



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ

к лабораторной работе №1

По курсу: «Моделирование»

Студент

ИУ7И-76Б

(Группа)

Нгуен Ф. С.

(Подпись, дата)

(И.О. Фамилия)

Преподаватель

Рудаков И.В.

(Подпись, дата)

(И.О. Фамилия)

Москва, 2021 г.

Оглавление

Формализация задачи.....	3
Равномерное распределение:.....	3
Нормальное распределение:	4
Результаты работы	5
Равномерное распределение:.....	5
Нормальное распределение:	7
Код программы	9

Формализация задачи

Равномерное распределение:

Равномерное распределение — распределение случайной величины, принимающей значения, принадлежащие некоторому промежутку конечной длины, характеризующееся тем, что плотность вероятности на этом промежутке всюду постоянна.

Равномерное распределение обозначают $X \sim R(a, b)$, где $a, b \in \mathbb{R}$.

Функция распределения равномерной непрерывной случайной величины:

$$F(x) = \begin{cases} 0 & \text{при } x \leq a \\ \frac{x-a}{b-a} & \text{при } a \leq x \leq b \\ 1 & \text{при } x > b \end{cases} \quad (1)$$

Плотность распределения равномерной непрерывной случайной величины:

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{при } a \leq x \leq b \\ 0 & \text{иначе} \end{cases} \quad (2)$$

Нормальное распределение:

Нормальное распределение - распределение вероятностей, которое в одномерном случае задаётся функцией плотности вероятности, совпадающей с функцией Гаусса:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3)$$

где параметр μ — математическое ожидание (среднее значение), медиана и мода распределения, а параметр σ - среднеквадратическое отклонение (σ^2 - дисперсия) распределения.

Функция распределения:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \quad (4)$$

Обозначают нормальное распределение $X \sim N(\mu, \sigma^2)$.

Стандартным нормальным распределением называется нормальное распределение с математическим ожиданием $\mu = 0$ и стандартным отклонением $\sigma = 1$.

Математическое ожидание μ характеризует положение «центра тяжести» вероятностной массы нормального распределения. Получается, что график плотности распределения случайной величины, имеющей нормальное распределение, симметричен относительно $x = \mu$. Дисперсия σ характеризует разброс значений случайной величины относительно «центра тяжести».

Результаты работы

Равномерное распределение:

Равномерное распределение

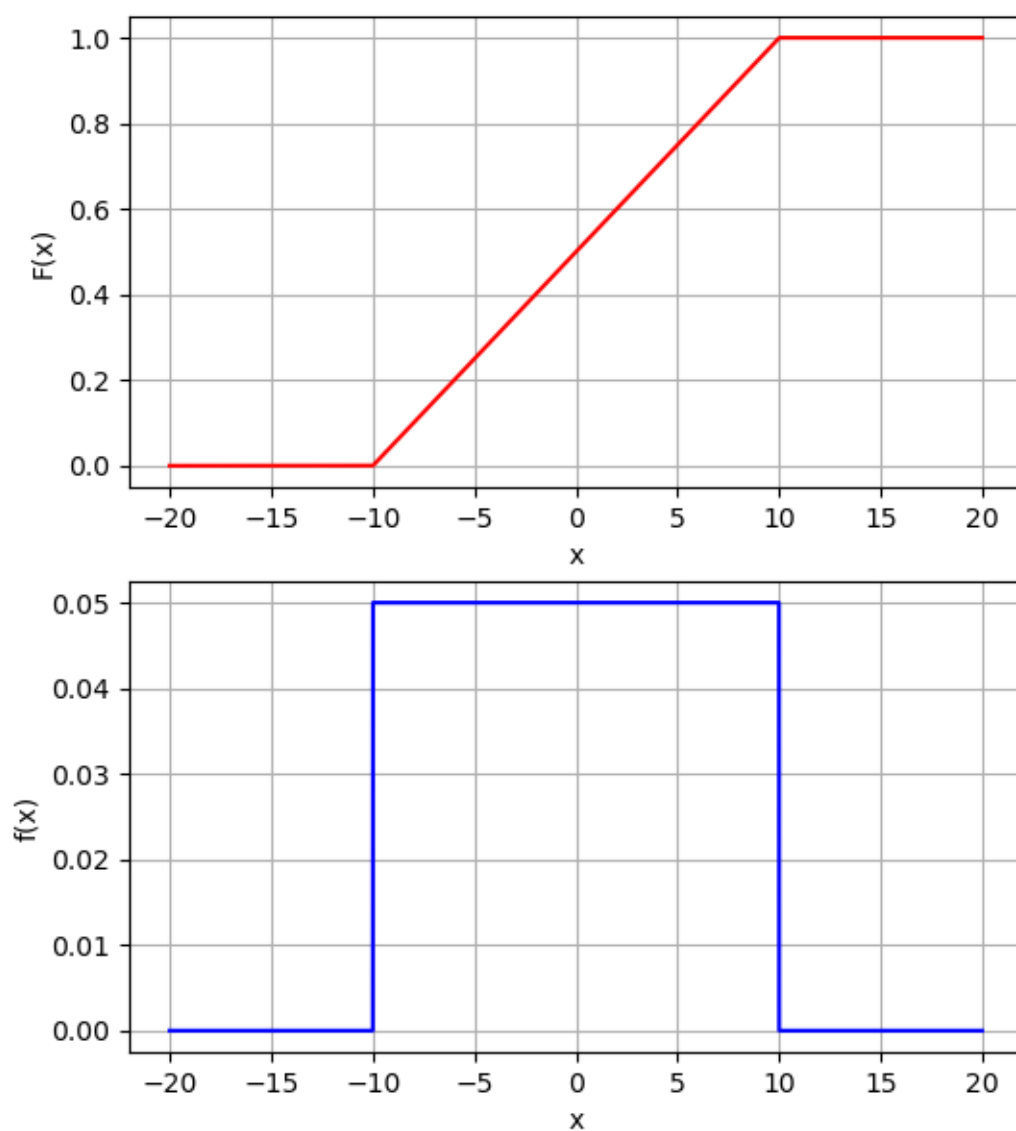


Рисунок 1 графики функции распределения и плотности распределения равномерной случайной величины при $a = -10$, $b = 10$

Равномерное распределение

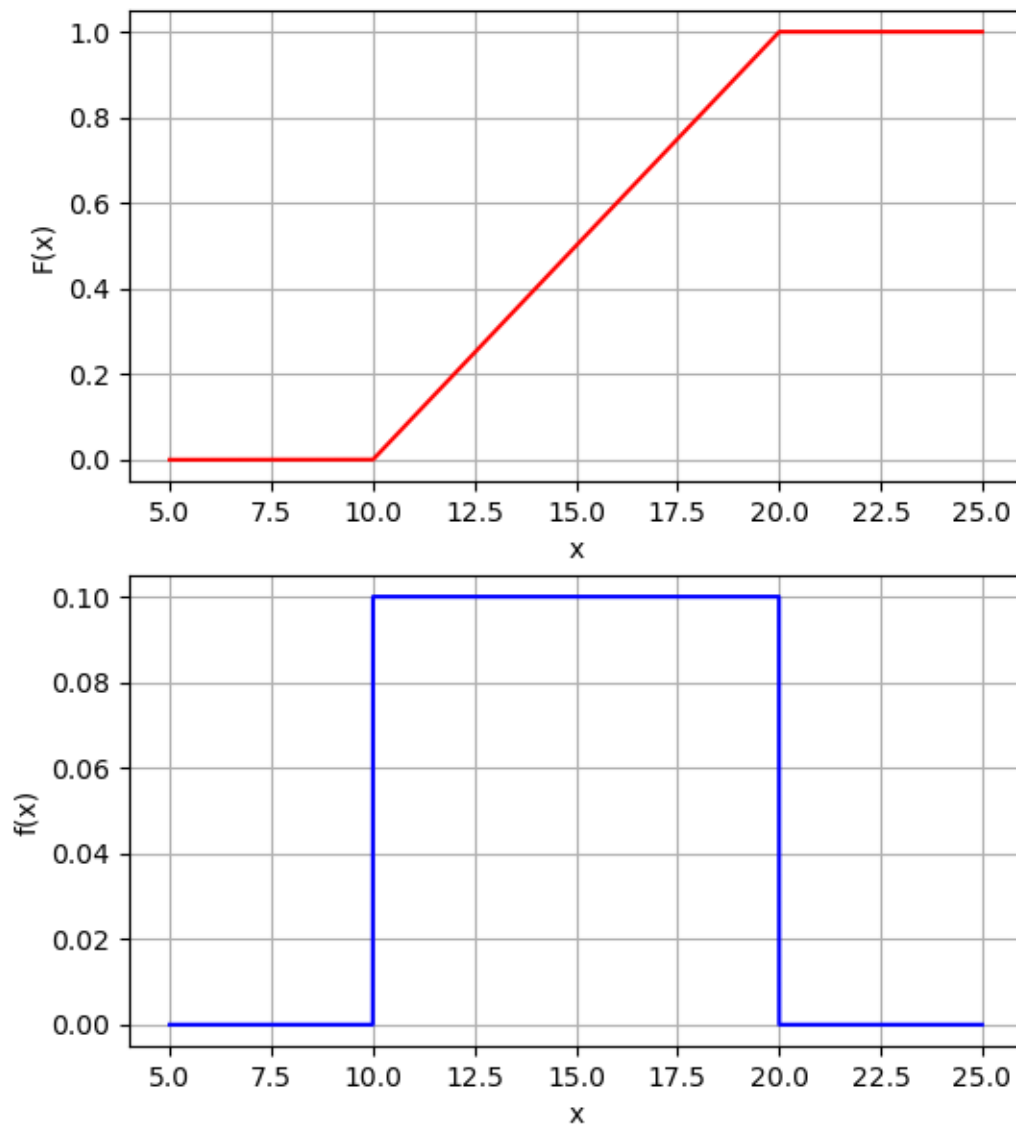


Рисунок 2 - графики функции распределения и плотности распределения равномерной случайной величины при $a = 10$, $b = 20$.

Нормальное распределение:

