) = (D - W) f,$$

$$(\Delta f)_i = d_i f_i - \sum_{j \in V} w_{ij} f_j = \sum_{j \in V} w_{ij} (f_i - f_j).$$

Properties:

- Associated (regularization) functional:

$$\langle f, \Delta f \rangle = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2.$$

- If the graph is connected, only the first eigenvalue is zero and the corresponding eigenvector is $v^{(2)} = \mathbf{1}$.

Relaxation of ratio cut

Given a partition $C, \overline{C}$ define the function,

$$f_i^{(C)} = \begin{cases} \sqrt{|\overline{C}|/|C|} & i \in C, \\ -\sqrt{|C|/|\overline{C}|} & i \in \overline{C}. \end{cases}$$

$$\left\langle f^{(C)}, \Delta f^{(C)} \right\rangle = n \, \mathrm{RCut}(C, \overline{C}), \qquad \left\| f^{(C)} \right\|^2 = n, \qquad \left\langle f^{(C)}, \mathbf{1} \right\rangle = 0.$$

Optimal ratio cut: $\min_{C \subset V} \left\{ \frac{\langle f^{(C)}, \Delta f^{(C)} \rangle}{\| f^{(C)} \|^2} \;\middle|\; \left\langle f^{(C)}, \mathbf{1} \right\rangle = 0 \right\}.$

Relaxation of the ratio cut: $\min_{f \in \mathbb{R}^V} \left\{ \frac{\langle f, \Delta f \rangle}{\| f \|^2} \;\middle|\; \langle f, \mathbf{1} \rangle = 0 \right\}.$

$\Rightarrow$ Rayleigh-Ritz principle: solution is the second eigenvalue $\lambda^{(2)}$.

$\Rightarrow$ other relaxations leading to a semi-definite program are also possible.

Isoperimetric inequality

The ratio Cheeger cut:

$$\mathrm{RCC}(C,\overline{C}) = \frac{\mathrm{cut}(C,\overline{C})}{\min\{|C|,|\overline{C}|\}} \qquad \Big(\mathrm{RCut} = \mathrm{cut}(C,\overline{C})\Big(\frac{1}{|C|}+\frac{1}{|\overline{C}|}\Big)\Big).$$

Optimal ratio Cheeger cut: $h_{\mathrm{RCC}} = \inf_C \mathrm{RCC}(C,\overline{C})$.

Transformation of the second eigenvector $v^{(2)}$ into partition:

$$h^*_{\mathrm{RCC}} = \min_{C_t=\{i\in V\,|\,v^{(2)}(i)>t\}} \mathrm{RCC}(C_t,\overline{C_t}).$$

Using the isoperimetric inequality one can prove:

$$\frac{h_{\mathrm{RCC}}}{\max_i d_i} \leq \frac{h^*_{\mathrm{RCC}}}{\max_i d_i} \leq 2\sqrt{\frac{h_{\mathrm{RCC}}}{\max_i d_i}}.$$

The upper bound is achieved - tree-cross-path graph constructed by Guattery and Miller (1998).

The graph p-Laplacian

Does there exist an operator Δ_p which fulfills:

$$\langle f, \Delta_p f \rangle = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} |f_i - f_j|^p.$$

Yes ! The graph p-Laplacian:

$$(\Delta_p f)_i = \sum_{j \in V} w_{ij} \, |f_i - f_j|^{p-1} \operatorname{sign}(f_i - f_j),$$

First properties:

- One recovers the standard graph Laplacian for $p = 2$.
- The p-Laplacian (for $p \neq 2$) is non-linear,

$$\Delta_p(\alpha f) \neq \alpha \, \Delta_p f \qquad \text{for } \alpha \in \mathbb{R}.$$

How to define eigenvectors for a non-linear operator ?

Eigenvalues and eigenvectors of the p-Laplacian

Definition of an eigenvector:

$$(\Delta_p v)_i = \lambda_p \, |v_i|^{p-1} \operatorname{sign}(v_i), \quad \forall \, i = 1, \ldots, n.$$

Note: eigenvectors are invariant under rescaling.

