Social network analysis

Motahhareh Nadimi
Data Science -University of Trieste
2018-2020

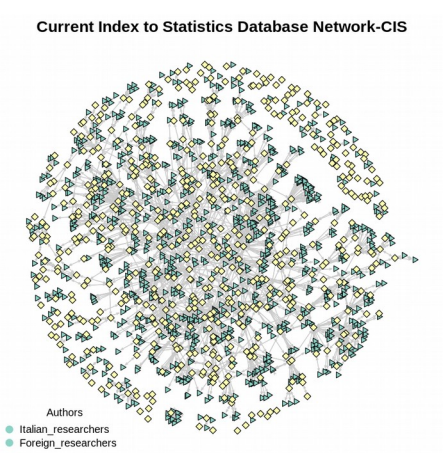
Contents:

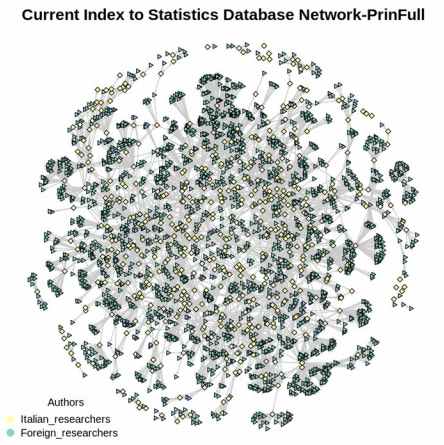
Introduction:

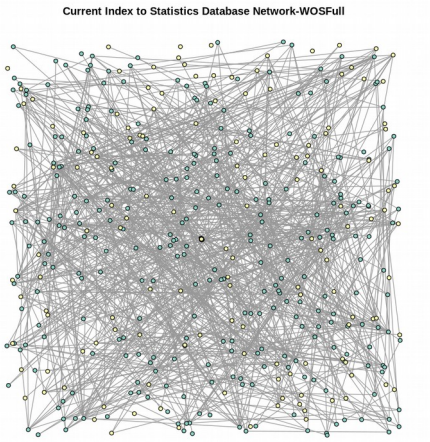
DATA SET:PrinFULL, WOSFull and CISFull NETWORK

Loading and pre processing the data
In this step, I did pre processing step, because attribute data frame only include for Italian authors, so I added the attribute for Italian_researchers and named other authors as Foreign_researchers. In the Descriptive analysis and community detection, I worked on the whole data set just in the Ergm part, induced sub graph of Italian authors has been used.

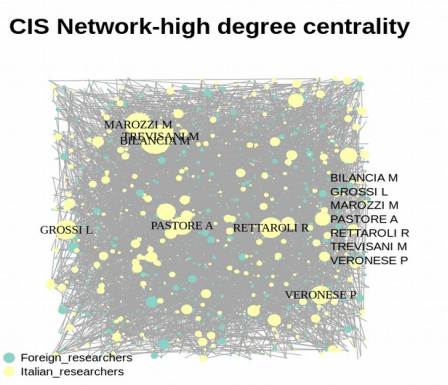
Visualizing Large Network:
The PrinFull network has 2839 vertices and total edges is equal to 9379, this characteristics for CisFull data 1525 vertices and 2534 edges and for the WOSFull network the node count is 5291 and edge count is 427238.

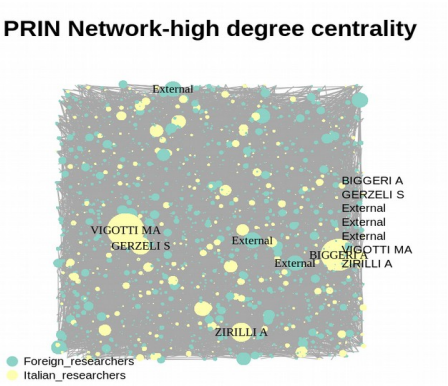


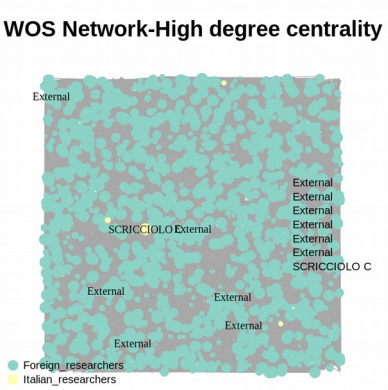




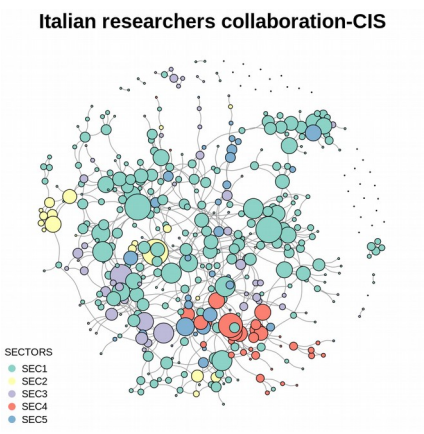
Visualizing the network consisting of only these authors belonging to the most represented nodes in terms of degree. Also the highest node belongs to sector 1 in each network.

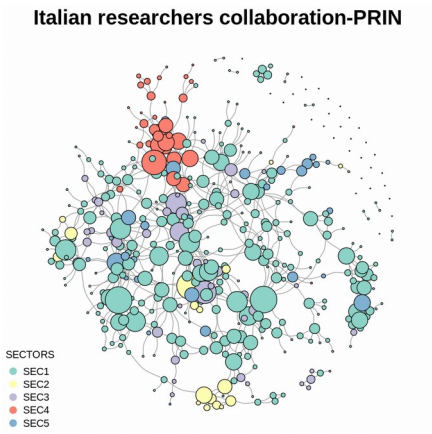


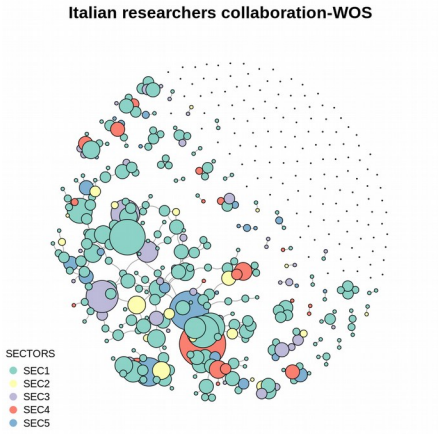




Visualizing based on the Italian researchers
The size of nodes is proportional to degree, most of vertices belongs to sector one, and for other sectors it seems that they are sparse graphs.



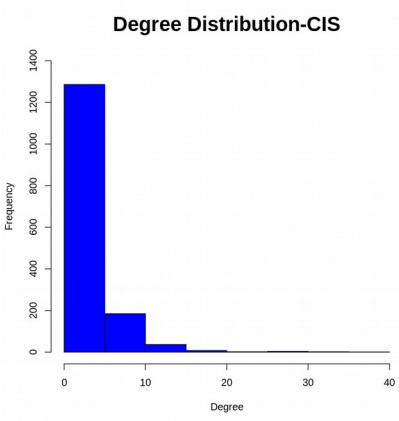


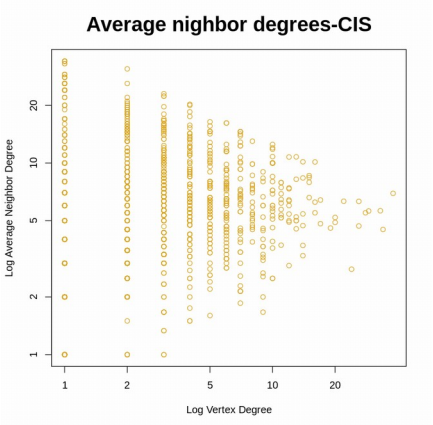


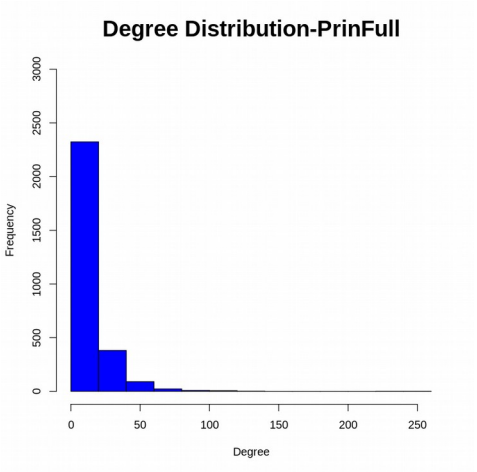
Descriptive analysis:

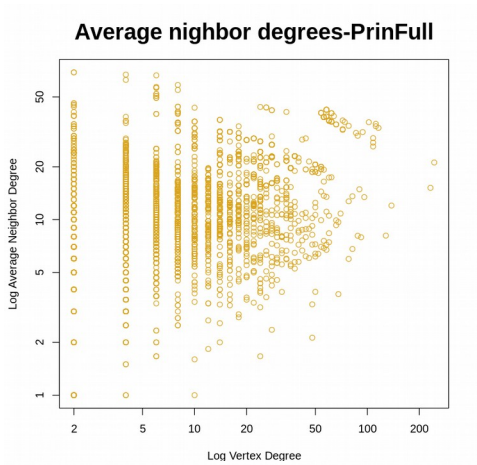
	CisFull	PrinFull	WOSFull
size of whole network	1525	2839	5291
size of Italian network	465	440	469
Density	0.002	0.002	0.03
Edge count	2534	9379	426435
Isolated node	60	7	26
The size of largest component	1277	2696	4852
The longest shortest path	19	17	16
Degree centrality	0.02	0.04	0.20
Closeness centrality	0.001	0.0008	0.0003
Eigen centrality	0.99	0.98	0.85
Betweenness centrality	0.09	0.16	
Count of component	114	27	103
Transitivity	0.29	0.53	0.90

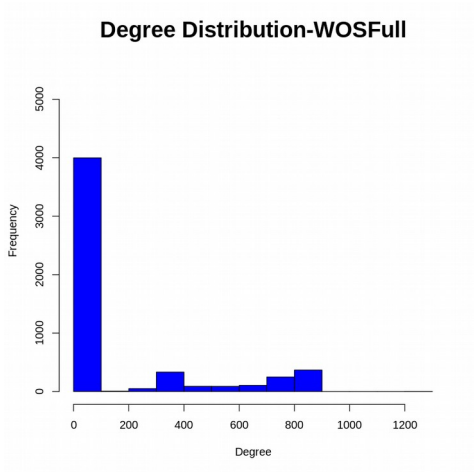
Degree Visualizing:
While there is a substantial number of nodes of quite low degree, there are also a small number of nodes with higher order of degree magnitude.
The plot "Log Average Neighbor Degree" suggests that while there is a tendency of nodes of higher degrees to link with similar nodes, nodes of lower degree tend to link with nodes of both lower and higher degrees.

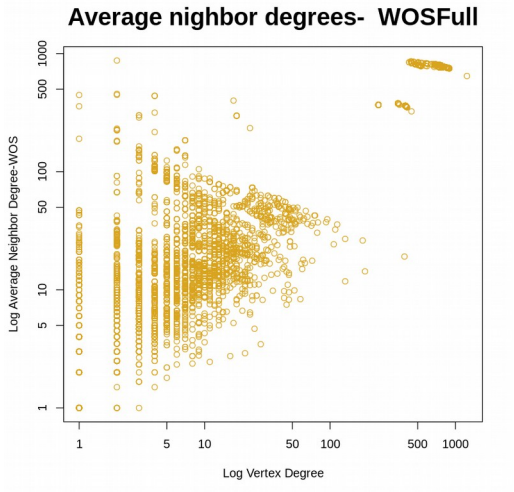












The summery of degree is as below:

Degree	Min	1st Quarter	Median	Mean	3rd Qu	Max
CisFull	0	1	2	3.3	4	38
PrinFull	0	2	6.6	8	8	122
WOSFull	0	5	10	161.5	52	1234

Correlations among a set of centrality measures for PrinFull network:
this number for CisFull is 0.7%. I couldn't calculate for the WOSFull.

	Degree	Betweenness
Degree	1	53%
Betweenness	53%	1

Clique:
The size of the largest clique for CisFull, PrinFull AND WOSFull respectively: 11, 29, 679.
Summery of clique of Italian researcher for PrinFull network: there are 277 nodes (cliques of size one) and 387 edges (cliques of size two), followed by 132 triangles (cliques of size three).

Clique	1	2	3	4	5
PrinFull	277	387	132	18	1

Clique	1	2	3	4			
CisFull	465	524	146	1			

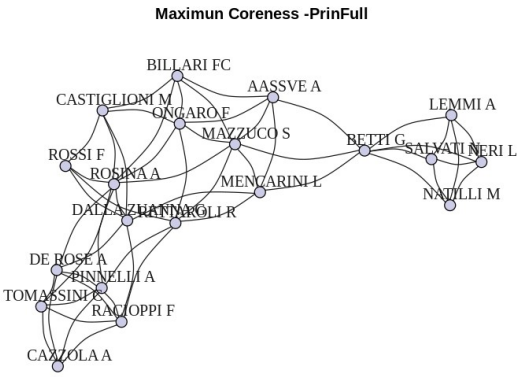
Clique	1	2	3	4	5	6	7
WOSFull	469	399	129	41	21	7	1

Visualizing the largest clique for PrinFull: can be find in the code.

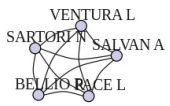
The Coreness Layout plot:
A k-core is a maximal sub graph where each vertex is connected to at least k others.
For example for PrinFull network, the network is made up of the highest k-core is mot in the center of network.

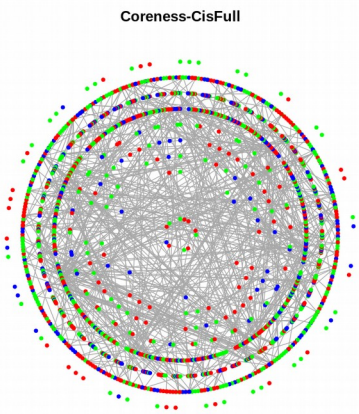
The plot in right, shows visualization about vertices in max core.

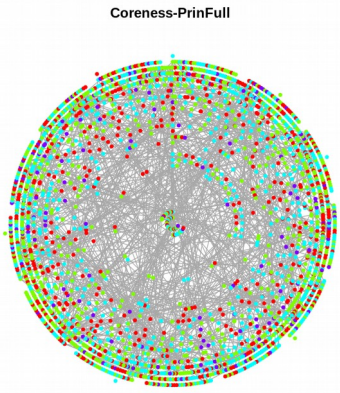
CisFull		PrinFull		WOSFull	
Core	Vertices	core	Vertices	core	Vertices
0	90	0	3	0	141
1	146	1	102	1	146
2	149	2	143	2	149
3	80	3	140	3	26
		4	25	6	7

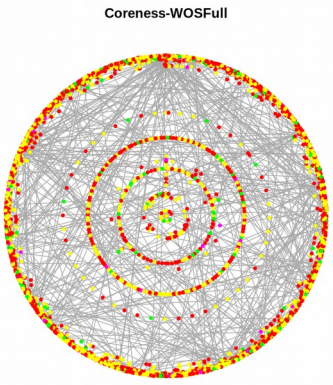


Visualization of k core :









Community detection analysis:

A census of all connected components within this graph shows that there is clearly a giant component. This component contains more than 90% of the vertices in all the network.

PrinFull	1	2	3	4	6	7	13	14	18	42	2696
	7	8	3	1	1	2	1	1	1	1	1

CisFull	1	2	3	4	6	7	8	11	14	1277
	60	29	8	6	3	1	1	2	1	1

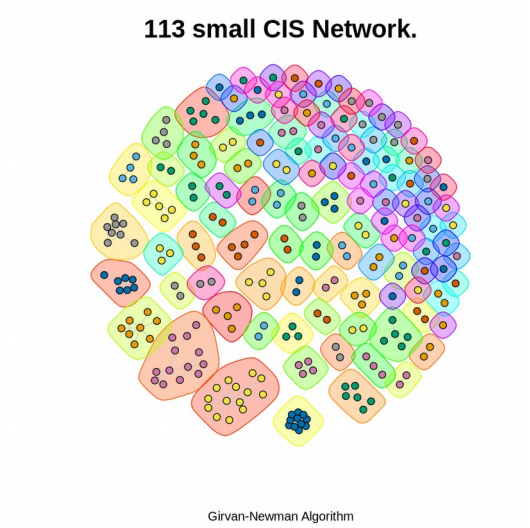
WOSFull	1	2	3	4	5	6	7	8	9	10	11	12	13	27	30	32	4852
	26	23	18	7	8	3	2	2	2	3	1	3	1	1	1	1	1

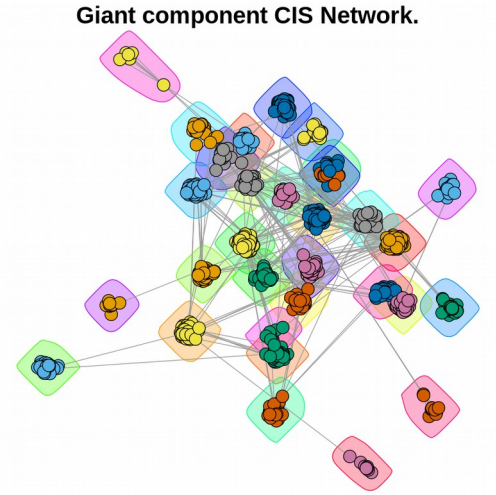
	average path length in the giant component	the longest paths(diameter) of giant component	vertex connectivity	edge connectivity	CUT vertices
CisFull	7	19	1	1	272
PrinFull	6	17	1	1	262
WOSFull	5	16	1	1	234

- In the case of the giant component of the network, the vertex and edge connectivity are both equal to one. Thus it requires the removal of only a single well-chosen vertex or edge in order to break this sub graph into additional components.
- In the giant component of the data, almost 10% of the vertices are cut vertices(262) out of all nodes.

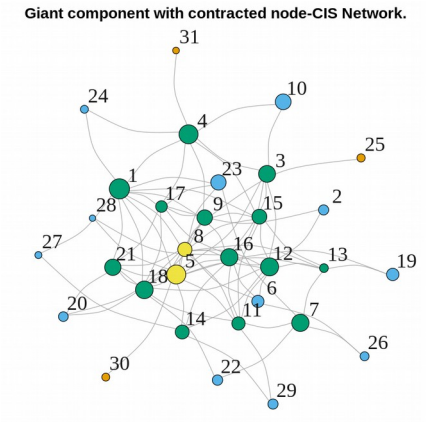
From all network data, I extracted all clusters by decomposing graph function. A quick glance at the clusters reveals that the graph is not composed of a single connected component. For example for PrinFull network data, 2696 of the nodes are in a single large component(giant component) and the remaining 143 are in 26 small components. The 2696 nodes in the big component overwhelm the smaller components and the 26 small components act as visual clutter for the big component. I separated the 26 small components. Work flow is the same for CisFull and WOSFull, for CisFull the components is 114 and for WOSFull is 103.
So for each network, there is a bunch of small component and a giant component. Then I run some algorithm like fast greedy community detection, eigenvector community, label propagation community, edge betweenness community and cluster louvain. In 3 network data set, the edge betweenness and cluster louvain show higher modularity than other algorithm. For the CisFull network and PrinFull I run edge betweenness, but for WOSFull , I run louvain, in my computer also in the google clobe, the edge betweenness seems to be quite slow for WOSFull and there is no result.

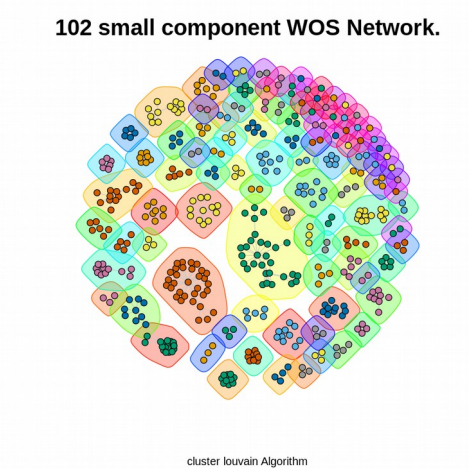
For the CisFull I run girvan newman algorithm,it clustered the giant component into 31 components with modularity 0.87.
For the PrinFull I implemented the girvan newman algorithm. For the giant components, the algorithm decomposes the giant component into 43 with modularity 0.89. For WOSFull, the giant component is divided to 23 with modularity 0.36. I wanted to run edge Betweenness, but I couldn't run it on WOSFull network.

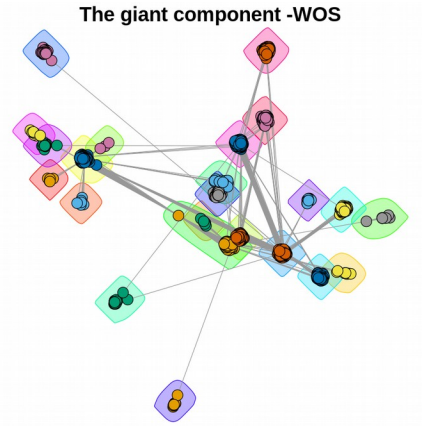


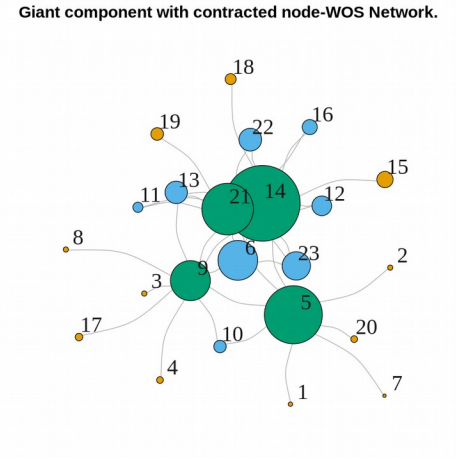


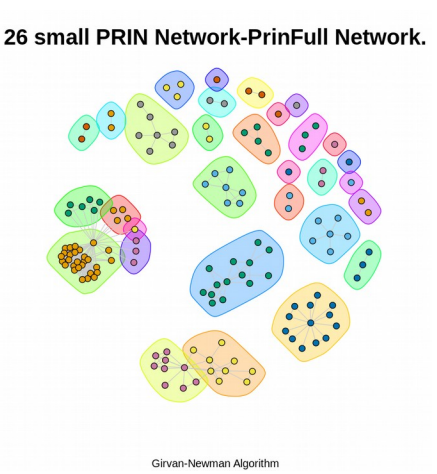
Contract the Communities
Plotting a single node for each community. Here, I make the area of each community vertex proportional to the number of members of that community. I colored the vertices using a coarse grouping based on their degrees.

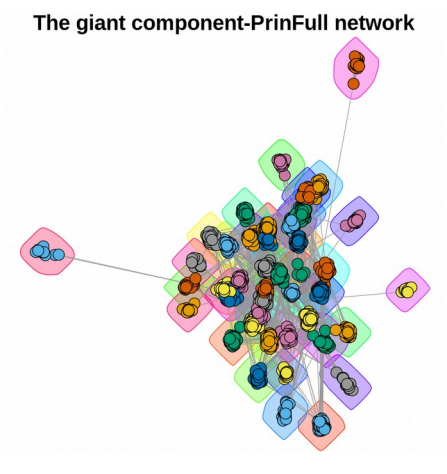


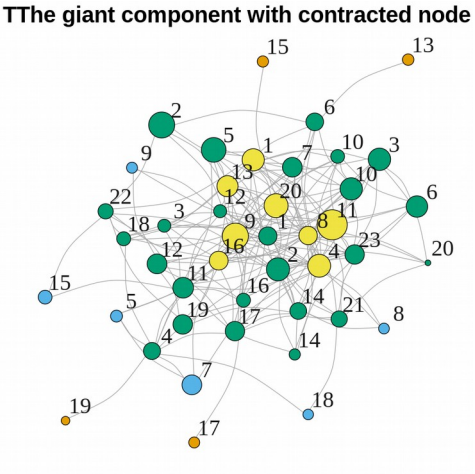








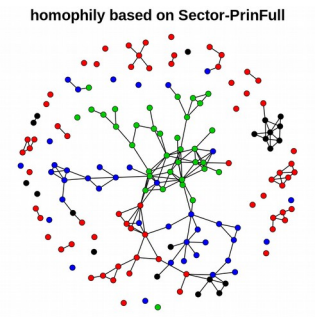


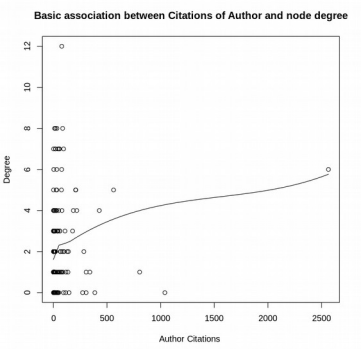


Exponential Random Graph Model :

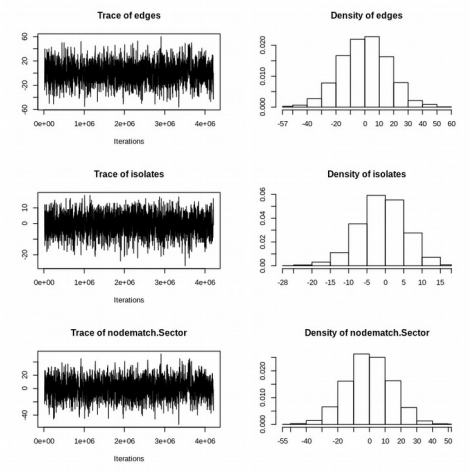
PrinFull Network:

I defined an induced sub graph for Italian researchers who work in sector 2,3,4,5. So the size of network seems good with 163 vertices and total edges is equal to 187. The transitivity of this sub-graph is 36% so I didn't fit the model with transitivity.
First I created a NULL model that includes only the edges term. For the NULL model I got a negative edge parameter(-4.242), since the network is rather sparse. And conditional probability of having a tie is equal to 0.01 that is equal to density.
Then I visualize the relation between degree and nodes attribute(citations). there is no particular association, so I didn't take citations as a nodal parameter. Also the left hand plot, shows homophily in sector.



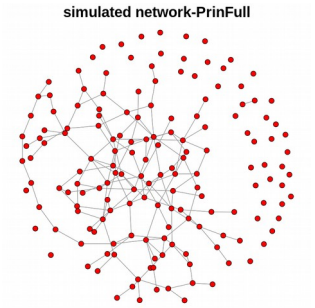


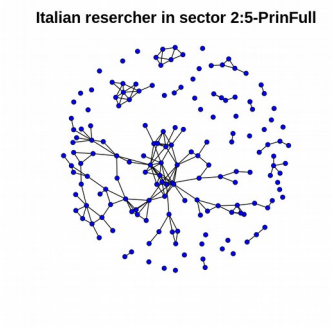
The final model is like below :
ergm(l2 ~ edges + isolates + nodematch('Sector'))
there is a good estimation around 60% for isolated node. Also for citation the estimate is positive with low error.
we are aiming for the trace to look like a hairy caterpillar. And there shouldn't be a burn in the plot.



SIMULATION

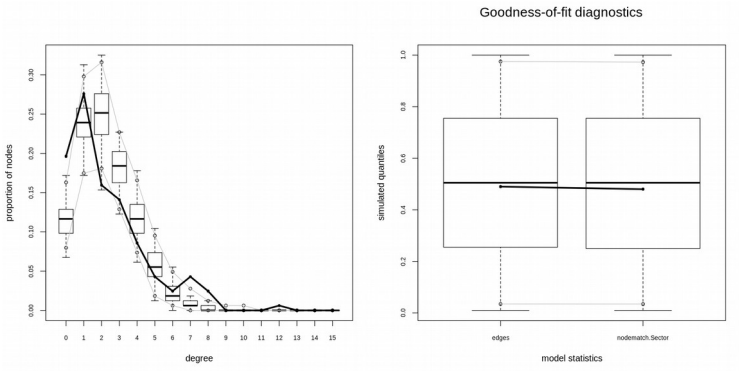
The result for simulation is as below:
Simulated network:
edges: -5.32
isolates: 0.6
node match Sector:2.32 with 0.18 error.



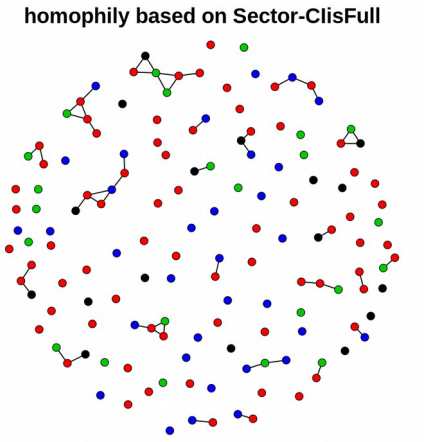


The density of observed and simulated is 0.01.

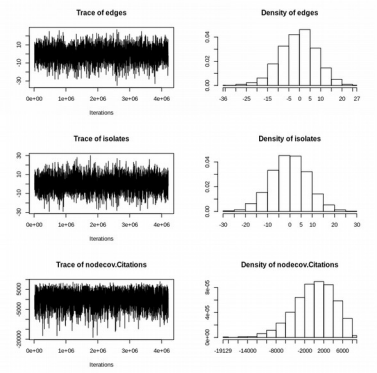
The goodness of model:



CisFull Network: The subset of data is the same as PrinFull, I created a NULL model that includes only the edges term, for the NULL model I got a negative edge parameter(-5.3) since the network is rather sparse. And conditional probability of having a tie is equal edges: 0.004 that is equal to density.
Then I visualize the relation between degree and node attribute (citations), there is no particular association, so at the beginning I didn't take citations as a nodal parameter, but when I added the citation to model, I get better result for density. Also the plot below, shows homophily based on sector, it shows there is no homophily, so when I fit model based on nodematch sector I get very bad result as :-0.65, and the result for nodecov citations is 0.0003 with very low error.

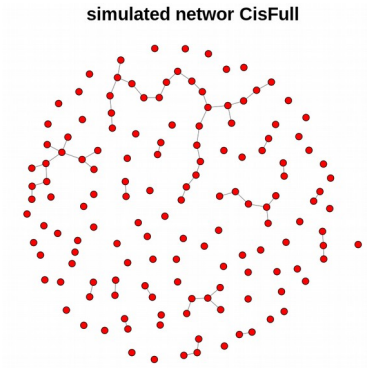


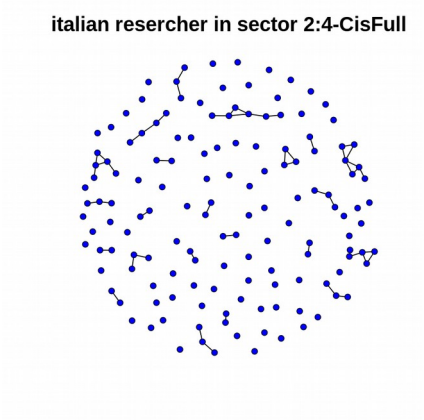
So the final model is as below:
ergm(l2 ~ edges + isolates+nodecov('Citations'))
There is a good estimation around 0.5 for isolated node. And also for citation the estimate is positive with low error.

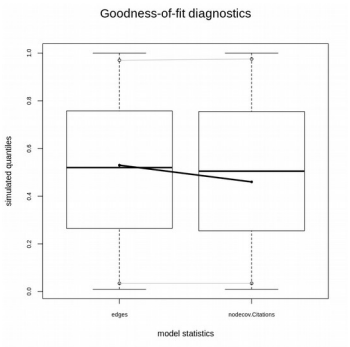


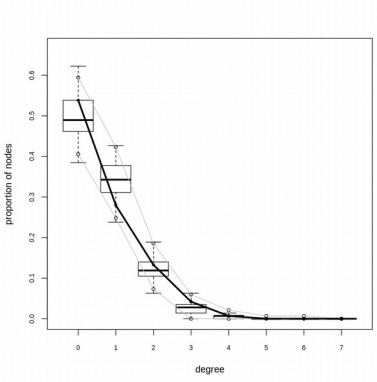
SIMULATION

The result for simulation is as below:
Simulated network:
Edges: -4.8
Isolates: 0.5
Node match sector:0.0003
The density of observed and simulated
is 0.004
goodness of model







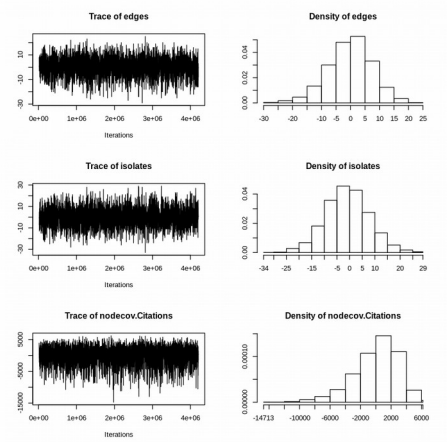


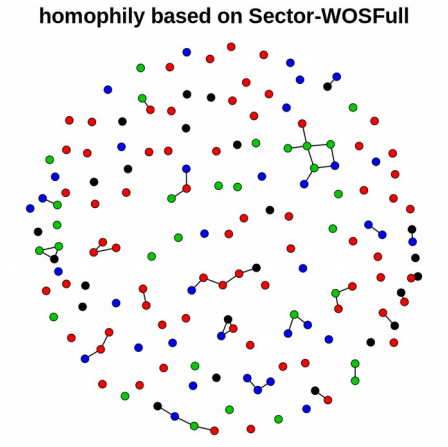
WOSFull Network:
The subset of data is the same as PrinFull, I created a NULL model that include only the edges term, for the NULL model I got a negative edge parameter(-5.6) since the network is rather sparse. And conditional probability of having a tie is equal to 0.003 that is equal to density.

I started with model ergm(l2 ~ edges + isolates + nodematch('Sector'))
Edges : --5.2108 error :0.3761
Isolates 0.4481 error :0.3155
Nodematch.Sector: 0.1549 error: 0.3252
As we cansee the error is high, because plot there is no homophily in the graph based on sector. I fit model as ergm(l2 ~ edges + isolates), I didn't

get a good result on density of simulated, then I add nodecov to model,so I got better result.

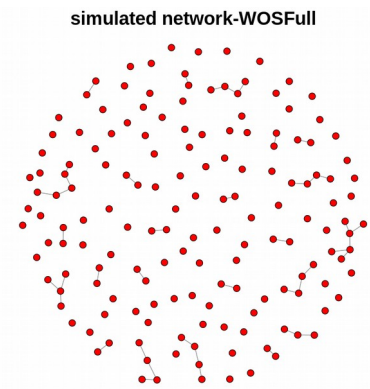
The final model:
ergm(l2 ~ edges + isolates+nodecov('Citations'))

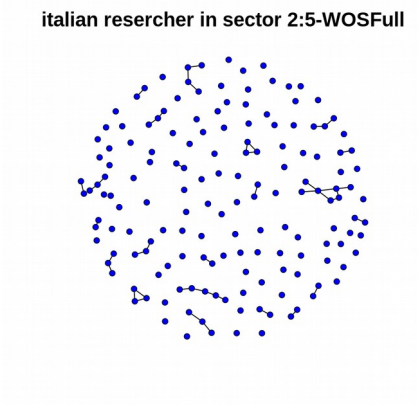




So the final model is as below:
ergm(l2 ~ edges + isolates+nodecov('Citations'))

SIMULATION
The result for simulation is as below:
Simulated network:
Edges: -5.1
Isolates: 0.44
Nodecov.Citations 0.0001
The density of observed and simulated is 0.003 the same as observed model.





goodness of model