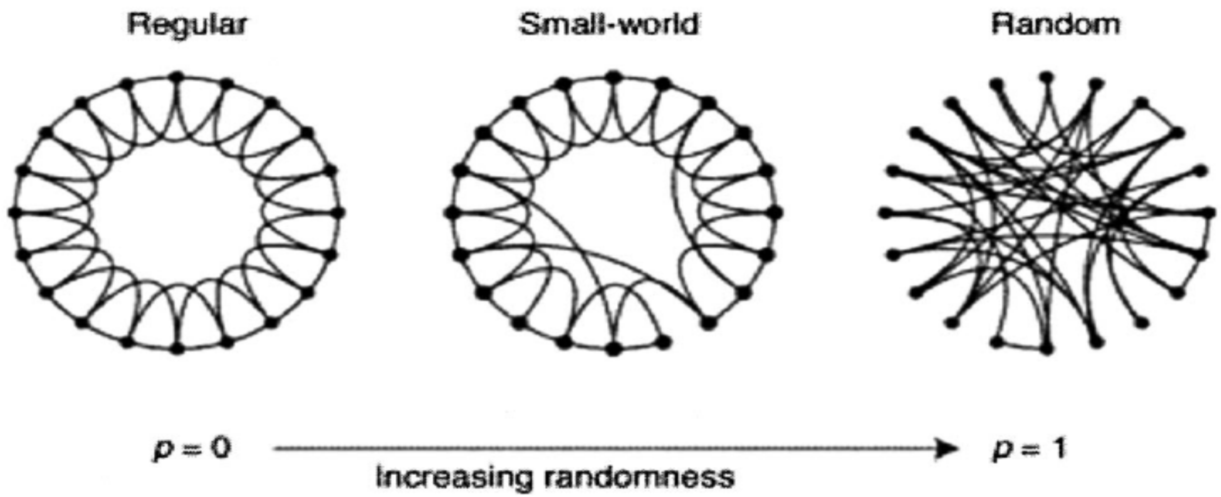
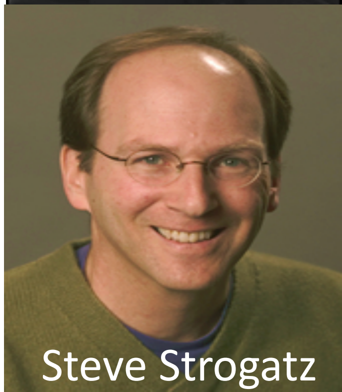
Exercise with School Network

Collective dynamics of ‘small-world’ networks

Duncan J. Watts* & Steven H. Strogatz



Duncan Watts



Steve Strogatz

Small world phenomenon:
applicable to other kinds of networks

Same pattern:

high clustering

$$C_{\text{network}} >> C_{\text{random graph}}$$

low average shortest path

$$l_{\text{network}} \approx \ln(N)$$

Collective dynamics of 'small-world' networks

Duncan J. Watts* **& Steven H. Strogatz**

Department of Theoretical and Applied Mechanics, Kimball Hall, Cornell University, Ithaca, New York 14853, USA

Table 1 Empirical examples of small-world networks

	L_{actual}	L_{random}	C_{actual}	C_{random}
Film actors	3.65	2.99	0.79	0.00027
Power grid	18.7	12.4	0.080	0.005
C. elegans	2.65	2.25	0.28	0.05

Comparison with "random graph" used to determine whether real-world network is "small world"

Network	size	av. shortest path	Shortest path in fitted random graph	Clustering (averaged over vertices)	Clustering in random graph
Film actors	225,226	3.65	2.99	0.79	0.00027
MEDLINE co-authorship	1,520,251	4.6	4.91	0.56	1.8×10^{-4}
E.Coli substrate graph	282	2.9	3.04	0.32	0.026
C.Elegans	282	2.65	2.25	0.28	0.05

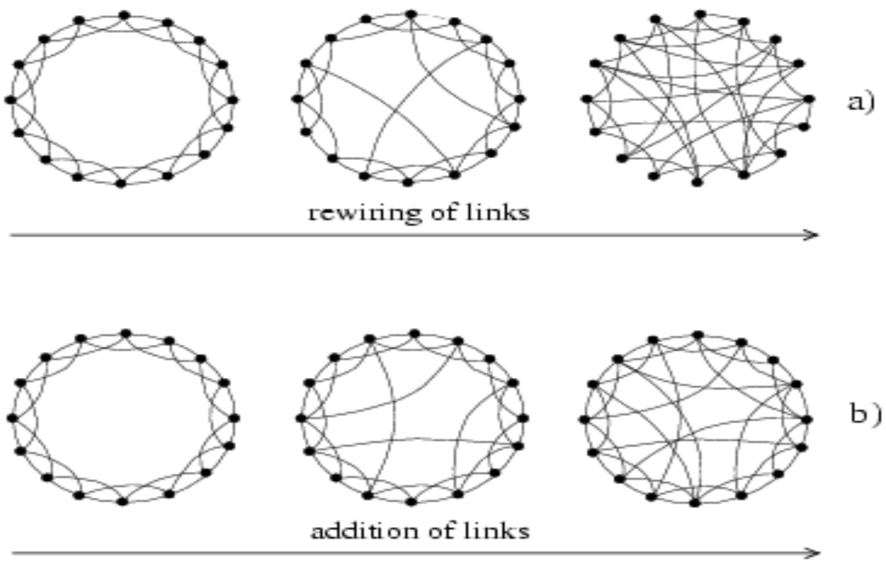
Watts-Strogatz model:

Generating small world graphs

- Each node has K>=4 nearest neighbors (local)
- tunable: vary the probability *p* of rewiring any given edge
- small *p*: regular lattice
- large *p*: classical random graph

Watts-Strogatz model:

Generating small world graphs



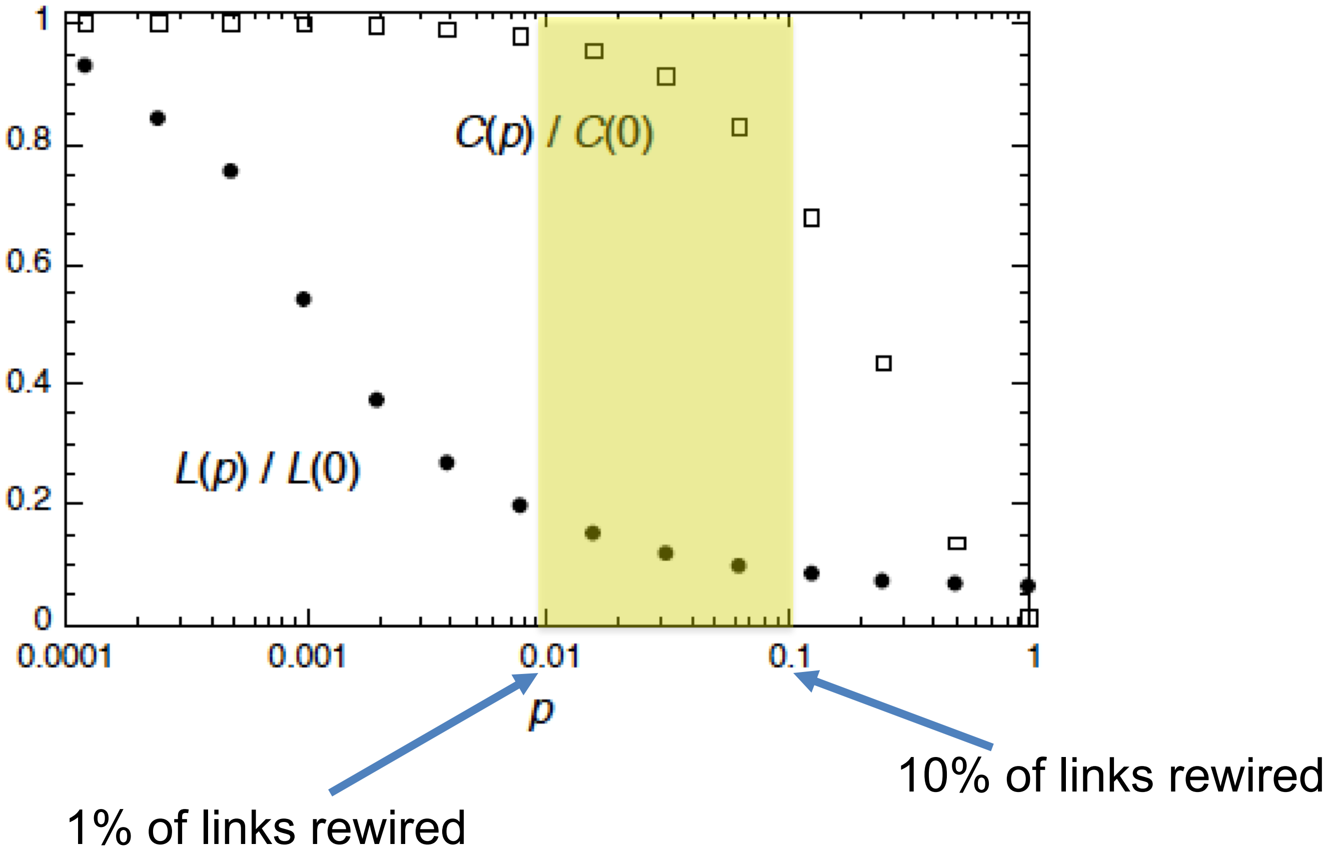
Select a fraction p of edges
Reposition on of their endpoints

Add a fraction p of additional edges leaving underlying lattice intact

- As in many network generating algorithms
  - Disallow self-edges
  - Disallow multiple edges

Source: Watts, D.J., Strogatz, S.H.(1998) Collective dynamics of 'small-world' networks. Nature 393:440-442.

Watts/Strogatz model:
Change in clustering coefficient and average path length as a function of the proportion of rewired edges



Between 1% and 10% rewired the shortest path decreases and the C stays high!!

Source: Watts, D.J., Strogatz, S.H.(1998) Collective dynamics of 'small-world' networks. Nature 393:440-442.

Definition of Clustering Coefficient

`clustering`(*G, nodes=None, weight=None*) [source]

Compute the clustering coefficient for nodes.

For unweighted graphs, the clustering of a node u is the fraction of possible triangles through that node that exist,

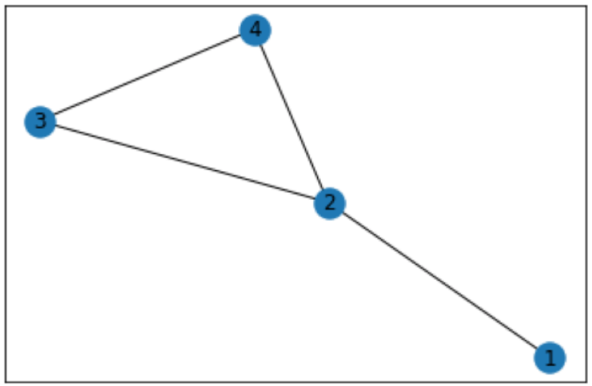
$$c_u = \frac{2T(u)}{deg(u)(deg(u) - 1)},$$

where $T(u)$ is the number of triangles through node u and $deg(u)$ is the degree of u.

Note there are extensions of the definitions for Weighted and Directed Graphs

Next class we study the calculations of Paths and clustering coefficient in ipynb and cover Routing in Networks

Example 1 Average Clustering Coefficient




```
nx.clustering(g)
```

{'1': 0, '2': 0.3333333333333333, '3': 1.0, '4': 1.0}

```
nx.average_clustering(g)
```

0.5833333333333333

```
(0+0.333333+1+1)/4
```

0.58333325

T(1) = 0   2/(1)(0)

T(2)=1   2/(3)(2)

T(3)=1   2/(2)(1)

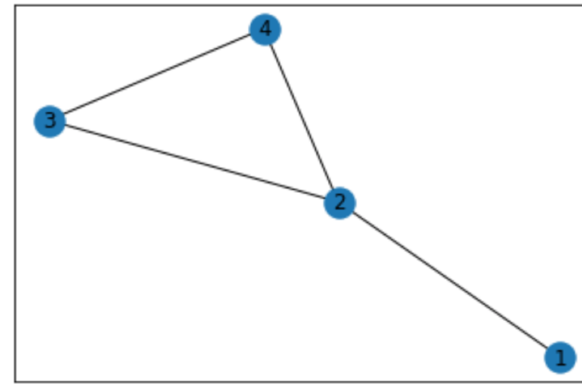
T(4)=1   2/(2)(1)

$$\frac{2T(u)}{deg(u)(deg(u)-1)}$$

<C>=0.5833

Example 1, Average Shortest Path

```
1 has  3 paths
['1', '2'] length  1
['1', '2', '3'] length  2
['1', '2', '4'] length  2
2 has  3 paths
['2', '1'] length  1
['2', '3'] length  1
['2', '4'] length  1
3 has  3 paths
['3', '2'] length  1
['3', '4'] length  1
['3', '2', '1'] length  2
4 has  3 paths
['4', '2'] length  1
['4', '3'] length  1
['4', '2', '1'] length  2
```



<L>=16/12

Number of Paths in Undirected Graph:
nodes(nodes-2)

In ReadingSocialNetwork.ipynb we analyze the Results of a School Survey

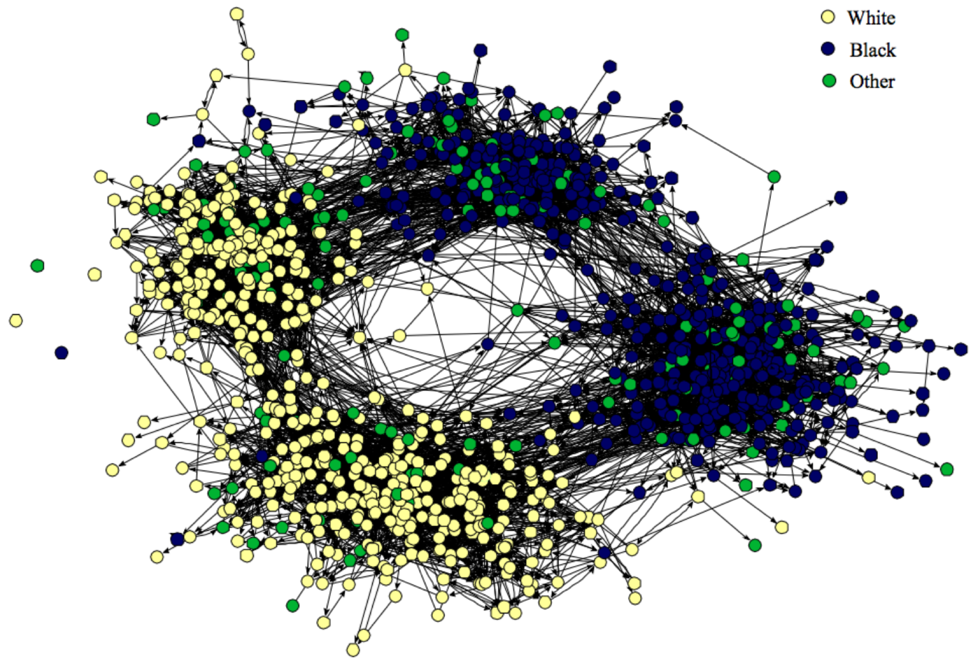


Fig. 3.4 *Friendship network of children in a U.S. school. Friendships are determined by asking the participants, and hence are directed, since A may say that B is their friend but not vice versa. Vertices are color coded according to race, as marked, and the split from left to right in the figure is clearly primarily along lines of race. The split from top to bottom is between middle school and high school, i.e., between younger and older children. Picture courtesy of James Moody.*

In the SchoolEdges.csv and SchoolNodes.csv you find the result of a survey on friendships in a school.

Every student was given a paper-and-pencil questionnaire and a copy of a list with every schoolmate. Weighted dyadic links were generated based on the number of sheared activities.

Weights were in the range from 1, meaning the student nominated the friend without reporting any activity, to 6 meaning that the student nominated the friend and reported participating in five activities with (him/her). Answer the following questions

~/CE88/Lec1/Powerlaw — less SchoolNodes.csv

```
Id Label "Nodes " Sex Race Grade Scode totalnoms
1 1 1 1 1 10 1 0
2 2 2 2 1 12 1 0
3 3 3 1 5 8 1 3
4 4 4 2 1 12 1 4
5 5 5 2 3 12 1 10
6 6 6 2 1 12 1 10
7 7 7 1 1 7 1 9
8 8 8 2 1 11 1 8
```

Sex 1,2,0 for Male, Femake or Unknown

Race: 1,2,3,4,5 for White, Black, Hispanic, Asian, Other
Grade from 7 to 12

Scode school code
Totalnoms —Identification numbers of the friends that the respondent nominated

	source	target	type	id	label	weight
0	3	37	Directed	2127	NaN	1.0
1	3	41	Directed	2128	NaN	1.0
2	3	105	Directed	2129	NaN	1.0
3	4	35	Directed	2130	NaN	2.0
4	4	62	Directed	2131	NaN	1.0
5	4	100	Directed	2132	NaN	4.0
6	4	110	Directed	2133	NaN	2.0
7	5	4	Directed	2134	NaN	5.0
8	5	23	Directed	2135	NaN	5.0
9	5	46	Directed	2136	NaN	5.0
10	5	49	Directed	2137	NaN	5.0
11	5	56	Directed	2138	NaN	3.0
12	6	17	Directed	2139	NaN	2.0

```python
deg = G.degree()
to_keep = []
for node in G.nodes():
    if deg[node] != 0:
        to_keep.append(node)
    else:
        print("Node: ",node," degree: ",deg[node])
#Create the network only with connected nodes
G=G.subgraph(to_keep)
```

```
Node:  1  degree:  0
Node:  64  degree:  0
Node:  67  degree:  0
```

Write the neighbours of each node

```python
for node in G.nodes():
    print ("Neighbors of ", node, " are : ", list(G.neighbors(node)))
```

```
Neighbors of  2  are :  []
Neighbors of  3  are :  [37, 41, 105]
Neighbors of  4  are :  [35, 62, 100, 110]
Neighbors of  5  are :  [4, 23, 46, 49, 56]
Neighbors of  6  are :  [17, 55, 110]
Neighbors of  7  are :  [24, 34, 39, 40, 45, 48, 69, 92]
Neighbors of  8  are :  []
Neighbors of  9  are :  [35, 38, 65, 104]
Neighbors of  10  are :  [77]
Neighbors of  11  are :  [21, 63, 82, 84, 87, 102, 111]
Neighbors of  12  are :  [7, 19, 34, 39, 40, 45, 50, 70]
Neighbors of  13  are :  []
Neighbors of  14  are :  [2, 17, 19, 25, 59, 68, 81, 103, 107]
Neighbors of  15  are :  [7, 24, 34, 39, 40, 45, 48, 92]
Neighbors of  16  are :  [57, 65, 95, 96]
Neighbors of  17  are :  []
Neighbors of  18  are :  [8, 9, 13, 28, 84, 90, 97, 100]
Neighbors of  19  are :  [12, 14, 16, 18, 101]
```

Question 3: Check here https://networkx.github.io/documentation/networkx-1.10/reference/classes.digraph.html The function DiGraph.predecessors() and write the nodes the links that nominated each node

Question 4: Write the number of nodes and edges

Question 5: Write the average in_degree, out_degree, and degree of the network

Question 6: Calculate Average Clustering Coefficient, and average shortest path in the Network

```
0.24046205463552878
```

```
2.677397023191416
```