The Engineering World #DataScience 27 & 28

May 31, 2018

AKKAL BAHADUR BIST
DATA SCIENTIST AT
KATHMANDU INSTITUTE OF APPLIED SCIENCES (KIAS)
Center for Conservation Biology (CCB)

1 NETWORK ANALYSIS

1.0.1 Network analysis use cases

Social media marketing analysis, Infrastructure system design, Financial risk management, Public health management

1.0.2 Network

A body of connected data that's evaluated during graph analysis

1.0.3 Graph

A data visualization schematic depicting the data that comparises a network

1.0.4 Network analysis vocablary

Nodes: the vertices around which a graph is formed

Edges: the lines that connect vertices within a graph

Directed graph(aka digraph): a graph where there is a direction assign to each edge that connects a node

Directed edge: an edge feature that has been assign a direction between nodes

Undirected graph: a graph where all edges are bidirectional

Undirected eddge: a bidirectional edge feature

Graph size: the number of edge in a graph

Graph order: number of vertices is a graph

Degree: the number of edges connected to a vertex, with loops counted twice

1.0.5 Graph generator

The functions that generates graphs

graph generator has most important application is Synthetic veriation of A particular graph

Type of graph generators Graph drawing algorithms
Network analysis algorithms
Algorithmic routing for graphs
Graph search algorithms
Subgraphs algorithms

2 GRAPH OBJECT NETWORK ANALYSIS

You + Machine Learning = Scientific Discovery

2.0.1 Working with Graph objects

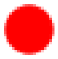
```
In [1]: import numpy as np
        import pandas as pd
        from pylab import rcParams
        import seaborn as sb
        import matplotlib.pyplot as plt
        import networkx as nx
```

```
In [2]: %matplotlib inline
        rcParams ['figure.figsize'] = 5,4
        sb.set_style ('whitegrid')
```

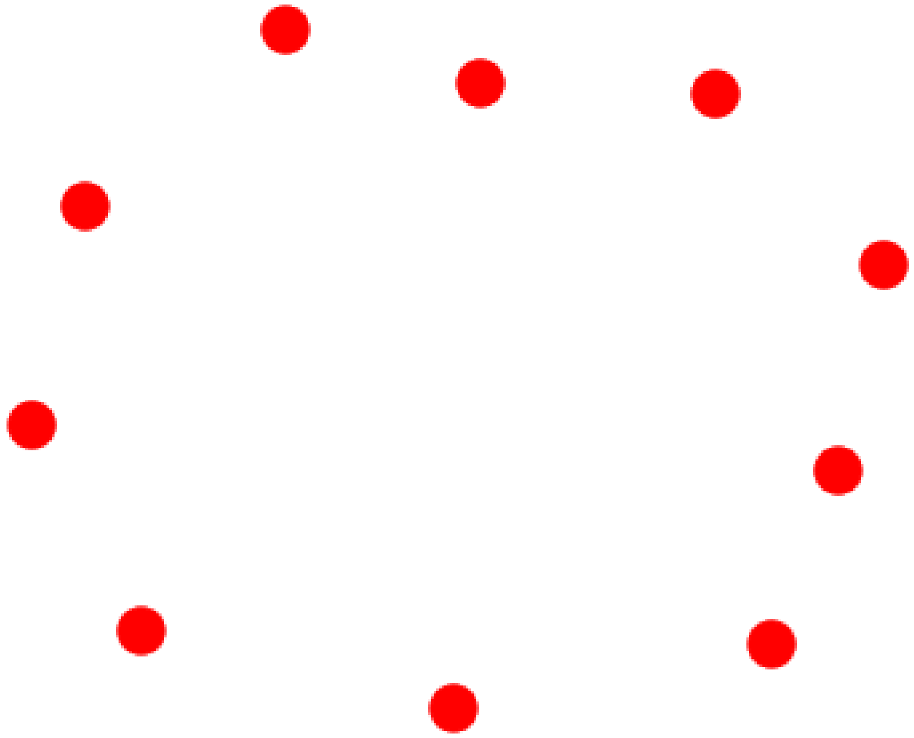
2.0.2 Creating graph objects

```
In [3]: G = nx.Graph() #empty graph drawing
        nx.draw(G)
```

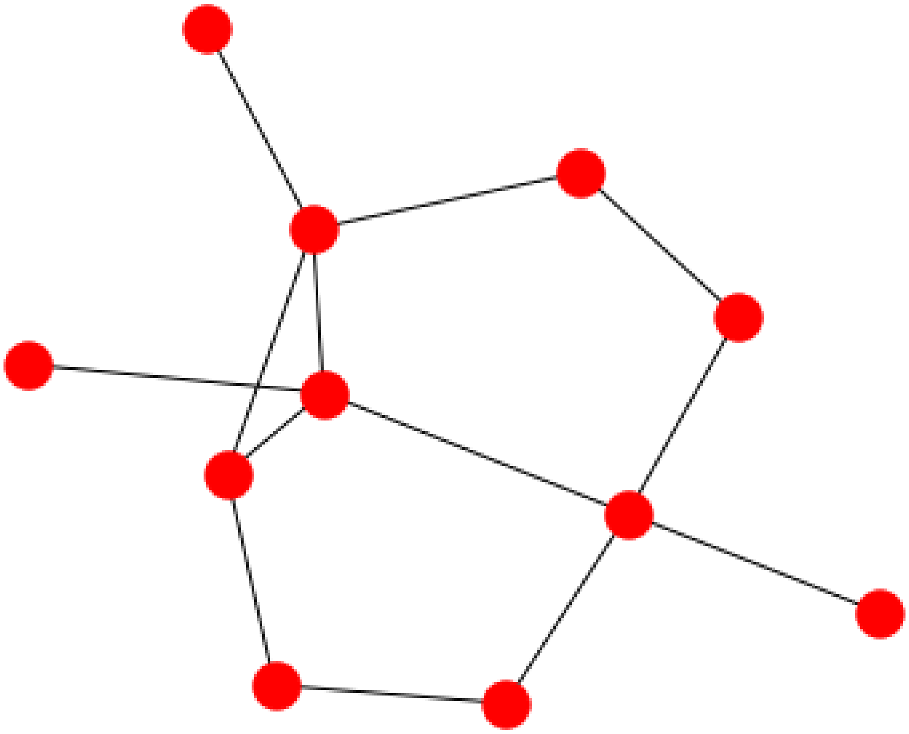
```
In [4]: G.add_node(1)
        nx.draw(G)
```



```
In [5]: G.add_nodes_from([1,2,3,4,5,6,7,8,9,10])
        nx.draw(G)
```

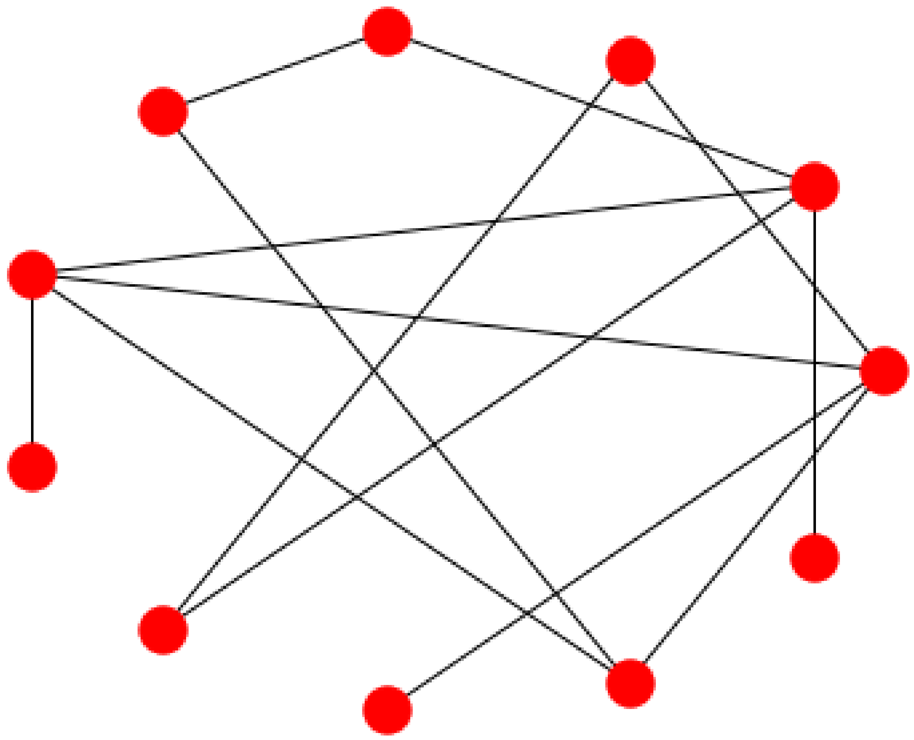


```
In [6]: G.add_edges_from([(2,4),(2,6),(2,8),(1,3),(1,10),(2,12),(1,6),(3,8),(1,10),(5,10),(5,4),
        nx.draw(G)
```

