CS-E5740 Complex Networks, Answers to exercise set 5

Sara Cabodi, Student number: 784287

October 28, 2019

Compile with `pdflatex ex_template.tex`

Problem 1

a) I loaded the network given in file *pagerank_network.edg* and the visualization of the network is reported below.

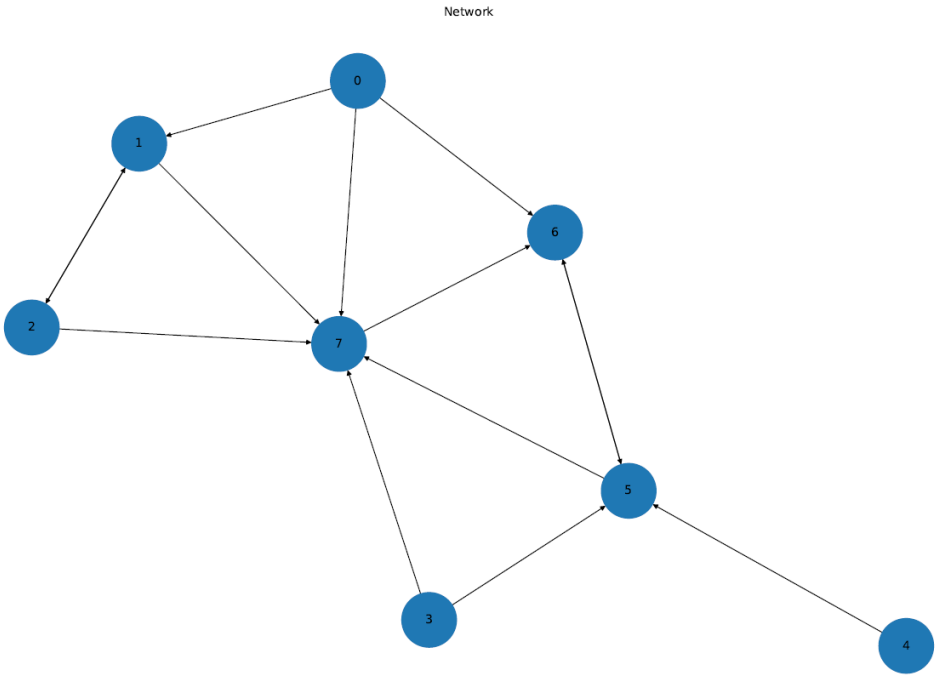
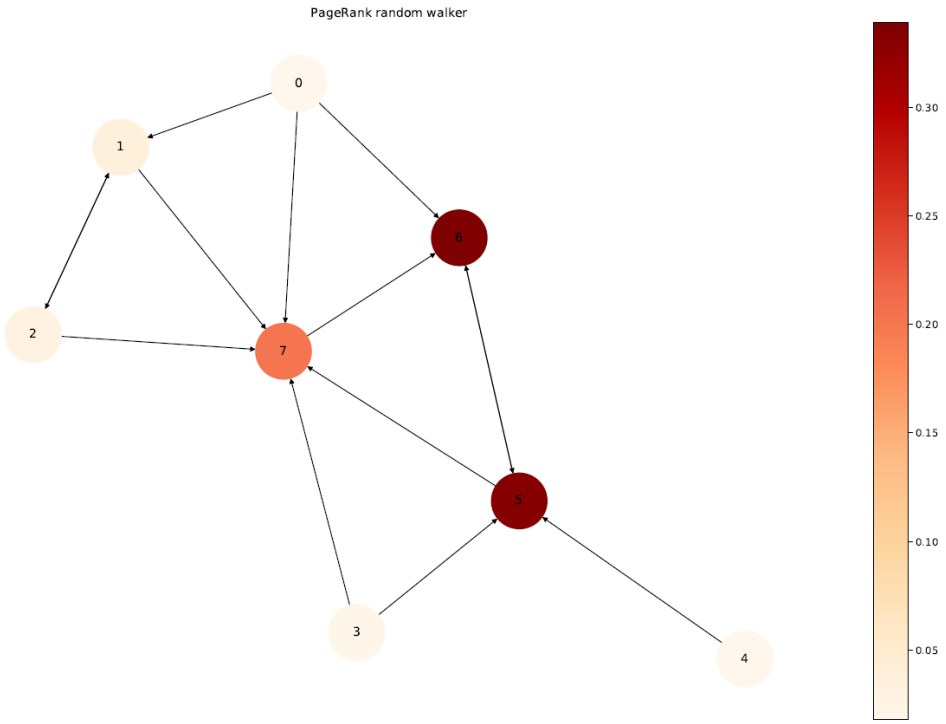
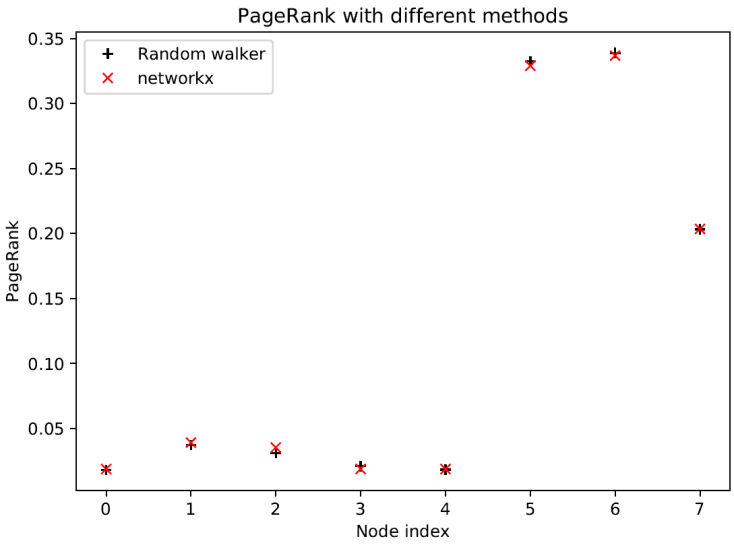


Figure 1: Pagerank network

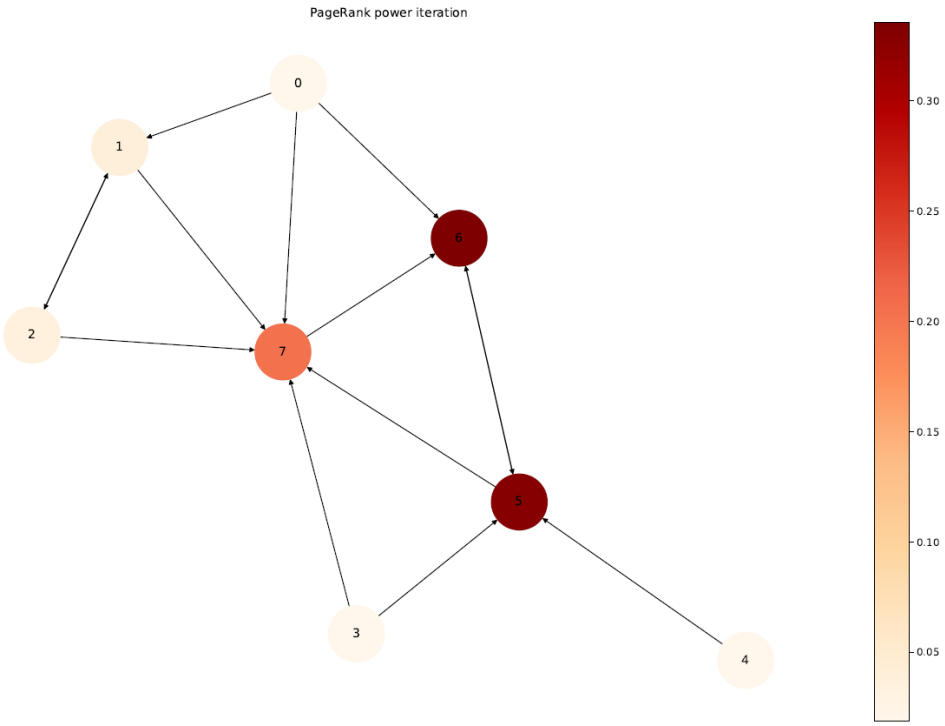
b) The result of the computation of PageRank on the network obtained from point a) by simulating a random walker with $d = 0.85$ and $N_{steps} = 10000$ is shown below



The plot of the comparison between the results obtained with the random walker and the function *nx.pagerank* shows that the results obtained by the random walker matches the ones got from the networkX funtion



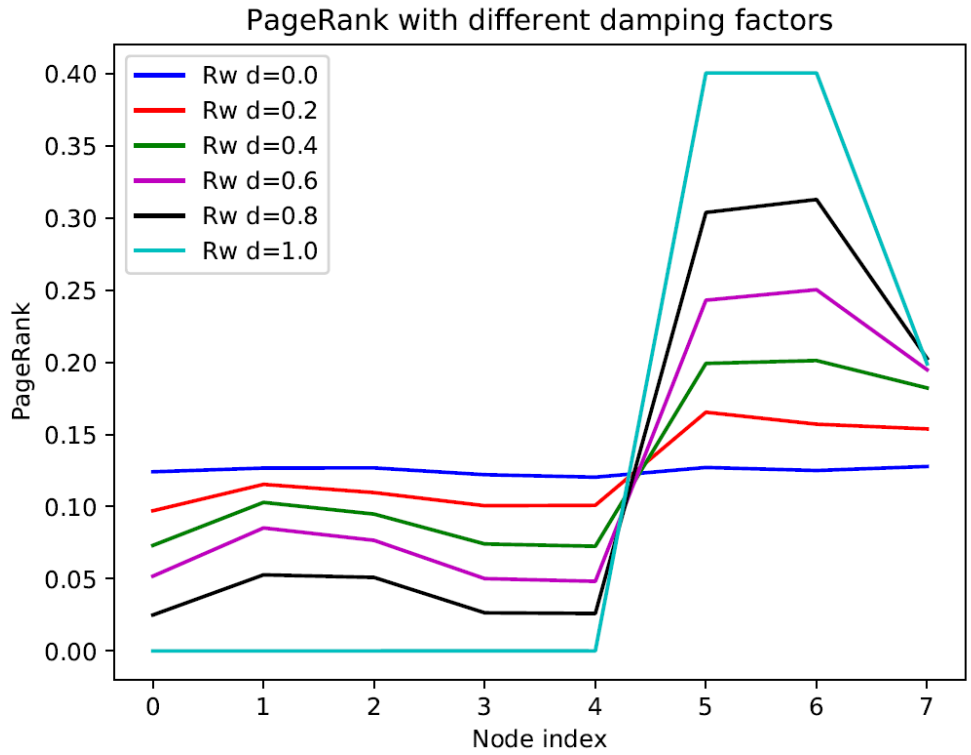
c) The result of the computation of PageRank on the network obtained from point a) using a power iteration algorithm with $d = 0.85$ and $N_{iterations} = 10$ is the following



d) I based my estimation on a network of 1000 nodes where each node has in and out degrees of 5 and for the random walk trial I considered $N_{steps} = 1000 * N_{nodes}$ and $N_{iterations} = 10000$.
The random walker computation took approximately 18.4 seconds, while the one with power iteration took around 0.5367 seconds.
To calculate the estimation for the RankPage for 26 million webpages, I multiplied both times for a term $t = 26 * 10^6/10^3 = 26 * 10^3$ and then dividing the result for 3600, obtaining the following rought estimation of hours: $h_{rw} = 132.88$ hours and $h_{pi} = 3.87$ hours.

e) The PageRank of a page depends on its in_degree and the importance of the pages that link to it. Whenever in a directed graph we identify strongly connected components (SCCs), we can identify the DAG of SCCs: given the topological order of SCCs, one could apply the power iteration algorithm on sub-networks given by SCCs, sorted by increasing topological order, so that each SCC has incoming edges only from internal nodes or nodes from already processed SCCs. I expect a faster convergence of the algorithm, as ranks of nodes in other SCCs (with outgoing edges to the current one) are already stable.

f) The value of the damping factor d represents the probability for the random walker to move to one of the neighbours of the current node. When $d = 0$ node choice is fully random on a global basis, so the algorithm should converge faster, with uniform rank values (shown in the figure with blue line). On the other hand, $d = 1$ makes rank values fully dependent on in_degrees and local walks, so we should expect a less uniform distribution. Intermediate values of d makes the ranks one ranging between the two extremes and being more compliant to the distribution obtained with $d = 0.85$ in the other points of the problem.



g) The results obtained by the *wikipedia_network* analysis are shown below.

```
---Highest PageRank:---
0.03519319071432259 : Graph_theory
0.02036135061984469 : Social_network
0.016771511398301818 : Mathematics
0.016462083632076074 : Social_network_analysis
0.014703296264824403 : Social_networking_service
---Highest In-degree:---
82 : Social_network
73 : Social_network_analysis
63 : Small_world_experiment
62 : Social_networking_service
62 : Orkut
---Highest Out-degree:---
140 : Network_science
82 : Social_network
73 : Social_network_analysis
67 : Small-world_network
65 : Small_world_experiment
```

The analysis highlights a higher correlation between PageRank and in_degree, in particular the first one depends on the second one. This is shown in the case of social network. As for the out_degree, it helps distribute the contribution of the in_degree (it usually lowers the PageRank), but there are not other signs of correlation between the two measures.