

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Студент Заборовская А. Д.

Группа ИУ7-73Б

Оценка _____

Преподаватель Рудаков И. В.

Москва — 2022 г.

1 | Задание

Разработать программу для построения графиков функции распределения и функции плотности распределения для следующих видов распределений:

- равномерного распределения;
- экспоненциального распределения.

Реализовать графический интерфейс, который будет давать пользователю возможность выбора вида распределения и настройки его параметров.

2 | Реализация

2.1 Равномерное распределение

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

Обозначение равномерного распределения: $X \sim R(a, b)$, $a, b \in R$.

Функция распределения:

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

Функция плотности распределения:

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

На листинге 2.1 приведены функции для вычисления значений и построения функций для равномерного распределения.

Листинг 2.1: Функции для равномерного распределения

```
1 def uni_func_pl(x, a, b):
2     if x < a or x > b:
3         return 0
4     else:
5         return 1 / (b - a)
6
7 def uni_func(x, a, b):
8     if x <= a:
9         return 0
10    elif x > b:
11        return 1
12    else:
13        return (x - a) / (b - a)
```

```

14
15 def begin_uni(entry_a, entry_b):
16     try:
17         a = float(entry_a.get())
18         b = float(entry_b.get())
19     except:
20         tk.messagebox.showerror(title='Ошибка_ввода',
21                                 message='Введите_корректные_данные')
22         return
23
24     if (a > b):
25         tk.messagebox.showerror(title='Ошибка_ввода',
26                                 message='Левая_граница_должна_быть_меньше_правой')
27         return
28     xu1, xu2, xu3 = [], [], []
29     xu = []
30     yu = []
31     yup1, yup2, yup3 = [], [], []
32     gr1 = a - (b - a) / 2
33     gr2 = b + (b - a) / 2
34     i = gr1
35
36     while i < gr2:
37         if i < a:
38             xu1.append(i)
39         elif i > b:
40             xu3.append(i)
41         else:
42             xu2.append(i)
43         xu.append(i)
44         i += (gr2 - gr1) / 100
45     for i in range(len(xu)):
46         yu.append(uni_func(xu[i], a, b))
47
48     for i in range(len(xu1)):
49         yup1.append(uni_func_pl(xu1[i], a, b))
50
51     for i in range(len(xu2)):
52         yup2.append(uni_func_pl(xu2[i], a, b))
53
54     for i in range(len(xu3)):
55         yup3.append(uni_func_pl(xu3[i], a, b))
56
57     fig, ax = plt.subplots(figsize=(5, 3))
58     ax.grid()
59     plt.plot(xu1, yup1, color='m')
60     plt.plot(xu2, yup2, color='m')

```

```

61 plt.plot(xu3, yu3, color='m')
62 plt.title('Функция_плотности_распределения')
63
64 fig2, ax2 = plt.subplots(figsize=(5, 3))
65 ax2.grid()
66 plt.plot(xu, yu)
67 plt.title('Функция_распределения')
68
69 plt.show()

```

2.2 Экспоненциальное распределение

Экспоненциальное распределение – это абсолютно непрерывное распределение, моделирующее время между двумя последовательными свершениями одного и того же события.

Обозначение экспоненциального распределения: $X \sim Exp(\gamma), \gamma > 0$

Функция распределения:

$$F(x) = \begin{cases} 0 & \text{при } x \leq 0 \\ 1 - \exp^{-\gamma x} & \text{при } x > 0 \end{cases}$$

Функция плотности распределения:

$$f(x) = \begin{cases} 0 & \text{при } x < 0 \\ \gamma \exp^{-\gamma x} & \text{при } x \geq 0 \end{cases}$$

На листинге 2.2 приведены функции для вычисления значений и построения функций для экспоненциального распределения.

Листинг 2.2: Функции для экспоненциального распределения

```

1 def exp_func(x, gamma):
2     gamma = -gamma
3     if x <= 0:
4         return 0
5     else:
6         return 1 - exp(x * gamma)
7
8 def exp_func_pl(x, gamma):
9     if x < 0:
10        return 0
11    else:
12        return gamma * exp(x * (-gamma))
13
14 def begin_exp(entry_al):
15     try:
16        gamma = float(entry_al.get())

```

```

17     except:
18         tk.messagebox.showerror( title='Ошибка_ввода',
19                                 message='Введите_корректные_данные')
20         return
21
22     if gamma <= 0:
23         tk.messagebox.showerror( title='Ошибка_ввода',
24                                 message='Параметр_должен_быть_больше_0')
25         return
26     xe = []
27     ye = []
28     yep = []
29     i = 0
30     gr2 = 10 / gamma
31
32     while i < gr2:
33         xe.append(i)
34         i += (gr2 + 1) / 100
35     for i in range(len(xe)):
36         ye.append(exp_func(xe[i], gamma))
37         yep.append(exp_func_pl(xe[i], gamma))
38
39     fig, ax = plt.subplots(figsize=(5, 3))
40     ax.grid()
41     plt.plot(xe, yep, color='m')
42     plt.title('Функция_плотности_распределения')
43
44     fig2, ax2 = plt.subplots(figsize=(5, 3))
45     ax2.grid()
46     plt.plot(xe, ye)
47     plt.title('Функция_распределения')
48
49     plt.show()

```

3 | Примеры работы

На рисунках 3.1 и 3.2 приведены примеры работы программы.

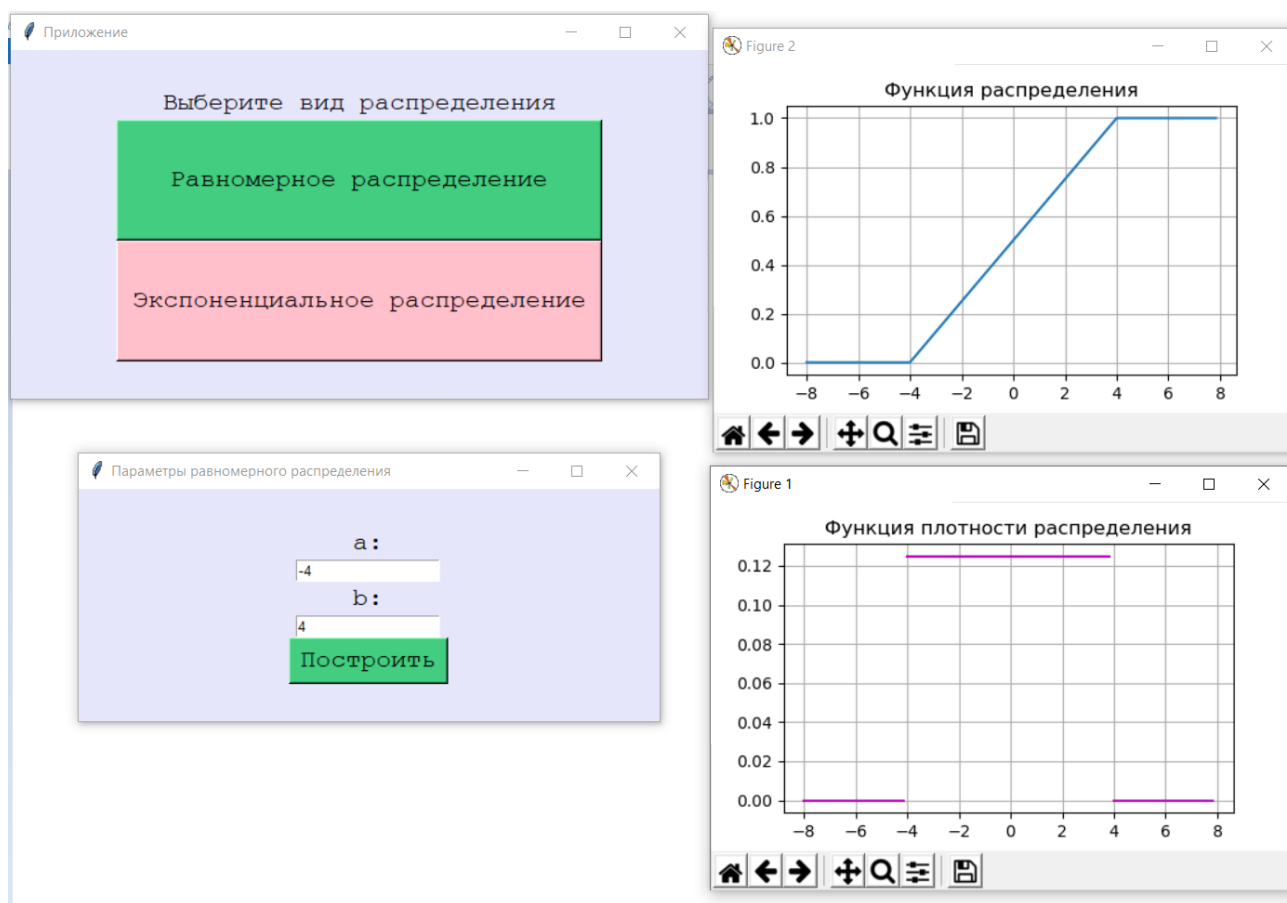


Рис. 3.1: Пример работы для равномерного распределения

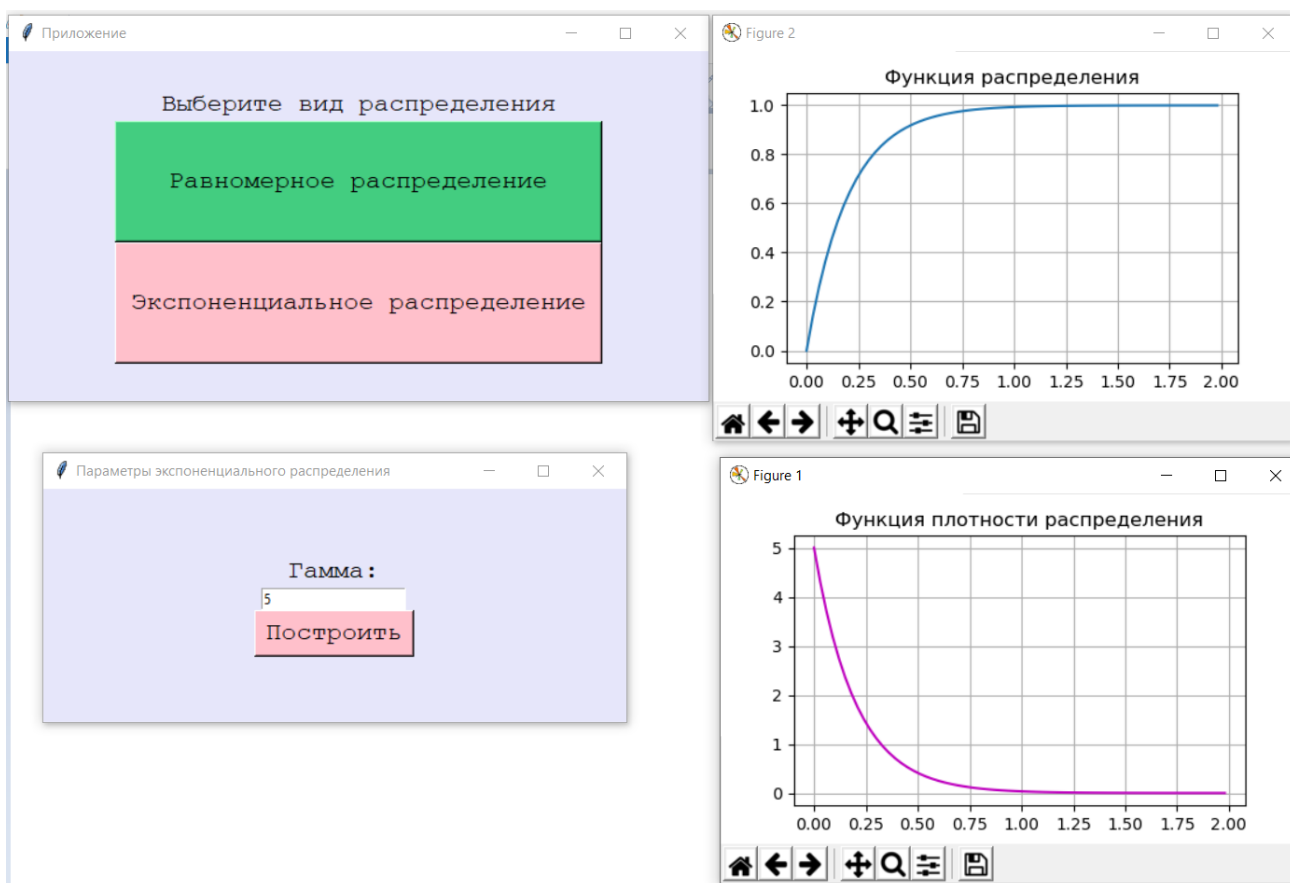


Рис. 3.2: Пример работы для экспоненциального распределения