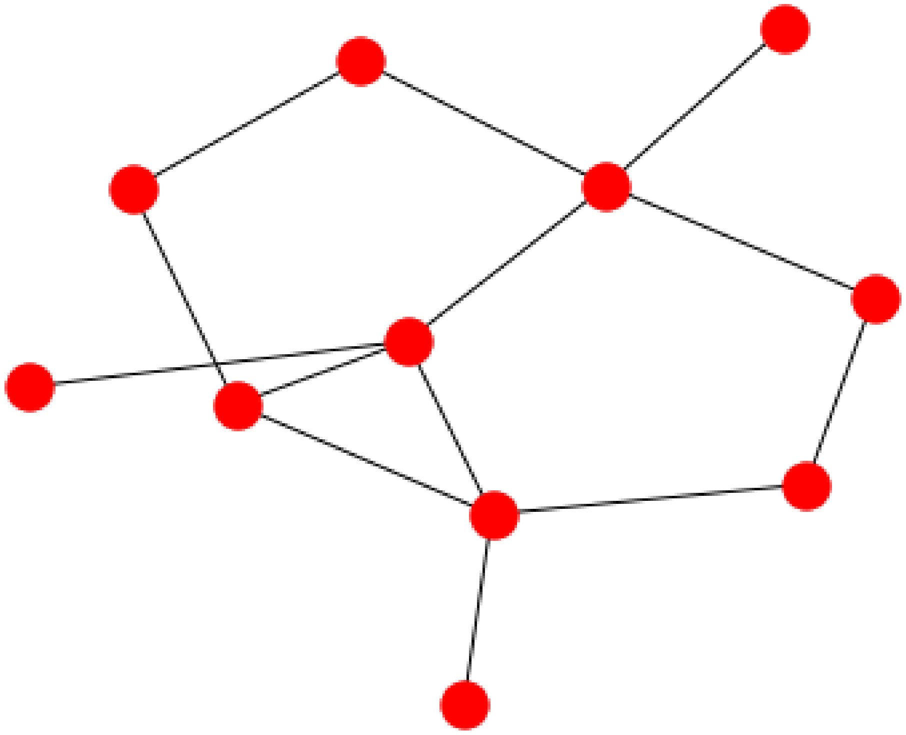


2.0.3 The basics about drawing graph objects

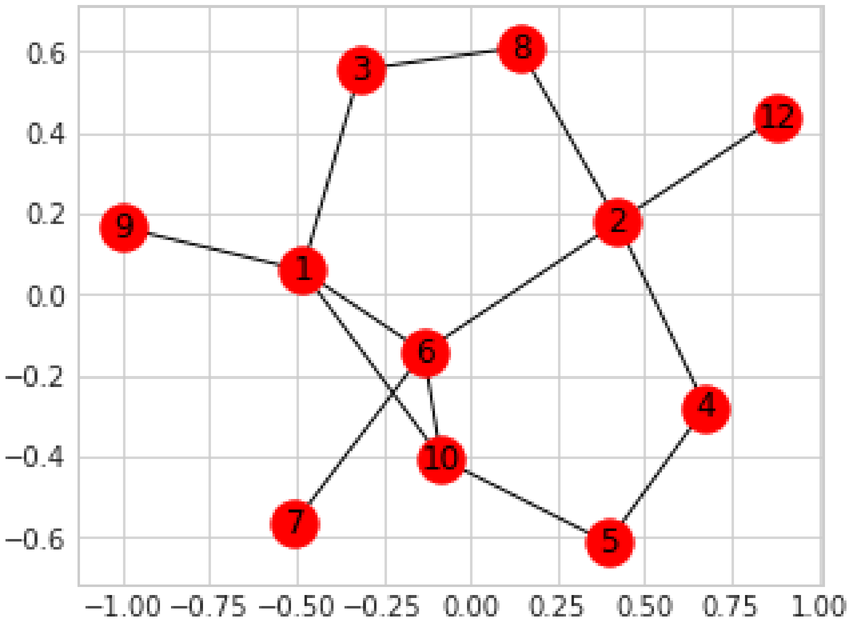
```
In [7]: nx.draw_circular(G)
```



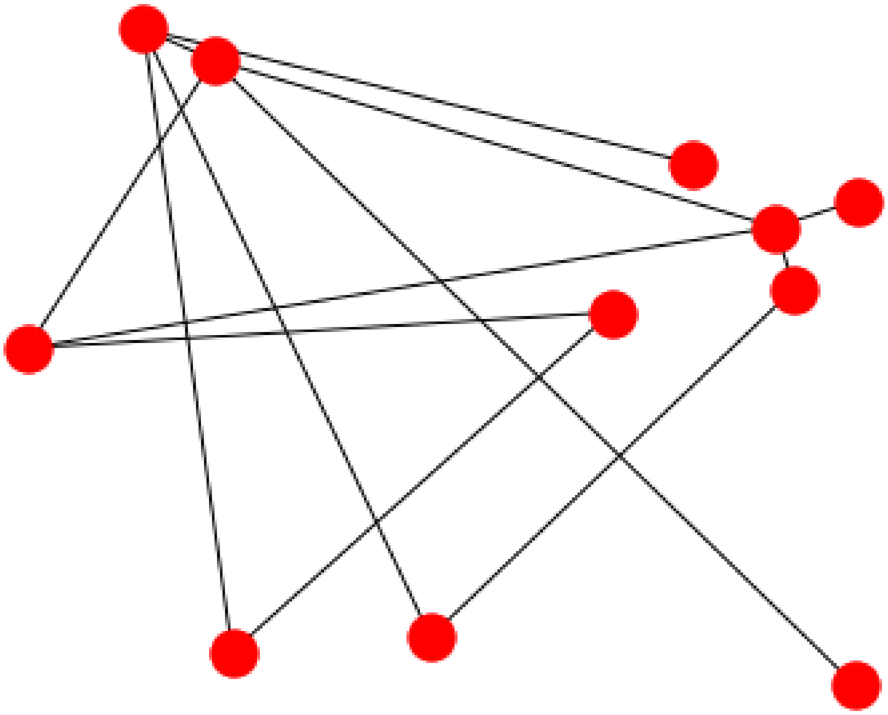
```
In [8]: nx.draw_kamada_kawai(G)
```



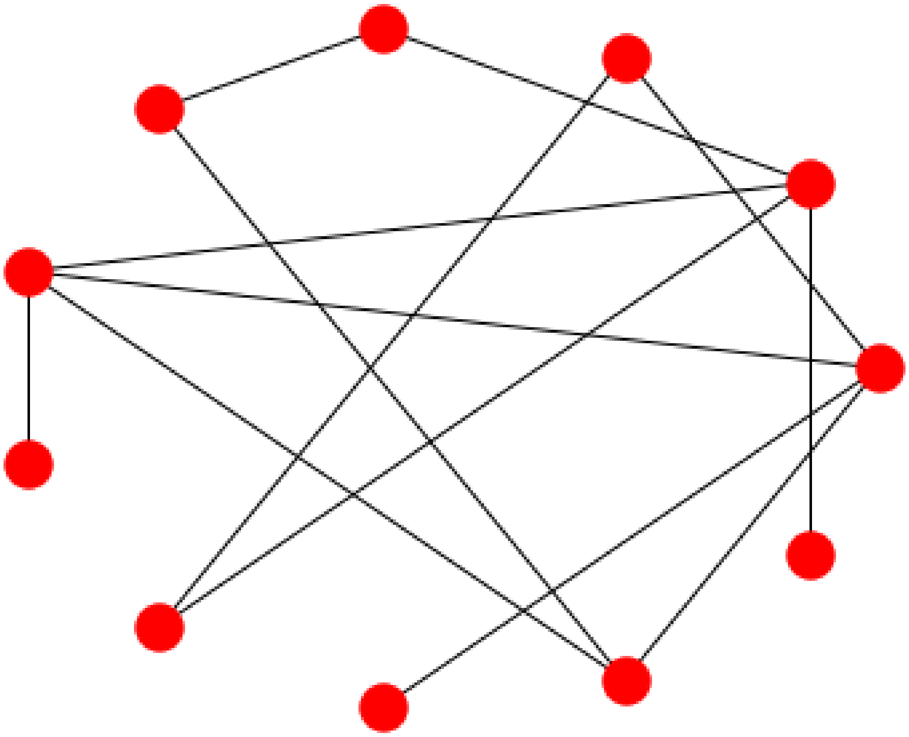
```
