

Федеральное государственное автономное образовательное учреждение
высшего образования

Национальный Исследовательский Университет ИТМО



Теория вероятностей
Практическая работа № 5
Вариант 20

Выполнил: Кравцов Кирилл Денисович Р3211

Преподаватель: Селина Елена Григорьевна

г. Санкт-Петербург

2023 г.

Исходный код программы:

```
main.py

import pandas as pd
import numpy as np

import logger
import plots

def read_sample():
    return np.loadtxt('sample5.txt', delimiter=',')

def expected_value(variation_series):
    expected_value = 0
    for value in variation_series:
        expected_value += value / n
    return expected_value

def dispersion(variation_series, expected_value):
    dispersion = 0
    for value in variation_series:
        delta = (value - expected_value)
        dispersion += delta ** 2 / n
    return dispersion

def standard_deviation_library(variation_series):
    return np.std(variation_series)

def statistical_series(variation_series):
    x, counts = np.unique(variation_series,
return_counts=True)
    p = counts / len(variation_series)
    F_x = np.cumsum(p)
    data = {'x': x, 'Количество': counts, 'Частоты': p,
'F(x)': F_x}
    return pd.DataFrame(data)

def fill_bins_and_frequencies(bins, frequencies):
    for i in range(m):
        count = 0
        left = x_start + h * i
        right = x_start + h * (i + 1)
        for val in variation_series:
            if left <= val < right:
                count += 1
        frequencies.append(count / n / h)
        bins.append((left + right) / 2)
```

```

# Выборка
sample = read_sample()
n = sample.size
logger.log_sample(sample, n)

# 1. Вариационный ряд
variation_series = np.sort(sample)
logger.log_variation_series(variation_series)

# 2. Экстремальные значения и размах
extreme_1 = variation_series[0]
extreme_n = variation_series[n - 1]
variation_size = extreme_n - extreme_1
logger.log_extremes(extreme_1, extreme_n, variation_size)

# 3. Оценки математического ожидания и среднеквадратичного
отклонения
# а) Математическое ожидание
expected_value = expected_value(variation_series)
logger.log_expected_value(expected_value)

# б) Среднеквадратичное отклонение
dispersion = dispersion(variation_series, expected_value)
standard_deviation = dispersion ** (1 / 2)
fixed_dispersion = dispersion * n / (n - 1)
logger.log_standard_deviation(standard_deviation)
logger.log_dispersions(dispersion, fixed_dispersion)

# 4. Эмпирическая функция распределения и её график
df = statistical_series(variation_series)
logger.log_statistical_series(df)
plots.show_statistical_series(df)

# 5. Построение гистограммы и полигона приведенных частот
группированной выборки
m = int(1 + np.log2(n)) + 1 # количество интервалов разбиения
h = variation_size / m      # ширина шага
x_start = extreme_1 - h / 2 # определение начала интервала

bins = [] # инициализация массивов интервалов
frequencies = [] # инициализация количества вхождений
fill_bins_and_frequencies(bins, frequencies)
plots.show_bar_chart(bins, frequencies, h)

# полигон частотностей
for i in range(len(frequencies)):
    frequencies[i] *= h
plots.show_polygon_1(bins, frequencies)

```

```
# ПОЛИГОН ЧАСТОТ
for i in range(len(frequencies)):
    frequencies[i] *= n
plots.show_polygon_2(bins, frequencies)
```

logger.py

```
def log_sample(sample, n):
    print('Выборка:', sample)
    print('Размер:', n)
    print()

def log_variation_series(variation_series):
    print('Вариационный ряд')
    print(variation_series)
    print()

def log_extremes(extreme_1, extreme_n, variation_size):
    print('Первая порядковая статистика  $x(1)$  =', extreme_1)
    print('n-ая порядковая статистика  $x(n)$  =', extreme_n)
    print('Размах выборки  $w^*$  =', variation_size)
    print()

def log_expected_value(expected_value):
    print('Математическое ожидание  $M$  =', expected_value)
    print()

def log_standard_deviation(standard_deviation):
    print('Среднеквадратичное отклонение  $\sigma$  =',
standard_deviation)
    print()

def log_dispersions(dispersion, fixed_dispersion):
    print('Выборочная дисперсия  $dispersion$  =', dispersion)
    print('Исправленное СКО  $s^2 =$ :', fixed_dispersion)
    print()

def log_statistical_series(df):
    print(df[['x', 'F(x)']])
    print()
```

```
plots.py
```

```
import matplotlib.pyplot as plt
```

```
def show_statistical_series(df):  
    plt.step(df['x'], df['F(x)'], where='post')  
    plt.title('Эмпирическая функция распределения')  
    plt.xlabel('x')  
    plt.ylabel('F(x)')  
    plt.show()
```

```
def show_bar_chart(bins, frequencies, h):  
    plt.bar(bins, frequencies, width=h, edgecolor='black')  
    plt.title('Гистограмма частот')  
    plt.xlabel('x')  
    plt.ylabel('p*_i / h')  
    plt.show()
```

```
def show_polygon_1(bins, frequencies):  
    plt.plot(bins, frequencies, marker='*')  
    plt.xlabel('x')  
    plt.ylabel('p*_i')  
    plt.title('Полигон приведенных частот')  
    plt.show()
```

```
def show_polygon_2(bins, frequencies):  
    plt.plot(bins, frequencies, marker='*')  
    plt.xlabel('x')  
    plt.ylabel('n_i')  
    plt.title('Полигон приведенных частот')  
    plt.show()
```

Результат работы программы:

Выборка: [-1.73 -0.73 1.66 -0.8 0.62 1.52 1.63 1.04
0.42 -1.21 0.9 -1. 0.24 0.62 0.55 -1.45 -1.45 -0.52 0.17
-1.31]

Размер: 20

Вариационный ряд

[-1.73 -1.45 -1.45 -1.31 -1.21 -1. -0.8 -0.73 -0.52 0.17
0.24 0.42 0.55 0.62 0.62 0.9 1.04 1.52 1.63 1.66]

Первая порядковая статистика $x(1) = -1.73$

n -ая порядковая статистика $x(n) = 1.66$

Размах выборки $w^* = 3.3899999999999997$

Математическое ожидание $M = -0.041499999999999926$

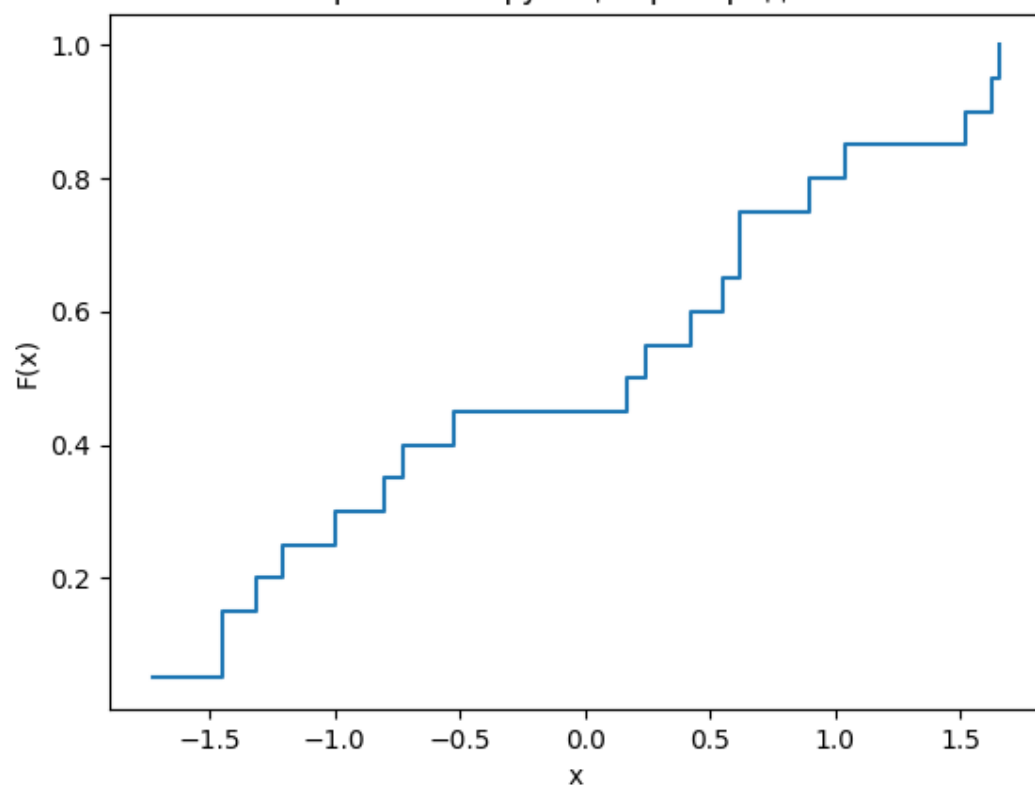
Среднеквадратичное отклонение $\sigma = 1.0893955893062905$

Выборочная дисперсия $\text{dispersion} = 1.1867827499999999$

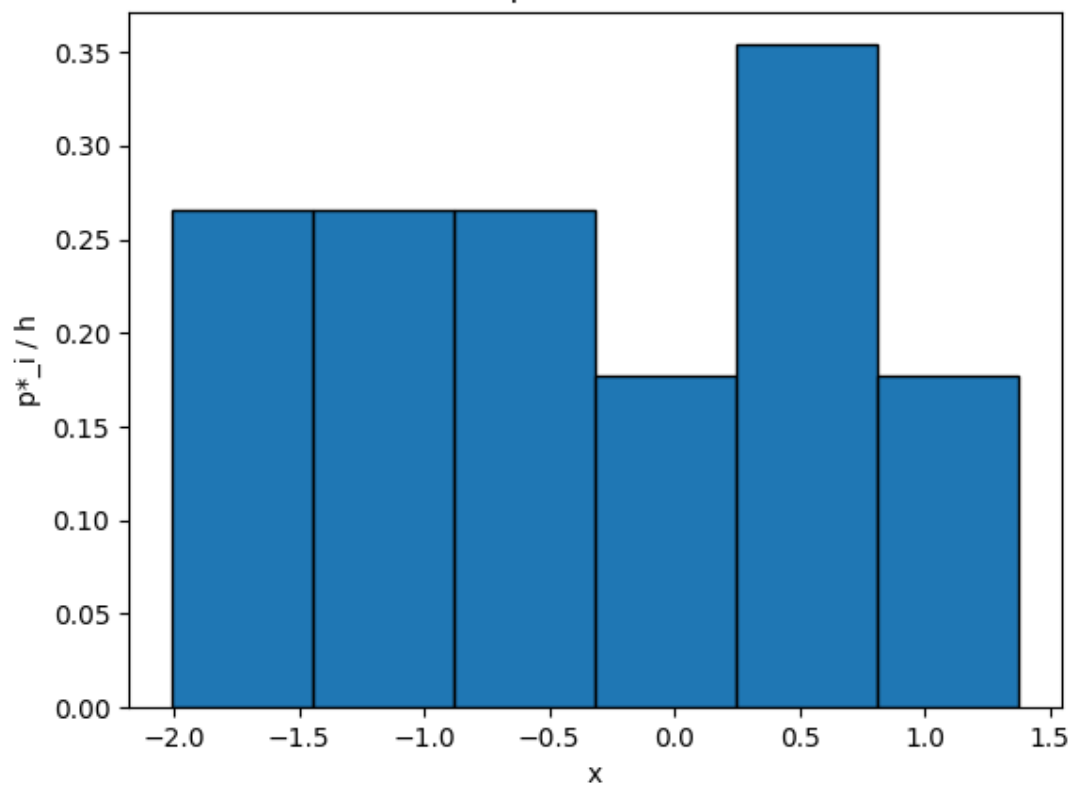
Исправленное СКО $s^2 = : 1.249245$

x F(x)			x F(x)		
0	-1.73	0.05	9	0.24	0.55
1	-1.45	0.15	10	0.42	0.60
2	-1.31	0.20	11	0.55	0.65
3	-1.21	0.25	12	0.62	0.75
4	-1.00	0.30	13	0.90	0.80
5	-0.80	0.35	14	1.04	0.85
6	-0.73	0.40	15	1.52	0.90
7	-0.52	0.45	16	1.63	0.95
8	0.17	0.50	17	1.66	1.00

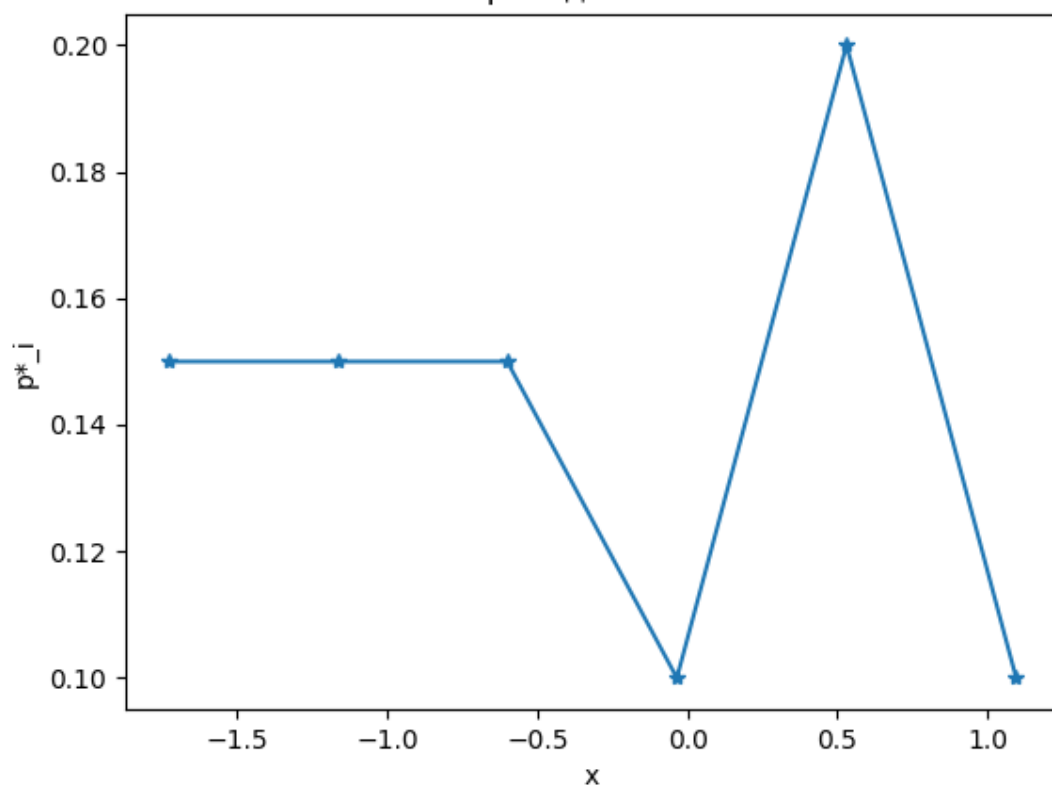
Эмпирическая функция распределения



Гистограмма частот



Полигон приведенных частот



Полигон приведенных частот

