Algorithm Foundations of Data Science

Lecture 4: Centrality

MING GAO

DaSE@ECNU
(for course related communications)
mgao@dase.ecnu.edu.cn

Apr. 4, 2018

Outline

Centrality

Motivations

The idea of centrality is to identify important nodes in networks.

- Influential or popular users (Crowdsourcing platforms, information propagation networks, celebrities in social networks etc)
- Authority/Expert users (Stack Overflow, ResearchGate, etc.)
- Brokers (Network services, Telecom interaction, etc.)

Depending on the network, centrality may carry different meanings.

Centrality

Motivations

The idea of centrality is to identify important nodes in networks.

- Influential or popular users (Crowdsourcing platforms, information propagation networks, celebrities in social networks etc)
- Authority/Expert users (Stack Overflow, ResearchGate, etc.)
- Brokers (Network services, Telecom interaction, etc.)

Depending on the network, centrality may carry different meanings.

Centrality measurements

- Degree centrality
- Eigenvector centrality
- Closeness centrality

Centrality

Motivations

The idea of centrality is to identify important nodes in networks.

- Influential or popular users (Crowdsourcing platforms, information propagation networks, celebrities in social networks etc)
- Authority/Expert users (Stack Overflow, ResearchGate, etc.)
- Brokers (Network services, Telecom interaction, etc.)

Depending on the network, centrality may carry different meanings.

Centrality measurements

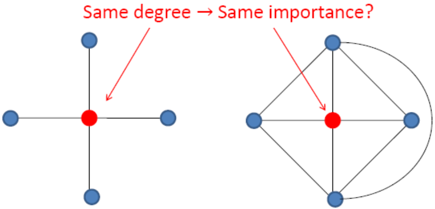
- Degree centrality
- Eigenvector centrality
- Closeness centrality
- Betweenness centrality
- Clustering coefficient
- PageRank and HITS

Degree centrality

Definition

For a undirected network, let $A_{ij} = 1$ if nodes i and j are connected; 0 otherwise. Degree centrality is defined as

$$C_D(i) = \sum_{j=1}^{n} A_{ij}.$$



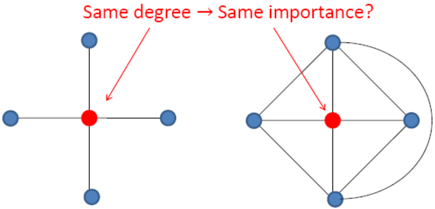
Degree centrality

Definition

For a undirected network, let $A_{ij} = 1$ if nodes i and j are connected; 0 otherwise. Degree centrality is defined as

$$C_D(i) = \sum_{j=1}^{n} A_{ij}.$$

- Normalized degree centrality is $\frac{C_D(i)}{n-1}$ since max degree is n - 1.
- An assumption is that all neighbors are the same.



Eigenvector centrality

Definition

Eigenvector centrality of node i is defined as

$$C_E(i) = x_i = \frac{1}{\lambda} \sum_{j=1}^{n} A_{ij} x_j.$$

Eigenvector centrality

Definition

Eigenvector centrality of node i is defined as

$$C_E(i) = x_i = \frac{1}{\lambda}\sum_{j=1}^{n} A_{ij}x_j.$$

- In matrix form, let $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ be the vector of eigenvector centralities of nodes, we therefore have $\lambda\mathbf{x} = A\mathbf{x}$.

Eigenvector centrality

Definition

Eigenvector centrality of node i is defined as

$$C_E(i) = x_i = \frac{1}{\lambda} \sum_{j=1}^{n} A_{ij} x_j.$$

- In matrix form, let $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ be the vector of eigenvector centralities of nodes, we therefore have $\lambda \mathbf{x} = A\mathbf{x}$.
- Assuming that x_i is non-negative for all i, it can be shown that:
  - λ is the largest eigenvalue of A;

Eigenvector centrality

Definition

Eigenvector centrality of node i is defined as

$$C_E(i) = x_i = \frac{1}{\lambda} \sum_{j=1}^{n} A_{ij} x_j.$$

- In matrix form, let $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ be the vector of eigenvector centralities of nodes, we therefore have $\lambda \mathbf{x} = A\mathbf{x}$.
- Assuming that x_i is non-negative for all i, it can be shown that:
  - λ is the largest eigenvalue of A;
  - $\mathbf{x}$ is the corresponding eigenvector.

Eigenvector centrality

Definition

Eigenvector centrality of node i is defined as

$$C_E(i) = x_i = \frac{1}{\lambda} \sum_{j=1}^{n} A_{ij} x_j.$$

- In matrix form, let $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ be the vector of eigenvector centralities of nodes, we therefore have $\lambda \mathbf{x} = A\mathbf{x}$.
- Assuming that x_i is non-negative for all i, it can be shown that:
  - λ is the largest eigenvalue of A;
  - $\mathbf{x}$ is the corresponding eigenvector.
- PageRank, where weights are transition probabilities in a directed weighted network, is a variant of eigenvector centrality.

Closeness centrality

Definition

Closeness measures how long it will take to spread information from i to all other nodes sequentially.

Closeness centrality

Definition

Closeness measures how long it will take to spread information from i to all other nodes sequentially.

- Geodesic path $d(i, j)$ is the shortest path between nodes i and j (may not be unique).

Closeness centrality

Definition

Closeness measures how long it will take to spread information from i to all other nodes sequentially.

- Geodesic path $d(i,j)$ is the shortest path between nodes i and j (may not be unique).
- Farness of node i: $d(i) = \sum_{j \neq i} d(i,j)$ (sum of geodesic distance from i to every other node).

Closeness centrality

Definition

Closeness measures how long it will take to spread information from i to all other nodes sequentially.

- Geodesic path $d(i,j)$ is the shortest path between nodes i and j (may not be unique).
- Farness of node i: $d(i) = \sum_{j \neq i} d(i,j)$ (sum of geodesic distance from i to every other node).
- **Closeness centrality** of node i is defined as

$$C_C(i) = \frac{1}{d(i)}.$$

Betweenness centrality

Definition

Betweenness centrality of node i is defined as

$$C_B(i) = \sum_{j=1}^{n} \sum_{k:k>j}^{n} \frac{g_{jk}(i)}{g_{jk}},$$

where g_{jk} denotes $\#$ geodesic paths between nodes j and k, and $g_{jk}(i)$ denotes $\#$ geodesic paths passing through node i.

