Markov Chains and Spectral Clustering

Ning Liu and William J. Stewart
Department of Computer Science
North Carolina State University
Raleigh, NC 27695–8206

A presentation in honor of Dr. Guenter Haring for having attained that state of grace to which we all aspire
— Professor Emeritus.

Network Science

- Network science application areas
  - — rankings of soccer teams and players
  - — grocery placement
  - — spheres of influence, etc.
  - — social networking

- Markov chains and spectral clustering
  - — Google/Yahoo!
  - — Reverse engineering problem,

  i.e., draw the Markov chain from the eigenvectors

Markov Chains and Spectral Clustering

Miroslav Fiedler "Algebraic Connectivity of Graphs, 1973"
— developed a spectral partitioning method to obtain the minimum cut on an undirected graph (symmetric system).

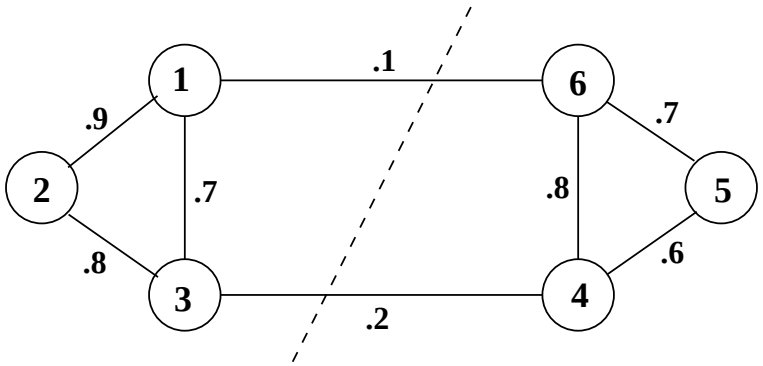
The vector that results from the spectral decomposition — the Fiedler vector — allows the nodes to be partitioned into two subsets.

Spectral decomposition is applied to the Laplacian matrix.

Alternative approach based on the dominant eigenvectors of a Markov chain — and more broadly applicable to non-symmetric systems.

Spectral Graph Partitioning

- Graph partitioning/clustering:
  — group the vertices of a connected graph into disjoint partitions.
- Objective:
  — minimize the total cut weight
  — maximize group cohesion.



The Laplacian Matrix and Minimum Cut:

$$L_{ij} = \begin{cases} \sum_k w_{ik}, & \text{if } i = j \\ -w_{ij}, & \text{if } i \neq j,\ i \text{ and } j \text{ are adjacent} \\ 0, & \text{otherwise,} \end{cases}$$

1. L is symmetric, positive semi-definite. As such its eigenvalues are all real and non-negative. Furthermore the eigenvectors of L constitute a full set of n real and orthogonal vectors.

2. $Le = 0$, where e is a column vector whose elements are all equal to 1. Thus 0 is the smallest eigenvalue of L and e is its corresponding eigenvector.

3. For any vector x, we have

$$x^T L x = \sum_{\{i,j\} \in E} w_{ij}(x_i - x_j)^2. \qquad (1)$$

Given a partition of V into V_1 and V_2, a partition vector p is defined as

$$p_i = \begin{cases} +1, & \text{vertex } i \in V_1, \\ -1, & \text{vertex } i \in V_2. \end{cases} \tag{2}$$

Observe that $p^T p = n$. Also, from Equation (1),

$$p^T L p = \sum_{\{i,j\} \in E} w_{ij}(p_i - p_j)^2.$$

— edge weights within V_1 and V_2 are not counted,

— edges connecting partitions are multiplied by 4.

Note: $\text{cut}(V_1, V_2) = \sum_{i \in V_1, j \in V2} w_{ij}$; $\quad p^T L p = 4\,\text{cut}(V_1, V_2)$. Hence

Raleigh Quotient:

$$\frac{p^T L p}{p^T p} = \frac{1}{n} \cdot 4\,\text{cut}(V_1, V_2). \tag{3}$$

Maximum and minimum of the Rayleigh quotient can be obtained as the largest and smallest eigenvalues of the Laplacian matrix L:

$$\lambda_{max} = \max_{x \neq 0} \frac{x^T L x}{x^T x} \quad \text{and} \quad \lambda_{min} = \min_{x \neq 0} \frac{x^T L x}{x^T x}, \tag{4}$$

where x is the eigenvector of L corresponding to λ_{max} and λ_{min}.

