

```
In [1]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
# 以上3つのライブラリが「3種の神器」
# 小数第3位までを表示する
# %precision 3
# 図を別のウインドウで表示する(デフォルトは inline)
# %matplotlib notebook
```

```
In [2]: # 二項分布 B(n,p) の定義
from scipy.special import comb # 組合せ数を使う
n=80 # パラメータの設定 (大きな n)
p=0.04 # パラメータの設定 (小さな p)
# x_range=np.arange(n+1) # (本来) x の範囲は 0 から n までの n+1 個の整数
x_range=np.arange(0, 12, 1) # x の範囲の指定、始点、終点、刻み幅
def bin(x):
    if x in x_range:
        return comb(n, x)*p**x*(1-p)**(n-x)
    else:
        return 0
BinProb = np.array([bin(x) for x in x_range]) # x_range(定義域)の各 x に対して確率を計算
```

```
In [3]: # ポアソン分布 Po(lam) の定義
from scipy.special import factorial # 階乗を使う
lam=n*p # パラメータ λ=np に設定
# x_range=np.arange(12) # x の範囲は 0 から 12個の整数 (つまり、11まで)
# x_range=np.arange(0, 15, 1) # x の範囲の指定、始点、終点、刻み幅
def po(x):
    if x in x_range:
        return np.power(lam, x)/factorial(x)*np.exp(-lam)
    else:
        return 0
PoProb = np.array([po(x) for x in x_range]) # x_set(定義域)の各 x に対して確率を計算
```

```
In [4]: # 確率分布表で比較
DT=pd.DataFrame(x_range, columns=[' x '])
DT[' Binomial ']=BinProb
DT[' Poisson ']=PoProb
DT
```

Out[4]:

	x	Binomial	Poisson
0	0	0.038168	0.040762
1	1	0.127226	0.130439
2	2	0.209394	0.208702
3	3	0.226843	0.222616
4	4	0.181947	0.178093
5	5	0.115233	0.113979
6	6	0.060017	0.060789
7	7	0.026436	0.027789
8	8	0.010051	0.011116
9	9	0.003350	0.003952
10	10	0.000991	0.001265
11	11	0.000263	0.000368

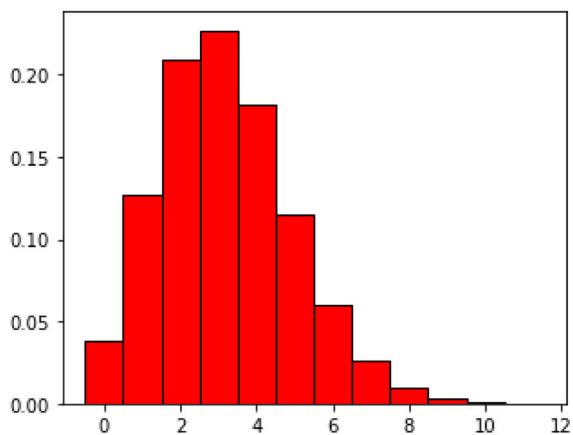
```
In [5]: DT['Difference']=DT['Binomial']-DT['Poisson']
DT
```

Out[5]:

	x	Binomial	Poisson	Difference
0	0	0.038168	0.040762	-0.002594
1	1	0.127226	0.130439	-0.003213
2	2	0.209394	0.208702	0.000691
3	3	0.226843	0.222616	0.004227
4	4	0.181947	0.178093	0.003854
5	5	0.115233	0.113979	0.001254
6	6	0.060017	0.060789	-0.000772
7	7	0.026436	0.027789	-0.001353
8	8	0.010051	0.011116	-0.001064
9	9	0.003350	0.003952	-0.000602
10	10	0.000991	0.001265	-0.000274
11	11	0.000263	0.000368	-0.000105

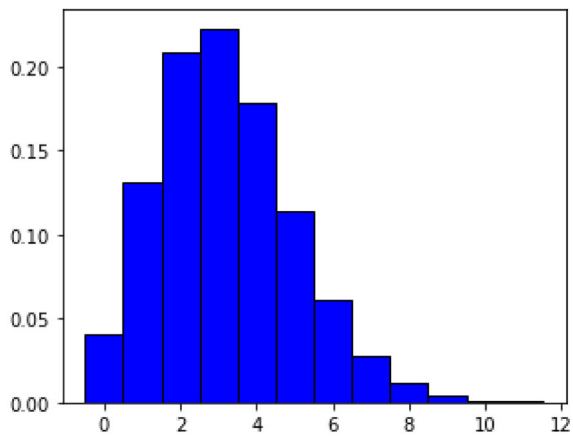
```
In [6]: # 確率分布の描画 (密度関数を念頭にヒストグラム風に)
fig = plt.figure(figsize=(5, 4))
plt.bar(x_range, BinProb, width=1, color='red', ec='black')
```

Out[6]: <BarContainer object of 12 artists>



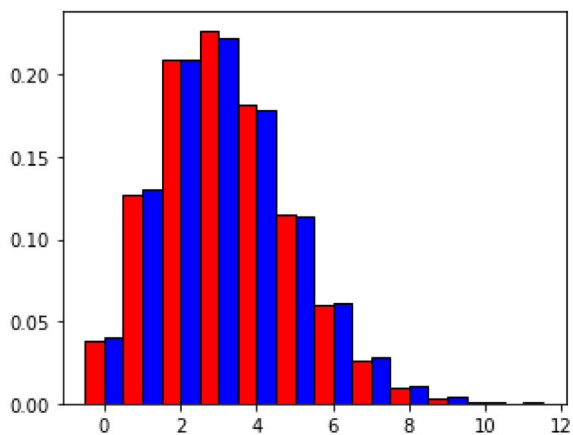
```
In [7]: fig = plt.figure(figsize=(5, 4))
plt.bar(x_range, PoProb, width=1, color='blue', ec='black')
```

Out[7]: <BarContainer object of 12 artists>



```
In [8]: fig = plt.figure(figsize=(5, 4))
plt.bar(x_range-0.25, BinProb, width=0.5, color='red', ec='black')
plt.bar(x_range+0.25, PoProb, width=0.5, color='blue', ec='black')
```

Out[8]: <BarContainer object of 12 artists>



```
In [9]: # 確率分布の比較
fig = plt.figure(figsize=(5, 4))
plt.plot(x_range, BinProb, color='red', label='Binomial')
plt.plot(x_range, PoProb, color='blue', label='Poisson')
plt.legend() # 凡例の表示
```

Out[9]: <matplotlib.legend.Legend at 0x1ae40d0e850>

