Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 «Программная инженерия» –

Системное и прикладное программное обеспечение

**Отчёт**

**По практической работе №5**

**По теории вероятности**

**Вариант: 3**

Выполнил:

студент 2 курса

Батманов Даниил Евгеньевич

Группа: Р3207

Приняла:

Селина Елена Георгиевна

Отчёт принят «\_\_»\_\_\_\_\_2023 г.

Оценка: \_\_\_\_\_\_\_\_\_\_\_

г. Санкт-Петербург, 2023

**Оглавление**

[Задание 3](#_Toc151661213)

[Исходный код программы 4](#_Toc151661214)

[Вывод программы 8](#_Toc151661215)

# 

# 

# 

# Задание

Каждый студент получает выборку из 20 чисел. Необходимо определить следующие статистические характеристики: вариационный ряд, экстремальные значения и размах, оценки математического ожидания и среднеквадратического отклонения, эмпирическую функцию распределения и её график, гистограмму и полигон приведенных частот группированной выборки. Для расчета характеристик и построения графиков нужно написать программу на одном из языков программирования. Листинг программы и результаты работы должны быть представлены в отчете по практической работе.

Исходная выборка: [-0.03, 0.73, -0.59, -1.59, 0.38, 1.49, 0.14, -0.62, -1.59, 1.45, -0.38, -1.49, -0.15, 0.63, 0.06, -1.59, 0.61, 0.62, -0.05, 1.56]

# Исходный код программы

import math  
import matplotlib.pyplot as plt  
import numpy as np  
from scipy.interpolate import CubicSpline  
  
# Исходная выборка  
sample = [-0.03, 0.73, -0.59, -1.59, 0.38, 1.49, 0.14, -0.62, -1.59, 1.45, -0.38, -1.49, -0.15, 0.63, 0.06, -1.59, 0.61, 0.62, -0.05, 1.56]  
  
# Вариационный ряд  
variation\_range = list(set(sample))  
variation\_range = sorted(variation\_range)  
  
# Экстремальные значения и размах  
min\_value = variation\_range[0]  
max\_value = variation\_range[-1]  
range\_value = round(max\_value - min\_value, 2)  
  
# Частота  
frequency = []  
for i in variation\_range:  
 frequency.append(sample.count(i))  
  
# Статистический ряд  
sample\_len = len(sample)  
statistical\_series = []  
for i in range(len(variation\_range)):  
 el = []  
 el.append(variation\_range[i])  
 el.append(frequency[i])  
 el.append(frequency[i] / sample\_len)  
 statistical\_series.append(el)  
  
# Интервальный статистический ряд  
partial\_interval\_length = round(range\_value / (1 + math.log(sample\_len, 2)), 2) # Длина частичного интервала (по формуле Стерджеса)  
interval\_count = math.ceil(1 + math.log(sample\_len, 2))  
x\_start = min\_value - partial\_interval\_length / 2  
variation\_range\_len = len(variation\_range)  
  
intervals = []  
for i in range(interval\_count):  
 el = []  
 el.append(x\_start)  
 x\_start = round(x\_start + partial\_interval\_length, 2)  
 el.append(x\_start)  
 intervals.append(el)  
  
frequency\_for\_intervals = []  
for i in range(interval\_count):  
 frequency\_for\_intervals.append(0)  
for i in variation\_range:  
 if i >= intervals[0][0] and i < intervals[0][1]:  
 frequency\_for\_intervals[0] += 1  
 elif i >= intervals[1][0] and i < intervals[1][1]:  
 frequency\_for\_intervals[1] += 1  
 elif i >= intervals[2][0] and i < intervals[2][1]:  
 frequency\_for\_intervals[2] += 1  
 elif i >= intervals[3][0] and i < intervals[3][1]:  
 frequency\_for\_intervals[3] += 1  
 elif i >= intervals[4][0] and i < intervals[4][1]:  
 frequency\_for\_intervals[4] += 1  
 elif i >= intervals[5][0] and i < intervals[5][1]:  
 frequency\_for\_intervals[5] += 1  
  
interval\_statistical\_series = []  
for i in range(interval\_count):  
 el = []  
 el.append(intervals[i])  
 el.append(frequency\_for\_intervals[i])  
 el.append(round(frequency\_for\_intervals[i] / variation\_range\_len, 2))  
 interval\_statistical\_series.append(el)  
  
# Среднее значение  
average\_value = round(sum(sample) / sample\_len, 2)  
  
# Выборочная дисперсия  
sum\_for\_dis = 0  
for i in statistical\_series:  
 sum\_for\_dis += (i[0] - average\_value) \*\* 2 \* i[1]  
dispersion = round(1 / sample\_len \* sum\_for\_dis, 2)  
  
# Среднеквадратическое отклонение  
standard\_deviation = round(dispersion \*\* 0.5, 2)  
  
# Исправленная выборочная дисперсия  
сorrected\_sample\_variance = round(sample\_len / (sample\_len - 1) \* dispersion, 2)  
  
# Исправленное выборочное среднеквадратическое отклонение  
corrected\_standard\_deviation = round(сorrected\_sample\_variance \*\* 0.5)  
  
# Мода M\_o  
max\_frequency = max(frequency)  
for i in statistical\_series:  
 if i[1] == max\_frequency:  
 mode = i[0]  
  
# Медиана M\_e  
mid\_index = int(len(variation\_range) / 2)  
if len(variation\_range) % 2 == 0:  
 median = (variation\_range[mid\_index] + variation\_range[mid\_index - 1]) / 2  
else:  
 median = variation\_range[mid\_index]  
  
# Эмпирическая функция распределения  
def print\_empirical\_distribution\_function(statistical\_series):  
 x = statistical\_series[0][0]  
 frequency\_counter = 0  
 y\_values = []  
 print(f'Эмпирическая функция распределения:\n{frequency\_counter}, при x <= {x}')  
 for i in statistical\_series:  
 y\_values.append(frequency\_counter)  
 x\_old = x  
 x = round(x + i[0], 2)  
 frequency\_counter = round(frequency\_counter + i[2], 2)  
 print(f'{frequency\_counter}, при {x\_old} < x <= {x}')  
  
 # Строим график  
 plt.step(variation\_range, y\_values, where='post')  
 plt.xlabel('x')  
 plt.ylabel('F(x)')  
 plt.title('Эмпирическая функция распределения')  
 plt.grid(True)  
 plt.show()  
  
# Графическое изображение статистического распределения  
def print\_statistical\_distribution(statistical\_series):  
 y\_values\_stat = []  
 for i in statistical\_series:  
 y\_values\_stat.append(i[2])  
  
 # Строим график  
 plt.plot(variation\_range, y\_values\_stat)  
 plt.xlabel('x')  
 plt.ylabel('f(x)')  
 plt.title('Графическое изображение статистического распределения')  
 plt.grid(True)  
 plt.show()  
  
# Гистограмма частот  
def print\_frequency\_histogram():  
 # Extract heights and midpoints  
 heights = [round(i[2] / interval\_count, 2) for i in interval\_statistical\_series]  
 midpoints = [(interval[0] + interval[1]) / 2 for interval in intervals]  
 bar\_widths = [(interval[1] - interval[0]) for interval in intervals]  
  
 for i, (midpoint, height) in enumerate(zip(midpoints, heights)):  
 width = bar\_widths[i]  
 plt.bar(midpoint, height, width=width, align='center', alpha=0.7, label='Гистограмма' if i == 0 else '')  
  
 cs = CubicSpline(midpoints, heights)  
  
 smooth\_midpoints = np.linspace(min(midpoints), max(midpoints), 100)  
 smooth\_values = cs(smooth\_midpoints)  
  
 plt.plot(smooth\_midpoints, smooth\_values, label='Функция (Интерполяция)')  
  
 plt.xlabel('Интервалы')  
 plt.ylabel('Высота столбов')  
 plt.title('Гистограмма с функцией')  
 plt.legend()  
 plt.show()  
  