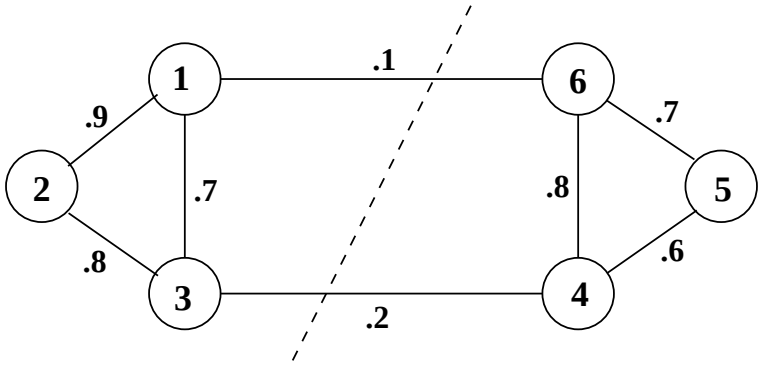
Minimum value of the Rayleigh quotient is zero
— the smallest eigenvalue of L corresponding to the eigenvector e.

All the vertices of the graph are in the same set
— the trivial partition.

The second smallest eigenvalue of L, the Fiedler value,
provides the optimal value

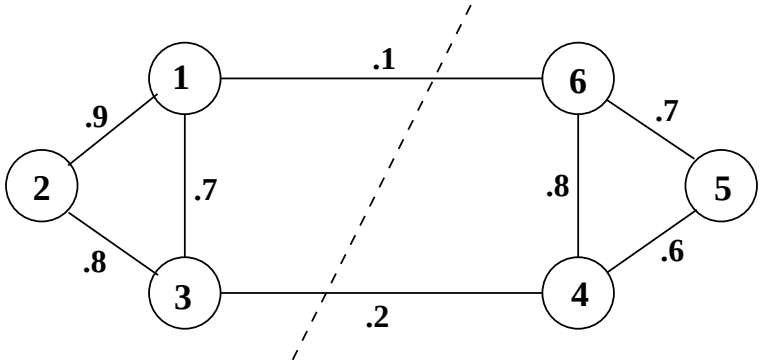
Its corresponding eigenvector is the Fiedler vector.

Example:



Using the definition of the Laplacian matrix L, we have

$$\mathbf{L} = -\begin{pmatrix} -1.7 & 0.9 & 0.7 & 0 & 0 & 0.1 \\ 0.9 & -1.7 & 0.8 & 0 & 0 & 0 \\ 0.7 & 0.8 & -1.7 & 0.2 & 0 & 0 \\ 0 & 0 & 0.2 & -1.6 & 0.6 & 0.8 \\ 0 & 0 & 0 & 0.6 & -1.3 & 0.7 \\ 0.1 & 0 & 0 & 0.8 & 0.7 & -1.6 \end{pmatrix}.$$



The eigenvalues and associated eigenvectors of L are given by

$$\textbf{Eigenvalues} = \begin{pmatrix} 0.0000 & 0.1876 & 1.9832 & 2.2582 & 2.5487 & 2.6222 \end{pmatrix},$$

$$\textbf{Eigenvectors} = \begin{pmatrix} 0.4082 & -0.4080 & 0.0864 & -0.4285 & 0.3379 & 0.6014 \\ 0.4082 & -0.4401 & 0.1094 & -0.0975 & 0.1841 & -0.7644 \\ 0.4082 & -0.3731 & -0.1359 & 0.5501 & -0.5755 & 0.2046 \\ 0.4082 & 0.3670 & -0.5473 & 0.3544 & 0.5229 & 0.0091 \\ 0.4082 & 0.4514 & 0.7652 & 0.2025 & 0.0271 & 0.0483 \\ 0.4082 & 0.4027 & -0.2778 & -0.5810 & -0.4966 & -0.0990 \end{pmatrix}.$$

The Signless Laplacian Matrix and Maximum Association:

— to maximize the total edge weight within two clusters.

$$M_{ij} = \begin{cases} \sum_k w_{ik}, & \text{if } i = j \\ +w_{ij}, & \text{if } i \neq j,\ i \text{ and } j \text{ are adjacent} \\ 0, & \text{otherwise,} \end{cases}$$

Given a graph $G = (V, E)$ and two clusters V_1, V_2 where $V_1 \cap V_2 = \emptyset$ and $V_1 \cup V_2 = V$, the cohesion/association is defined as

$$\text{Cohesion}(V_1, V_2) = \left(\sum_{i,j \in V_1} w_{ij} + \sum_{i,j \in V_2} w_{ij} \right). \qquad (5)$$

We seek to maximize this quantity over all partitions of G.

