

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
імені Вадима Гетьмана

Кафедра інформатики та системології

ЛАБОРАТОРНА РОБОТА № 5

з дисципліни «Системи і методи штучного інтелекту»

Тема: «Лінійна регресія. Метод найменших квадратів. Інтерполяція»

Виконав: Виноградов Н.Р. ІН-405

Перевірив: Волошин А.П.

Київ – 2025

Мета роботи: Опрацювати поняття «лінійна регресія» і дослідити метод найменших квадратів та набути навички роботи в середовищі Python.

Хід роботи:

Завдання 2. Експериментально отримані N-значень величини Y при значеннях величини X. Відшукати параметри функції за методом найменших квадратів. Побудувати графіки, де в декартовій системі координат нанести експериментальні точки і графік апроксимуючої функції

Варіант 4

X 2 4 6 8 10 12

Y 6,5 4,4 3,8 3,5 3,1 3,0

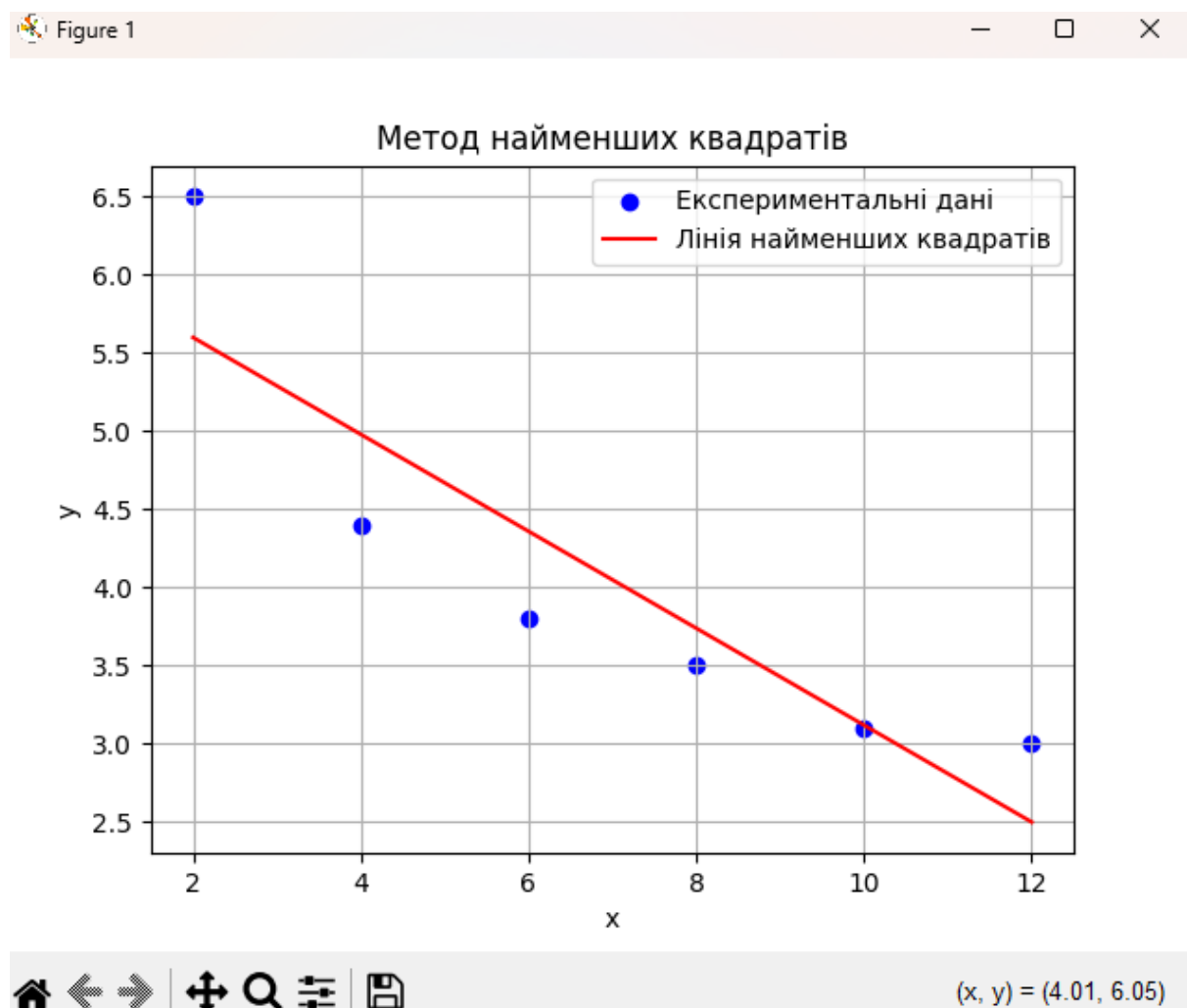


Рисунок 1.1 – Графік лінії найменших квадратів

```
PS C:\Codes\python_codes\AI\openAI_KNN>
Рівняння прямої: y = 6.220 + -0.310·x
PS C:\Codes\python_codes\AI\openAI_KNN>
```

Рисунок 1.2 – Результат виконання коду

Завдання № 3: Виконати інтерполяцію функції, задану в табличній формі в п'яти точках. Розрахунки виконати в середовищі Python.



Рисунок 2.1 – Графік інтерполяційного поліному

```
Коефіцієнти полінома (від вищого до нижчого ступеня):  
[-852.77777778 1480.55555556 -864.02777778 186.25 -8.18 ]  
Значення полінома у x=0.2: 4.989  
Значення полінома у x=0.5: 0.709
```

Рисунок 2.2 – Результат роботи коду

Висновок

Під час виконання лабораторної роботи було опрацьовано поняття «лінійна регресія» і досліджено метод найменших квадратів, набуто навички роботи в середовищі Python.

