

Факультет программной инженерии и компьютерной техники

Дисциплина: Теория вероятностей

Практическая работа №5

Вариант 3

Выполнил: Васильев А. Ю.

№ группы: P3215

Преподаватель: Селина Е.Г.

2022 год, Санкт-Петербург

**Задание по практической работе**

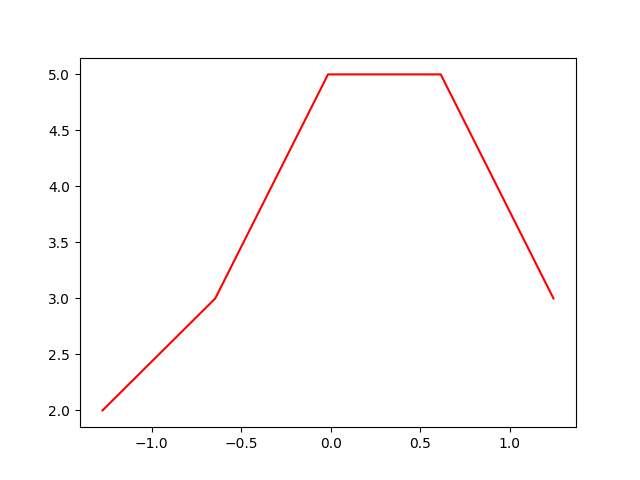
Каждый студент получает выборку из 20 чисел. Необходимо определить следующие статистические характеристики: вариационный ряд, экстремальные значения и размах, оценки математического ожидания и среднеквадратического отклонения, эмпирическую функцию распределения и её график, гистограмму и полигон приведенных 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

[-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]

**Листинг программы**

Сначала вычисляются требуемые математические величины, дальше происходит визуализация, выводится 3 графика

import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
row = np.array(  
 [-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])  
  
  
def print\_answer(row):  
 print(f"Исходный ряд: {row}")  
 variation\_row\_not\_unique =row.copy()  
 variation\_row\_not\_unique.sort()  
  
 variation\_row = np.unique(row.copy(), )  
 variation\_row.sort()  
 print(f"Вариационный ряд: {variation\_row}")  
 max = variation\_row[variation\_row.size - 1]  
 min = variation\_row[0]  
 print(f"Эстремальные значения\nМинимум: {min}\nМаксимум: {max}")  
 selection\_scope = max - min  
 print(f"Размах: {selection\_scope}")  
 math\_waiting = np.sum(variation\_row) / variation\_row.size  
 temp\_array = variation\_row.copy()  
 standard\_deviation = (np.sum((temp\_array - math\_waiting) \*\* 2) / temp\_array.size) \*\* 0.5  
 print(f"Математическое ожидание: {math\_waiting}\nСреднеквадратичное отклонение: {standard\_deviation}")  
  
 selection\_volume = variation\_row.size  
 empirical\_values = [(i / selection\_volume) for i in range(selection\_volume)]  
 print("f(x) value")  
 for i in range(variation\_row.size - 1):  
 print(f"({variation\_row[i]};{variation\_row[i + 1]}] : {empirical\_values[i + 1]}")  
  
 # Эмпирическая функция распределения  
 plt.step(variation\_row, empirical\_values, color="r")  
 plt.show()  
  
 # Гистограмма  
  
 # формула Стерджесса  
 interval\_count = int(1 + np.log2(selection\_volume))  
  
 sns.histplot(variation\_row, stat="density", color="r", bins=interval\_count)  
 plt.show()  
  
 # Полигон группированной выборки  
 interval\_width = selection\_scope / interval\_count  
  
 counter = 0  
 last\_element = variation\_row\_not\_unique[0]  
 for i in range(interval\_count):  
 for j in range(row.size):  
 if (variation\_row\_not\_unique[j] < (last\_element + interval\_width) and variation\_row\_not\_unique[j] >= last\_element):  
 counter += 1  
 print(f"[{last\_element};{last\_element + interval\_width}) : {counter}")  
 counter = 0  
 last\_element += interval\_width  
  
 y, x = np.histogram(variation\_row, np.arange(min, max + 0.1, interval\_width))  
 interval\_centres = [(x[i] + x[i - 1]) \* 0.5 for i in range(1, x.size)]  
 plt.plot(interval\_centres, y, color="r")  
 plt.show()  
  
  
print\_answer(row)

**Примеры и результаты работы программы** ****