Given the partition vector p

$$p_i = \begin{cases} +1, & \text{vertex } i \in V_1, \\ -1, & \text{vertex } i \in V_2, \end{cases} \quad \Longrightarrow \quad p^T M p = \sum_{\{i,j\} \in E} w_{ij}(p_i + p_j)^2.$$

Notice that this time:

— edges connecting the two subsets V_1 and V_2 do not contribute

— edges within each cluster contribute 4 times their weight.

$$\frac{p^T M p}{p^T p} = \frac{1}{n} \cdot 4\,\text{Cohesion}(V_1, V_2)$$

The optimal partition is the one that maximizes this Raleigh quotient.

$$\lambda_{min} \leq p^T M p / p^T p \leq \lambda_{max}$$

where λ_{min} and λ_{max} are the smallest and largest eigenvalues of M.

Properties of the signless Laplacian:

— all its eigenvalues are real and positive.

— the right-hand eigenvector corresponding to the largest eigenvalue is the only eigenvector whose elements are all nonzero and positive.

The eigenvector of M corresponding to the second largest eigenvalue provides node clustering information.

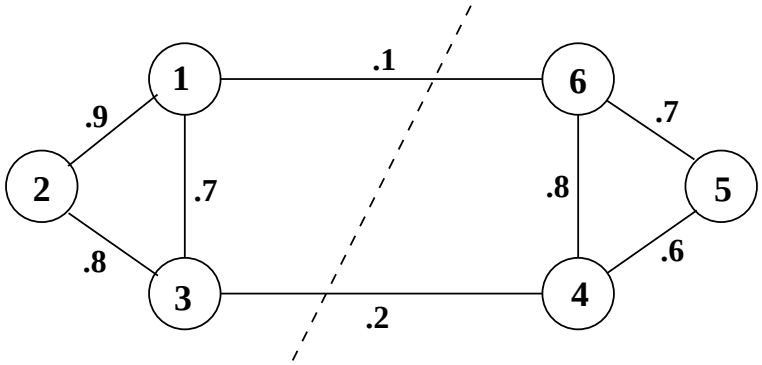
The second largest eigenvalue of the signless Laplacian provides a quantitative evaluation of the total weight of edges within clusters.

The Laplacian L and signless Laplacian M can be obtained as $L = D - A$ and $M = D + A$ (and hence $M = 2D - L$), where

— A is the adjacency matrix;

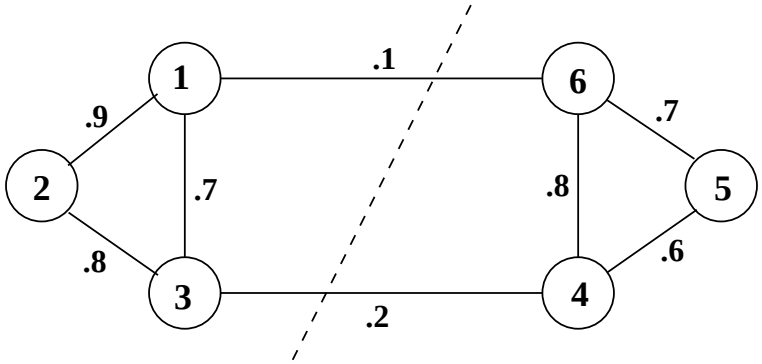
— D is a diagonal matrix with $D_{ii} = \sum_k w_{ik}$.

Example:



Using the definition of the signless Laplacian matrix M, we have

$$\mathbf{M} = \begin{pmatrix} 1.7 & 0.9 & 0.7 & 0 & 0 & 0.1 \\ 0.9 & 1.7 & 0.8 & 0 & 0 & 0 \\ 0.7 & 0.8 & 1.7 & 0.2 & 0 & 0 \\ 0 & 0 & 0.2 & 1.6 & 0.6 & 0.8 \\ 0 & 0 & 0 & 0.6 & 1.3 & 0.7 \\ 0.1 & 0 & 0 & 0.8 & 0.7 & 1.6 \end{pmatrix}.$$



The eigenvalues and associated eigenvectors of M are given by

$$\textbf{Eigenvalues} = \begin{pmatrix} 0.6918 & 0.7581 & 0.8153 & 1.1043 & 2.8996 & 3.3309 \end{pmatrix},$$

$$\textbf{Eigenvectors} = \begin{pmatrix} -0.3318 & -0.2012 & 0.5223 & 0.4930 & -0.1426 & 0.5597 \\ 0.1546 & 0.5654 & -0.5136 & 0.1750 & -0.1764 & 0.5752 \\ 0.1784 & -0.4393 & -0.0195 & -0.6850 & -0.1041 & 0.5429 \\ -0.3566 & 0.5115 & 0.3130 & -0.3852 & 0.5802 & 0.1677 \\ -0.4591 & -0.4137 & -0.5880 & 0.1696 & 0.4830 & 0.1015 \\ 0.7045 & -0.1182 & 0.1389 & 0.2828 & 0.6064 & 0.1509 \end{pmatrix}.$$

Indicators of State Clustering

The significance of subdominant, right-hand eigenvectors

System equilibrium is defined by the stationary probability vector.

Associate a real number with each state
— its "distance" from this equilibrium position.

Let $w_i^{(1)} = (0, 0, \ldots, 1, \ldots, 0) \implies$ system is initially in state i.

Let $x_1, x_2, \ldots, x_n$ be the left-hand eigenvectors of P

$$\text{i.e.,} \quad x_j^T P = \lambda_j x_j^T \quad \text{for all } j = 1, 2, \ldots, n,$$

arranged into descending order

Writing $w_i^{(1)}$ as a linear combination of these eigenvectors:

$$w_i^{(1)} = c_{i1} x_1^T + c_{i2} x_2^T + \ldots + c_{in} x_n^T$$

$$w_i^{(1)} = c_{i1}x_1^T + c_{i2}x_2^T + \ldots + c_{in}x_n^T$$

Repeated postmultiplication of $w_i^{(1)}$ by P yields the steady-state.

$$\begin{aligned} w_i^{(1)}P &= c_{i1}x_1^T P + c_{i2}x_2^T P + \ldots + c_{in}x_n^T P \qquad (6) \\ &= c_{i1}x_1^T + c_{i2}\lambda_2 x_2^T + \ldots + c_{in}\lambda_n x_n^T = w_i^{(2)}, \qquad (7) \end{aligned}$$

In general

$$w_i^{(k+1)} = c_{i1}x_1^T + c_{i2}\lambda_2^k x_2^T + \ldots + c_{in}\lambda_n^k x_n^T.$$

If the system initially starts in some other state $j \neq i$, we have

$$w_j^{(k+1)} = c_{j1}x_1^T + c_{j2}\lambda_2^k x_2^T + \ldots + c_{jn}\lambda_n^k x_n^T.$$

Only the constant coefficients differ

Observe what happens when $\lambda_2^k \gg \lambda_l^k$ for $l \geq 3$.

Considering all possible starting states:

$$\begin{pmatrix} w_1^{(k+1)} \\ w_2^{(k+1)} \\ \vdots \\ w_n^{(k+1)} \end{pmatrix} = \begin{pmatrix} c_{11}x_1^T + c_{12}\lambda_2^k x_2^T + \ldots + c_{1n}\lambda_n^k x_n^T \\ c_{21}x_1^T + c_{22}\lambda_2^k x_2^T + \ldots + c_{2n}\lambda_n^k x_n^T \\ \vdots \\ c_{n1}x_1^T + c_{n2}\lambda_2^k x_2^T + \ldots + c_{nn}\lambda_n^k x_n^T \end{pmatrix},$$

i.e.,

$$W^{(k+1)} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{pmatrix} \begin{pmatrix} 1 & & & \\ & \lambda_2^k & & \\ & & \ddots & \\ & & & \lambda_n^k \end{pmatrix} \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{pmatrix} \equiv C\Lambda^k X^T$$

We now need to find C.

Recall that

$$W^{(1)} = (w_1^{(1)}, w_2^{(1)}, \ldots, w_n^{(1)})^T = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

and that

$$w_i^{(1)} = c_{i1}x_1^T + c_{i2}x_2^T + \ldots + c_{in}x_n^T, \quad i = 1, 2, \ldots, n.$$

Thus

$$W^{(1)} = CX^T,$$

and since $W^{(1)} = I$, we obtain

$$I = CX^T,$$

i.e., $C = (X^T)^{-1} = Y$, the set of right-hand eigenvectors of P.

Therefore a relative distance of each state from the stationary probability vector is obtained from the second column of the matrix C:

$$\begin{pmatrix} w_1^{(k+1)} \\ w_2^{(k+1)} \\ \vdots \\ w_n^{(k+1)} \end{pmatrix} = \begin{pmatrix} c_{11}x_1^T + c_{12}\lambda_2^k x_2^T + \ldots + c_{1n}\lambda_n^k x_n^T \\ c_{21}x_1^T + c_{22}\lambda_2^k x_2^T + \ldots + c_{2n}\lambda_n^k x_n^T \\ \vdots \\ c_{n1}x_1^T + c_{n2}\lambda_2^k x_2^T + \ldots + c_{nn}\lambda_n^k x_n^T \end{pmatrix},$$

Subsequent columns may be used to obtain subsidiary effects.

- States whose corresponding component value in this vector is large are, in a relative sense, far from the equilibrium position.
- States corresponding to component values that are relatively close together form a cluster, or a subset, of states.

Markov Random Walks on Graphs

The adjacency matrix A of a graph can be converted to a transition probability matrix P to generate a random walk on a graph.

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

Since $L = D - A$ and $M = D + A$, we have

$$I - P = D^{-1}L \qquad \text{and} \qquad I + P = D^{-1}M$$

Property: If (λ, v) is an eigen-solution of $Pv = \lambda v$, then it is also an eigen-solution of the generalized eigenvalue problems

$$(1 - \lambda)Dv = Lv \quad \text{and} \quad (1 + \lambda)Dv = Mv.$$

Proof: Given an eigen-pair (λ, v) such that $Pv = \lambda v$, then, since $P = D^{-1}A$ and $L = D - A$, we have

$$Pv = \lambda v \quad \Rightarrow \quad D^{-1}Av = \lambda v \quad \Rightarrow \quad D^{-1}(D - L)v = \lambda v$$

$$\Rightarrow \quad Iv - D^{-1}Lv = \lambda v \quad \Rightarrow \quad (1 - \lambda)Dv = Lv \tag{8}$$

A similar result holds for the Laplacian matrix $M = D + A$.

The eigenvectors of the generalized eigenvalue problem $(1 - \lambda)Dv = Lv$ provide a heuristic solution for the minimum balanced cut on a graph. (Shi and Malik).

It therefore follows that the right eigenvectors of P also provide the same balanced cut solution.

Property: The eigenvalues of the probability matrix P derived from a random walk on a graph are real.

Proof: $P = D^{-1}A$ has a symmetric structure. Also, since

$$D^{1/2}PD^{-1/2} = D^{-1/2}AD^{-1/2},$$

P is similar to $D^{-1/2}AD^{-1/2}$, which is symmetric. Result follows.

This provides an alternative way to calculate the left- and right-hand eigenvectors of P, i.e., $Px_R = \lambda x_R$ and $P^T x_L = \lambda x_L$.

$$Px_R = \lambda x_R \quad \Rightarrow \quad D^{-1}Ax_R = \lambda x_R.$$

If we premultiply $D^{1/2}$ on both sides, we obtain

$$D^{-1/2}AD^{-1/2}(D^{1/2}x_R) = \lambda(D^{1/2}x_R).$$

$D^{1/2}x_R$ is an eigenvector of the symmetric matrix $D^{-1/2}AD^{-1/2}$.

Now consider x_L. Since A is symmetric

$$P^T x_L = \lambda x_L \quad \Rightarrow \quad AD^{-1}x_L = \lambda x_L.$$

Premultiplying with $D^{-1/2}$ on both sides gives

$$D^{-1/2}AD^{-1/2}(D^{-1/2}x_L) = \lambda(D^{-1/2}x_L).$$

$\Longrightarrow$ $D^{-1/2}x_L$ is an eigenvector of the symmetric matrix $D^{-1/2}AD^{-1/2}$.

Procedure:

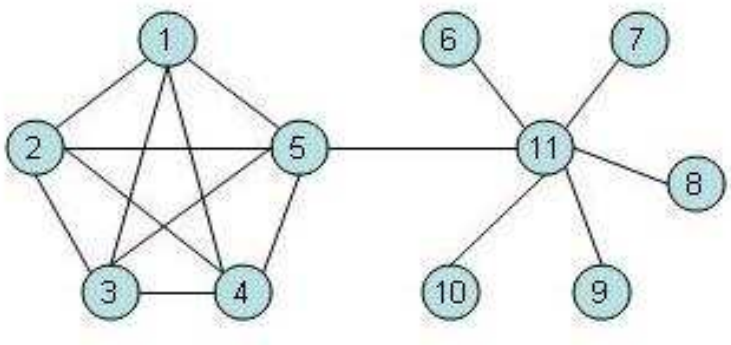
Compute (λ, v), the eigenvalue/vector of $D^{-1/2}AD^{-1/2}$, and then set

$$x_R = D^{-1/2}v \quad \text{and} \quad x_L = D^{1/2}v.$$

Clustering Techniques for Markov Chains

If the Markov chain is ergodic, then spectral decomposition on P provides two clustering measures for graph nodes:

1. The balanced minimum cut (alternatively, maximum cohesion) — the eigenvector associated with the second smallest (positive) eigenvalue.
2. The "distance" from each state to the steady state — the eigenvector associated with the second largest modulus of eigenvalues (not necessarily positive).



$$\lambda_1 = 0.8852; \quad \lambda_2 = -0.9336.$$

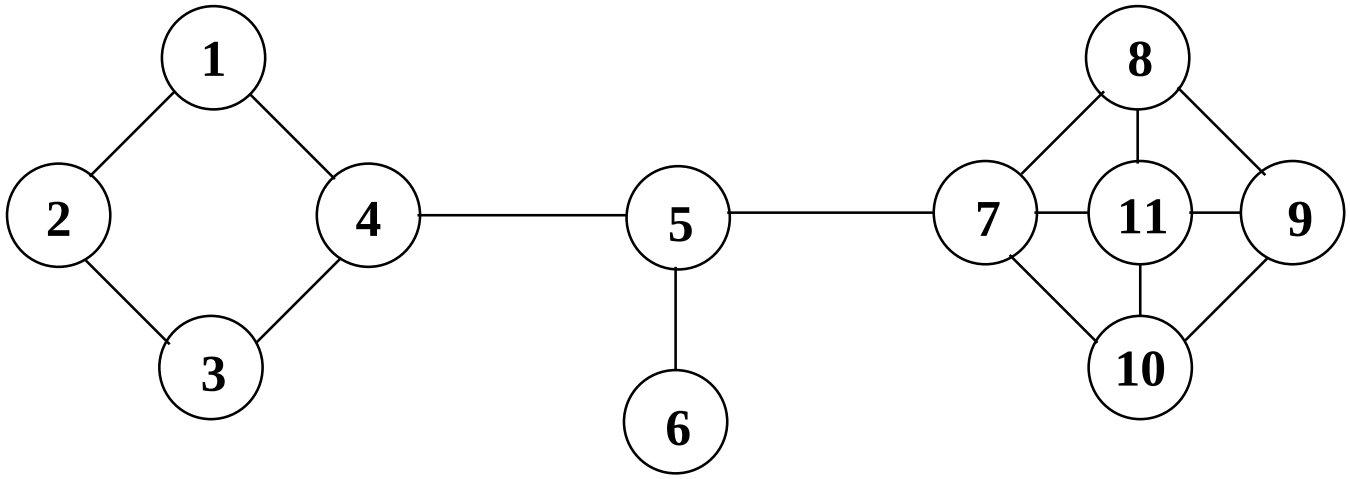
$$(\mathbf{v_1}, \mathbf{v_2}) = \begin{pmatrix} 1.0 & 1.0 \\ 1.0 & 1.0 \\ 1.0 & 1.0 \\ 1.0 & 1.0 \\ 0.5411 & -6.7333 \\ -1.8138 & -29.3810 \\ -1.8138 & -29.3810 \\ -1.8138 & -29.3810 \\ -1.8138 & -29.3810 \\ -1.8138 & -29.3810 \\ -1.6058 & 27.4286 \end{pmatrix}; \quad (\mathbf{v_1}, \mathbf{v_2}) = \begin{pmatrix} -0.5513 & -0.0340 \\ -0.5513 & -0.0340 \\ -0.5513 & -0.0340 \\ -0.5513 & -0.0340 \\ -0.2983 & 0.2292 \\ 1.0 & 1.0 \\ 1.0 & 1.0 \\ 1.0 & 1.0 \\ 1.0 & 1.0 \\ 1.0 & 1.0 \\ 0.8853 & -0.9335 \end{pmatrix}.$$

