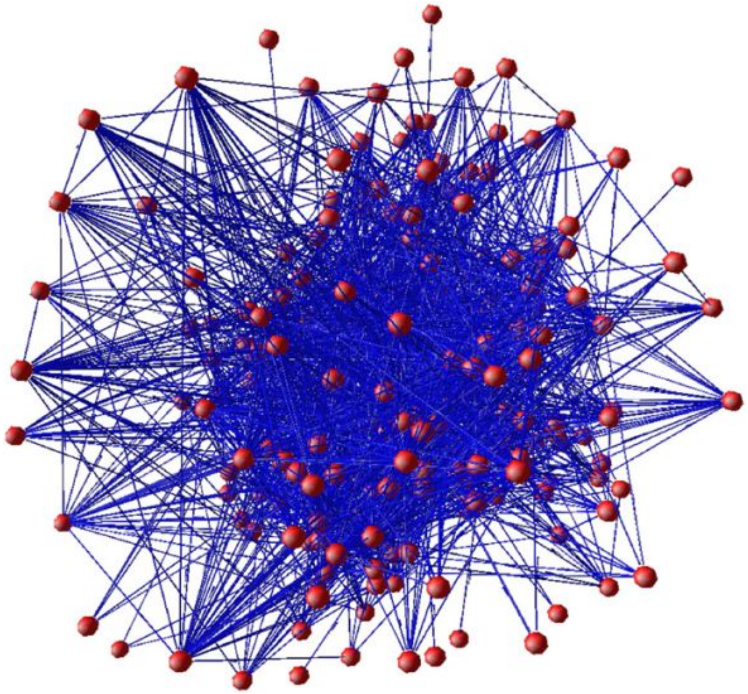
Lab on Graph Theoretical Analysis

CS – 590.21 Analysis and Modeling of Brain Networks

Department of Computer Science
University of Crete



Brief Summary

- **Graph**: an ordered pair *G* = (*V*, *E*) comprising set *V* of **vertices** (or nodes) & a set *E* of **edges** (or links).

- **Degree of Connectivity**: the number of links a node has to other nodes.

- **Clustering Coefficient:** the extent to which the neighbors of a given node link to each other.

- **Shortest Path Length**: the fewest number of links(/steps) between two nodes.

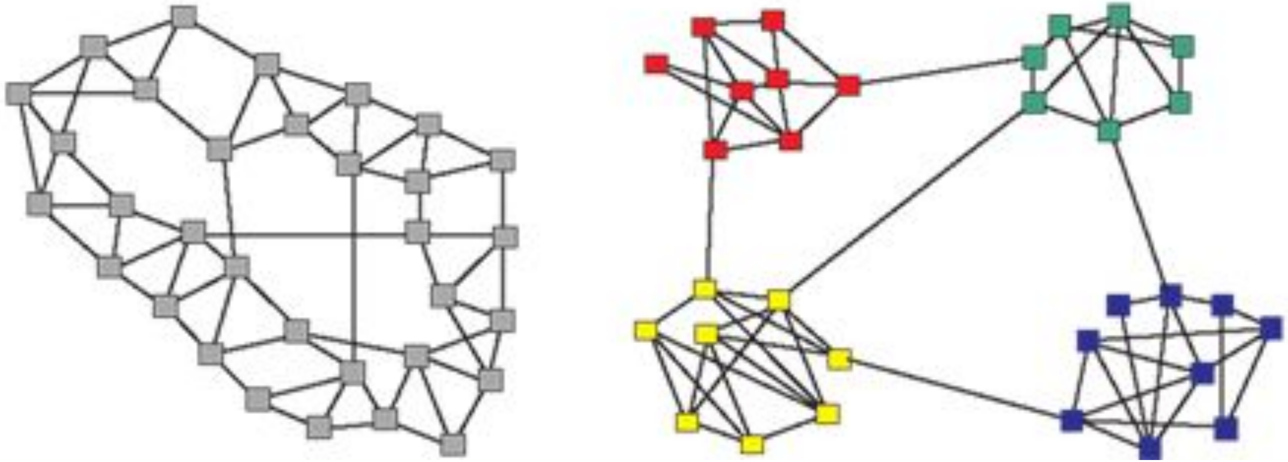
Adapted from: http://networksciencebook.com/

Small-Worldness

High Average Clustering Coefficient

AND

Small Average Shortest Path Length



Adapted from: https://doi.org/10.3389/fnins.2010.00200

Lattice and Random Graph Construction (in this assignment)

1. **Full Randomization or Latticization**
   a. **Lattice:** Watts-Strogatz
   b. **Random:**
      i. Erdos-Renyi G(n,p) model
      ii. Erdos-Renyi G(n,M) model

2. **Degree Distribution Preserving Randomization or Latticization**
   a. **Sporns Real-Based (both random and lattice)**

Small-World Indices (for this assignment)

1. σ (sigma)

$$\sigma = \frac{C}{L} \times \frac{L_r}{C_r}$$

- Where C and L are the Clustering Coefficient (CC) and Average Shortest Path Length (ASPL or L) of the observed network, respectively, and *Cr* and *Lr* are the Clustering Coefficient and Average Shortest Path Length from the equivalent random graph.
- For **σ>1** the observed network qualifies as **small-world**.

2. ω (omega)

$$\omega = \frac{L_r}{L} - \frac{C}{C_l}$$

- Where L is the the ASPL of the observed network, Lr is the ASPL of the random network, C is the CC of the observed network and Cl is the CC of the lattice.
- $-1<\omega<1$, with values around 0 (**ω≈0**) considered **small-world**.

