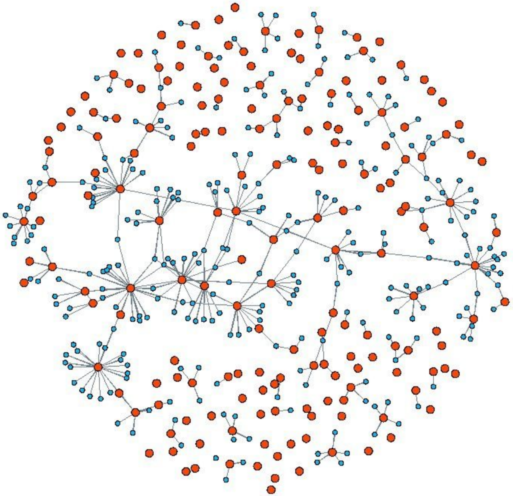
The Kemeny Constant and the Kemeny Decomposition Algorithm

Numerical Methods for Markov Chains

Sebastiano Scardera

Università di Pisa

Introduction



Setting

Definition (Markov influence graph)

Let $(\mathbb{S}, E)$ be a directed graph with finite nodes $\mathbb{S} = \{1, ..., n\}$ and $E \subseteq \mathbb{S} \times \mathbb{S}$. We can define a Markov chain on it with a transition matrix P such that:

- for $(i,j) \in E, \; P(i,j) > 0$ (probability of going from node i to node j);
- for $(i,j) \notin E, \; P(i,j) = 0$;
- if there exists $i \in \mathbb{S}$ for which there is no $j \in \mathbb{S}$ such that $(i,j) \in E$, we artificially add the self-loop (i,i) to E.

The triple $(\mathbb{S}, E, P)$ is called the *Markov influence graph*.

Setting

Definition (Ergodic projector)

Let P be the transition matrix, then it can be shown that

$$\exists \Pi_P = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} P^n.$$

$(\Pi_P(i,j))_{\mathbb{S}\times\mathbb{S}}$ is called the *ergodic projector*.

Definition (Deviation matrix)

We define the *deviation matrix* as:

$$D_P = (I - P - \Pi_P)^{-1} - \Pi_P,$$

where I is the identity matrix of the appropriate size.

Setting

Definition (Fundamental matrix)

$Z_P = D_P + \Pi_P$ is called the *fundamental matrix*.

Definition (Drazin inverse)

Let $A \in \mathbb{K}^{n \times n}$, then we call the *Drazin inverse* of A, if it exists, the unique matrix $A^{\#}$ such that:

$$AA^{\#}A = A^{\#}, \qquad A^{\#}AA^{\#} = A \qquad \text{and} \qquad AA^{\#} = A^{\#}A.$$

Lemma

If $A = I - P$, it can be shown[1] that the following properties hold:

i) $A^{\#} = (I - P)^{\#}$ and $D = A^{\#}$;

ii) $\Pi_P = I - AA^{\#}$ and $\Pi_P^2 = \Pi_P$ (Π_P is a projector);

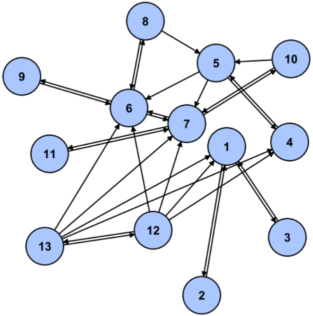
iii) $Z_P \Pi_P = \Pi_P$ or equivalently $D_P \Pi_P = 0$.

[1] Nazarathy, Yoni. "Linear Control Theory and Structured Markov Chains."

Example 0

Consider a Markov influence graph $(\mathbb{S}, E, P)$ where:

- $\mathbb{S} = \{1, ..., 13\}$;
- $(i, j) \in E$ if agent i influences agent j;
- $P(i, j)$ measures the weight of the influence of i on j (normalized).

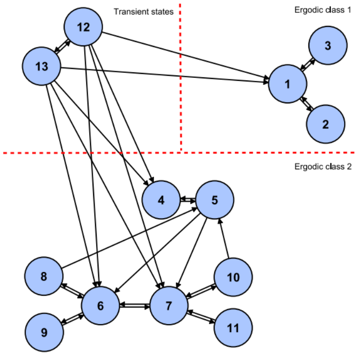


$$P = \begin{bmatrix}
0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.99 & 0 & 0.005 & 0.005 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.005 & 0.995 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.005 & 0 & 0.995 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0.222 & 0 & 0 & 0.222 & 0 & 0.111 & 0.111 & 0 & 0 & 0 & 0 & 0 & 0.333 \\
0.083 & 0 & 0 & 0.75 & 0 & 0.042 & 0.042 & 0 & 0 & 0 & 0 & 0.083 & 0
\end{bmatrix}$$

Example 0

Definition (Ergodic class)

A closed and irreducible set of states is called an *ergodic class*.



Ergodic projector:

$$\Pi_P = \begin{bmatrix} 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0.13 & 0.06 & 0.06 & 0.08 & 0.08 & 0.2 & 0.2 & 0.04 & 0.06 & 0.04 & 0.06 & 0 & 0 \\ 0.05 & 0.03 & 0.03 & 0.09 & 0.09 & 0.24 & 0.24 & 0.05 & 0.07 & 0.05 & 0.07 & 0 & 0 \end{bmatrix}$$

Ergodic class 1 (rows 1–3, columns 1–3); Ergodic class 2 (rows 4–11, columns 4–11)

Example 0

Definition (Ergodic class)

A closed and irreducible set of states is called an *ergodic class*.

Ranking by influence

1st ergodic class:

$1 \succ 2 \sim 3$

2nd ergodic class:

$6 \sim 7 \succ 5 \succ 4 \succ 9 \sim 11 \succ 8 \sim 10$

Ergodic projector:

$$\Pi_P = \begin{bmatrix} 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0.11 & 0.26 & 0.26 & 0.05 & 0.08 & 0.05 & 0.08 & 0 & 0 \\ 0.13 & 0.06 & 0.06 & 0.08 & 0.08 & 0.2 & 0.2 & 0.04 & 0.06 & 0.04 & 0.06 & 0 & 0 \\ 0.05 & 0.03 & 0.03 & 0.09 & 0.09 & 0.24 & 0.24 & 0.05 & 0.07 & 0.05 & 0.07 & 0 & 0 \end{bmatrix}$$

Ergodic class 1

Ergodic class 2

Unichain case (without transient states)

One possible direct distance is given by the matrix M.

Definition (Mean first passage time matrix)

$M(i,j)$ is the average time to first pass from i to j

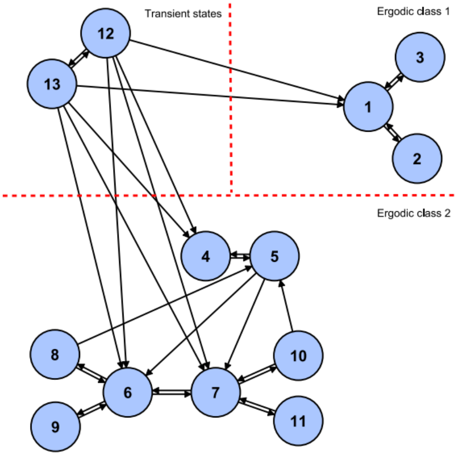
In the case with a single ergodic class and without transient states, the explicit expression for M is:

$$M = (I - D_P + ee^T \cdot \mathrm{dg}(D_P)) \cdot \mathrm{dg}(\Pi_P)^{-1},$$

where e is the vector of all 1's of the appropriate size and $\mathrm{dg}(D_P)$ is the matrix that has D_P as its diagonal and 0 elsewhere.

Unichain case (without transient states)

Example 0



$$M = \begin{bmatrix} 2 & 3 & 3 & - & - & - & - & - & - & - & - \\ 1 & 4 & 4 & - & - & - & - & - & - & - & - \\ 1 & 4 & 4 & - & - & - & - & - & - & - & - \\ - & - & - & 9.6 & 1 & 400.9 & 400.9 & 416.9 & 412.5 & 416.9 & 412.5 \\ - & - & - & 8.6 & 9.5 & 399.9 & 399.9 & 415.9 & 411.5 & 415.9 & 411.5 \\ - & - & - & 1508.6 & 1500 & 3.8 & 3.8 & 16 & 11.7 & 19.8 & 15.4 \\ - & - & - & 1508.6 & 1500 & 3.8 & 3.8 & 19.8 & 15.4 & 16 & 11.7 \\ - & - & - & 1502.1 & 1493.5 & 3 & 6.8 & 19 & 14.7 & 22.8 & 18.4 \\ - & - & - & 1509.6 & 1501 & 1 & 4.8 & 17 & 12.7 & 20.8 & 16.4 \\ - & - & - & 1502.1 & 1493.5 & 6.8 & 3 & 22.8 & 18.4 & 19 & 14.7 \\ - & - & - & 1509.6 & 1501 & 4.8 & 1 & 20.8 & 16.4 & 17 & 12.7 \end{bmatrix}$$

Unichain case (without transient states)

Definition 1 (Kemeny constant for unichain)

The *Kemeny constant* for a Markov chain with a single ergodic state and without transient states is given by:

$$K_P = \sum_{j \in \mathbb{S}} M(i,j)\pi_P(j), \ \forall i \in \mathbb{S},$$

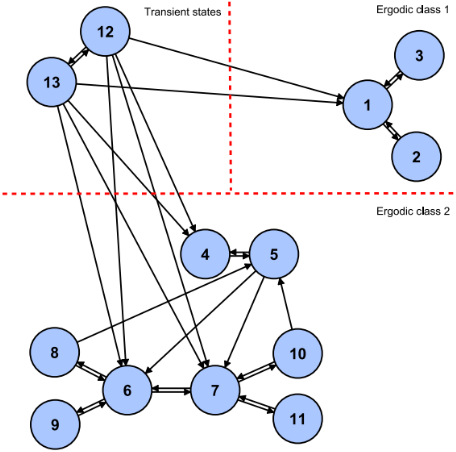
where π_P is the stationary probability vector for P.

Remark

- K_P provides the average number of steps required to reach any chosen state following the distribution given by π_P;
- K_P is constant with respect to the initial state;
- The lower the value of K_P, the better the connectivity of the graph.

Unichain case (without transient states)

Example 0



2nd ergodic class: $K_P = 321.5$

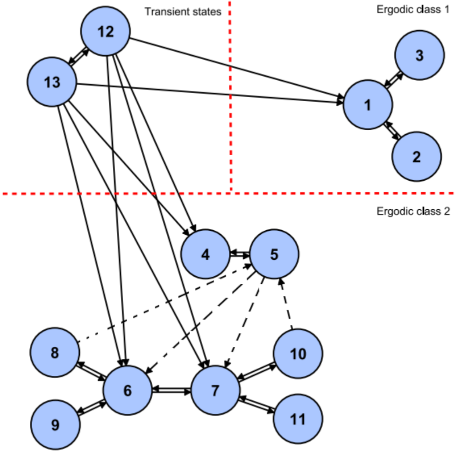
- Idea: "Derive" K_P and identify the critical edges;
- The value of the derivative (defined later) will be smaller on the edges: $(8,5), (10,5), (5,6)$ and $(5,7)$;
- By removing these edges, we obtain the subgraphs:

$$\{4,5\} \text{ with constant } K_1 = 1.5,$$

$$\{6,...,11\} \text{ with constant } K_2 = 6.2.$$

Unichain case (without transient states)

Example 0



2nd ergodic class: $K_P = 321.5$

- Idea: "Derive" K_P and identify the critical edges;
- The value of the derivative (defined later) will be smaller on the edges: $(8, 5), (10, 5), (5, 6)$ and $(5, 7)$;
- By removing these edges, we obtain the subgraphs:

$$\{4, 5\} \text{ with constant } K_1 = 1.5,$$

$$\{6, ..., 11\} \text{ with constant } K_2 = 6.2.$$

Multichain case

Unichain case (without transient states): $K_P = \sum_{j \in \mathbb{S}} M(i,j)\pi_P(j), \ \forall i \in \mathbb{S}.$

To generalize to the multichain case, there may be some critical points:

- Unlike the unichain case, the initial state influences the average number of definitive visits because once we enter an ergodic class, we do not leave it.
- The matrix M is only significant in the unichain case.

Multichain case

$$P = \begin{bmatrix} P_1 & 0 & 0 & \cdots & 0 \\ 0 & P_2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & P_E & 0 \\ P_{T1} & P_{T2} & \cdots & P_{TE} & P_{TT} \end{bmatrix} \qquad \Pi_P = \begin{bmatrix} \Pi_1 & 0 & 0 & \cdots & 0 \\ 0 & \Pi_2 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \Pi_E & 0 \\ R_1 & R_2 & \cdots & R_E & 0 \end{bmatrix}$$

where E is the number of ergodic classes, T is the set of transient states (possibly empty), and Π_i is the ergodic projector related to the transition matrix P_i.

Multichain case

$$D_P = \begin{bmatrix} D_{P_1} & 0 & 0 & \cdots & 0 \\ 0 & D_{P_2} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & D_{P_E} & 0 \\ D_{P_{T1}} & D_{P_{T2}} & \cdots & D_{P_{TE}} & D_{P_{TT}} \end{bmatrix},$$

where

$D_{P_i} = (I - P_i + \Pi_i)^{-1} - \Pi_i, \quad i = 1, \ldots, E$ and $D_{P_{TT}} = (I - P_{TT})^{-1}$ because $\Pi_{TT} = 0$.

Furthermore,

$$D_{P_{T_i}} = (I - P_{TT})^{-1} \cdot (P_{T_i} - R_i) \cdot (I - P_i + \Pi_i)^{-1} - R_i.$$

Multichain case

$$D_P = \begin{bmatrix} D_{P_1} & 0 & 0 & \cdots & 0 \\ 0 & D_{P_2} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & D_{P_E} & 0 \\ D_{P_{T1}} & D_{P_{T2}} & \cdots & D_{P_{TE}} & D_{P_{TT}} \end{bmatrix},$$

Definition 2 (Kemeny's constant)

$$K_P = \text{tr}(D_P) + 1$$

Exploiting the linearity of the trace:

$$K_P = \sum_{i=1}^{E} K_{P_i} + \text{tr}(D_{P_{TT}}) - E + 1,$$

$$\text{where } K_{P_i} = \sum_{j \in E_i} M(k,j)\pi_P(j), \ \forall k \in E_i.$$

Graph derivative

For $(i, j) \in E$, let's consider the transition matrix:

$$R_{ij} = P - e_i e_i^T P + e_i e_j = P - e_i(e_i^T P - e_j),$$

and perturb P using R_{ij}:

$$P_{ij}(\theta) = (1 - \theta)P + \theta R_{ij}, \quad \text{for } \theta \in (0, 1).$$

Graph derivative

Theorem

For all $(i,j) \in E$, we have:

$$\frac{d}{d\theta} K_{P_{ij}(\theta)} = (D_{P_{ij}(\theta)})^2(j,i) - (P \cdot (D_{P_{ij}(\theta)})^2)(i,i);$$

in matrix form:

$$\left(\frac{d}{d\theta} K_{P_{ij}(\theta)} \Big|_{\theta=0} \right)_{(i,j)\in E} = ((D_P)^2)^T - \text{dg}(P \cdot (D_P)^2) ee^T,$$

where e is the vector of all 1's and $\text{dg}(A)$ is the matrix A with elements off the diagonal replaced by 0.

Graph derivative

Lemma 1

If P and P' are two Markov chains on the same state space $\mathbb{S}$ and have the same ergodic classes, then:

$$\Pi_P P' \Pi_P = \Pi_P \qquad \text{and} \qquad \Pi_P \Pi_{P'} \Pi_P = \Pi_P.$$

Graph derivative

Proof outline

Lemma

If P and P' are two Markov chains on the same state space $\mathbb{S}$ and have the same ergodic classes, then:

$$\Pi_P P' \Pi_P = \Pi_P \qquad \text{and} \qquad \Pi_P \Pi_{P'} \Pi_P = \Pi_P.$$

Proof.

- $\Pi_P(i,\cdot) = [\alpha_{i1}\pi_{P_1}^T \ \alpha_{i2}\pi_{P_2}^T \ \dots \ \alpha_{iE}\pi_{P_E}^T \ 0 \ \dots \ 0]$;
- $\Pi_P(i,\cdot)P'\Pi_P = [\alpha_{i1}\pi_{P_1}^T P_1' \Pi_{P_1} \ \dots \ \alpha_{iE}\pi_{P_E}^T P_E' \Pi_{P_E} \ 0 \ \dots \ 0]$;
- $\pi_{P_j}^T P_j'$ is stochastic and Π_{P_j} has all rows equal to $\pi_{P_j}^T$, hence $\pi_{P_j}^T P_j' \Pi_{P_j} = \pi_{P_j}^T$;
- it follows that $\Pi_P(i,\cdot)P'\Pi_P = \Pi_P(i,\cdot)$.

□

Graph derivative

Proof outline

$$P(\theta) = (1-\theta)P + \theta Q, \quad \theta \in [0,1].$$

Theorem 1

If P and Q have the same ergodic classes, then:

$$\frac{d}{d\theta}\Pi_{P(\theta)} = D_{P(\theta)}(Q-P)\Pi_{P(\theta)} + \Pi_{P(\theta)}(Q-P)D_{P(\theta)}.$$

Graph derivative

Proof outline

Theorem

$$\frac{d}{d\theta}\Pi_{P(\theta)} = D_{P(\theta)}(Q-P)\Pi_{P(\theta)} + \Pi_{P(\theta)}(Q-P)D_{P(\theta)}.$$

Proof.

Let's define the *differential matrix* $U(\Delta) = (P(\theta+\Delta) - P(\theta))Z_{P(\theta)}$.
The following identity holds[2]:

$$\Pi_{P(\theta+\Delta)} = Z_{P(\theta)}(I - U(\Delta))^{-1}\Pi_{P(\theta)} + \Pi_{P(\theta+\Delta)}U(\Delta).$$

□

[2] Schweitzer PJ (1968) Perturbation theory and finite Markov chains

Graph derivative

Proof outline

Theorem

$$\frac{d}{d\theta}\Pi_{P(\theta)} = D_{P(\theta)}(Q-P)\Pi_{P(\theta)} + \Pi_{P(\theta)}(Q-P)D_{P(\theta)}.$$

Proof.

From $\frac{d}{d\theta}\Pi_{P(\theta)} = Z_{P(\theta)}(I - U(\Delta))^{-1}(Q-P)\Pi_{P(\theta)} + \Pi_{P(\theta+\Delta)}(Q-P)Z_{P(\theta)}$, using:

$$Z_{P(\theta)} = D_{P(\theta)} + \Pi_{P(\theta)} \qquad \text{and} \qquad \Pi_{P(\theta)}P\Pi_{P(\theta)} = \Pi_{P(\theta)}Q\Pi_{P(\theta)} = \Pi_{P(\theta)},$$

we obtain: $\frac{d}{d\theta}\Pi_{P(\theta)} = D_{P(\theta)}(Q-P)\Pi_{P(\theta)} + \Pi_{P(\theta)}(Q-P)D_{P(\theta)}$. □

Theorem 2

If P and Q have the same ergodic classes:

$$\frac{d}{d\theta}D_{P(\theta)} = D_{P(\theta)}(Q-P)D_{P(\theta)} - (D_{P(\theta)})^2(Q-P)\Pi_{P(\theta)} - \Pi_{P(\theta)}(Q-P)(D_{P(\theta)})^2$$

Conclusion of the proof of the main theorem.

Using the theorems and the definition of $K_{P(\theta)}$:

$$\frac{d}{d\theta}K_{P(\theta)} = \frac{d}{d\theta}(\mathrm{tr}(D_{P(\theta)}) + 1) = \mathrm{tr}\left(\frac{d}{d\theta}D_{P(\theta)}\right) = \mathrm{tr}\left((Q-P)(D_{P(\theta)})^2\right).$$

In our case, $Q = R_{ij}$ and $P_{ij}(\theta) = (I-\theta)P + \theta R_{ij}$:

$$\begin{aligned}\frac{d}{d\theta}K_{P_{ij}(\theta)} &= \mathrm{tr}\left((R_{ij}-P)(D_{P_{ij}(\theta)})^2\right) = e_i^T(R_{ij}-P)(D_{P_{ij}(\theta)})^2 e_i \\ &= (-e_i^T P + e_j^T)(D_{P_{ij}(\theta)})^2 e_i = (D_{P_{ij}(\theta)})^2(j,i) - (P\cdot(D_{P_{ij}(\theta)})^2)(i,i)\end{aligned}$$

□ □

Hypothesis

1. If $\frac{d}{d\theta} K_{P_{ij}}(\theta)\Big|_{\theta=0} \simeq -\epsilon$:
   - Increasing $P(i,j)$ significantly improves the connectivity of the graph;
   - Cutting the edge (i,j) (i.e., setting $P(i,j) = 0$ and normalizing) divides the graph in a significant way;
2. Cutting the edges in increasing order with respect to $\frac{d}{d\theta} K_{P_{ij}}(\theta)\Big|_{\theta=0}$ for each (i,j) leads to a natural decomposition of the network.

