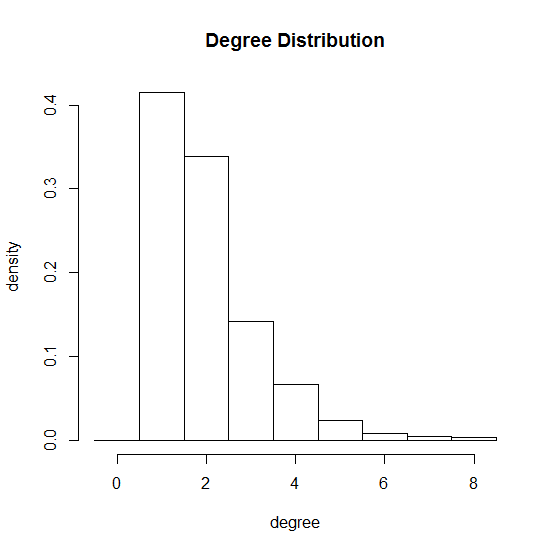
Problem 2

1. Here we use function sample\_pa to create the network and the diameter of the graph is 29.56 for which we create the network 100 times and take the mean value. The degree distribution is as following:



1. Here we implement the function is.connected to test the connectivity of the network, which is always connected when we repeated for 100 times.

For GCC: Since the entire graph is connected and GCC is the largest connected component, so the GCC here is the whole graph.

Community Structure: We use the function cluster\_fast\_greedy to find the structure and use modularity() to get the modularity of the graph.

Number of communities: 32.

Modularity: 0.9345737.

Why is it so large?

The modularity for the graph is 0.9345737. The reason that it has such a large modularity is because the graph is generated based on Barabasi model, which has preferential attachment mechanism, in which the new added nodes tend to be linked to the nodes with high degree. So it has dense connections between the nodes within certain communities but sparse connections between nodes in different communities. Thus the modularity is large for this graph.

1. Also we use sample\_pa to generate the network and the structure as well as modularity:

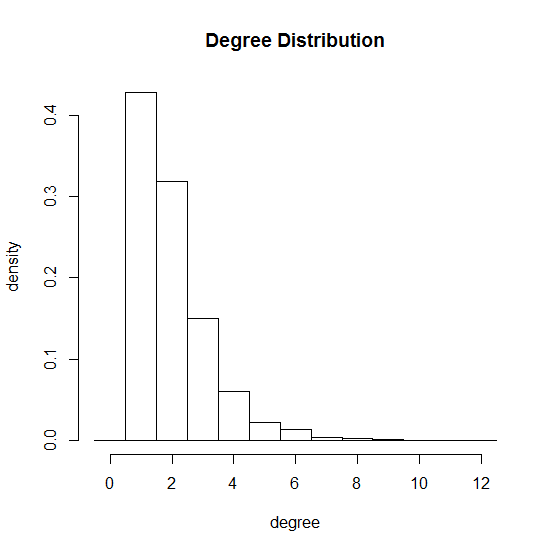
Number of communities: 105

Modularity: 0.9791046

Obviously, the modularity is slightly larger than the 1000 nodes graph.

For the graph which has 10000 nodes:

Degree Distribution is as following:



(d)Here we create the network based on function neighborhood() and the degree distribution of nodes j is as following:

