MATH 829: Introduction to Data Mining and Analysis
Clustering III

Dominique Guillot

Departments of Mathematical Sciences
University of Delaware

April 29, 2016

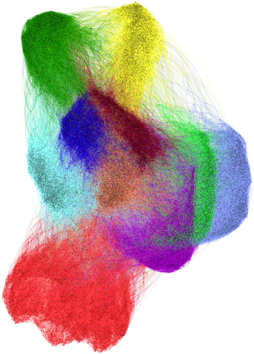
This lecture is based on U. von Luxburg, *A Tutorial on Spectral Clustering*, Statistics and Computing, 17 (4), 2007.

Graph cuts

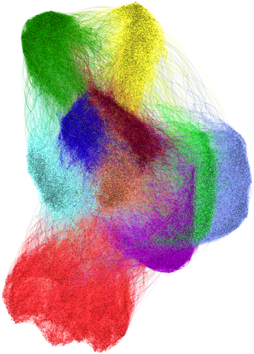
- G graph with (weighted) adjacency matrix $W = (w_{ij})$.
- We define:

$$W(A,B) := \sum_{i \in A, j \in B} w_{ij}.$$

- $|A|$:= number of vertices in A.
- $\mathrm{vol}(A) := \sum_{i \in A} d_i$.



Graph cuts



- G graph with (weighted) adjacency matrix $W = (w_{ij})$.
- We define:

$$W(A, B) := \sum_{i \in A, j \in B} w_{ij}.$$

- $|A| :=$ number of vertices in A.
- $\mathrm{vol}(A) := \sum_{i \in A} d_i$.

Given a partition $A_1, \ldots, A_k$ of the vertices of G, we let

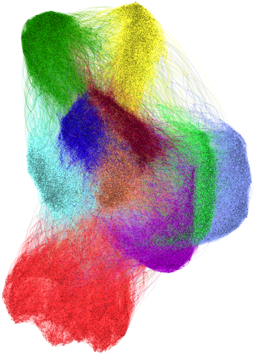
$$\mathrm{cut}(A_1, \ldots, A_k) := \frac{1}{2} \sum_{i=1}^{k} W(A_i, \overline{A}_i).$$

Graph cuts

- G graph with (weighted) adjacency matrix $W = (w_{ij})$.
- We define:

$$W(A,B) := \sum_{i\in A, j\in B} w_{ij}.$$

- $|A| :=$ number of vertices in A.
- $\mathrm{vol}(A) := \sum_{i\in A} d_i$.



Given a partition $A_1, \ldots, A_k$ of the vertices of G, we let

$$\mathrm{cut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} W(A_i, \overline{A}_i).$$

The min-cut problem consists of solving:

$$\min_{\substack{V = A_1\cup\cdots\cup A_k \\ A_i\cap A_j = \emptyset\ \forall i\neq j}} \mathrm{cut}(A_1, \ldots, A_k).$$

Graph cuts (cont.)

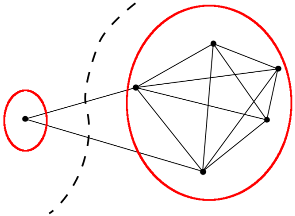
- The min-cut problem can be solved efficiently when $k = 2$ (see Stoer and Wagner 1997).

Graph cuts (cont.)

- The min-cut problem can be solved efficiently when $k = 2$ (see Stoer and Wagner 1997).
- In practice it often does not lead to satisfactory partitions.

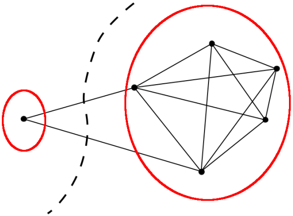
Graph cuts (cont.)

- The min-cut problem can be solved efficiently when $k = 2$ (see Stoer and Wagner 1997).
- In practice it often does not lead to satisfactory partitions.
- In many cases, the solution of min-cut simply separates one individual vertex from the rest of the graph.



Graph cuts (cont.)

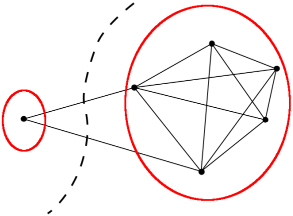
- The min-cut problem can be solved efficiently when $k = 2$ (see Stoer and Wagner 1997).
- In practice it often does not lead to satisfactory partitions.
- In many cases, the solution of min-cut simply separates one individual vertex from the rest of the graph.



