

```
import pandas as pd
import numpy as np
import scipy.stats as st
```

Лабораторная работа №2

Задание 2

{Фамилия Имя}, {Номер группы}, Вариант {Номер варианта}, ({Дата})

По данным выборки, удовлетворяющей нормальному закону распределения со средним квадратическим отклонением S , вычислить:

1. выборочное среднее;
2. доверительный интервал для математического ожидания при доверительной вероятности γ .

Данные

```
LB_NUM = 2
LB_SURNAME = 'Фамилия'
LB_NAME = "Имя"
LB_VARIANT = 23
LB_EXERCISE_NUM = 2
print(f'Фамилия, Имя: {LB_SURNAME}, {LB_NAME}\nНомер лабораторной работы: {LB_NUM}\nВариант лабораторной работы: {LB_VARIANT}\nНомер задания лабораторной работы: {LB_EXERCISE_NUM}')
```

Фамилия, Имя: Фамилия, Имя
Номер лабораторной работы: 2
Вариант лабораторной работы: 23
Номер задания лабораторной работы: 2

```
data = pd.Series([21.3, 26.5, 39.5, 17.5, 27.5, 38.2, 29.0, 30.1, 27.5, 32.7, 19.4, 22.2, 36.4, 37.3, 25.0, 25.7, 31.8, 24.1, 30.0, 27.4, 27.0, 22.9, 30.0, 28.2, 23.9, 18.1, 18.9, 26.9, 27.8, 16.5, 32.6, 33.4, 26.3, 20.4, 24.9, 28.7, 41.5, 22.4, 20.5, 35.3, 14.1, 34.4, 35.0, 23.7, 25.2, 31.5, 37.2, 20.7, 33.2, 28.0, 22.0, 31.7, 23.4, 30.8, 35.0, 31.4, 28.6,
```

```
13.9, 34.6, 31.4, 30.8, 19.4, 29.3, 27.3, 37.6, 24.6, 32.4, 25.4,  
27.7, 28.6, 33.7, 26.9, 29.4, 28.2, 32.9, 29.5,  
20.5, 26.5, 32.2, 32.9, 26.4, 28.9, 29.2, 32.6, 23.2, 31.7, 17.8,  
24.9, 28.5, 35.6, 27.2, 29.1, 29.9, 27.2, 31.2,  
27.8, 34.1, 29.0, 26.5, 24.8])  
confidence_prob = .99  
data
```

```
0      21.3  
1      26.5  
2      39.5  
3      17.5  
4      27.5  
...  
95     27.8  
96     34.1  
97     29.0  
98     26.5  
99     24.8  
Length: 100, dtype: float64
```

```
print(f"Доверительная вероятность: {confidence_prob}")
```

Доверительная вероятность: 0.99

```
data.describe()
```

```
count      100.000000  
mean       28.025000  
std        5.594954  
min        13.900000  
25%        24.875000  
50%        28.200000  
75%        31.725000  
max        41.500000  
dtype: float64
```

Всего элементов ряда

```
data_len = len(data)
```

```
data_len
```

```
100
```

Выборочное среднее

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

```
mean = np.mean(data)
mean
```

```
28.025
```

Выборочная дисперсия

$$s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

```
var = np.var(data, ddof=1)
var
```

```
31.303510101010097
```

Среднеквадратическое отклонение

$$s = \sqrt{s^2}$$

```
std = np.std(data, ddof=1)
std
```

```
5.594953985602571
```

Доверительный интервал для математического ожидания при неизвестной генеральной дисперсии σ

$$\left(\bar{x} - t(\gamma, n-1) \cdot \frac{s}{\sqrt{n}}, \quad \bar{x} + t(\gamma, n-1) \cdot \frac{s}{\sqrt{n}} \right)$$

Считаем интервал без модуля stats

```
def confidence_interval_for_expectation_t(data, confidence_prob):
    mean = np.mean(data)
    std = np.std(data, ddof=1)
    n = len(data)
    # t.ppf считает левостороннюю область, а нам требуется двусторонняя
    alpha = (1 - confidence_prob) / 2
    gamma = 1 - alpha
    t_value = st.t.ppf(gamma, n - 1)
    delta = t_value * std / np.sqrt(n)
    return (mean - delta, mean + delta)

confidence_interval_for_expectation_t(data, confidence_prob)
```

```
(26.555538232398934, 29.494461767601063)
```

Считаем интервал с модулем stats

```
interval_expectation = st.t.interval(confidence_prob, df=data_len - 1,
loc=mean, scale=st.sem(data))
interval_expectation
```

```
(26.555538232398934, 29.494461767601063)
```

```
print(f"Вывод: Таким образом с вероятностью {confidence_prob} можно
утверждать, \пчто математическое ожидание лежит в интервале
{interval_expectation}")
```

Вывод: Таким образом с вероятностью 0.99 можно утверждать,
что математическое ожидание лежит в интервале (26.555538232398934,
29.494461767601063)