

گزارش آمار و احتمال

CA1

نیلوفر مرتضوی

220701096

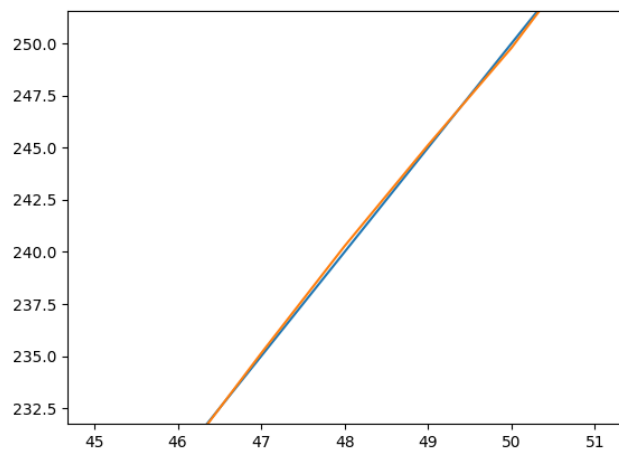
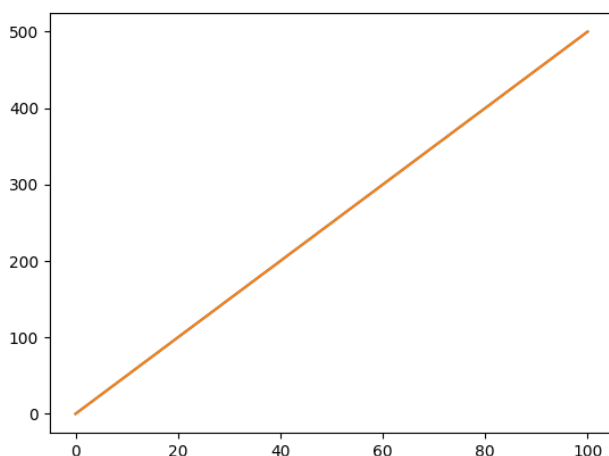
سوال اول

برای این بخش یک تابع به نام `mybinomial` نوشتم که با گرفتن ورودی های ابعاد آرایه دو بعدی (ماتریکس) مورد نظر و احتمال پیشامد 1 در آزمایش ها (p)، درواقع یک ماتریکس از عدد های رندوم 1 یا 0 میسازد و در نهایت جمع ردیف های این ماتریکس را با استفاده از `numpy.sum` برمیگرداند. در واقع هر کدام از این ردیف های نشان دهنده یک یک نمونه از توزیع دو جمله ای متناظر با n بار تکرار یک توزیع برنولی است.

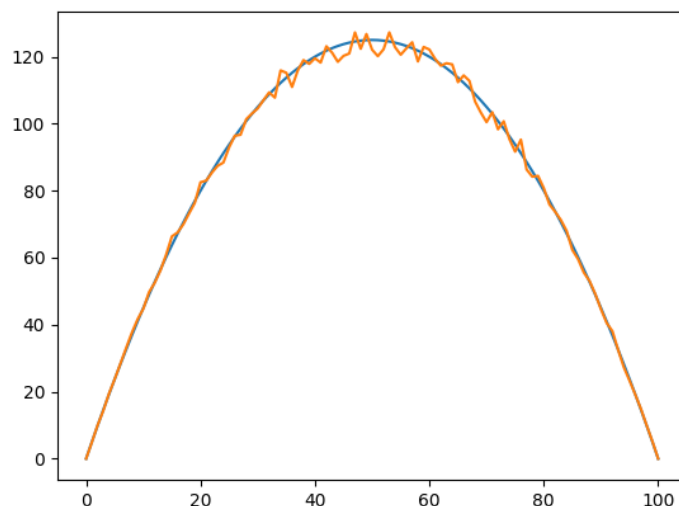
در قسمت بعدی 4 تابع نوشتم هر کدام برای محاسبه یکی از مقادیر واریانس کاربردی/نظری و میانگین کاربردی/نظری.

درواقع در هر یک از این توابع در یک حلقه تمام احتمالات بین 0 و 1 را با فاصله 0.01 بررسی میکند و نتایج هر کدام به ترتیب داخل یک آرایه (y)، `append` میکند. در نهایت برای تمام دیتا های ذخیره شده نمودار رسم کردم به این ترتیب دو گراف خواهیم داشت که میخواهیم در تفاوت واریانس نظری و کاربردی و همچنین تفاوت میانگین نظری و کاربردی را مشاهده کنیم:

در مقایسه مقادیر میانگین ها متوجه میشویم که تفاوت دو دیتا خروجی بسیار کم و قابل چشم پوشی است.

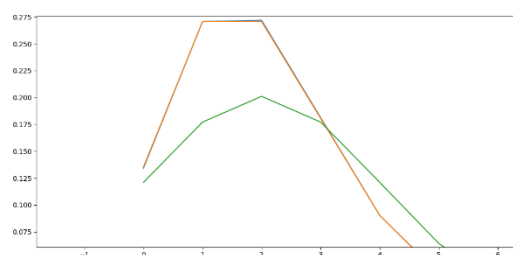
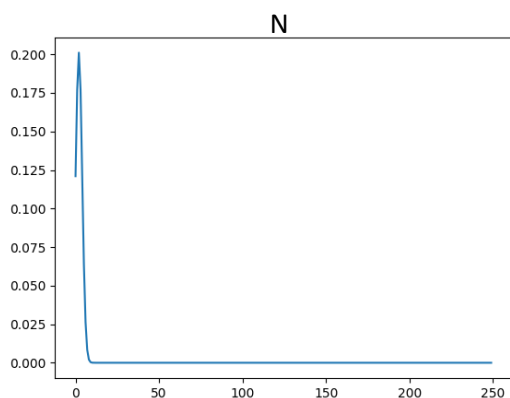
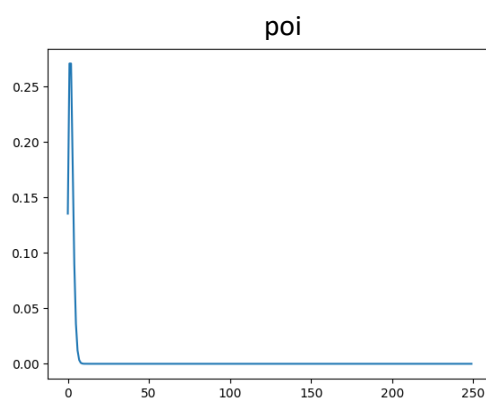
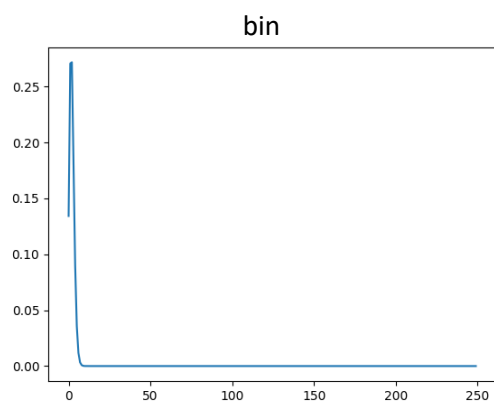


اما در مقایسه نمودار واریانس ها میتوان دید که این دو با هم اختلاف دارند و درواقع واریانس کاربردی (practical) حالت سینوسی بیشتری دارد.



سوال دوم

در این سوال با استفاده از کتابخانه `scipy.stats` و توابع `norm`, `poisson`, `binom` شروع به رسم نمودار ها کردم. نمودار ها را بر اساس میانگین تصادفات 250 محاسبه کردم و مشاهدات زیر صورت گرفت:



مشاهده میکنیم که نمودار های توزیع دو جمله ای و پواسون منطبق هستند اما نمودار توزیع نرمال کمی اختلاف دارد دلیل این اختلاف هم وجود n بزرگ و p بسیار کوچک است در نتیجه در این شکل از سوالات توزیع پواسون تقریب بهتری برای ما در مقایسه با نرمال است.

سوال سوم

برای قسمت الف در واقع باید صدک 90ام را پیدا کنیم. از تابع `norm.ppf` با $\mu=80$, $\sigma=12$ استفاده میکنیم.

برای قسمت ب باید اختلاف `ppf` های چارک دوم (0.5) و چارک سوم (0.75) را بدست بیاوریم.

برای قسمت ج باید از تابع `norm.cdf` استفاده کنیم و اختلاف `cdf` های 80 و 90 را بدست بیاوریم که مساوی با احتمال نمره بین 80 و 90 خواهد بود.

در نهایت پاسخ سوالات به شکل زیر خواهد بود:

```
6 mu = 80
7 std = 12
8 var = std * std
9
10 def a(): #الف
11     result = norm.ppf(0.9 , mu , std)
12     return result
13 def b(): #ب
14     result= norm.ppf(0.75 , mu , std) - norm.ppf(0.5 , mu , std)
15     return result
16 def c(): #ج
17     result = norm.cdf(90 , mu , var) - norm.cdf(80 , mu , var)
18     return result
19
20 print(a())
21 print(b())
22 print(c())
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\NanoCamp\Desktop\Python> python -u "c:\Users\NanoCamp\Desktop\Pytho
95.3786187865352
8.093877002352983
0.02768207363273223
PS C:\Users\NanoCamp\Desktop\Python>
```

سوال چهارم

الگوریتم پیاده سازی این سوال دقیقاً شبیه سوال دوم است و فقط اطلاعات اولیه مسئله متفاوت است. با رسم نمودار ها و مشاهده انها متوجه میشویم که این بار نمودار توزیع دوجمله ای و نرمال تشابه بسیاری دارند اما نمودار توزیع پواسون اخلاف قابل توجهی با دو توزیع قبلی دارد.

در این سوال هم مانند سوال دوم n بزرگ داریم اما تفاوت آن نبود p کوچک است که به همین دلیل در این سوال توزیع پواسون تقریب نزدیک تری برای ماست.

