



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования

**«Дальневосточный федеральный университет»
(ДФУ)**

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

ЛАБОРАТОРНАЯ РАБОТА №3

По курсу «Теория вероятностей и математическая статистика»

Студент группы Б9123-01.03.02ии
Моттуева Уруйдана Михайловна

г. Владивосток

2025

Ход работы:

1. Краткое математическое описание

Имеется выборка X_1, X_2, \dots, X_n из X .

Эмпирическая функция распределения $\hat{F}_n(x)$ определяется как:

$$\hat{F}_n(x) = \frac{1}{n} \sum_{i=1}^n I(X_i \leq x),$$

где $I(X_i \leq x)$ – индикаторная функция, равная 1, если $X_i \leq x$, и 0 в противном случае.

Доверительный интервал для ЭФР с уровнем доверия $1 - \alpha$ можно оценить как:

$$P(U(x) < F_X(x) < L(x)) \geq 1 - \alpha,$$

где

$$L(x) = \max(\hat{F}_n(x) - \varepsilon_n; 0),$$

$$U(x) = \min(\hat{F}_n(x) + \varepsilon_n; 1),$$

$$\varepsilon_n = \sqrt{\frac{1}{2n} \ln\left(\frac{2}{\alpha}\right)}.$$

Для 95%-го доверительного интервала используется приближение:

$$\varepsilon_n = \sqrt{\frac{1}{2n} \ln\left(\frac{2}{\alpha}\right)} = \sqrt{\frac{\ln(2 / 0.05)}{2n}} = \sqrt{\frac{\ln(40)}{2n}}.$$

В Python реализация без использования `scipy.stats`:

```
1. def empirical_distrib_func(data, x):
2.     n = len(data)
3.     count_less_or_equal = sum(1 for value in data if value <= x)
4.     EFR = count_less_or_equal / n
5.
6.     epsilon = np.sqrt(np.log(40) / (2 * n))
7.     lower_bound = max(0, EFR - epsilon)
8.     upper_bound = min(1, EFR + epsilon)
9.     return EFR, lower_bound, upper_bound
```

где строки 2-4 это определение ЭФР, 6-8 – поиск доверительного интервала.

В Python реализация ЭФР с использованием `scipy.stats`:

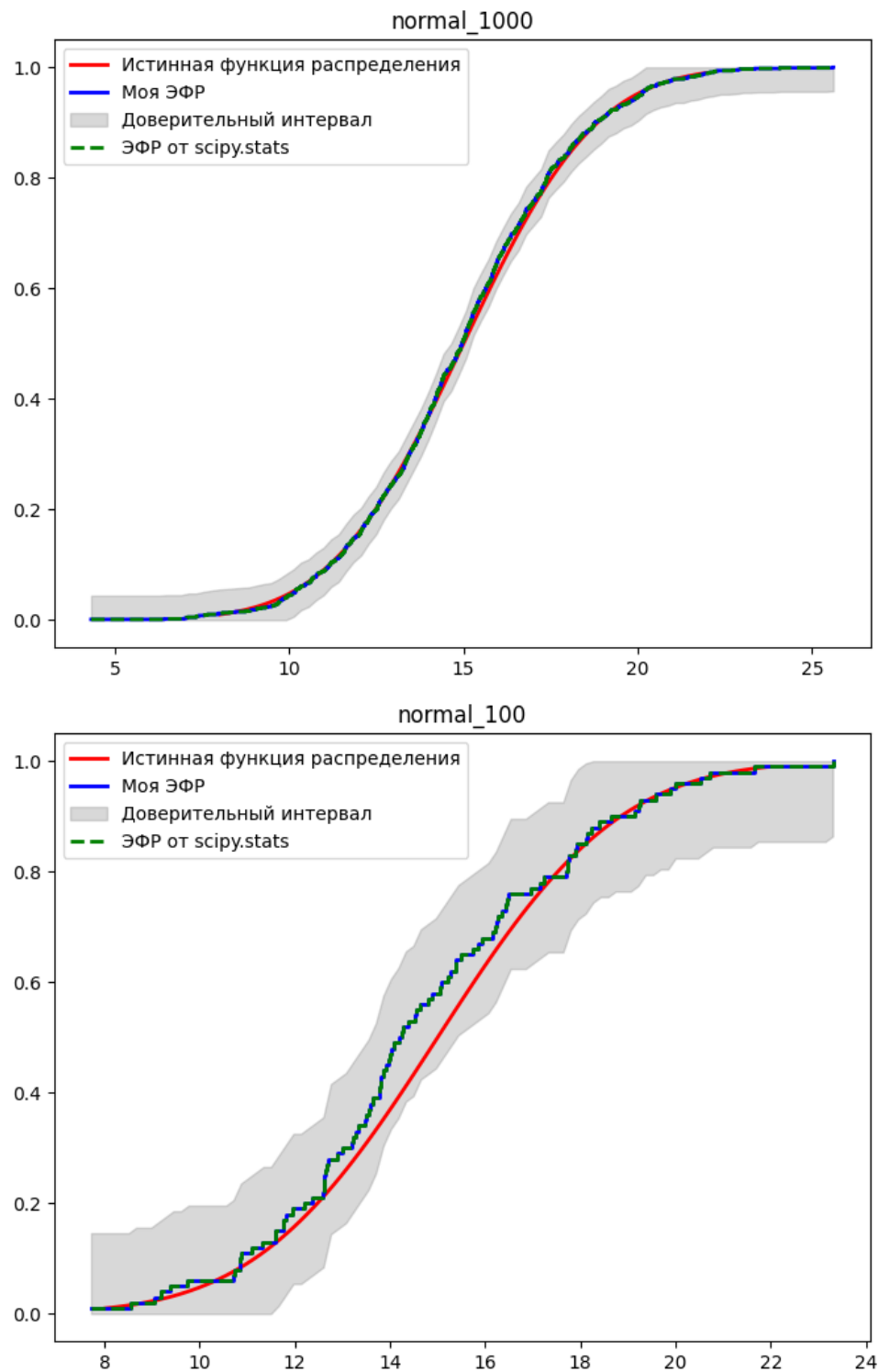
```
1. def built_in_ecdf(data, x):
2.     ecdf_obj = ECDF(data)
3.     EFR = ecdf_obj(x)
4.
5.     n = len(data)
6.     std_error = np.sqrt(EFR * (1 - EFR) / n)
7.
8.     trust_int = t.interval(1 - 0.05, df=n-1, loc=EFR, scale=std_error)
9.
```

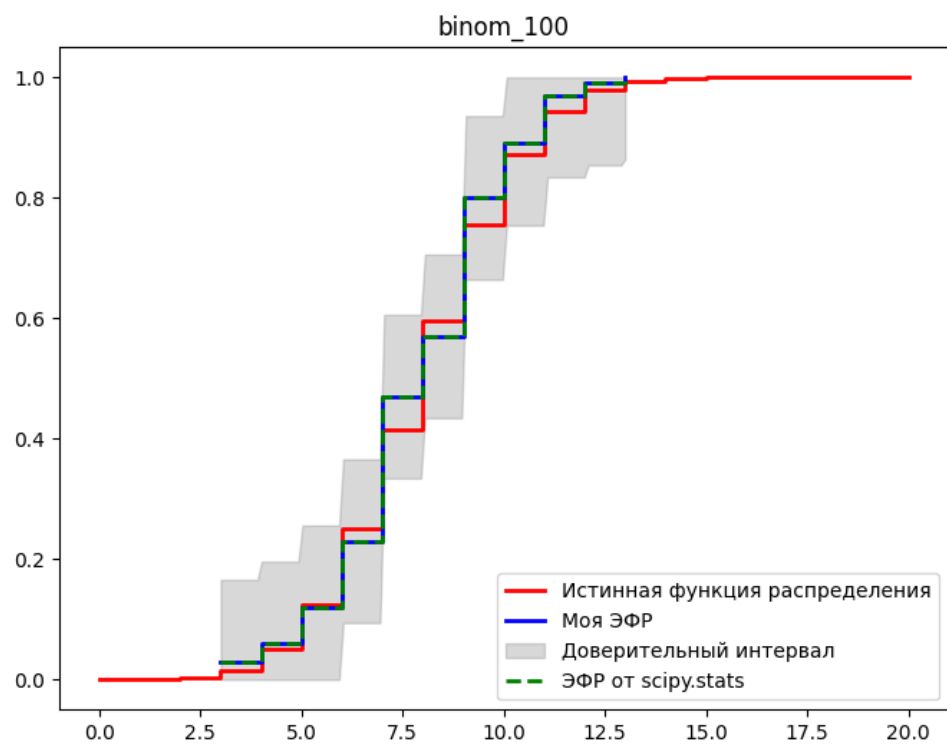
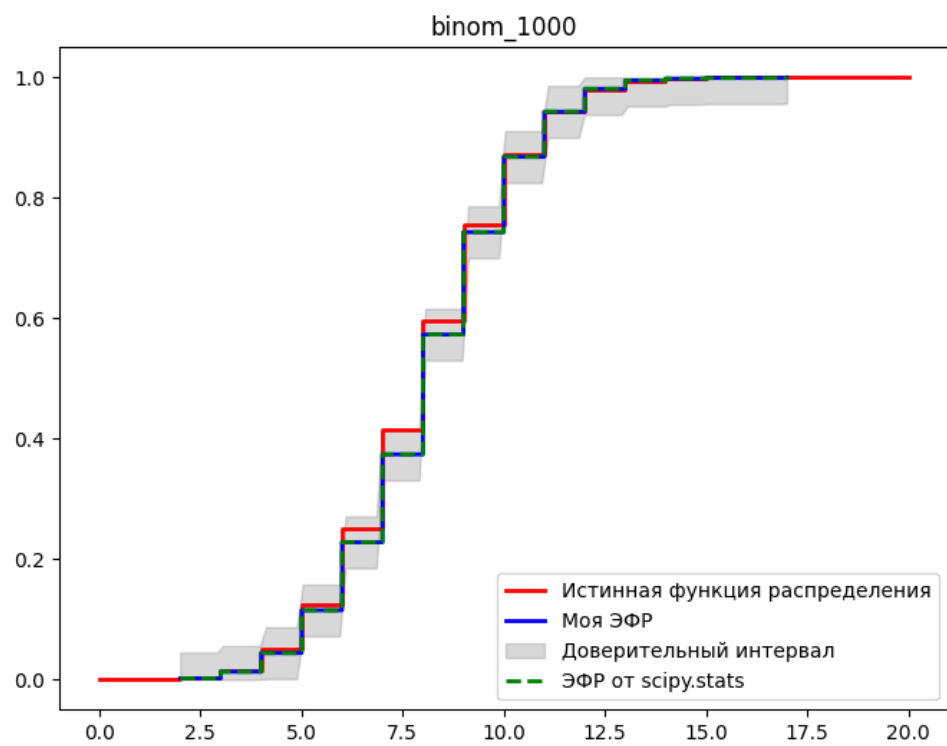
```

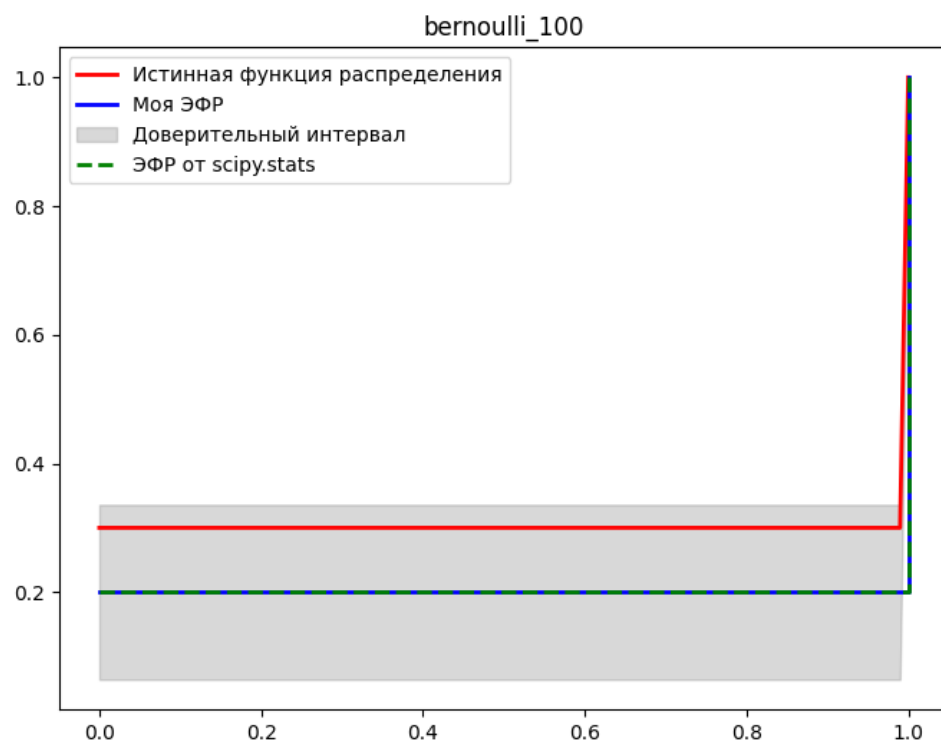
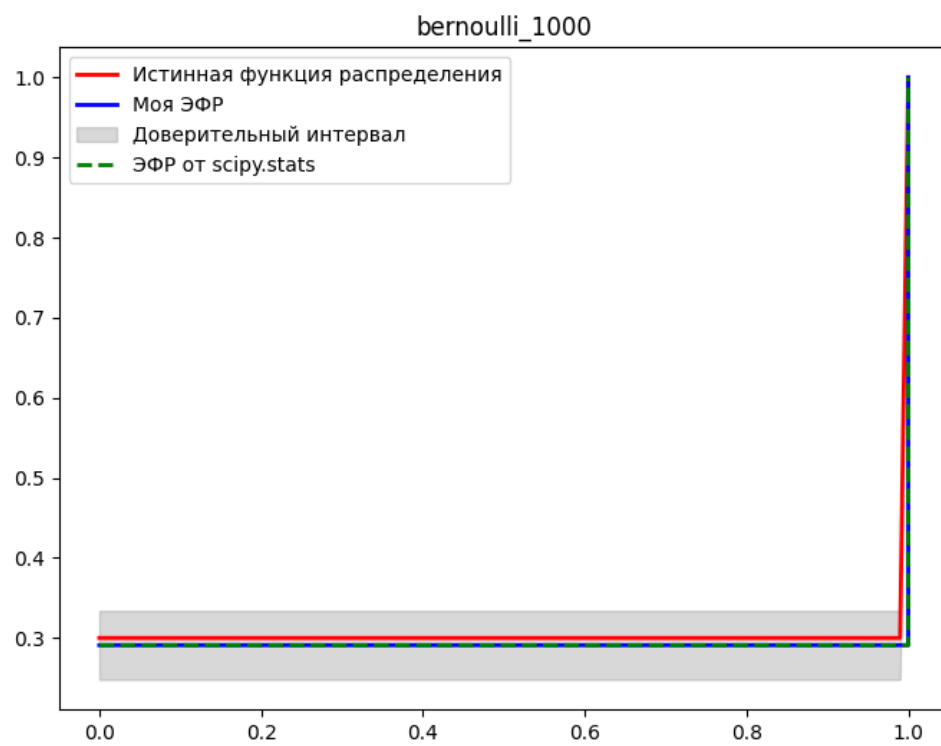
10.         lower_bound = max(0, trust_int[0])
11.         upper_bound = min(1, trust_int[1])
12.
13.         return EFR, lower_bound, upper_bound
14.

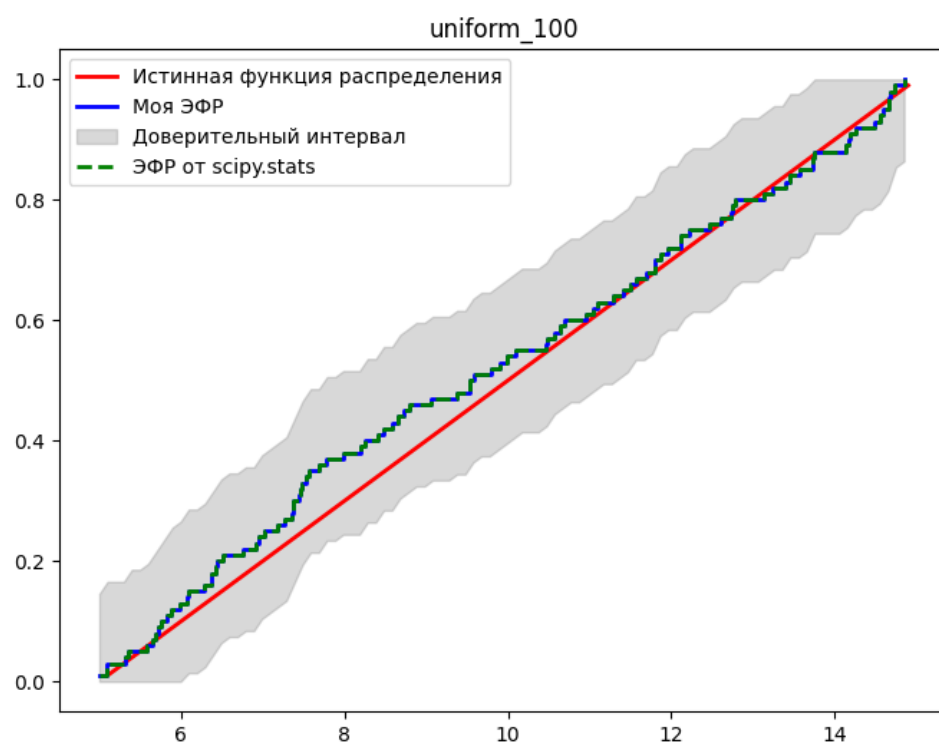
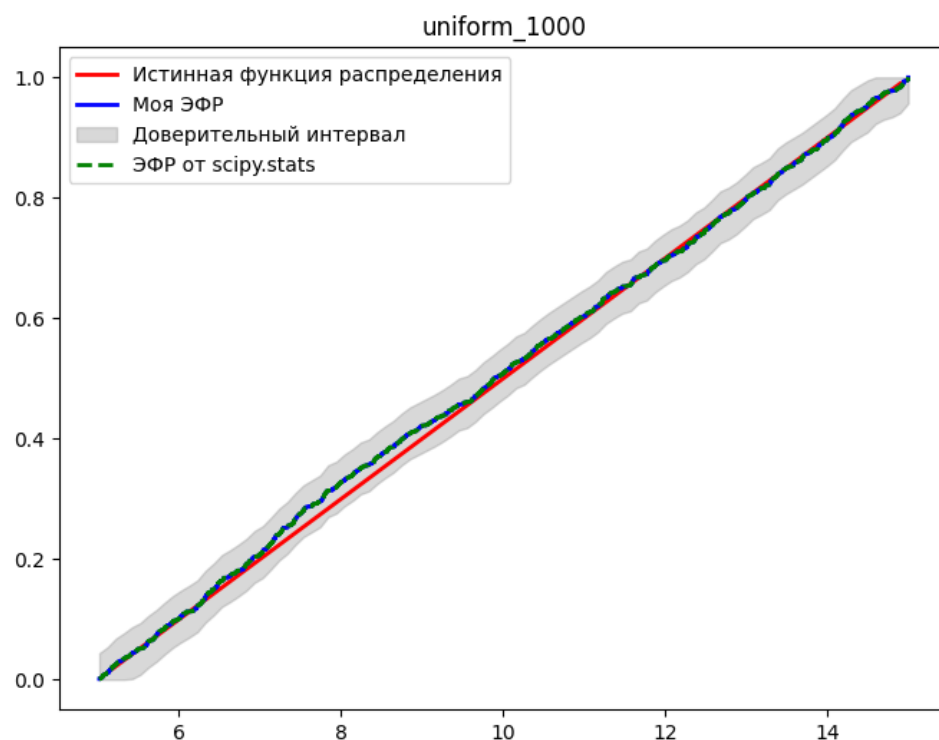
```

2. Выходные данные









https://colab.research.google.com/drive/1EPjb1T1TRLe_F5U0u8RtHi7Ay_GgEkPE?usp=sharing – ссылка на колаб с полным кодом программы