From the second clustering measure (using v_2):

(a) The first five elements in cluster 1 have relatively small modulus (states are closely linked) while others in cluster 2 have relatively large modulus (opposite structure).

(b) The value of vertex 5 is closer to cluster 2 while the value of vertex 11 is closer to cluster 1. Because vertex 5 and 11 have the possibility of transitioning to the other cluster in a single step; they are the connecting vertices between clusters.

Example 2:



Eigenvalues of P: $\lambda_1 = 1, \quad \lambda_2 = 0.9328, \ \ldots, \quad \lambda_{11} = -0.9653.$

$$(1,\ 0.9328,\ 0.6281,\ 0,\ 0,\ 0,\ 0,\ -0.3216,\ -0.5539,\ -0.7201,\ -0.9653)$$

$$(\mathbf{v_2}, \mathbf{v_3}, \mathbf{v_{11}}) = \begin{pmatrix} -0.2700 & 0.1801 & 0.3132 \\ -0.2895 & 0.2868 & -0.3245 \\ -0.2700 & 0.1801 & 0.3132 \\ -0.2142 & -0.0605 & -0.2802 \\ -0.0595 & -0.4743 & 0.1849 \\ -0.0638 & -0.7552 & -0.1916 \\ 0.1116 & -0.0780 & -0.0637 \\ 0.1578 & 0.0896 & 0.0264 \\ 0.1700 & 0.1478 & -0.0211 \\ 0.1578 & 0.0896 & 0.0264 \\ 0.1601 & 0.0991 & 0.0083 \end{pmatrix}.$$

NCD Markov Chains

Example 3: Courtois 8×8 transition probability matrix, P:

$$\left(\begin{array}{cccc|cc|ccc}
0.85 & 0.0 & 0.149 & 0.0009 & 0.0 & 0.00005 & 0.0 & 0.00005 \\
0.1 & 0.65 & 0.249 & 0.0 & 0.0009 & 0.00005 & 0.0 & 0.00005 \\
0.1 & 0.8 & 0.0996 & 0.0003 & 0.0 & 0.0 & 0.0001 & 0.0 \\
\hline
0.0 & 0.0004 & 0.0 & 0.7 & 0.2995 & 0.0 & 0.0001 & 0.0 \\
0.0005 & 0.0 & 0.0004 & 0.399 & 0.6 & 0.0001 & 0.0 & 0.0 \\
\hline
0.0 & 0.00005 & 0.0 & 0.0 & 0.00005 & 0.6 & 0.2499 & 0.15 \\
0.00003 & 0.0 & 0.00003 & 0.00004 & 0.0 & 0.1 & 0.8 & 0.0999 \\
0.0 & 0.00005 & 0.0 & 0.0 & 0.00005 & 0.1999 & 0.25 & 0.55
\end{array}\right)$$

- The eigenvalues of P are

$$\begin{pmatrix} 1.0 & 0.9998 & 0.9985 & 0.7500 & 0.5501 & 0.4000 & 0.3007 & -0.1495 \end{pmatrix}.$$

- The right-hand eigenvectors v_2 and v_3 are both important.

$$(\mathbf{v_2}, \mathbf{v_3}) = \left(\begin{array}{rr} -0.3536 & -0.4876 \\ -0.3536 & -0.4878 \\ -0.3536 & -0.4883 \\ \hline -0.3536 & 0.3783 \\ -0.3536 & 0.3777 \\ \hline 0.3536 & 0.0073 \\ 0.3536 & 0.0073 \\ 0.3536 & 0.0073 \end{array}\right).$$