Коди:

Завдання 2

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

# Вхідні дані
x = np.array([2, 4, 6, 8, 10, 12]).reshape(-1, 1)
y = np.array([6.5, 4.4, 3.8, 3.5, 3.1, 3.0])

# Побудова моделі
model = LinearRegression()
model.fit(x, y)

# Отримання коефіцієнтів
intercept = model.intercept_
slope = model.coef_[0]

print(f"Рівняння прямої:  $y = \{intercept:.3f\} + \{slope:.3f\} \cdot x$ ")

# Побудова графіка
plt.scatter(x, y, color='blue', label='Експериментальні дані')
plt.plot(x, model.predict(x), color='red', label='Лінія найменших квадратів')
plt.title('Метод найменших квадратів')
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
plt.grid(True)
plt.show()
```

Завдання 3

```
import numpy as np
import matplotlib.pyplot as plt
from numpy.polynomial.polynomial import Polynomial

# Вхідні точки
x = np.array([0.1, 0.3, 0.4, 0.6, 0.7])
```

```
y = np.array([3.2, 3.0, 1.0, 1.8, 1.9])
# Побудова полінома 4-го ступеня
coeffs = np.polyfit(x, y, 4)
poly = np.poly1d(coeffs)
# Виведення коефіцієнтів
print("Коефіцієнти полінома (від вищого до нижчого ступеня):")
print(coeffs)
# Побудова графіка
x_vals = np.linspace(min(x), max(x), 200)
y_vals = poly(x_vals)
plt.plot(x_vals, y_vals, label='Інтерполяційний поліном', color='orange')
plt.scatter(x, y, color='green', label='Задані точки')
plt.title('Інтерполяція поліномом 4-го ступеня (Завдання 3)')
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
plt.grid(True)
plt.show()
# Обчислення значень у проміжних точках
print(f"Значення полінома у x=0.2: {poly(0.2):.3f}")
print(f"Значення полінома у x=0.5: {poly(0.5):.3f}")
```