StevenGlasford_Final_Project_P1

April 25, 2020

[33]:
```
import scipy.io as sio
from scipy.sparse import csc_matrix as csc
import networkx as nx
import matplotlib.pyplot as plt
import numpy as np

mat_contents = sio.loadmat("Homo_sapiens.mat")
dense = mat_contents['network'].todense()
numberOfNodes = np.shape(dense)[0]
numberOfEdges = np.sum(np.tril(dense))
averageDegree = (np.sum(dense)+np.trace(dense))/np.shape(dense)[0]
numberOfSelfLoops = nx.number_of_selfloops(G)
#print(dense)

#G=nx.read_edgelist(dense)
G = nx.from_numpy_matrix(dense)


print(
    "Data from NetworkX\n"+
    nx.info(G) +

    "\n\nData from calculations from the adjacency list"+
    "\nIs the graph directed: "  + str(nx.is_directed(G)) +
    "\nNumber of nodes: " + str(numberOfNodes) +
    "\nNumber of edges: " + str(numberOfEdges) +
    "\nNumber of self loops: " + str(numberOfSelfLoops) +
    "\nAverage degree: " + str(averageDegree) +
    "\n\nThe information from NetworkX is the same as the \n" +
    "calulations on the adjacency matrix, so we can have a high\n" +
    "level of confidence that our network is correct."
)
```

```
Data from NetworkX
Name:
Type: Graph
```

```
Number of nodes: 3890
Number of edges: 38739
Average degree:  19.9172

Data from calculations from the adjacency list
Is the graph directed: False
Number of nodes: 3890
Number of edges: 38739.0
Number of self loops: 894
Average degree: 19.917223650385605

The information from NetworkX is the same as the
calulations on the adjacency matrix, so we can have a high
level of confidence that our network is correct.
```

[ ]:

[ ]: