Airport network

Lincetto Riccardo 1156313 – Homework 2

Link analysis

In an airport network it doesn't make much sense to distinguish between qualities of a node as a "content provider" or as an "expert" because they are not related to what the network tries to capture. Computing then hub and authority scores with HITS algorithm isn't meaningful because the best hubs would result in being the best authorities too. This is due even to the fact that the network is almost symmetrical. It is much more interesting then to assess airports importance with the PageRank algorithm, because each node is ranked with a single value which is coherent with the network structure. Nodes with the best ranks are anyway the airports with the highest degrees. The following table displays the non-ordered top 16 airports according to 4 different algorithms: the first one is my version of PageRank which exploits the teleportation concept, the others are the ones already implemented in MATLAB and they compute respectively PageRank, hub and authority scores.

Rank	PageRank	Hub	Authority
ATL	ATL	BRU	BRU
ORD	ORD	CDG	CDG
DFW	DFW	IST	IST
CDG	CDG	FRA	FRA
IST	IST	LHR	LHR
DEN	DEN	AMS	AMS
IAH	IAH	DUB	DUB
JFK	JFK	LGW	LGE
LAX	LAX	MAN	MAN
FRA	FRA	FCO	FCO
AMS	AMS	MAD	MAD
DXB	DXB	BCN	BCN
PEK	PEK	DUS	DUS
SYD	SYD	MUC	MUC
DME	DME	ZRH	ZRH
YYZ	YYZ	VIE	VIE

My version of algorithm converges in 168 iterations, where to declare convergence it is used a parameter $\epsilon = 10^{-15}$ and the following condition: the norm of the vector of ranks computed at the current iteration minus the one computed at the previous iteration must be smaller than ϵ, where the choice of such a small ϵ is allowed by the fact that the algorithm is quite fast. It is possible to see from the above table that my version of PageRank gives the same results of the built-in version of MATLAB and that, as previously said, hub and authority scores give the same list of nodes, which is anyway different from the one obtained with PageRank. These considerations can be extended to Fig.1 and Fig.2, where the scores obtained for each node are compared (for a better resolution check *ranks_large.bmp* and *hubauth_large.bmp* in Image folder): here it is possible to see that there is a small displacement between consistent scores, but this is negligible in practice.

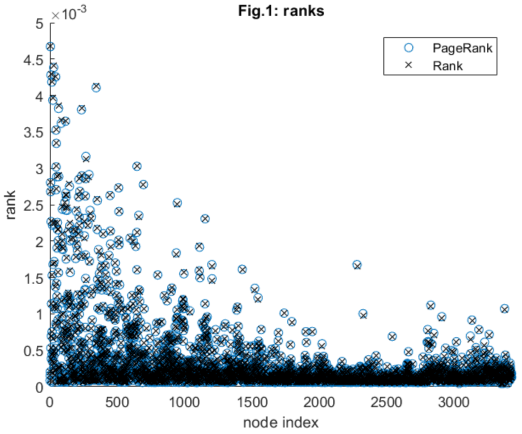

Fig.1: ranks

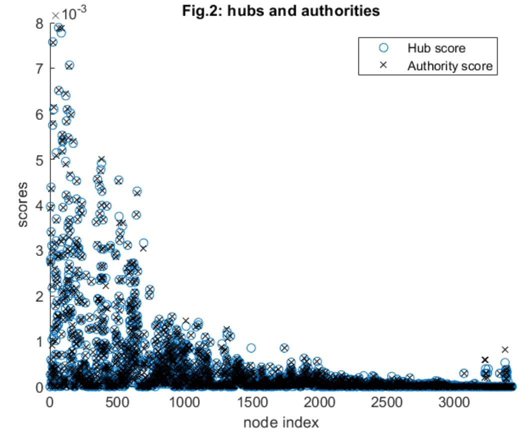

Fig.2: hubs and authorities

Link prediction

For general airport networks the link prediction problem requires a lot of effort because having a meaningful prediction would require considering even geographical distances between airports instead of focusing only on common neighbours, as most of local techniques do. For this reason, I decided to keep computational complexity as low as possible implementing the preferential attachment technique to predict links. In the MATLAB framework a similarity matrix S is computed with the values s_{ij} indicating how likely there will be a link between to disconnected nodes i, j. Moreover, this technique proved to give good results in the US airport network according to Lu and Zhou (2011) – *"Link prediction in complex networks: a survey*".

Community detection

To split the airport network into communities I decided to use the spectral clustering algorithm: the value of k has been decided heuristically looking at the "eigengaps" of the sorted list of the smallest eigenvalues of the normalized Laplacian matrix. These eigenvalues are reported in Fig.3.

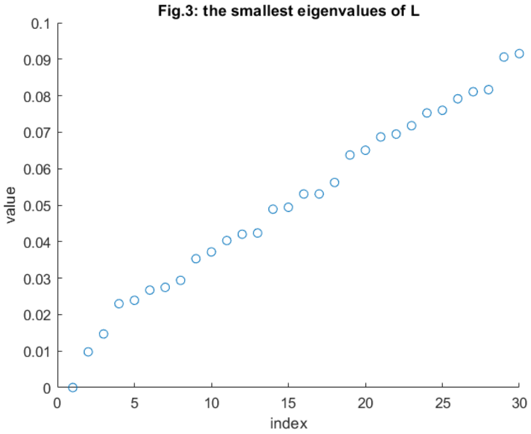
Fig.3: the smallest eigenvalues of L

From this image it makes sense to set $k = 2$ neglecting the null eigenvalue and then, representing each node of the giant component in the bidimensional space generated by v_{N-1}, v_{N-2}, Fig.4 is obtained (check in the image folder *clusters_large.bmp* for a better visualization).

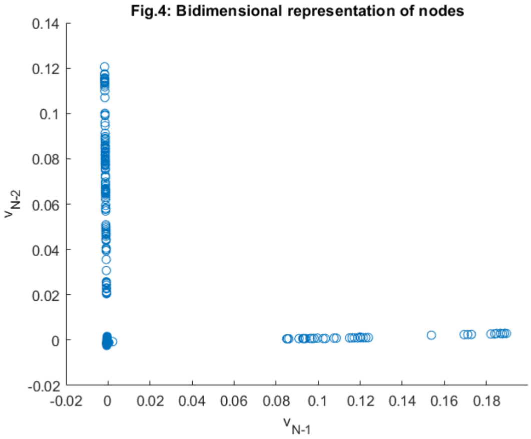
Fig.4: Bidimensional representation of nodes

From this image we can recognize 3 main clusters and using k-means algorithm to split the network in 3 communities gives as result a network partitioned as in Fig.5.

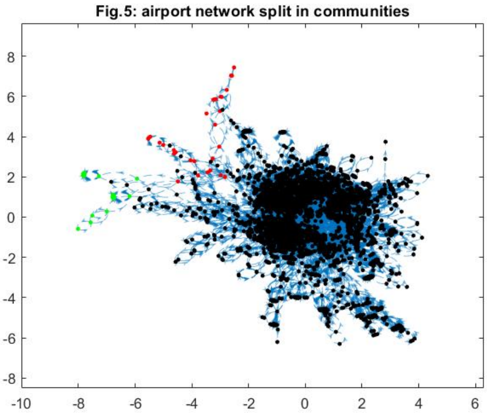
Fig.5: airport network split in communities

In this figure the 2 small communities highlighted in red and green dots are made of marginal nodes, in the sense that they are weakly connected to the rest of the giant component. The main problem of this partitioning is the unbalanced number of nodes belonging to each community: in the first one there are 3326 airports out of 3378, while in the other ones 20 and 32 respectively. Solving this problem might require exploiting modularity to understand which is the best number of clusters to split the network in, combined to using a higher dimensional space to represent each node.