Motivation by generalized Rayleigh-Ritz principle

$$F_p(f) := \frac{\langle f, \Delta_p f \rangle}{\|f\|_p^p} = \frac{1}{2} \frac{\sum_{i,j=1}^n w_{ij} |f_i - f_j|^p}{\sum_{i=1}^n |f_i|^p}.$$

Theorem:

- F_p has critical point at $v \in \mathbb{R}^V \iff v$ is p-eigenvector of Δ_p. The eigenvalue λ_p is then $\lambda_p = F_p(v)$,
- If the graph is connected, only the first eigenvalue is zero, $\lambda_p^{(1)} = 0$, and the first eigenvector is $v_p^{(1)} = \mathbf{1}$.

We need the second eigenvector for clustering !

Eigenvalues and eigenvectors of the p-Laplacian

Characterization of the second eigenvector for $p = 2$:

$$v^{(2)} = \mathop{\arg\min}_{f \in \mathbb{R}^n} \left\{ \frac{\langle f, \Delta_2\, f\rangle}{\|f\|_2^2} \;|\; \langle f, \mathbf{1}\rangle = 0. \right\},$$

equivalent: $$v^{(2)} = \mathop{\arg\min}_{f \in \mathbb{R}^n} \frac{\langle f, \Delta_2\, f\rangle}{\min_{c\in\mathbb{R}} \|f - c\,\mathbf{1}\|_2^2}.$$

Motivation for the general definition of $F_p^{(2)} : \mathbb{R}^V \rightarrow \mathbb{R}$,

$$F_p^{(2)}(f) = \frac{\langle f, \Delta_p f\rangle}{\min_{c\in\mathbb{R}} \|f - c\mathbf{1}\|_p^p} = \frac{\sum_{i,j=1}^n w_{ij} |f_i - f_j|^p}{\min_{c\in\mathbb{R}} \|f - c\mathbf{1}\|_p^p}.$$

Theorem

- The second eigenvalue $\lambda_p^{(2)}$ of Δ_p is the global minimum of $F_p^{(2)}$,
- The second eigenvector $v_p^{(2)}$ of Δ_p can be computed using the global minimizer of $F_p^{(2)}$.

To which balanced graph cut criterion corresponds $\lambda_p^{(2)}$?

Theoretical Motivation for p-Spectral Clustering

Corresponding relaxation for the p-Laplacian:

For $p > 1$ the second eigenvalue $\lambda_p^{(2)}$ of the p-Laplacian is the solution of a relaxation of the problem:

$$\min_{C \subset V} \ \mathrm{cut}(C, \overline{C}) \left| \frac{1}{|C|^{\frac{1}{p-1}}} + \frac{1}{|\overline{C}|^{\frac{1}{p-1}}} \right|^{p-1},$$

Interpolation between the special cases,

$$p = 2, \quad \min_{C \subset V} \ \mathrm{RCut}(C, \overline{C}),$$

$$p \rightarrow 1, \quad \min_{C \subset V} \ \mathrm{RCC}(C, \overline{C}).$$

The limit $p \rightarrow 1$ follows with

$$\lim_{\alpha \rightarrow \infty} (a^\alpha + b^\alpha)^{1/\alpha} = \max\{a, b\}.$$

Theoretical Motivation for p-Spectral Clustering II

Thresholding the second eigenvector $v_p^{(2)}$ to get the partition:

$$h^*_{\mathrm{RCC}} = \min_{C_t=\{i\in V \,|\, v_p^{(2)}(i)>t\}} \mathrm{RCC}(C_t, \overline{C_t}).$$

Extension of isoperimetric inequality by Amghibech (2003)

Theorem 1. *Denote by $\lambda_p^{(2)}$ the second eigenvalue of the p-Laplacian Δ_p.*

$$\textit{For any } p>1, \qquad \left(\frac{2}{\max_i d_i}\right)^{p-1}\left(\frac{h_{\mathrm{RCC}}}{p}\right)^{p} \;\le\; \lambda_p^{(2)} \;\le\; 2^{p-1} h_{\mathrm{RCC}}\ .$$

Motivation for p-Spectral Clustering (Bühler, Hein (2009)):

Theorem 2. *For $p>1$,*

$$\frac{h_{\mathrm{RCC}}}{\max_i d_i} \;\le\; \frac{h^*_{\mathrm{RCC}}}{\max_i d_i} \;\le\; p\left(\frac{h_{\mathrm{RCC}}}{\max_i d_i}\right)^{\frac{1}{p}}$$

