Homework №1

Liza Vlasova

4 октября 2022 г.

1 Prove the following:

$$\frac{1}{n}\sum_{1}^{n}(x_i-\mathbb{E}\xi)^2=\frac{1}{n}\sum_{1}^{n}(x_i-\bar{x})^2+(\mathbb{E}\xi-\bar{x})^2,$$

$\xi\sim\mathcal{P}$ is a random variable, $x_1,\ldots x_n$ is a sample from $\mathcal{P}$,

$$\bar{x}=\frac{1}{n}\sum_{1}^{n}x_i.$$

***Solution* :**

$$\frac{1}{n}\sum_{1}^{n}(x_i-\mathbb{E}\xi)^2=\frac{1}{n}\sum_{i=1}^{n}x_i^2-2\mathbb{E}\xi\cdot\frac{1}{n}\sum_{i=1}^{n}x_i+\frac{1}{n}(n\mathbb{E}\xi)^2=$$

$$\frac{1}{n}\sum_{i=1}^{n}x_i^2-2\mathbb{E}\xi\overline{x}+(\mathbb{E}\xi)^2=\frac{1}{n}\sum_{i=1}^{n}x_i^2-2\mathbb{E}\xi\overline{x}+(\mathbb{E}\xi)^2+2\overline{x}^2-2\overline{x}^2=$$

$$\frac{1}{n}\sum_{i=1}^{n}x_i^2-\frac{2\overline{x}}{n}\sum_{i=1}^{n}x_i+\frac{1}{n}\sum_{i=1}^{n}\overline{x}^2+(\mathbb{E}\xi)^2-2\mathbb{E}\xi\overline{x}+\overline{x}^2=$$

$$\frac{1}{n}(\sum_{i=1}^{n}x_i^2-2\overline{x}\sum_{i=1}^{n}x_i+\sum_{i=1}^{n}\overline{x}^2)+((\mathbb{E}\xi)^2-2\mathbb{E}\xi\overline{x}+\overline{x}^2)=\frac{1}{n}\sum_{1}^{n}(x_i-\bar{x})^2+(\mathbb{E}\xi-\overline{x})^2,$$

as was to be shown.

2 Given the following sample $x_1,\ldots x_{25}$

$$[2,0,1,3,3,1,1,2,0,2,1,0,0,0,3,4,1,3,2,2,0,1,2,1,2]$$

Calculate:
- Empirical distribution
- $\bar{x}$ (sample mean), s^2 (sample variance).
Suppose that this sample is actually generated from Binomial Distribution $\mathrm{Bin}(7, 0.2)$**. Compare the estimates you obtained with actual distribution.**

***Solution* :**

$$\widehat{F}_\xi(x) = \begin{cases} 0, x < 0, \\ 0.24, 0 \leq x < 1, \\ 0.52, 1 \leq x < 2, \\ 0.8, 2 \leq x < 3, \\ 0.96, 3 \leq x < 4, \\ 1, x \geq 4 \end{cases}$$

$$\overline{x} = \frac{0 \cdot 6 + 1 \cdot 7 + 2 \cdot 7 + 3 \cdot 4 + 4 \cdot 1}{25} = 1.48$$

$$s^2 = \frac{1.48^2 \cdot 6 + 0.48^2 \cdot 7 + (2 - 1.48)^2 \cdot 7 + (3 - 1.48)^2 \cdot 4 + (4 - 1.48)^2 \cdot 1}{25} = 1.2896$$

Let $\xi \sim Bin(7, 0.2)$, then:

- $\mathbb{E}\xi = np = 7 \cdot 0.2 = 1.4$

- $D\xi = npq = 7 \cdot 0.2 \cdot 0.8 = 1.12$

The values are not strongly different. Though we see the difference in sample and theoretical variance, which shows that there is still not that enough objects in a sample to be absolutely the same as for Bin(7, 0.2). It should be mentioned that the values would go closer to theoretical if we increase the count of objects in a sample.

Task 3. Write a function (on python), which for given (λ, n) generates a sample of size n from the distribution $Exp(\lambda)$. $F_\nu(x) = 1 - e^{-\lambda x}, \nu \sim Exp(\lambda)$.

Given that $F_\nu(x) = 1 - e^{-\lambda x}$, we can calculate that $F_\nu^{-1}(x) = -\frac{1}{\lambda} ln(1-x)$. Let's use the inverse function method which stands that given the uniform distribution $\mathcal{U}(0,1)$ we can generate a sample from any other distribution by substitution the uniform distribution into the inverse function.

This results in that $-\frac{1}{\lambda} ln(1 - \mathcal{U}(0,1)) \sim Exp(\lambda)$

Let's implement the stuff discussed before. In addition to that let's compare the results generated by our generator and by the scipy `expon` class. We can see that the results we get are very close to each other.

