سوال 1:

بخش 1: ابتدا دیتاست را به کمک کتابخانه pandas ورودی گرفتم و در متغیر df که همان Dataframe هست دخیره کردم. سپس ستونهای مربوط به مترو و بی آر تی را به کمک کتابخانه matplotlib و دستور hist() رسم کردم.

بخش دوم: هر کدام از متغیر های تصادفی X و Y که تعداد اتفاقات یک پیشامد (تعداد گذر مترو و اتوبوس) در یک بازه زمانی خاص یا را نشان می دهند که این همان توزیع پواسون است. از روی شکل نمودار های آن ها هم می توان به این توزیع پی برد. برای محاسبه پارامتر های این دو توزیع پواسون هم از دستور mean() که برای کتابخانه pandas است استفاده میکنیم چون که پارامتر توزیع پواسون با میانگین آن برابر است و برای هم مترو و بی آر تی، آنها را محاسبه می کنیم که برای متغیر تصادفی X، پارامتر آن برابر 3.5316 و برای Y برابر 2.0636 می شود.

بخش سوم: در این قسمت از ما خواسته شده که density histogram را برای گذر های مترو بکشیم.

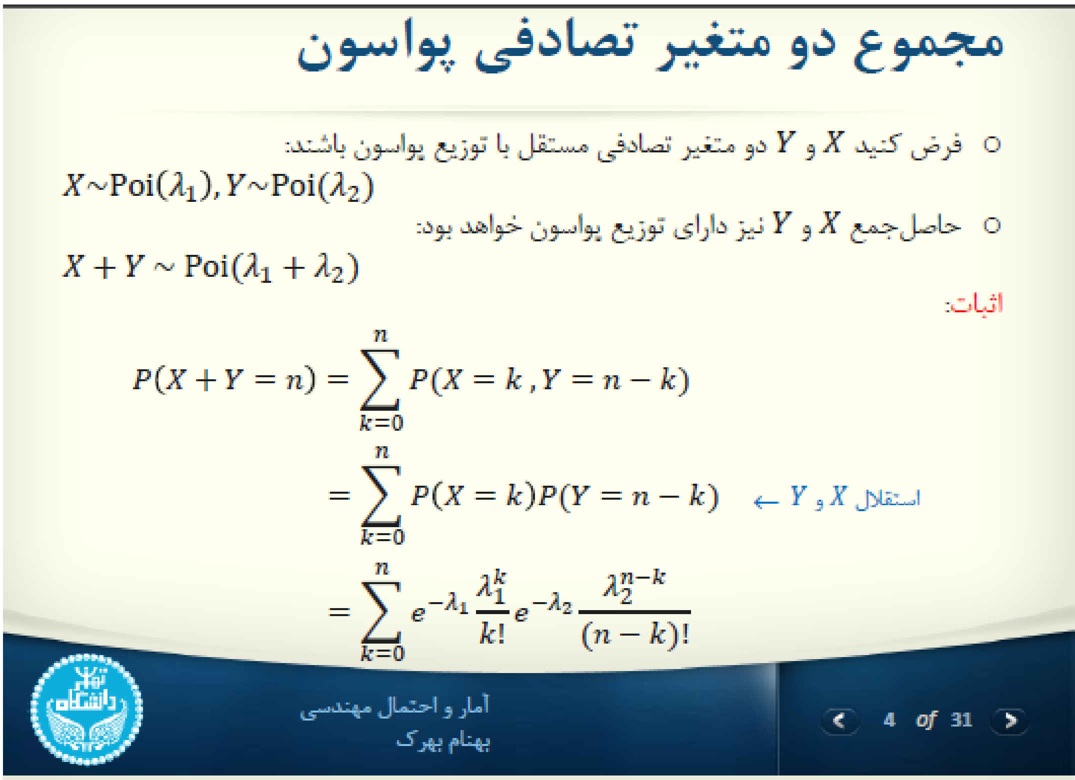
برای این کار یکی از ورودی های تابع hist() که density هست را برابر با True قرار میدهیم.

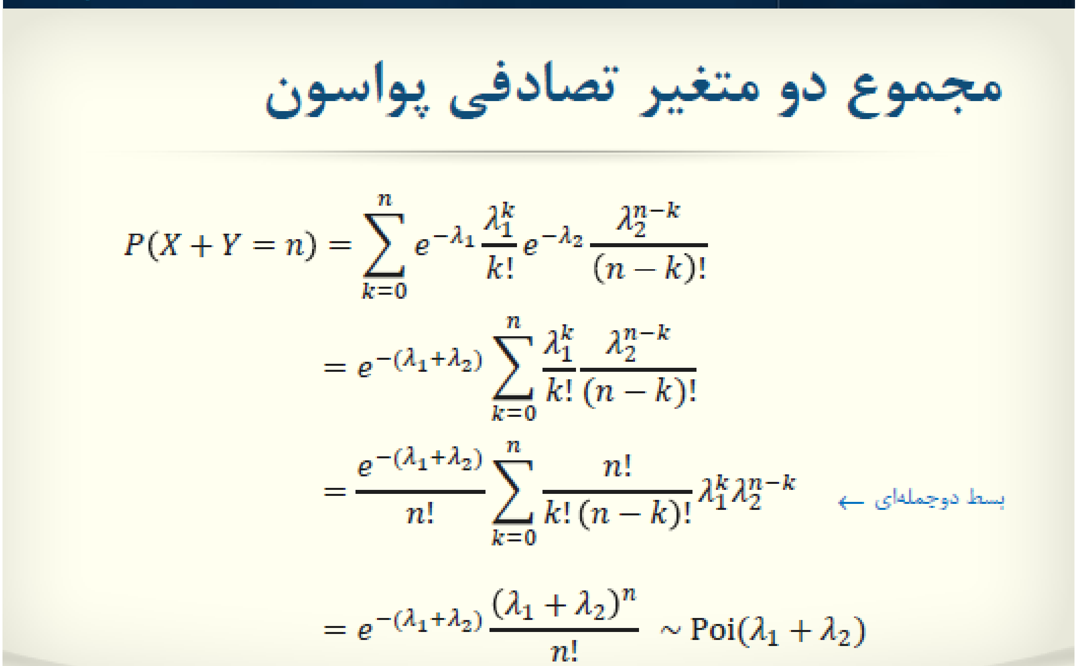
بخش چهار: در این قسمت باید نمودار توزیع برای گذر های مترو را بکشیم توسط پارامتری که در بخش دو بدست آوردیم و با بخش سوم نمودارها را مقایسه کنیم. برای این کار توسط تابع poisson.pmf() که در scipy.stats است نمودار توزیع را میکشیم و مشاهده میشود که نمودارهای بخش سوم و چهارم با تقریب خوبی بر هم منطبق هستند.

بخش پنجم: دو متغیر تصادفی X & Y از هم مستقل هستند. بنابراین جمعشان یعنی متغیر تصادفی Z یک متغیر تصادفی جدید با پارامتر مجموع پارامترهای X , Y میشود. نمودار این متغیر به کمک stats.poisson میکشیم. همچنین یکبار هم گذرهای مترو و بی آر تی را نظیر به نظیر درایه ها با هم جمع میکنیم و در یک دیتافریم دخیره میکنیم(df['Z']).

هیستوگرام این را هم رسم میکنیم. مشاهده میشود میشود که دو نمودار رسم شده با تقریب خوبی بر هم منطبق هستند.

قسمت شش: توزیع W و پارامترهای آن را اینگونه بدست می آوریم:





بخش 7 و 8: یک بار نمودار را با استفاده از توزیع و پارامتری که در قسمت قبل بدست آمده رسم کرده که از توزیع دو جمله

ای پیروی میکند و یک بار هم به صورت عملی احتمال X به شرط اینکه X + Y برابر 8 شود را حساب کرده و نمودار این دو را رسم میکنیم. این دو نمودار هم بر هم با تقریب خوبی منطبق اند.

سوال 2:

بخش اول و دوم: در این قسمت باید تابعی بنویسیم که مقادیر nو k را دریافت می کند و هر دفعه یک عدد تصادفی به عنوان کوپن با استفاده از numpy.randint() بسازد و

تعداد دفعات لازم برای دیدن تمام n کوپن را حساب کند و k دفعه این را تکرار و انجام دهد و سپس میانگین جواب آزمایش ها را برگرداند.

اگر این آزمایش ها را به تعداد زیاد تکرار کنیم(k -> ∞)

مشاهده میشود که این مقدار به nlgn میل میکند. (29.316)

to collect all n coupons, we need

$$T = T_1 + T_2 + \ldots + T_n.$$

So

$$E[T] = E[T_1] + E[T_2] + \ldots + E[T_n].$$

After collecting $k-1$ coupons, for next purchase:

- Failure = getting coupon you already have.
  There are $k-1$ choices.
- Success = getting a new coupon.
  There are $n-k+1$ choices.

Then T_k is geometric random variable with

success probability $p = \dfrac{n-k+1}{n}$.

So $$E[T_k] = \frac{1}{p} = \frac{n}{n-k+1}.$$

$$\begin{aligned}
E[T] &= E[T_1] + E[T_2] + E[T_3] + \ldots + E[T_n] \\
&= \frac{n}{n} + \frac{n}{n-1} + \frac{n}{n-2} + \ldots + \frac{n}{1} \\
&= n\left(\frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \ldots + \frac{1}{1}\right) \\
&= n\left(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}\right).
\end{aligned}$$

This is harmonic series, and

$E[T]$ is approximately $n\displaystyle\int_1^n \frac{1}{x}\,dx = n\log n.$

(این عکس ها از موضوع تمرین که در یک pdf بود

برداشتم 😊 البته این موضوع در کلاس هم اثبات شده)

بخش سوم و چهارم و پنجم:

T = X

هر کدام از Ti ها از توزیع هندسی پیروی میکنند اما P هر کدام متفاوت از دیگری است چون پس از مشاهده یک کوپن، احتمال مشاهده کوپن بعدی به دلیل دیده شدن یک سری کوپن از آنها تغییر می کند برای هر کدام از این Ti ها MGF را به صورت زیر محاسبه می کنیم:

$$\phi_X(s) = \sum_{k=1}^{\infty}(1-p)^{k-1}pe^{ks} = pe^s\sum_{m=0}^{\infty}\left((1-p)e^s\right)^m = \frac{pe^s}{1-(1-p)e^s}$$

الان هر کدام از این Ti ها را به صورتی تابعی از s و با استفاده از sympy و p که برای هر کدام بدست می آوریم، حساب میکنیم و در نهایت **امید ریاضی** T می شود مجموع امید ریاضی این Ti ها. MGFمتغیر تصادفی T را حساب کردیم، با توجه به اینکه مشتق اول تابع MGF در نقطه s=0 برابر گشتاور مرتبه اولT یا همان امید ریاضی T است، امید ریاضی T را هم از این طریق بدست می آوریم.

که برابر 29.2896825396826 می شود و تقریبا برابر مقداری است که در قسمت قبل بدست آوردیم.

سوال 4:

قسمت اول و دوم و سوم:

ابتدا دیتاست را دریافت می کنیم و طبق گفته سوال اعداد 201 و 202 ام را از دیتافریم حذف می کنیم و با قراردادن threshold اعداد را باینری می کنیم .سپس در قسمت 3، یک سطر با label 8 و یا با label 9 را انتخاب کرده و آن را در جایی (آرایه) ذخیره میکنیم و به یک آرایه 28 * 28تبدیل می کنیم با استفاده از numpy.reshape() چون هر تصویر از 784 پیکسل تشکیل شده و یک تصویر28 * 28تشکیل می شود. الان هم به کم matplotlib.pyplot.imshow() تصویر آن را میکشیم.

بخش چهار: بخش هایی که باید انجام بدهیم در صورت سوال آمده:

```
# calculate P(N = n| Y = p) which is a bernouli distribution
    # calculate integral(0 -> 1) fy * pny
    ## pny =
    ## integral =
    ## post =
```

بعد از انجام این بخش، نمودار توزیع کشیده میشود و مد و میانگین به هم بسیار نزدیک هستند چون داده ها به تعداد خوبی هستند و p برابر با 0.67 میشود. که همان مد یا میانگین توزیع بتا پسین میشود.