

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 № 1

Виконав:	Мельничук Дмитро Олегович	Перевірів:	Вечерковська А.С.
Група	ІПЗ-24(1)	Дата перевірки	
Форма навчання	денна	Оцінка	
Спеціальність	121		
2022			

Центральні тенденції та міра дисперсії

Мета – навчитись використовувати на практиці набуті знання про центральні тенденції та міри.

Завдання

1. Побудувати таблицю частот та сукупних частот для переглянутих фільмів.
1. Визначити фільм, який був переглянутий частіше за інші.
2. Знайти Моду та Медіану заданої вибірки.
3. Порахувати Дисперсію та Середнє квадратичне відхилення розподілу.
4. Побудувати гістограму частот для даного розподілу.
5. Зробити висновок з вигляду гістограми, про закон розподілу.

Математична модель:

Спочатку треба побудувати таблицю частот, в якій буде 2 стовпчика:

- Кількість переглядів фільму
- Кількість повторень переглядів фільму

Для того аби побудувати таблицю сукупних частот необхідно додати до таблиці частот ще один стовпчик, у якому буде загальна сума переглядів певного фільму.

Фільм, який був переглянутий частіше за інші, буде той, який має найбільше значення в таблиці сукупних частот.

Мода – це значення величини, що трапляється найбільше у вибірці. В даному випадку можна сказати, що модою буде фільм, який має найбільшу кількість повторних переглядів

Медіана – значення з відсортованої вибірки, яке знаходиться у її середині.

Для того аби знайти середнє квадратичне відхилення скористаємося формулою:

$$S = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2}.$$

Де підкореневий вираз є дисперсією

n – кількість елементів у вибірці

x_i – поточний елемент

\bar{x} – середнє арифметичне вибірки, вираховується за формулою:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + \dots + x_n).$$

Псевдокод алгоритму:

```
import math
import numpy as np
import matplotlib.pyplot as plt

nameoffile = input()
data = np.genfromtxt(nameoffile, dtype='float')

myfile = open("output1.txt", mode='w', encoding='latin_1')
print(data)

uniqueEl_array=[]

uniqueElCount = 0
datasort = sorted(data)
for ele in datasort:
    if(ele not in uniqueEl_array):
        uniqueElCount += 1
        uniqueEl_array.append(ele)

print(uniqueEl_array)

name = [0 for i in range(uniqueElCount)]
freq = [1 for i in range(uniqueElCount)]
helpUniqEl = 0
helpUniqEl2 = -1
for i in range(len(datasort)):
    for j in range(uniqueElCount):
        if (datasort[i] == name[j]):
            freq[helpUniqEl2] += 1
            break
        if (j == uniqueElCount - 1):
            name[helpUniqEl] = datasort[i]
            helpUniqEl += 1
            helpUniqEl2 += 1
count = [(name[i],freq[i]) for i in range(len(freq))]
print()

print("  Name   | Count | ReactCount ")
print("-----+-----+-----")
myfile.write("\n      Name   | Count | ReactCount ")
myfile.write("\n      -----+-----+-----")

reactCount = 0
for i in range(len(count)):
    reactCount += count[i][1]
    myfile.write("\n      " + str(count[i][0]) + "\t|   " + str(count[i][1]) + "   | \t" +
+ str(reactCount))
    if (int(count[i][0]) < 10):
        print("\t" + str(count[i][0]) + "\t|   " + str(count[i][1])+"   | \t"+
str(reactCount))
    else:
        print("\t" + str(count[i][0]) + "\t|   " + str(count[i][1])+"   | \t"+
str(reactCount))
```

```

for i in range(int(len(freq))):
    if (freq[i] == max(freq)):
        maxfilm = ('Film: ' + str(name[i]) + ' Freq: ' + str(freq[i]))
print('moda : ' + maxfilm)
myfile.write('\nmoda : ' + maxfilm)
sortedData = sorted(data)
print('mediana: ' + str(sortedData[int(len(sortedData) / 2)]))
myfile.write('\nmediana: ' + str(sortedData[int(len(sortedData) / 2)]))

sumX = 0
for i in range(int(len(freq))):
    sumX += int(name[i]) * freq[i]

sumX = sumX/int(len(data))

varX = 0
for i in range(int(len(data))):
    varX += (int(data[i]) - sumX)**2

varX = varX / (len(data) - 1)

print("Dispersion = " + str(round(varX, 4)))
print("Standard deviation = " + str(round(math.sqrt(varX), 4)))
myfile.write("\nDispersion = " + str(round(varX, 4)))
myfile.write("\nStandard deviation = " + str(round(math.sqrt(varX), 4)))
myfile.close()

print(sorted(data))
plt.hist(sorted(data), bins = 20, align = 'mid', facecolor = 'DARKBLUE' )
plt.show()

```

Випробування алгоритму

[10. 1. 66. 75. 1. 1. 12. 10. 97. 12. 66.]

[(1.0, 3), (10.0, 2), (12.0, 2), (66.0, 2), (75.0, 1), (97.0, 1)]

Name	Count	Total count
-----+-----+-----		
1.0	3	3.0
10.0	2	20.0
12.0	2	24.0
66.0	2	132.0
75.0	1	75.0
97.0	1	97.0

Max count of vieves = 132.0

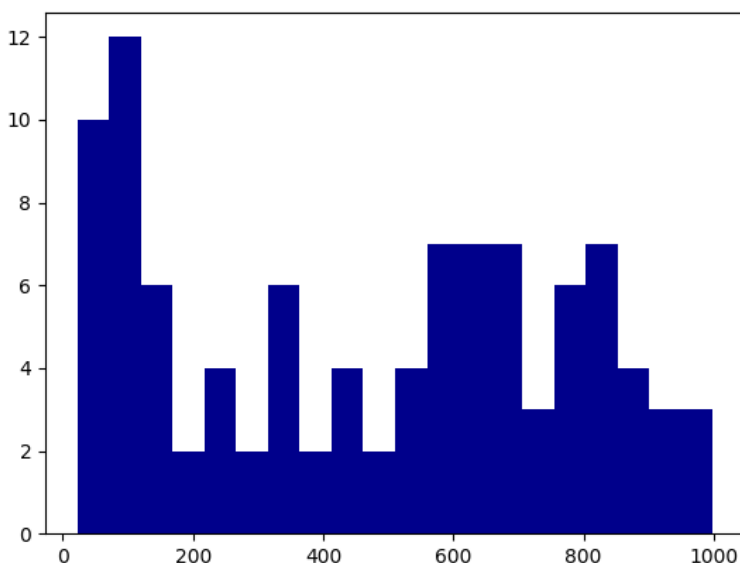
moda : Film: 1.0 Freq: 3

mediana: 12.0

Dispersion = 1303.6909

Standard deviation = 36.1067

При введенні 100 даних отримали таку гістограму:



Побудувавши гістограму для 100 елементів, в якій на 1 поділку приходить 5 даних бачимо, що найбільшу кількість переглядів складають перші 10 даних, потім тенденція знижується, і знов підвищується на проміжку в 50-80 елементів.

Висновок: навчився розробляти код для набутих на лекціях знаннях, а саме пошук однакових елементів, їх кількість, та робота з ними. Також шукати моду медіану та середнє квадратичне відхилення.