

# Лабораторная работа № 1

## Изучение инструментальной среды имитационного моделирования для исследования характеристик беспроводных сетей стандарта 5G

Выполнил студент 2 курса  
учебной группы НММ-02-22  
Мулин Иван

### Цели

- Ознакомиться с базовыми функциями библиотек `matplotlib` и `numpy` языка программирования Python и научиться их применять.

### Задача

Выполнить указанные ниже задания:

- Построить матрицу с количеством строк 10 и столбцов 2, переменные заполняются случайными числами с равномерным распределением. Полученные сгенерированные случайные числа представить на графике в виде точек.
- Сгенерировать 1000 случайных чисел с любым известным распределением и построить их гистограмму, математическое ожидание, дисперсию.
- Сгенерировать случайную точку, равномерно распределенную в квадрате со стороной  $a$

### Ход работы

Для выполнения заданий потребуется подключить библиотеки `numpy` и `matplotlib`:

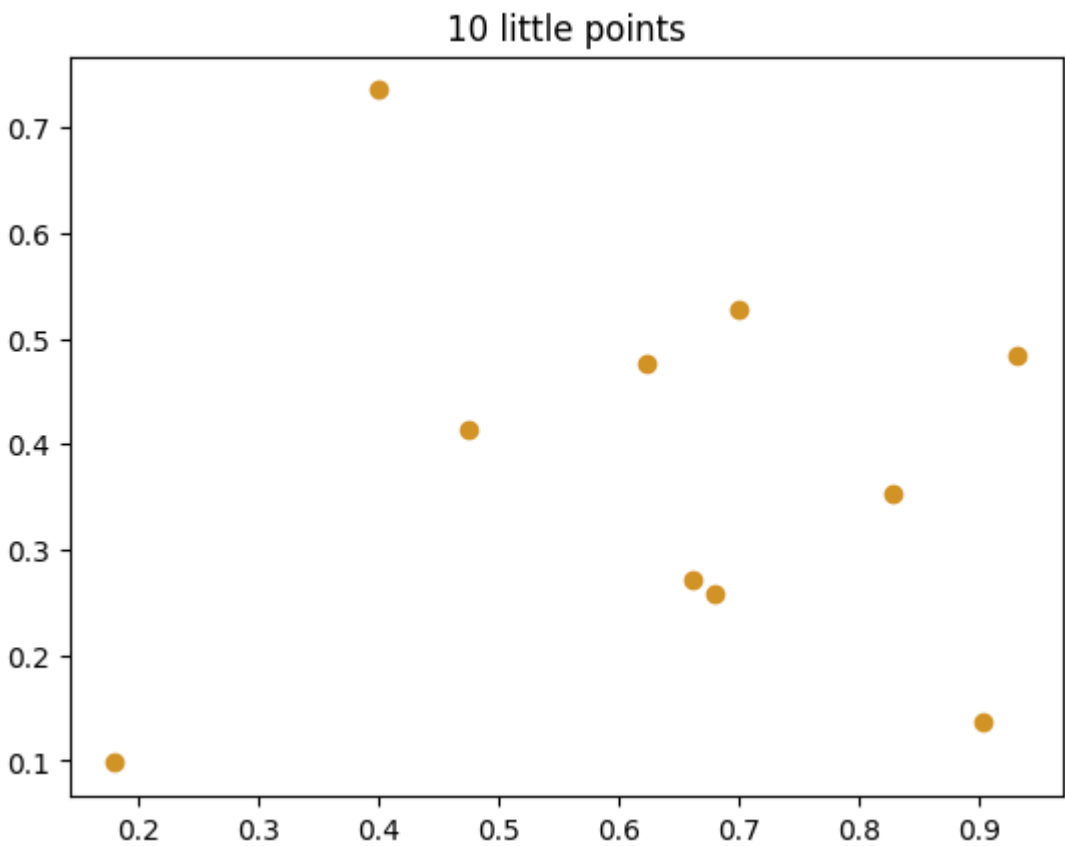
```
In [1]: import numpy as np
import matplotlib.pyplot as plt
```

#### Задание 1

```
In [2]: points = np.ndarray = np.random.uniform(size=(10, 2))

plt.figure(dpi=100)
plt.plot(points[:, 0], points[:, 1], marker="o", color="#d19226", linestyle="none")
plt.title("10 little points")
```

Out[2]: Text(0.5, 1.0, '10 little points')



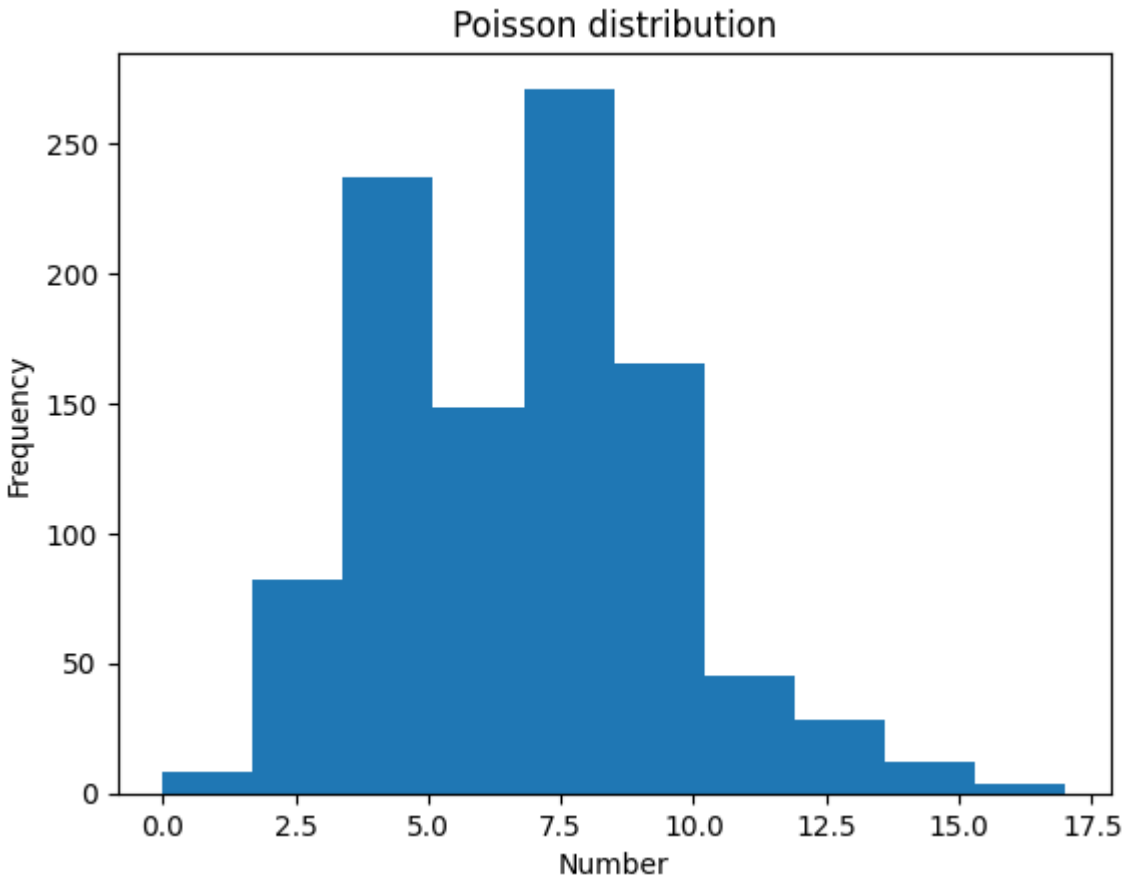
#### Задание 2

```
In [3]: numbers = np.ndarray = np.random.poisson(7, 1000)

plt.hist(numbers, histtype="barstacked")
plt.xlabel("Number")
plt.ylabel("Frequency")
plt.title("Poisson distribution")

print(f"Mean = {np.mean(numbers)}")
print(f"Variance = {np.var(numbers)}")
```

Mean = 6.861  
Variance = 7.043679

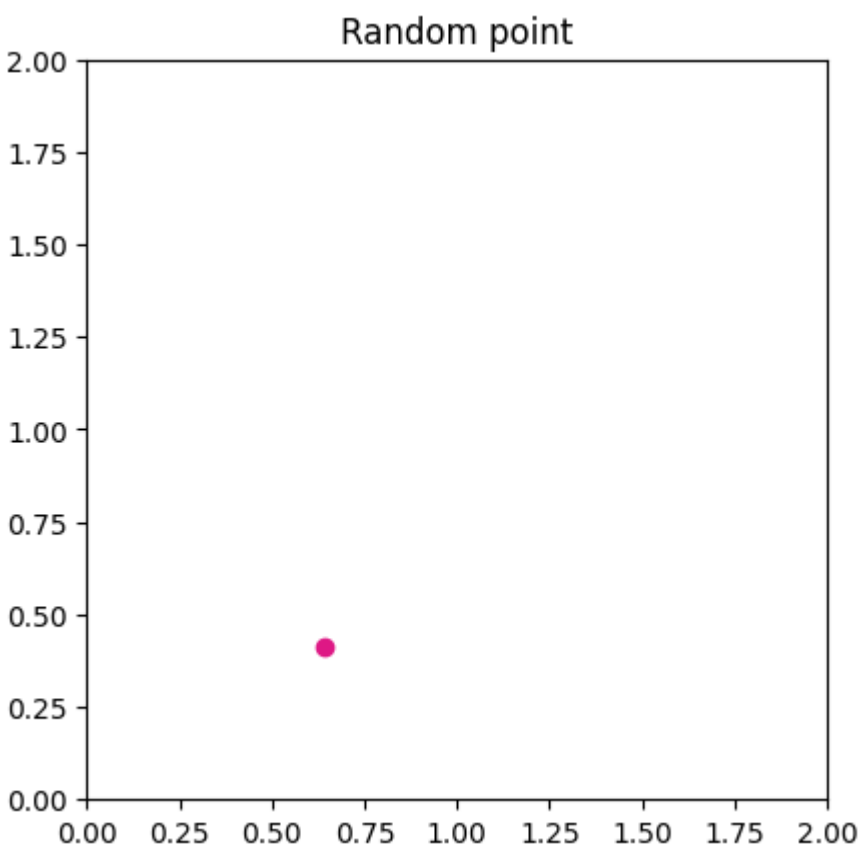


#### Задание 3

```
In [4]: a = float = 2.0
z = np.ndarray = np.random.uniform(size=(1, 2))

plt.figure(dpi=100)
plt.xlim(0, a)
plt.ylim(0, a)
plt.axis("scaled")
plt.plot(z[:, 0], z[:, 1], marker="o", color="#de1885", linestyle="none")
plt.title("Random point")
```

Out[4]: Text(0.5, 1.0, 'Random point')



### Результаты

При выполнении данной работы были изучены основные функции библиотек `matplotlib` и `numpy`, необходимые для дальнейшего выполнения индивидуального задания.