Betweenness centrality

Definition

Betweenness centrality of node i is defined as

$$C_B(i) = \sum_{j=1}^{n} \sum_{k:k>j}^{n} \frac{g_{jk}(i)}{g_{jk}},$$

where g_{jk} denotes # geodesic paths between nodes j and k, and $g_{jk}(i)$ denotes # geodesic paths passing through node i.

- For a communication network, an important node is strategically located on the paths linking many pairs of others.

Betweenness centrality

Definition

Betweenness centrality of node i is defined as

$$C_B(i) = \sum_{j=1}^{n} \sum_{k:k>j}^{n} \frac{g_{jk}(i)}{g_{jk}},$$

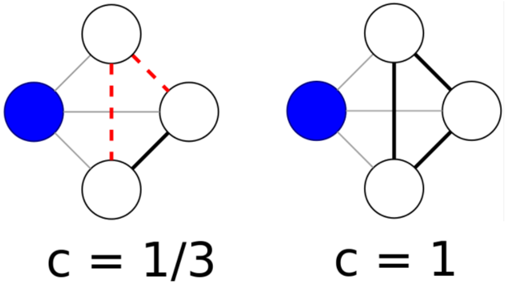
where g_{jk} denotes # geodesic paths between nodes j and k, and $g_{jk}(i)$ denotes # geodesic paths passing through node i.

- For a communication network, an important node is strategically located on the paths linking many pairs of others.
- Nodes with high betweenness will exert substantial influence by virtue not of being in the middle of the network but of lying "between" other vertices.

Clustering coefficient

Definition

Given a graph $G = (V, E)$, $N(v_i) = \{v_j | (v_i, v_j) \in E\}$ and $d(v_i) = |N(v_i)|$.

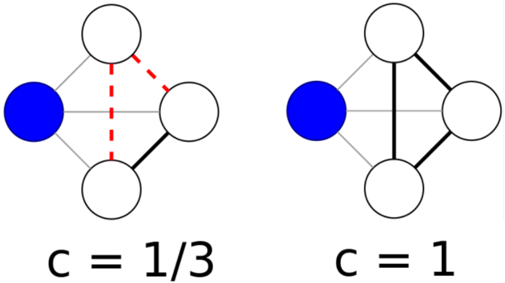


Clustering coefficient

Definition

Given a graph $G = (V, E)$, $N(v_i) = \{v_j | (v_i, v_j) \in E\}$ and $d(v_i) = |N(v_i)|$.

- Local clustering coefficient of v_i: $C_i = \frac{2|\{e_{jk} | v_j, v_k \in N(v_i), (v_j, v_k) \in E\}|}{d(v_i)(d(v_i) - 1)}$.

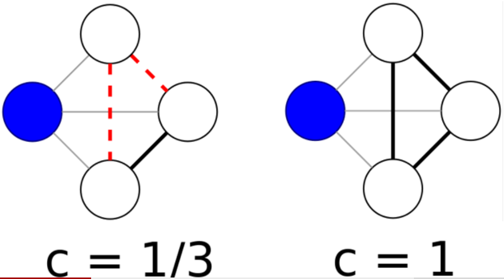


Clustering coefficient

Definition

Given a graph $G = (V, E)$, $N(v_i) = \{v_j | (v_i, v_j) \in E\}$ and $d(v_i) = |N(v_i)|$.

- Local clustering coefficient of v_i: $C_i = \frac{2|\{e_{jk} | v_j, v_k \in N(v_i), (v_j, v_k) \in E\}|}{d(v_i)(d(v_i)-1)}$.
- Global clustering coefficient: $\overline{C} = \frac{1}{n} \sum_{i=1}^{n} C_i$.



Weighted networks

Weighted networks

- Weighted edges in networks
  - Function of duration, e.g., duration of chat.
  - Emotional intensity, e.g., number of emails between nodes.
  - Intimacy, e.g., mutual confiding, rating items.
- The number of weighted networks are growing due to Web 2.0.
- Information is lost when modeling weighted networks as binary networks.

Weighted networks

Weighted networks

- Weighted edges in networks
  - Function of duration, e.g., duration of chat.
  - Emotional intensity, e.g., number of emails between nodes.
  - Intimacy, e.g., mutual confiding, rating items.
- The number of weighted networks are growing due to Web 2.0.
- Information is lost when modeling weighted networks as binary networks.

Extensions of centrality for weighted networks

Focus on tie weights but not number of ties.

Weighted networks

Weighted networks

- Weighted edges in networks
  - Function of duration, e.g., duration of chat.
  - Emotional intensity, e.g., number of emails between nodes.
  - Intimacy, e.g., mutual confiding, rating items.
- The number of weighted networks are growing due to Web 2.0.
- Information is lost when modeling weighted networks as binary networks.

Extensions of centrality for weighted networks

Focus on tie weights but not number of ties.

- Degree centrality: sum of weights of ties connected to the node.
- Closeness centrality: shortest path = least costly path.

Degree centrality for weighted networks

Definition

Let w_{ij} be weight of edge between nodes i and j. Degree centrality of node i is defined as

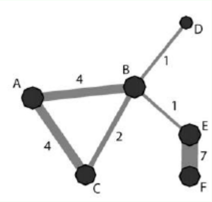
$$C_{WD}(i) = \sum_{j=1}^{n} w_{ij}.$$

Degree centrality for weighted networks

Definition

Let w_{ij} be weight of edge between nodes i and j. Degree centrality of node i is defined as

$$C_{WD}(i) = \sum_{j=1}^{n} w_{ij}.$$

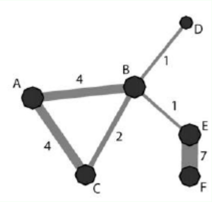


Degree centrality for weighted networks

Definition

Let w_{ij} be weight of edge between nodes i and j. Degree centrality of node i is defined as

$$C_{WD}(i) = \sum_{j=1}^{n} w_{ij}.$$



Example

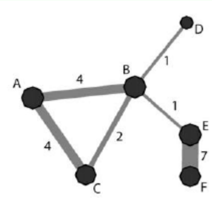
- Nodes A and B have the same weighted degree;

Degree centrality for weighted networks

Definition

Let w_{ij} be weight of edge between nodes i and j. Degree centrality of node i is defined as

$$C_{WD}(i) = \sum_{j=1}^{n} w_{ij}.$$



Example

- Nodes A and B have the same weighted degree;
- But B is connected to twice many nodes as A.

Degree centrality for weighted networks cont.

Definition

Let α be the tuning parameters to determine the relative importance of number of ties:

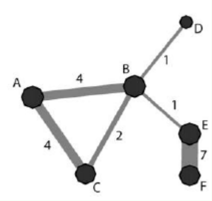
$$C_{WD,\alpha}(i) = C_D(i) \times \left(\frac{C_{WD}(i)}{C_D(i)}\right)^{\alpha} = C_D(i)^{(1-\alpha)} \times C_{WD}(i)^{\alpha}.$$

Degree centrality for weighted networks cont.

Definition

Let α be the tuning parameters to determine the relative importance of number of ties:

$$C_{WD,\alpha}(i) = C_D(i) \times \left(\frac{C_{WD}(i)}{C_D(i)}\right)^{\alpha} = C_D(i)^{(1-\alpha)} \times C_{WD}(i)^{\alpha}.$$

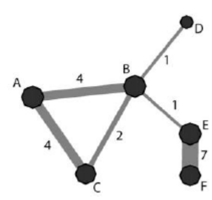


Degree centrality for weighted networks cont.

Definition

Let α be the tuning parameters to determine the relative importance of number of ties:

$$C_{WD,\alpha}(i) = C_D(i) \times \left(\frac{C_{WD}(i)}{C_D(i)}\right)^{\alpha} = C_D(i)^{(1-\alpha)} \times C_{WD}(i)^{\alpha}.$$



Analysis

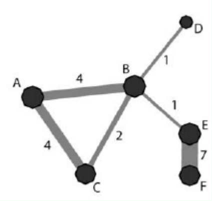
- $\alpha = 1 \rightarrow C_{WD,\alpha}(i) = C_{WD}(i)$.

Degree centrality for weighted networks cont.

Definition

Let α be the tuning parameters to determine the relative importance of number of ties:

$$C_{WD,\alpha}(i) = C_D(i) \times \left(\frac{C_{WD}(i)}{C_D(i)}\right)^{\alpha} = C_D(i)^{(1-\alpha)} \times C_{WD}(i)^{\alpha}.$$



Analysis

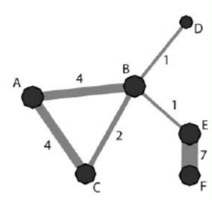
- $\alpha = 1 \rightarrow C_{WD,\alpha}(i) = C_{WD}(i)$.
- $\alpha = 0 \rightarrow C_{WD,\alpha}(i) = C_D(i)$.

Degree centrality for weighted networks cont.

Definition

Let α be the tuning parameters to determine the relative importance of number of ties:

$$C_{WD,\alpha}(i) = C_D(i) \times \left(\frac{C_{WD}(i)}{C_D(i)}\right)^{\alpha} = C_D(i)^{(1-\alpha)} \times C_{WD}(i)^{\alpha}.$$



Analysis

- $\alpha = 1 \rightarrow C_{WD,\alpha}(i) = C_{WD}(i)$.
- $\alpha = 0 \rightarrow C_{WD,\alpha}(i) = C_D(i)$.
- What happens when $\alpha > 1$?

Degree centrality for directed networks

Definition

$Outdeg(i)$ and $Indeg(i)$ are defined as:

$$C_{WD,\alpha,out}(i) = C_{D,out}(i) \times \left(\frac{C_{WD,out}(i)}{C_{D,out}(i)}\right)^{\alpha},$$

$$C_{WD,\alpha,in}(i) = C_{D,in}(i) \times \left(\frac{C_{WD,in}(i)}{C_{D,in}(i)}\right)^{\alpha},$$

PageRank

Definition

Given a directed graph, find its most interesting or central node.

PageRank

Definition

Given a directed graph, find its most interesting or central node.

- Inlinks are "good"; inlinks from a "good" site are better than inlinks from a "bad" site; but inlinks from sites with many outlinks are not as "good".
- The directed graph is modeled as a random walk. The most "popular" node has steady state probability.
- Let A be the transition matrix (induced by adjacency matrix), the algorithm finds $\mathbf{x}$ s.t. $A\mathbf{x} = \mathbf{x}$.
- Thus, $\mathbf{x}$ is the eigenvector that corresponds to the highest eigenvalues.
- Why does such a $\mathbf{x}$ exist? $\mathbf{x}$ exist if A is irreducible aperiodic; $\lambda A + (1 - \lambda)[\frac{1}{n}]$ otherwise (e.g., $\lambda = 0.15$).

HITS: Hyperlink-Induced Topic Search

Given the web and a query, find the most “authoritative” Web pages for this query. Let transition matrix be A, authority vector be $\mathbf{v}$, and hub vector be $\mathbf{u}$.

HITS: Hyperlink-Induced Topic Search

Given the web and a query, find the most "authoritative" Web pages for this query. Let transition matrix be A, authority vector be $\mathbf{v}$, and hub vector be $\mathbf{u}$.

1. Find all pages containing the query terms.

HITS: Hyperlink-Induced Topic Search

Given the web and a query, find the most "authoritative" Web pages for this query. Let transition matrix be A, authority vector be $\mathbf{v}$, and hub vector be $\mathbf{u}$.

1. Find all pages containing the query terms.
2. Expand by one move forward and backward.
   - Authority update: update each node's Authority score to be equal to the sum of the Hub scores of each node that points to it, i.e., $\mathbf{v} = A^T\mathbf{u}$.

HITS: Hyperlink-Induced Topic Search

Given the web and a query, find the most "authoritative" Web pages for this query. Let transition matrix be A, authority vector be $\mathbf{v}$, and hub vector be $\mathbf{u}$.

1. Find all pages containing the query terms.
2. Expand by one move forward and backward.
   - Authority update: update each node's Authority score to be equal to the sum of the Hub scores of each node that points to it, i.e., $\mathbf{v} = A^T\mathbf{u}$.
   - Hub update: update each node's Hub score to be equal to the sum of the Authority scores of each node that it points to, i.e., $\mathbf{u} = A\mathbf{v}$.

HITS: Hyperlink-Induced Topic Search

Given the web and a query, find the most "authoritative" Web pages for this query. Let transition matrix be A, authority vector be $\mathbf{v}$, and hub vector be $\mathbf{u}$.

