

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмних систем і технологій

Звіт
Лабораторна робота № 2

з дисципліни
«Ймовірнісні основи програмної інженерії»

Виконав:	Антонова Олександра Лаврентійовича	Перевірила:	Марцафей А.
Група	ІПЗ-24(2)	Дата перевірки	
Форма навчання	денна		

Оцінка

Спеціальність	121		
2022			

Лабораторна робота 2

Лінійне перетворення та Графічне зображення даних

Мета: навчитись використовувати на практиці набуті знання про лінійні перетворення та графічне зображення даних.

Завдання:

1. Знайдіть $Q1$, $Q3$ та $P90$.
2. Знайдіть середнє та стандартне відхилення цих оцінок.
3. Через незадоволення низькими оцінками викладач вирішив використати шкалу форми $y = ax + b$, щоб відредагувати оцінки. Він хотів, щоб середнє значення масштабних оцінок становило 95, а оцінка 100, щоб залишалася рівною 100.
4. Показати дані за допомогою діаграми "стовбур – листя".
5. Відобразити дані за допомогою коробкового графіка.
6. Зробити висновок.

Побудова математичної моделі

1. Для розрахунку 1-ого та 3-ого квартилю та 90-ого персантилю будемо використовувати формулу:

$$n = k \cdot 100 (N + 1), \text{ де}$$

k – номер персантилю

N – кількість елементів

n – номер елементу, який є персантилем

Тепер для знаходження його значення нам потрібно використати наступну формулу:

$$P_n = x_n + \text{хдроб} * (x_{n+1} - x_n), \text{ де}$$

P_n - значення n -ого персантилю

x_n – Значення елемента з вибірки, який стоїть за номером цілої частини n

Хдроб – значення дробової частини n

x_{n+1} – Значення елемента з вибірки, який стоїть за номером цілої частини $n + 1$

2. Для знаходження середнього відхилення використаємо формулу:

$$\frac{1}{n} \sum_{i=1}^n |x_i - m(X)|, \text{ де}$$

n – кількість елементів

x_i – поточний елемент

$m(x)$ – середнє значення з вибірки

Щоб знайти *стандартне відхилення* використаємо:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - x_{cp})^2}{n - 1}}, \text{ де}$$

n - кількість елементів

x_i – поточний елемент

$x_{\text{ср}}$ – середнє значення з вибірки

S – стандартне відхилення

3. Для виконання завдання потрібно знайти значення a та b , розв'язавши систему рівнянь

$$\begin{cases} \text{Max} = \text{Max} * a + b \\ \text{Aver} = \bar{x} * a + b \end{cases}, \text{ де}$$

Max – Значення максимальної оцінки

Aver - значення бажаної середньої оцінки

\bar{x} – середнє значення оцінок

a і b – значення невідомих

Тепер щоб змінити значення оцінок нам потрібно підставити a і b в формулу:

$$y = ax + b$$

де:

x – значення поточної оцінки

y – значення бажаної оцінки

Код алгоритму

1. Обчислення кватилів та перцентилів

```
def task1(num):  
    index = num * (lens + 1) - 1  
    result = data[int(index)] + (index % int(index)) * (data[int(index) + 1] - data[int(index)])  
    return result
```

2. Знаходження середнього та стандартного відхилення оцінок.

```

25 def task2():
26     sum = 0
27     totalsum = 0
28     totalSum = 0
29     for i in range(lens):
30         sum += data[i]
31
32     for i in range(lens):
33         totalsum += (data[i] - (sum / lens)) ** 2
34         totalSum += abs(data[i] - (sum / lens))
35
36     result = totalsum / lens
37     Result = totalSum / lens
38     print("Стандартне відхилення = ", str(round(math.sqrt(result))))
39     print("Середнє відхилення = ", str(round(Result)))
40
41     sus.write("\nСтандартне відхилення = " + str(round(math.sqrt(result))))
42     sus.write("\nСереднє відхилення = " + str(round(Result)))

```

3. Обчислення відредагованих оцінок

```

44 def task3():
45     sum = 0
46     result = []
47     for i in data:
48         sum += i
49     a = np.array([[100, 1, ], [(sum / lens), 1, ]])
50     #|100 = 100*a + b
51     #|95 = 74.2*a + b
52     x = solve(a, np.array([100, 95]))
53     for i in range(lens):
54         result.append(round(x[0] * data[i] + x[1]))
55     print("Старі оцінки: " + str(data))
56     sus.write("\nСтарі оцінки: " + str(data))
57
58     print("\ny = " + str(x[0]) + "*x + " + str(x[1]))
59     sus.write("\ny = " + str(x[0]) + "*x + " + str(x[1]))
60
61     print("\nНові оцінки: " + str(result))
62     sus.write("\nНові оцінки: " + str(result))

```

4. Побудова діаграми "стовбур – листя".

```

64 def task4():
65     print("Діаграма стовбур-листя")
66     print("-----")
67
68     sus.write("\nДіаграма стовбур-листя")
69     sus.write("\n-----")
70
71     i = min(data)
72
73     while i <= max(data):
74         mas = []
75         for j in range(lens):
76             if i < data[j] < i + 10:
77                 mas.append(data[j] % 10)
78             elif data[j] == i:
79                 mas.append(0)
80             if len(mas) != 0:
81                 print(str(i / 10) + " \t| " + str(mas))
82                 sus.write("\n" + str(i / 10) + " \t| " + str(mas))
83             i += 10
84     print("Ключ = " + str(data[0]))
85     sus.write("\nКлюч = " + str(data[0]))

```

5. Відображення даних за допомогою коробкового графіка.

```

87 def task5():
88     plt.boxplot(data)
89     plt.grid()
90     plt.show()

```

6.

Повний код програми

```

import math
import matplotlib.pyplot as plt
import numpy as np
from scipy.linalg import solve
#=====
sus= open("result.txt", "w")
data = []
for i in open("input_101.txt"):
    data.append(int(i.strip()))
data = np.delete(data, 0)

lens = len(data)
data = sorted(data)

```

```

        print("Послідовність:", data)
    sus.write("Послідовність:" + str(data))

#=====
    def task1(num):
        index = num * (lens + 1) - 1
        result = data[int(index)] + (index % int(index)) *
            (data[int(index) + 1] - data[int(index)])
        return result
#=====
    def task2():
        sum = 0
        totalsum = 0
        totalSum = 0
        for i in range(lens):
            sum += data[i]

        for i in range(lens):
            totalsum += (data[i] - (sum / lens)) ** 2
            totalSum += abs(data[i] - (sum / lens))

        result = totalsum / lens
        Result = totalSum / lens
        print("Стандартне відхилення = ",
            str(round(math.sqrt(result))))
        print("Середнє відхилення = ", str(round(Result)))

        sus.write("\nСтандартне відхилення = " +
            str(round(math.sqrt(result))))
        sus.write("\nСереднє відхилення = " + str(round(Result)))
#=====
    def task3():
        sum = 0
        result = []
        for i in data:
            sum += i
        a = np.array([[100, 1, ], [(sum / lens), 1, ]])
        #|100 = 100*a + b
        #|95 = 74.2*a + b
        x = solve(a, np.array([100, 95]))
        for i in range(lens):
            result.append(round(x[0] * data[i] + x[1]))
        print("Старі оцінки: " + str(data))
        sus.write("\nСтарі оцінки: " + str(data))

        print("\ny = " + str(x[0]) + "*x + " + str(x[1]))
        sus.write("\ny = " + str(x[0]) + "*x + " + str(x[1]))

```

```

        print("\nНові оцінки: " + str(result))
        sus.write("\nНові оцінки: " + str(result))
#=====

        def task4():
            print("Діаграма стовбур-листя")
            print("-----")

            sus.write("\nДіаграма стовбур-листя")
            sus.write("\n-----")

            i = min(data)

            while i <= max(data):
                mas = []
                for j in range(lens):
                    if i < data[j] < i + 10:
                        mas.append(data[j] % 10)
                    elif data[j] == i:
                        mas.append(0)
                    if len(mas) != 0:
                        print(str(i / 10) + " \t| " + str(mas))
                sus.write("\n" + str(i / 10) + " \t| " + str(mas))
                i += 10
            print("Ключ = " + str(data[0]))
            sus.write("\nКлюч = " + str(data[0]))
#=====

        def task5():
            plt.boxplot(data)
            plt.grid()
            plt.show()
            print("Task 1:")

            Q1 = task1(1 / 4)
            Q3 = task1(3 / 4)
            P90 = task1(0.9)

            print("Q1 = ", Q1)
            print("\nQ3 = ", Q3)
            print("\nP90 = ", P90)

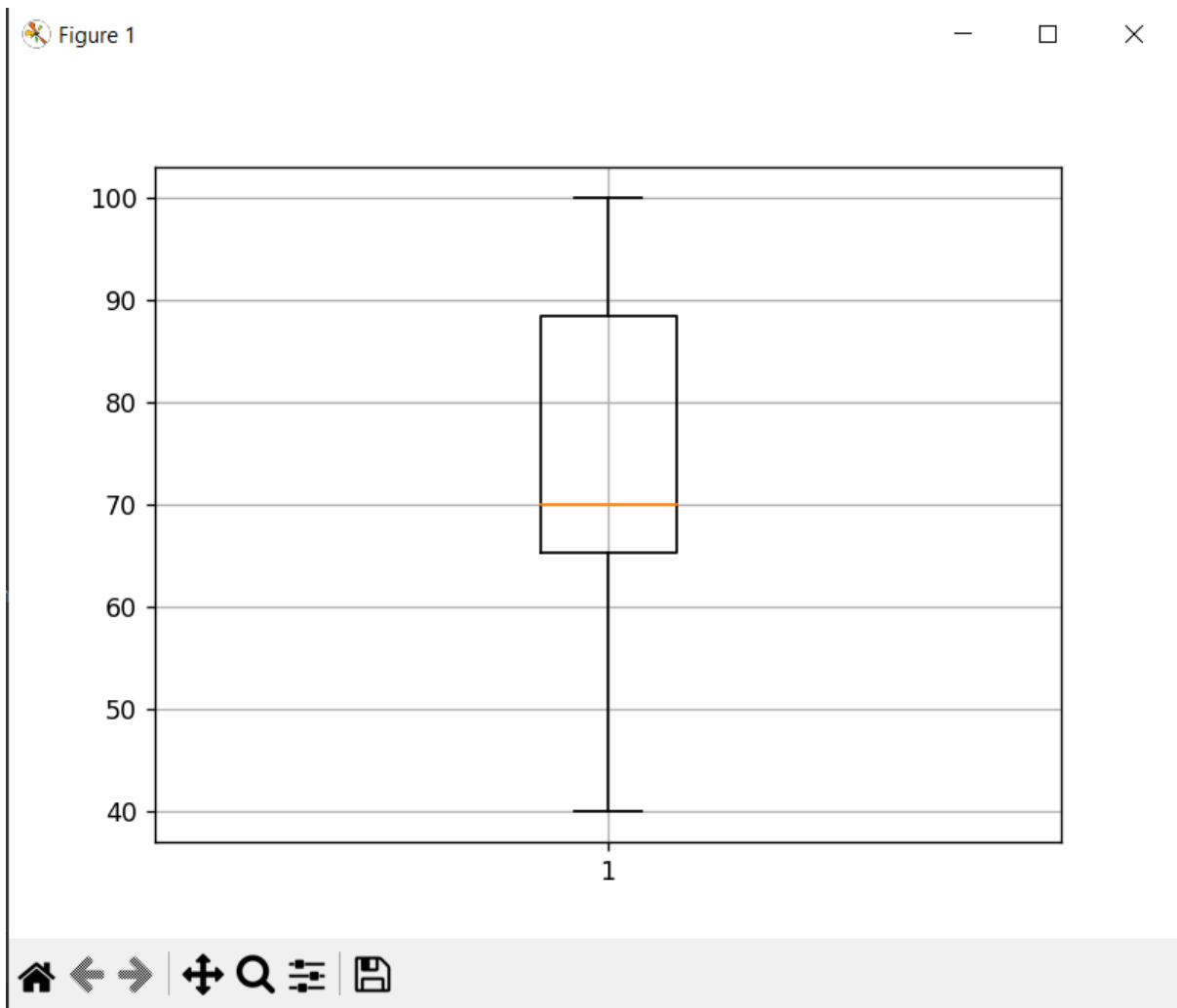
            sus.write("\nQ1 = ")
            sus.write(str(Q1))
            sus.write("\nQ3 = ")
            sus.write(str(Q3))
            sus.write("\nP90 = ")
            sus.write(str(P90))
            print("Task 2:")

```



```
task2()  
print("Task 3 :")  
task3()  
print("Task 4:")  
task4()  
task5()  
sus.close()
```

Результат виконання



а також отримали вихідний файл з результатами

```
Послідовність: [40, 62, 65, 66, 70, 70, 84, 90, 95, 100]
```

```
Task 1:
```

```
Q1 = 64.25
```

```
Q3 = 91.25
```

```
P90 = 99.5
```

```
Task 2:
```

```
Стандартне відхилення = 17
```

```
Середнє відхилення = 14
```

```
Task 3 :
```

```
Старі оцінки: [40, 62, 65, 66, 70, 70, 84, 90, 95, 100]
```

```
 $y = 0.19379844961240322 \cdot x + 80.62015503875968$ 
```

```
Нові оцінки: [88, 93, 93, 93, 94, 94, 97, 98, 99, 100]
```

```
Task 4:
```

```
Діаграма стовбур-листя
```

```
-----
```

```
4.0      | [0]
```

```
6.0      | [2, 5, 6]
```

```
7.0      | [0, 0]
```

```
8.0      | [4]
```

```
9.0      | [0, 5]
```

```
10.0     | [0]
```

```
Ключ = 40
```

```
Process finished with exit code 0
```

Висновок: я навчився використовувати на практиці набуті знання про лінійні перетворення та графічне зображення даних.