Bioinformatics@Data Science A.Y. 2018-2019

Network Biology 2nd Part

Pochiraju Venkata Naga Sai Krishna Abhinay (1819771)

Rachuri Mani Niharika (1819748)

Group no. 7

Abstract

In this project, using the Seed Genes Interactome (SGI), the Intersection Interactome (I) and the Union Interactome (U) built in the first part of the Network Biology Project, we compute the main network measures (Global and Local) for SGI, U and I. Then we apply Markov and Louvain clustering algorithms to discover the Putative Disease Modules, and their corresponding overrepresented GO categories and Pathways. Using the DIAMOnD tool, we perform the same enrichment analysis.

1 Network Measures (Global and Local) for SGI, I and U

To calculate the global and local network measures for all the 3 interactomes, we first read the csv files that were generated in the first part of the Network Biology project. Then we use the network library in python and draw the network graphs for the three interactomes. The network graphs for SGI, I and U interactomes are shown in Figures 1,2 and 3 respectively. The results are shown in Table 3.

1.1 Global Measures

We calculate various Global measures using the following functions:

Measure	Function/Algorithm
No. of Nodes	G.nodes()
No. of Links	G.edges()
No. of Connected Components	nx.number_connected_components(G)
No. of Isolated nodes	nx.number_of_isolates(G)
Average Path Length	nx.average_shortest_path_length(G) – for all the component subgraphs and take the average
Average Degree	nx.average_degree_connectivity(G)/G.degree() for all the subgraphs and take the average
Average Clustering coefficient	nx.average_clustering(G)
Network Diameter	nx.diameter(G) for all the component subgraphs
Network Radius	nx.radius(G) for all the component subgraphs
Centralization	Freeman Centralization

Table 1. Functions/Algorithms used for the Global measures

1.2 Local Measures

First, we isolate the Largest Connected Component (LCC) of SGI, I and U and calculate the global and local measures for the corresponding LCC. The network graphs of LCC_I, LCC_U and LCC_SGI are shown in figures 4, 5 and 6 respectively.

1.2.1 Global measures of the LCC

The global measures are measured similarly as before using the same functions. The results are stored in the files: global_I.csv, global_U.csv and global_SGI.csv for Intersection, Union and Seed Gene Interactomes respectively. Some rows from the results are shown in Table 4.

1.2.2 Local measures of the LCC

The local measures are calculated using the following functions:

Measure	Function/Algorithm
Node Degree	nx.degree_centrality(G)
Betweenness Centrality	nx.betweenness_centrality(G)
EigenVector Centrality	nx.eigenvector_centrality_numpy(G)
Closeness Centrality	nx.closeness_centrality(G)
Ratio betweenness/Node degree	ratio of above calculated values

Table 2: Functions/Algorithms used for the Local measures

The results for the Intersection, Union and Seed Gene Interactomes are saved in the files: local_I.csv, local_U.csv and local_SGI.csv respectively. The top 20 Betweenness Centrality Genes are shown in Table 5.

2 Clustering Methods for Disease Modules discovery

In this part, we apply Markov Cluster Algorithm (MCL) and Louvain Clustering Algorithm to the above derived data of the Largest Connected Component (LCC) of Intersection and Union interactomes. For MCL, we use the library markov_clustering in python to get and plot the clusters. In the case of Louvain clustering, we used two methods: 1. Using the Louvain library to get the clusters, but in this case, networkx graph is not supported, so we convert the I and U graphs from networkx graph to igraph and the use the function louvain.find_partition(); 2. Using the community library in python and the function community.best_partition(G) which uses the louvain method described in Fast unfolding of communities in large networks, Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Renaud Lefebvre, Journal of Statistical Mechanics: Theory and Experiment 2008(10), P10008 (12pp). This function returns the partitions (community) of each node. Depending on the community number, we separate this into a list for each number.

Once we cluster the networks, we consider all the modules in which seed genes are statistically overrepresented ($p<0.05$). This is performed by applying a hypergeometric test which is available as a library(hypergeom) in python which can be imported from scipy.stats. Performing this task gives us the Putative Disease modules. We saved the results in two groups: one to classify Louvain and Markov, and the other group separates Union Interactome to Intersection Interactome with filenames: pdmLouvain.csv, pdmMarkov.csv, pdmU.csv and pdmI.csv respectively. The network graphs for Markov and Louvain are shown in Figures 7-10. The results are shown in Tables 6 and 7.

3 Enrichment Analysis on the disease modules

In this part, we are supposed to find Gene Ontology categories and Pathways for the putative disease modules (U and I) found in the previous part. So, using excel we get all the unique set of all the genes in each module (U and I) and use the online GUI of InnateDB to perform these tasks.

Note: we tried to perform these tasks programmatically using the KEGG module from bioservices. Specifically using the function get_pathway_by_gene(), but we could get only id and the pathway name. But using the GUI of InnateDB (http://www.innatedb.com/redirect.do?go=batchPw and http://www.innatedb.com/redirect.do?go=batchGo) we get more detailed results. So, we proceeded to use the online portal.

First, we save all the unique genes in both I and U networks into csv files (GeneListI.csv and GeneListU.csv) using python. Then using the file uniprot-list.xlsx from the 1st part of the homework and using basic excel functions index and match, we get the corresponding UniProt IDs of the genes and save them in the files: GeneListI.xls and GeneListU.xls for Intersection and Union interactomes respectively. Then we utilize the links: http://www.innatedb.com/redirect.do?go=batchPw and http://www.innatedb.com/redirect.do?go=batchGo to get and download the GO categories and pathways for each case. The results are stored in the files: PDM-I-GO.xls and PDM-I-Pathways.xls for the intersection interactome and PDM-U-GO.xls, and PDM-U-Pathways.xls for the union interactome respectively.

4 Putative disease proteins using DIAMOnD

The tool DIAMOnd which runs the DIAMOnD algorithm as described in *A DIseAse MOdule Detection (DIAMOnD) Algorithm derived from a systematic analysis of connectivity patterns of disease proteins in the Human Interactome. PlOS Comp Bio (in press), 2015. by Susan Dina Ghiassian, Joerg Menche & Albert-Laszlo Barabasi* is utilized to find the putative disease modules. We download the source-code of DIAMOnd from github and run the following command in the command line: (first we give each gene an numeric index)
'python .\DIAMOnD.py .\biogridPPI.txt .\seed_genes.txt 200 .\out.txt'
where, biogridPPI.txt is the reference input interactome from BioGRID used in the first part of the project. seed_genes.txt is the list of our seed genes. 200 is to limit the number of putative disease proteins (n). out.txt is the output from DIAMOnD.
The output list has 200 DIAMOnD nodes which are then saved in an excel file to find out the Gene names using basic excel functions using the list of genes and their corresponding numbers that we allocated. And then, again using excel functions we find the UniProt IDs of those genes and save them in an xls file (outDiamondUniprot.xls) and again use the links used above, to get the pathways and GO categories. The final GO analysis output is saved in the files: PDM-DIAMOnD-Pathways.xls and PDM-DIAMOnD-GO.xls. The top 40 Genes are shown in Table 8.

5 Figures and Tables

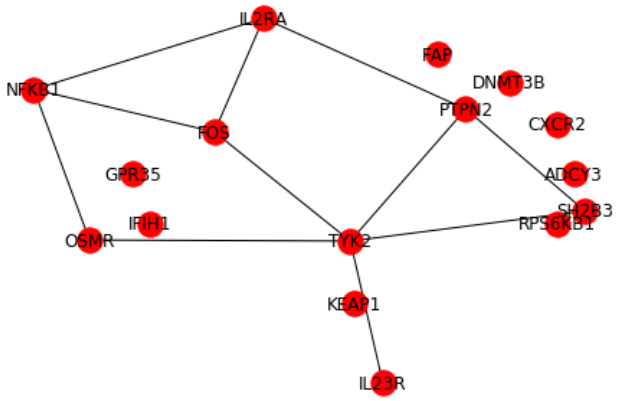

Figure 1: Seed Gene Interactome (SGI)

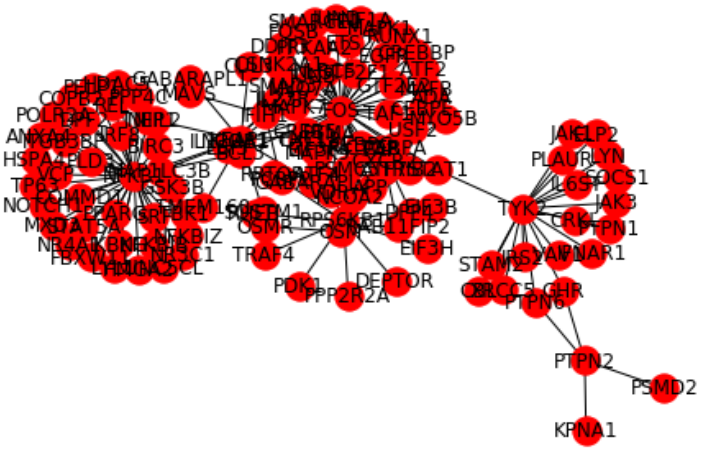

Figure 2: Intersection Interactome

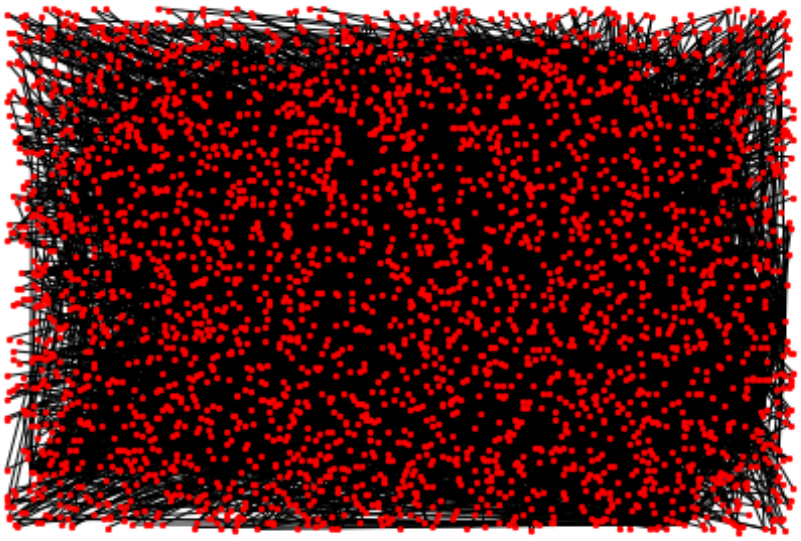

Figure 3: Union Interactome

Measure	SGI	I	U
Nodes	16	131	3223
Links	25	129	4901
Connected Components	9	9	2
Isolated Nodes	0	0	0
Avg Path Length	0.1904761904	1.6436033102	2.2744682122
Avg Degree	3.125	1.9694656488	3.0412659013
Avg Clustering Coeff	0.1520833333	0.0	0.0976450988
Diameter	3	2,2,8,2,1,1,2,1,2	10,1
Radius	2	1,1,4,1,1,1,1,1,1	5,1
Centralization	0.0183703703	0.0020887574	0.00010203818

Table 3: Global measures of SGI, I and U

Measure	SGI	I	U
Nodes	8	91	3221
Links	17	94	4900
Avg Path Length	1.714286	3.659096	3.548936
Avg Degree	{2: 6.0, 7: 3.4285714285714284, 5: 3.8, 4: 3.5, 3: 4.333333333333333}	{1: 32.719512195195, 2: 24.7, 18: 1.166666666667, 37: 2.01351351351, 4: 1.5}	{1: 483.88689655172413, 2: 465.1136363636364, 4: 440.7025316455696, 3: 500.1490514905149 ….. }
Avg Degree Val	4.25	2.065934	3.042533
Avg Clustering Coeff	0.304167	0	0.097706
Network Diameter	3	8	10
Network Radius	2	4	5
Centralization	0.06414	0.004361	0.000102

Table 4: Global measures of LCC-SGI, LCC-I and LCC-U

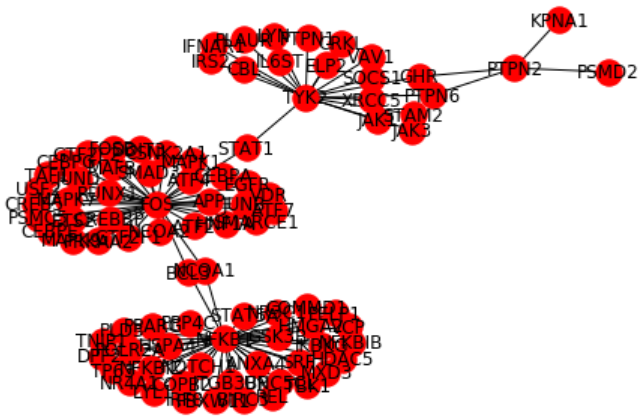

Figure 4: Network Graph of LCC_I

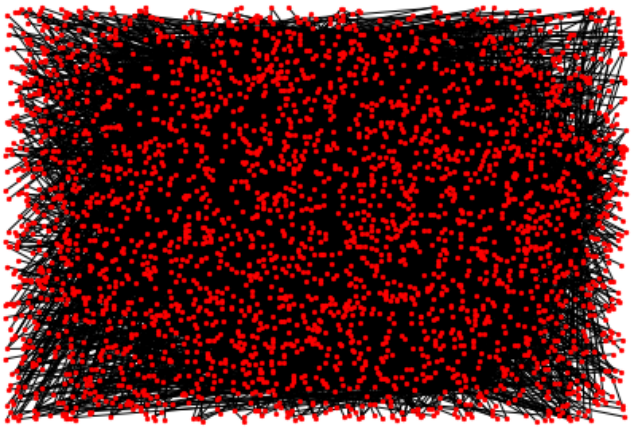

Figure 5: Network Graph of LCC_U

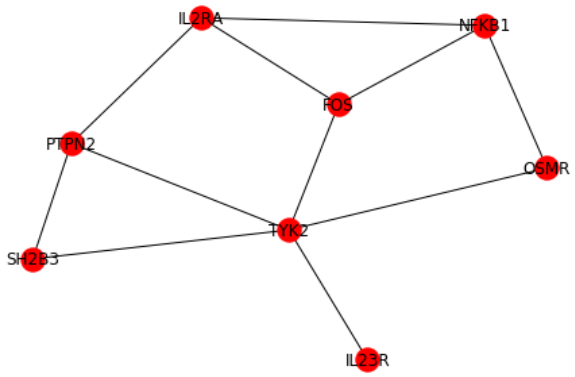

Figure 6: Network Graph of LCC_SGI

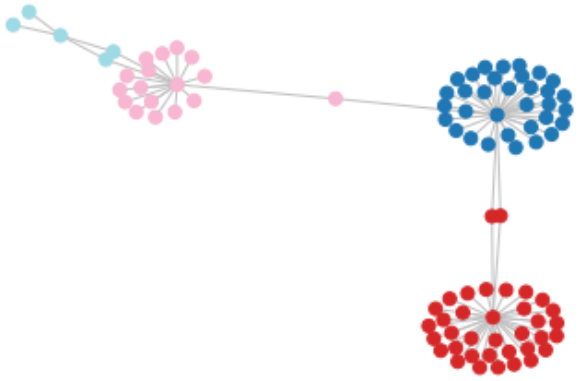

Figure 7: Markov Clustering – Intersection Interactome

Figure 8: Markov Clustering – Union Interactome

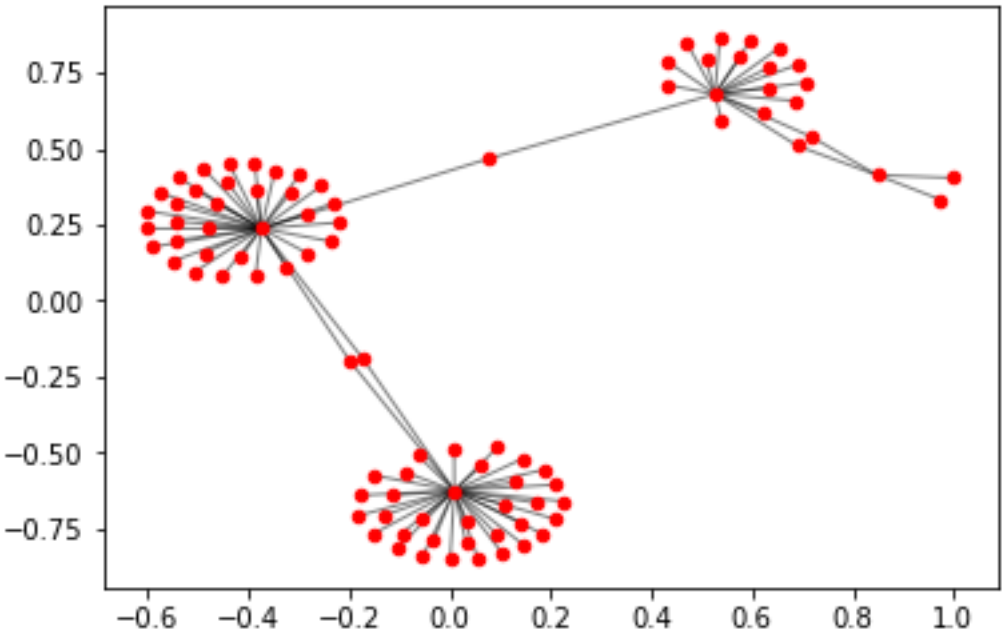

Figure 9: Louvain Clustering – Intersection Interactome

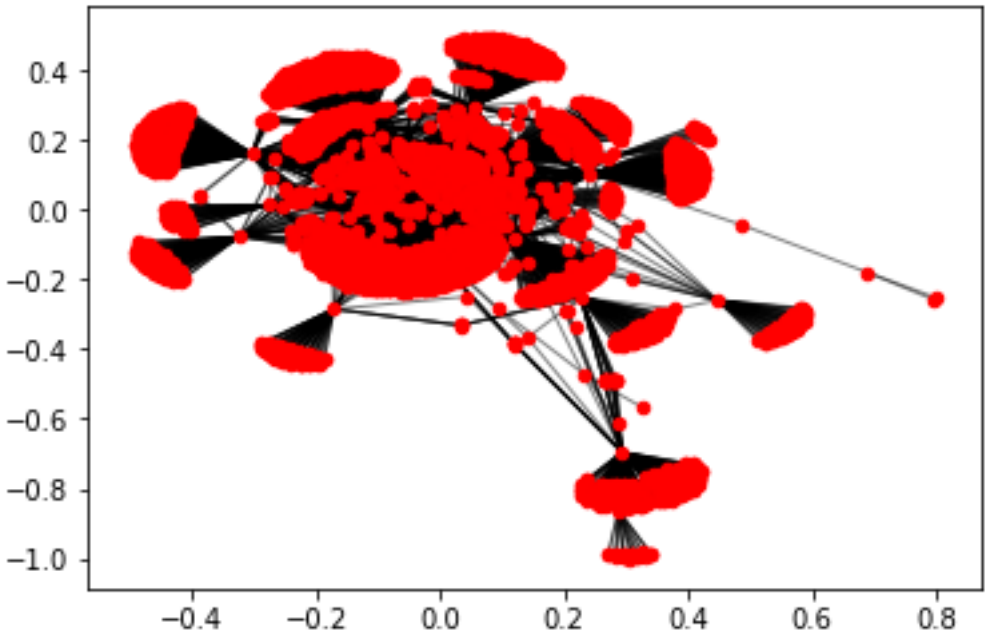

Figure 10: Louvain Clustering – Union Interactome

Number	Intersection Interactome	Union Interactome
1	FOS	RPS6KB1
2	NFKB1	GPR35
3	TYK2	CXCR2
4	STAT1	RAVER1
5	BCL3	ADCY3
6	NCOA1	FAP
7	PTPN2	IL2RA
8	GHR	YES1
9	PTPN6	CAV1
10	APP	OR5B21
11	PPP4C	GNAO1
12	VDR	CXCR4
13	JUNB	IL23R
14	CBL	ATP1A1
15	SRF	ATP5B
16	CEBPA	SPART
17	GSK3B	STAT3
18	HDAC5	TNFRSF6B
19	IL6ST	FYN
20	IKBKG	SOCS3

