Generation and Properties of Large Directed Erdös-Renyi Model Random Graphs

Michael Gordo

Matthew Grunert

ICSI 660 – Network Science

December 18, 2013

Introduction

This project focused on the generation of large directed graphs using the Erdös-Renyi random graph model. Our generation used the **G**(n,p) method, taking n as the number of nodes in the graph and p as the independent probability that a directed edge exists between any given pair of nodes. We generated a total of nine graphs, with $n = \{10{,}000, 100{,}000, 1{,}000{,}000\}$ and $np = \{0.75, 1.0, 1.25\}$.

Goals

We have five goals we want to measure using this project:

I. The number of nodes with indegree zero,
II. The number of nodes with outdegree zero,
III. The size of the largest weakly connected component,
IV. The size of the largest strongly connected component, and
V. The number of directed cycles of length k (for small k).

Methodology

To accomplish our first goal, two fields were built into the graph object: arrays of integers to hold the respective value of each node as corresponding to the index of the array. Thus, completing our first and second goals becomes trivially easy. We implement the weakly connected components aspect of our project using breadth-first search on an undirected clone of the original graph. Strongly connected components are found using depth-first search on the original directed graph. Finally, Mr. Gordo has implemented his own algorithm for finding directed cycles of length k; we know this method to be accurate up to size $k = 10$.

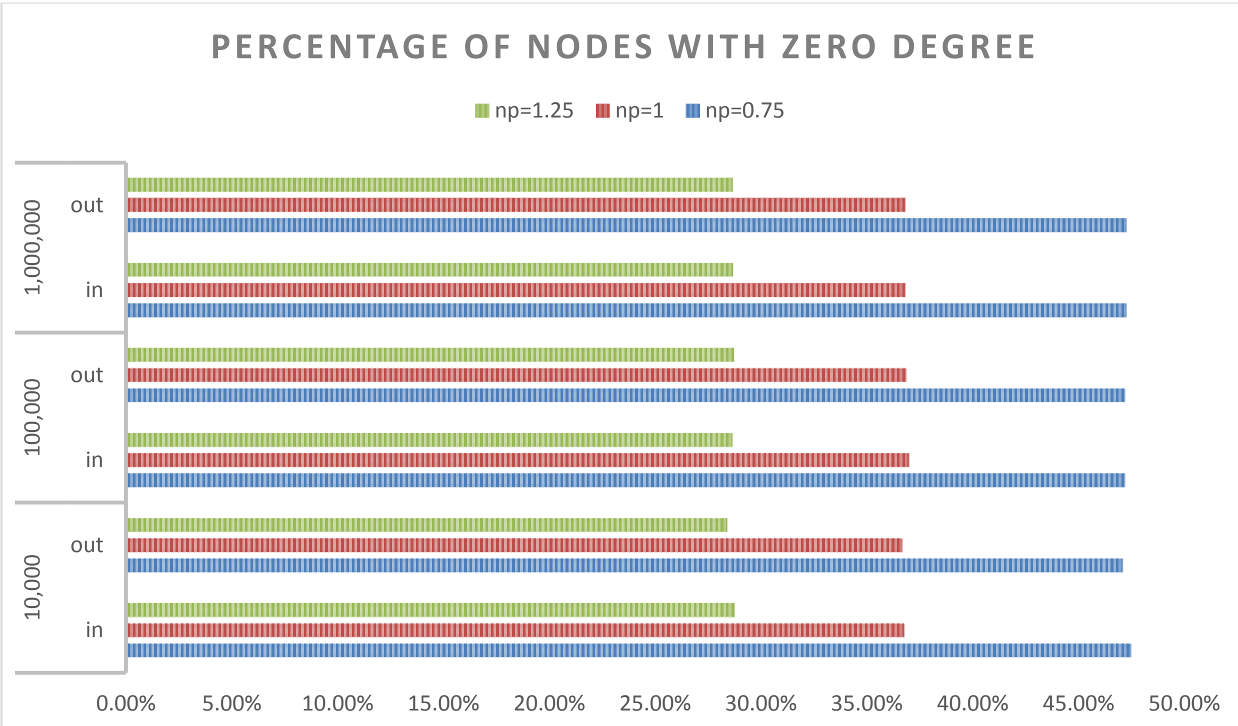
Data

Below is the table showing the percentages of each n with indegree or outdegree of zero:

n		**np**		
		0.75	**1**	**1.25**
10,000	**in**	47.47%	36.75%	28.76%
	out	47.08%	36.67%	28.41%
100,000	**in**	47.18%	36.98%	28.65%
	out	47.18%	36.85%	28.72%
1,000,000	**in**	47.27%	36.80%	28.68%
	out	47.26%	36.80%	28.66%

As can be plainly seen, the percentage of nodes with indegree or outdegree of zero appears near-constant across the random graph generation. This is because each edge is included in the graph with equal probability.

For easier comparison:



Below is the table showing the number of connected components within the graph:

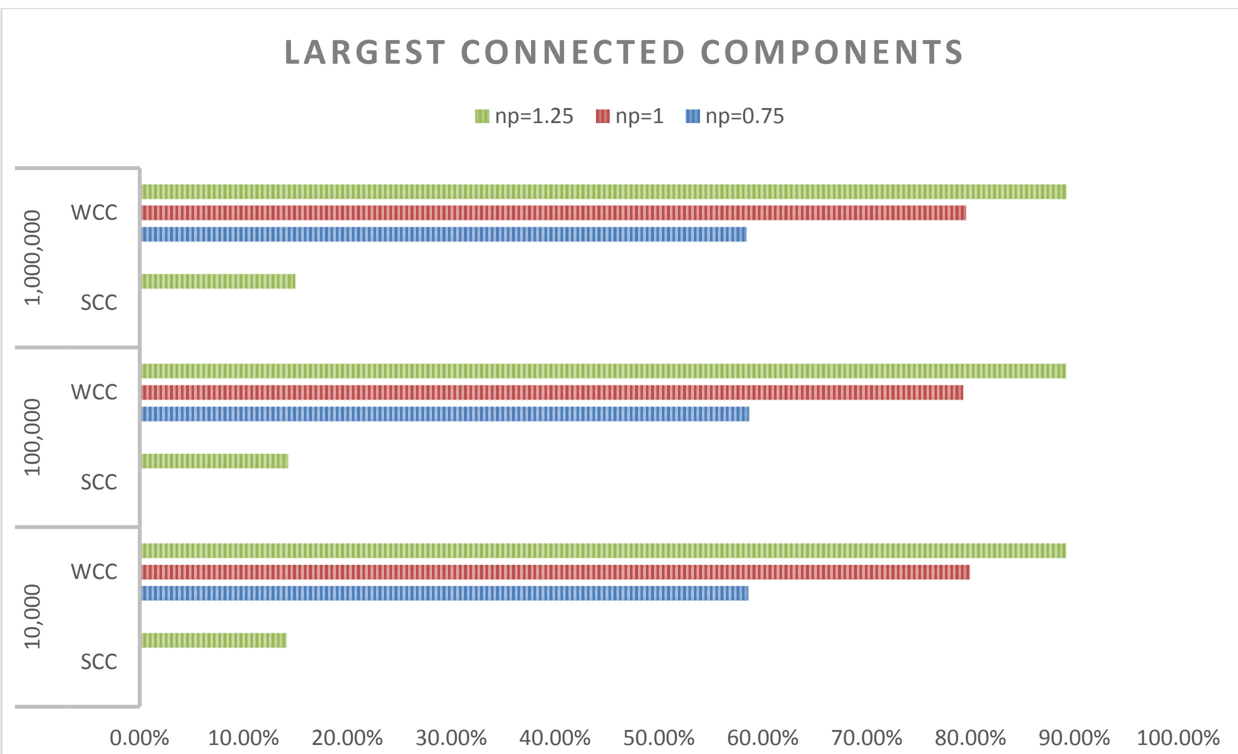
n		np		
		0.75	**1**	**1.25**
10,000	**SCC**	4	7	1,416
	WCC	5,860	7,991	8,918
100,000	**SCC**	1	7	14,300
	WCC	58,644	79,307	89,196
1,000,000	**SCC**	3	11	149,914
	WCC	583,919	795,813	892,278

