22mcb1002

May 17, 2023

1 Barabasi-Albert model

Without using pre-defined function

[2]:

```
import numpy as np
import random as rd
import networkx as nx
import matplotlib.pyplot as plt
```

Nodes, Edge and empty graph

[14]:

```
n = 50
m = 2
```

[16]:

```
ba_graph = nx.Graph()
ba_graph.add_nodes_from(range(m + 1))
```

Network growth

[17]:

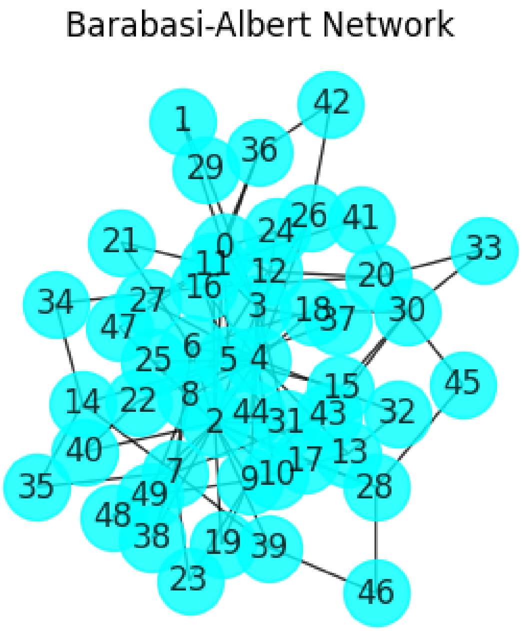
```
for i in range(m + 1, n):
    selected_nodes = rd.choices(list(ba_graph.nodes()), k=m)
    ba_graph.add_node(i)
    ba_graph.add_edges_from([(i, node) for node in selected_nodes])
```

[40]:

```
plt.figure(figsize=(8, 4))
plt.subplot(1, 2, 1)
nx.draw(ba_graph, with_labels=True, node_size=600, alpha=0.8, node_color =␣
 ↪'cyan')
plt.title("Barabasi-Albert Network")
```

[40]:

```
Text(0.5, 1.0, 'Barabasi-Albert Network')
```

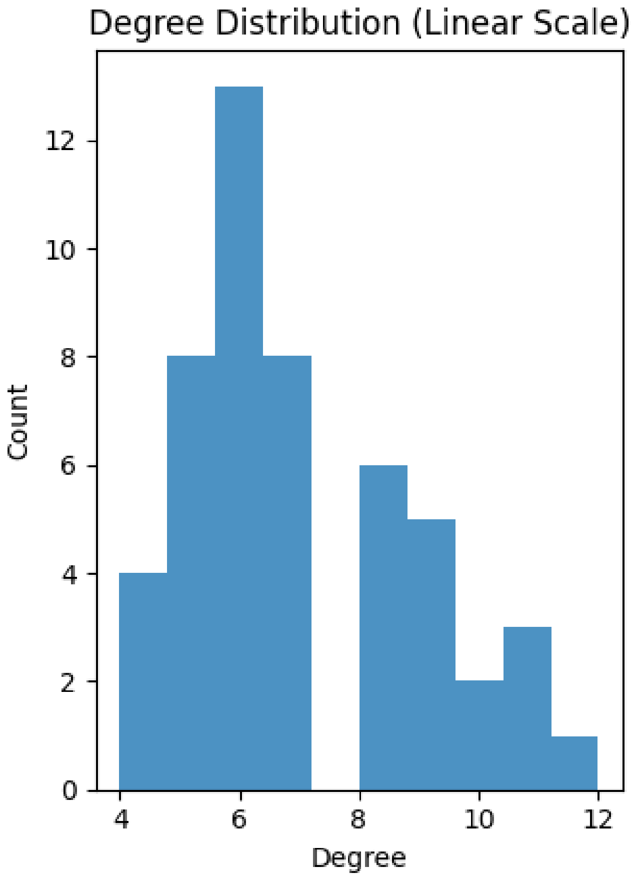


Degree distribution on linear scale

[26]:

```
plt.subplot(1, 2, 2)
degrees = [degree for node, degree in ba_graph.degree()]
plt.hist(degrees, bins=10, alpha=0.8)
plt.xlabel("Degree")
plt.ylabel("Count")
plt.title("Degree Distribution (Linear Scale)")

plt.tight_layout()
plt.show()
```



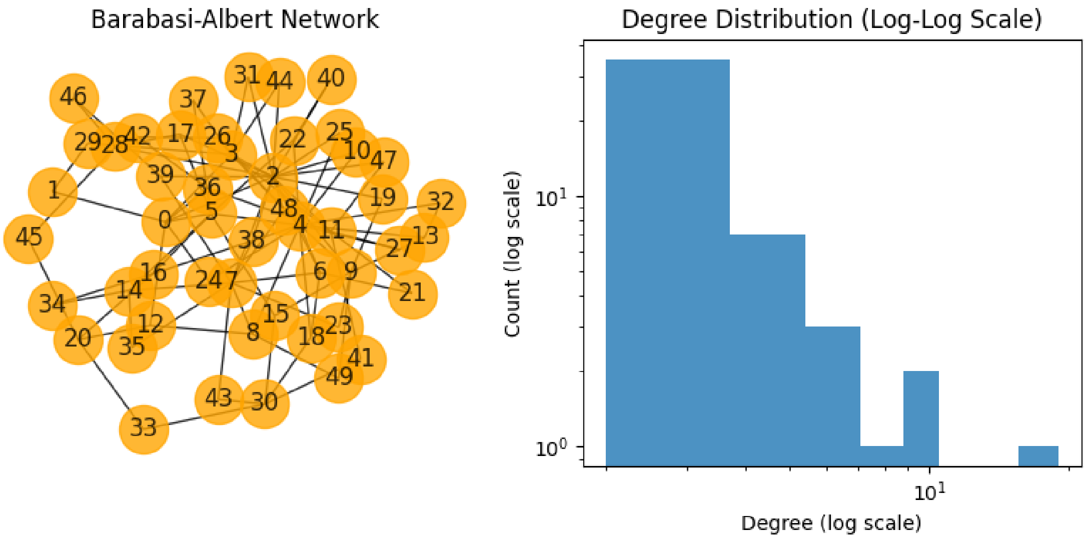
Degree distribution on log scale

[39]:

```
plt.figure(figsize=(8, 4))
plt.subplot(1, 2, 1)
nx.draw(ba_graph, with_labels = True, node_size = 600, alpha = 0.8, node_color␣
  ↪= 'orange')
plt.title("Barabasi-Albert Network")

plt.subplot(1, 2, 2)
degrees = [degree for node, degree in ba_graph.degree()]
plt.hist(degrees, bins=10, alpha=0.8)
plt.xscale('log')
plt.yscale('log')
plt.xlabel("Degree (log scale)")
plt.ylabel("Count (log scale)")
plt.title("Degree Distribution (Log-Log Scale)")

plt.tight_layout()
plt.show()
```



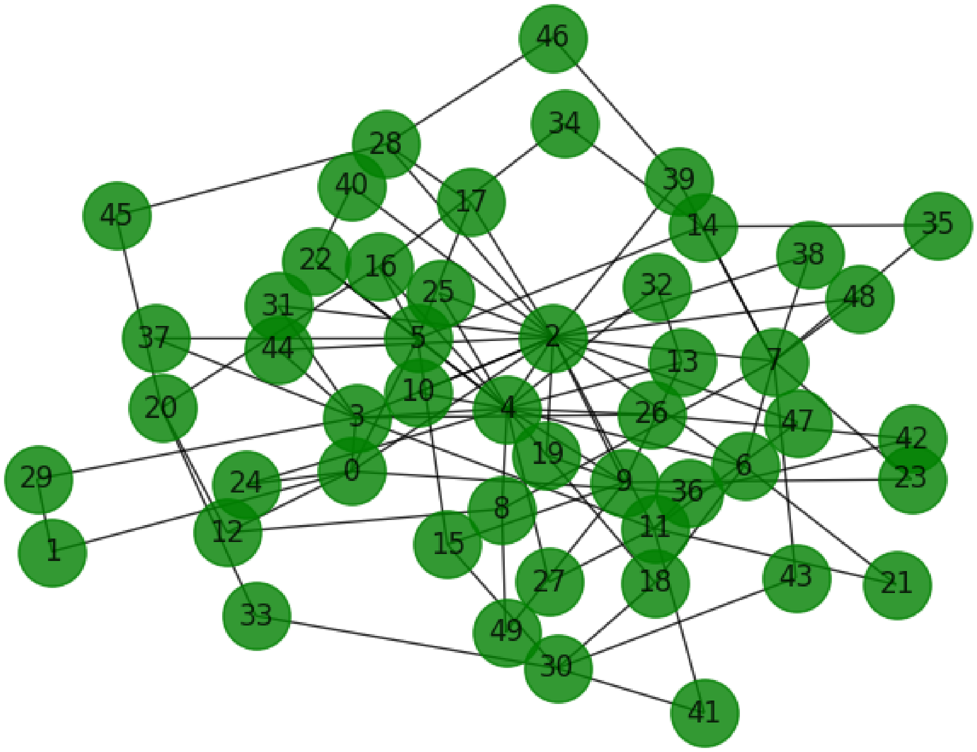
Using pre-defined function

```
[37]: n = 50
      m = 2

      ba_graph = nx.barabasi_albert_graph(n, m)

      nx.draw(ba_graph, with_labels = True, node_size = 800, alpha = 0.8, node_color␣
        ↪= 'green')
      plt.title("Barabasi-Albert Network")
      plt.show()
```

Barabasi-Albert Network


46
34
28
39
40
17
45
14
35
22
16
38
32
25
31
48
37
5
2
44
13
7
10
20
4
26
3
47
42
19
29
0
6
24
9
36
23
8
11
12
15
1
27
18
43
21
33
49
30
41