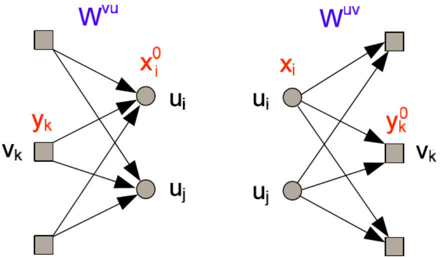
1. Find all pages containing the query terms.
2. Expand by one move forward and backward.
    - Authority update: update each node's Authority score to be equal to the sum of the Hub scores of each node that points to it, i.e., $\mathbf{v} = A^T\mathbf{u}$.
    - Hub update: update each node's Hub score to be equal to the sum of the Authority scores of each node that it points to, i.e., $\mathbf{u} = A\mathbf{v}$.
    - Update rule: $\begin{cases} \mathbf{v}^{(t)} = (A^TA)\mathbf{v}^{(t-1)} \\ \mathbf{u}^{(t)} = (AA^T)\mathbf{u}^{(t-1)} \end{cases}$ if given initial vector with $\sum_i \mathbf{v}_i^{(0)} = \sum_i \mathbf{u}_i^{(0)} = 1$.

HITS: Hyperlink-Induced Topic Search

Given the web and a query, find the most "authoritative" Web pages for this query. Let transition matrix be A, authority vector be $\mathbf{v}$, and hub vector be $\mathbf{u}$.

1. Find all pages containing the query terms.
2. Expand by one move forward and backward.
   - Authority update: update each node's Authority score to be equal to the sum of the Hub scores of each node that points to it, i.e., $\mathbf{v} = A^T\mathbf{u}$.
   - Hub update: update each node's Hub score to be equal to the sum of the Authority scores of each node that it points to, i.e., $\mathbf{u} = A\mathbf{v}$.
   - Update rule: $\begin{cases} \mathbf{v}^{(t)} = (A^TA)\mathbf{v}^{(t-1)} \\ \mathbf{u}^{(t)} = (AA^T)\mathbf{u}^{(t-1)} \end{cases}$ if given initial vector with $\sum_i \mathbf{v}_i^{(0)} = \sum_i \mathbf{u}_i^{(0)} = 1$.
   - Thus, $\mathbf{v}$ ($\mathbf{u}$) is the eigenvector that corresponds to the highest eigenvalues of A^TA (AA^T).

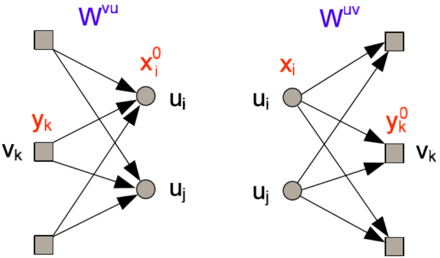
Co-HITS for bipartite graph



Iterative framework

Given a query q, The initial relevance scores x_i^0 and y_j^0 are respectively defined by $x_i^0 = f(q, u_i)$, and $y_j^0 = f(q, v_j)$ for u_i and v_j.

Co-HITS for bipartite graph



Iterative framework

Given a query q, The initial relevance scores x_i^0 and y_j^0 are respectively defined by $x_i^0 = f(q, u_i)$, and $y_j^0 = f(q, v_j)$ for u_i and v_j.

- Update rule: $\begin{cases} x_i^{(t)} = (1-\lambda_u)x_i^{(0)} + \lambda_u \sum_{k\in V} w_{ki}^{vu} y_k^{(t-1)} \\ y_k^{(t)} = (1-\lambda_v)y_k^{(0)} + \lambda_v \sum_{i\in U} w_{ik}^{uv} x_i^{(t-1)} \end{cases}$

Co-HITS for bipartite graph cont.

Regularization framework

- $R_1 = \frac{1}{2}\sum_{i,j\in U} w_{i,j}^{uu}\left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{x_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in U}\left(x_i - x_i^{(0)}\right)^2$, where $d_{ii} = \sum_j w_{i,j}$.

Co-HITS for bipartite graph cont.

Regularization framework

- $R_1 = \frac{1}{2}\sum_{i,j\in U} w_{i,j}^{uu}\left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{x_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in U}\left(x_i - x_i^{(0)}\right)^2$, where $d_{ii} = \sum_j w_{i,j}$.
- $R_2 = \frac{1}{2}\sum_{i,j\in V} w_{i,j}^{vv}\left(\frac{y_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in V}\left(y_i - y_i^{(0)}\right)^2$.

Co-HITS for bipartite graph cont.

Regularization framework

- $R_1 = \frac{1}{2}\sum_{i,j\in U} w_{i,j}^{uu}\left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{x_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in U}\left(x_i - x_i^{(0)}\right)^2$, where $d_{ii} = \sum_j w_{i,j}$.
- $R_2 = \frac{1}{2}\sum_{i,j\in V} w_{i,j}^{vv}\left(\frac{y_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in V}\left(y_i - y_i^{(0)}\right)^2$.
- $R_3 = \frac{1}{2}\sum_{i\in U,j\in V} w_{i,j}^{uv}\left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}}\right)^2 + \frac{1}{2}\sum_{j\in V,i\in U} w_{j,i}^{vu}\left(\frac{y_j}{\sqrt{d_{jj}}} - \frac{x_i}{\sqrt{d_{ii}}}\right)^2$.

Co-HITS for bipartite graph cont.

Regularization framework

- $R_1 = \frac{1}{2} \sum_{i,j \in U} w_{i,j}^{uu} \left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{x_j}{\sqrt{d_{jj}}} \right)^2 + \mu \sum_{i \in U} \left(x_i - x_i^{(0)} \right)^2$, where $d_{ii} = \sum_j w_{i,j}$.
- $R_2 = \frac{1}{2} \sum_{i,j \in V} w_{i,j}^{vv} \left(\frac{y_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}} \right)^2 + \mu \sum_{i \in V} \left(y_i - y_i^{(0)} \right)^2$.
- $R_3 = \frac{1}{2} \sum_{i \in U, j \in V} w_{i,j}^{uv} \left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}} \right)^2 + \frac{1}{2} \sum_{j \in V, i \in U} w_{j,i}^{vu} \left(\frac{y_j}{\sqrt{d_{jj}}} - \frac{x_i}{\sqrt{d_{ii}}} \right)^2$.
- Cost function: $R = \lambda_r (R_1 + \alpha R_2) + (1 - \lambda_r) R_3$.

Co-HITS for bipartite graph cont.

Regularization framework

- $R_1 = \frac{1}{2}\sum_{i,j\in U} w_{i,j}^{uu}\left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{x_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in U}\left(x_i - x_i^{(0)}\right)^2$, where $d_{ii} = \sum_j w_{i,j}$.
- $R_2 = \frac{1}{2}\sum_{i,j\in V} w_{i,j}^{vv}\left(\frac{y_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}}\right)^2 + \mu\sum_{i\in V}\left(y_i - y_i^{(0)}\right)^2$.
- $R_3 = \frac{1}{2}\sum_{i\in U,j\in V} w_{i,j}^{uv}\left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}}\right)^2 + \frac{1}{2}\sum_{j\in V,i\in U} w_{j,i}^{vu}\left(\frac{y_j}{\sqrt{d_{jj}}} - \frac{x_i}{\sqrt{d_{ii}}}\right)^2$.
- Cost function: $R = \lambda_r(R_1 + \alpha R_2) + (1-\lambda_r)R_3$.
- Rewrite:

$$\min_F \frac{1}{2}\sum_{i,j=1}^{m+n} w_{i,j}\left(\frac{f_i}{\sqrt{d_{ii}}} - \frac{f_j}{\sqrt{d_{jj}}}\right)^2 + \frac{\mu}{2}\sum_{i=1}^{m+n}(f_i - f_i^{(0)})^2$$

$$s.t. W = \begin{pmatrix} W^{uu} & \beta W^{uv} \\ \beta W^{vu} & W^{vv} \end{pmatrix}, F = \begin{pmatrix} X \\ Y \end{pmatrix}, \text{ and } \beta = \frac{1-\lambda_r}{\lambda_r}$$

Recall graph Laplacian

Definition

Given graph G, let d_v be the degree of node v, adjacency matrix be A, and $D = diag(d_1, d_2, \cdots, d_n)$.

Recall graph Laplacian

Definition

Given graph G, let d_v be the degree of node v, adjacency matrix be A, and $D = diag(d_1, d_2, \cdots, d_n)$.

- (Combinatorial) Laplacian of G: $L = D - A$, i.e.,

$$L(u,v) = \begin{cases} d_v, & \text{if } u = v; \\ -1, & \text{if } u \text{ and } v \text{ are adjacent ;} \\ 0, & \text{otherwise.} \end{cases}$$

Recall graph Laplacian

Definition

Given graph G, let d_v be the degree of node v, adjacency matrix be A, and $D = diag(d_1, d_2, \cdots, d_n)$.

- (Combinatorial) Laplacian of G: $L = D - A$, i.e.,
$$L(u,v) = \begin{cases} d_v, & \text{if } u = v; \\ -1, & \text{if } u \text{ and } v \text{ are adjacent }; \\ 0, & \text{otherwise.} \end{cases}$$
- Normailized Laplacian of G: $\mathcal{L} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2}$,
i.e., $\mathcal{L}(u,v) = \begin{cases} 1, & \text{if } u = v; \\ -\frac{1}{\sqrt{d_u d_v}}, & \text{if } u \text{ and } v \text{ are adjacent }; \\ 0, & \text{otherwise.} \end{cases}$

Recall graph Laplacian

Definition

Given graph G, let d_v be the degree of node v, adjacency matrix be A, and $D = diag(d_1, d_2, \cdots, d_n)$.

- (Combinatorial) Laplacian of G: $L = D - A$, i.e.,
$$L(u,v) = \begin{cases} d_v, & \text{if } u = v; \\ -1, & \text{if } u \text{ and } v \text{ are adjacent }; \\ 0, & \text{otherwise.} \end{cases}$$
- Normailized Laplacian of G: $\mathcal{L} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}AD^{-1/2}$,
i.e., $\mathcal{L}(u,v) = \begin{cases} 1, & \text{if } u = v; \\ -\frac{1}{\sqrt{d_u d_v}}, & \text{if } u \text{ and } v \text{ are adjacent }; \\ 0, & \text{otherwise.} \end{cases}$
- For weighted graph G, Laplacian and normalized Laplacian can be defined in a same manner.

Recall graph Laplacian

Definition

Given graph G, let d_v be the degree of node v, adjacency matrix be A, and $D = diag(d_1, d_2, \cdots, d_n)$.

- (Combinatorial) Laplacian of G: $L = D - A$, i.e.,
$$L(u,v) = \begin{cases} d_v, & \text{if } u = v; \\ -1, & \text{if } u \text{ and } v \text{ are adjacent }; \\ 0, & \text{otherwise.} \end{cases}$$
- Normailized Laplacian of G: $\mathcal{L} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}AD^{-1/2}$, i.e., $$\mathcal{L}(u,v) = \begin{cases} 1, & \text{if } u = v; \\ -\frac{1}{\sqrt{d_u d_v}}, & \text{if } u \text{ and } v \text{ are adjacent }; \\ 0, & \text{otherwise.} \end{cases}$$
- For weighted graph G, Laplacian and normalized Laplacian can be defined in a same manner.
- The regularization of graph G: $F^T\mathcal{L}F = \frac{1}{2}\sum_{i,j=1}^{m+n} w_{i,j}\left(\frac{f_i}{\sqrt{d_{ii}}} - \frac{f_j}{\sqrt{d_{jj}}}\right)^2$.

Solution for regularization framework of Co-HITS

Solution

- $\frac{\partial R}{\partial F} = 2F^T\mathcal{L} + 2\mu(F - F^{(0)})^T = \mathbf{0}^T$, i.e., $\mathcal{L}F + \mu(F - F^{(0)}) = \mathbf{0}$.

Solution for regularization framework of Co-HITS

Solution

- $\frac{\partial R}{\partial F} = 2F^T\mathcal{L} + 2\mu(F - F^{(0)})^T = \mathbf{0}^T$, i.e.,
  $\mathcal{L}F + \mu(F - F^{(0)}) = \mathbf{0}$.
- A closed-form solution can be derived as $F = (\mathcal{L} - \mu I)^{-1}F^{(0)}$.

Solution for regularization framework of Co-HITS

Solution

- $\frac{\partial R}{\partial F} = 2F^T\mathcal{L} + 2\mu(F - F^{(0)})^T = \mathbf{0}^T$, i.e., $\mathcal{L}F + \mu(F - F^{(0)}) = \mathbf{0}$.
- A closed-form solution can be derived as $F = (\mathcal{L} - \mu I)^{-1}F^{(0)}$.
- For the large-scale graph learning problem, the matrix $\mathcal{L}$ is usually very large but sparse, which can be loaded in a relatively small storage space. However, the inverse matrix $(\mathcal{L} - \mu I)^{-1}$ will be very dense, and may need a huge space to save it.

Solution for regularization framework of Co-HITS

Solution

- $\frac{\partial R}{\partial F} = 2F^T\mathcal{L} + 2\mu(F - F^{(0)})^T = \mathbf{0}^T$, i.e., $\mathcal{L}F + \mu(F - F^{(0)}) = \mathbf{0}$.
- A closed-form solution can be derived as $F = (\mathcal{L} - \mu I)^{-1}F^{(0)}$.
- For the large-scale graph learning problem, the matrix $\mathcal{L}$ is usually very large but sparse, which can be loaded in a relatively small storage space. However, the inverse matrix $(\mathcal{L} - \mu I)^{-1}$ will be very dense, and may need a huge space to save it.
- To balance the storage space and the computation time of the inverse matrix, we suggest to approximate the $\mathcal{L}$ in a specific subgraph with a submatrix $\widehat{\mathcal{L}}$, which consists of the top-n entities according to the initial ranking scores $\widehat{F}^{(0)}$.

Solution for regularization framework of Co-HITS

Solution

- $\frac{\partial R}{\partial F} = 2F^T\mathcal{L} + 2\mu(F - F^{(0)})^T = \mathbf{0}^T$, i.e., $\mathcal{L}F + \mu(F - F^{(0)}) = \mathbf{0}$.
- A closed-form solution can be derived as $F = (\mathcal{L} - \mu I)^{-1}F^{(0)}$.
- For the large-scale graph learning problem, the matrix $\mathcal{L}$ is usually very large but sparse, which can be loaded in a relatively small storage space. However, the inverse matrix $(\mathcal{L} - \mu I)^{-1}$ will be very dense, and may need a huge space to save it.
- To balance the storage space and the computation time of the inverse matrix, we suggest to approximate the $\mathcal{L}$ in a specific subgraph with a submatrix $\widehat{\mathcal{L}}$, which consists of the top-n entities according to the initial ranking scores $\widehat{F}^{(0)}$.
- $\widehat{F} = (\widehat{\mathcal{L}} - \mu I)^{-1}\widehat{F}^{(0)}$.

Network robustness

Motivations

- In a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn.

Network robustness

Motivations

- In a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn.
- In IP networks, service providers therefore aim to monitor, manage and optimize their networks to keep their networks robust.

Network robustness

Motivations

- In a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn.
- In IP networks, service providers therefore aim to monitor, manage and optimize their networks to keep their networks robust.
- In social platforms, some external or internal events may be detected from the burst of user interaction networks.

Network robustness

Motivations

- In a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn.
- In IP networks, service providers therefore aim to monitor, manage and optimize their networks to keep their networks robust.
- In social platforms, some external or internal events may be detected from the burst of user interaction networks.

Existing measurements include:

- Node connectivity and edge connectivity

Network robustness

Motivations

- In a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn.
- In IP networks, service providers therefore aim to monitor, manage and optimize their networks to keep their networks robust.
- In social platforms, some external or internal events may be detected from the burst of user interaction networks.

Existing measurements include:

- Node connectivity and edge connectivity
- Cheeger ratio, vertex expansion, and edge expansion

Network robustness

Motivations

- In a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn.
- In IP networks, service providers therefore aim to monitor, manage and optimize their networks to keep their networks robust.
- In social platforms, some external or internal events may be detected from the burst of user interaction networks.

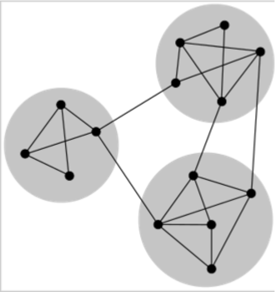
Existing measurements include:

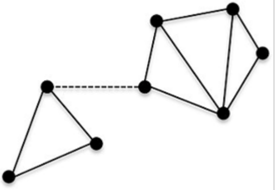
- Node connectivity and edge connectivity
- Cheeger ratio, vertex expansion, and edge expansion
- Algebraic connectivity and R-energy

Connectivity robustness

Node connectivity or edge connectivity

Node connectivity (edge connectivity) $v(G)$ ($\epsilon(G)$) of a network G is defined by the minimum number of nodes (edges) that are removed to break the networks into multiple connected components.





Expander robustness

Definitions

Let $G = (V, E)$ be a connected and undirected network.

- $\partial(S)$ is the edge boundary of S (i.e., the set of edges with exactly one endpoint in S).
- $\partial_{out}(S)$ is the outer vertex boundary of S (i.e., the set of vertices in $V \backslash S$ with at least one neighbor in S).
- $vol(S)$ is the total degree of all vertices in S.

Expander robustness

Definitions

Let $G = (V, E)$ be a connected and undirected network.

- $\partial(S)$ is the edge boundary of S (i.e., the set of edges with exactly one endpoint in S).
- $\partial_{out}(S)$ is the outer vertex boundary of S (i.e., the set of vertices in $V \backslash S$ with at least one neighbor in S).
- $vol(S)$ is the total degree of all vertices in S.

Based on above concepts, we have a of robustness metrics:

- Cheeger ratio: $h(G) = \min_{S \subset V} \frac{|\partial(S)|}{\min\{vol(S), vol(\overline{S})\}}$

Expander robustness

Definitions

Let $G = (V, E)$ be a connected and undirected network.

- $\partial(S)$ is the edge boundary of S (i.e., the set of edges with exactly one endpoint in S).
- $\partial_{out}(S)$ is the outer vertex boundary of S (i.e., the set of vertices in $V \backslash S$ with at least one neighbor in S).
- $vol(S)$ is the total degree of all vertices in S.

Based on above concepts, we have a of robustness metrics:

- Cheeger ratio: $h(G) = \min_{S \subset V} \frac{|\partial(S)|}{\min\{vol(S), vol(\overline{S})\}}$
- Vertex expansion: $h_v(G) = \min_{S \subset V, 0<|S|<|V|/2} \frac{|\partial_{out}(S)|}{|S|}$

Expander robustness

Definitions

Let $G = (V, E)$ be a connected and undirected network.

- $\partial(S)$ is the edge boundary of S (i.e., the set of edges with exactly one endpoint in S).
- $\partial_{out}(S)$ is the outer vertex boundary of S (i.e., the set of vertices in $V \backslash S$ with at least one neighbor in S).
- $vol(S)$ is the total degree of all vertices in S.