Kemeny decomposition algorithm

FUNCTION KDA(P,CO_A,CO_B,SC):

INPUT:
P = Markov chain transition matrix
CO_A = Condition A
CO_B = Condition B
SC = True, when edges are simmetrically cut, False otherwise.

START:

- Initialize cut transition matrix $P^c = P$, and set $\mathbb{E} = E$.
- While CO_A:
  - → For all $(i,j) \in \mathbb{E}$, calculate $\frac{d}{d\theta} K(P^c_{ij}(\theta))\Big|_{\theta=0}$.
  - → While CO_B:
    - △ Determine $(i^\star, j^\star) = \arg\min_{(i,j)\in\mathbb{E}} \frac{d}{d\theta} K(P^c_{ij}(\theta))\Big|_{\theta=0}$.
    - △ Set $P^c(i^\star, j^\star) = 0$ and normalize the $i^\star$ th row of P^c.
    - △ Set $\mathbb{E} = \mathbb{E} \setminus \{(i^\star, j^\star)\}$.
    - △ If SC = True:
      - ⋆ Set $P^c(j^\star, i^\star) = 0$ and normalize the $j^\star$th row of P^c.
      - ⋆ Set $\mathbb{E} = \mathbb{E} \setminus \{(j^\star, i^\star)\}$.
  - → End While
- End While

OUTPUT:
P^c = Decomposed Markov chain transition matrix.

Condition	Label	Specification
CO_A	CO_A_1(i)	Number of times performed $< i$
	CO_A_2(E)	Number of ergodic classes in P^c is $< E$
CO_B	CO_B_1(e)	Number of edges cut is $< e$
	CO_B_2(E)	Number of ergodic classes in P^c is $< E$
	CO_B_3(q)	Not all edges with $\frac{d}{d\theta} K(P^c_{ij}(\theta)) < q$ are cut

Figure: Possible conditions for CO_A and CO_B

Recommendations from the authors:

- Large dimensions: CO_A = CO_A_1(1);
- Known ergodic classes: CO_B = CO_B_2(E).

Applications

Definition (Nearly complete decomposable Markov chain)

A Markov chain P is called *nearly complete decomposable* if P is irreducible and, up to permutations, can be written as shown below, where $P_{ii}, i = 1, \ldots, k$, are square matrices with rows summing up to $1 - \epsilon$.

$$P = \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1k} \\ P_{21} & P_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & P_{(k-1)k} \\ P_{k1} & \cdots & P_{k(k-1)} & P_{kk} \end{bmatrix}$$

Nearly complete decomposable Markov chain

Courtois matrix

$$P = \begin{bmatrix} .85 & 0 & .149 & .0009 & 0 & .00005 & 0 & .00005 \\ .1 & .65 & .249 & 0 & .0009 & .00005 & 0 & .00005 \\ .1 & .8 & .0996 & .0003 & 0 & 0 & .0001 & 0 \\ 0 & .0004 & 0 & .7 & .2995 & 0 & .0001 & 0 \\ .0005 & 0 & .0004 & .399 & .6 & .0001 & 0 & 0 \\ 0 & .00005 & 0 & 0 & .00005 & .6 & .2499 & .15 \\ .00003 & 0 & .00003 & .00004 & 0 & .1 & .8 & .0999 \\ 0 & .00005 & 0 & 0 & .00005 & .1999 & .25 & .55 \end{bmatrix},$$

with stationary probability vector:

$$\pi_P^\top = \begin{bmatrix} 0.089 & 0.093 & 0.04 & 0.159 & 0.119 & 0.12 & 0.278 & 0.102 \end{bmatrix}.$$

Applying `KDA(`P`,CO_A_2(2), CO_B_1(1), FALSE)`

$$\Pi_{P^c} = \begin{bmatrix} .175 & .182 & .08 & .322 & .241 & 0 & 0 & 0 \\ .175 & .182 & .08 & .322 & .241 & 0 & 0 & 0 \\ .175 & .182 & .08 & .322 & .241 & 0 & 0 & 0 \\ .175 & .182 & .08 & .322 & .241 & 0 & 0 & 0 \\ .175 & .182 & .08 & .322 & .241 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .241 & .556 & .204 \\ 0 & 0 & 0 & 0 & 0 & .241 & .556 & .204 \\ 0 & 0 & 0 & 0 & 0 & .241 & .556 & .204 \end{bmatrix}$$

Nearly complete decomposable Markov chain

Courtois matrix

$$P = \begin{bmatrix} .85 & 0 & .149 & .0009 & 0 & .00005 & 0 & .00005 \\ .1 & .65 & .249 & 0 & .0009 & .00005 & 0 & .00005 \\ .1 & .8 & .0996 & .0003 & 0 & 0 & .0001 & 0 \\ 0 & .0004 & 0 & .7 & .2995 & 0 & .0001 & 0 \\ .0005 & 0 & .0004 & .399 & .6 & .0001 & 0 & 0 \\ 0 & .00005 & 0 & 0 & .00005 & .6 & .2499 & .15 \\ .00003 & 0 & .00003 & .00004 & 0 & .1 & .8 & .0999 \\ 0 & .00005 & 0 & 0 & .00005 & .1999 & .25 & .55 \end{bmatrix},$$

with stationary probability vector:

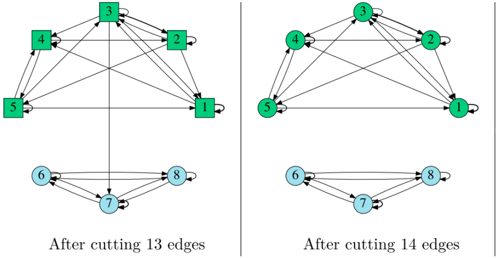
$$\pi_P^\top = \begin{bmatrix} 0.089 & 0.093 & 0.04 & 0.159 & 0.119 & 0.12 & 0.278 & 0.102 \end{bmatrix}.$$

Applying `KDA(`P`,CO_A_2(3), CO_B_1(1), FALSE)`

$$\Pi_{P^c} = \begin{bmatrix} .402 & .417 & .182 & 0 & 0 & 0 & 0 & 0 \\ .402 & .417 & .182 & 0 & 0 & 0 & 0 & 0 \\ .402 & .417 & .182 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .571 & .429 & 0 & 0 & 0 \\ 0 & 0 & 0 & .571 & .429 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .241 & .556 & .204 \\ 0 & 0 & 0 & 0 & 0 & .241 & .556 & .204 \\ 0 & 0 & 0 & 0 & 0 & .241 & .556 & .204 \end{bmatrix}$$

Nearly complete decomposable Markov chain

Applying `KDA(`P`,CO_A_2(3), CO_B_1(1), FALSE)`:

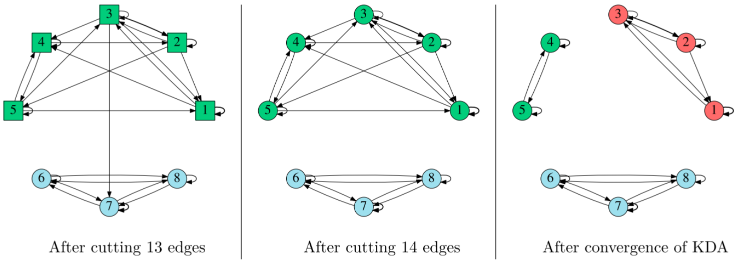


Remark

- $P(4,7) = .0001 > P(1,8) = .00005$ but $(4,7)$ is cut first;

Nearly complete decomposable Markov chain

Applying KDA(P,CO_A_2(3), CO_B_1(1), FALSE):



Remark

- $P(4,7) = .0001 > P(1,8) = .00005$ but $(4,7)$ is cut first;
- KDA(P,CO_A_2(2), CO_B_1(1), FALSE) = KDA(P,CO_A_1(1), CO_B_2(2), FALSE);
- KDA(P,CO_A_2(3), CO_B_1(1), FALSE) = KDA(P,CO_A_1(1), CO_B_2(3), FALSE).

Data Clustering

Consider n data vectors of arbitrary dimension $x_1, ..., x_n$. The number of clusters C is unknown. Let S be the similarity matrix such that: $S(i,j) =$ similarity between x_i and x_j. Normalizing, we obtain the transition matrix P.
We use the Gaussian similarity function:

$$S(i,j) = exp\Big(- \frac{\|x_i - x_j\|^2}{\sigma^2/\delta}\Big), \quad \text{for all } i,j = 1, ..., n,$$

where,

$$\delta > 0, \qquad \sigma^2 = \frac{1}{n-1}\sum_{i=1}^{n} \|x_i - \mu\|^2 \qquad \text{and} \qquad \mu = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

$\delta > 0$ is a user-chosen parameter that controls the width σ^2/δ of the neighborhoods of the data vectors.

Data Clustering

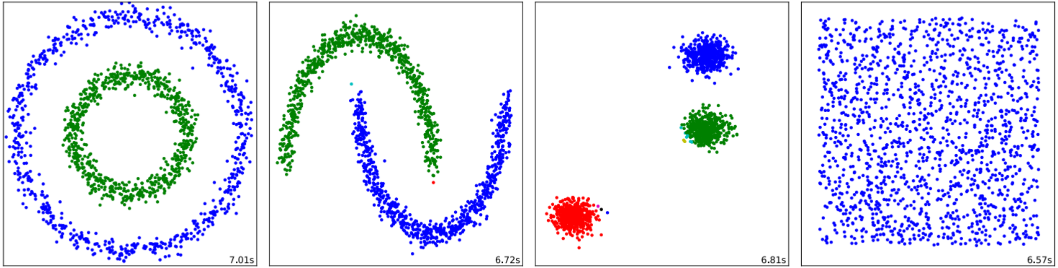
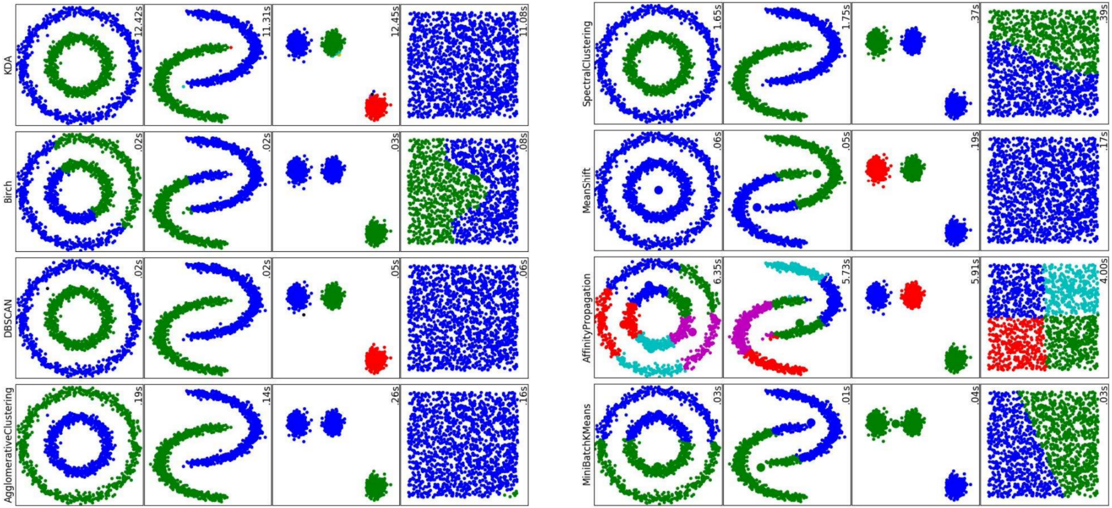


Figure: `KDA(`P`, CO_A_1(1), CO_B_3(0), TRUE)` Applied to Four Different Data Sets ($\delta = 6.5$)

Data Clustering



Conclusions

We have seen how the Kemeny constant and its derivative are good indicators of network connectivity. In particular, the Kemeny decomposition algorithm is able to identify the most significant edges in the network dynamics.
KDA has a wide range of applications and in the future, it could be applied to large real-life networks. In the future, the change in sign of the derivative of the Kemeny constant could be applied as a natural stopping criterion, and possible relationships between KDA and DBSCAN could be explored.

Thank you for your attention!