- We would like clusters to have a reasonably large number of points.

Graph cuts (cont.)

- The min-cut problem can be solved efficiently when $k = 2$ (see Stoer and Wagner 1997).
- In practice it often does not lead to satisfactory partitions.
- In many cases, the solution of min-cut simply separates one individual vertex from the rest of the graph.



- We would like clusters to have a reasonably large number of points.
- We therefore modify the min-cut problem to enforce such constraints.

Balanced cuts

The two most common objective functions that are used as a replacement to the min-cut objective are:

1. RatioCut (Hagen and Kahng, 1992):

$$\mathrm{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{|A_i|}.$$

Balanced cuts

The two most common objective functions that are used as a replacement to the min-cut objective are:

1. RatioCut (Hagen and Kahng, 1992):

$$\mathrm{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{|A_i|}.$$

2. Normalized cut (Shi and Malik, 2000):

$$\mathrm{Ncut}(A_1, \ldots, A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{\mathrm{vol}(A_i)} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{\mathrm{vol}(A_i)}.$$

Balanced cuts

The two most common objective functions that are used as a replacement to the min-cut objective are:

1. RatioCut (Hagen and Kahng, 1992):

$$\mathrm{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{|A_i|}.$$

2. Normalized cut (Shi and Malik, 2000):

$$\mathrm{Ncut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{\mathrm{vol}(A_i)} = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{\mathrm{vol}(A_i)}.$$

- Note: both objective functions take larger values when the clusters A_i are "small".

Balanced cuts

The two most common objective functions that are used as a replacement to the min-cut objective are:

1. RatioCut (Hagen and Kahng, 1992):

$$\text{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{|A_i|}.$$

2. Normalized cut (Shi and Malik, 2000):

$$\text{Ncut}(A_1, \ldots, A_k) := \frac{1}{2}\sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{\text{vol}(A_i)} = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{\text{vol}(A_i)}.$$

- Note: both objective functions take larger values when the clusters A_i are "small".
- Resulting clusters are more "balanced".

Balanced cuts

The two most common objective functions that are used as a replacement to the min-cut objective are:

1. RatioCut (Hagen and Kahng, 1992):

$$\text{RatioCut}(A_1, \ldots, A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{|A_i|}.$$

2. Normalized cut (Shi and Malik, 2000):

$$\text{Ncut}(A_1, \ldots, A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \overline{A}_i)}{\text{vol}(A_i)} = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{\text{vol}(A_i)}.$$

- Note: both objective functions take larger values when the clusters A_i are "small".
- Resulting clusters are more "balanced".
- However, the resulting problems are NP hard - see Wagner and Wagner (1993).

Spectral clustering

Spectral clustering provides a way to *relax* the RatioCut and the Normalized cut problems.

Spectral clustering

Spectral clustering provides a way to *relax* the RatioCut and the Normalized cut problems.

Strategy:

1. Express the original problem as a linear algebra problem involving discrete/combinatorial constraints.

Spectral clustering

Spectral clustering provides a way to *relax* the RatioCut and the Normalized cut problems.

Strategy:

1. Express the original problem as a linear algebra problem involving discrete/combinatorial constraints.
2. Relax/remove the constraints.

Spectral clustering

Spectral clustering provides a way to *relax* the RatioCut and the Normalized cut problems.

Strategy:

1. Express the original problem as a linear algebra problem involving discrete/combinatorial constraints.
2. Relax/remove the constraints.

RatioCut with $k = 2$: solve

$$\min_{A \subset V} \mathrm{RatioCut}(A, \overline{A}).$$

Spectral clustering provides a way to *relax* the RatioCut and the Normalized cut problems.

Strategy:

1. Express the original problem as a linear algebra problem involving discrete/combinatorial constraints.
2. Relax/remove the constraints.