$\Longrightarrow$ **upper bound gets tight as $p \to 1$!**

How to minimize $F_p^{(2)}$

Minimization of $F_p^{(2)}$

- $F_p^{(2)}$ is non-convex and is minimized over non-convex domain
  $\Longrightarrow$ direct minimization for small p leads to suboptimal local minima.
- Idea:
  - for $p = 2$ we know the global minimizer and $F_p^{(2)}$ is continuous in p,
  - (local) minima for close p should be close,

  $\Longrightarrow$ decrease p in small steps and optimize for fixed p with a pseudo-Newton method (sparsity !).
- Empirical observation:
  - we can solve large-scale problems (70000 points),
  - runtime increases dramatically as $p \rightarrow 1$ ($p = 2$, 10s; $p = 1.2$, $4660s$)
  - found Cheeger cut is always at least as good as the cut found by standard spectral clustering - often much better.

p-Spectral Clustering

```
Input: weight matrix $W$, number of desired clusters $k$, choice of $p$-Laplacian.
Initialization: cluster $C_1 = V$, number of clusters $s = 1$
repeat
    Minimize $F_p^{(2)} : \mathbb{R}^{C_i} \to \mathbb{R}$ for the chosen $p$-Laplacian for each cluster $C_i$, $i = 1, \ldots, s$.
    Compute optimal threshold for dividing each cluster $C_i$.
    Choose to split the cluster $C_i$ so that the total multi-partition cut criterion is minimized.
    $s \Leftarrow s + 1$
until number of clusters $s = k$
```

Multi-Partition Criterion: $\mathrm{RCut}(C_1, \ldots, C_k) = \sum_{i=1}^{k} \frac{\mathrm{cut}(C_i, \overline{C_i})}{|C_i|}$.

Experimental setting

Neighborhood graph:

symmetric k-NN graph with $k = 10$ and weights w_{ij} defined as

$$w_{ij} = \max\{\theta_i(j), \theta_j(i)\}, \qquad \text{where} \;\; \theta_i(j) = e^{-\frac{4}{\sigma_i^2}\|x_i - x_j\|^2},$$

with σ_i being the Euclidean distance of x_i to its k-nearest neighbor.

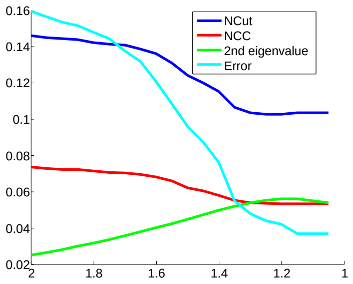
Evaluation:

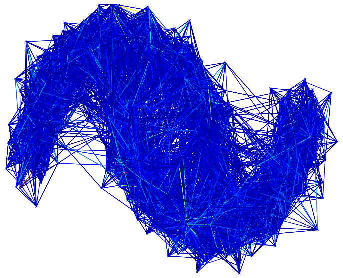
We used supervised datasets with known number of classes $K \Longrightarrow K$ clusters. Agreement of the found clusters $C_1, \ldots, C_K$ with the class structure is measured using

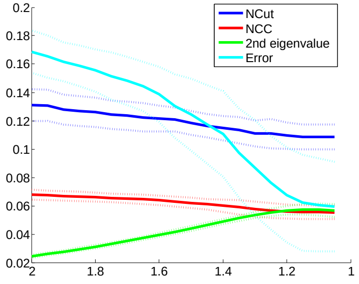
$$\mathrm{err}(C_1, .., C_k) = \frac{1}{|V|} \sum_{i=1}^{k} \sum_{j \in C_i} {}_{Y_j \neq Y_i'},$$

where Y_j is the true label of j and Y_i' is the dominant label in cluster C_i.

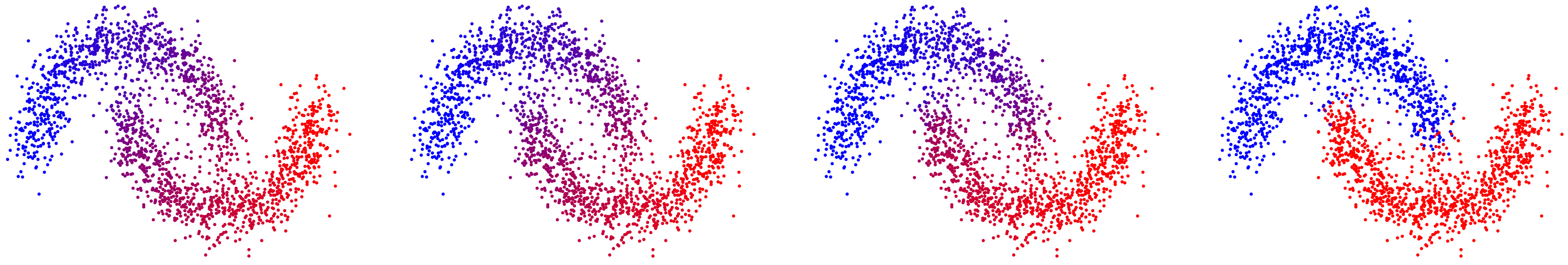
Experimental results - High-dimensional toy data

