Spectral Clustering
Group Learning Presentation

Xiaowei Zhou

Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology

2011-05-6
Outline

1. Introduction
   - Overview
   - Graph Notations

2. Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3. Interpretation
   - Graph cut point of view
   - Random walk point of view

4. Discussion
What is the problem?

Clustering:

- Left case: k-means works.
- Right case: k-means doesn’t work.
Definition: Spectral methods refer to the use of eigenvalues, eigenvectors, singular values and singular vectors.

Applications:

- Community Detection
- Image Segmentation
- Speech Separation
1 Introduction
   - Overview
   - Graph Notations

2 Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3 Interpretation
   - Graph cut point of view
   - Random walk point of view

4 Discussion
Here we consider an undirected graph $G = (V, E)$, where $V = \{v_1, \cdots, v_n\}$ is the vertex set and $E = \{e_{ij}\}$ is the edge set.

The degree $d_i$ of vertex $i$ is the number of edges connected to $i$.

Two ways of measuring size of $G$ are considered:

1. $|G| := \text{the number of vertices in } G$
2. $\text{vol}(G) := \sum_{i \in V} d_i$
Adjacency Matrix

For a graph with $n$ vertices, the adjacency matrix is a $|V| \times |V|$ matrix defined as:

$$
A := \begin{cases} 
A_{ij} = 1 & \text{if there is an edge } e_{ij} \\
A_{ij} = 0 & \text{if there is no edge} \\
A_{ii} = 0 
\end{cases}
$$

$$
A = \begin{bmatrix}
0 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 \\
1 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 \\
\end{bmatrix}
$$

(adjacency matrix example)
The incidence matrix of a graph is a $|E| \times |V|$ matrix defined as follows:

$$\nabla := \begin{cases} 
\nabla_{ev} = -1 & \text{if } v \text{ is the initial vertex of edge } e \\
\nabla_{ev} = 1 & \text{if } v \text{ is the terminal vertex of edge } e \\
\nabla_{ev} = 0 & \text{if } v \text{ is not in } e
\end{cases}$$

$$\nabla = \begin{bmatrix}
-1 & 1 & 0 & 0 \\
1 & 0 & -1 & 0 \\
0 & -1 & 1 & 0 \\
0 & -1 & 0 & +1
\end{bmatrix}$$

The incidence matrix is used as a gradient operator:

$$\nabla f(e_{ij}) = f_j - f_i$$
Laplace Matrix

The Laplacian matrix of a graph is a \(|V| \times |V|\) matrix:

1. \( L = D - A \), where \( D = \text{diag}(d_1, \cdots, d_n) \) is the degree matrix.
2. \( L = \nabla^T \nabla \)

\[
\begin{bmatrix}
2 & -1 & -1 & 0 \\
-1 & 3 & -1 & -1 \\
-1 & -1 & 2 & 0 \\
0 & -1 & 0 & 1 \\
\end{bmatrix}
\]
To use the graph to model both neighborship and similarity between samples, we assign a weight $w_{ij}$ to each edge $e_{ij}$.

A common similarity measure:

$$w_{ij} = e^{\frac{||v_i - v_j||^2}{\sigma}}$$

$A$, $D$, $L$ are modified to weighted version:

- $A = W := [w_{ij}]_{|V| \times |V|}$ (if there is no edge between  $i, j$)
- $D = diag(d_1, \cdots, d_n)$ where $d_{ij} = \sum_j w_{ij}$
- $L = D - A = \nabla^T W \nabla$
Construct Similarity Graphs

- *The $\epsilon$-neighborhood graph*: connect all points whose pairwise distances are smaller than $\epsilon$.
- *$k$-nearest neighbor graphs*: connect each vertex to its $k$ nearest neighbors.
- *The fully connected graph*: connect all points with each other.
Outline

1. Introduction
   - Overview
   - Graph Notations

2. Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3. Interpretation
   - Graph cut point of view
   - Random walk point of view

4. Discussion
The laplacian matrix of a graph has following properties:

1. For every vector \( f \in \mathbb{R}^n \):
   \[
   f^T L f = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2
   \]

2. \( L \) is positive semi-definite.

3. The smallest eigenvalue of \( L \) is 0, the corresponding eigenvector is the constant one vector \( \mathbf{1} \).

4. \( L \) has \( n \) non-negative eigenvalues \( 0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n \).
Number of Connected Components

Suppose \( G = \bigcup_{i=1}^{k} A_i \), where \( A_1, \cdots, A_k \) are isolated connected subgraphs. Then, the multiplicity of the eigenvalue 0 of \( L \) equals the number of connected components \( k \). The eigenspace of eigenvalue 0 is spanned by the indicator vector \( 1_{A_1}, \cdots, 1_{A_k} \).

Proof.

Two key points:

1. When \( k = 1 \), the multiplicity of the eigenvalue 0 is exactly 1.
2. When \( k > 1 \), \( L \) is a block diagonal matrix:

\[
L = \begin{bmatrix}
L_1 & & \\
& \ddots & \\
& & L_k
\end{bmatrix}
\]

The spectrum of \( L \) is given by the union of the spectra of \( L_i \).
A toy example

Histogram of the sample

Eigenvalues

Eigenvector 1  Eigenvector 2  Eigenvector 3  Eigenvector 4  Eigenvector 5
The Fiedler vector

- The second smallest eigenvalue $\lambda_2$ of $L$ is called the Fiedler value of a graph. It measures the connectivity of a graph: the further from 0, the more connected.
- The eigenvector $u_2$ corresponding to $\lambda_2$ is called the Fiedler vector of a graph, which is important for graph Bi-partition.
- $u_2$ is the solution for the following minimization:

$$\min_f f^T L f$$

s.t. $f^T f = 1, f^T 1 = 0$
1. Introduction
   - Overview
   - Graph Notations

2. Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3. Interpretation
   - Graph cut point of view
   - Random walk point of view

4. Discussion
Spectral Clustering Algorithm

The algorithm flow:

- **Input:** a data set \( x_1, \ldots, x_n \) where \( x_i \in \mathbb{R}^l \)
- 1. Construct a similarity graph.
- 2. \( L = D - A \)
- 3. Compute the first \( k \) eigenvectors \( u_1, \ldots, u_k \) of \( L \)
- 4. Let \( Y \in \mathbb{R}^{n \times k} \) be the matrix composed of \( u_1, \ldots, u_k \) as columns. Let \( y_i \) be the vector corresponding to the \( i \)-th row of \( Y \).
- 5. Cluster the points \( y_i \) with \( k \)-means algorithm into clusters \( C_1, \ldots, C_k \).

- **Output** the clusters \( A_1, \ldots, A_k \) with \( A_i = \{ j \mid y_j \in C_i \} \)
Spectral clustering: Dimension reduction using graph Laplacian followed by typical clustering.

- **Dimension Reduction**: Given a set $x_1, \cdots, x_n$ of $n$ points $\mathbb{R}^l$, find a set of points $f_1, \cdots, f_n$ in $\mathbb{R}^k$ ($k \ll l$) such that $f_i$ best represents $x_i$.

- **Laplacian Eigenmaps (LE)**: Dimensional reduction using graph Laplacian.

Find $F' = \begin{bmatrix} f'_1 & \cdots & f'_n \end{bmatrix}$ by: $\min_F \sum_{i,j} W_{ij}\|f_i - f_j\|_2^2$, s.t. $F'F = I$

where $W_{ij}$ is the weight of edges in the adjacency graph.

M. Belkin, P. Niyogi
Laplacian Eigenmaps for Dimensionality Reduction and Data Representation
*Neural computation 2003*
Can we unroll and present the data in a low-dimensional space?
The Laplacian Eigenmaps can map the points in a nonlinear manifold in \( \mathbb{R}^l \) to the points in \( \mathbb{R}^k \), and it preserves local information optimally in certain sense.

Construction of adjacency graph is critical.
Outline

1. Introduction
   - Overview
   - Graph Notations

2. Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3. Interpretation
   - Graph cut point of view
   - Random walk point of view

4. Discussion
Graph Cut

- Cut:
  \[
  \text{cut}(A, \overline{A}) = \frac{1}{2} \sum_{i \in A, j \in \overline{A}} w_{ij}
  \]  

- Min Cut:
  \[
  \text{MCut}(A_1, \cdots, A_k) = \sum_{i=1}^{k} \text{cut}(A_i, \overline{A_i})
  \]

- Ratio Cut
  \[
  \text{RCut}(A, \overline{A}) = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A_i})}{|A_i|}
  \]

- Normalized Cut
  \[
  \text{NCut}(A, \overline{A}) = \sum_{i=1}^{k} \frac{\text{cut}(A_i, \overline{A_i})}{\text{vol}(A_i)}
  \]
Min Cut \( (k = 2) \)

- Define a vector \( f = (f_1, \cdots, f_n)^T \in \mathbb{R}^n \) to indicate the belonging of vertices:

\[
f_i = \begin{cases} 
1 & \text{if } v_i \in A \\
-1 & \text{if } v_i \in \overline{A}
\end{cases}
\]

- The Min Cut can be done by minimizing:

\[
\text{MCut}(A, \overline{A}) = \frac{1}{8} \sum_{i,j} w_{ij} (f_i - f_j)^2 = \frac{1}{4} f^T L f
\]

- Any eigenvector of \( L \) corresponding to eigenvalue 0 can minimize above energy. Usually trivial solutions are obtained.
Ratio Cut \( (k = 2) \)

- The cost of Ratio Cut is:

\[
\text{RatioCut}(A, \overline{A}) = \frac{\text{cut}(A, \overline{A})}{|A|} + \frac{\text{cut}(A, \overline{A})}{|\overline{A}|} \tag{5}
\]

- Define the indicator vector as:

\[
f_i = \begin{cases} 
\sqrt{|\overline{A}|/|A|} & \text{if } v_i \in A \\
-\sqrt{|A|/|\overline{A}|} & \text{if } v_i \in \overline{A}
\end{cases} \tag{6}
\]
Ratio Cut ($k = 2$)

$$f'Lf = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2$$

$$= \frac{1}{2} \sum_{i \in A, j \in \overline{A}} w_{ij} \left( \sqrt{\frac{|A|}{|A|}} + \sqrt{\frac{|A|}{|A|}} \right)^2 + \frac{1}{2} \sum_{i \in \overline{A}, j \in A} w_{ij} \left( -\sqrt{\frac{|A|}{|A|}} - \sqrt{\frac{|A|}{|A|}} \right)^2$$

$$= \text{cut}(A, \overline{A}) \left( \frac{|A|}{|A|} + \frac{|A|}{|A|} + 2 \right)$$

$$= \text{cut}(A, \overline{A}) \left( \frac{|A| + |A|}{|A|} + \frac{|A| + |A|}{|A|} \right)$$

$$= |V| \cdot \text{RatioCut}(A, \overline{A}).$$
Relaxing $f$ to be $\mathbb{R}^n$, in Ratio Cut, we are also minimizing $f^T L f$.

But now we have more constraints from Eq. 9:

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

$$\|f\|^2 = \sum_{i=1}^{n} f_i^2 = |A| \left( \frac{|A|}{|A|} \right) + |\overline{A}| \left( \frac{|A|}{|A|} \right) = |\overline{A}| + |A| = n.$$ 

Thus, we can solve Ratio Cut by:

$$\min_{f} f^T L f \quad \text{s.t.} \quad f \perp 1, f^T f = n \quad (7)$$

The solution $f^*$ is the Fiedler vector of $L$. 

Xiaowei Zhou (HKUST) Spectral Clustering Group Learning 28 / 37
The cost of Normalized Cut is:

\[
\text{NCut}(A, \overline{A}) = \frac{\text{cut}(A, \overline{A})}{\text{vol}(A)} + \frac{\text{cut}(A, \overline{A})}{\text{vol}(\overline{A})}
\] (8)

Define the indicator vector as:

\[
f_i = \begin{cases} 
\sqrt{\frac{\text{vol}(\overline{A})}{\text{vol}(A)}} & \text{if } v_i \in A \\
-\sqrt{\frac{\text{vol}(A)}{\text{vol}(\overline{A})}} & \text{if } v_i \in \overline{A} 
\end{cases}
\] (9)
Similarly we can rewrite Normalized Cut as:

$$\min_{f} \quad f'Lf \quad \text{s.t.} \quad Df \perp 1, \quad f'Df = \text{vol}(V).$$ \hspace{1cm} (10)

Substitute $g = D^{\frac{1}{2}}f$, the problem terms to be:

$$\min_{g} \quad g'D^{-\frac{1}{2}}LD^{-\frac{1}{2}}g \quad \text{s.t.} \quad g \perp D^{\frac{1}{2}}1, \quad g'g = \text{vol}(V).$$ \hspace{1cm} (11)

The solution $g^*$ is the Fiedler vector of $D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$.

Substitute $f = D^{-\frac{1}{2}}g$ back, we can see $f^*$ is the Fiedler vector of $D^{-1}L$. 

General Case: $k \neq 2$

Given a partition of $V$ into $k$ sets $A_1, \cdots, A_k$, we define $k$ indicator vectors $f_1, \cdots, f_k$ where $f_j$ is the indicator vector for $A_j$. The values of the indicator vectors are defined in the table. Then, the cut problems can be viewed as following minimization problems:

<table>
<thead>
<tr>
<th></th>
<th>indicator vectors</th>
<th>obj. function</th>
<th>constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mcut</td>
<td>$f_{ij} = \begin{cases} 1 &amp; \text{if } v_i \in A_j \ 0 &amp; \text{otherwise} \end{cases}$</td>
<td>$\text{Tr}(F'LF)$</td>
<td></td>
</tr>
<tr>
<td>Rcut</td>
<td>$f_{ij} = \begin{cases} 1/\sqrt{</td>
<td>A_j</td>
<td>} &amp; \text{if } v_i \in A_j \ 0 &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>Ncut</td>
<td>$f_{ij} = \begin{cases} 1/\sqrt{\text{vol}(A_j)} &amp; \text{if } v_i \in A_j \ 0 &amp; \text{otherwise} \end{cases}$</td>
<td>$\text{Tr}(F'LF)$</td>
<td>$F'DF = I$</td>
</tr>
</tbody>
</table>
Outline

1. Introduction
   - Overview
   - Graph Notations

2. Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3. Interpretation
   - Graph cut point of view
   - Random walk point of view

4. Discussion
Consider a random walk on graph $G$. The transition probability of jumping in one step from vertex $v_i$ to vertex $v_j$ is proportional to the edge weight $w_{ij}$ and is given by $p_{ij} := w_{ij}/d_i$. The transition matrix $P = (p_{ij})_{i,j=1,...,n}$ of the random walk turns out to be the normalized adjacency matrix:

$$P = D^{-1}W.$$ 

If the graph is connected and non-bipartite, then the random walk always possesses a unique stationary distribution $\pi = (\pi_1, \cdots, \pi_n)$, where $\pi_i = d_i/vol(V)$. 
If $P(B|A)$ denotes the probability we start random walk from a vertex in subset $A$ and transit to a vertex in subset $B$, then:

$$P(\overline{A}|A) + P(A|\overline{A}) = \text{Ncut}(A, \overline{A}).$$  \hspace{1cm} (12)

**Proof:**

$$P(B|A) = \frac{P(B, A)}{P(A)} = \frac{\sum_{i \in A, j \in B} \pi_i p_{ij}}{\sum_{i \in A} \pi_i}$$

$$= \frac{\sum_{i \in A, j \in B} \frac{d_i}{\text{vol}(V)} w_{ij} d_i}{\sum_{i \in A} \frac{d_i}{\text{vol}(V)}} = \frac{\sum_{i \in A, j \in B} w_{ij}}{\sum_{i \in A} d_i} = \frac{\sum_{i \in A, j \in B} w_{ij}}{\text{vol}(A)}$$

$$P(\overline{A}|A) + P(A|\overline{A}) = \frac{\sum_{i \in A, j \in \overline{A}} w_{ij}}{\text{vol}(A)} + \frac{\sum_{i \in A, j \in \overline{A}} w_{ij}}{\text{vol}(\overline{A})}.$$
Outline

1. Introduction
   - Overview
   - Graph Notations

2. Spectral Clustering
   - Graph Laplacian
   - Spectral Clustering

3. Interpretation
   - Graph cut point of view
   - Random walk point of view

4. Discussion
Why Use Spectral Clustering?

- Only the pairwise similarity is needed. Thus, it can handle complex data with missing values or hybrid data type.
- No assumption on the data distribution. Suitable for explore complex structures like manifold learning or community detection.
- Deterministic solution.
Following issues are critical for spectral clustering:

1. Parameters of constructing similarity graphs.
2. The number of clusters.
3. Computational efficiency.