RatioCut with $k = 2$: solve

$$\min_{A \subset V} \mathrm{RatioCut}(A, \overline{A}).$$

Given $A \subset V$, let $f \in \mathbb{R}^n$ be given by

$$f_i := \begin{cases} \sqrt{|\overline{A}|/|A|} & \text{if } v_i \in A \\ -\sqrt{|A|/|\overline{A}|} & \text{if } v_i \notin A. \end{cases}$$

Relaxing RatioCut

Let $L = D - W$ be the (unnormalized) Laplacian of G. Then

$$
\begin{aligned}
f^T L f &= \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2 \\
&= \frac{1}{2} \sum_{i \in A, j \in \overline{A}} w_{ij} \left(\sqrt{\frac{|\overline{A}|}{|A|}} + \sqrt{\frac{|A|}{|\overline{A}|}} \right)^2 + \frac{1}{2} \sum_{i \in \overline{A}, j \in A} w_{ij} \left(-\sqrt{\frac{|\overline{A}|}{|A|}} - \sqrt{\frac{|A|}{|\overline{A}|}} \right)^2 \\
&= W(A, \overline{A}) \left(2 + \frac{|\overline{A}|}{|A|} + \frac{|A|}{|\overline{A}|} \right) \\
&= W(A, \overline{A}) \left(\frac{|A| + |\overline{A}|}{|A|} + \frac{|A| + |\overline{A}|}{|\overline{A}|} \right) \\
&= |V| \cdot \frac{1}{2} \left(\frac{W(A, \overline{A})}{|A|} + \frac{W(\overline{A}, A)}{|\overline{A}|} \right) \\
&= |V| \cdot \mathrm{RatioCut}(A, \overline{A}).
\end{aligned}
$$

since $|A| + |\overline{A}| = |V|$, and $W(A, \overline{A}) = W(\overline{A}, A)$.

Relaxing RatioCut (cont.)

- We showed:

$$f^T L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2 = |V| \cdot \mathrm{RatioCut}(A, \overline{A}).$$

Relaxing RatioCut (cont.)

- We showed:

$$f^T L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2 = |V| \cdot \mathrm{RatioCut}(A, \overline{A}).$$

- Moreover, note that

$$\sum_{i=1}^{n} f_i = \sum_{i \in A} \sqrt{\frac{|\overline{A}|}{|A|}} - \sum_{i \in \overline{A}} \sqrt{\frac{|A|}{|\overline{A}|}} = |A| \cdot \sqrt{\frac{|\overline{A}|}{|A|}} - |\overline{A}| \cdot \sqrt{\frac{|A|}{|\overline{A}|}} = 0.$$

Relaxing RatioCut (cont.)

- We showed:

$$f^T L f = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2 = |V| \cdot \mathrm{RatioCut}(A, \overline{A}).$$

- Moreover, note that

$$\sum_{i=1}^{n} f_i = \sum_{i \in A}\sqrt{\frac{|\overline{A}|}{|A|}} - \sum_{i \in \overline{A}}\sqrt{\frac{|A|}{|\overline{A}|}} = |A| \cdot \sqrt{\frac{|\overline{A}|}{|A|}} - |\overline{A}| \cdot \sqrt{\frac{|A|}{|\overline{A}|}} = 0.$$

Thus $f \perp \mathbb{1}$.

Relaxing RatioCut (cont.)

- We showed:

$$f^T L f = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2 = |V| \cdot \mathrm{RatioCut}(A, \overline{A}).$$

- Moreover, note that

$$\sum_{i=1}^{n} f_i = \sum_{i\in A}\sqrt{\frac{|\overline{A}|}{|A|}} - \sum_{i\in \overline{A}}\sqrt{\frac{|A|}{|\overline{A}|}} = |A| \cdot \sqrt{\frac{|\overline{A}|}{|A|}} - |\overline{A}| \cdot \sqrt{\frac{|A|}{|\overline{A}|}} = 0.$$

Thus $f \perp \mathbb{1}$.

- Finally,

$$\|f\|_2^2 = \sum_{i=1}^{n} f_i^2 = |A| \cdot \frac{|\overline{A}|}{|A|} + |\overline{A}| \cdot \frac{|A|}{|\overline{A}|} = |V| = n.$$

Relaxing RatioCut (cont.)

- We showed:

$$f^T L f = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2 = |V| \cdot \mathrm{RatioCut}(A, \overline{A}).$$

- Moreover, note that

$$\sum_{i=1}^{n} f_i = \sum_{i \in A} \sqrt{\frac{|\overline{A}|}{|A|}} - \sum_{i \in \overline{A}} \sqrt{\frac{|A|}{|\overline{A}|}} = |A| \cdot \sqrt{\frac{|\overline{A}|}{|A|}} - |\overline{A}| \cdot \sqrt{\frac{|A|}{|\overline{A}|}} = 0.$$

Thus $f \perp \mathbb{1}$.

- Finally,

$$\|f\|_2^2 = \sum_{i=1}^{n} f_i^2 = |A| \cdot \frac{|\overline{A}|}{|A|} + |\overline{A}| \cdot \frac{|A|}{|\overline{A}|} = |V| = n.$$

Thus, we have showed that the Ratio-Cut problem is equivalent to

$$\min_{A \subset V} f^T L f$$

$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}, f_i \text{ defined as above.}$$

Relaxing RatioCut (cont.)

We have:

$$\min_{A \subset V} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}, f_i \text{ defined as above.}$$

Relaxing RatioCut (cont.)

We have:

$$\min_{A \subset V} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}, f_i \text{ defined as above.}$$

- This is a discrete optimization problem since the entries of f can only take two values: $\sqrt{|\overline{A}|/|A|}$ and $-\sqrt{|A|/|\overline{A}|}$.

Relaxing RatioCut (cont.)

We have:

$$\min_{A \subset V} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}, f_i \text{ defined as above.}$$

- This is a discrete optimization problem since the entries of f can only take two values: $\sqrt{|\overline{A}|/|A|}$ and $-\sqrt{|A|/|\overline{A}|}$.
- The problem is NP hard.

Relaxing RatioCut (cont.)

We have:

$$\min_{A \subset V} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}, f_i \text{ defined as above.}$$

- This is a discrete optimization problem since the entries of f can only take two values: $\sqrt{|\overline{A}|/|A|}$ and $-\sqrt{|A|/|\overline{A}|}$.
- The problem is NP hard.

The natural relaxation of the problem is to **remove the discreteness condition** on f and solve

$$\min_{f \in \mathbb{R}^n} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}.$$

Relaxing RatioCut (cont.)

- Using properties of the Rayleigh quotient, it is not hard to show that the solution of

$$\min_{f \in \mathbb{R}^n} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}.$$

is an eigenvector of L corresponding to the second eigenvalue of L.

Relaxing RatioCut (cont.)

- Using properties of the Rayleigh quotient, it is not hard to show that the solution of

$$\min_{f \in \mathbb{R}^n} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}.$$

is an eigenvector of L corresponding to the second eigenvalue of L.

- Clearly, if $\tilde{f}$ is the solution of the problem, then

$$\tilde{f}^T L \tilde{f} \leq \min_{A \subset V} \mathrm{RatioCut}(A, \overline{A}).$$

Relaxing RatioCut (cont.)

- Using properties of the Rayleigh quotient, it is not hard to show that the solution of

$$\min_{f \in \mathbb{R}^n} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}.$$

is an eigenvector of L corresponding to the second eigenvalue of L.

- Clearly, if $\tilde{f}$ is the solution of the problem, then

$$\tilde{f}^T L \tilde{f} \leq \min_{A \subset V} \mathrm{RatioCut}(A, \overline{A}).$$

- How do we get the clusters from $\tilde{f}$?
- We could set

$$\begin{cases} v_i \in A & \text{if } f_i \geq 0 \\ v_i \in \overline{A} & \text{if } f_i < 0. \end{cases}$$

Relaxing RatioCut (cont.)

- Using properties of the Rayleigh quotient, it is not hard to show that the solution of

$$\min_{f \in \mathbb{R}^n} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}.$$

is an eigenvector of L corresponding to the second eigenvalue of L.

- Clearly, if $\tilde{f}$ is the solution of the problem, then

$$\tilde{f}^T L \tilde{f} \leq \min_{A \subset V} \text{RatioCut}(A, \overline{A}).$$

- How do we get the clusters from $\tilde{f}$?
- We could set

$$\begin{cases} v_i \in A & \text{if } f_i \geq 0 \\ v_i \in \overline{A} & \text{if } f_i < 0. \end{cases}$$

- More generally, we *cluster* the coordinates of f using K-means.

Relaxing RatioCut (cont.)

- Using properties of the Rayleigh quotient, it is not hard to show that the solution of

$$\min_{f \in \mathbb{R}^n} f^T L f$$
$$\text{subject to } f \perp \mathbb{1}, \|f\| = \sqrt{n}.$$

is an eigenvector of L corresponding to the second eigenvalue of L.

- Clearly, if $\tilde{f}$ is the solution of the problem, then

$$\tilde{f}^T L \tilde{f} \leq \min_{A \subset V} \mathrm{RatioCut}(A, \overline{A}).$$

- How do we get the clusters from $\tilde{f}$?
- We could set

$$\begin{cases} v_i \in A & \text{if } f_i \geq 0 \\ v_i \in \overline{A} & \text{if } f_i < 0. \end{cases}$$

- More generally, we *cluster* the coordinates of f using K-means.

This is the **unnormalized spectral clustering algorithm** for $k = 2$.

Relaxing RatioCut : $k > 2$

- We saw that the second eigenvector of L solves our relaxation of the RatioCut problem for $k = 2$.

Relaxing RatioCut : $k > 2$

- We saw that the second eigenvector of L solves our relaxation of the RatioCut problem for $k = 2$.
- How do we proceed when we want $k > 2$ clusters?

- We saw that the second eigenvector of L solves our relaxation of the RatioCut problem for $k = 2$.
- How do we proceed when we want $k > 2$ clusters?

Given a partition $A_1, \ldots, A_k$ of V, we define k indicator **vectors**

$$h_j = (h_{1,j}, \ldots, h_{n,j}) \in \mathbb{R}^n \qquad (j = 1, \ldots, k)$$

as follows:

$$h_{i,j} := \begin{cases} \frac{1}{\sqrt{|A_j|}} & \text{if } v_i \in A_j \\ 0 & \text{otherwise.} \end{cases}$$

Relaxing RatioCut : $k > 2$

- We saw that the second eigenvector of L solves our relaxation of the RatioCut problem for $k = 2$.
- How do we proceed when we want $k > 2$ clusters?

Given a partition $A_1, \dots, A_k$ of V, we define k indicator **vectors**

$$h_j = (h_{1,j}, \dots, h_{n,j}) \in \mathbb{R}^n \qquad (j = 1, \dots, k)$$

as follows:

$$h_{i,j} := \begin{cases} \frac{1}{\sqrt{|A_j|}} & \text{if } v_i \in A_j \\ 0 & \text{otherwise.} \end{cases}$$

Let $H := (h_{ij}) \in \mathbb{R}^{n \times k}$. Note that the columns h_i of H are orthonormal, i.e., $H^T H = I_{k \times k}$.

Relaxing RatioCut : $k > 2$

- We saw that the second eigenvector of L solves our relaxation of the RatioCut problem for $k = 2$.
- How do we proceed when we want $k > 2$ clusters?

Given a partition $A_1, \dots, A_k$ of V, we define k indicator **vectors**

$$h_j = (h_{1,j}, \dots, h_{n,j}) \in \mathbb{R}^n \qquad (j = 1, \dots, k)$$

as follows:

$$h_{i,j} := \begin{cases} \frac{1}{\sqrt{|A_j|}} & \text{if } v_i \in A_j \\ 0 & \text{otherwise.} \end{cases}$$

Let $H := (h_{ij}) \in \mathbb{R}^{n \times k}$. Note that the columns h_i of H are orthonormal, i.e., $H^T H = I_{k \times k}$.

A similar calculation as we did before shows that (exercise):

$$h_i^T L h_i = \frac{\text{cut}(A_i, \overline{A}_i)}{|A_i|}.$$

Relaxing RatioCut : $k > 2$

- Now,

$$h_i^T L h_i = (H^T L H)_{ii}.$$

Relaxing RatioCut : $k > 2$

- Now,

$$h_i^T L h_i = (H^T L H)_{ii}.$$

- Thus,

$$\mathrm{RatioCut}(A_1, \ldots, A_k) = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} h_i^T L h_i = \mathrm{Tr}(H^T L H).$$

Relaxing RatioCut : $k > 2$

- Now,

$$h_i^T L h_i = (H^T L H)_{ii}.$$

- Thus,

$$\mathrm{RatioCut}(A_1, \ldots, A_k) = \sum_{i=1}^{k} \frac{\mathrm{cut}(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} h_i^T L h_i = \mathrm{Tr}(H^T L H).$$

- So the problem

$$\min_{\substack{V = A_1 \cup \cdots \cup A_k \\ A_i \cap A_j = \emptyset \ \forall i \neq j}} \mathrm{RatioCut}(A_1, \ldots, A_k)$$