Table 5: Top 20 Betweenness Centrality Genes

SeedGeneList	SeedGene	p-value	Algorithm Used	module ID	Num of Seed Genes	Num of Genes in Module
['NFKB1']	NFKB1	0.004263	Markov	2	1	5
['TYK2']	TYK2	0.004847	Markov	3	1	4
['FOS']	FOS	0	Louvain	1	1	3
['NFKB1']	NFKB1	0.004263	Louvain	2	1	5
['TYK2']	TYK2	0.008529	Louvain	3	1	4

Table 6: Putative Disease modules – Intersection interactome

SeedGeneList	SeedGene	p-value	Algorithm Used	module ID	Num of Seed Genes	Num of Genes in Module
['IL2RA']	IL2RA	0.000479	Markov	22	1	5
['RNF186']	RNF186	0.000872	Markov	25	1	6
['OR5B21']	OR5B21	0.00406	Markov	5	1	6
['FAP']	FAP	0.000255	Louvain	6	1	3
['OR5B21']	OR5B21	0.005284	Louvain	4	1	6
['GPR35']	GPR35	0.003558	Louvain	4	1	5
['RNF186']	RNF186	0.005284	Louvain	4	1	6

Table 7: Putative Disease modules – Union Interactome

#rank	DIAMOnD_node	Gene	UniProt
1	809	STAT3	P40763
2	32	RELA	Q04206
3	924	JAK2	O60674
4	810	SOCS3	O14543
5	428	STAT1	P42224
6	802	STAT5A	P42229
7	104	EGFR	P00533
8	308	TP53	P04637
9	1248	HIF1A	Q16665
10	1256	FYN	P06241
11	1325	ABL1	P00519
12	1687	LCK	P06239
13	160	BRCA1	P38398
14	1762	STAT5B	P51692
15	474	SOCS1	O15524
16	1188	PIK3R1	P27986
17	87	CAV1	Q03135
18	1677	IL2RB	P14784
19	632	JUN	P05412
20	881	SYK	P43405
21	852	TRAF2	Q12933
22	1449	SRC	P12931
23	1108	AR	P10275
24	203	CHUK	O15111
25	792	STAT6	P42226
26	141	IRF1	P10914
27	2506	INSR	P06213
28	179	HSP90AA1	P07900
29	247	JAK1	P23458
30	1247	CCL2	P13500
31	1998	TRAF6	Q9Y4K3
32	96	APP	P05067
33	318	ELAVL1	Q15717
34	2250	CSF3R	Q99062
35	1138	RASA1	P20936
36	995	CD86	P42081
37	2121	PTPN11	Q06124
38	2343	PRKCA	P17252
39	446	TGFBR2	P37173
40	2132	PTK2B	Q14289

Table 8: Top 40 Genes from DIAMOnD

References

GuyAllard. (2018, December 11). GuyAllard/markov_clustering. Retrieved from https://github.com/guyallard/markov_clustering

Lynn, D. (n.d.). Gene Ontology Analysis - Upload Data. Retrieved from https://www.innatedb.com/redirect.do?go=batchGo

Lynn, D. (n.d.). Pathway Analysis - Upload Data. Retrieved from https://www.innatedb.com/redirect.do?go=batchPw

ChristianChristian 308216, & GijsGijs 1. (n.d.). How to use hyper-geometric test. Retrieved from https://stats.stackexchange.com/a/300314

Ahollocou. (2018, April 17). Ahollocou/cylouvain. Retrieved from https://github.com/ahollocou/cylouvain

Scipy.stats.hypergeom¶. (n.d.). Retrieved from https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.stats.hypergeom.html

Barabasilab. (2015, January 15). Barabasilab/DIAMOnD. Retrieved from https://github.com/barabasilab/DIAMOnD