Нормальное распределение

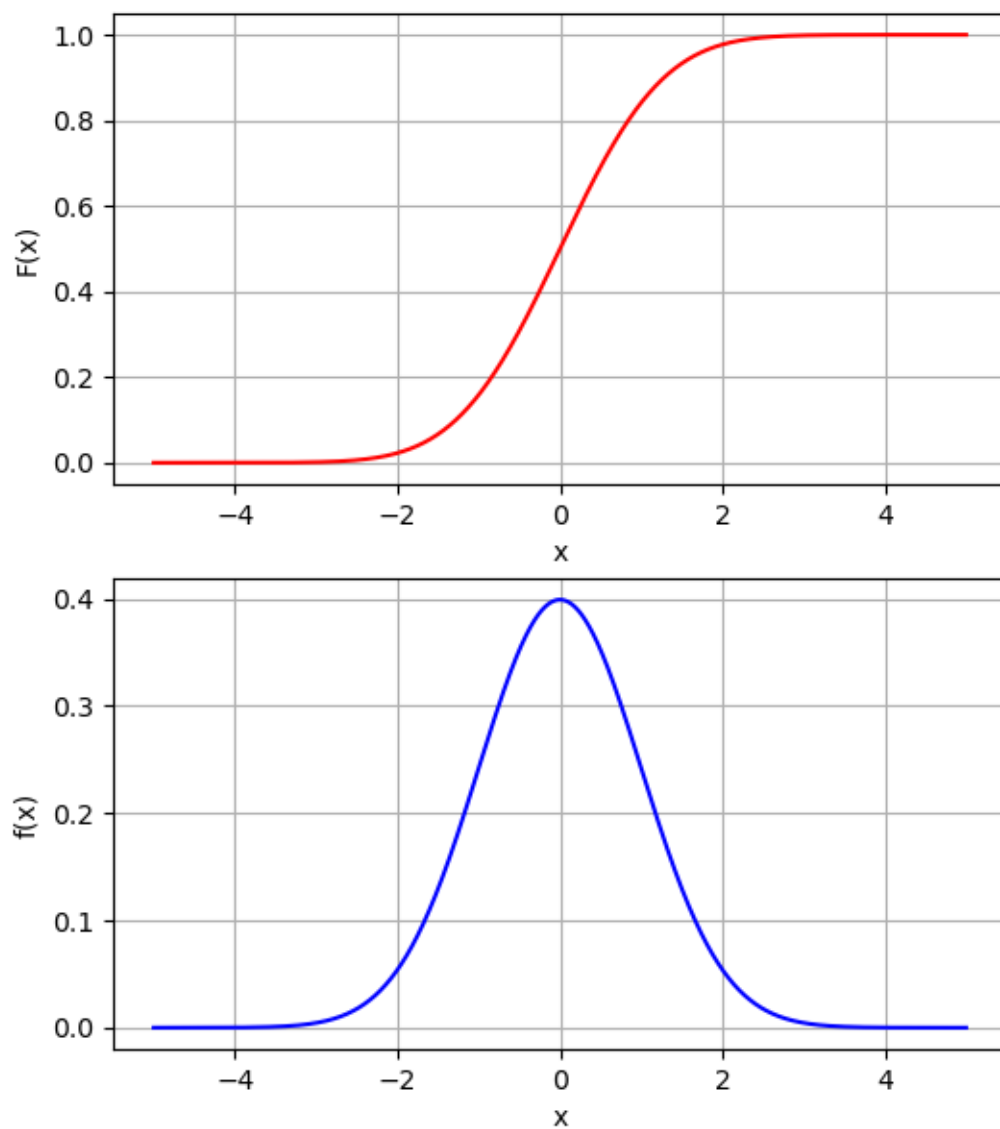


Рисунок 3 - графики функции распределения и плотности распределения нормальной случайной величины при $\mu = 0$, $\sigma = 1$.

Нормальное распределение

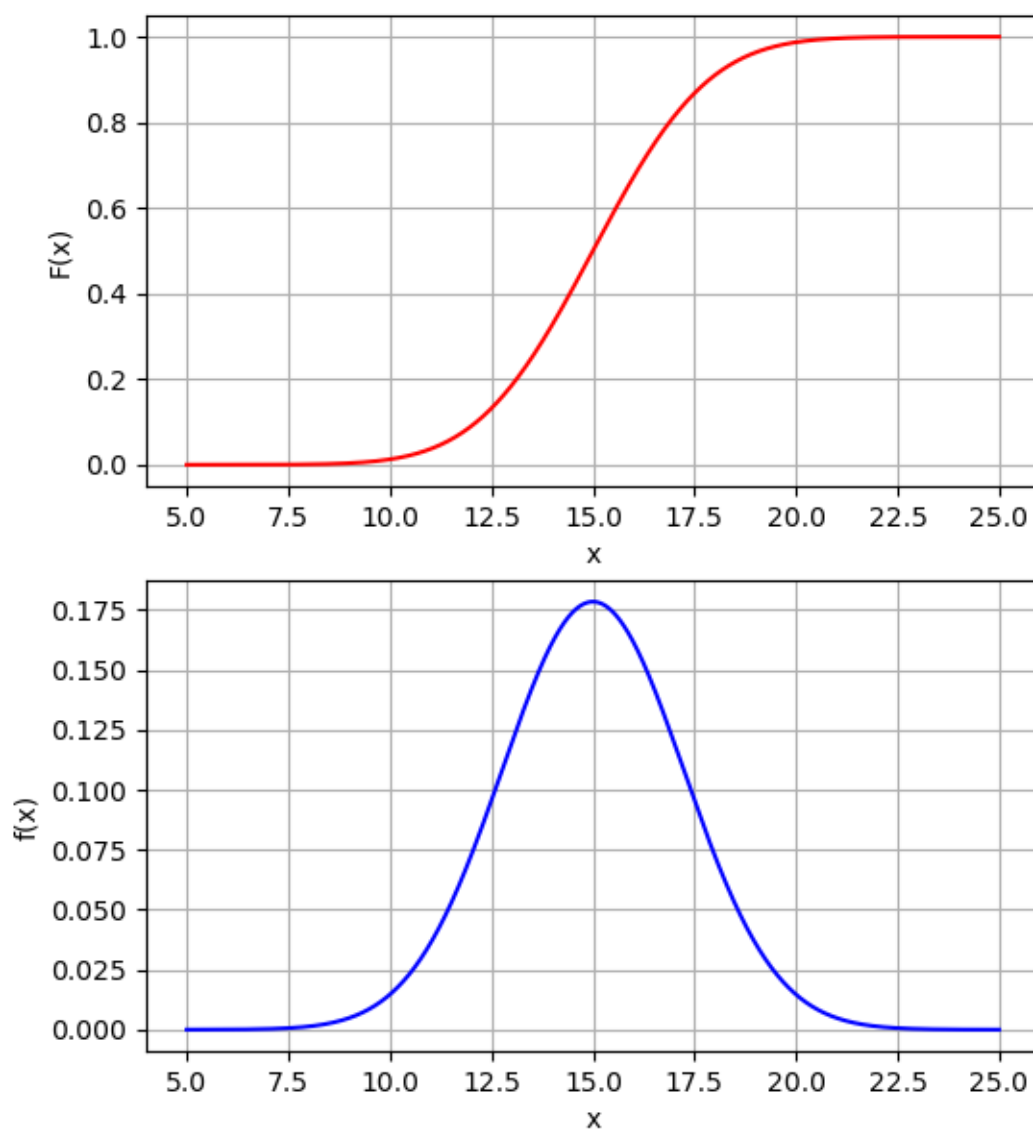


Рисунок 4 - графики функции распределения и плотности распределения нормальной случайной величины при $\mu = 15$, $\sigma = 5$.

Код программы

Main.py

```
import matplotlib.pyplot as plt
from math import sqrt
from scipy.stats import norm
import numpy as np

#Равномерное распределение
def ud_function(a, b, x):
    #return (x - a) / (b - a) if a <= x < b else 0 if x <
a else 1
    if (x < a):
        return 0
    if (x > b):
        return 1
    return (x - a) / (b - a)

def ud_density(a, b, x):
    if (a <= x <= b):
        return 1 / (b - a)
    return 0

#Нормальное распределение
def norm_function(x, mu, sigma):
    return norm.cdf(x, mu, sqrt(sigma))

def norm_density(x, mu, sigma):
    return norm.pdf(x, mu, sqrt(sigma))

def draw_graphics(x, y_function, y_density, name):
    fig, axs = plt.subplots(2, figsize=(6, 7))
    fig.suptitle(name)

    axs[0].plot(x, y_function, color='red')
    axs[1].plot(x, y_density, color='blue')

    axs[0].set_xlabel('x')
    axs[0].set_ylabel('F(x) ')

    axs[1].set_xlabel('x')
    axs[1].set_ylabel('f(x) ')

    axs[0].grid(True)
```

```

    axs[1].grid(True)

def main():
    print('Равномерное распределение:')
    a = float(input("Input a: "))
    b = float(input("Input b: "))
    print('Нормальное распределение:')
    mu = float(input("Input mu: "))
    sigma = float(input("Input sigma: "))

    delta = b - a
    x = np.arange(a - delta / 2, b + delta / 2, 0.001)
    y_function = [ud_function(a, b, _x) for _x in x]
    y_density = [ud_density(a, b, _x) for _x in x]
    draw_graphics(x, y_function, y_density, 'Равномерное
распределение')

    x = np.arange(mu - 5 * sigma, mu + 5 * sigma, 0.001)
    y_function = norm_function(x, mu, sigma)
    y_density = norm_density(x, mu, sigma)
    draw_graphics(x, y_function, y_density, 'Нормальное
распределение')

    plt.show()

if __name__ == '__main__':
    main()

```