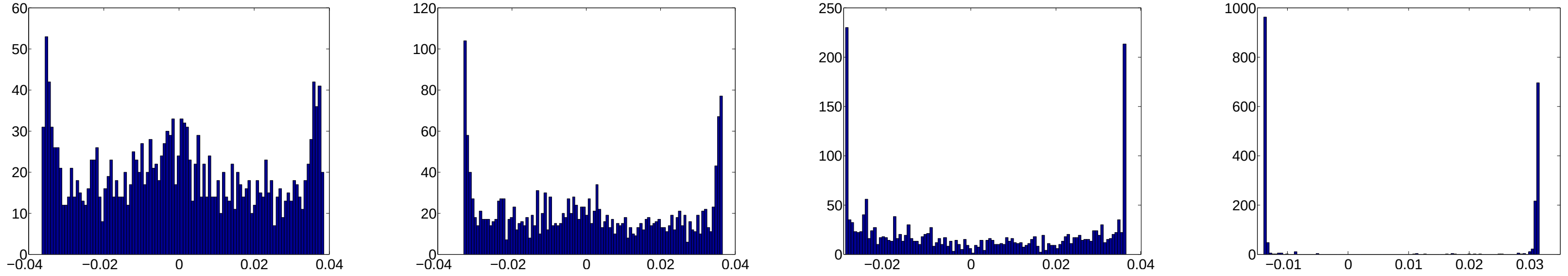




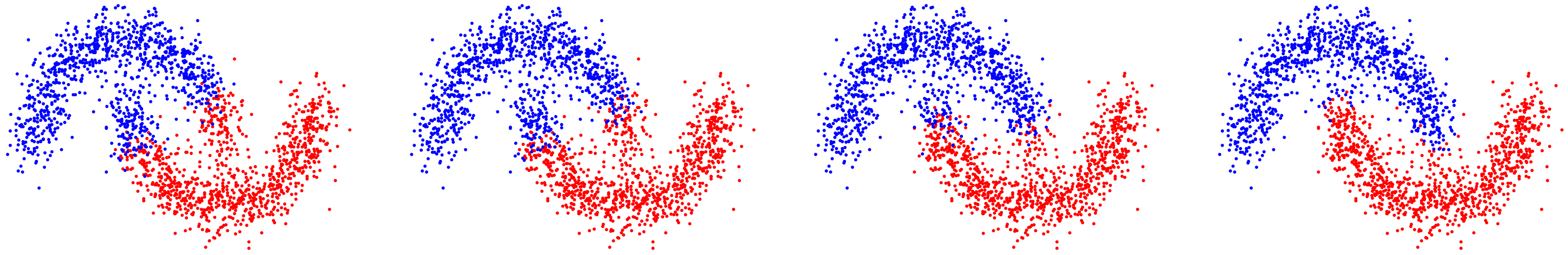


$p = 2.0$ $p = 1.7$ $p = 1.4$ $p = 1.1$



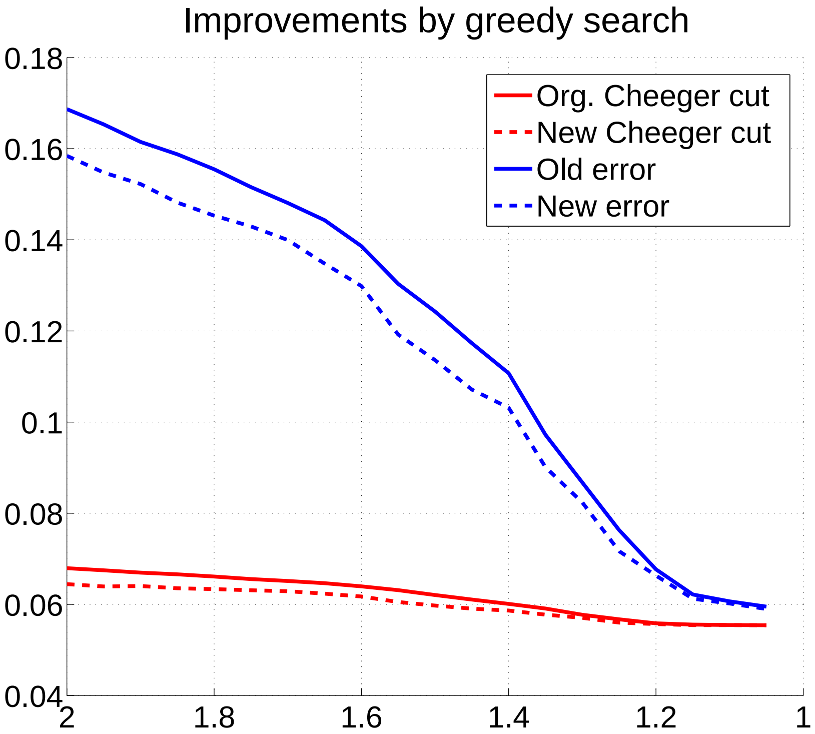


`NCut 0.1461 NCC 0.07374` `NCut 0.1408 NCC 0.07050` `NCut 0.1155 NCC 0.05791` `NCut 0.1035 NCC 0.05330`



$$\mathrm{NCC}(C,\overline{C}) = \frac{\mathrm{cut}(C,\overline{C})}{\min\{\mathrm{vol}(C),\mathrm{vol}(\overline{C})\}}, \qquad \mathrm{NCut}(C,\overline{C}) = \mathrm{cut}(C,\overline{C})\left(\frac{1}{\mathrm{vol}(C)}+\frac{1}{\mathrm{vol}(\overline{C})}\right).$$

How stable is the result ?



Test of optimality by greedy search:

- flip the assignment for each vertex,
- compute the resulting Cheeger cut,
- flip the vertex leading to the smallest Cheeger cut,
- repeat until no flip leads to a better Cheeger cut.

$\Longrightarrow$ often zero resp. only a few flips for small values of p

$\Longrightarrow$ zero flips $\Longrightarrow$ found local optima.

Experimental results - USPS and MNIST

	USPS		MNIST	
p	RCut	Error	RCut	Error
2.0	0.819	0.233	0.225	0.189
1.9	0.741	0.142	0.209	0.172
1.8	0.718	0.141	0.186	0.170
1.7	0.698	0.139	0.170	0.169
1.6	0.684	0.134	0.164	0.170
1.5	0.676	0.133	0.161	0.133
1.4	0.693	0.141	0.158	0.132
1.3	0.684	0.138	0.155	0.131
1.2	0.679	0.137	0.153	0.129