Relaxing RatioCut : $k > 2$

- Now,

$$h_i^T L h_i = (H^T L H)_{ii}.$$

- Thus,

$$\text{RatioCut}(A_1, \dots, A_k) = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} h_i^T L h_i = \text{Tr}(H^T L H).$$

- So the problem

$$\min_{\substack{V = A_1 \cup \dots \cup A_k \\ A_i \cap A_j = \emptyset \ \forall i \neq j}} \text{RatioCut}(A_1, \dots, A_k)$$

is equivalent to

$$\min_{H \in \mathbb{R}^{n \times k}} \text{Tr}(H^T L H)$$

$$\text{subject to } H^T H = I_{k \times k}, H \text{ defined as above.}$$

Relaxing RatioCut : $k > 2$

- Now,

$$h_i^T L h_i = (H^T L H)_{ii}.$$

- Thus,

$$\text{RatioCut}(A_1, \ldots, A_k) = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A}_i)}{|A_i|} = \sum_{i=1}^{k} h_i^T L h_i = \text{Tr}(H^T L H).$$

- So the problem

$$\min_{\substack{V = A_1 \cup \cdots \cup A_k \\ A_i \cap A_j = \emptyset \ \forall i \neq j}} \text{RatioCut}(A_1, \ldots, A_k)$$

is equivalent to

$$\min_{H \in \mathbb{R}^{n \times k}} \text{Tr}(H^T L H)$$

$$\text{subject to } H^T H = I_{k \times k}, H \text{ defined as above.}$$

- As before, we consider a natural relaxation of the problem:

$$\min_{H \in \mathbb{R}^{n \times k}} \text{Tr}(H^T L H)$$

$$\text{subject to } H^T H = I_{k \times k}.$$

- Using the Rayleigh-Ritz theorem, we obtain that the solution of the problem

$$\min_{H \in \mathbb{R}^{n \times k}} \mathrm{Tr}(H^T L H)$$
$$\text{subject to } H^T H = I_{k \times k}.$$

is given by the matrix containing the first k (normalized) eigenvectors of L.

Relaxing RatioCut : $k > 2$

- Using the Rayleigh-Ritz theorem, we obtain that the solution of the problem

$$\min_{H \in \mathbb{R}^{n \times k}} \operatorname{Tr}(H^T L H)$$
$$\text{subject to } H^T H = I_{k \times k}.$$

is given by the matrix containing the first k (normalized) eigenvectors of L.
- How do we get the clusters?

Relaxing RatioCut : $k > 2$

- Using the Rayleigh-Ritz theorem, we obtain that the solution of the problem

$$\min_{H \in \mathbb{R}^{n \times k}} \operatorname{Tr}(H^T L H)$$
$$\text{subject to } H^T H = I_{k \times k}.$$

is given by the matrix containing the first k (normalized) eigenvectors of L.

- How do we get the clusters?
- Before the relaxation, the rows of the optimal H indicate to which cluster each vertex belongs to.

Relaxing RatioCut : $k > 2$

- Using the Rayleigh-Ritz theorem, we obtain that the solution of the problem

$$\min_{H \in \mathbb{R}^{n\times k}} \mathrm{Tr}(H^T L H)$$
$$\text{subject to } H^T H = I_{k\times k}.$$

is given by the matrix containing the first k (normalized) eigenvectors of L.
- How do we get the clusters?
- Before the relaxation, the rows of the optimal H indicate to which cluster each vertex belongs to.
- Similar to what we did when $k = 2$, we cluster the **rows** of the matrix H (containing the first k eigenvectors of L as columns) using the K-means algorithm.

The **unnormalized spectral clustering algorithm**:

Unnormalized spectral clustering

Input: Similarity matrix $S \in \mathbb{R}^{n \times n}$, number k of clusters to construct.

- Construct a similarity graph by one of the ways described in Section 2. Let W be its weighted adjacency matrix.
- Compute the unnormalized Laplacian L.
- **Compute the first k eigenvectors $u_1, \dots, u_k$ of L.**
- Let $U \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $u_1, \dots, u_k$ as columns.
- For $i = 1, \dots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the i-th row of U.
- Cluster the points $(y_i)_{i=1,\dots,n}$ in $\mathbb{R}^k$ with the k-means algorithm into clusters $C_1, \dots, C_k$.

Output: Clusters $A_1, \dots, A_k$ with $A_i = \{j \mid y_j \in C_i\}$.

Source: von Luxburg, 2007.

