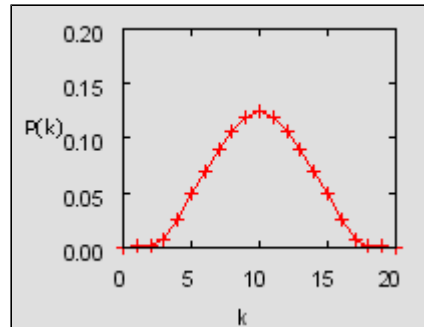
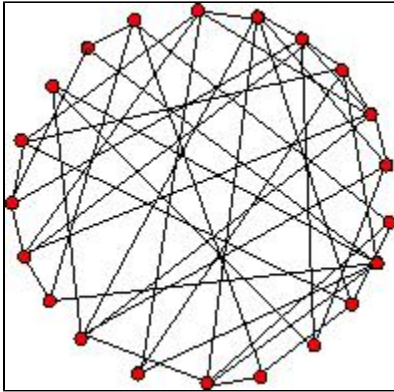


Random Graph

Description

The random graph model generates a graph that has a fixed number of nodes which are connected randomly by undirected edges, see image to the left below. The number of edges depend on a specified probability. The edge probability is chosen based on the number of nodes in the graph. The model most commonly used for this purpose was introduced by Gilbert. This is known as the $G(n,p)$ model. "n" being the number of vertices and "p" the linking probability. The number of edges created according to this model is not known in advance. Erdős-Rényi introduced a similar model where all the graphs with "m" edges are equally probable and "m" varies between 0 and $n(n-1)/2$. This is known as the $G(n,m)$ model. The degree distribution is Poissonian in both the generating mechanisms, see image to the right below.



Pros & Cons

It is a mathematically well understood model and has been studied in detail. It can be used to generate networks of any size. It does not generate networks with high clustering coefficient and scale free structure.

Applications

Very few real world networks are random. However, random networks are a theoretical construct that is well understood and their properties can be exactly solved. They are commonly used as a reference, e.g., in tests of network robustness and epidemic spreading.

Implementation Details

The present random graph generator implements the $G(n,p)$ model by Gilbert. The input is the total number of nodes in the network and their wiring probability. The output generates a network in which each pair of nodes is connected by an undirected edge with the probability specified in the input.

Usage Hints

A small random network can be generated by using an input of 50 nodes and a wiring probability of 0.5. A wiring probability of 0 would generate a network without any edges and a wiring probability of 1 with "n" nodes will generate a network with " $n(n-1)$ " edges. The wiring probability should be chosen dependent on the number of vertices. For a large number of vertices the wiring probability should be smaller.

Links

- [Source Code](#)

Acknowledgements

The FORTRAN implementation was done by Santo Fortunato and integrated by Santo Fortunato and Weixia (Bonnie) Huang. Documentation compiled by Soma Sanyal.

References

1. Gilbert, E. N. (1959). Random Graphs. *Ann. Math. Stat.* 30:1141.
2. Erdős, P. and Rényi, A. (1959). On Random Graphs I. *Publicationes Mathematicae Debrecen*, 6:290--297.
3. [Batagelj, V. and Brandes, U. \(2005\). Efficient generation of large random networks.](#) *Physical Review E* 71:036113-036118. (Access restricted to APS subscribers.)

See Also



The license could not be verified: License Certificate has expired! [Generate a Free license now.](#)