Variables - edges

edges

10148x17 table

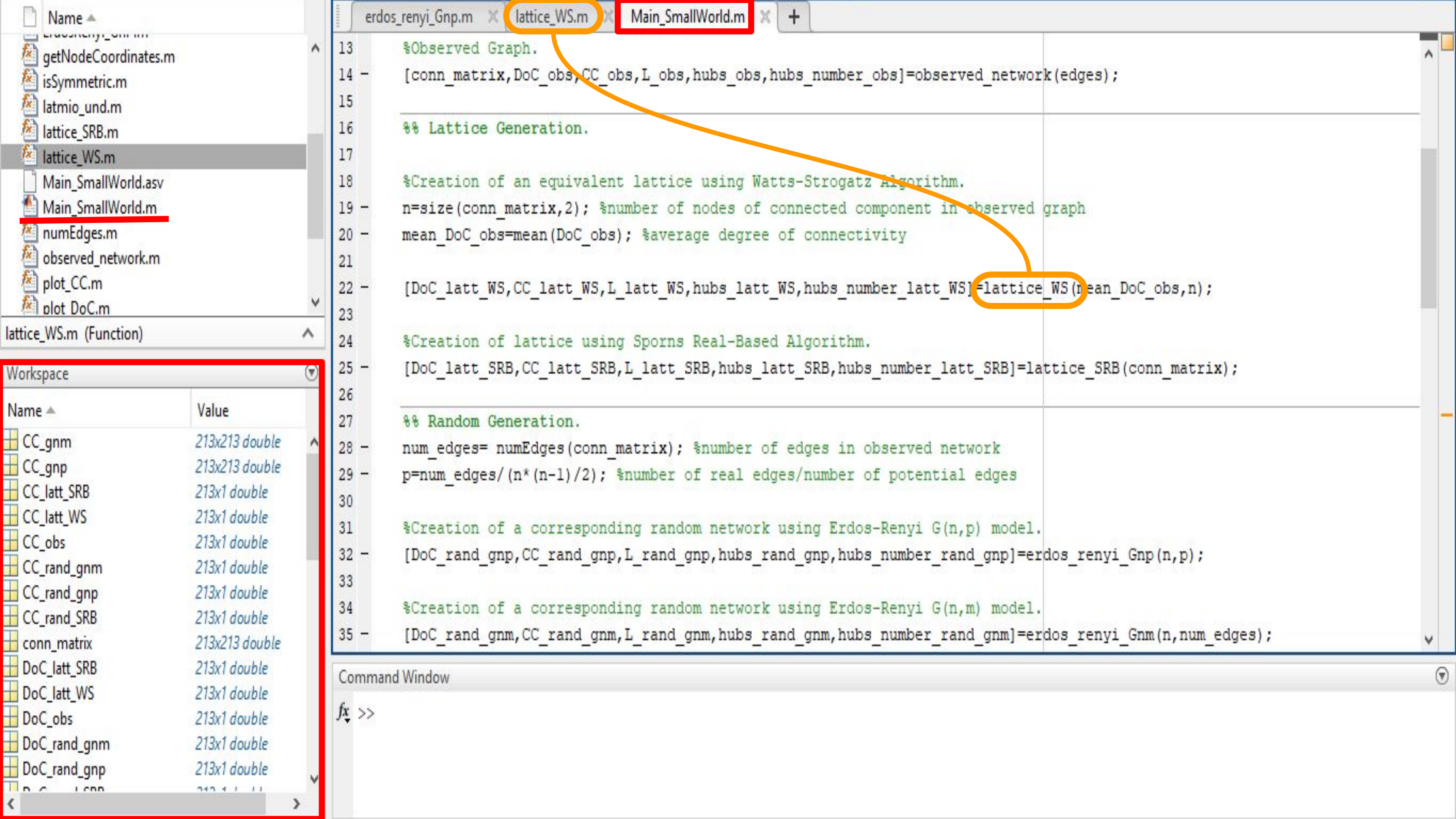
	1 NeuronA	2 NeuronB	3 STTC	4 CtrlGrpMean	5 CtrlGrpStDev	6 CtrlGrpMedian	7 Percentile	8 zScore
1	1	10	'0.0932535'	'0.000313198'	'0.0221142'	'-0.000210348'	1	'4.20274312...
2	1	17	'0.103349'	'0.00024101'	'0.0227866'	'0.000827137'	1	'4.52493965...
3	1	34	'0.0936288'	'-0.000602214'	'0.0218013'	'0.000277358'	1	'4.32226582...
4	1	68	'0.0949678'	'0.00148126'	'0.0223358'	'0.00152172'	1	'4.18550219...
5	1	207	'0.135214'	'-0.000788736'	'0.0220065'	'-0.000960173'	1	'6.18011660...
6	2	3	'0.0970788'	'0.000715448'	'0.0212133'	'0.000273276'	1	'4.54259129...
7	2	8	'0.123532'	'0.000352203'	'0.0208391'	'1.98821e-05'	1	'5.91099409...
8	2	16	'0.109333'	'0.000285499'	'0.0222037'	'-0.00109478'	998	'4.91123105...
9	2	17	'0.169904'	'-0.0015808'	'0.0213361'	'-0.00197385'	1	'8.03730766...
10	2	21	'0.118788'	'0.000371422'	'0.0218273'	'-0.00152466'	1	'5.42515922...
11	2	22	'0.0898491'	'0.000287487'	'0.0212201'	'-0.00024564'	1	'4.22060277...
12	2	26	'0.0997537'	'-0.000150144'	'0.0210437'	'-0.00077601'	1	'4.74744669...
13	2	29	'0.0839295'	'-0.00135977'	'0.0203219'	'-0.00248244'	1	'4.19691416...
14	2	34	'0.192007'	'-0.000504009'	'0.0262263'	'-0.00123013'	998	'7.34038003...
15	2	39	'0.121467'	'-0.000296711'	'0.0218462'	'-0.00211532'	1	'5.57367922...
16	2	60	'0.104538'	'0.00115485'	'0.0231726'	'-0.00138431'	996	'4.46143937...
17	2	61	'0.154889'	'0.00297409'	'0.022765'	'0.00276085'	1	'6.67317856...
18	2	62	'0.11552'	'0.000437361'	'0.0215361'	'0.00107283'	1	'5.34370842...
19	2	69	'0.0882548'	'-0.000788404'	'0.0204421'	'-0.00119314'	1	'4.35587361...
20	2	74	'0.0943669'	'0.00088515'	'0.0223695'	'0.0013643'	998	'4.17898254...
21	2	76	'0.110878'	'0.00058802'	'0.0211628'	'0.00236625'	1	'5.21150225...
22	2	77	'0.118562'	'-0.00122352'	'0.0222184'	'-0.00239881'	1	'5.39127569...
23	2	80	'0.0935168'	'0.000424138'	'0.0209075'	'4.48356e-05'	1	'4.45259653...

