Spectral Clustering

Seungjin Choi

Department of Computer Science
POSTECH, Korea
seungjin@postech.ac.kr

1

Spectral Clustering Algorithm: Bipartioning

1. Construct affinity matrix

\[ W_{ij} = \begin{cases} \exp\{-\beta \|v_i - v_j\|^2\} & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \]

2. Calculate the graph Laplacian \( L = D - W \) where \( D = \text{diag}\{d_1, \ldots, d_n\} \) and \( d_i = \sum_j W_{ij} \).

3. Compute the second smallest eigenvector of the graph Laplacian (denoted by \( u = [u_1 \cdots u_n]^T \), Fiedler vector)

4. Partition \( u_i \)'s by a pre-specified threshold value and assign data points \( v_i \) to cluster.

Two Moons Data

![Two Moons Data Plot](image)
Graphs

- Consider a connected graph $G(\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{v_1, \ldots, v_n\}$ and $\mathcal{E}$ denote a set of vertices and a set of edges, respectively, with pairwise similarity values being assigned as edge weights.

- **Adjacency matrix** (similarity, proximity, affinity matrix): $W = [W_{ij}] \in \mathbb{R}^{n \times n}$.

- Degree of nodes: $d_i = \sum_j W_{ij}$.

- Volume: $\text{vol}(S_i) = d_{S_i} = \sum_{i \in S_i} d_i$. 
Neighborhood Graphs

Gaussian similarity function is given by

\[ w(v_i, v_j) = W_{ij} = \exp \left\{ -\frac{||v_i - v_j||^2}{2\sigma^2} \right\} \].

- \( \epsilon \)-neighborhood graph
- \( k \)-nearest neighbor graph

Graph Laplacian

(Unnormalized) graph Laplacian is defined as \( L = D - W \).

1. For every vector \( x \in \mathbb{R}^n \), we have

\[ x^\top Lx = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} W_{ij} (x_i - x_j)^2 \geq 0. \] (positive semidefinite)

2. The smallest eigenvalue of \( L \) is 0 and the corresponding eigenvector is \( 1 = [1 \cdots 1]^\top \), since \( D1 = W1 \), i.e., \( L1 = 0 \).

3. \( L \) has \( n \) nonnegative eigenvalues, \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n = 0 \).

Normalized Graph Laplacian

Two different normalization methods are popular, including:

- Symmetric normalization:

\[ L_s = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}. \]

- Normalization related to random walks:

\[ L_{rw} = D^{-1} L = I - D^{-1} W. \]
1. For every vector \( x \in \mathbb{R}^n \), we have
\[
x^\top L x = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} W_{ij} \left( \frac{x_i}{\sqrt{d_i}} - \frac{x_j}{\sqrt{d_j}} \right)^2.
\]

2. \( L_{sym} \) and \( L_{rw} \) are positive semidefinite and have \( n \) nonnegative real-valued eigenvalues, \( \lambda_1 \geq \cdots \geq \lambda_n = 0 \).

3. \( \lambda \) is an eigenvalue of \( L_{rw} \) with eigenvector \( u \) if and only if \( \lambda \) is an eigenvalue of \( L_s \) with eigenvector \( D^{1/2}u \).

4. \( \lambda \) is an eigenvalue of \( L_{rw} \) with eigenvector \( u \) if and only if \( \lambda \) and \( u \) solves the generalized eigenvalue problem \( Lu = \lambda Du \).

5. 0 is an eigenvalue of \( L_{rw} \) with the constant one vector \( 1 \) as eigenvector. 0 is an eigenvalue of \( L_s \) with eigenvector \( D^{1/2}1 \).

### Normalized Spectral Clustering: Shi-Malik

1. Construct a neighborhood graph with corresponding adjacency matrix \( W \).

2. Compute the unnormalized graph Laplacian \( L = D - W \).

3. Find the \( k \) smallest eigenvectors of \( L \) and form the matrix \( U = [u_1 \cdots u_k] \in \mathbb{R}^{n \times k} \).

4. Treating each row of \( U \) as a point in \( \mathbb{R}^k \), cluster them into \( k \) groups using \( k \)-means algorithm.

5. Assign \( v_i \) to cluster \( j \) if and only if row \( i \) of \( U \) is assigned to cluster \( j \).

### Normalized Spectral Clustering: Ng-Jordan-Weiss

1. Construct a neighborhood graph with corresponding adjacency matrix \( W \).

2. Compute the normalized graph Laplacian \( L_s = D^{-1/2}LD^{-1/2} \).

3. Find the \( k \) smallest eigenvectors \( u_1, \ldots, u_k \) of \( L_s \) and form the matrix \( U = [u_1 \cdots u_k] \in \mathbb{R}^{n \times k} \).

4. Form the matrix \( \tilde{U} \) from \( U \) by re-normalizing each row of \( U \) to have unit norm, i.e., \( \tilde{U}_{ij} = U_{ij} / (\sum_j U_{ij})^{1/2} \).

5. Treating each row of \( \tilde{U} \) as a point in \( \mathbb{R}^k \), cluster them into \( k \) groups using \( k \)-means algorithm.

6. Assign \( v_i \) to cluster \( j \) if and only if row \( i \) of \( \tilde{U} \) is assigned to cluster \( j \).
Where does this spectral clustering algorithm come from?

• Spectral graph partitioning
• Properties of block (diagonal) matrix
• Markov random walk

Graph Partitioning: Bipartitioning

• Consider a connected graph $G(V, E)$ where $V = \{v_1, \ldots, v_n\}$ and $E$ denote a set of vertices and a set of edges, respectively, with pairwise similarity values being assigned as edge weights.

• Graph bipartitioning involves taking the set $V$ apart into two coherent groups, $S_1$ and $S_2$, satisfying $V = S_1 \cup S_2$, ($|V| = n$), and $S_1 \cap S_2 = \emptyset$, by simply cutting edges connecting the two parts

• Adjacency matrix (similarity, proximity, affinity matrix): $W = [W_{ij}] \in \mathbb{R}^{n \times n}$.

• Degree of nodes: $d_i = \sum_j W_{ij}$.

• Volume: $\text{vol}(S_1) = d_{S_1} = \sum_{i \in S_1} d_i$. 

Pictorial Illustration: Cut and Volume
Graph Partitioning

The task is to find \( k \) disjoint sets, \( S_1, \ldots, S_k \), given \( G = (V, E) \), where \( S_1 \cap \cdots \cap S_k = \emptyset \) and \( S_1 \cup \cdots \cup S_k = V \) such that a certain cut criterion is minimized.

1. Bipartitioning: \( \text{cut}(S_1, S_2) = \sum_{i \in S_1} \sum_{j \in S_2} W_{ij} \).

2. Multiway partitioning: \( \text{cut}(S_1, \ldots, S_k) = \sum_{i=1}^{k} \text{cut}(S_i, \bar{S}_i) \).

3. Ratio cut: \( R_{\text{cut}}(S_1, \ldots, S_k) = \sum_{i=1}^{k} \frac{\text{cut}(S_i, \bar{S}_i)}{|S_i|} \).

4. Normalized cut: \( N_{\text{cut}}(S_1, \ldots, S_k) = \sum_{i=1}^{k} \frac{\text{cut}(S_i, \bar{S}_i)}{\text{vol}(S_i)} \).

Cut: Bipartitioning

The degree of dissimilarity between \( S_1 \) and \( S_2 \) can be computed by the total weights of edges that have been removed.

\[
\text{cut}(S_1, S_2) = \sum_{i \in S_1} \sum_{j \in S_2} W_{ij}.
\]

Introducing bipolar indicator vector, \( x = q_1 - q_2 \in \{+1, -1\}^n \), the cut criterion is simplified as

\[
\text{Cut}(S_1, S_2) = \frac{1}{4} x^\top L x.
\]

The balanced cut involves the following combinatorial optimization problem

\[
\arg \min x^\top L x \quad \text{subject to } 1^\top x = 0, \quad x \in \{1, -1\}.
\]

Dropping the integer constrains (spectral relaxation), leads to the symmetric eigenvalue problem. The second smallest eigenvector of \( L \) corresponds to the solution, since the smallest eigenvalue of \( L \) is 0 and its associated eigenvector is \( 1 \). The second smallest eigenvector is known as Fiedler vector.

Rcut and Unnormalized Spectral Clustering: \( k = 2 \)

Define the indicator vector \( x = [x_1 \cdots x_n]^\top \) with entries

\[
x_i = \begin{cases} 
\sqrt{|S_i|/|S|} & \text{if } v_i \in S \\
-\sqrt{|S_i|/|\bar{S}|} & \text{if } v_i \in \bar{S}.
\end{cases}
\]

Then one can easily see that

\[
x^\top L x = 2|V|R_{\text{cut}}(S, \bar{S}),
\]

\[
x^\top 1 = 0,
\]

\[
||x|| = \sqrt{n}.
\]
\[
\arg \min_{S \subseteq V} \text{Rcut}(S, \overline{S}) \equiv \arg \min_{S \subseteq V} x^\top L x,
\]
subject to \(x^\top 1 = 0\),
x \(i\) is defined in previous slide,
\(\|x\| = \sqrt{n}\).

The relaxation by discarding the condition on the discrete values on \(x_i\) and instead allowing \(x_i \in \mathbb{R}\), leads to
\[
\arg \min_{x \in \mathbb{R}^n} x^\top L x, \quad \text{subject to } x^\top 1 = 0 \text{ and } \|x\| = \sqrt{n}.
\]

The eigenvector associated with the second smallest eigenvalue of \(L\). Rounding the eigenvector gives an approximate solution to the ratio cut problem.

25

Rcut and Unnormalized Spectral Clustering: \(k > 2\)

Define the indicator matrix \(X = [x_1 \cdots x_k] \in \mathbb{R}^{n \times k}\) and \(x_i = [x_{1,i} \cdots x_{n,i}]^\top \in \mathbb{R}^n\) with entries
\[
x_{i,j} = \begin{cases} 
1/\sqrt{|S_j|} & \text{if } v_i \in S_j \\
0 & \text{if } v_i \in \overline{S}_j.
\end{cases}
\]

Then we have
\[
x_i^\top L x_i = 2 \frac{\text{cut}(S_i, \overline{S}_i)}{|S_i|}, \quad X^\top X = I.
\]

Then the relaxed problem becomes
\[
\arg \min_{X \in \mathbb{R}^{n \times k}} \text{tr} \{ X^\top L X \} \quad \text{subject to } X^\top X = I.
\]

26

Ncut and Normalized Spectral Clustering: \(k = 2\)

Define the indicator vector \(x = [x_1 \cdots x_n]^\top\) with entries
\[
x_i = \begin{cases} 
\sqrt{\text{vol}(S)/\text{vol}(\overline{S})} & \text{if } v_i \in S \\
-\sqrt{\text{vol}(\overline{S})/\text{vol}(S)} & \text{if } v_i \in \overline{S}.
\end{cases}
\]

Then one can easily see that
\[
x^\top L x = 2|\mathcal{V}| \text{Ncut}(S, \overline{S}),
\]
\([Dx]^\top 1 = 0,\]
\(x^\top D x = \text{vol}(\mathcal{V}).\)
\[ \arg \min_{S \subset V} \text{Ncut}(S, \overline{S}) \equiv \arg \min_{S \subset V} x^\top L x, \]
subject to \((D x)^\top 1 = 0,
\]
x is defined in previous slide,
\[ x^\top D x = \text{vol}(\mathcal{V}). \]

Relaxing the problem gives
\[ \arg \min_{x \in \mathbb{R}^n} x^\top L x, \quad \text{subject to} \ (D x)^\top 1 = 0 \text{ and } x^\top D x = \text{vol}(\mathcal{V}). \]

Define \( y = D^{1/2} x \), then the problem is
\[ \arg \min_{y \in \mathbb{R}^n} y^\top D^{-1/2} L D^{-1/2} y, \quad \text{subject to} \ y^\top D^{1/2} 1 = 0 \text{ and } \|y\|^2 = \text{vol}(\mathcal{V}). \]

The solution \( y \) is given by the 2nd smallest eigenvector of \( L_s \), implying that \( x \) is the 2nd smallest eigenvector of \( L_{rw} \), or equivalently the 2nd smallest generalized eigenvector of \( Lu = \lambda Du \).

\[ \text{Ncut and Normalized Spectral Clustering: } k > 2 \]

Define the indicator matrix \( X = [x_1 \cdots x_k] \in \mathbb{R}^{n \times k} \) and \( x_i = [x_{1,i} \cdots x_{n,i}]^\top \in \mathbb{R}^n \) with entries
\[ x_{i,j} = \begin{cases} 1/\sqrt{\text{vol}(S_j)} & \text{if } v_i \in S_j \\ 0 & \text{if } v_i \in \overline{S}_j. \end{cases} \]

Then we have
\[ x_i^\top L x_i = 2 \frac{\text{cut}(S_i, \overline{S}_i)}{\text{vol}(S_i)}, \quad X^\top X = I, \quad x_i^\top D x_i = 1, \]

\[ \text{Ncut}(S_1, \ldots, S_k) = \frac{1}{2} \sum_{i=1}^k x_i^\top L x_i = \frac{1}{2} \text{tr} \left\{ X^\top LX \right\}. \]

\[ \text{Markov Random Walk View of Normalized Cut} \]

- Melia and Shi 2001
- Probabilistic interpretation of normalized cut
- Data points are clustered on the basis of the eigenvectors of the resulting transition probability matrix (constructed by the weights)
Transition Probability Matrix

- We define a Markov random walk over the graph by constructing a transition probability matrix from the edge weights
  \[ P_{ij} = \frac{W_{ij}}{\sum_j W_{ij}}, \]
  where \( \sum_j P_{ij} = 1 \) for all \( i \).
- The random walk proceeds by successively selecting points according to \( j \sim P_{ij} \) where \( i \) specifies the current location.
- If the graph is connected and non-bipartite (ergodic Markov chain), then the random walk always possesses a unique stationary distribution \( \pi = [\pi_1 \cdots \pi_n]^T \) such that \( \pi^T P = \pi^T \), which is given by \( \pi_i = d_i / \text{vol}(G) \).