# Вывод результатов  
print(f'Вариационный ряд: {variation\_range}')  
print(f'Экстремальные значения: Минимум = {min\_value}, Максимум = {max\_value}')  
print(f'Статистический ряд (x\_i, n\_i, p\_i): {statistical\_series}')  
print(f'Интервальный статистический ряд (интервал, n\_i, p\_i): {interval\_statistical\_series}')  
print(f'Среднее значение: {average\_value}')  
print(f'Выборочная дисперсия: {dispersion}')  
print(f'Среднеквадратическое отклонение: {standard\_deviation}')  
print(f'Исправленная выборочная дисперсия: {сorrected\_sample\_variance}')  
print(f'Исправленное выборочное среднеквадратическое отклонение: {corrected\_standard\_deviation}')  
print(f'Размах: {range\_value}')  
print(f'Мода M\_o: {mode}')  
print(f'Медиана M\_e: {median}')  
print\_empirical\_distribution\_function(statistical\_series)  
print\_statistical\_distribution(statistical\_series)  
print\_frequency\_histogram()

# Вывод программы

Вариационный ряд: [-1.59, -1.49, -0.62, -0.59, -0.38, -0.15, -0.05, -0.03, 0.06, 0.14, 0.38, 0.61, 0.62, 0.63, 0.73, 1.45, 1.49, 1.56]

Экстремальные значения: Минимум = -1.59, Максимум = 1.56

Статистический ряд (x\_i, n\_i, p\_i): [[-1.59, 3, 0.15], [-1.49, 1, 0.05], [-0.62, 1, 0.05], [-0.59, 1, 0.05], [-0.38, 1, 0.05], [-0.15, 1, 0.05], [-0.05, 1, 0.05], [-0.03, 1, 0.05], [0.06, 1, 0.05], [0.14, 1, 0.05], [0.38, 1, 0.05], [0.61, 1, 0.05], [0.62, 1, 0.05], [0.63, 1, 0.05], [0.73, 1, 0.05], [1.45, 1, 0.05], [1.49, 1, 0.05], [1.56, 1, 0.05]]

Интервальный статистический ряд (интервал, n\_i, p\_i): [[[-1.885, -1.29], 2, 0.11], [[-1.29, -0.7], 0, 0.0], [[-0.7, -0.11], 4, 0.22], [[-0.11, 0.48], 5, 0.28], [[0.48, 1.07], 4, 0.22], [[1.07, 1.66], 3, 0.17]]

Среднее значение: -0.02

Выборочная дисперсия: 0.97

Среднеквадратическое отклонение: 0.98

Исправленная выборочная дисперсия: 1.02

Исправленное выборочное среднеквадратическое отклонение: 1

Размах: 3.15

Мода M\_o: -1.59

Медиана M\_e: 0.1

Эмпирическая функция распределения:

0, при x <= -1.59

0.15, при -1.59 < x <= -3.18

0.2, при -3.18 < x <= -4.67

0.25, при -4.67 < x <= -5.29

0.3, при -5.29 < x <= -5.88

0.35, при -5.88 < x <= -6.26

0.4, при -6.26 < x <= -6.41

0.45, при -6.41 < x <= -6.46

0.5, при -6.46 < x <= -6.49

0.55, при -6.49 < x <= -6.43

0.6, при -6.43 < x <= -6.29

0.65, при -6.29 < x <= -5.91

0.7, при -5.91 < x <= -5.3

0.75, при -5.3 < x <= -4.68

0.8, при -4.68 < x <= -4.05

0.85, при -4.05 < x <= -3.32

0.9, при -3.32 < x <= -1.87

0.95, при -1.87 < x <= -0.38

1.0, при -0.38 < x <= 1.18





