notes

Name: Camille Scott
Email: camille.scott.w@gmail.com

May 3, 2016

```
In [2]: %pylab inline
```

```
Populating the interactive namespace from numpy and matplotlib
```

```
In [6]: import networkx as nx
        import pandas as pd
        import seaborn as sns
        import scipy as sp
        import scipy.stats as sps
```

```
In [7]: sns.set_style('white')
        sns.set_context('notebook')
```

0.1 Lecture 3

0.1.1 Power laws, Preferential Attachment, and master equations

0.1.2 April 5, 2016

Emergenceo of a Giant Component in Erdos-Renyi Random Graphs

```
In [8]: G = nx.random_graphs.erdos_renyi_graph(100, .1)
```

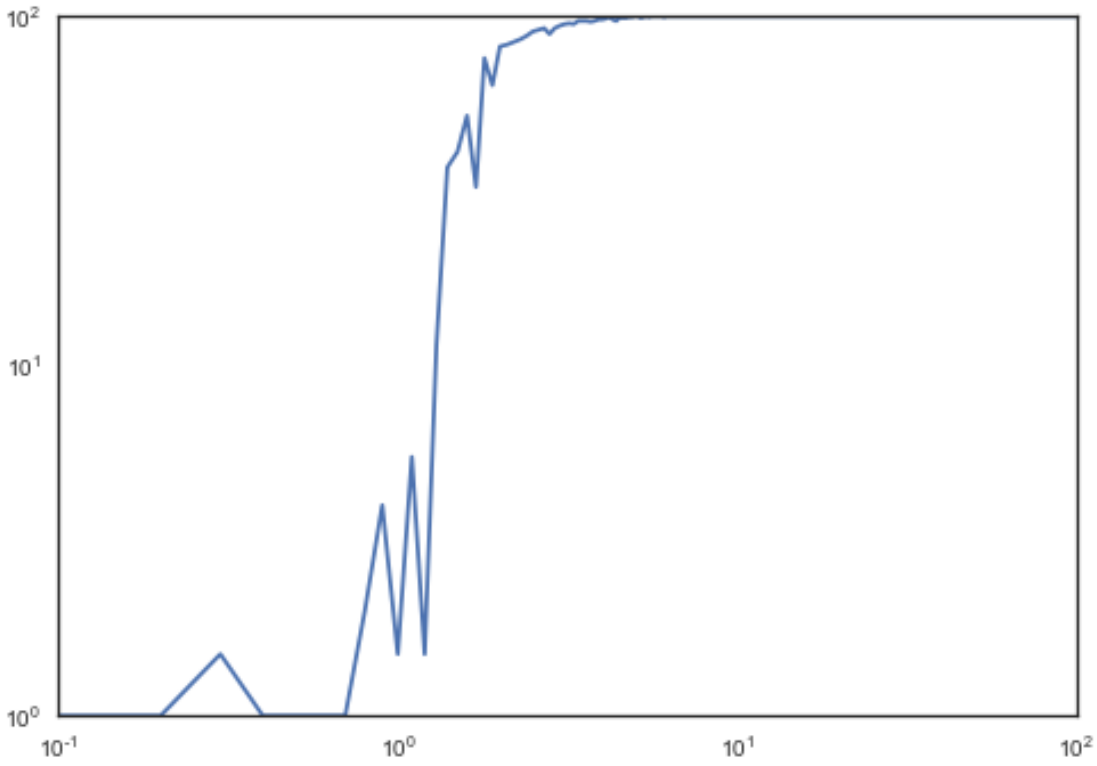
```
In [43]: def random_graph_maxc(N=100):
             P = np.arange(0,1,.001)
             graphs = [nx.random_graphs.erdos_renyi_graph(N, p) for p in P]
             c_max = [float(len(next(nx.connected_components(g)))) for g in graphs]

             return pd.Series(c_max, index=P*N)
```

```
In [45]: maxc_df = pd.DataFrame([random_graph_maxc() for _ in xrange(10)])
```

```
In [51]: maxc_df.median().plot(logx=True, logy=True)
```

```
Out[51]: <matplotlib.axes._subplots.AxesSubplot at 0x153d49c90>
```



Preferential Attachment

- $P_r(t+1 \to n_j) = d_j / \sum_i d_i$

In [ ]:

In [ ]:

In [ ]:

In [ ]: