Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики» (СибГУТИ)

Кафедра ВС

Лабораторная работа 1

по дисциплине «Моделирование»

Выполнил: ст. гр. ЗМП-41 Лёвкин И. А.

Проверил: Родионов А.С.

Новосибирск 2025

**Задание**

Построить датчик независимых, одинаково распределённых случайных величин по заданной плотности.

**Ход работы:**

**Проверка условия нормировки**

Плотность распределения:

Условие нормировки:

Вычисляем интегралы:

1. Для
2. Для

Решение:

**Функция распределения F(x)**

1. Для :
2. Для :

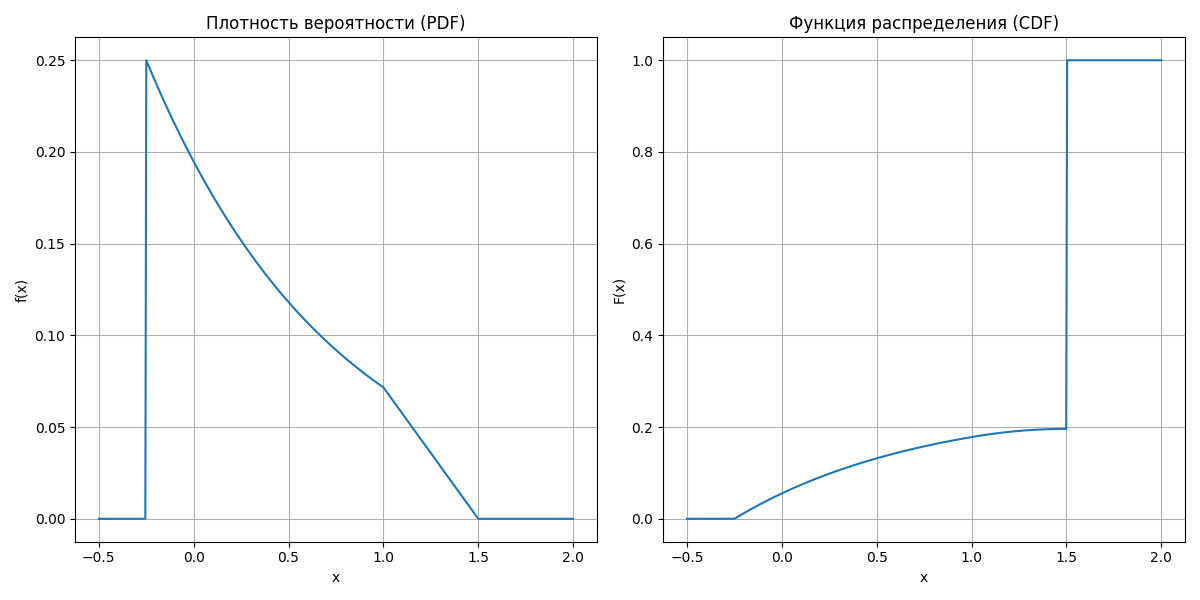
**Метод обратного преобразования**

Генерируем и находим

1. Если :
2. Если :

**Результаты:**

Задание 1

****

**Листинг**

import numpy as np

import matplotlib.pyplot as plt

# Константа для экспоненциальной части

def exp\_part(x):

return 0.25 \* np.exp(-(x + 0.25))

# Находим значение f(1) из экспоненциальной части

f1 = exp\_part(1)

# Находим коэффициент k так, чтобы f(x) была непрерывна в точке x = 1

# f1 = k \* (1 - 1.5) => k = -f1 / 0.5

k = -2 \* f1

def pdf(x):

if -0.25 < x < 1:

return exp\_part(x)

elif 1 < x < 1.5:

return k \* (x - 1.5)

else:

return 0.0

def cdf(x):

if x <= -0.25:

return 0.0

elif x >= 1.5:

return 1.0

else:

xs = np.linspace(-0.25, x, 1000)

dx = xs[1] - xs[0]

return np.sum([pdf(xi) for xi in xs]) \* dx

# Векторизация

pdf\_vec = np.vectorize(pdf)

cdf\_vec = np.vectorize(cdf)

# Графики

x\_vals = np.linspace(-0.5, 2, 500)

pdf\_vals = pdf\_vec(x\_vals)

cdf\_vals = cdf\_vec(x\_vals)

plt.figure(figsize=(12, 6))

# PDF

plt.subplot(1, 2, 1)

plt.plot(x\_vals, pdf\_vals)

plt.title("Плотность вероятности (PDF)")

plt.xlabel("x")

plt.ylabel("f(x)")

plt.grid(True)

plt.subplot(1, 2, 2)

plt.plot(x\_vals, cdf\_vals)

plt.title("Функция распределения (CDF)")

plt.xlabel("x")

plt.ylabel("F(x)")

plt.grid(True)

plt.tight\_layout()

plt.show()

print(k)