Directed Graph Laplacian Applications and Computations

D L Boley
University of Minnesota

Joint work with Golshan Golnari, Alejandro Buendia, Jacob Cadavez, Zhi-Li Zhang.

Graph Analysis: Random Walk Model

- Many properties of a graph can be obtained or estimated from properties of the so-called Fundamental Tensor derived using Random Walk model.
  - average hitting times, commute times.
  - distances or affinities between nodes.
  - betweenness measures.
  - importance/centrality measures.
  - bottlenecks in computer communication networks, road networks.
  - influence propagation.
- Much existing theory is for undirected graphs
- Some can be extended to directed graphs.
- Much of this material is from [Boley et al., 2010; Boley et al., 2018; Golnari et al., 2019].

Undirected vs Directed graphs

Undirected Graph

- social networks:
  friends and contact lists
- passive electrical networks
- recommender systems:
  e.g. bipartite graph:
  users $\leftrightarrow$ movies.
- the internet, computer communication networks.

Directed Graph

- the WWW: random walk on relaxed graph yields pagerank.
- road network with one-way streets.
- wireless device network with mix of high and low-powered devices.
- propagation of influence or trust in social networks.

Basics: Graphs and Matrices

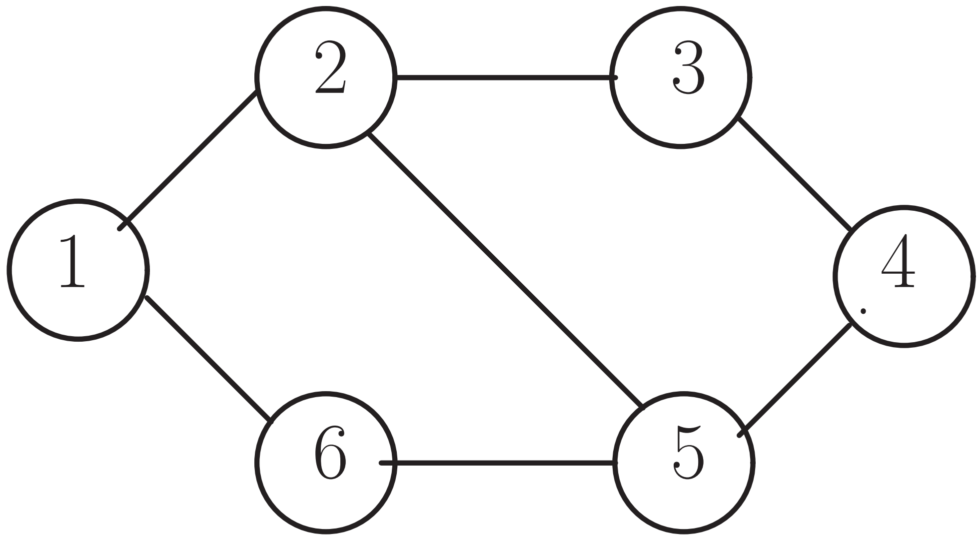
- Graph represented by
  - Adjacency Matrix A s.t. $a_{ij} \neq 0$ when $\exists$ an edge $i \to j$.
  - Markov chain transition matrix P s.t. p_{ij} = probability of transition from node i to node j.
  - Undirected graph $\Longleftrightarrow$ symmetric adjacency matrix $\Longleftrightarrow$ reversible Markov chain.
  - Assume no absorbing states $\Longleftrightarrow$ strongly connected.
- Related Quantities
  - $\mathbf{d} = A \cdot \mathbf{1}$ vector of node (out) degrees,
  - $D = \mathrm{diag}(\mathbf{d})$ = diagonal matrix of degrees,
  - $\boldsymbol{\pi}$ = vector of stationary probabilities, s.t. $\boldsymbol{\pi}^{\mathrm{T}} P = \boldsymbol{\pi}^{\mathrm{T}}$,
  - Π = diagonal matrix of stationary probabilities,
  - $Z = (I - P + \mathbf{1}\boldsymbol{\pi}^{\mathrm{T}})^{-1}$ = Fundamental Matrix [Grinstead & Snell, 2006].

Alternative Laplacians

Laplacians lead to many graph properties (many for undirected graphs)

- $L^{\mathrm{a}} = D - A = D(I - P)$ "combinatorial," based on node degrees.
  - Matrix Tree Theorem $\rightarrow$ number of spanning 'trees' anchored at each node (DiGraphs too) [Brualdi & Ryser, 1991; Chebotarev & Shamis, 2006]
  - smallest graph cut relative to number of nodes in each half [Shi & Malik, 2000; Spielman & Teng, 1996; von Luxburg, 2007].
- $L = \Pi(I - P)$ "Random Walk" $= L^{\mathrm{a}} \cdot$ vol*2 if undirected.
  - pseudo-inverse leads to average commute times/resistances [Doyle & Snell, 1984; Chandra et al., 1989; Klein & Randic, 1993; Boley et al., 2011].
  - pseudo-inverse leads to metric embedding in $\mathbb{R}^n$ [Gower & Legendre, 1986; Fouss et al., 2007].
- $L^{\mathrm{p}} = I - P = I - D^{-1}A = D^{-1}L^{\mathrm{a}}$ "normalized"
  - smallest graph cut relative to number of edges in each half [von Luxburg, 2007].
  - Consensus dynamics over nodes of a graph: $\dot{\mathbf{x}} = -L\mathbf{x}$ (DiGraphs too). [Olfati-Saber et al., 2004, 2006], [Bamieh et al., 2008], [Young et al., 2010, 2011].
- $\mathcal{L} = D^{1/2} L^{\mathrm{p}} D^{-1/2} = D^{-1/2} L^{\mathrm{a}} D^{-1/2} =$ symmetrized normalized Laplacian.
  - shares same eigenvalues as $L^{\mathrm{p}} = I - P$.

Example – Undirected Graph



$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \qquad \mathbf{d} = \begin{pmatrix} 2 \\ 3 \\ 2 \\ 2 \\ 3 \\ 2 \end{pmatrix} \qquad \boldsymbol{\pi} = \frac{1}{14} \cdot \begin{pmatrix} 2 \\ 3 \\ 2 \\ 2 \\ 3 \\ 2 \end{pmatrix}$$

Laplacians

- $L^{\mathrm{a}} = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 0 & -1 & 3 & -1 \\ -1 & 0 & 0 & 0 & -1 & 2 \end{pmatrix} = 14 * L$
- Number of spanning 'trees': $\det(L^{\mathrm{a}}_{[2:6],[2:6]}) = 15$.
- Eigenvalues are 0, 1, 2, 3, 3, 5.
- Eigenvector corresp. to 1 (Fiedler vector): $(1, 0, -1, -1, 0, 1)/2$. Used in Spectral Graph Partitioning.
- Volume = number of edges = $^1\!/_2\mathrm{trace}(L^{\mathrm{a}}) = 7$.

Fundamental Tensor: Number of Visits

$\boldsymbol{N}(i, j, k)$ for all i, j, k.
$=$ average number of visits to j starting from i before reaching k.

Fundamental Tensor: Number of Visits

- Partition $P = \left[\begin{array}{c|c} P_{11} & \mathbf{p}_{12} \\ \hline \mathbf{p}_{21}^T & p_{nn} \end{array}\right]$.

- If last row replaced with $[\mathbf{0}^T, 1]$, then $[P_{11}^k]_{ij}$ is the probability of being in node j starting in node i at the $k-th$ step, before reaching n.

- $[I + P_{11} + P_{11}^2 + \cdots]_{ij} = [(I - P_{11})^{-1}]_{ij} \stackrel{\text{def}}{=} \mathbf{N}(i,j,n)$
$= \#$ visits to j starting from i before reaching n.

- $(I - P_{11})^{-1} = [\Pi_{1,\ldots,n-1}^{-1} \underbrace{\Pi_{1,\ldots,n-1}(I - P_{11})}_{L_{11}}\ ]^{-1} = L_{11}^{-1}\Pi_{1,\ldots,n-1}$.

- Since $L \cdot \mathbf{1} = \mathbf{0}$, $\mathbf{1}^T L = \mathbf{0}^T$, can write $(I - P_{11})^{-1}$ in terms of $M \stackrel{\text{def}}{=} L^+$ to yield $\mathbf{N}(i,j,n) = (m_{ij} + m_{nn} - m_{in} - m_{nj})\pi_j$.

- Choice of destination node n is arbitrary, so have Tensor:
$\mathbf{N}(i,j,k) = (m_{ij} + m_{kk} - m_{ik} - m_{kj})\pi_j$ for all i, j, k.
$=$ average number of visits to j starting from i before reaching k.

Tensor

$$\boldsymbol{N} =$$

```
k=1                                      k=4
 ( 0    0    0    0    0    0   )         (2.80 2.40 0.80  0    1.80 2.00)
 ( 0    2.20 1.33 1.20 1.60 0.53)         (1.60 2.80 0.93  0    1.60 1.33)
 ( 0    2.00 2.67 2.00 2.00 0.67)         (0.80 1.40 1.47  0    0.80 0.67)
 ( 0    1.80 2.00 2.80 2.40 0.80)         ( 0    0    0    0    0    0   )
 ( 0    1.60 1.33 1.60 2.80 0.93)         (1.20 1.60 0.53  0    2.20 1.33)
 ( 0    0.80 0.67 0.80 1.40 1.47)         (2.00 2.00 0.67  0    2.00 2.67)

k=2                                      k=5
 (1.47  0    0.13 0.27 0.60 0.93)         (1.87 1.20 0.53 0.27  0    0.93)
 ( 0    0    0    0    0    0   )         (0.80 1.80 0.80 0.40  0    0.40)
 (0.13  0    1.47 0.93 0.60 0.27)         (0.53 1.20 1.87 0.93  0    0.27)
 (0.27  0    0.93 1.87 1.20 0.53)         (0.27 0.60 0.93 1.47  0    0.13)
 (0.40  0    0.40 0.80 1.80 0.80)         ( 0    0    0    0    0    0   )
 (0.93  0    0.27 0.53 1.20 1.87)         (0.93 0.60 0.27 0.13  0    1.47)

k=3                                      k=6
 (2.67 2.00  0    0.67 2.00 2.00)         (1.47 1.40 0.80 0.67 0.80  0   )
 (1.33 2.20  0    0.53 1.60 1.20)         (0.93 2.80 1.60 1.33 1.60  0   )
 ( 0    0    0    0    0    0   )         (0.80 2.40 2.80 2.00 1.80  0   )
 (0.67 0.80  0    1.47 1.40 0.80)         (0.67 2.00 2.00 2.67 2.00  0   )
 (1.33 1.60  0    0.93 2.80 1.60)         (0.53 1.60 1.20 1.33 2.20  0   )
 (2.00 1.80  0    0.80 2.40 2.80)         ( 0    0    0    0    0    0   )
```

- $\boldsymbol{N}(i,j,k)$ = average number of visits to j on the way from i to k.
- $\boldsymbol{N}(i,j,k)/\boldsymbol{N}(j,j,k)$ = prob. of passing j on the way from i to k.

Get Pseudo-Inv of Laplacian

1. Compute normalized Laplacian $L = I - P$.
2. Compute inverse of the upper $(n-1) \times (n-1)$ part: $I - P_{11}$
3. Solve for the stationary probabilites: $(\pi_1, \ldots, \pi_{n-1}) = -(L^{\mathrm{p}}_{11})^{-1} \ell^{\mathrm{p}}_{12} \pi_n$;
4. Form random walk Laplacian $\mathbf{L} = \textsc{Diag}(\boldsymbol{\pi}) \cdot L = \mathbf{\Pi}(I - P)$.
5. Compute the inverse of $\mathbf{L}_{11}^{-1} = (I - P_{11})^{-1} \mathbf{\Pi}_1^{-1}$
6. Compute desired pseudoinverse $\mathbf{M}$

$$\mathbf{M} = \begin{pmatrix} R_{\mathbf{1}} \\ \frac{-1}{n} \mathbf{1}^T \end{pmatrix} \mathbf{L}_{11}^{-1} \left(R_{\mathbf{1}}, \ \frac{-1}{n} \mathbf{1} \ \right),$$

   where $R_{\mathbf{1}} = (I_{n-1} - \frac{1}{n} \mathbf{1}\mathbf{1}^T)$.

7. $\mathbf{N}(i,j,k) = (m_{ij} + m_{kk} - m_{ik} - m_{kj})\pi_j$ for all i, j, k.

Get Pseudo-Inv of Laplacian

1. Compute normalized Laplacian $L = I - P$.
2. $\boxed{\text{Compute inverse of the upper } (n-1)\times(n-1) \text{ part: } I - P_{11}}$ Expensive
3. Solve for the stationary probabilites: $(\pi_1, \ldots, \pi_{n-1}) = -(L_{11}^{\mathrm{p}})^{-1}\ell_{12}^{\mathrm{p}}\pi_n$;
4. Form random walk Laplacian $\mathbf{L} = \mathrm{D{\scriptsize IAG}}(\boldsymbol{\pi}) \cdot L = \mathbf{\Pi}(I - P)$.
5. Compute the inverse of $\mathbf{L}_{11}^{-1} = (I - P_{11})^{-1}\mathbf{\Pi}_1^{-1}$
6. Compute desired pseudoinverse $\mathbf{M}$

$$\mathbf{M} = \begin{pmatrix} R_{\mathbf{1}} \\ \frac{-1}{n}\mathbf{1}^T \end{pmatrix} \mathbf{L}_{11}^{-1} \left(R_{\mathbf{1}}, \ \frac{-1}{n}\mathbf{1} \ \right),$$

   where $R_{\mathbf{1}} = (I_{n-1} - \frac{1}{n}\mathbf{1}\mathbf{1}^T)$.

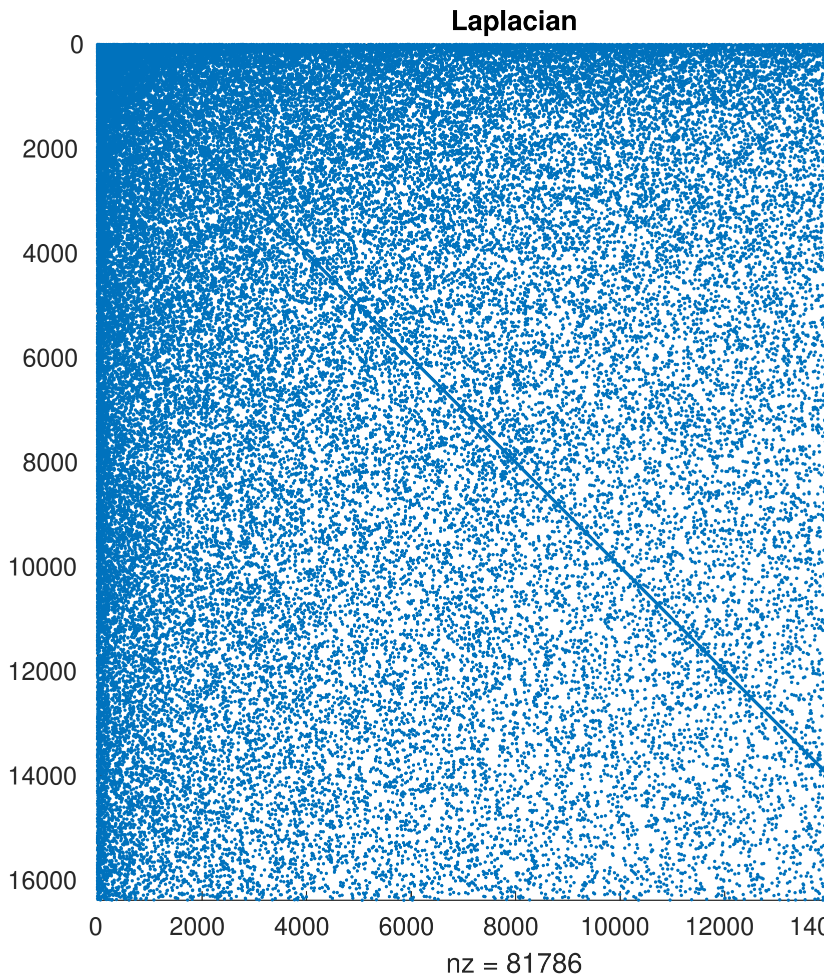
7. $\boldsymbol{N}(i,j,k) = (m_{ij} + m_{kk} - m_{ik} - m_{kj})\pi_j$ for all i, j, k.

Other Relations for Pseudo-Inv

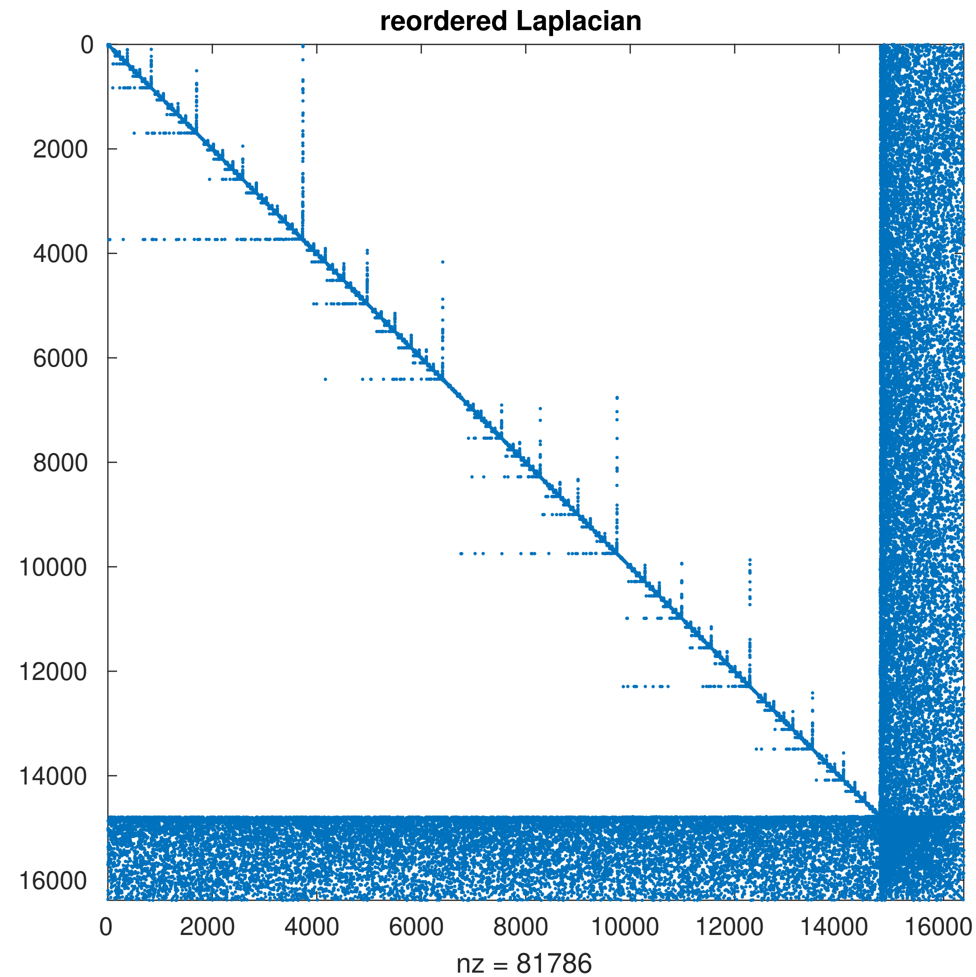
- $L^+ = (L + \mathbf{v}\mathbf{v}^T)^{-1} - \mathbf{v}\mathbf{v}^T$
  - where $L\mathbf{v} = Ł^T\mathbf{v} = 0$, $\mathbf{v}^T\mathbf{v} = 1$, $(L + \mathbf{v}\mathbf{v}^T)^{-1}$ exists.
  - If right & left nullspaces don't match, apply diagonal scaling.

- If A is an invertible matrix, can use Schur complement to get inverse of top left block in terms of entire inverse, and vv:
  - $(A^{-1})_{11} = (A_{11} - A_{12} * (A_{22})^{-1} * A_{21})^{-1}$
  - $(A_{11})^{-1} = (A^{-1})_{11} - (A^{-1})_{12} * (inv(A^{-1})_{22})^{-1} * (A^{-1})_{21}$

Re-order Laplacian for Small World Graphs



original order



approx minimum degree ordering

Cost for Small World Graphs

number of			time in csec	
vertices	edges	LU fill	LU	backsolve
1,024	4,059	20,620	5	2
2,048	8,140	66,851	2	< 1
4,096	16,314	205,826	4	< 1
8,192	32,671	763,440	12	1
16,384	65,402	2,804,208	56	5
32,768	130,884	10,740,194	250	19
65,536	261,882	43,504,911	1,363	82
131,072	523,920	168,455,437	7,989	328

- Double the size $\Longrightarrow$
  LU cost grows by about a factor of 5 instead of a factor of $2^3 = 8$.

Hitting and Commute Times

Adding up previous gives

- $\mathbf{H}(i,k) = \sum_j \mathbf{N}(i,j,k) = m_{kk} - m_{ik} + \sum_j (m_{ij} - m_{kj})\pi_j$
  - $\mathbf{H}(i,n) = (I - L_{11})^{-1}\mathbf{1}$
- $\mathbf{C}(i,k) = \mathbf{H}(i,k) + \mathbf{H}(k,i) = m_{kk} + m_{ii} - m_{ik} - m_{ki}$.
- Above holds also for strongly connected directed graphs (arbitrary Markov chain with no transient states).
- Could add along other dimensions to get betweenness measures, etc.

Commute Times

- Pseudo inverse of $L = L^{\mathrm{a}}/14$ is a Gram matrix:

$$M = L^{+} = \frac{7}{90} \cdot \begin{pmatrix} \mathbf{83} & -1 & -37 & -43 & -19 & 17 \\ -1 & \mathbf{47} & -1 & -19 & -7 & -19 \\ -37 & -1 & \mathbf{83} & 17 & -19 & -43 \\ -43 & -19 & 17 & \mathbf{83} & -1 & -37 \\ -19 & -7 & -19 & -1 & \mathbf{47} & -1 \\ 17 & -19 & -43 & -37 & -1 & \mathbf{83} \end{pmatrix}$$

- $\Longrightarrow$ expected commute times in random walk $[(\ell_2 \text{ metric})^2]$

$$\mathbf{C} = \begin{bmatrix} \mathrm{diag}(L^{+}) \cdot \mathbf{1}^{\mathrm{T}} \\ + \mathbf{1} \cdot \mathrm{diag}(L^{+}) \\ - L^{+} - (L^{+})^{\mathrm{T}} \end{bmatrix} = \frac{14}{15} \cdot \begin{pmatrix} 0 & 11 & 20 & 21 & 14 & 11 \\ 11 & 0 & 11 & 14 & 9 & 14 \\ 20 & 11 & 0 & 11 & 14 & 21 \\ 21 & 14 & 11 & 0 & 11 & 20 \\ 14 & 9 & 14 & 11 & 0 & 11 \\ 11 & 14 & 21 & 20 & 11 & 0 \end{pmatrix}.$$

- Red numbers: average extra cost of detour thru given node.

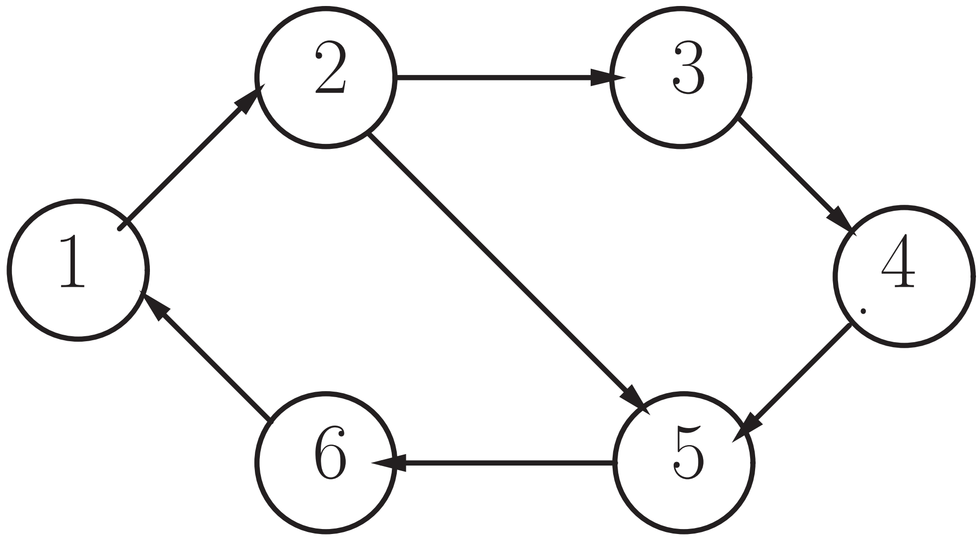
Embedding

- $L^+ = \mathbf{S}^T\mathbf{S}$ with

$$\mathbf{S} = \begin{array}{c} \begin{array}{cccccc} \mathbf{s}_1 & \mathbf{s}_2 & \mathbf{s}_3 & \mathbf{s}_4 & \mathbf{s}_5 & \mathbf{s}_6 \end{array} \\ \begin{pmatrix} 2.5408 & -.0306 & -1.1326 & -1.3163 & -.5816 & .52040 \\ 0 & 1.9117 & -.0588 & -.7941 & -.2941 & -.7647 \\ 0 & 0 & 2.2736 & -.0947 & -.9473 & -1.2315 \\ 0 & 0 & 0 & 2.02070 & -.5774 & -1.4434 \\ 0 & 0 & 0 & 0 & 1.4142 & -1.4142 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

- For all i, j, $\|\mathbf{s}_i - \mathbf{s}_j\|_2^2 = C_{ij}$.
- Since $L^+\mathbf{1} = \mathbf{0}$, the columns of $\mathbf{S}$ are already centered.
- Previous red numbers are distance^2 from origin $=$ Centrality [83, 47, 83, 83, 47, 83] $\times$ $(7/90)$.

Example – Directed Graph



$$P = \begin{pmatrix} 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \\ 1.0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{d} = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \quad \not\propto \quad \boldsymbol{\pi} = \begin{pmatrix} 0.2 \\ 0.2 \\ 0.1 \\ 0.1 \\ 0.2 \\ 0.2 \end{pmatrix}$$

Laplacian from Probabilities

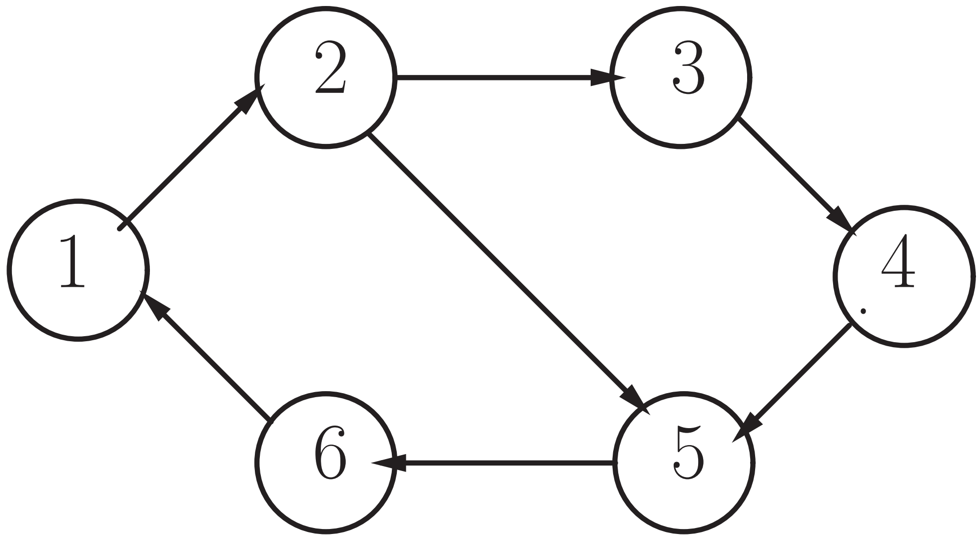
- Obtain Tensor & commute times same way, but from $L = \Pi - \Pi P$:

$$L = \begin{pmatrix} 0.2 & -0.2 & 0 & 0 & 0 & 0 \\ 0 & 0.2 & -0.1 & 0 & -0.1 & 0 \\ 0 & 0 & 0.1 & -0.1 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & -0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.2 & -0.2 \\ -0.2 & 0 & 0 & 0 & 0 & 0.2 \end{pmatrix}, \begin{matrix}\text{null} \\ \text{vec}\end{matrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$M = L^{+} = \frac{5}{6} \begin{pmatrix} \mathbf{3} & 2 & 0 & -2 & -1 & -2 \\ -2 & \mathbf{3} & 1 & -1 & 0 & -1 \\ -3 & -4 & \mathbf{6} & 4 & -1 & -2 \\ -1 & -2 & -4 & \mathbf{6} & 1 & 0 \\ 1 & 0 & -2 & -4 & \mathbf{3} & 2 \\ 2 & 1 & -1 & -3 & -2 & \mathbf{3} \end{pmatrix}$$

Laplacians: only $\Pi - \Pi P$ has null vector $(1, \ldots, 1)$ on both sides.

Hitting & Commute Times



H (hitting times)

$$\begin{pmatrix} 0 & \mathbf{1} & \mathbf{6} & \mathbf{7} & \mathbf{3} & \mathbf{4} \\ 4 & 0 & 5 & 6 & 2 & 3 \\ 4 & 5 & 0 & 1 & 2 & 3 \\ 3 & 4 & 9 & 0 & 1 & 2 \\ 2 & 3 & 8 & 9 & 0 & 1 \\ 1 & 2 & 7 & 8 & 4 & 0 \end{pmatrix}$$

C (commute times)

$$\begin{pmatrix} 0 & \mathbf{5} & \mathbf{10} & \mathbf{10} & \mathbf{5} & \mathbf{5} \\ 5 & 0 & 10 & 10 & 5 & 5 \\ 10 & 10 & 0 & 10 & 10 & 10 \\ 10 & 10 & 10 & 0 & 10 & 10 \\ 5 & 5 & 10 & 10 & 0 & 5 \\ 5 & 5 & 10 & 10 & 5 & 0 \end{pmatrix}$$

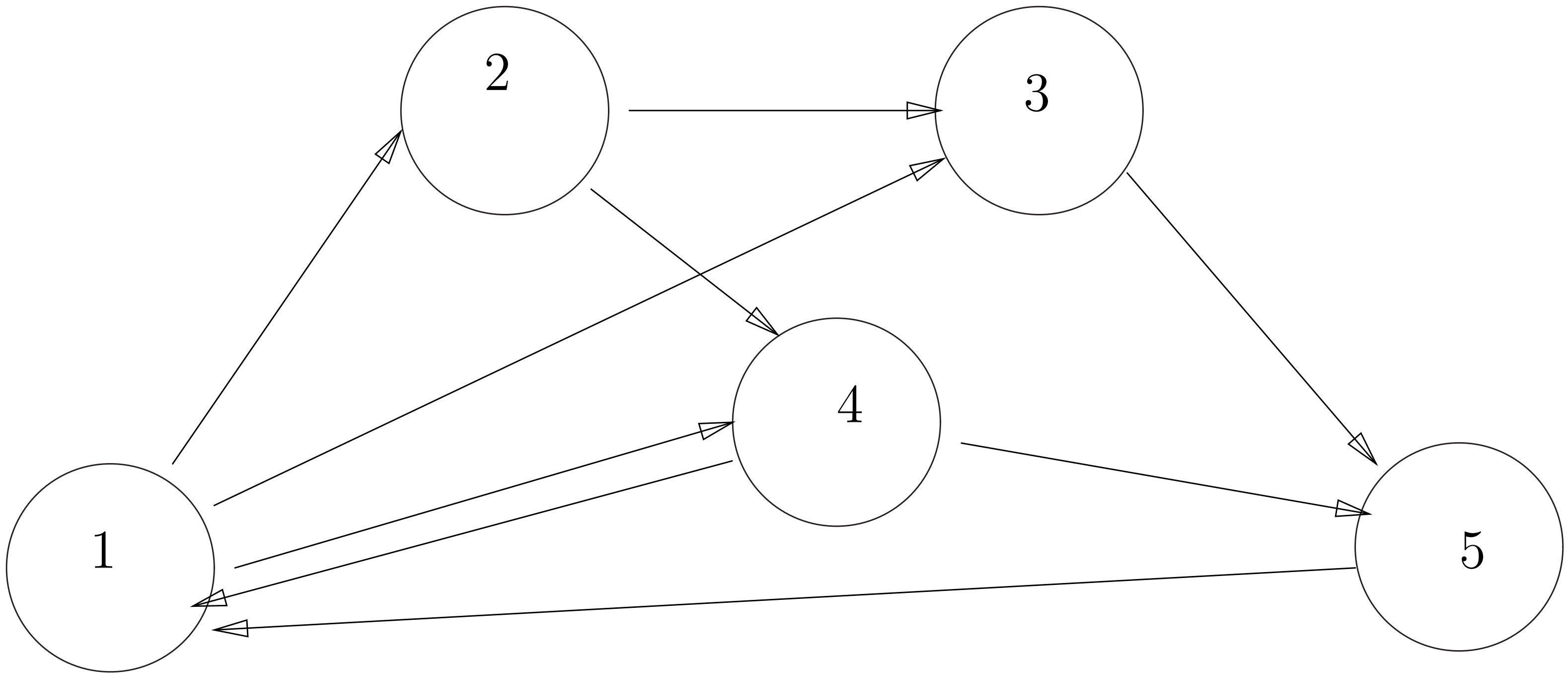
- Only nodes 3, 4 are peripheral. Others are all equally important.
- Same reflected in average commute times from node 2.

Tensor

$N =$

```
k=1
 (   0    0    0    0    0    0)
 (   0  1.0  0.5  0.5  1.0  1.0)
 (   0    0  1.0  1.0  1.0  1.0)
 (   0    0    0  1.0  1.0  1.0)
 (   0    0    0    0  1.0  1.0)
 (   0    0    0    0    0  1.0)

k=2
 (1.0    0    0    0    0    0)
 (   0    0    0    0    0    0)
 (1.0    0  1.0  1.0  1.0  1.0)
 (1.0    0    0  1.0  1.0  1.0)
 (1.0    0    0    0  1.0  1.0)
 (1.0    0    0    0    0  1.0)

k=3
 (2.0  2.0    0    0  1.0  1.0)
 (1.0  2.0    0    0  1.0  1.0)
 (   0    0    0    0    0    0)
 (2.0  2.0    0  1.0  2.0  2.0)
 (2.0  2.0    0    0  2.0  2.0)
 (2.0  2.0    0    0  1.0  2.0)

k=4
 (2.0  2.0  1.0    0  1.0  1.0)
 (1.0  2.0  1.0    0  1.0  1.0)
 (   0    0  1.0    0    0    0)
 (   0    0    0    0    0    0)
 (2.0  2.0  1.0    0  2.0  2.0)
 (2.0  2.0  1.0    0  1.0  2.0)

k=5
 (1.0  1.0  0.5  0.5    0    0)
 (   0  1.0  0.5  0.5    0    0)
 (   0    0  1.0  1.0    0    0)
 (   0    0    0  1.0    0    0)
 (   0    0    0    0    0    0)
 (1.0  1.0  0.5  0.5    0  1.0)

k=6
 (1.0  1.0  0.5  0.5  1.0    0)
 (   0  1.0  0.5  0.5  1.0    0)
 (   0    0  1.0  1.0  1.0    0)
 (   0    0    0  1.0  1.0    0)
 (   0    0    0    0  1.0    0)
 (   0    0    0    0    0    0)
```

Example – Directed Graph



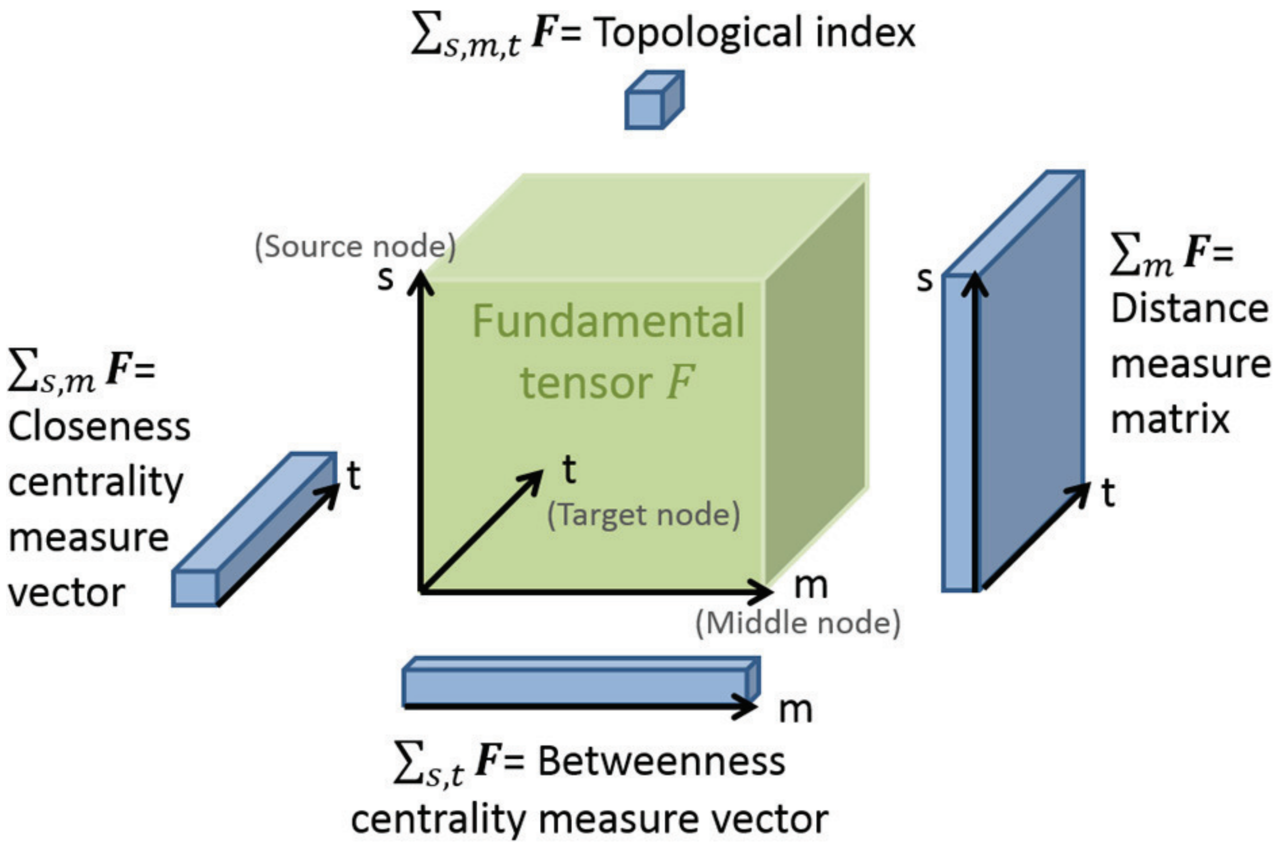

```
P =                                                     D =    π =
 (  0      .333333 .333333 .333333   0    )   (3)    (12)
 (  0        0     .500000 .500000   0    )   (2)    ( 4)      1
 (  0        0        0       0    1.00000)   (1)    ( 6)  *  --
 (.500000    0        0       0     .500000)  (2)    ( 6)     37
 (  1        0        0       0      0    )   (1)    ( 9)
```

Laplacian from Probabilities

- Obtain Tensor & commute times same way, but from $L = \Pi - \Pi P$:

```
L =       (     12      -4      -4      -4       0)
       1  (      0       4      -2      -2       0)
      --  (      0       0       6       0      -6)
      37  (     -3       0       0       6      -3)
          (     -9       0       0       0       9)

M =       (      8      -1      -1      -1      -5)
      37  (    -10      26      -4      -4      -8)
     ---  (     -2     -11      19     -11       5)
     180  (      0      -9      -9      21      -3)
          (      4      -5      -5      -5      11)
```

Tensor Applications



Tensor Applications

- Closeness(k): $\sum_{ij} \boldsymbol{N}(i,j,k)$
  (7.25000 30.5000 18.8333 21.3333 9.44444)

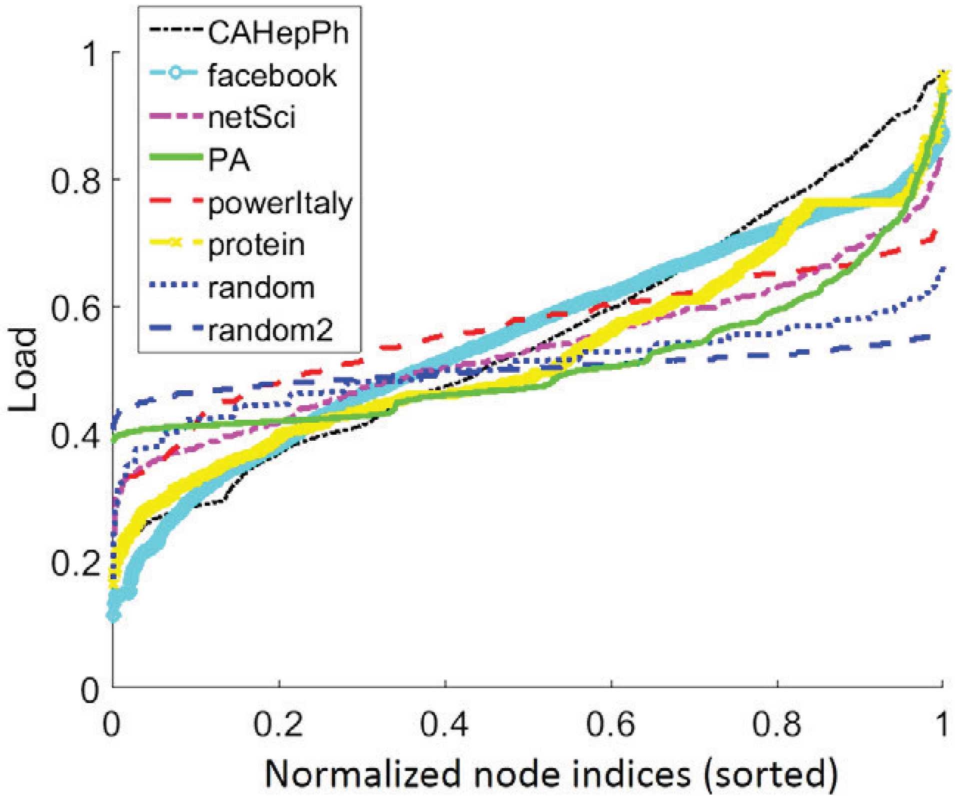
- Betweenness(j): $\sum_{ik} \boldsymbol{N}(i,j,k)$
  (28.3333 9.44444 14.1666 14.1666 21.2500)

- Probability of passage (i,j,k): $\widetilde{\boldsymbol{N}}(i,j,k) = \boldsymbol{N}(i,j,k)/\boldsymbol{N}(j,j,k)$
  [or zero if $i=k$ or $j=k$].

- node load: chance of passage averaged over all start/end nodes
  $\sum_{ik} \widetilde{\boldsymbol{N}}(i,j,k) \ / \ (n-1)^2$.
  (.796875 .475000 .572916 .531250 .748958)

- prob of passiing 2 when starting from 1 before reaching 5:
  .400 (full network); .500 (if 4 removed); .333 (if 4 is to be avoided)

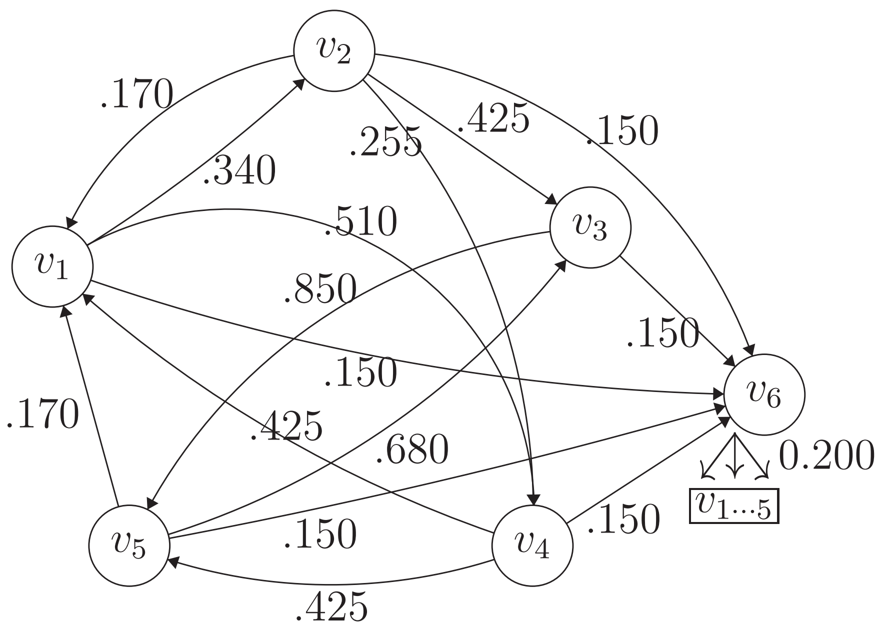
Load examples



- High Energy Physics Collaboration Network
- sampled Facebook network
- netSci: co-author network of network scientists
- Italian power grid
- protein-protein interaction network
- synthetic: Preferential Attachment
- synthetic: random Erdös-Rényi (8 or 40 initial links)

- Load is most balanced in ER network, less so in PA.
- Co-author networks: more imbalanced.
- power network: more balanced load.

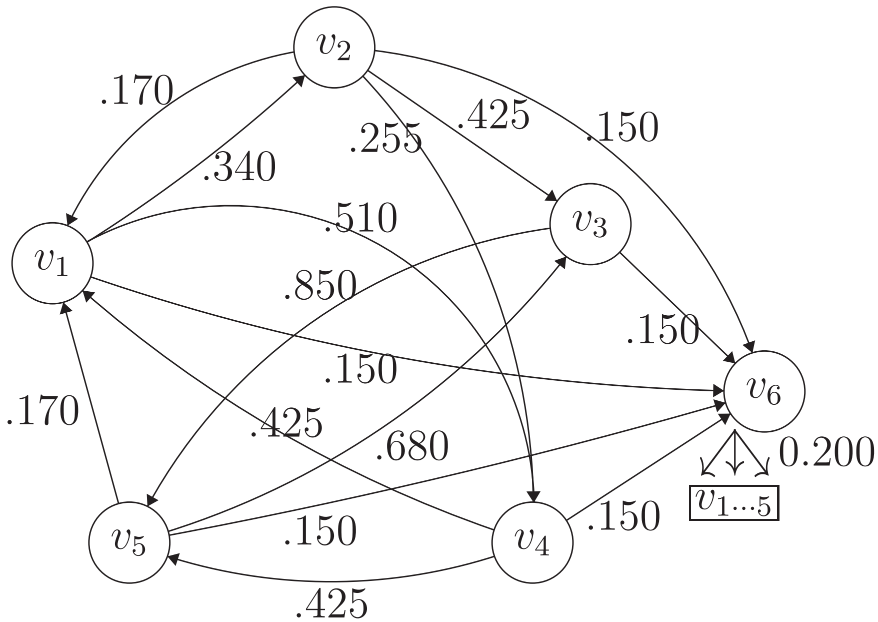
Influence Propagation



- Network shows how much agent i trusts agent j, for $i, j{=}1, \ldots, 5$ after direct interaction.
- Node 6 is extra evaporation node to control influence locality.

- Seek to predict trust between new pairs

Influence Propagation



- Network shows how much agent i trusts agent j, for i, j=1, ..., 5 after direct interaction.
- Node 6 is extra evaporation node to control influence locality.

- Seek to predict trust between new pairs: predicted trust of j by i = $\mathbf{PHT}(i, j) = \boldsymbol{N}(i, j, 6)/\boldsymbol{N}(j, j, 6)$. $\boxed{\text{A}}$
- Example: from point of view of agent 4:

$$\mathbf{PHT}(4, 1{:}5) = (0.60,\ 0.29,\ 0.53,\ 1.00,\ 0.66),$$

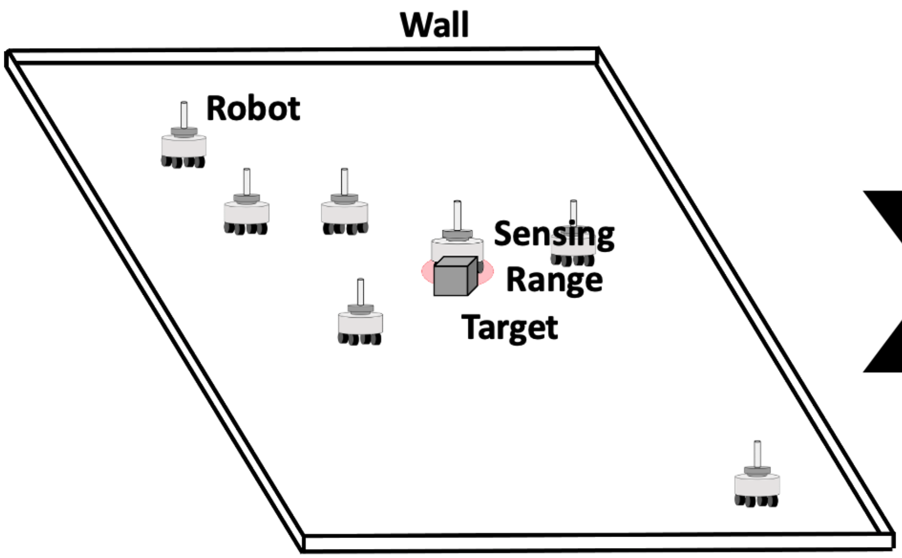
- If agent 2 is a bad actor to be avoided then we can use Fundamental Tensor to compute trust values based on avoiding 2:

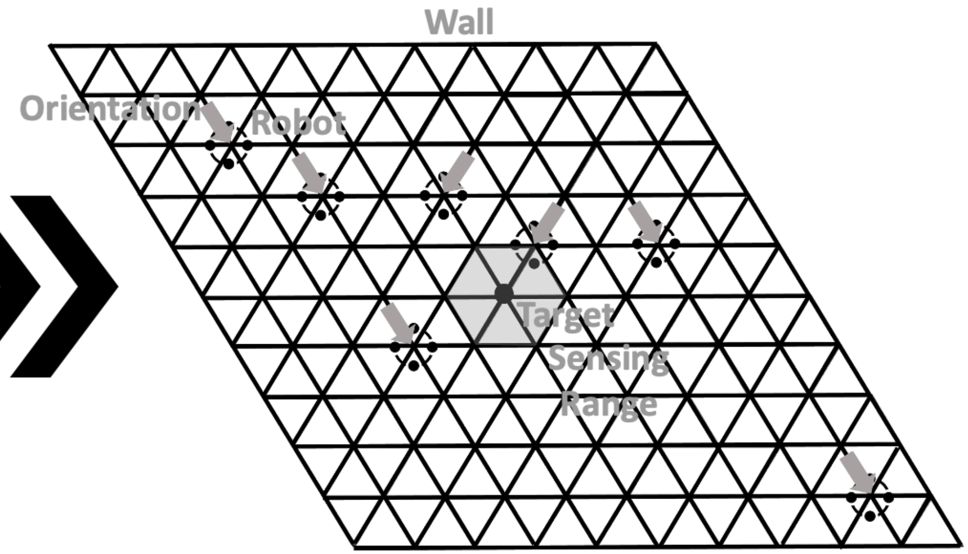
$$\mathbf{PHT}_2(4, 1{:}5) = (0.60,\ 0.00,\ 0.39,\ 1.00,\ 0.54)$$

[plug $\widetilde{\boldsymbol{N}} = \boldsymbol{N}(:,:,6) - \frac{\boldsymbol{N}(:,2,6)}{\boldsymbol{N}(2,2,6)}\boldsymbol{N}(2,:,6)$ into $\boxed{\text{A}}$.

- Now most trustworthy node for 4 is node 1 instead of node 5.

Correlated Random Walk for Robots





Continuous domain $\Rightarrow$ random walk on discrete graph

- Fast computation of aggregate properties
- Use Hitting Time as an example

- Triangles to get equal link lengths
- State = (position, orientation)

Probability Transition Matrix

- $\mathbf{P}_{ij}$ = probability of moving to node j from i.
- Goal nodes lead directly to absorbing state, put last.
- Partition matrix where $\boxed{\mathbf{P} = \begin{bmatrix} \mathbf{Q} & \mathbf{r} \\ \mathbf{0} & \mathit{1} \end{bmatrix}}$
  - $\mathbf{Q} \iff$ transient states
  - $\mathit{1} \iff$ absorbing state (nest)
  - $\mathbf{r} \iff$ transition from transient states to nest

Average Hitting Times

- $\mathbf{h}_\mu$ = vector of average Hitting Times (to reach nest) from any transient node
- $\mathbf{h}_{\sigma^2}$ = vector of variances of hitting times
- Obtain both values directly by solving system of linear equations [matrix of coefs = $(\mathbf{I} - \mathbf{Q})$]:

$$(\mathbf{I} - \mathbf{Q})\mathbf{h}_\mu = \mathbf{c} = \text{vector of all ones}$$

$$(\mathbf{I} - \mathbf{Q})\mathbf{h}_{\sigma^2} = 2\mathbf{h}_\mu - (\mathbf{I} - \mathbf{Q})\mathbf{h}_\mu - (\mathbf{I} - \mathbf{Q})\mathbf{h}_\mu^2$$

(Grinstead Snell 1997, Kemeny Snell 1976)

Solving The Linear System

- Dimension of system to solve: approx $6d^2 \times 6d^2$ where d = dimension of Arena
- System very sparse: at most 6 entries per row.
- Can be solved efficiently by iterative sparse matrix methods

arena size	matrix $\mathbf{P}$ dimension	number non-zeros	setup time (sec)	solve time (sec)
0×30	5767	31223	1.5	0.05
0×150	136807	807863	38.4	5.4
0×300	543607	3236663	44.9	35.6

Conclusions

- "Random Walk" Laplacian for strongly connected directed graph
- Fundamental Tensor - fast way to encode many properties
  - Centrality
  - Betweenness
  - Load/Bottleneck
  - Influence/trust Propagation

Conclusions

- "Random Walk" Laplacian for strongly connected directed graph
- Fundamental Tensor - fast way to encode many properties
  - Centrality
  - Betweenness
  - Load/Bottleneck
  - Influence/trust Propagation

Future Work

- Non-strongly connected graphs
- Fast computation of Laplacian pseudo-inverse
- Updates when nodes die or become pariahs.

Conclusions

- "Random Walk" Laplacian for strongly connected directed graph
- Fundamental Tensor - fast way to encode many properties
  - Centrality
  - Betweenness
  - Load/Bottleneck
  - Influence/trust Propagation

Future Work

- Non-strongly connected graphs
- Fast computation of Laplacian pseudo-inverse
- Updates when nodes die or become pariahs.

Thank You!

END END END END END END END END END

Interpretations – Undirected Graphs

- Commute times correspond to effective resistances.
  [Doyle & Snell, 1984; Chandra et al., 1989; Klein & Randic, 1993].

- Eigenvalues of

$$L^{\mathrm{ds}} = I - P = \frac{1}{6} \cdot \begin{pmatrix} 6 & -3 & 0 & 0 & 0 & -3 \\ -2 & 6 & -2 & 0 & -2 & 0 \\ 0 & -3 & 6 & -3 & 0 & 0 \\ 0 & 0 & -3 & 6 & -3 & 0 \\ 0 & -2 & 0 & -2 & 6 & -2 \\ -3 & 0 & 0 & 0 & -3 & 6 \end{pmatrix}$$

  are 0, $\boxed{1/2}$, 5/6, 7/6, 3/2, 2. The $\boxed{1/2}$ is related to the expander graph or Cheeger bound of the graph. [Chung, 2005; Zhou et al., 2005].

- Also $\boxed{1/2} \leftrightarrow$ mixing rate for random walk over the graph.

- The corresponding eigenvector used in spectral graph partitioning $(-1, 0, 1, 1, 0, -1)$.

Incidence Matrix

- The incidence matrix $\mathbf{N}$ has n columns and $vol(G)$ rows. Each column corresponds to a node (vertex) of graph G and each row corresponds to an edge (in some arbitrary order).

- The j-th row represents the edge $e_j = (i, j)$, and looks like

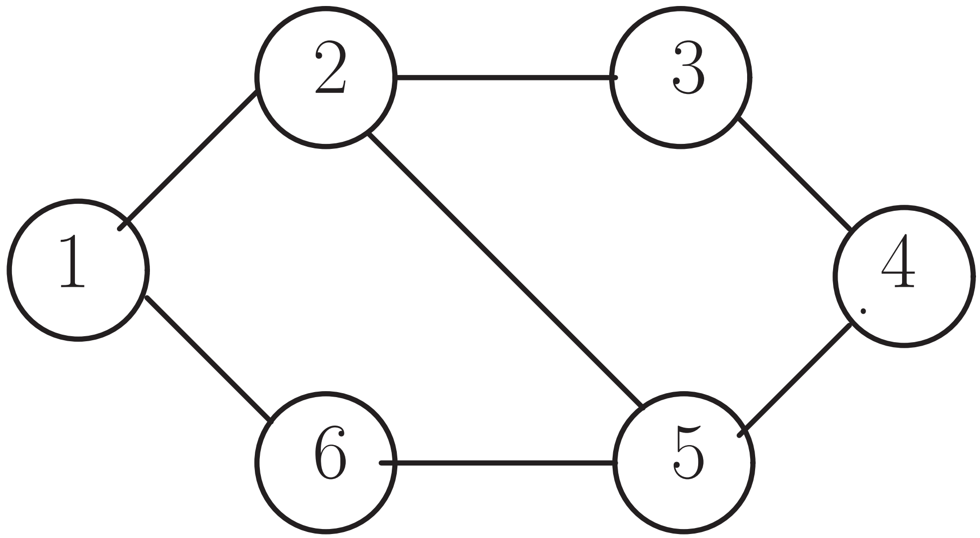
$$0, \ldots, 0, 1, 0, \ldots, 0, -1, 0, \ldots, 0$$

  where the nonzero entries are in columns i, j corresponding to the vertices connected by that edge.

- Then a simple calculation shows $L = D - A = \mathbf{N}^T\mathbf{N}$, where $A =$ adjacency matrix and $D =$ diagonal matrix of degrees.

- In general: if $\mathbf{v}$ is a vector of voltages, then $\mathbf{N}\mathbf{v}$ is the vector of currents across each link, assuming unit conductances.

Example Incidence Matrix



$$\mathbf{N} = \begin{pmatrix} -1 & +1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & +1 \\ 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 & -1 & 0 \\ 0 & 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & 0 & +1 & -1 \end{pmatrix}$$

Resistances

[Doyle & Snell, 1984; Chandra et al., 1989; Klein & Randic, 1993].

- Current = Incidence_matrix · Voltage (using unit resistances):

$$
\begin{array}{ccccc}
\mathbf{I} & = & \mathbf{N} & \cdot & \mathbf{V} \\
\begin{pmatrix} \mathrm{i}_1 \\ \vdots \\ \mathrm{i}_7 \end{pmatrix} & = & \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 1 & 0 \end{pmatrix} & \cdot & \begin{pmatrix} \mathrm{v}_1 \\ \vdots \\ \mathrm{v}_6 \end{pmatrix}
\end{array}
$$

- Kirchoff's law: If unit current is injected between nodes i & j, then net current through every other vertex must be zero:

$$\mathbf{e}_i - \mathbf{e}_j = \mathbf{N}^T \mathbf{I} = \cdots = \mathbf{N}^T \mathbf{N} \mathbf{v} = L^{\mathrm{a}} \mathbf{v}.$$

- Solve for voltages = $\mathbf{v} = (L^{\mathrm{a}})^+ (\mathbf{e}_i - \mathbf{e}_j)$.
- Net voltage drop i to j = effective resistance = $\mathrm{v}_i - \mathrm{v}_j = (\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}} \mathbf{v} = (\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}} (L^{\mathrm{a}})^+ (\mathbf{e}_i - \mathbf{e}_j)$.

Resistances

$\mathbf{e}_i - \mathbf{e}_j \perp \mathrm{Nullsp}(\mathbf{N}^T\mathbf{N})$, so can use pseudo-inverse to find voltages.

- Solve for voltages $\mathbf{v} = (\mathbf{N}^T\mathbf{N})^+ \cdot (\mathbf{e}_i - \mathbf{e}_j) = (L^{\mathrm{a}})^+(\mathbf{e}_i - \mathbf{e}_j)$.
- Effective resistance between nodes i & j is

$$
\begin{aligned}
\mathrm{v}_i - \mathrm{v}_j &= (\mathbf{e}_i^T - \mathbf{e}_j^T) \cdot \mathbf{v} \\
&= (\mathbf{e}_i^T - \mathbf{e}_j^T) \cdot (\mathbf{N}^T\mathbf{N})^+ \cdot (\mathbf{e}_i - \mathbf{e}_j) \\
&= (\mathbf{e}_i^T - \mathbf{e}_j^T) \cdot \quad (L^{\mathrm{a}})^+ \quad \cdot (\mathbf{e}_i - \mathbf{e}_j) \\
&= [(L^{\mathrm{a}})^+]_{ii} + [(L^{\mathrm{a}})^+]_{jj} - [(L^{\mathrm{a}})^+]_{ij} - [(L^{\mathrm{a}})^+]_{ji}.
\end{aligned}
$$

- Collect matrix of effective resistances: (= commute times)

$$\alpha\mathbf{C} = \mathrm{diag}(L^{\mathrm{a}})^+ \cdot \mathbf{1}^{\mathrm{T}} + \mathbf{1} \cdot \mathrm{diag}(L^{\mathrm{a}})^+ - (L^{\mathrm{a}})^+ - [(L^{\mathrm{a}})^+]^{\mathrm{T}}.$$

- The entries $\mathbf{C}_{ij}$ are squares of a Euclidean metric. [Schoenberg, 1935; Schoenberg, 1938; Berg et al., 1984],

Vector showing 2 classes

- Define $\mathbf{v} = \{\alpha, -\beta\}^n$ where $v_i = \alpha > 0$ is node i is in class A, and $v_i = -\beta < 0$ if node i is in class B.
- Then the non-zero entries of the vector $\mathbf{N}\mathbf{v}$ are in the positions corresponding to the edges with one end in class A and the other end in class B.
- Hence $\mathbf{v}^T\mathbf{N}^T\mathbf{N}\mathbf{v} = \mathbf{v}^T L\mathbf{v} = \mathrm{cut}(\mathrm{A},\mathrm{B})(\alpha+\beta)^2$.
- Also $\mathbf{v}^T\mathbf{v} = n_{\mathrm{A}}\alpha^2 + n_{\mathrm{B}}\beta^2$.
- Also $\mathbf{v}^T D\mathbf{v} = d_{\mathrm{A}}\alpha^2 + d_{\mathrm{B}}\beta^2$
- Here $n_{\mathrm{A}} = \#$ vertices in class A, $d_{\mathrm{A}} =$ sum of all degrees of nodes in class A. Ditto for class B. And $n = n_{\mathrm{A}} + n_{\mathrm{B}} =$ total number of vertices, and $d = d_{\mathrm{A}} + d_{\mathrm{B}} =$ 2 times total number of edges.

Cut relative to |nodes|

- Let $\alpha^2 = n_{\mathrm{B}}/n_{\mathrm{A}}$, $\beta^2 = n_{\mathrm{A}}/n_{\mathrm{B}}$.
- Then $\mathbf{v}^T L\mathbf{v} == \mathrm{cut}(\mathrm{A},\mathrm{B})\left(\frac{n_{\mathrm{A}}+n_{\mathrm{B}}}{\sqrt{n_{\mathrm{A}}n_{\mathrm{B}}}}\right)^2 = \mathrm{cut}(\mathrm{A},\mathrm{B})\frac{n^2}{n_{\mathrm{A}}n_{\mathrm{B}}}$,
- and $\mathbf{v}^T\mathbf{v} = n_{\mathrm{A}}(n_{\mathrm{B}}/n_{\mathrm{A}}) + n_{\mathrm{B}}(n_{\mathrm{A}}/n_{\mathrm{B}}) = n$.
- Hence
$$\frac{\mathbf{v}^T L\mathbf{v}}{\mathbf{v}^T\mathbf{v}} = \frac{\mathrm{cut}(\mathrm{A},\mathrm{B})}{n_{\mathrm{A}}n_{\mathrm{B}}}n$$
- Also $\mathbf{v}^T\mathbf{1} = n_{\mathrm{A}}\alpha - n_{\mathrm{B}}\beta = \sqrt{n_{\mathrm{A}}n_{\mathrm{B}}} - \sqrt{n_{\mathrm{B}}n_{\mathrm{A}}} = 0$.
- Hence
$$\frac{\mathbf{v}^T L\mathbf{v}}{\mathbf{v}^T\mathbf{v}} \geq \min_{\mathbf{x}\perp\mathbf{1}} \frac{\mathbf{x}^T L\mathbf{x}}{\mathbf{x}^T\mathbf{x}}.$$

Cut relative to |edges|

- Now look at minimal cut relative to the number of edges in each half.
- Let $\alpha^2 = d_\mathrm{B}/d_\mathrm{A}$, $\beta^2 = d_\mathrm{A}/d_\mathrm{B}$.
- Then $\mathbf{v}^T L \mathbf{v} == \mathrm{cut(A,B)} \left(\frac{d_\mathrm{A} + d_\mathrm{B}}{\sqrt{d_\mathrm{A} d_\mathrm{B}}} \right)^2 = \mathrm{cut(A,B)} \frac{d^2}{d_\mathrm{A} d_\mathrm{B}}$,
- and $\mathbf{v}^T D \mathbf{v} = d_\mathrm{A}(d_\mathrm{B}/d_\mathrm{A}) + d_\mathrm{B}(d_\mathrm{A}/d_\mathrm{B}) = d$.
- Hence

$$\frac{\mathbf{v}^T L \mathbf{v}}{\mathbf{v}^T D \mathbf{v}} = \frac{\mathrm{cut(A,B)}}{d_\mathrm{A} d_\mathrm{B}} d$$

Generalized Eigenvalue Problem

- Let $\mathbf{w} = D^{1/2}\mathbf{v}$. Then $\mathbf{w}^T\sqrt{\mathbf{d}} = \mathbf{v}^T\mathbf{d} = \alpha d_{\mathrm{A}} - \beta d_{\mathrm{B}} = 0$.

- Also $\mathcal{L}\sqrt{\mathbf{d}} = D^{-1/2}LD^{-1/2}\sqrt{\mathbf{d}} = 0$.

- The Rayleigh Quotient is

$$\frac{\mathbf{v}^T L\mathbf{v}}{\mathbf{v}^T D\mathbf{v}} = \frac{\mathbf{w}^T\mathcal{L}\mathbf{w}}{\mathbf{w}^T\mathbf{w}} \geq \min_{\mathbf{x}\perp\sqrt{\mathbf{d}}} \frac{\mathbf{x}^T\mathcal{L}\mathbf{x}}{\mathbf{x}^T\mathbf{x}} = \lambda_2(\mathcal{L}).$$

Relation to Random Walk

- The smallest non-zero eigenvalue of $\mathcal{L}$ is related to best edge-relative cut.
- The eigenvalues of $\mathcal{L}$ are the same as the eigenvalues of $I - P$:

$$D^{-1/2}\mathcal{L}D^{1/2} = D^{-1/2}(I - D^{-1/2}AD^{-1/2})D^{1/2} = I - D^{-1}A = I - P.$$

- The smallest non-zero eigenvalue of $\mathcal{L}$ corresponds to second largest eigenvalue of P, i.e., the mixing rate.
- The largest eigenvalue of $\mathcal{L}$ corresponds to the smallest (most negative) eigenvalue of P. The latter is at least -1 (exactly -1 iff random walk is 2-cyclic, periodic). So the former is at most 2, and exactly equal to 2 iff graph is bipartite.

OUTLINE

Cheeger Bounds

- Denote the eigenvalue of $\mathcal{L}$ as $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n \leq 2$.
- The basic Cheeger bound is [Chung, 2005]

$$2h_G \geq \lambda_2(\mathcal{L}) \geq {}^1\!/_2 h_G^2$$

where

$h_G =$ minimum cut relative to the edge weights,

$\lambda_2(\mathcal{L}) =$ 2nd smallest eigenvalue of $\mathcal{L} = I - D^{-1/2} A D^{-1/2}$.

Isoperimetric Constant

Definitions: [Chung, 2005]

- Neighborhood of set X of nodes, $N(X)$, is the set of nodes not in X but with an edge to X.
- $g_G = \min_{X:vol(X)\leq vol(\overline{X})} \dfrac{vol(N(X))}{vol(X)}$

2 bounds: [Chung, 2005]

- $\lambda_2 \geq \dfrac{g_G^2}{2d(2+2g_G+g_G^2)}$.
- $g_G \geq \dfrac{1-(1-\lambda')^2}{(1-\lambda')^2+\frac{vol(X)}{vol(\overline{X})}} \geq (1-(1-\lambda')^2)(1-\dfrac{vol(X)}{vol(\overline{X})}$,

  where $\lambda' = \frac{2\lambda_2}{\lambda_2+\lambda_n}$ if $1-\lambda_2 < \lambda_n - 1$, and $\lambda' = \lambda_2$ o.w.

Conclusions

- Introduced "Random Walk" Laplacian for strongly connected directed graph
- Fundamental Tensor - fast way to encode many properties
- Laplacian Related to Average Commute Times
- Laplacian Related to Electric Resistance
- Laplacian Related to mixing times and Graph Cuts.

Lemma 1 – Inverse of Submatrix

Let $L = \begin{pmatrix} L_{11} & \mathbf{l}_{12} \\ \mathbf{l}_{21}^{\mathrm{T}} & l_{nn} \end{pmatrix}$ be an $n \times n$ irreducible matrix s.t. $\mathsf{nullity}(L) = 1$.
Let $M = L^{+}$ be the pseudo-inverse of L partitioned similarly and assume $(\mathbf{u}^{\mathrm{T}}, 1)L = 0$, $L(\mathbf{v}; 1) = 0$, where $\mathbf{u}, \mathbf{v}$ are $(n-1)$-vectors.

Then the inverse of the $(n-1) \times (n-1)$ matrix L_{11} exists and is given by

$$\begin{aligned}
L_{11}^{-1} &= X \stackrel{\text{def}}{=} (I_{n-1} + \mathbf{v}\mathbf{v}^{\mathrm{T}}) M_{11} (I_{n-1} + \mathbf{u}\mathbf{u}^{\mathrm{T}}) \\
&= (I_{n-1}\,,\, -\mathbf{v}) \begin{pmatrix} M_{11} & \mathbf{m}_{12} \\ \mathbf{m}_{21}^{\mathrm{T}} & m_{nn} \end{pmatrix} \begin{pmatrix} I_{n-1} \\ -\mathbf{u}^{\mathrm{T}} \end{pmatrix} \\
&= M_{11} - \mathbf{m}_{12}\mathbf{u}^{\mathrm{T}} - \mathbf{v}\mathbf{m}_{21}^{\mathrm{T}} + m_{nn}\mathbf{v}\mathbf{u}^{\mathrm{T}}.
\end{aligned}$$

If $\mathbf{u} = \mathbf{v} = \mathbf{1}$ then $[L_{11}^{-1}]_{ij} = m_{ij} + m_{nn} - m_{in} - m_{nj}$.

Proof

- Idea: Plug prospective inverse X in to verify $XL_{11} = I$:

$$
\begin{aligned}
XL_{11} &= (I_{n-1}\,,\, -\mathbf{v}) \begin{pmatrix} M_{11} & \mathbf{m}_{12} \\ \mathbf{m}_{21}^{\mathrm{T}} & m_{nn} \end{pmatrix} \begin{pmatrix} I_{n-1} \\ -\mathbf{u}^{\mathrm{T}} \end{pmatrix} L_{11} \\
&= (I_{n-1}\,,\, -\mathbf{v}) \begin{pmatrix} M_{11} & \mathbf{m}_{12} \\ \mathbf{m}_{21}^{\mathrm{T}} & m_{nn} \end{pmatrix} \begin{pmatrix} L_{11} \\ \mathbf{l}_{21}^{\mathrm{T}} \end{pmatrix} && \boxed{\mathrm{A}} \\
&= (I_{n-1}\,,\, -\mathbf{v}) ML \begin{pmatrix} I_{n-1} \\ \mathbf{0}^{\mathrm{T}} \end{pmatrix} \\
&= (I_{n-1}\,,\, -\mathbf{v}) \begin{pmatrix} I_{n-1} \\ \mathbf{0}^{\mathrm{T}} \end{pmatrix} = I_{n-1} && \boxed{\mathrm{B}}
\end{aligned}
$$

$\boxed{\mathrm{A}}$ From $(\mathbf{u}^{\mathrm{T}}, 1)L = (\mathbf{u}^{\mathrm{T}} L_{11} + \mathbf{l}_{21}^{\mathrm{T}}\;,\; \mathbf{u}^{\mathrm{T}} \mathbf{l}_{12} + l_{nn}) = 0.$

$\boxed{\mathrm{B}}$ From $ML = I_n - \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix} (\mathbf{v}^T, 1)/(\mathbf{v}^{\mathrm{T}}\mathbf{v} + 1)$ (ortho projector).

Lemma 2 – Conditionally Definite

If M is a symmetric positive semi-definite Gram matrix of inner products,
Then $\mathbf{C} = \mathbf{d}_M\mathbf{1}^{\mathrm{T}} + \mathbf{1}\mathbf{d}_M^{\mathrm{T}} - 2M$ s.t. $c_{ij} = m_{ii} + m_{jj} - 2m_{ij}$ is the conditionally definite matrix of squared distances. [here $\mathbf{d}_M = (m_{11}; \ldots; m_{nn})$]
Note "Conditionally definite" means $\mathbf{x}^{\mathrm{T}}\mathbf{C}\mathbf{x} \leq 0$ for all $\mathbf{x} \perp \mathbf{1}$, and for simplicity $c_{ii} = 0, \forall i$. A typical example is a matrix of pairwise squared ℓ_2 distances.

If $\mathbf{C}$ is a conditionally definite matrix,
Then one can find a matching semi-definite Gram matrix M.
Note: A prospective uncentered M is given by $2\widehat{M} = \mathbf{c}_k\mathbf{1}^{\mathrm{T}} + \mathbf{1}\mathbf{c}_k^{\mathrm{T}} - \mathbf{C}$, where $\mathbf{c}_k$ is some arbitrarily selected column out of $\mathbf{C}$.
The result can be centered around the origin, yielding:

$$M = \left(I - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right)\widehat{M}\left(I - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right) = -{}^1\!/_2\left(I - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right)\mathbf{C}\left(I - \frac{\mathbf{1}\mathbf{1}^{\mathrm{T}}}{n}\right).$$

[Schoenberg, 1935; Schoenberg, 1938; Berg et al., 1984; Gower & Legendre, 1986]

Proof: AWLOG $\mathbf{x}_1 = 0$. Then $c_{1k} = c_{k1} = \|x_k\|_2^2$.
So $c_{ij} = m_{ii} + m_{jj} - 2m_{ij} = c_{i1} + c_{1j} - 2m_{ij}$.

References

Bamieh, B., Jovanovic, M., Mitra, P., & Patterson, S. (2008). Effect of topological dimension on rigidity of vehicle formations: Fundamental limitations of local feedback. *Proc. CDC* (pp. 369–374). Cancun, Mexico.

Berg, C., Christiansen, J., & Ressel, P. (1984). *Harmonic analysis on semigroups*. Springer Verlag.

Boley, D., Buendia, A., & Golnari, G. (2018). Random walk laplacian and network centrality measures. arXiv 1808.02912.

Boley, D., Ranjan, G., & Zhang, Z.-L. (2010). Commute times for a directed graph using an asymmetric laplacian. U of Mn CS&E Dept report TR 10-005, Univ. of Minn., Dept. of Comp uter Science and Eng. (a revised version is to appear in LAA 2011).

Boley, D., Ranjan, G., & Zhang, Z.-L. (2011). Commute times for a directed graph using an asymmetric Laplacian. *Linear Algebra and Appl.*, *435*, 224–242.

Brualdi, R. A., & Ryser, H. J. (1991). *Combinatorial matrix theory*. Cambridge Univ. Press.

Chandra, A., Raghavan, P., Ruzzo, W., Smolensky, R., & Tiwari, P. (1989). The electrical resistance of a graph captures its commute and cover times. *Proc. of Annual ACM Symposium on Theory of Computing* (pp. 574–586).

Chebotarev, P., & Shamis, E. (2006). Matrix-forest theorems.

Chung, F. (2005). Laplacians and the Cheeger inequality for directed graphs. *Annals of Combinatorics*, *9*, 1–19.

Doyle, P., & Snell, J. (1984). *Random walks and electric networks.* The Math. Assoc. of Am. `front.math.ucdavis.edu/math.PR/0001057`.

Fouss, F., Pirotte, A., Renders, J., & Saerens, M. (2007). Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Trans. on Knowledge and Data Engineering*, *19*, 355–369.

Golnari, G., Zhang, Z.-L., & Boley, D. (2019). Markov fundamental tensor and its applications to network analysis. *Linear Algebra and Appl.*, *564*, 126–158.

Gower, J., & Legendre, P. (1986). Metric and euclidean properties of dissimilarities coefficients. *J. Classification*, *3*, 5–48.

Grinstead, C. M., & Snell, J. L. (2006). *Introduction to probability.* American Mathematical Society. 2nd edition, `www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/book.html`.

Klein, D., & Randic, M. (1993). Resistance distance. *J. Math. Chemistry*, *12*, 81–95.

Olfati-Saber, R., Murray, R. M., A, & B (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. Auto. Contr.*, *49*, 1520–1533.

Schoenberg, I. J. (1935). Remarks to Maurice Fréchet's article "Sur la définition axiomatique d'une classe d'espace distanciés vectoriellement applicable sur l'espace de Hilbert". *Annals of Mathematics*, *36*, 724–732.

Schoenberg, I. J. (1938). Metric spaces and positive definite functions. *Trans. of the Amer. Math Soc.*, *44*, 522–536.

Shi, J., & Malik, J. (2000). Normalised cuts and image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, *22*, 888–905.

Spielman, D. A., & Teng, S.-H. (1996). Spectral partitioning works:planar graphs and finite element meshes. *37th Annual Symposium on Foundations of Computer Science*. IEEE Computer Soc. Press.

von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, *17*, 395–416. Max Planck Institute for Biological Cybernetics. Technical Report No. TR-149.

Young, G. F., Scandovi, L., & Leonard, N. (2010). Robustness of noisy consensus dynamics with directed communication. *Proc. ACC* (pp. 6312–6317).

Zhou, D., Huang, J., & Schölkopf, B. (2005). Learning from labeled and unlabeled data on a directed graph. *Proc. 22nd Int'l Conf. Machine Learning* (pp. 1041–1048).