Random Walk: Properties

If we start from \( i_0 \), the distribution of points \( i_t \) that we end up with after \( t \) steps, is given by

\[
\begin{align*}
    i_1 & \sim P_{i_0i_1}, \\
    i_2 & \sim \sum_{i_1} P_{i_0i_1} P_{i_1i_2} = [P^2]_{i_0i_2}, \\
    i_3 & \sim \sum_{i_1} \sum_{i_2} P_{i_0i_1} P_{i_1i_2} P_{i_2i_3} = [P^3]_{i_0i_3}, \\
    & \vdots \\
    i_t & \sim [P^t]_{i_0i_t},
\end{align*}
\]

where \( P^t = PP \cdots P \) and \([.]_{ij}\) denotes the \( i, j \) component of the matrix. The distribution of points that we end up with in after \( t \) random steps converges as \( t \) increases.

Ncut via Transition Probabilities

Define \( P(S \mid \overline{S}) = P(X_{t+1} \in S \mid X_t \in \overline{S}) \) where \( X_t \) is a state of Markov random walk model at time \( t \). The Ncut and Markov random walk model has the following relation:

\[ \text{Ncut}(S, \overline{S}) = P(S \mid S) + P(S \mid \overline{S}). \]

The minimization of Ncut actually seeks a cut through the graph such that a random walk seldom transitions from \( S \) to \( \overline{S} \) or vice versa.

Stochastic Matrix

The stochastic matrix \( P \) is defined by \( P = D^{-1}W \). To find out how \( P^t \) behaves for large \( t \), it is useful to examine the eigen-decomposition of the following symmetric matrix

\[ D^{-\frac{1}{2}}WD^{-\frac{1}{2}} = \lambda_1 u_1 u_1^T + \lambda_2 u_2 u_2^T + \cdots + \lambda_n u_n u_n^T. \]

The symmetric matrix is related to \( P^t \) since

\[ (D^{-\frac{1}{2}}WD^{-\frac{1}{2}}) \cdots (D^{-\frac{1}{2}}WD^{-\frac{1}{2}}) = D^{\frac{1}{2}} (P \cdots P) D^{-\frac{1}{2}}. \]

This allows us to write the \( t \) step transition probability matrix in terms of the eigenvalues/vectors of the symmetric matrix

\[
\begin{align*}
P^t &= D^{-\frac{1}{2}} \left( D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \right)^t D^{\frac{1}{2}} \\
    &= D^{-\frac{1}{2}} \left( \lambda_1 u_1 u_1^T + \lambda_2 u_2 u_2^T + \cdots + \lambda_n u_n u_n^T \right) D^{\frac{1}{2}},
\end{align*}
\]

where \( \lambda_1 = 1 \) and \( P^\infty = D^{-\frac{1}{2}} \left( u_1 u_1^T \right) D^{\frac{1}{2}}. \)
Spectral Clustering: Stochastic Matrix

- We are interested in the largest correction to the asymptotic limit
  \[ P^t \approx P^\infty + D^{-\frac{1}{2}} \left( \lambda_t u_2 u_2^\top \right) D^{\frac{1}{2}}. \]

- Note that \( [u_2 u_2^\top]_{ij} = u_{i2} u_{j2} \) and thus the largest correction term increases the probability of transitions between points that share the same sign of \( u_{i2} \) and decreases transitions across points with different signs.

- Binary spectral clustering: Divide the points into clusters based on the sign of the elements of \( u_2 \)
  \[ u_{j2} > 0 \Rightarrow \text{cluster 1}, \quad \text{otherwise cluster 0}. \]

Equivalence

**Proposition 1.** If \( \lambda, x \) are solutions of \( P x = \lambda x \) and \( P = D^{-1} W \), then \( (1 - \lambda), x \) are solutions of \( L x = \lambda D x \).

This proposition shows the equivalence between the spectral clustering formulated by the normalized cut and the eigenvalue/vectors of the stochastic matrix \( P \).

The largest eigenvector of \( P \) is \( 1 \) containing no information. The second smallest eigenvector in the normalized cut, corresponds to the second largest eigenvector of the stochastic matrix.

Suggested Further Readings