True/Cluster	0	1	2	3	4	5	6	7	8
0	6845	5	7	0	5	8	26	4	3
1	1	7794	32	8	21	1	2	16	2
2	38	47	6712	25	15	5	8	114	26
3	5	6	31	6939	30	61	2	45	22
4	3	45	2	1	6750	0	14	5	4
5	15	1	4	92	39	6087	61	5	9
6	23	17	6	0	9	23	6797	0	1
7	1	83	22	1	116	2	0	7067	1
8	18	51	13	507	112	122	23	18	5961
9	15	15	3	117	6708	11	4	77	8

Datasets of handwritten digits:

USPS (9278 digits) and MNIST (70000 digits) - 10 classes

- cut and error decrease as $p \to 1$,
- the error could even be better since class 1 has been split into two clusters and class 4 and 9 have been merged
  $\Longrightarrow$ confusion table on the right.

Summary and Outlook

- The normalized cut criterion has a different population limit dependent on the employed graph type $\Longrightarrow$ provides first understanding of the modeling aspect of graphs in machine learning.
- p-Spectral clustering as a generalization of spectral clustering yields partitions with better cut values $\Longrightarrow$ in the limit of $p \to 1$ the resulting partition approximates the Cheeger cut arbitrarily well.
  - application of p-spectral clustering to image segmentation.
  - coarse-fine grain approach to speed up calculations.
  - higher order eigenvectors for dimensionality reduction ?

Thank you for your attention !