Here is that table same table as percentages:

n		np		
		0.75	**1**	**1.25**
10,000	**SCC**	4	7	1,416
	WCC	5,860	7,991	8,918
100,000	**SCC**	1	7	14,300
	WCC	58,644	79,307	89,196
1,000,000	**SCC**	3	11	149,914
	WCC	583,919	795,813	892,278

Here it is interesting to note that the number of strongly connected components within the graph jumps to more than ten percent of the nodes for *np* = 1.25, contrary to the relatively tiny components of the other *np* values, which don't seem to vary excessively, despite greatly varying *n*. The number of weakly connected components within the graph seems to grow logarithmically, although the size of the component stays near-constant across varying *n*; in fact, it only changes by a single order of magnitude.

For easier comparison:



Finally, we have the results from Mr. Gordo's *k*-cycle algorithm:

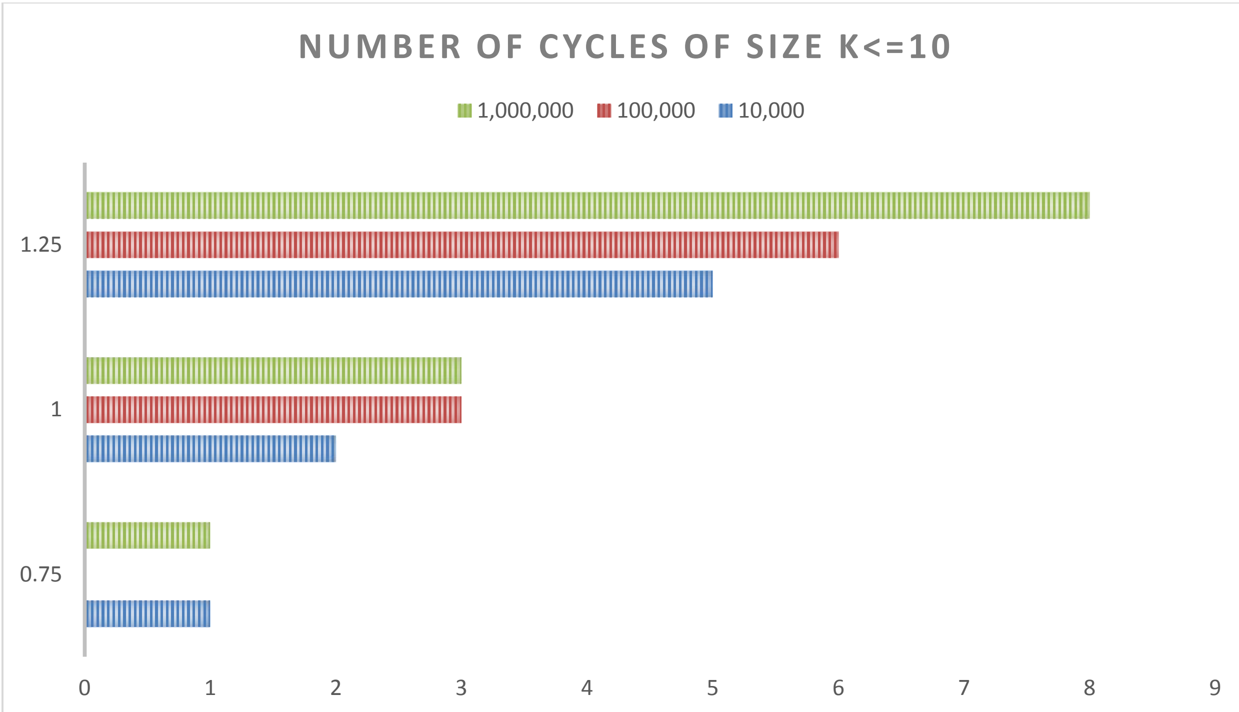
n		np		
		0.75	**1**	**1.25**
10,000	**k=2**	0	0	1
	k=3	0	0	1
	k=4	1	1	1
100,000	**k=2**	0	1	0
	k=3	0	0	0
	k=4	0	0	1
1,000,000	**k=2**	0	0	0
	k=3	1	0	1
	k=4	0	0	1