Variables - conn_matrix

conn_matrix

213x213 double

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	
2	0	0	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	
3	0	1	0	1	1	0	1	1	0	0	1	1	1	1	0	1	1	
4	0	1	1	0	1	0	1	0	0	0	0	1	0	1	0	1	1	
5	0	1	1	1	0	0	1	0	0	0	1	0	1	1	0	1	1	
6	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
7	0	0	1	1	1	0	0	1	0	0	1	1	0	1	0	1	1	
8	0	1	1	0	0	0	1	0	0	0	0	0	0	1	0	1	1	
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
10	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1	1	0	
12	0	1	1	1	0	0	1	0	0	0	1	0	0	0	0	1	1	
13	0	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	
14	1	0	1	1	1	0	1	1	0	0	1	0	0	0	1	1	0	
15	0	1	0	0	0	0	0	0	1	0	1	0	0	1	0	1	1	
16	0	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	1	
17	1	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1	0	
18	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	1	1	
19	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
20	0	1	1	1	1	0	1	1	0	0	0	0	0	0	0	1	0	
21	0	1	1	1	1	0	1	1	0	0	1	1	0	1	1	1	1	
22	0	1	1	1	1	0	1	0	0	0	1	0	0	1	1	1	1	
23	0	0	1	1	1	0	1	1	0	0	0	0	0	0	0	1	0	


Name
getNodeCoordinates.m
isSymmetric.m
latmio_und.m
lattice_SRB.m
lattice_WS.m
Main_SmallWorld.asv
Main_SmallWorld.m
numEdges.m
observed_network.m
plot_CC.m
plot_DoC.m
lattice_WS.m (Function)
erdos_renyi_Gnp.m
lattice_WS.m
Main_SmallWorld.m
%Observed Graph.
[conn_matrix,DoC_obs,CC_obs,L_obs,hubs_obs,hubs_number_obs]=observed_network(edges);
%% Lattice Generation.
%Creation of an equivalent lattice using Watts-Strogatz Algorithm.
n=size(conn_matrix,2); %number of nodes of connected component in observed graph
mean_DoC_obs=mean(DoC_obs); %average degree of connectivity
[DoC_latt_WS,CC_latt_WS,L_latt_WS,hubs_latt_WS,hubs_number_latt_WS]=lattice_WS(mean_DoC_obs,n);
%Creation of lattice using Sporns Real-Based Algorithm.
[DoC_latt_SRB,CC_latt_SRB,L_latt_SRB,hubs_latt_SRB,hubs_number_latt_SRB]=lattice_SRB(conn_matrix);
%% Random Generation.
num_edges= numEdges(conn_matrix); %number of edges in observed network
p=num_edges/(n*(n-1)/2); %number of real edges/number of potential edges
%Creation of a corresponding random network using Erdos-Renyi G(n,p) model.
[DoC_rand_gnp,CC_rand_gnp,L_rand_gnp,hubs_rand_gnp,hubs_number_rand_gnp]=erdos_renyi_Gnp(n,p);
%Creation of a corresponding random network using Erdos-Renyi G(n,m) model.
[DoC_rand_gnm,CC_rand_gnm,L_rand_gnm,hubs_rand_gnm,hubs_number_rand_gnm]=erdos_renyi_Gnm(n,num_edges);
Workspace
Name
Value
CC_gnm 213x213 double
CC_gnp 213x213 double
CC_latt_SRB 213x1 double
CC_latt_WS 213x1 double
CC_obs 213x1 double
CC_rand_gnm 213x1 double
CC_rand_gnp 213x1 double
CC_rand_SRB 213x1 double
conn_matrix 213x213 double
DoC_latt_SRB 213x1 double
DoC_latt_WS 213x1 double
DoC_obs 213x1 double
DoC_rand_gnm 213x1 double
DoC_rand_gnp 213x1 double
Command Window
>>

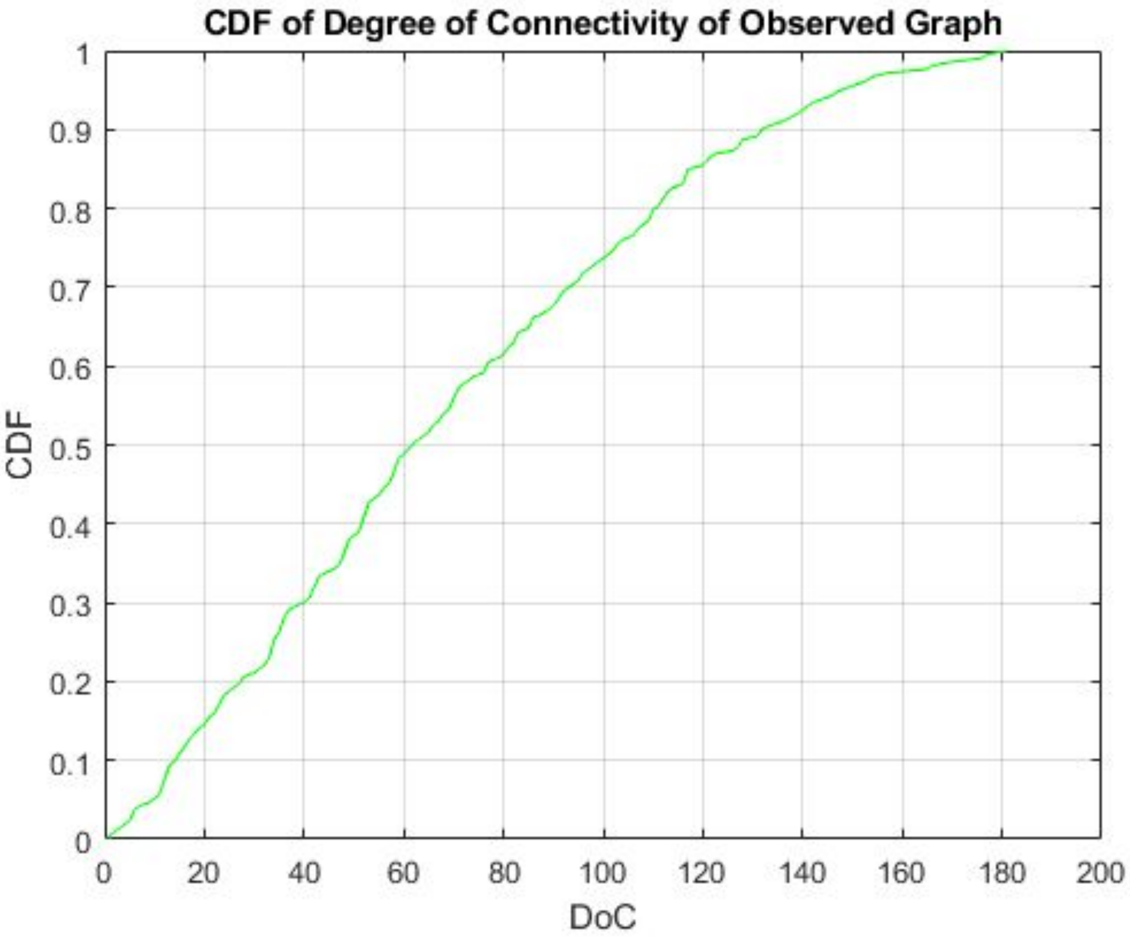
erdos_renyi_Gnp.m | lattice_WS.m | Main_SmallWorld.m | observed_network.m

```
%Produce a random graph according to Erdos-Renyi G(n,p) model, ie given the
%number of nodes and a probability of connecting two nodes, and calculate its DoC, CC, L and hubs.
%Input:
%n: number of nodes of observed graph
%p: probability of connecting two nodes, calculated as the fraction of the number of edges
%of observed graph over the number of its total potential edges
%Output:
%DoC_rand_gnp: Degree of Connectivity of generated graph
%CC_rand_gnp: Clustering coefficient of generated graph
%L_rand_gnp: Average Shortest Path Length of generated graph
%hubs_rand_gnp: DoC of hubs of generated graph
%hubs_number_rand_gnp: Number of hubs of generated graph

function [DoC_rand_gnp,CC_rand_gnp,L_rand_gnp,hubs_rand_gnp,hubs_number_rand_gnp]=erdos_renyi_Gnp(n,p)


seed=0; %you can choose another number randomly for a different random graph
[G,n,m] = ErdosRenyi_GnP(n,p,seed,1);
G=full(G);

%% Graph-Theoretical Analysis.
%Find number of Connected Components.
comp_rand=connected_components_v1(G,size(G,1));
```


CDF of Degree of Connectivity of Observed Graph
CDF
DoC
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
20
40
60
80
100
120
140
160
180
200

Assignment

Dataset: FS13 mouse

- OGB data
- Spontaneous Activity
- dt=0.3sec
- z-score threshold=4
- Undirected graph

Deadline: 05/11/2019