

Project 2

Dr.Safayi
June 21, 2022

هدف ما در این پروژه بررسی احتمال ناهمبندی گراف شبکه در دو حالت ایستا و پویا و نشان دادن تاثیر الگوهای این دو حالت بر robustnes شبکه.

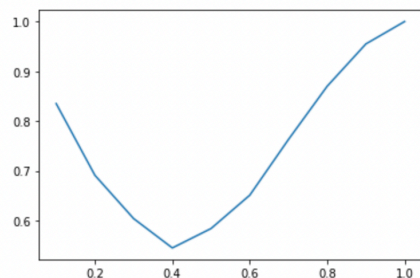
فاز ۱

طبق خواسته اول احتمالات کاهش پذیری گراف را برای هر نود در تابع Irreducible محاسبه کردیم و برای ۳ نوع گراف گفته شده نمودار کشیدیم و خطای شبیه سازی را نیز محاسبه کردیم.

Erdos-Renyi

```
In [126]: n = 5
p = 0.3
Ger = nx.erdos_renyi_graph(n,p)
prob = []
for pbad in node_p:
    prob.append(irreducible(Ger,n,pbad))
plt.plot(node_p,prob)
err = st.t.interval(alpha = 0.95, df = len(prob) - 1, loc = np.mean(prob), scale = st.sem(prob))
print('khataye shabih sazi va fasele etminan baraye motevassete an :',err)
```

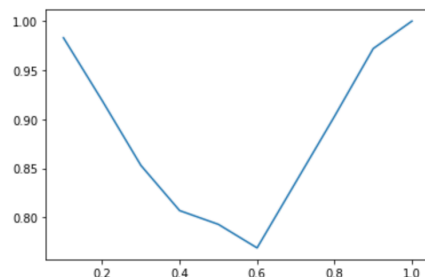
khataye shabih sazi va fasele etminan baraye motevassete an : (0.6353186549637071, 0.8642813450362927)



Barabasi-Albert

```
In [115]: n = 5
# m is the number of edges to attach from a new node to existing nodes
m = 2
Gba = nx.barabasi_albert_graph(n, m)
prob = []
for pbad in node_p:
    prob.append(irreducible(Gba,n,pbad))
plt.plot(node_p,prob)
err = st.t.interval(alpha = 0.95, df = len(prob) - 1, loc = np.mean(prob), scale = st.sem(prob))
print('khataye shabih sazi va fasele etminan baraye motevassete an :',err)
```

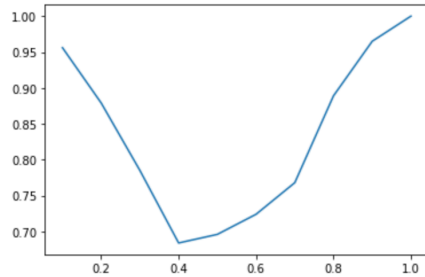
khataye shabih sazi va fasele etminan baraye motevassete an : (0.8235139595564124, 0.9434860404435877)



Watts–Strogatz

```
In [116]: n = 5
# k is : number of nodes connected to k nearest neighbors in ring topology
k = 2
# p is : the probability of rewiring each edge
p = 0.5
Gws = nx.watts_strogatz_graph(n, k, p)
prob = []
for pbad in node_p:
    prob.append(irreducible(Gws,n,pbad))
plt.plot(node_p,prob)
err = st.t.interval(alpha = 0.95, df = len(prob) - 1, loc = np.mean(prob), scale = st.sem(prob))
print('khataye shabih sazi va fasele etminan baraye motevassete an :',err)

khataye shabih sazi va fasele etminan baraye motevassete an : (0.750278060150335, 0.918921939849665)
```

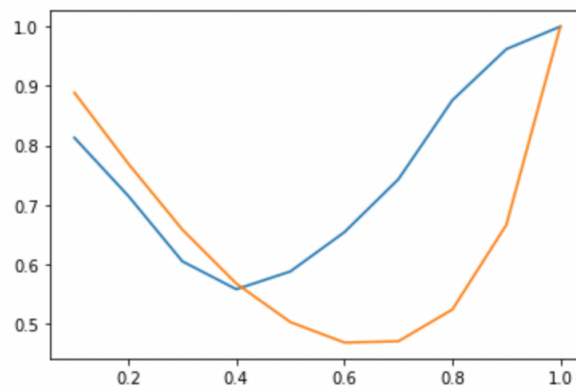


همچنین برای اثبات فرمول های گفته شده برای ۳ گراف گفته شده نمودار می کشیم :

$$P\{G \text{ is connected}\} \approx e^{-(1-p)\sum_{i \in V} p^{k_i}} \quad (۳)$$

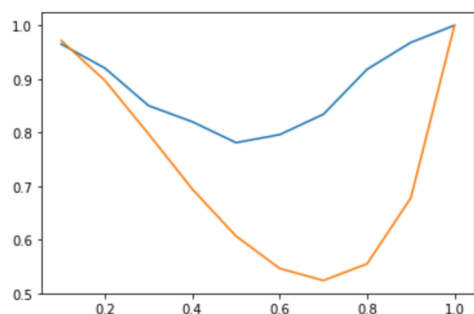
```
In [127]: Ger3Res = connection_prob(Ger)
prob = []
for pbad in node_p:
    prob.append(irreducible(Ger,n,pbad))
plt.plot(node_p,prob)
plt.plot(node_p, Ger3Res)

Out[127]: [<matplotlib.lines.Line2D at 0x7f960856e730>]
```



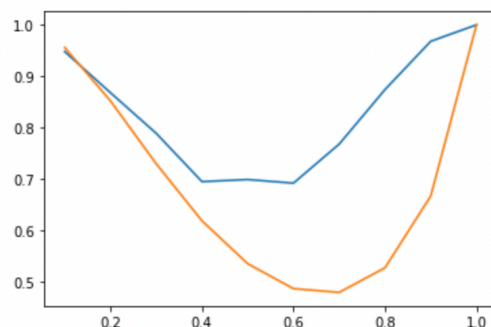
```
In [128]: Gba3Res = connection_prob(Gba)
prob = []
for pbad in node_p:
    prob.append(irreducible(Gba,n,pbad))
plt.plot(node_p,prob)
plt.plot(node_p, Gba3Res)
```

Out[128]: [<matplotlib.lines.Line2D at 0x7f96292d5400>]



```
In [129]: Gws3Res = connection_prob(Gws)
prob = []
for pbad in node_p:
    prob.append(irreducible(Gws,n,pbad))
plt.plot(node_p,prob)
plt.plot(node_p, Gws3Res)
```

Out[129]: [<matplotlib.lines.Line2D at 0x7f96293a81c0>]



همانطور که مشخص است شکل مشابهی دارند اما اختلاف دارند و این اختلاف به علت خطاست.

و همچنین برای قسمت فرمول ۶ که پیاده سازی شده و اختلاف آن ها که باز هم به علت خطا بوده نشان داده

شده است:

prove num6 formula

```
In [19]: def phi(graph, p):
n = graph.number_of_nodes()
result = 0
for i in range(1, n):
    ci = len(list(nx.all_node_cuts(graph, i)))
    result += ci * (p ** i) * ((1 - p) ** (n - i))

return result

def Rel(graph, p):
return 1 - phi(graph, p)
```

```
In [24]: prob6 = []
for pbad in node_p:
    prob6.append(Rel(Gba,pbad))
prob = []
for pbad in node_p:
    prob.append(irreducible(Gba,n,pbad))
plt.plot(node_p,prob)
plt.plot(node_p, prob6)
```

Out[24]: [<matplotlib.lines.Line2D at 0x7fa3c0c7be20>]