The total number of cycles of size { $2 \leq k \leq 10$ }:

n		np		
		0.75	**1**	**1.25**
10,000	**k=2..10**	1	2	5
100,000	**k=2..10**	0	3	6
1,000,000	**k=2..10**	1	3	8

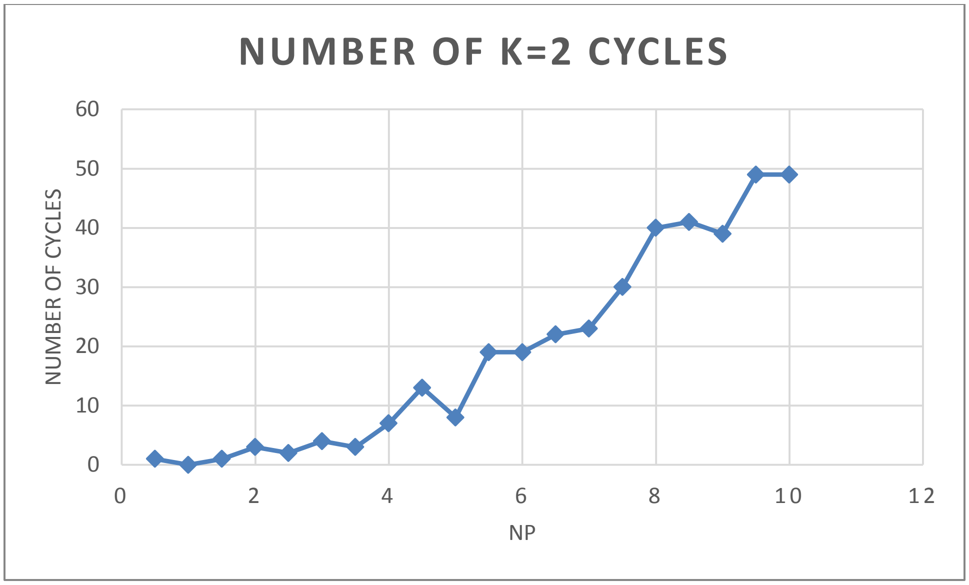
The number of directed *k*-cycles is almost negligibly tiny for *np* = 0.75. We see a marked increase when *np* = 1.0, though it is more interesting to note that the number of directed *k*-cycles increases dramatically when *np* = 1.25.

A view for easier comparison is available on the next page:



For n = 10,000:

np	# of k=2 cycles
0.5	1
1	0
1.5	1
2	3
2.5	2
3	4
3.5	3
4	7
4.5	13
5	8
5.5	19
6	19
6.5	22
7	23
7.5	30
8	40
8.5	41
9	39
9.5	49
10	49



As expected, the number of *k*-cycles of size 2 increases as *np* increases.

Conclusion

We have concluded from this project that several aspects of Erdös-Renyi random graph generation seem to behave nearly independently of the supplied *np*. The percentages of nodes with zero-degree measures are almost constant across the board, while the number of connected components only seems to vary dramatically for *np* = 1.25. What we found most interesting is how the number of directed *k*-cycles stays very small for *np* = {0.75, 1.0} and then jumps for *np* = 1.25; this holds with our previous observations.