Based on above concepts, we have a of robustness metrics:

- Cheeger ratio: $h(G) = \min_{S \subset V} \frac{|\partial(S)|}{\min\{vol(S), vol(\bar{S})\}}$
- Vertex expansion: $h_v(G) = \min_{S \subset V, 0<|S|<|V|/2} \frac{|\partial_{out}(S)|}{|S|}$
- Edge expansion: $h_e(G) = \min_{S \subset V, 0<|S|<|V|/2} \frac{|\partial(S)|}{|S|}$

Spectrum robustness

Algebraic connectivity

Algebraic connectivity $\lambda(G)$ is defined by the second smallest eigenvalue of the Laplacian matrix of network G.

- $\lambda(G) \leq v(G) \leq \epsilon(G)$.
- $\lambda(G) = 0$ if G is disconnected.

Eigenvalues of normalized Laplacian

Spectrum robustness

Algebraic connectivity

Algebraic connectivity $\lambda(G)$ is defined by the second smallest eigenvalue of the Laplacian matrix of network G.

- $\lambda(G) \leq v(G) \leq \epsilon(G)$.
- $\lambda(G) = 0$ if G is disconnected.

Eigenvalues of normalized Laplacian

The eigenvalues of the normalized Laplacian matrix of a network with n vertices satisfy the following properties:

Spectrum robustness

Algebraic connectivity

Algebraic connectivity $\lambda(G)$ is defined by the second smallest eigenvalue of the Laplacian matrix of network G.

- $\lambda(G) \leq v(G) \leq \epsilon(G)$.
- $\lambda(G) = 0$ if G is disconnected.

Eigenvalues of normalized Laplacian

The eigenvalues of the normalized Laplacian matrix of a network with n vertices satisfy the following properties:

- $0 = \zeta_1 \leq \zeta_2 \leq \frac{n}{n-1} \leq \zeta_n \leq 2$, for $i = 1, 2, \cdots, n$;

Spectrum robustness

Algebraic connectivity

Algebraic connectivity $\lambda(G)$ is defined by the second smallest eigenvalue of the Laplacian matrix of network G.

- $\lambda(G) \leq v(G) \leq \epsilon(G)$.
- $\lambda(G) = 0$ if G is disconnected.

Eigenvalues of normalized Laplacian

The eigenvalues of the normalized Laplacian matrix of a network with n vertices satisfy the following properties:

- $0 = \zeta_1 \leq \zeta_2 \leq \frac{n}{n-1} \leq \zeta_n \leq 2$, for $i = 1, 2, \cdots, n$;
- $\zeta_2 = \cdots = \zeta_n = \frac{n}{n-1}$ if and only if G is a clique;

Spectrum robustness

Algebraic connectivity

Algebraic connectivity $\lambda(G)$ is defined by the second smallest eigenvalue of the Laplacian matrix of network G.

- $\lambda(G) \leq v(G) \leq \epsilon(G)$.
- $\lambda(G) = 0$ if G is disconnected.

Eigenvalues of normalized Laplacian

The eigenvalues of the normalized Laplacian matrix of a network with n vertices satisfy the following properties:

- $0 = \zeta_1 \leq \zeta_2 \leq \frac{n}{n-1} \leq \zeta_n \leq 2$, for $i = 1, 2, \cdots, n$;
- $\zeta_2 = \cdots = \zeta_n = \frac{n}{n-1}$ if and only if G is a clique;
- G has at least k connected components if and only if $\zeta_j = 0$, for $j = 1, 2, \cdots, k$.

R-energy

Definition

The robustness energy (R-energy) of G is defined as

$$E(G) = \frac{1}{n-1}\sum_{i=2}^{n}(\lambda_i - \overline{\lambda})^2,$$

where λ_i are eigenvalues of normalized Laplacian of network G, and $\overline{\lambda} = \frac{1}{n-1}\sum_{i=2}^{n}\lambda_i$.

R-energy

Definition

The robustness energy (R-energy) of G is defined as

$$E(G) = \frac{1}{n-1}\sum_{i=2}^{n}(\lambda_i - \overline{\lambda})^2,$$

where λ_i are eigenvalues of normalized Laplacian of network G, and $\overline{\lambda} = \frac{1}{n-1}\sum_{i=2}^{n}\lambda_i$.

Question:

How to compute the value of R-energy?

R-energy

Definition

The robustness energy (R-energy) of G is defined as

$$E(G) = \frac{1}{n-1} \sum_{i=2}^{n} (\lambda_i - \overline{\lambda})^2,$$

where λ_i are eigenvalues of normalized Laplacian of network G, and $\overline{\lambda} = \frac{1}{n-1} \sum_{i=2}^{n} \lambda_i$.

Question:

How to compute the value of R-energy?

- $E(G) = \frac{1}{n-1} \sum_{i=1}^{n} \sum_{j \neq i}^{n} \frac{A_{ij}}{d(v_i) d(v_j)} - \frac{n}{(n-1)^2}$ (smaller is better).

R-energy

Definition

The robustness energy (R-energy) of G is defined as

$$E(G) = \frac{1}{n-1}\sum_{i=2}^{n}(\lambda_i - \overline{\lambda})^2,$$

where λ_i are eigenvalues of normalized Laplacian of network G, and $\overline{\lambda} = \frac{1}{n-1}\sum_{i=2}^{n}\lambda_i$.

Question:

How to compute the value of R-energy?

- $E(G) = \frac{1}{n-1}\sum_{i=1}^{n}\sum_{j\neq i}^{n}\frac{A_{ij}}{d(v_i)d(v_j)} - \frac{n}{(n-1)^2}$ (smaller is better).
- $E(G)$ is reasonable robustness metric to evaluate a disconnected network.

R-energy

Definition

The robustness energy (R-energy) of G is defined as

$$E(G) = \frac{1}{n-1}\sum_{i=2}^{n}(\lambda_i - \overline{\lambda})^2,$$

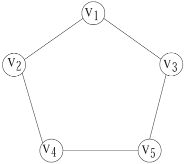
where λ_i are eigenvalues of normalized Laplacian of network G, and $\overline{\lambda} = \frac{1}{n-1}\sum_{i=2}^{n}\lambda_i$.

Question:

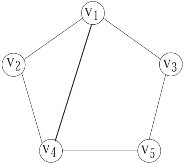
How to compute the value of R-energy?

- $E(G) = \frac{1}{n-1}\sum_{i=1}^{n}\sum_{j\neq i}^{n}\frac{A_{ij}}{d(v_i)d(v_j)} - \frac{n}{(n-1)^2}$ (smaller is better).
- $E(G)$ is reasonable robustness metric to evaluate a disconnected network.
- $E(G)$ can be efficiently computed in $O(|V| + |E|)$.

Example of network robustness metrics



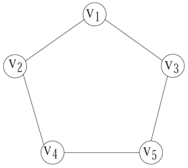
(a)



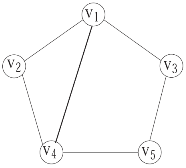
(b)

networks	Connectivity			Expansion		
	node	edge	algebraic	vertex	edge	Cheeger
Figure 1(a)	2	2	1.382	1	1	0.5
Figure 1(b)	2	2	1.382	1	1	0.5

Example of network robustness metrics



(a)



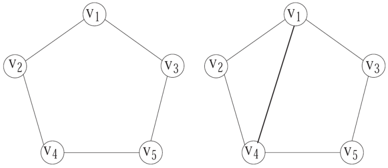
(b)

networks	Connectivity			Expansion		
	node	edge	algebraic	vertex	edge	Cheeger
Figure 1(a)	2	2	1.382	1	1	0.5
Figure 1(b)	2	2	1.382	1	1	0.5

Analysis

- The table illustrates that they are unreasonable to evaluate the robustness of networks.

Example of network robustness metrics



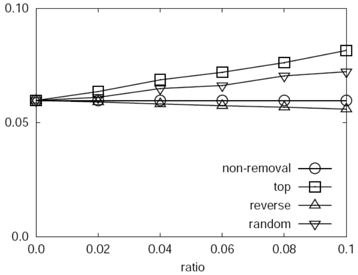
(a) (b)

networks	Connectivity			Expansion		
	node	edge	algebraic	vertex	edge	Cheeger
Figure 1(a)	2	2	1.382	1	1	0.5
Figure 1(b)	2	2	1.382	1	1	0.5

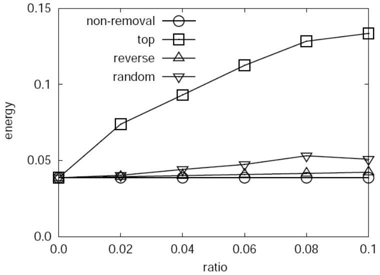
Analysis

- The table illustrates that they are unreasonable to evaluate the robustness of networks.
- The R-energies of networks shown in Figures (a) and (b) are 0.222 and 0.074, respectively. Thus, R-energy is more reasonable.

R-energy: application I

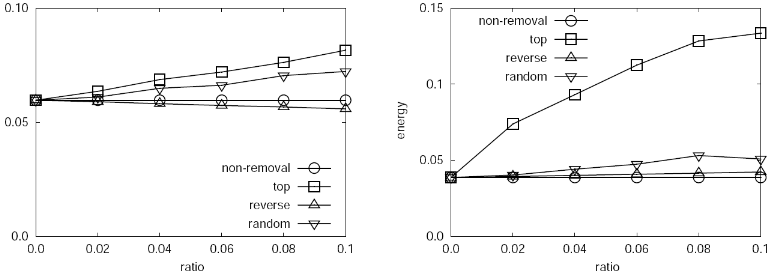


(a) CA-HepPh



(b) Email-Enron

R-energy: application I

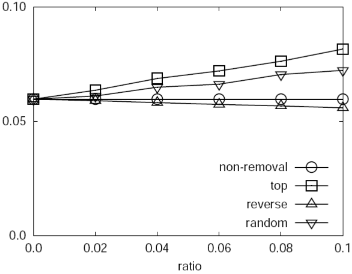


(a) CA-HepPh (b) Email-Enron

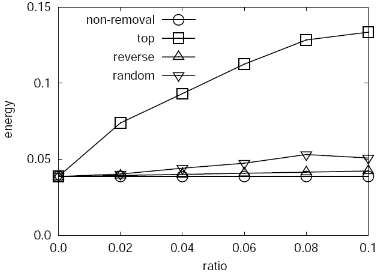
Analysis

- Networks become less robust sooner when vertices of the highest degrees are removed.

R-energy: application I



(a) CA-HepPh

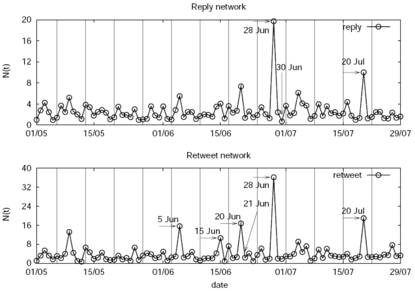


(b) Email-Enron

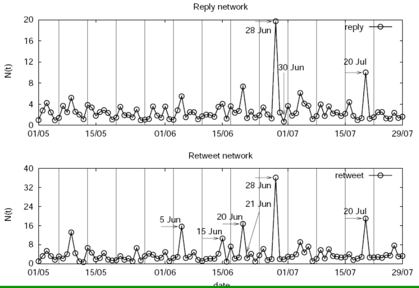
Analysis

- Networks become less robust sooner when vertices of the highest degrees are removed.
- Networks remain robust or become slightly more robust when vertices of the smallest degrees are removed.

R-energy: application II



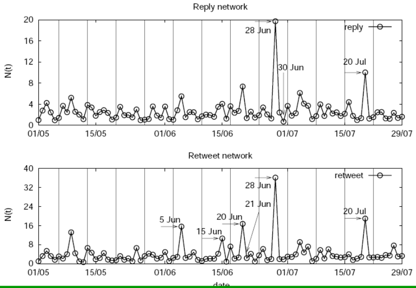
R-energy: application II



Analysis

- On June 28, the top three words from retweets with highest frequency difference are “tax”, “Obamacar” and “scotu”. Actually, the Obamacare healthcare law was upheld by the Supreme Court of United States, and there were concerns about tax increase as its outcome.

R-energy: application II



Analysis

- On June 28, the top three words from retweets with highest frequency difference are “tax”, “Obamacar” and “scotu”. Actually, the Obamacare healthcare law was upheld by the Supreme Court of United States, and there were concerns about tax increase as its outcome.
- Twitter goes down in worst crash in 8 months on July 20.

Take-home messages

- Centrality
  - Degree
  - Eigenvector
  - Closeness
  - Betweenness
  - Clustering coefficient
- Weighted graph centrality
- PageRank and HITS
  - PageRank
  - HITS
  - Co-HITS
- Graph robustness

Acknowledgements

- Data science sources from CMU: http://eliassi.org/datasci15syllabus.htm
- Weighted graph centrality
- PageRank and HITS
- Graph robustness