Normalized spectral clustering

- Relaxing the RatioCut leads to unnormalized spectral clustering.

Normalized spectral clustering

- Relaxing the RatioCut leads to unnormalized spectral clustering.
- By relaxing the Ncut problem, we obtain the **Normalized spectral clustering** algorithm of Shi and Malik (2000).

Normalized spectral clustering

- Relaxing the RatioCut leads to unnormalized spectral clustering.
- By relaxing the Ncut problem, we obtain the **Normalized spectral clustering** algorithm of Shi and Malik (2000).

Normalized spectral clustering according to Shi and Malik (2000)

Input: Similarity matrix $S \in \mathbb{R}^{n \times n}$, number k of clusters to construct.

- Construct a similarity graph by one of the ways described in Section 2. Let W be its weighted adjacency matrix.
- Compute the unnormalized Laplacian L.
- **Compute the first k generalized eigenvectors $u_1, \dots, u_k$ of the generalized eigenproblem $Lu = \lambda Du$.**
- Let $U \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $u_1, \dots, u_k$ as columns.
- For $i = 1, \dots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the i-th row of U.
- Cluster the points $(y_i)_{i=1,\dots,n}$ in $\mathbb{R}^k$ with the k-means algorithm into clusters $C_1, \dots, C_k$.

Output: Clusters $A_1, \dots, A_k$ with $A_i = \{j \mid y_j \in C_i\}$.

Source: von Luxburg, 2007.

Normalized spectral clustering

- Relaxing the RatioCut leads to unnormalized spectral clustering.
- By relaxing the Ncut problem, we obtain the **Normalized spectral clustering** algorithm of Shi and Malik (2000).

Normalized spectral clustering according to Shi and Malik (2000)

Input: Similarity matrix $S \in \mathbb{R}^{n \times n}$, number k of clusters to construct.

- Construct a similarity graph by one of the ways described in Section 2. Let W be its weighted adjacency matrix.
- Compute the unnormalized Laplacian L.
- **Compute the first k generalized eigenvectors $u_1, \dots, u_k$ of the generalized eigenproblem $Lu = \lambda Du$.**
- Let $U \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $u_1, \dots, u_k$ as columns.
- For $i = 1, \dots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the i-th row of U.
- Cluster the points $(y_i)_{i=1,\dots,n}$ in $\mathbb{R}^k$ with the k-means algorithm into clusters $C_1, \dots, C_k$.

Output: Clusters $A_1, \dots, A_k$ with $A_i = \{j \mid y_j \in C_i\}$.

Source: von Luxburg, 2007.

- Note: The solutions of $Lu = \lambda Du$ are the eigenvectors of L_{rw}.

See von Luxburg (2007) for details.

The normalized clustering algorithm of Ng et al.

- Another popular variant of the spectral clustering algorithm was provided by Ng, Jordan, and Weiss (2002).
- The algorithm uses L_{sym} instead of L (unnormalized clustering) or L_{rw} (Shi and Malik's normalized clustering).

The normalized clustering algorithm of Ng et al.

- Another popular variant of the spectral clustering algorithm was provided by Ng, Jordan, and Weiss (2002).
- The algorithm uses L_{sym} instead of L (unnormalized clustering) or L_{rw} (Shi and Malik's normalized clustering).

Normalized spectral clustering according to Ng, Jordan, and Weiss (2002)

Input: Similarity matrix $S \in \mathbb{R}^{n \times n}$, number k of clusters to construct.

- Construct a similarity graph by one of the ways described in Section 2. Let W be its weighted adjacency matrix.
- Compute the normalized Laplacian L_{sym}.
- **Compute the first k eigenvectors $u_1, \dots, u_k$ of L_{sym}.**
- Let $U \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $u_1, \dots, u_k$ as columns.
- **Form the matrix $T \in \mathbb{R}^{n \times k}$ from U by normalizing the rows to norm 1,** that is set $t_{ij} = u_{ij} / (\sum_k u_{ik}^2)^{1/2}$.
- For $i = 1, \dots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the i-th row of T.
- Cluster the points $(y_i)_{i=1,\dots,n}$ with the k-means algorithm into clusters $C_1, \dots, C_k$.

Output: Clusters $A_1, \dots, A_k$ with $A_i = \{j \mid y_j \in C_i\}$.

Source: von Luxburg, 2007.

See von Luxburg (2007) for details.