In [9]: nx.draw_networkx(G)
```



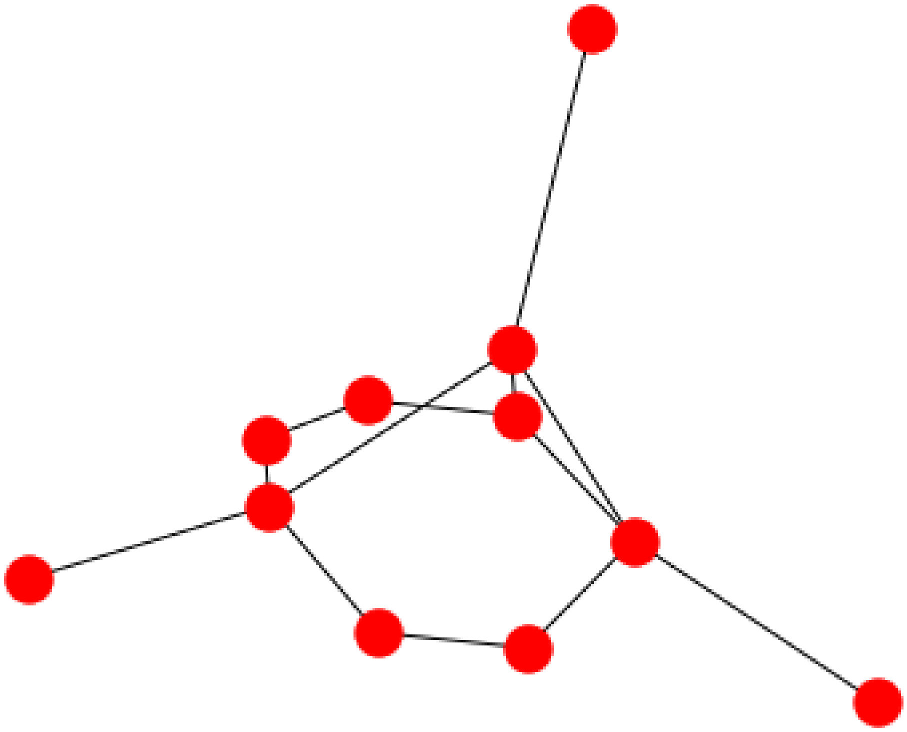
```
In [10]: nx.draw_random(G)
```



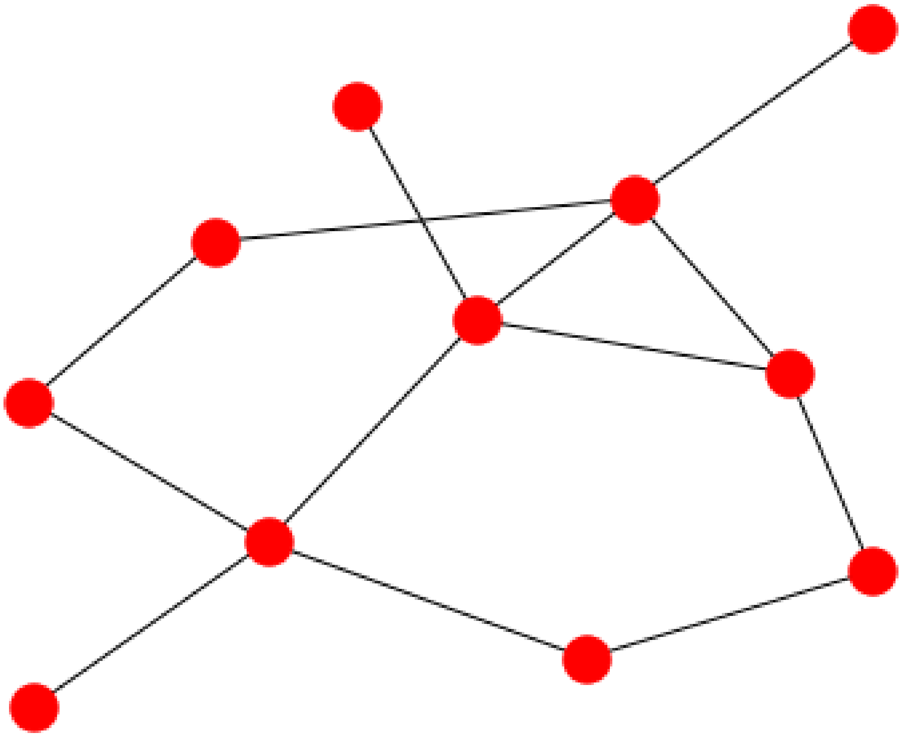
```
In [11]: nx.draw_shell(G)
```



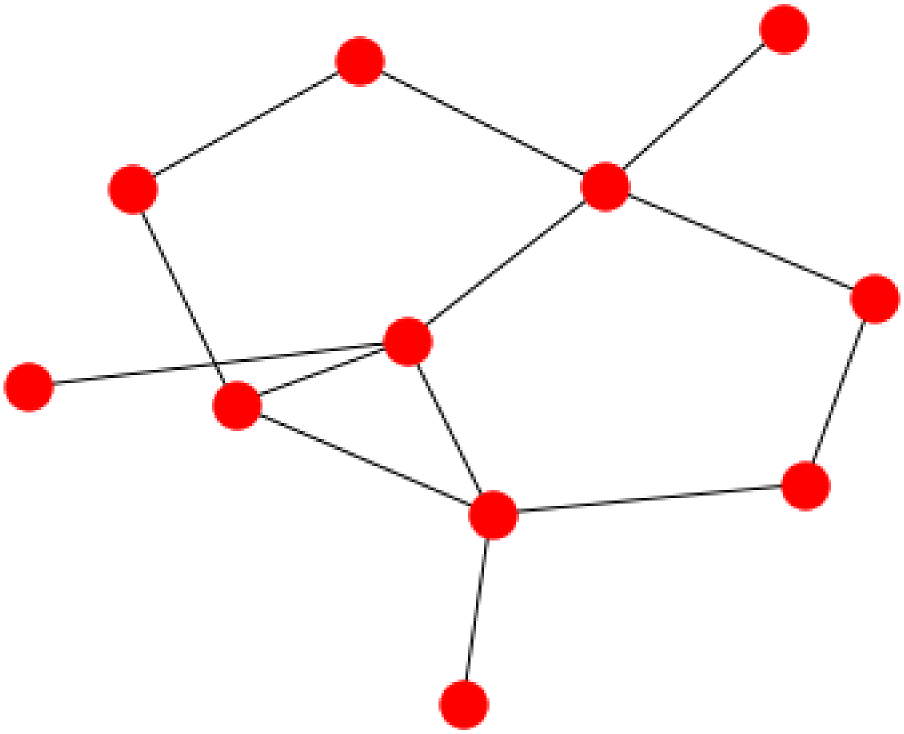
```
In [12]: nx.draw_spectral(G)
```



```
In [13]: nx.draw_spring(G)
```

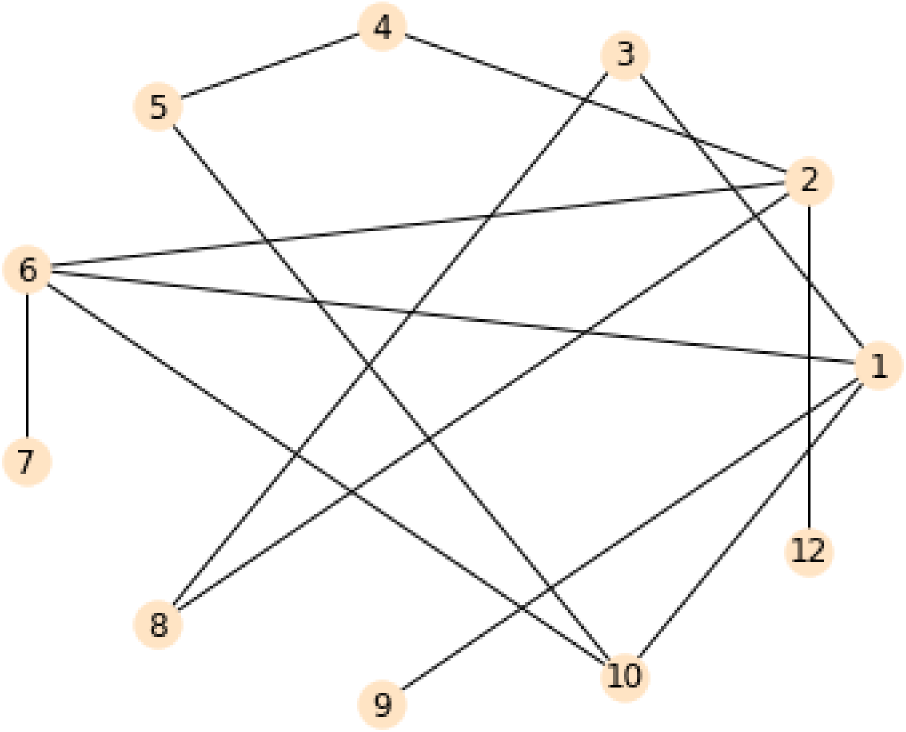


```
In [14]: nx.draw_kamada_kawai(G)
```

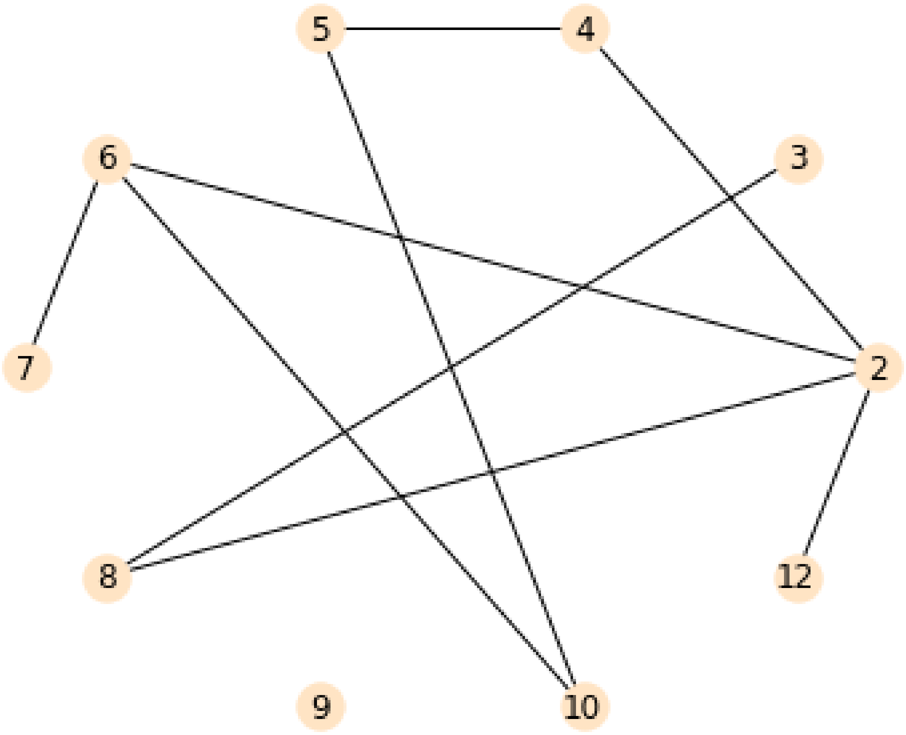


2.0.4 Labeling and coloring your graph plots

```
In [15]: nx.draw_circular(G, node_color = 'bisque', with_labels = True) #add node color and labe
```



```
In [16]: G.remove_node(1) #remove some nodes from graph plots
         nx.draw_circular(G, node_color = 'bisque', with_labels = True) #add node color and labe
```



2.0.5 Identify graph proporties

```
In [17]: sum_stats = nx.info(G)
         print (sum_stats)
```

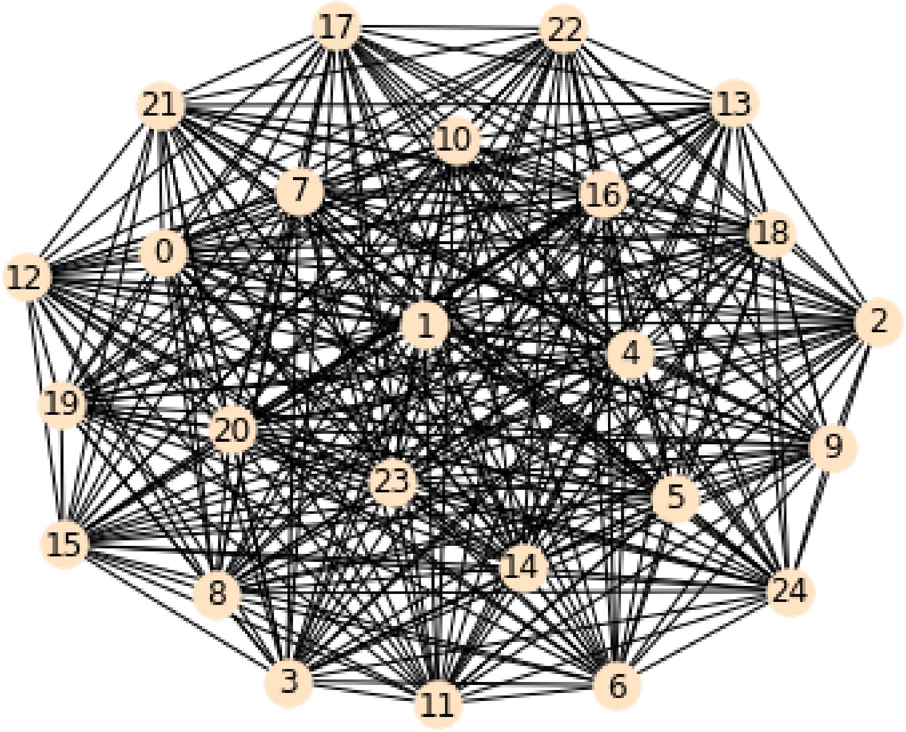
```
Name:
Type: Graph
Number of nodes: 10
Number of edges: 9
Average degree:   1.8000
```

```
In [18]: print (nx.degree(G))
```

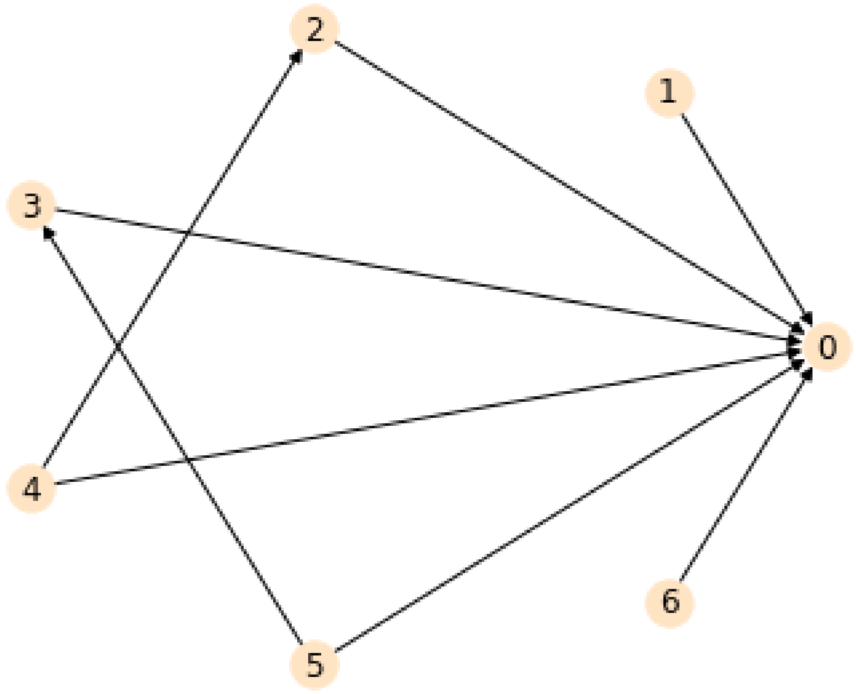
```
[(2, 4), (3, 1), (4, 2), (5, 2), (6, 3), (7, 1), (8, 2), (9, 0), (10, 2), (12, 1)]
```

2.0.6 Using graph generator

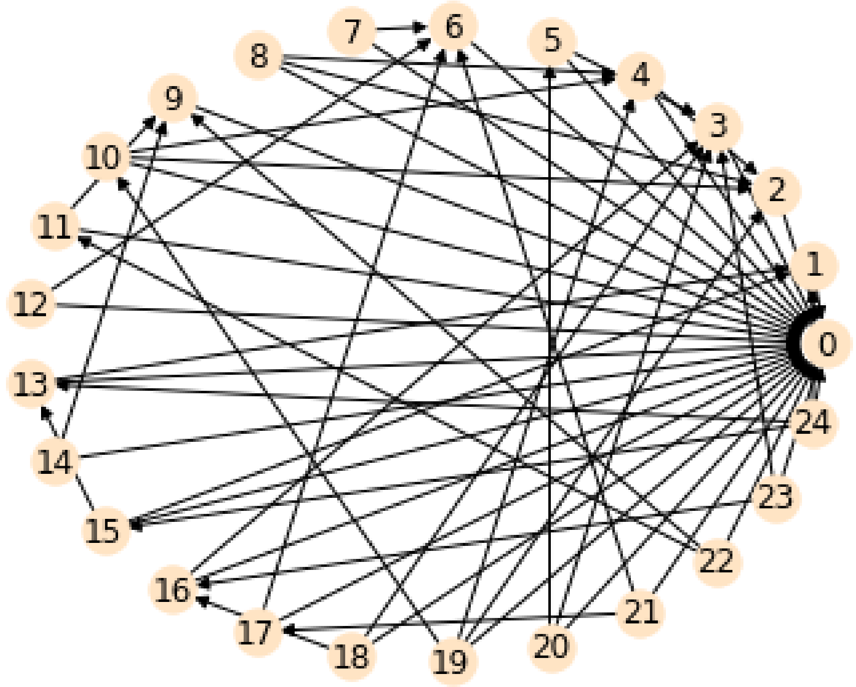
```
In [19]: G = nx.complete_graph(25)
         nx.draw(G, node_color='bisque', with_labels=True)
```



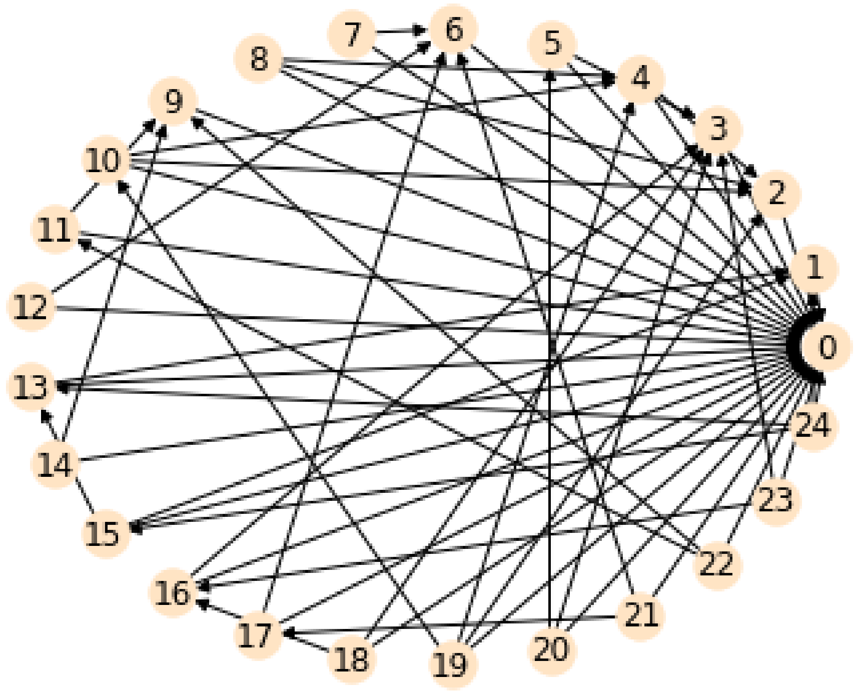
```
In [20]: G = nx.gnc_graph(7, seed=25) #add a direction arrow in the graph
         nx.draw_circular(G, node_color='bisque', with_labels=True)
```



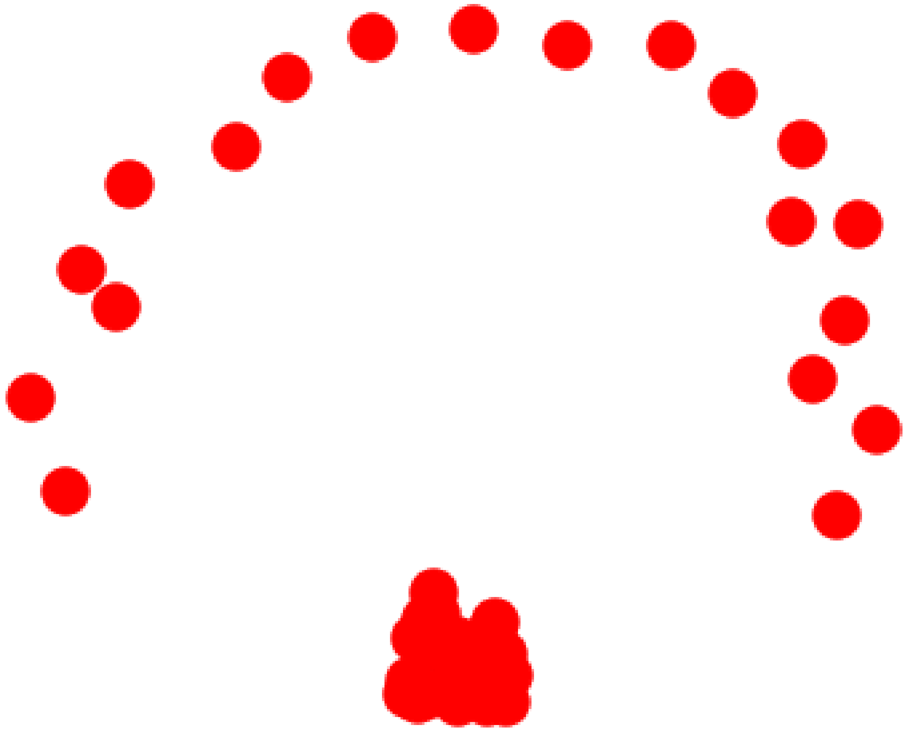
```
In [21]: G = nx.gnc_graph(25, seed=25)
         nx.draw_circular(G, node_color='bisque', with_labels=True)
```



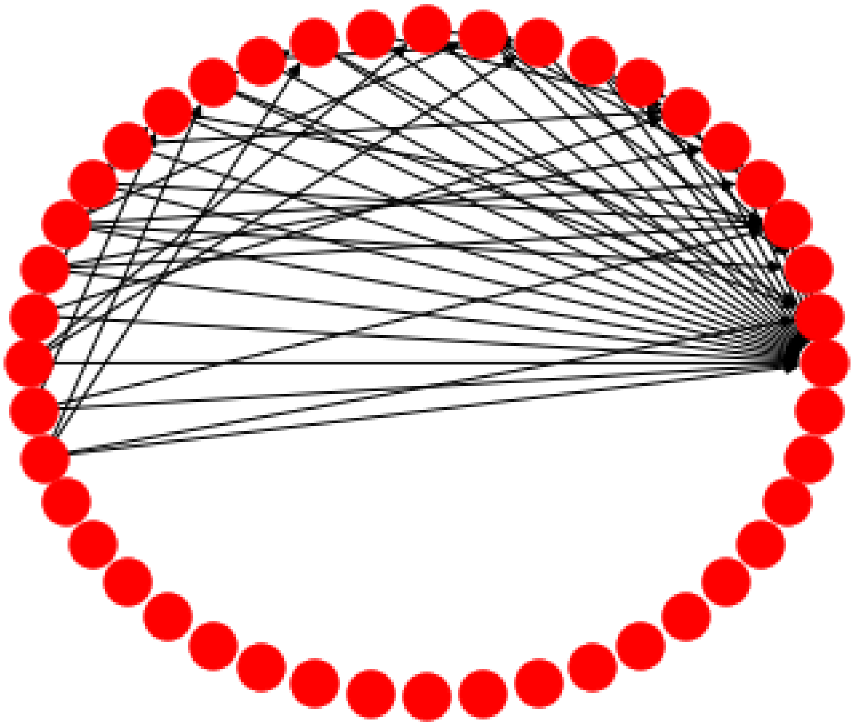
```
In [22]: ego_G = nx.ego_graph(G, 15, radius=5)
         nx.draw_circular(G, node_color='bisque', with_labels=True)
```



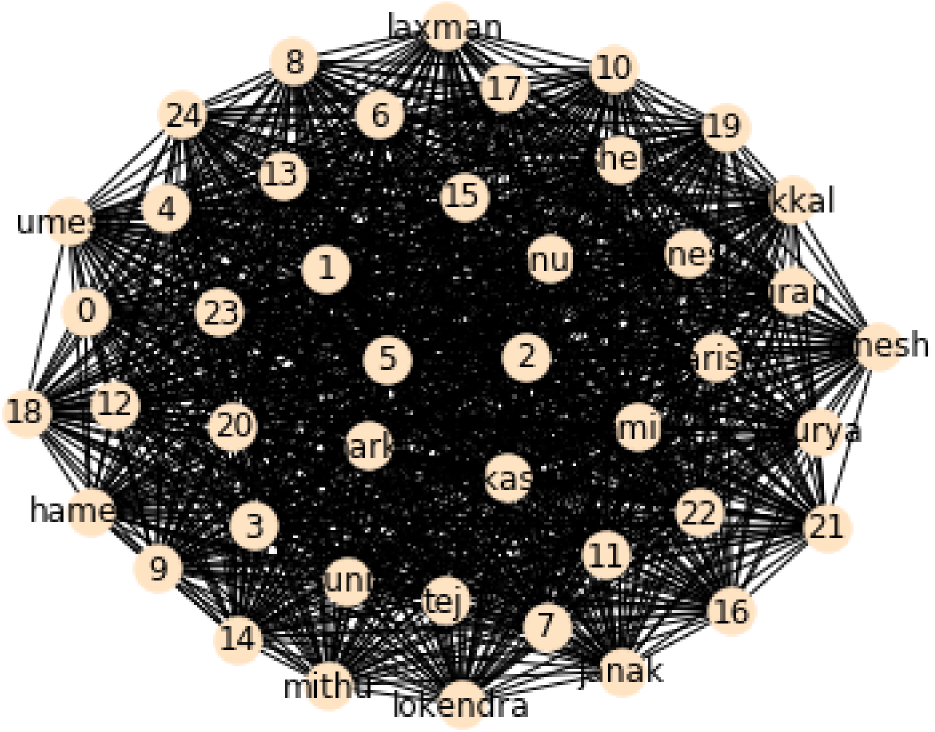
```
In [23]: G.add_nodes_from(['akkal','janak','laxman','bikash','dinesh','amin','sunil','kiran','ha
         nx.draw(G)
```



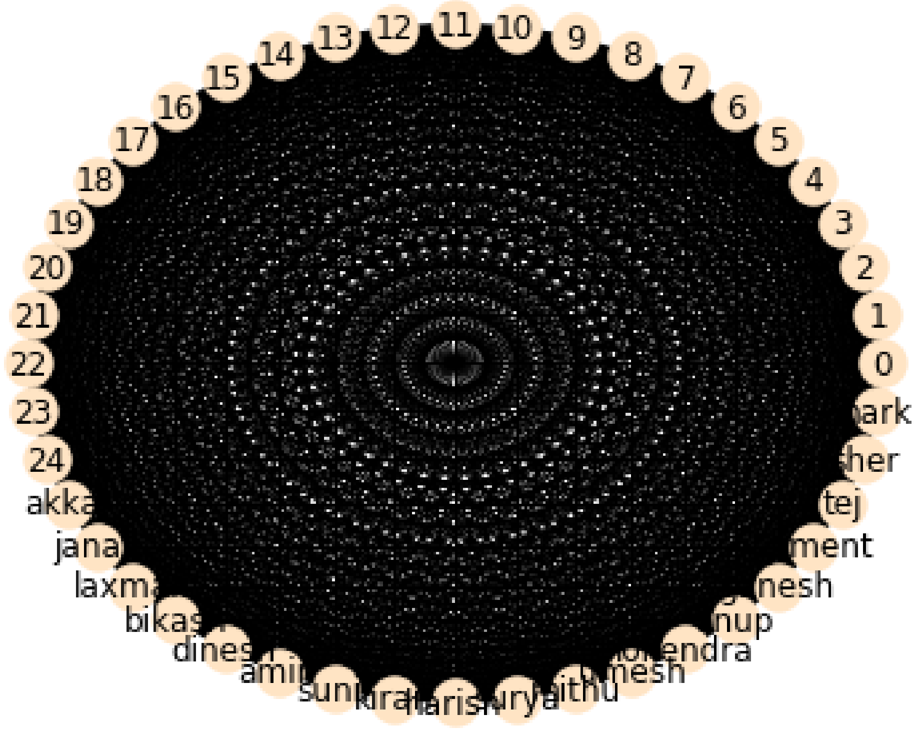
```
In [24]: nx.draw_circular(G)
```



```
In [25]: G = nx.complete_graph(G)
         nx.draw(G, node_color='bisque', with_labels=True)
```



```
In [26]: ego_G = nx.ego_graph(G, 5, radius=5)
         nx.draw_circular(G, node_color='bisque', with_labels=True)
```


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
akka
jana
laxm
bikas
dines
ami
sun
kira
haris
urya
aithu
mesh
endra
nup
nesh
ment
tej
her
ark