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
«Лінійне перетворення та Графічне зображення даних»

Виконав:	Сирота Ангеліна Олександрівна	**Перевірила**:	Вечерковська Анастасія Сергіївна
Група	ІПЗ-21	Дата перевірки	
Форма навчання	денна	Оцінка	
Спеціальність	121		

2022

Мета – навчитись використовувати на практиці набуті знання про міри в двовимірній статистиці.

Хід роботи

Постановка задачі:

Написати програму, що зчитує дані з файла і виконує наступні функції:

- Намалювати діаграму розсіювання для даних. Указати, чи існує тренд у даних. Якщо так, то вказати, чи є це негативним трендом, чи позитивним.
- Знайти центр ваги і коваріацію.
- Знайти рівняння лініїї регресії у від х.
- Розрахувати коефіцієнт кореляції між даними.
- Зробити висновок про залежності.

Усі результати програма записує в окремий текстовий файл.

Побудова математичної моделі:

Діаграма розсіювання: один з типів математичних діаграм, що використовує декартову систему координат для відображення значень двох змінних для набору даних. Дані показані у вигляді набору точок, кожен з яких має значення однієї змінної, тобто визначає її положення на горизонтальній осі та значення іншої змінної – її положення на вертикальній осі.

Тренд у даних: це основна тенденція змінення певного процесу . Лінія тренду — це лінія, уздовж якої розташовуються на діаграмі точки, що зображають дані з певного ряду даних.

Центр ваги даних: $G = (\bar{x}; \overline{y})$

Коваріація даних: це міра спільної мінливості двох випадкових змінних. Якщо більші значення однієї змінної здебільшого відповідають більшим значенням іншої, й те саме виконується для менших значень, тобто змінні схильні демонструвати подібну поведінку, то коваріація є додатною.

$$cov(x, y) = \frac{1}{n}\sum_{i=1}^{n} x_i y_i - \bar{x}\bar{y}$$

Лінія регресії: $y = b_1 x + b_0$

$$b_1 = \frac{cov(x, y)}{Var(x)}$$

$$b_0 = \bar{y} - b_1 \bar{x}$$

Коефіцієнт кореляції: використовується в науці для вимірювання ступеня лінійної залежності між двома змінними.

$$r = \frac{cov(x, y)}{\sigma_x \sigma_y}$$

Псевдокод алгоритму:

Заповнення масиву значеннями елементів:

```
for item in f:
    item = item.replace(',','.')                  // замінити , на .
    x.append(float(item.partition('\t')[0]))   // до \t
    y.append(float(item.partition('\t')[2]))   // після \t
```

Побудова гістограми розсіювання:

```
pyplot.scatter(x, y, edgecolor = 'k', alpha = 0.5)
pyplot.title("Діаграма розсіювання")
pyplot.xlabel("Час, проведений у супермаркеті")
pyplot.ylabel("Сума покупки")
pyplot.show()
```

Визначення середнього значення:

```
for i in range(arr):
   numerator += arr[i]
ave = numerator / n
```

Визначення центра ваги:

```
Xave = average(x)                    // середнє значення
Yave = average(y)

print(G(Xave, Yave))
```

Визначення коваріації:

```
Xave = average(x)                    // середнє значення
Yave = average(y)

for i in range(len(x)):
   sumXY += x[i] * y[i]
Cov = 1 / len(x) * sumXY - Xave * Yave
```

Визначення дисперсії:

```
Xave = average(arr)                  // середнє значення

for i in range(arr):
   sumX += arr[i]^2

dis = 1 / n * sumX – Xave^2
return dis
```

Знаходження рівняння лінії регресії y від x і побудова графіка:

```
# y = b1x + b0
   b1 = cov(x, y, False) / Var(x)                    // за формулою коваріація / дисперсія x

   Xave = average(x)                                 // середнє значення
   Yave = average(y)

   b0 = Yave - b1 * Xave

   print(Лінія регресії: y = b1 * x + b0)

   #тренд
   if b1 > 0:
      print("Тренд є позитивним")
   elif b1 < 0:
      print("Тренд є негативним")

   # побудова
   y1 = []
   for x1 in x:
      res = b1 * x1 + b0
      y1.append(res)

   pyplot.plot(x, y1, color = 'lawngreen', label = 'Лінія регресії', linewidth = 2)

   pyplot.scatter(x, y, edgecolor = 'k', alpha = 0.5)
   pyplot.title("Діаграма розсіювання")
   pyplot.xlabel("Час, проведений у супермаркеті")
   pyplot.ylabel("Сума покупки")
   pyplot.legend()

   pyplot.show()
```

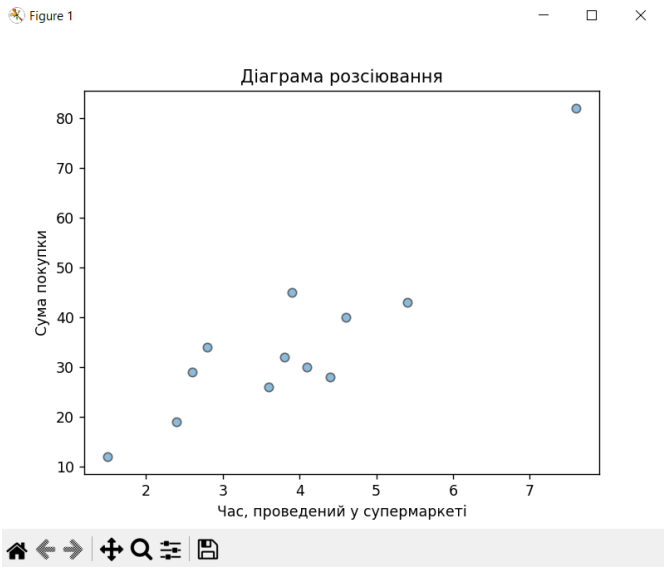
Розрахунок коефіцієнта кореляції:

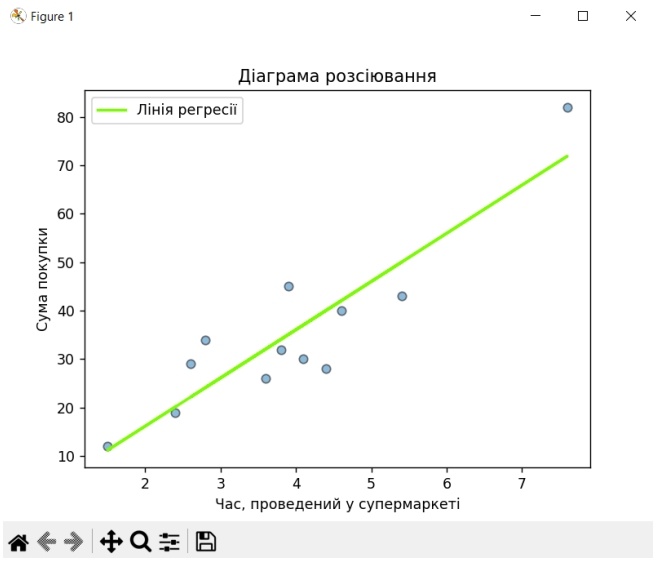
```
   sx = sqrt(Var(x))                 // стандартне відхилення
   sy = sqrt(Var(y))

   r = cov(x, y) / (sx * sy)

   print(r)

   # висновки щодо значення коефіцієна кореляції
   r0 = sqrt(3) / 2
   if r = 1 or r = -1:
      print("Точки лежать на лінії регресії")
   elif r > r0 or r < r0:
      print("Між даними існує сильна лінійна залежність")
   elif r = 0:
      print("Дані лінійно незалежні")
   else:
      print("Між даними існує слабка лінійна залежність")
```

```
if r < 0:
    print("Залежність є негативною")
elif r > 0:
    print("Залежність є позитивною")
```

Випробування алгоритму:

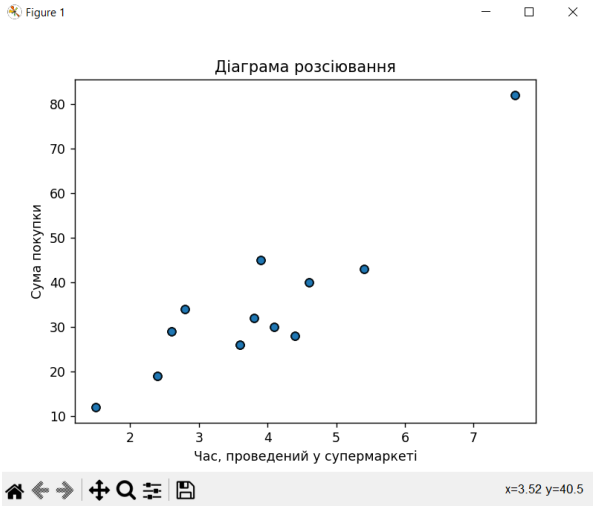
Набір з 12 точок:

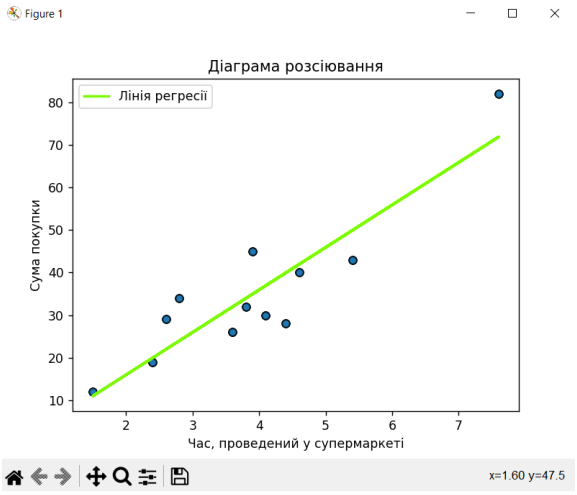





```
Введіть значення кількості елементів у вхідному файлі (10/100):
10
Центр ваги: G( 3.892 , 35.0 )

Коваріація: cov = 23.0

Лінія регресії: y = 9.953 * x -3.735
Тренд є позитивним

Коефіцієнт кореляції: 0.901

Між даними існує сильна лінійна залежність
Залежність є позитивною
```

Набір із 100 точок:






```
Введіть значення кількості елементів у вхідному файлі (10/100):
100
Центр ваги: G( 3.856 , 34.5 )

Коваріація: cov = 22.592

Лінія регресії: y = 9.982 * x -3.991
Тренд є позитивним

Коефіцієнт кореляції: 0.902

Між даними існує сильна лінійна залежність
Залежність є позитивною
```

Висновок: в ході цієї лабораторної роботи було зчитано дані з вхідних файлів і намальовано діаграми розсіювання для цих даних. Для обох наборів даних тренд є позитивним. Знайдено центр ваги і коваріацію. Знайдено рівняння лінії регресії y від x. Для цього обчислено коефіцієнти b1, b0 ($y = b_1 x + b_0$). Отриману пряму нанесено на графік. Для визначення міцності лінійної залежності між змінними обчислено коефіцієнт кореляції. В обох випадках залежність є сильною і позитивною. Вихідні дані записувались в окремий текстовий файл.