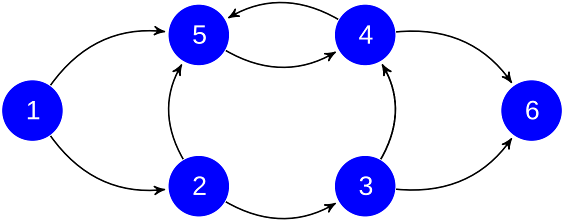
PageRank: Ranking of nodes in graphs

Gonzalo Mateos
Dept. of ECE and Goergen Institute for Data Science
University of Rochester
gmateosb@ece.rochester.edu
http://www.ece.rochester.edu/~gmateosb/

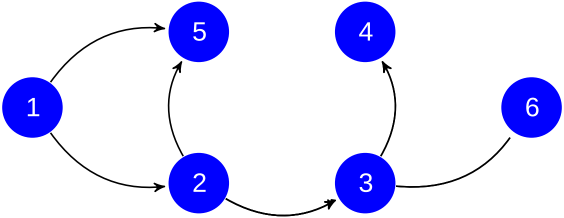
October 15, 2019

Graphs



- **Graph** ⇒ A set of V of vertices or nodes $j = 1, \ldots, J$
  - ⇒ Connected by a set of edges E defined as ordered pairs (i, j)
- In figure ⇒ Nodes are $V = \{1, 2, 3, 4, 5, 6\}$
  - ⇒ Edges $E = \{(1, 2), (1, 5), (2, 3), (2, 5), (3, 4), \ldots$
    $(3, 6), (4, 5), (4, 6), (5, 4)$

How well connected nodes are?



Connectivity ranking

- Key insight: There is information in the structure of the network
- Knowledge is distributed through the network
  - The network (not the nodes) knows the rankings

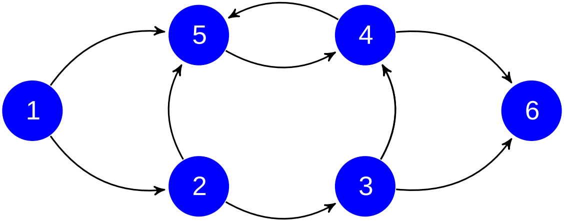
- Idea exploited by Google's PageRank to rank webpages

  ... by social scientists to study trust & reputation in social networks

  ... by ISI to rank scientific papers, transactions & magazines ...

Preliminary definitions

- Graph $G = (V, E) \Rightarrow$ vertices $V = \{1, 2, \ldots, J\}$ and edges E



Definition of rank

- Agent A chooses node i, e.g., web page, at random for initial visit
- Next visit randomly chosen between links in the neighborhood $n(i)$
  - All neighbors chosen with

Equiprobable random walk

- Formally, let A_n be the node visited at time n
- Define transition probability P_{ij} from node i into node j

$$P_{ij} := P \quad A_{n+1} = j \quad A_n = i$$

- Next visit equiprobable among js

Formal definition of rank

- Def: Rank r_i of i-th node is the time average of number of visits

$$r_i := \lim_{n\to\infty} \frac{1}{n} \sum_{m=1}^{n} \mathbb{I}\{A_m = i\}$$

  - ⇒ Define vector of ranks $\mathbf{r} := [r_1, r_2, \ldots, r_J]^T$

- Rank r_i can be approximated by average r_{ni} at time n

$$r_{ni} := \frac{1}{n} \sum_{m=1}^{n} \mathbb{I}\{A_m = i\}$$

  - ⇒ Since $\lim_{n\to\infty} r_{ni} = r_i$, it holds $r_{ni} \approx r_i$ for n sufficiently large
  - ⇒ Define vector of approximate ranks $\mathbf{r}_n := [r_{n1}, r_{n2}, \ldots, r_{nJ}]^T$

- If modified graph is connected, rank independent of initial visit

Ranking algorithm

```
Output : Vector $r(i)$ with ranking of node $i$
Input  : Scalar $n$ indicating maximum number of iterations
Input  : Vector $N(i)$ containing number of neighbors of $i$
Input  : Matrix $N(i;j)$ containing indices $j$ of neighbors of $i$

m = 1;  r=zeros(J,1); % Initialize time and ranks
$A_0$ = random(`unid',J); % Draw first visit uniformly at random
while $m < n$ do
    jump = random(`unid',$N_{A_{m-1}}$);  % Neighbor uniformly at random
    $A_m = N(A_{m-1}, \text{jump})$;  % Jump to selected neighbor
    $r(A_m) = r(A_m) + 1$;  % Update ranking for $A_m$
    $m = m + 1$;
end
$r = r/n$;  % Normalize by number of iterations n
```

Ranked class graph

Convergence metrics

- Recall r is vector of ranks and r_n of rank iterates
- By definition $\lim_{n\to\infty} r_n = r$. How fast r_n converges to r (r given)?
- Can measure by ℓ_2 distance between r and r_n

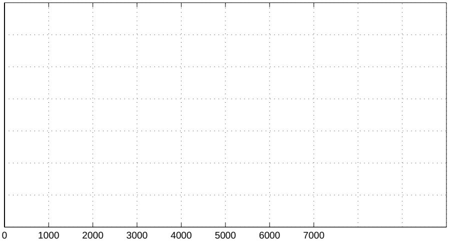
$$\epsilon_n := \|r - r_n\|_2 = \left(\sum_{i=1}^{J} (r_{ni} - r_i)^2 \right)^{1/2}$$

- If interest is only on highest ranked nodes, e.g., a web search
  - ⇒ Denote $r^{(i)}$ as the index of the i-th highest ranked node
  - ⇒ Let $r_n^{(i)}$ be the index of the i-th highest ranked node at time n
- First element wr]TJ/F28 9.k Td at fgb ll

Evaluation of convergence metrics

Distance

First element wrongly ranked



- Distance close to 10^2 in 5 10^3 iterations

- Bad: Two highest ranks in 4 10^3 iterations
- Awful: Six best ranks in 8 10^3 iterations

- (Very) slow convergence

When does this algorithm converge?