

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА
ШЕВЧЕНКА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмних систем і технологій

Дисципліна
«Ймовірнісні основи програмної інженерії»

Лабораторна робота № 2

Виконав:	Ковалець Олег Сергійович	Перевірила:	Марцафей А.С.
Група	ІПЗ-22	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 Q_1 , Q_3 та P_{90}

Для того, щоб знайти n -й кuartиль потрібно вирішити вираз:

$(n/4) * (\text{к-сть елементів послідовності} + 1)$,

Персентиль від n шукається за такою ж формулою, але замість $n/4$ підставляємо $n/100$.

Отже, напишемо програму для розрахунку цих 1 та 3 кuartилів та персентиль(90) за заданими даними з файлів:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 writer = open("output.txt", "w")
5
6 file = input("Enter a name of the file: ")
7 reader = open(file).read().splitlines()
8 del reader[0]
9 rdr = [int(i) for i in reader]
10 rdr.sort()
11
12
13 def task1():
14     q1 = (1/4)*(len(rdr) + 1)
15     i, d = divmod(q1, 1)
16     q1 = rdr[int(i-1)] + d*(rdr[int(i)] - rdr[int(i-1)])
17
18     q3 = (3/4)*(len(rdr) + 1)
19     i, d = divmod(q3, 1)
20     q3 = rdr[int(i-1)] + d*(rdr[int(i)] - rdr[int(i-1)])
21
22     p = 0.9*(len(rdr) + 1)
23     i, d = divmod(p, 1)
24     p = rdr[int(i-1)] + d*(rdr[int(i)] - rdr[int(i-1)])
25
26     writer.write("Quartile 1: " + str(q1) + "\nQuartile 3: " + str(q3) + "\nP90: " + str(p))
```

Маємо наступні результати для файлів **input_10.txt** та **input_100.txt** відповідно:

```
Quartile 1: 64.25  
Quartile 3: 91.25  
P90: 99.5
```

```
Quartile 1: 36.5  
Quartile 3: 78.75  
P90: 91.9
```

2. Знайдіть середнє та стандартне відхилення цих оцінок.

Для того, щоб знайти середнє (== стандартне відхилення) потрібно знайти значення дисперсії з минулої лабораторної роботи за заданими даними, та піднести це значення у квадратний корінь.

Отже, маємо формулу

$$D = \frac{\sum (x - \mu)^2}{n}$$

$$\sqrt{D}$$

та напишемо програму для знаходження даних величин:

```
29 def task2():  
30     average, deviation = 0, 0  
31     for i in range(len(rdr)):  
32         average += int(rdr[i])  
33  
34     average /= len(rdr)  
35     for i in range(len(rdr)):  
36         deviation += np.fabs(int(rdr[i]) - average)  
37  
38     deviation /= len(rdr)  
39     writer.write("Deviation: " + str(deviation))
```

Маємо наступні результати для файлів **input_10.txt** та **input_100.txt** відповідно:

```
Deviation: 14.440000000000001
```

```
Deviation: 21.243199999999999
```

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.

За прикладом з лекції

a) $\bar{x} = 64$, $\text{Var}(X) = 199.67$, and $\sigma_x = \sqrt{\text{Var}(X)} = 14.13$.

EXAMPLE 2

A professor gave a test to 12 students. The grades were:

60 55 74 68 78 45 83 40 60 85 50 70

Not being happy with the low grades, the professor decided to use a scale of the form $y = ax + b$ to shift the grades up. He wanted the average of the scaled grades to be 82 and a grade of 100 to remain 100.

b) Find a and b .

Solution

b) The two conditions set by the professor lead to the system:

$$\begin{cases} 100 = 100a + b \\ 82 = 64a + b \end{cases}$$

Solving this system gives: $a = 0.5$ and $b = 50$.

So, the scaling to be used should be: $y = 0.5x + 50$.

напишемо програму для вирішення даного завдання.

Отже, маємо систему, де перше рівняння -
оцінка, що залишається = максимальна оцінка * a + b , а друге -
середнє значення масштабних оцінок = середнє значення + b

Напишемо програму для вирішення даного завдання:

```

42 def task3():
43     average, deviation = 0, 0
44     for i in range(len(rdr)):
45         average += int(rdr[i])
46
47     average /= len(rdr)
48     for i in range(len(rdr)):
49         deviation += np.fabs(int(rdr[i]) - average)
50
51     deviation /= len(rdr)
52
53     a = np.array([[100, 1], [average, 1]])
54     b = np.array([100, 95])
55     answer = np.linalg.solve(a, b)
56     writer.write(str(answer))

```

Маємо наступні результати для файлів **input_10.txt** та **input_100.txt** відповідно:

```
[ 0.19379845 80.62015504]
```

```
[ 0.11893435 88.10656518]
```

4. Показати дані за допомогою діаграми "стовбур – листя".

Напишемо алгоритм для побудови даної діаграми:

```

59 def task4():
60     writer.write("Stem\tLeaf\n")
61     digits = [str(rdr[0])[0]]
62     count = 0
63     for i in range(len(rdr)):
64         for j in range(len(digits)):
65             if str(rdr[i])[0] != digits[j]:
66                 count += 1
67             if count == len(digits):
68                 digits += str(rdr[i])[0]
69                 digits.sort()
70                 count = 0
71
72     for i in range(len(digits)):
73         temp = str(digits[i]) + "\t\t"
74         for j in range(len(rdr)):
75             if str(rdr[j])[0] == digits[i]:
76                 temp += str(rdr[j])[1:] + "\t"
77     writer.write(temp + "\n")

```

Маємо наступні результати для файлів **input_10.txt** та **input_100.txt** відповідно:

Stem	Leaf
1	00
4	0
6	2 5 6
7	0 0
8	4
9	0 5

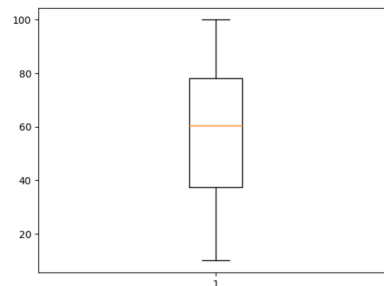
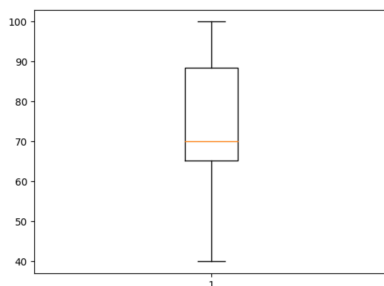
Stem	Leaf
1	0 2 4 4 5 6 6 9 9 00 00
2	2 2 2 2 2 5 5 5 6 8
3	1 5 5 5 6 6 8 8
4	1 2 4 5 6 6 7 9
5	0 1 1 1 2 3 4 5 6 6 7 8 8
6	0 0 1 1 1 1 3 4 5 6 7 8 8 8 9
7	0 0 1 3 3 6 6 7 7 7 8 8 9 9
8	0 1 1 2 2 2 3 3 5 7 7 8
9	1 2 2 4 7 7 8 9 9

5. Відобразити дані за допомогою коробкового графіка.

Скористаємось бібліотекою matplotlib для побудови коробкового графіка за заданими даними.

```
80 def task5():
81     plt.boxplot(rdr)
82     plt.savefig("output.png")
```

Маємо наступні результати для файлів **input_10.txt** та **input_100.txt** відповідно:



Висновок:

Протягом даної лабораторної роботи було використано на практиці набуті знання про лінійні перетворення та графічне зображення даних. Було знайдено значення кватилів, персентиля, відхилення за заданими даними. Також було побудовано за інформацією з вхідних файлів діаграму стовбур-листа та коробкову діаграму.