This session will be taught in Hybrid format

When joining on Zoom

- When your microphone or camera are switched on you will be able to be heard in the teaching room and may be seen on the screen in the room.
- If the screen is recorded, you may be captured on the recording if you choose to contribute and have your camera switched on.
- Chat messages to everyone and privately to the host may be visible on screen in the room.

When joining in the Teaching room

- Academic teaching staff should let you know when Zoom is connected and you may see the Zoom meeting on the screen.
- Once connected to Zoom, microphones may capture sound from the whole room. Your voice may be shared to Zoom when speaking at a normal talking volume.
- Once connected to Zoom, cameras will capture video images from the room. These may be shared to Zoom. The camera capture will be a wide shot of the whole room or block of seating. It will not capture close-up images of individual students.
- If the session is recorded, video from the room will be captured on the recording. Your voice will be captured on the recording if you choose to contribute.
- If you connect to the Zoom meeting from within the room (for example to use the chat), please keep your audio on mute or use headphones to avoid feedback issues with the room audio.

The recordings may be stored in the cloud ad any personal information within the recording will be processed in accordance to the General Data Protection Regulations 2018.
A substantial contribution is considered to be anything more than merely answering questions or participating in a group discussion. Where you make a substantial contribution to the delivery of the recorded events, a signed consent form will be obtained prior to the recording being made available for viewing. The Consent Form will address your personal informationand any copyright or other intellectual property in the recording.

(Complex) Networks Analysis

Centrality Analysis (Part II): Tutorial

Felipe Orihuela-Espina

November 12, 2024

Table of Contents

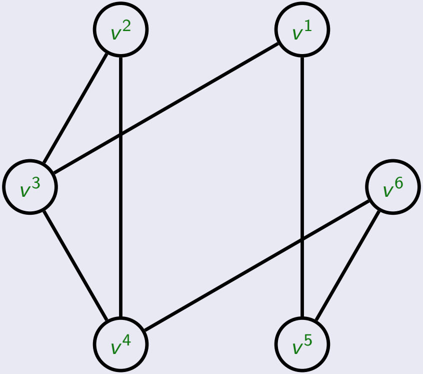
Section 1

Closeness centrality

PageRank

Exercise

Let be the network in the Figure below. Calculate the closeness centrality for all nodes and report the most central node(s).

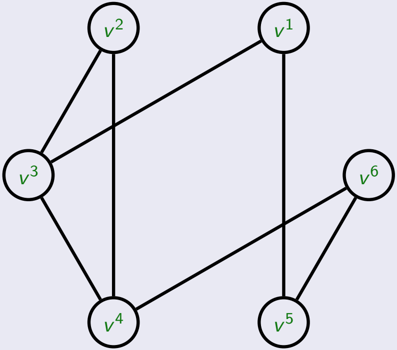


Closeness centrality

Exercise

Answer:

Let's start by calculating all pairwise distances (symmetric for an unweighted undirected graph)

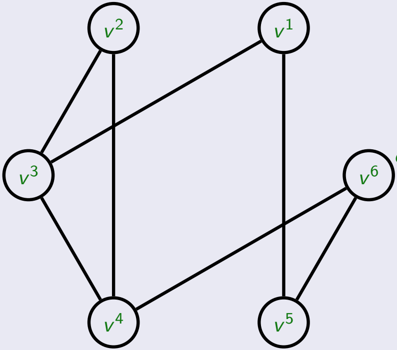


$$dist = \left\{ \begin{array}{c|cccccc} & v^1 & v^2 & v^3 & v^4 & v^5 & v^6 \\ \hline v^1 & 0 & 2 & 1 & 2 & 1 & 2 \\ v^2 & 2 & 0 & 1 & 1 & 3 & 2 \\ v^3 & 1 & 1 & 0 & 1 & 2 & 2 \\ v^4 & 2 & 1 & 1 & 0 & 2 & 1 \\ v^5 & 1 & 3 & 2 & 2 & 0 & 1 \\ v^6 & 2 & 2 & 2 & 1 & 1 & 0 \end{array} \right.$$

Closeness centrality

Exercise

Answer (Cont.):
Now we can calculate (using either rows or columns) the total sum of distances from each to node to all other nodes.

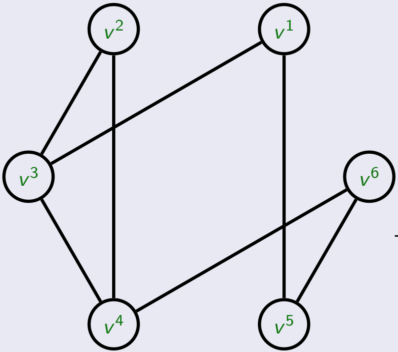


$$dist = \left\{ \begin{array}{c|cccccc|c} & v^1 & v^2 & v^3 & v^4 & v^5 & v^6 & \sum_{v^j} d(v^i, v^j) \\ \hline v^1 & 0 & 2 & 1 & 2 & 1 & 2 & 8 \\ v^2 & 2 & 0 & 1 & 1 & 3 & 2 & 9 \\ v^3 & 1 & 1 & 0 & 1 & 2 & 2 & 7 \\ v^4 & 2 & 1 & 1 & 0 & 2 & 1 & 7 \\ v^5 & 1 & 3 & 2 & 2 & 0 & 1 & 9 \\ v^6 & 2 & 2 & 2 & 1 & 1 & 0 & 8 \\ \hline \sum_{v^j} d(v^i, v^j) & 8 & 9 & 7 & 7 & 9 & 8 & \end{array} \right.$$

Closeness centrality

Exercise

Answer (Cont.):
Finally, we simply calculate the closeness centrality by calculating inverse. We can also normalize.



	v^1	v^2	v^3	v^4	v^5	v^6
$\sum_{v^j} d(v^i, v^j)$	8	9	7	7	9	8
$\frac{1}{\sum_{v^j} d(v^i, v^j)}$	1/8	1/9	1/7	1/7	1/9	1/8
$\frac{\lvert V \rvert - 1}{\sum_{v^j} d(v^i, v^j)}$	5/8	5/9	5/7	5/7	5/9	5/8

The most central nodes are v^3 and v^4.

Section 2

PageRank

Random networks

Algorithm 1: PageRank

Data: G: A directed graph
Data: $d \in [0,1] \subset \mathbb{R}$: A dumping parameter
Data: $\mathbf{e} : e^i \in \mathbb{R}$: A rank sources vector
Data: $\mathbf{p0} : p0^i \in \mathbb{R}$: PageRank scores initialization vector (initial prestiges)
Data: $maxIter \in \mathbb{N}$: Number of maximum iterations
Data: $tol \in \mathbb{R}$: Tolerance error
Data: $Dangling$: Outedges to be assigned to any dangling nodes
Result: $\mathbf{p} : p^i \in \mathbb{R}$: PageRank scores (final prestiges)

/* Initialization: */
/* - Set up the various parameters and build the normalized adjacency matrix */
$A \leftarrow adjacency(G)$ /* Retrieve adjacency matrix */
if $A = \emptyset$ **then**
  $\mathbf{p} \leftarrow \emptyset$
end if
$A^{norm} \leftarrow A$
$a^{norm}_{ij} \leftarrow \frac{a^{norm}_{ij}}{\sum_j a^{norm}_{ij}}$ /* Normalize */
if $\sum_i p0^i \neq 1$ **then**
  $p0^i \leftarrow \frac{p0^i}{\sum_i p0^i}$
end if

Algorithm 1: PageRank (cont)

if $\sum_i e^i \neq 1$ **then**
  $e^i \leftarrow \frac{e^i}{\sum_i e^i}$
end if
$Dangling_{ij} \leftarrow \frac{Dangling_{ij}}{\sum_j Dangling_{ij}}$ /* Normalize dangling */
$DanglingNodes \leftarrow v^i : \sum_j a_{ij} = 0$
for $v^i \in DanglingNodes$ **do**
  $a^{norm}_{ij} \leftarrow Dangling_{ij}$
end for
/* Main loop: */
$\mathbf{p} \leftarrow \mathbf{p0}$
for $iter = 1 : maxIter$ **do**
  $\mathbf{p}^{last} \leftarrow \mathbf{p}$
  $\mathbf{p} \leftarrow d({A^{norm}}^T * \mathbf{p}^{last}) + (1-d)\mathbf{e}$
  /* Check convergence, using L1 norm */
  if $\left\|\mathbf{p} - \mathbf{p}^{last}\right\|_1 < tol$ **then**
    break
  end if
end for

PageRank

Exercise

Implement the PageRank algorithm in some programming language.
Tip: In this exercise do not aim for efficiency or code elegance; instead focus on closely following the pseudo-code provided.
NOTE: Solution is provided in MATLAB.

PageRank

Exercise

Implement the PageRank algorithm in some programming language.
Tip: In this exercise do not aim for efficiency or code elegance; instead focus on closely following the pseudo-code provided.
NOTE: Solution is provided in MATLAB.

Answer:
Please open code:

`IDA2023_0005_CNA_PageRank.m`

MATLAB's internal algorihtm is reported here:

https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/moler/exm/chapters/pagerank.pdf

However, the solution provided follows the one in:
https://www.geeksforgeeks.org/page-rank-algorithm-implementation/
...which I feel is more didactic (though less efficient).

PageRank

Exercise

Create a 4 nodes graph where B had a link to pages C and A, page C had a link to page A, and page D had links to all three pages. A is a dangling node (no out-degree).

Compare your PageRank implementation outcome against some existing implementation.

NOTE: Solution is provided in MATLAB.

PageRank

Exercise

Create a 4 nodes graph where B had a link to pages C and A, page C had a link to page A, and page D had links to all three pages. A is a dangling node (no out-degree).
Compare your PageRank implementation outcome against some existing implementation.
NOTE: Solution is provided in MATLAB.
Answer:
Please open code:

`IDA2023_0005_CNA_PageRank.m`

Thank you! Questions?