

باسمه تعالی



## گزارش پروژه

نام و نام خانوادگی: علیرضا شکرانی و آرش دهقانی

پرسش تئوری ۰.۱ برای این کار باید بتواند  $m$  رابطه ی مشخص را درست حدس بزند که میشود

$$p^m (1 - p)^{n_e - m}$$

که  $n_e = \frac{n(n-1)}{2}$  این با این فرض است که برا ما مهم نباشد اگر دوستی های بیشتری به دست بیاید و بین دو کسی که دوستی وجود ندارد هم نباید دوستی برقرار کند.

پرسش تئوری ۰.۲ تعداد کل حالات  $\binom{n}{m}$  است پس احتمال برابر است با :

$$\frac{1}{\binom{n}{m}}$$

پرسش تئوری ۰.۳ فرض کنید  $k$  نزدیک ترین عدد صحیح کوچکتر از  $m/2$  باشد در این صورت  $k$  از  $m$  را باید درست حدس بزند و بقیه  $m - k$  تا را نباید دوستی به دست آورد. پس

$$\binom{m}{k} p^k (1 - p)^{m-k}$$

ن با این فرض است که برا ما مهم نباشد اگر دوستی های بیشتری به دست بیاید بلکه مهم آن است که ۲۰ درصد دوستی هایی را که وجود دارد، بیابد. پرسش شبیه سازی ۱ ابتدا با دستور

`nx.binomial_graph(n, p)`

در یک حلقه ده بار گراف رندوم میسازیم. سپس تعداد یال ها را می‌شماریم. چون هر یال هم ارز با یک دوستی است.

```
#1
n = 1000
p = 0.0034
s = 0
for i in range(0,10):
    M = nx.binomial_graph(n,p)
    s += M.number_of_edges() #number of friendships
print(s/10)
```

1707.5

شکل ۱ نتیجه در پایین تصویر

که با  $m$  تفاوت معنا داری دارد.  
 پرسش تئوری ۰۴. احتمال اینکه هر یال به وجود بیاید  $p$  است پس اگر  $X_{i,j}$  دوستی بین نفر  $i$  ام و  $j$  ام باشد، داریم:

$$\mathbb{E}[X_{i,j}] = p$$

طبق تعریف  $X_{i,j}$  برای  $s$  که تعداد یال های متاگراف است، داریم:

$$s = \frac{\sum_{i,j} X_{i,j}}{2} \Rightarrow \mathbb{E}[s] = \frac{\sum_{i,j} \mathbb{E}[X_{i,j}]}{2} = \frac{n(n-1)}{2} p$$

$$\Rightarrow \mathbb{E}[s] = \frac{n(n-1)}{2} p = \frac{1000(999)}{2} \cdot 0.00034 = 1698.3$$

که با مقدار شبیه سازی شده کمتر از ۱ درصد تفاوت دارد.  
 پس شرط تساوی به صورت زیر است:

$$m = \frac{n(n-1)}{2} p$$

۳

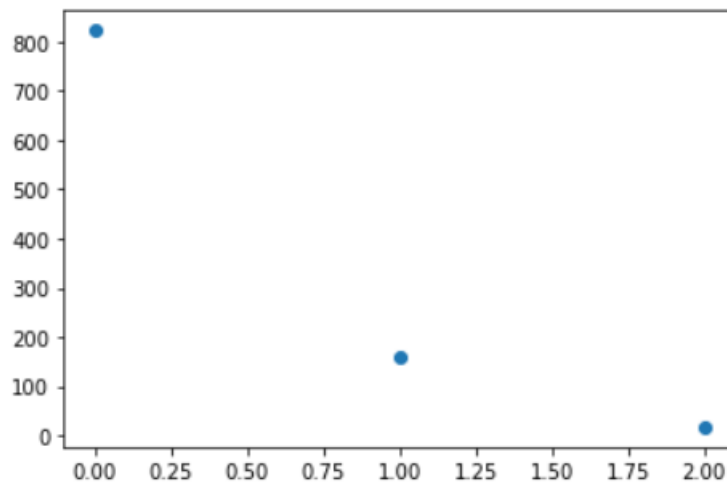
پرسش شبیه سازی ۰۲. ابتدا آرایه ای متشکل از ده گراف رندوم میسازیم سپس به ازای هر گراف رندوم، میانگین یال ها یا  $L$  را حساب کنیم. برای این کار کافی است تمام درجات را جمع و بر  $n$  تقسیم کنیم سپس در همان حلقه، تعداد گره ها با درجات بزرگتر از  $L$  را حساب میکنیم.

```
sum_social = 0
for M in graphs:
    social = 0
    # getting degrees in an array
    degrees = [val for (node,val) in M.degree()]
    L = sum(degrees)/n #mean of friendships
    #check being social
    for d in degrees:
        if(d > L):
            social+=1
    sum_social += social
print("mean of social = {}".format(sum_social/10))

mean of social = 144.2
```

شکل ۲ نتیجه در پایین تصویر

همچنین میانگین تعداد افرادی که  $i$  دوست دارند، بر حسب  $u$  برای گراف ها به شکل زیر است.



شکل ۳

پرسش تئوری ۵. دوباره با استفاده از تعریف  $X_{i,j}$  برای  $Y_i$  که تعداد دوست های راس  $i$  ام باشد، داریم:

$$Y_i = \sum_{j=1}^{n-1} X_{i,j} \Rightarrow \mathbb{E}[Y_i] = \sum_{j=1}^{n-1} \mathbb{E}[X_{i,j}]$$

$$\Rightarrow \mathbb{E}[Y_i] = (n-1)p = 999(0.00016) = 0.15984$$

پرسش تئوری ۶. مجموع روابط دوستی (مجموع یال ها) توزیع  $Binomial(\frac{n(n-1)}{2}, p)$  دارد. اگر مجموع یال ها را  $K$  بگیریم. برای  $L$  داریم:

$$\mathbb{P}[L = l] = \mathbb{P}\left[\frac{2K}{n} = l\right] = \mathbb{P}\left[K = \frac{nl}{2}\right] = \binom{\frac{n(n-1)}{2}}{\frac{nl}{2}} p^{\frac{nl}{2}} (1-p)^{\frac{n(n-1)}{2} - \frac{nl}{2}}$$

که این تابع در نزدیکی میانگین یعنی ۰/۱۵۹۸۴ (از بخش قبل) بیشینه میشود. پس از آن به طور نزولی افت میکند و به صفر میرود. از شبیه سازی هم میتوان نتیجه گرفت که به علت کوچک بودن  $np$  عدد  $L$  تقریباً یک عددی بین ۰ و ۱ است. به عبارت دیگر احتمال اینکه این عدد مساوی ۱ یا بزرگتر از ۱ باشد بسیار کوچک است. حال با فرض اینکه  $L$  کوچکتر از یک است، هر شخصی که یک دوست یا بیشتر داشته باشد، اجتماعی خواهد بود. یعنی کافست احتمال متمم پیشامد یعنی نداشتن دوست را از ۱ کم کنیم.

$$\mathbb{P}_{social} \simeq 1 - \mathbb{P}[\text{نداشتن دوست}] = 1 - (1-p)^{n-1} \simeq 0.1477$$

متغیر  $Y_i$  را متغیر تصادفی برنولی برای اجتماعی بودن نفر  $i$  ام در نظر بگیرید. و  $S$  را تعداد افراد اجتماعی

فرض کنید. داریم:

$$S = \sum_{i=1}^n Y_i \Rightarrow \mathbb{E}[S] = \sum_{i=1}^n \mathbb{E}[Y_i] = n P_{social} = 1000(0.14777) \simeq 147.8$$

که با شبیه سازی ۲ (شکل ۲) تنها تفاوت اندکی دارد.

۴

پرسش شبیه سازی ۳. برای روابط تراگذاری باید تعداد مثلث ها را بشماریم. برای این کار از دستور زیر استفاده میکنیم:

```
networkx.triangles(g)
```

که خروجی آن یک dictionary است با key هایی از جنس node و value هایی که تعداد مثلث مرتبط با هر گره را نشان میدهند. پس کافی است value های آن را جمع بزنیم و بر ۳ تقسیم کنیم.

```
#finding triangles
sum_triangles = 0
for g in graphs:
    triangles = nx.triangles(g)
    t = sum(triangles.values())/3    #sum of all nodes triangles divided by 3
    sum_triangles += t
print("mean of triangles => {}".format(sum_triangles/5))
```

```
mean of triangles => 4486.2
```

شکل ۴ نتیجه در پایین تصویر

تابعی برای پیدا کردن رقابت مینویسیم. این تابع ابتدا یک حلقه روی هر گره دارد سپس با دو حلقه دیگر روی همسایه ها آن گره، چک میکند که آیا دو گره همسایه به هم متصل هستند یا خیر. در صورت عدم اتصال ما یک رابطه رقابت خواهیم داشت:

```

#check connections
def nodes_connected(g,x,y):
    return x in g.neighbors(y)

#function to find competitions

def number_of_competitions(g):
    competitions = 0
    for node in g.nodes:
        friends = list(g.neighbors(node))
        length = sum(1 for _ in g.neighbors(node))
        for i in range(length):
            for j in range(i+1, length):
                if(not nodes_connected(g,friends[i],friends[j])):
                    competitions += 1
    return competitions

```

شکل ۵ تابع بالا برای چک کردن اتصال

این را برای هر گراف اعمال خواهیم کرد.

```

#finding mean competitions for 5 graphs

sum_competitions = 0
for g in graphs:
    sum_competitions += number_of_competitions(g)

print("number of competitions is => {}".format(sum_competitions/5))

number of competitions is => 1325543.4

```

شکل ۶ نتیجه در پایین تصویر

پرسش شبیه سازی ۴. حال برای تراگذاری متاگرافی با  $n = 2000$  و  $p = 0.2$  کد بخش قبل را اجرا میکنیم. با این اختلاف که تعداد مثلثات را در یک آرایه برای بخش بعد ذخیره میکنیم.

```
#finding triangles
triangles_arr = []

for g in graphs:
    triangles = nx.triangles(g)
    triangles = sum(triangles.values())/3    #sum of all nodes triangles divided by 3
    triangles_arr.append(triangles)
    print(triangles)
print("mean of triangles => {}".format(np.mean(triangles_arr)))
```

10662929.0  
10696804.0  
10580009.0  
10629075.0  
10598329.0  
mean of triangles => 10633429.2

شکل ۷ نتیجه برای هر گراف و میانگین در پایین تصویر

برای میانگین رابطه ی دوستی رفاقت در این بخش از الگوریتمی متفاوت استفاده میکنیم تا مرتبه زمانی اجرا شدن کد پایین بیاید.

برای هر راس تعداد مسیر دوتایی از یکی از راس های همسایه به دیگر مسیر همسایه اش را میشماریم. اگر  $d_i$  درجه راس  $i$  ام باشد، تعداد این مسیر ها  $d_i(d_i - 1)$  خواهد بود. بعضی از این مسیر ها بخشی از یک حلقه مثلثی خواهند بود و هر مثلث سه بار شمرده خواهد شد. به همین دلیل باید سه برابر تعداد مثلث هایی (که در کد تراگذاری به دست آوردیم) از تعداد مسیر ها کم کنیم تا تعداد روابط رقابتی به دست آید.

```

#finding mean competitions for 5 graphs
sum_paths = 0
sum_competitions = 0
i= 0
for g in graphs:
    sum_paths = 0
    for node in g.nodes:
        degree = g.degree(node)
        paths = degree * (degree - 1)/2
        sum_paths += paths
    competitions = sum_paths - triangles_arr[i] * 3 #every triangle path is counted 3 times
    sum_competitions += competitions
    i+=1

print("number of competitions is => {}".format(sum_competitions/5))

```

number of competitions is => 127685550.2

شکل ۸ نتیجه در پایین تصویر

پرسش تئوری ۷.  
راه شهودی:

ابتدا احتمال اینکه یک سه تایی (سه راس) تشکیل مثلث بدهد، پیدا میکنیم.  $(p^3)$  این احتمال در واقع به طور میانگین نسبت تعداد مثلثات به تعداد کل سه تایی هاست:

$$\binom{n}{3} p^3$$

راه ریاضی:

برای این بخش متوسط تعداد مثلث ها در یک گراف را میابیم. متغیر تصادفی  $X_{u,v,w}$  را اینگونه تعریف میکنیم که اگر تشکیل مثلث دادند، مساوی یک است در غیر این صورت مساوی صفر.

$$\mathbb{P}[X_{u,v,w} = 1] = p^3$$

چون متغیر برنولی است، داریم :

$$\mathbb{E}[X_{u,v,w}] = \mathbb{P}[X_{u,v,w} = 1] = p^3$$

حال تعداد مثلث ها را برابر  $T$  بگیریم. داریم:

$$T = \sum_{u,v,w} X_{u,v,w} \Rightarrow \mathbb{E}[T] = \sum_{u,v,w} \mathbb{E}[X_{u,v,w}]$$



نعداد کل سه تایی های نامرتب برابر با  $\binom{n}{3}$  است. پس میانگین روابط تراگذاری برابر است با :

$$\mathbb{E}[T] = \binom{n}{3} p^3$$

حال سراغ رابطه دوستی رقابت میرویم.  
راه شهودی:

احتمال اینکه یک سه تایی تشکیل رقابت دهند  $3p^2(1-p)$  است. چون سه راه برای انتخاب راس مشترک وجود دارد این احتمال در واقع به طور میانگین نسبت تعداد رقابت ها به تعداد کل سه تایی هاست:

$$3 \binom{n}{3} p^2 (1-p)$$

راه ریاضی:

متغیر تصادفی برنولی  $Y_{u,v,w}$  را این گونه تعریف میکنیم که اگر این سه تایی تشکیل رابطه ی رقابت بدهند، برابر ۱ است. در غیر این صورت برابر صفر است.  
برای تشکیل رابطه رقابت باید میان دو راس یال نباشد و میان آن دو راس و راس سوم یک یال وجود داشته باشد. پس به سه طریق قابل انجام است.

$$\mathbb{P}[Y_{u,v,w} = 1] = 3p^2(1-p)$$

چون متغیر برنولی است، داریم :

$$\mathbb{E}[Y_{u,v,w}] = \mathbb{P}[Y_{u,v,w} = 1] = 3p^2(1-p)$$

حال تعداد رقابت ها را برابر  $R$  بگیرد. داریم:

$$R = \sum_{u,v,w} Y_{u,v,w} \Rightarrow \mathbb{E}[R] = \sum_{u,v,w} \mathbb{E}[Y_{u,v,w}]$$

نعداد کل سه تایی های نامرتب برابر با  $\binom{n}{3}$  است. پس میانگین تعداد روابط رقابتی برابر است با :

$$\mathbb{E}[R] = 3 \binom{n}{3} p^2 (1-p)$$

پرسش تئوری ۸. برای محاسبه این کسر (بگیرید  $\eta$ )، احتمال این را که ۳ راس  $u, v, w$  تشکیل مثلث بدهند به شرط اینکه هر کدام حداقل به یک راس وصل باشند حساب میکنیم. پیشامد اتصال هر کدام از این سه راس به حداقل یک راس از سه راس را  $B$  بگیرد.

$$\eta = \mathbb{P}[\text{تراگذاری} | B] = \frac{\mathbb{P}[\text{تراگذاری} \cap B]}{\mathbb{P}[B]}$$

بنابر تعریف واضح است که  $B \subseteq$  (پیشامد تراگذاری). همچنین پیشامد تراگذاری و رقابت با هم ناسازگار بوده و  $B$  همان اجتماعشان است.

$$\Rightarrow \mathbb{P}[B] = \mathbb{P}[\text{تراگذاری}] + \mathbb{P}[\text{رقابت}]$$

$$\Rightarrow \mathbb{P}[B|\text{تراگذاری}] = \frac{\mathbb{P}[\text{تراگذاری}]}{\mathbb{P}[\text{تراگذاری}] + \mathbb{P}[\text{رقابت}]} = \frac{p^2}{p^2 + p^2(1-p)}$$

$$\eta = \frac{p}{2 - 2p}$$

که این همان کسر خواسته شده است. برای  $p = 0.2$  داریم  $\eta \cong 0.07692$  که برابر با کسر زیر است:

$$\frac{n_{\text{تراگذاری}}}{n_{\text{تراگذاری}} + n_{\text{رقابت}}} \cong 0.076876$$

برای  $p = 0.00016$  هم نتایج تئوری و شبیه سازی یکسان است. پرسش شبیه سازی ۵. برای هر گره وصل بودن همسایه های گره را در دو حلقه بررسی میکنیم.

```
for person in g.nodes:
    total = 0
    friends = list(g.neighbors(person))
    degree = g.degree[person]
    for i in range(degree):
        for j in range(i+1, degree):
            if nodes_connected(g, friends[i], friends[j]):
                total += 1
    totals.append(total)
print("mean of friendships => {}".format(sum(totals)/n))
```

mean of friendships => 0.012

شکل ۹ نتیجه در پایین تصویر

پرسش تئوری ۹. در مثال واقعی همان طور که از شکل صورت سوال نیز پیداست، روابط به صورت خوشه ای است به این معنا که چند گروه دوستی یا چند جامعه ی کوچکتر وجود دارد که در آن همه ی راس ها با هم در ارتباط هستند. تعدادی کمی یال هم بعضی راس های خوشه ها را به راس های خوشه های دیگر مربوط میکند. این به این معناست که مثلا در یک خوشه  $n_i$  تایی، تعداد دوستی های بین دوست های یک شخص در مرتبه  $n_i^2$  است (چون هر شخص  $n_i - 1$  تا دوست دارد و هر یک از دوستان او با  $n_i - 1$  نفر از این گروه دوستی دوست است پس دوستی های اطراف آن  $(n_i - 1)^2$  است). پس میانگین آن به مقدار قابل توجه بزرگتری از یک است. برخلاف شبیه سازی ۵ که کوچک تر از یک است. بنابراین در مثال واقعی این عدد بیشتر است.

پرسش تئوری ۱۰. فرض کنید  $X_i$  تعداد دوستی های بین دوستان شخص  $i$  ام و  $K_i$  تعداد دوست های فرد  $i$  ام باشد. ابتدا رابطه زیر را برای امید ریاضی داریم:

$$\mathbb{E}[X_i] = \mathbb{E}[\mathbb{E}[X|K_i = k]]$$

$X_i$  به شرط  $K_i = k$  توزیع  $Binomial(\frac{k(k-1)}{2}, p)$  دارد پس میانگین  $X_i$  برابر با  $\frac{k(k-1)}{2}p$  خواهد بود.

$$\mathbb{E}[X_i] = \mathbb{E}[\frac{k(k-1)}{2}p]$$

$K_i$  هم توزیع  $Binomial(n-1, p)$  دارد.

$$\begin{aligned} \sum_{k=1}^{n-1} k(k-1) \binom{n-1}{k} p^k (1-p)^{n-k-1} &= \sum_{k=1}^{n-1} \frac{(n-1)(n-2)(n-3)!}{(k-2)!(n-1-k)!} p^2 p^{k-2} (1-p)^{n-k-1} \\ &= (n-1)(n-2)p^2 \sum_{k=1}^{n-1} \binom{n-3}{k-2} p^{k-2} (1-p)^{n-k-1} \end{aligned}$$

که عبارت جلوی  $\sum$  همان باینومیل است و تمامی مقادیر باینومیل را جمع میکند. طبق تعریف تابع توزیع جمع آن یک است.

$$\begin{aligned} \mathbb{E}[k(k-1)] &= (n-1)(n-2)p^2 \\ \Rightarrow \mathbb{E}[X_i] &= \frac{(n-1)(n-2)p^2}{2} \end{aligned}$$

که برای این سوال مشود  $0.01346$  که نزدیک شبیه سازی ۵ است.

۵

پرسش شبیه سازی ۰۶. برای این بخش، ابتدا برای هر گره مینیموم فاصله ی آن گره تا گره های دیگر(که به آن راه دارد) را میابیم. سپس روی همین مقادیر میانگین گیری میکنیم و میانگین فاصله برای هر راس را حساب میکنیم. چک کردن اینکه گره person به node راه دارد و اینکه فاصله این دو چقدر است، به ترتیب با دستورات زیر صورت میگیرد:

```
networkx.shortest_paths.has_path(g, person, node)
networkx.shortest_paths.generic.shortest_path_length(g, person, node)
```

پس از این، روی اعداد به دست آمده از همه گره ها میانگین گیری میکنیم.(گره هایی که ارتباط ندارند یا گره های ایزوله اثری در میانگین ندارند)

```
# 6

n = 1000
p = 0.0033
g = nx.fast_gnp_random_graph(n, p)

averages = []
for person in g.nodes:
    if g.degree[person] != 0:
        total = 0
        num_of_paths = 0
        for node in g.nodes:
            if nx.algorithms.shortest_paths.has_path(g, person, node) and person != node:
                total += nx.algorithms.shortest_paths.generic.shortest_path_length(g, person, node)
                num_of_paths += 1
        averages.append(total/num_of_paths)

print("average distance => {}".format(np.mean(averages)))

average distance => 5.7213916566200975
```

شکل ۱۰ نتیجه در پایین تصویر

پرسش شبیه سازی ۰۷. ابتدا برای هر راس دورترین راس را پیدا کرده و فاصله آن را حساب میکنیم سپس مقدار بیشینه این فاصله را میان تمامی رئوس پیدا میکنیم.

```

#7
#function to find max distance in graph
def max_distance(g):
    mx = 0
    nodes = list(g.nodes)
    length = len(nodes)
    for i in range(length):
        for j in range(i+1, length):
            if nx.has_path(g, nodes[i], nodes[j]):
                tmp = nx.shortest_path_length(g, nodes[i], nodes[j])
                if tmp > mx:
                    mx = tmp
    return mx

#main code
n = 50
p = 0.34
graphs = []
for i in range(100):
    graphs.append(nx.fast_gnp_random_graph(n,p))
maxes = []
for g in graphs:
    max = max_distance(g)
    maxes.append(max)

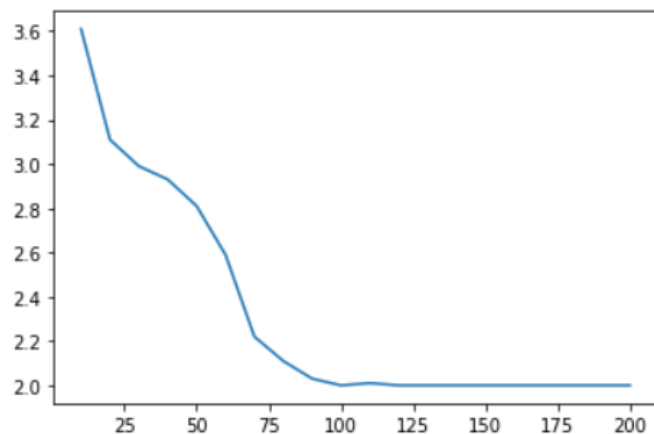
print("mean of max distance => {}".format(np.mean(maxes)))

```

mean of max distance => 2.79

شکل ۱۱ نتیجه در پایین تصویر، تابع بالا برای پیدا کردن قطر گراف

پرسش شبیه سازی ۰.۸ حال برای  $n$  از ۱۰ تا ۲۰۰ با گام ۲۰ شبیه سازی را تکرار میکنیم.



شکل ۱۲ نمودار

این نمودار در بینهایت به عدد ۲ میل میکند.  
 پرسش تئوری ۱۱. احتمال اینکه راسی مانند  $w$  همسایه مشترک  $u$  و  $v$  باشد، هم ارز است با احتمال اینکه هر دو به  $w$  وصل باشند. این احتمال برابر است با  $p^2$  و خواسته مساله متمم این است پس برای همه  $n - 2$  راس دیگر داریم:

$$\mathbb{P}[I_{u,v} = 1] = (1 - p^2)^{n-2}$$

پرسش تئوری ۱۲. طبق تعریف  $I_{u,v}$  داریم:

$$X = \sum_{u,v} I_{u,v} \Rightarrow \mathbb{E}[X] = \sum_{u,v} \mathbb{E}[I_{u,v}]$$

که چون متغیر برنولی است، داریم:  $\mathbb{E}[I_{u,v}] = \mathbb{P}[I_{u,v} = 1]$  پس:

$$\mathbb{E}[X_n] = \binom{n}{2} (1 - p^2)^{n-2}$$

پرسش تئوری ۱۳. با استفاده از نامساوی مارکف (چون متغیر تصادفی  $X_n$  مثبت است) داریم:

$$\mathbb{P}[X_n \geq 1] \leq \frac{\mathbb{E}[X_n]}{1}$$

$$\Rightarrow \mathbb{P}[X_n \geq 1] \leq \binom{n}{2} (1 - p^2)^{n-2}$$

حال با میل دادن  $n$  به بینهایت داریم: (با توجه به رابطه  $x = e^{\ln(x)}$ )

$$\lim_{n \rightarrow \infty} \binom{n}{2} (1 - p^2)^{n-2} = \lim_{n \rightarrow \infty} \frac{n(n-1)}{2} e^{\ln(1-p^2)(n-2)}$$

با توجه به اینکه  $1 - p^2 \leq 1$  پس توان نمایی به منفی بینهایت میرود. رشد و افت تابع نمایی از چندجمله ای بیشتر است پس

$$\lim_{n \rightarrow \infty} \binom{n}{2} (1 - p^2)^{n-2} = 0$$

$$\Rightarrow \lim_{n \rightarrow \infty} \mathbb{P}[X_n \geq 1] \leq 0 \Rightarrow \lim_{n \rightarrow \infty} \mathbb{P}[X_n \geq 1] = 0$$

پرسش تئوری ۰.۱۴. در پرسش قبل اثبات کردیم که وقتی  $n$  به بینهایت میل میکند،  $\mathbb{P}[X_n \geq 1]$  فارغ از مقدار  $P$ ، به صفر میل میکند. به عبارت دیگر در  $n$  های بسیار بزرگ تعداد راس هایی که هیچ همسایه مشترکی ندارند، با احتمال بالا (فارغ از مقدار  $p$ ) به صفر میل میکند و هر دو راس حداقل یک همسایه مشترک خواهند داشت. پس بیشترین فاصله بین دو راس ۲ خواهد بود. این با شبیه سازی ۸ تطابق کامل دارد.

۶

پرسش شبیه سازی ۰.۹. دوباره از کد پرسش ۳ قسمت تراگذاری بهره میبریم.

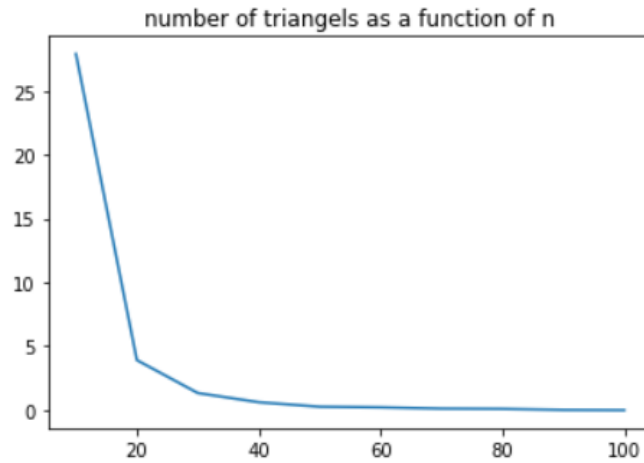
```
# 9

n = 100
p = 0.34
graphs = []
for i in range(100):
    graphs.append(nx.fast_gnp_random_graph(n, p))
triangles = []
for g in graphs:
    triangles.append(sum(nx.triangles(g).values())/3)
print("triangles => {}".format(sum(triangles)/100))

triangles => 6350.37
```

شکل ۱۳ نتیجه در پایین تصویر

پرسش شبیه سازی ۰.۱۰. این بار در بازه ی ۱۰ تا ۱۰۰ با گام ۱۰ و با  $p = \frac{c_0}{n}$  نمودار را میکشیم.

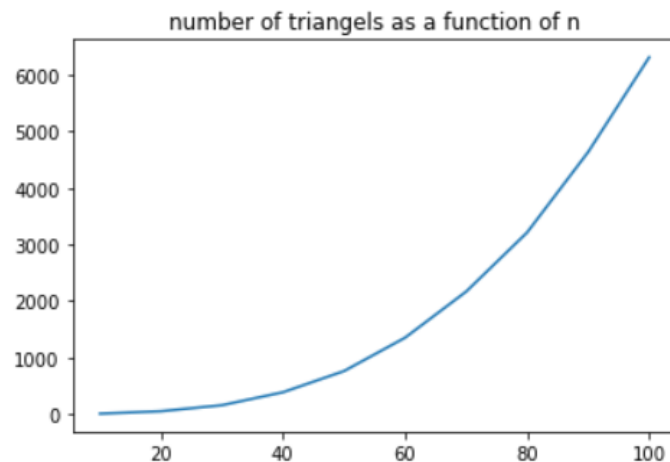


شکل ۱۴ نمودار به ازای  $p = \frac{60}{n^3}$

در بینهایت به صفر میل میکنند که این موضوع با تئوری هم خوانی دارد. همان طور که از تئوری ۷ دیدیم و در تئوری ۱۶ هم خواهیم دید، میانگین تعداد مثلث ها به صورت زیر است:  $p^3 \binom{n}{3}$  حال با جایگذاری و در  $n$  های بزرگ خواهیم داشت:

$$\frac{n(n-1)(n-2)}{6} \frac{60^3}{n^6} \approx \frac{60^3 n^3}{6n^6} \approx 0$$

پرسش شبیه سازی ۰.۱۱ این بار به ازای  $p = 0.34$  شبیه سازی بخش قبل را تکرار میکنیم.



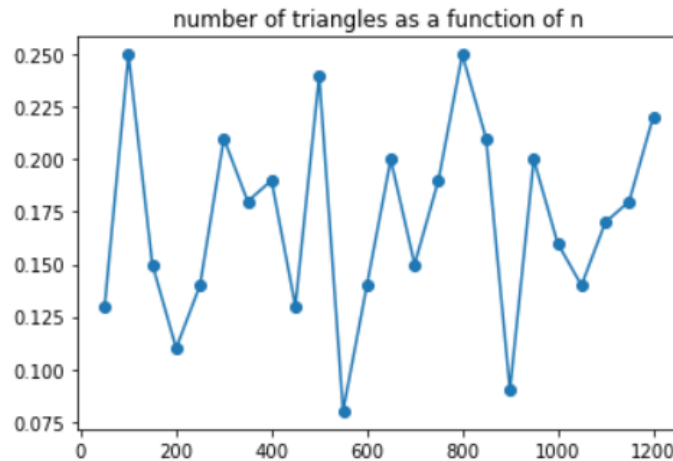
شکل ۱۵ نمودار به ازای  $p = 0.34$



در  $n$  های بزرگ به بینهایت میرود و این طبق تئوری منطقی است.

$$\binom{n}{3} p^3 \approx \frac{n^3}{6} \cdot \frac{1}{n^3} \sim n^3$$

که به مقدار متناهی میل نخواهد کرد.  
پرسش شبیه سازی ۱۲. حال به ازای  $p = \frac{1}{n}$  شبیه سازی را انجام میدهم.



شکل ۱۶ نمودار به ازای  $p = \frac{1}{n}$

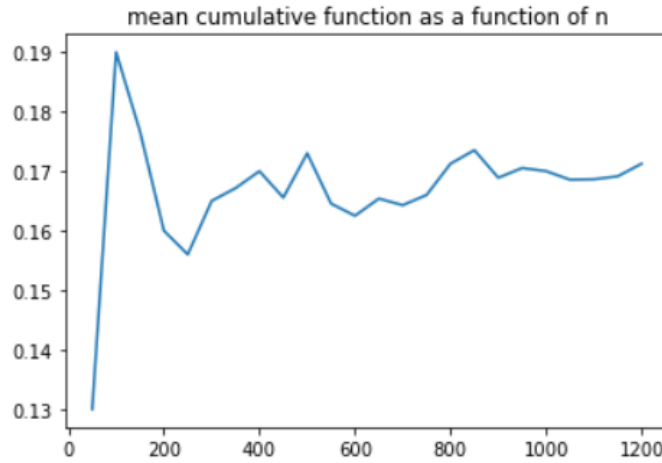
که حالت نوسانی دارد. این نمودار به علت مقادیر داده شده با تئوری ناسازگار است. طبق تئوری بایستی صعودی باشد و به یک مقدار ثابت میل کند.

$$\binom{n}{3} p^3 \approx \frac{n^3}{6} \left(\frac{1}{n}\right)^3 = \frac{1}{6}$$

حال با استفاده از دستور زیر داده های تجمعی را برای بردار داده های بخش قبل حساب میکنیم.

`networkx.cumsum(y)`

سپس در یک حلقه با تقسیم بر  $i + 1$  میانگین ها را حساب میکنیم که  $i$  اندیس آن داده داخل آرایه داده هاست.



شکل ۱۷ کد در بالای تصویر و نمودار تجمعی در پایین تصویر

که به دلیل ناسازگاری نمودار قبلی با تئوری، این هم تا حدی ناسازگاری دارد. پرسش تئوری ۰۱۵. بایستی میان هر جفت از آن ها یک یال با احتمال  $p$  باشد. پس:

$$\mathbb{P}[I_{u,v,w} = 1] = p^3$$

پرسش تئوری ۰۱۶. با استفاده از تعریف  $I_{u,v,w}$  داریم:

$$T_{r,n} = \sum_{u,v,w} I_{u,v,w} \Rightarrow \mathbb{E}[T_{r,n}] = \sum_{u,v,w} \mathbb{E}[I_{u,v,w}]$$

$I_{u,v,w}$  متغیر برنولی است پس  $\mathbb{E}[I_{u,v,w}] = \mathbb{P}[I_{u,v,w} = 1]$  و تعداد سه تایی  $(u, v, w)$  هم برابر است با  $\binom{n}{3}$  پس:

$$\mathbb{E}[T_{r,n}] = \binom{n}{3} p^3$$

پرسش تئوری ۰۱۷. طبق نامساوی مارکف داریم:

$$\mathbb{P}[T_{r,n} \geq 1] \leq \frac{\mathbb{E}[T_{r,n}]}{1} \Rightarrow \mathbb{P}[T_{r,n} \geq 1] \leq \binom{n}{3} p^3$$

$$\Rightarrow \mathbb{P}[T_{r,n} \geq 1] \leq \frac{n(n-1)(n-2)}{3! n^3}$$

حال با میل دادن  $n$  به بینهایت داریم:

$$\lim_{n \rightarrow \infty} \mathbb{P}[T_{r,n} \geq 1] \leq \lim_{n \rightarrow \infty} \frac{n(n-1)(n-2)}{3! n^3}$$

$$\lim_{n \rightarrow \infty} \mathbb{P}[T_{r,n} \geq 1] \leq 0 \Rightarrow \lim_{n \rightarrow \infty} \mathbb{P}[T_{r,n} \geq 1] = 0$$

پرسش تئوری ۱۸. با توجه به تعریف  $Y_n$  از شرط دوم داریم: (بگیرید:  $\mathbb{P}[Y_n = i] = P_n(i)$ )

$$\lim_{n \rightarrow \infty} \frac{\text{Var}[Y_n]}{\mathbb{E}^r[Y_n]} = 0 \Rightarrow \lim_{n \rightarrow \infty} \frac{\sum_{i=0}^{\infty} (i - \mathbb{E}[Y_n])^r}{\mathbb{E}^r[Y_n]} P_n(i) = 0$$

با توجه به خواص حد داریم:

$$\Rightarrow \sum_{i=0}^{\infty} \left( \lim_{n \rightarrow \infty} \frac{(i - \mathbb{E}[Y_n])^r}{\mathbb{E}^r[Y_n]} P_n(i) \right) = 0$$

چون همه ی جملات داخل جمع نامنفی هستند و مساوی صفر شدند، پس همه جملات صفر است. من جمله در  $i = 0$  در  $i = 0$  خواهیم داشت:

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{(\mathbb{E}[Y_n])^r}{\mathbb{E}^r[Y_n]} P_n(0) &= 0 \\ \Rightarrow \lim_{n \rightarrow \infty} P_n(0) &= 0 \Rightarrow \lim_{n \rightarrow \infty} \mathbb{P}[Y_n \geq 1] = 1 \end{aligned}$$

پرسش تئوری ۱۹. با توجه به اینکه  $I_{u,v,w}$  متغیر تصادفی برنولی هست، داریم:

$$\begin{aligned} \mathbb{E}[I_{u,v,w} I_{u',v',w'}] &= \mathbb{P}[I_{u,v,w} = 1, I_{u',v',w'} = 1](1)(1) + \mathbb{P}[I_{u,v,w} = 0, I_{u',v',w'} = 1](0)(1) \\ &+ \mathbb{P}[I_{u,v,w} = 1, I_{u',v',w'} = 0](1)(0) + \mathbb{P}[I_{u,v,w} = 0, I_{u',v',w'} = 0](0)(0) \\ \Rightarrow \mathbb{E}[I_{u,v,w} I_{u',v',w'}] &= \mathbb{P}[I_{u,v,w} = 1, I_{u',v',w'} = 1] \end{aligned}$$

سه حالت برای  $u, v, w, u', v', w'$  وجود دارد.

- هیچ یالی بین دو مثلث مشترک نباشد (هیچ یا یک راس مشترک)  
به ۶ یال نیاز داریم:

$$\mathbb{P}[I_{u,v,w} = 1, I_{u',v',w'} = 1] = p^6 \Rightarrow \mathbb{E}[I_{u,v,w} I_{u',v',w'}] = p^6$$

- یک یال مشترک باشد (دو راس مشترک)  
به ۵ یال نیاز داریم:

$$\mathbb{P}[I_{u,v,w} = 1, I_{u',v',w'} = 1] = p^5 \Rightarrow \mathbb{E}[I_{u,v,w} I_{u',v',w'}] = p^5$$

- سه یال مشترک (سه راس مشترک)  
به ۳ یال نیاز داریم:

$$\mathbb{P}[I_{u,v,w} = 1, I_{u',v',w'} = 1] = p^3 \Rightarrow \mathbb{E}[I_{u,v,w} I_{u',v',w'}] = p^3$$

پرسش تئوری ۲۰. با توجه به رابطه  $T_{r,n} = \sum_{u,v,w} I_{u,v,w}$  داریم:

$$\mathbb{E}[T_{r,n}^r] = \mathbb{E}\left[\left(\sum_{u,v,w} I_{u,v,w}\right)^r\right] = \sum_{u,v,w,u',v',w'} \mathbb{E}[I_{u,v,w} I_{u',v',w'}] \quad (20.1)$$

که حالت‌های مختلف میانگین آن را در بخش قبل حساب کردیم. حال سراغ تعداد حالت‌های مختلف آن می‌رویم:

• تعداد زوج مرتب مثلث‌هایی که با هم هیچ راس مشترکی ندارند یا یک راس مشترک دارند

$$\binom{n}{3} \binom{n-3}{3} + \binom{n}{3} (3) \binom{n-3}{2} \approx \left(\frac{n^3}{6}\right)^2 + \frac{n^3}{6} (3) \left(\frac{n^2}{2}\right) \approx \frac{n^6}{36}$$

که ۳ در جمله دوم تعداد راه‌های انتخاب راس‌های مشترک است

• تعداد زوج مرتب مثلث‌هایی که با هم دو راس مشترک دارند

$$\binom{n}{3} \binom{3}{1} \binom{n-3}{1} \approx \frac{n^3}{6} (3)(n) = \frac{n^4}{2}$$

• تعداد زوج مرتب مثلث‌هایی که با هم سه راس مشترک دارند

$$\binom{n}{3} \approx \frac{n^3}{6}$$

$$\mathbb{E}[T_{r,n}^r] \approx \frac{n^3}{6} p^3 + \frac{n^4}{2} p^5 + \frac{n^6}{36} p^6$$

اگر  $p$  به  $n$  وابسته نباشد، فقط جمله آخر میماند.

پرسش تئوری ۲۱. با استفاده از رابطه  $\text{Var}[T_{n,r}] = \mathbb{E}[T_{n,r}^2] - (\mathbb{E}[T_{n,r}])^2$  و جواب آخر پرسش تئوری ۱۶ داریم:

$$\text{Var}[T_{n,r}] \approx \frac{n^3}{6} p^3 + \frac{n^4}{2} p^5 + \frac{n^6}{36} p^6 - \frac{n^6}{36} p^6 = \frac{n^3}{6} p^3 + \frac{n^4}{2} p^5$$

$$\Rightarrow \lim_{n \rightarrow \infty} \frac{\text{Var}[Y_n]}{\mathbb{E}^2[Y_n]} = \lim_{n \rightarrow \infty} \frac{\frac{n^4}{2} p^5}{\frac{n^6}{36} p^6} = \frac{18}{c} \lim_{n \rightarrow \infty} \frac{1}{n^2} = 0$$

پرسش تئوری ۲۲. با توجه به قضیه ۱ و پرسش قبل، نتیجه میشود ۱  $\mathbb{P}[T_{r,n} \geq 1] = 1$  این به این معناست که احتمال نبود هیچ مثلثی در گراف، با افزایش  $n$  بسیار کوچک میشود.

پرسش تئوری ۲۳. ابتدا ثابت میکنیم اگر متغیر تصادفی  $X$  از توزیع  $poisson(\lambda)$  پیروی کند، آنگاه  $\mathbb{E}[(X)_r] = \lambda^r$

$$M_X(t) = \mathbb{E}[t^X] = \sum_{k=0}^{\infty} \frac{e^{-\lambda}(\lambda^k)}{k!} t^k = e^{-\lambda} \sum_{k=0}^{\infty} \frac{(t\lambda)^k}{k!} = e^{-\lambda} e^{t\lambda}$$

$$\Rightarrow M_X(t) = e^{\lambda(t-1)} \quad (22.1)$$

$$\Rightarrow \mathbb{E}[(X)_r] = \frac{\partial^r}{\partial t^r} M_X(t)|_{t=1} = \lambda^r e^{\lambda(t-1)}|_{t=1}$$

$$\Rightarrow \mathbb{E}[(X)_r] = \lambda^r$$

حال ثابت میکنیم که اگر  $\mathbb{E}[(X)_r] = \lambda^r$  آنگاه  $X \sim poisson(\lambda)$

$$\Rightarrow \frac{\partial^r}{\partial t^r} M_X(t)|_{t=1} = \lambda^r$$

حال تابع  $M_X(t)$  را حول  $t = 1$  بسط تیلور میدهیم:

$$M_X(t) = \sum_{r=0}^{\infty} \frac{1}{r!} \frac{\partial^r}{\partial t^r} M_X(t)|_{t=1} (t-1)^r = \sum_{r=0}^{\infty} \frac{\lambda^r}{r!} (t-1)^r = \sum_{r=0}^{\infty} \frac{(\lambda(t-1))^r}{r!}$$

که این همان سری متناظر با  $e^{\lambda(t-1)}$  است که با توجه به (۲۲.۱) همان تابع گشتاور فاکتوریل پواسون با پارامتر  $\lambda$  است. (با فرض اینکه یک تابع گشتاور فاکتوریل متعلق به تنها یک توزیع است)

پرسش تئوری ۲۴. با استفاده از پرسش تئوری ۱۶ در  $n$  های بزرگ داریم:

$$\mathbb{E}[(T_{r,n})_1] = \mathbb{E}[T_{r,n}] = \binom{n}{r} p^r \approx \frac{n^r}{r!} p^r = \frac{c^r}{r!}$$

برای دومی داریم:

$$\mathbb{E}[(T_{r,n})_2] = \mathbb{E}[T_{r,n}(T_{r,n} - 1)] = \mathbb{E}[T_{r,n}^2] - \mathbb{E}[T_{r,n}]$$

با توجه به نتایج پرسش ۱۶ و ۲۰ تئوری، داریم:

$$\Rightarrow \mathbb{E}[(T_{r,n})_2] \approx \frac{n^r}{r!} p^r + \frac{n^r}{r!} p^r + \frac{n^r}{r!} p^r - \frac{n^r}{r!} p^r = \frac{c^r}{r!} + \frac{c^r}{r!}$$

$$\approx \frac{c^r}{r!} = \lambda^r$$

$$\Rightarrow \mathbb{E}[(T_{r,n})_2] \sim \lambda^r$$

پرسش تئوری ۲۵.  $T_{r,n}$  جمع یک سری متغیر تصادفی برنولی است که در  $n$  های نه چندان بزرگ مستقل نیستند. اما با توجه به راهمایی سوال در  $n$  های بزرگ، تقریباً هیچ کدام از  $I_{u,v,w}$  ها راس مشترک ندارند.

طبق راهنمایی سول تعداد آن هایی که راس مشترک دارند بسیار کمتر از آنهایی است که ندارند). پس  $I_{u,v,w}$  ها مستقل خواهند شد. میدانیم که جمع  $n$  متغیر تصادفی برنولی با پارامتر  $p$  توزیع  $Binomial(n, p)$  خواهد داشت. در اینجا تعداد  $I_{u,v,w}$  ها  $\frac{n^r}{r!} \approx \binom{n}{r}$  است که احتمال یک بودن آن طبق بخش های قبل  $p^r$  است. پس در  $n$  های بزرگ  $T_{r,n} \sim Binomial(\binom{n}{r}, p^r)$

$$\Rightarrow M_{T_{r,n}}(t) = \mathbb{E}[t^{T_{r,n}}] = \sum_{k=0}^{\binom{n}{r}} \binom{\binom{n}{r}}{k} t^k (p^r)^k (1-p^r)^{\binom{n}{r}-k} = \sum_{k=0}^{\binom{n}{r}} \binom{\binom{n}{r}}{k} (tp^r)^k (1-p^r)^{\binom{n}{r}-k}$$

که عبارت بالا بسط دوجمله ایست.

$$\Rightarrow M_{T_{r,n}}(t) = (tp^r + 1 - p^r)^{\binom{n}{r}}$$

پس داریم:

$$\begin{aligned} \mathbb{E}[(T_{r,n})_r] &= \frac{\partial^r}{\partial t^r} M_{T_{r,n}}(t) \Big|_{t=1} = \binom{n}{r} \left( \binom{n}{r} - 1 \right) \dots \left( \binom{n}{r} - r + 1 \right) p^{r^2} (tp^r + 1 - p^r)^{\binom{n}{r}-r} \Big|_{t=1} \\ &\approx \frac{n^r}{r!} \left( \frac{n^r}{r!} - 1 \right) \dots \left( \frac{n^r}{r!} - r + 1 \right) p^{r^2} \approx \left( \frac{n^r}{r!} \right)^r p^r = \left( \frac{n^r}{r!} \frac{c^r}{n^r} \right)^r = \lambda^r \\ \mathbb{E}[(T_{r,n})_r] &\sim \lambda^r \end{aligned}$$

۷

پرسش شبیه سازی ۱۳. تابع زیر برای پیدا کردن راس های منفرد است.

`networkx.isolates(g)`

و تابع زیر برای چک کردن هم بندی (خروجی boolean است)

`networkx.is_connected(g)`

۱۰۰ متاگراف با  $n = ۱۵۰$  و  $p = ۰.۲$  میسازیم.

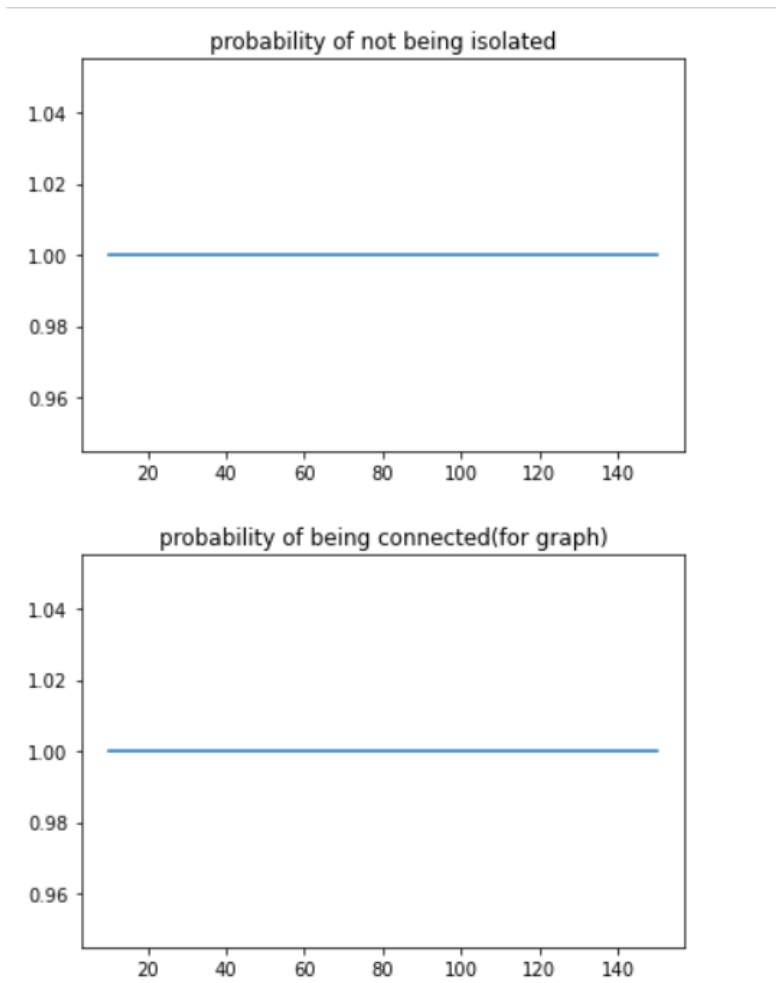
```
# 13

n = 150
p = 0.2
graphs = []
for i in range(100):
    graphs.append(nx.fast_gnp_random_graph(n, p))
isolates = 0
connected = 0
for g in graphs:
    isolates += len(list(nx.isolates(g))) / n
    connected += int(nx.is_connected(g))
print("probability of being isolated => {}".format(isolates/100))
print("probability of being connected => {}".format(connected/100))

probability of being isolated => 0.0
probability of being connected => 1.0
```

شکل ۱۸ نتیجه در پایان تصویر

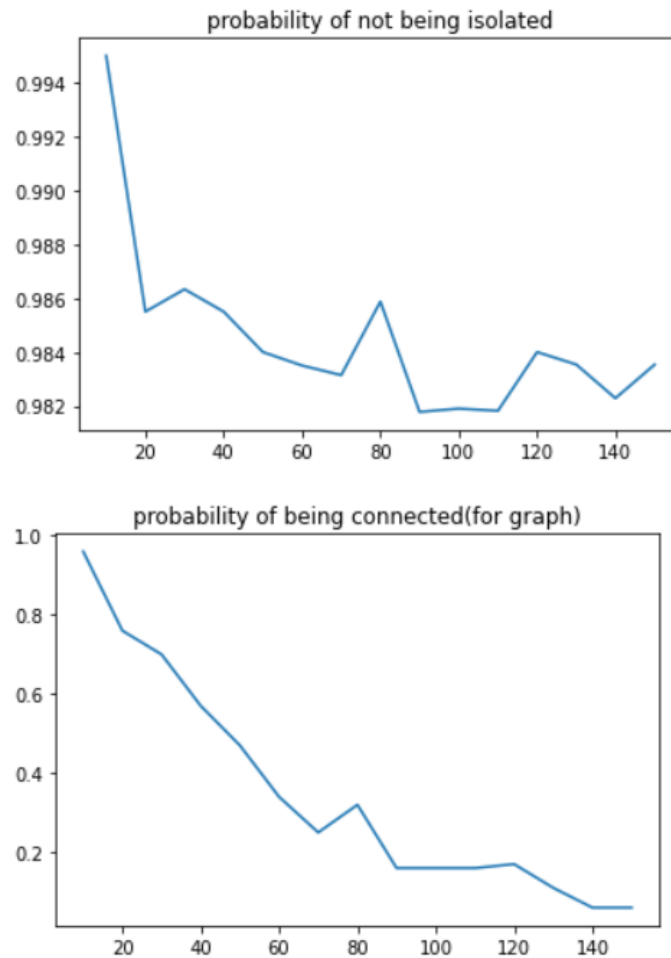
پرسش شبیه سازی ۰.۱۴ به ازای  $n$  از ۱۰ تا ۱۵۰ با گام ۱۰ و به ازای  $p = \frac{4 \ln(n)}{n}$  شبیه سازی قبل را تکرار میکنیم.



شکل ۱۹ نمودار بالا احتمال عدم وجود فرد بدون دوست و نمودار پایین احتمال هم‌بندی گراف

که این سازگار است با آنچه در تئوری خواهیم دید.  
 پرسش شبیه سازی ۰۱۵. این بار به ازای  $p = \frac{4}{n}$  شبیه سازی قبلی را تکرار میکنیم.





شکل ۲۰ نمودار بالا احتمال عدم وجود فرد بدون دوست و نمودار پایین احتمال هم‌بندی گراف

پرسش تئوری ۲۶.  $A_i^{(j)}$  را پیشامد اینکه  $i$  امین حالت انتخاب  $j$  راس از  $n$  راس، از بقیه راس ها جدا باشد. واضح است که پیشامد  $K_i^{(j)} = 1$  زیر مجموعه ی  $A_i^{(j)}$  است:

$$(K_i^{(j)} = 1) \subseteq A_i^{(j)} \Rightarrow \mathbb{P}[K_i^{(j)} = 1] \leq \mathbb{P}[A_i^{(j)}]$$

از طرفی احتمال اینکه همه  $j$  راس از بقیه راس ها جدا باشند،  $((1-p)^{n-j})^j$  است. پس از بالا داریم:

$$\Rightarrow \mathbb{P}[K_i^{(j)} = 1] \leq (1-p)^{j(n-j)}$$

پرسش تئوری ۲۷. با توجه به تعریف  $X_j$  و  $K_i^{(j)}$  داریم: (تعداد کل  $K_i^{(j)}$  ها  $\binom{n}{j}$  است.)

$$X_j = \sum_{i=1}^k K_i^{(j)} \Rightarrow \mathbb{E}[X_j] = \sum_{i=1}^{\binom{n}{j}} \mathbb{E}[K_i^{(j)}]$$

چون  $K_i^{(j)}$  متغیر تصادفی برنولی است، پس  $\mathbb{E}[K_j^{(i)}] = \mathbb{P}[K_i^{(j)} = 1]$  با توجه به نامساوی پرسش ۲۶ خواهیم داشت:

$$\mathbb{E}[X_j] \leq \sum_{i=1}^{\binom{n}{j}} (1-p)^{j(n-j)}$$

$$\Rightarrow \mathbb{E}[X_j] \leq \binom{n}{j} (1-p)^{j(n-j)}$$

پرسش تئوری ۲۸. با توجه به تعریف  $X$  داریم: (همچنین از نامساوی بخش قبل)

$$X = \sum_{j=1}^n X_j \Rightarrow \mathbb{E}[X] = \sum_{j=1}^n \mathbb{E}[X_j] \leq \sum_{j=1}^n \binom{n}{j} (1-p)^{j(n-j)}$$

با توجه به نامساوی صورت سوال خواهیم داشت:

$$\mathbb{E}[X] \leq \sum_{j=1}^n \frac{1}{n^{1/j}} = \frac{n}{n^{1/n}}$$

$$\Rightarrow \mathbb{E}[X] \leq \frac{1}{n^{1/n}}$$

در حد به سمت بینهایت طبق قضیه ساندویچ خواهیم داشت:

$$0 \leq \mathbb{E}[X] \leq \frac{1}{n^{1/n}} \Rightarrow \lim_{n \rightarrow \infty} 0 \leq \lim_{n \rightarrow \infty} \mathbb{E}[X] \leq \lim_{n \rightarrow \infty} \frac{1}{n^{1/n}}$$

$$\Rightarrow \lim_{n \rightarrow \infty} \mathbb{E}[X] = 0$$

پرسش تئوری ۲۹. ابتدا با استفاده از تعریف  $X$  داریم:

$$\mathbb{P}[\mathcal{G}(n, p) \text{ هم بند بودن}] = \mathbb{P}[X = 0] = 1 - \mathbb{P}[X \geq 1]$$

$$\lim_{n \rightarrow \infty} \mathbb{P}[\mathcal{G}(n, p) \text{ هم بند بودن}] = 1 - \lim_{n \rightarrow \infty} \mathbb{P}[X \geq 1] \quad (29.1)$$

طبق نامساوی مارکف داریم:

$$\mathbb{P}[X \geq 1] \leq \frac{\mathbb{E}[X]}{1}$$

$$\Rightarrow \lim_{n \rightarrow \infty} \mathbb{P}[X \geq 1] \leq \lim_{n \rightarrow \infty} \mathbb{E}[X]$$

طبق پرسش قبل  $\lim_{n \rightarrow \infty} \mathbb{E}[X] = 0$  پس :

$$\lim_{n \rightarrow \infty} \mathbb{P}[X \geq 1] = 0$$

با جایگذاری در (۲۹.۱) خواهیم داشت:

$$\lim_{n \rightarrow \infty} \mathbb{P}[\mathcal{G}(n, p) \text{ هم بند بودن}] = 1$$