In [1]:
```
from random import random
from math import log
from scipy.stats import expon
import matplotlib.pyplot as plt
```

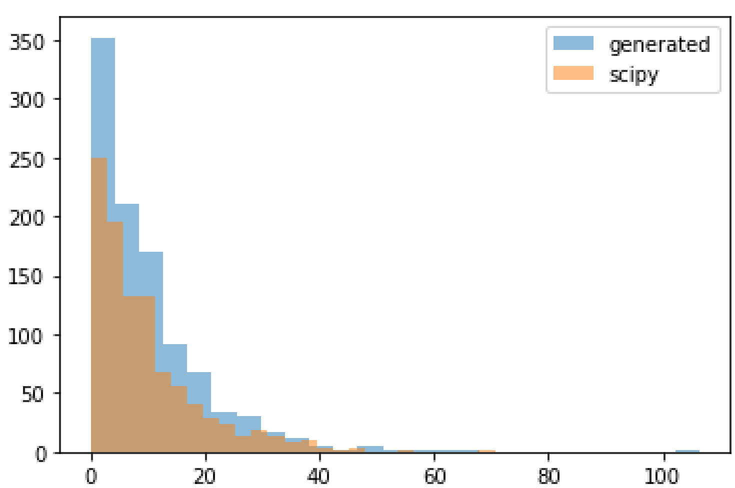
In [2]:
```
def generate_exp(l, n):
    return [-log(1 - random()) / l for i in range(n)]
```

In [3]:
```
lambda_parameter = 0.1

generated = generate_exp(lambda_parameter, 1000)
check = expon.rvs(size=1000, scale=1/lambda_parameter)

plt.hist(generated, alpha=0.5, bins=25, label='generated')
plt.hist(check, alpha=0.5, bins=25, label='scipy')
plt.legend()
```

Out[3]:
```
<matplotlib.legend.Legend at 0x2baf7ad6f08>
```

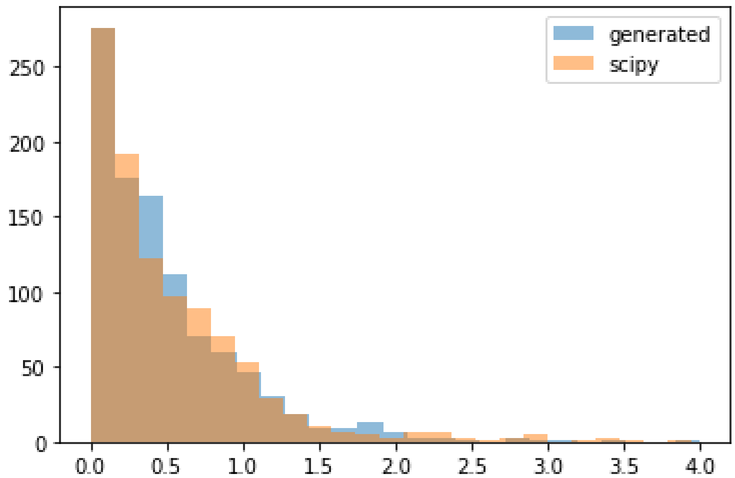


In [4]:
```
lambda_parameter = 2

generated = generate_exp(lambda_parameter, 1000)
check = expon.rvs(size=1000, scale=1/lambda_parameter)

plt.hist(generated, alpha=0.5, bins=25, label='generated')
plt.hist(check, alpha=0.5, bins=25, label='scipy')
plt.legend()
```

Out[4]: `<matplotlib.legend.Legend at 0x2baf7ad83c8>`

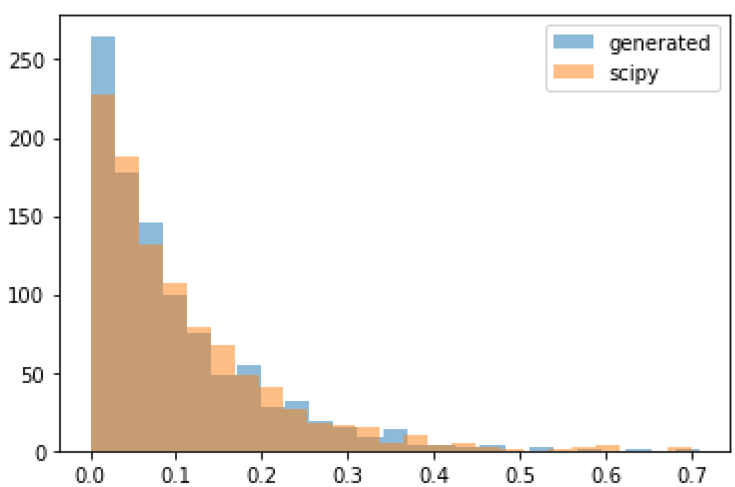


In [5]:
```
lambda_parameter = 10

generated = generate_exp(lambda_parameter, 1000)
check = expon.rvs(size=1000, scale=1/lambda_parameter)

plt.hist(generated, alpha=0.5, bins=25, label='generated')
plt.hist(check, alpha=0.5, bins=25, label='scipy')
plt.legend()
```

Out[5]: `<matplotlib.legend.Legend at 0x2baf8324248>`



In [ ]: