Міністерство освіти і науки України

Західноукраїнський національний університет

Факультет комп’ютерних інформаційних технологій

Кафедра інформаційно-обчислювальних систем і управління

Звіт про виконання лабораторної роботи №2

з дисципліни «Методи та системи штучного інтелекту»

Студента групи КН-33

Раківського Романа

Перевірив викладач

Коваль В. С.

м. Тернопіль – 2024

**Лабораторна робота №2**

Тема: Побудова регресора методом k-найближчих сусідів (k-nn)

Мета: Вивчити можливості аналізу даних з використанням регресора методом k-найближчих сусідів (k-nn)

**Вступ**

У даній роботі було реалізовано алгоритм KNN для регресії з метою прогнозування значень на основі згенерованих випадкових даних. Процес включав генерацію даних, їх нормалізацію, навчання моделі, оптимізацію параметра KK, а також візуалізацію результатів.

**Генерація випадкового набору даних**

Був згенерований набір з 1000 випадкових значень у діапазоні від 0 до 1000. Значення цільової змінної yyформувалися за допомогою лінійної функції з додаванням випадкового шуму, що надавало даним реалістичності.

np.random.seed(0)

X = np.random.rand(1000, 1) \* 1000 # Генерація 1000 випадкових значень в діапазоні [0, 1000]

y = 2 \* X.squeeze() + np.random.normal(0, 100, 1000) # Лінійна функція з шумом

**Нормалізація значень**

Для покращення навчання моделі було проведено нормалізацію значень вхідних даних за допомогою стандартного масштабування.

scaler = StandardScaler()

X\_scaled = scaler.fit\_transform(X)

**Розділення на навчальну та тестову вибірки**

Дані були розділені на навчальну (80%) та тестову (20%) вибірки для подальшого навчання та оцінки моделі.

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_scaled, y, test\_size=0.2, random\_state=42)

**Навчання KNN-регресора з різними значеннями KK**

Було проведено навчання моделі KNN-регресора для значень KK від 1 до 20. Для кожного KK розраховувалися середньоквадратична помилка (MSE) та коефіцієнт детермінації (R²).

k\_values = range(1, 21)

mse\_values = [] # Список для збереження значень середньоквадратичної помилки

r2\_values = [] # Список для збереження значень R^2

for k in k\_values:

knn = KNeighborsRegressor(n\_neighbors=k)

knn.fit(X\_train, y\_train)

y\_pred = knn.predict(X\_test)

mse = mean\_squared\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

mse\_values.append(mse)

r2\_values.append(r2)

**Вибір оптимального значення K**

Оптимальне значення K було визначено як те, яке забезпечує найменшу середньоквадратну помилку.

optimal\_k = k\_values[np.argmin(mse\_values)]

print(f"Оптимальне значення K: {optimal\_k}")

**Візуалізація результатів**

Для кращого розуміння роботи моделі були побудовані графіки:

* Графік середньоквадратичної помилки (MSE) в залежності від KK.
* Графік коефіцієнта детермінації (R²) в залежності від KK.
* Графік фактичних vs прогнозованих значень.
* Графік помилок регресії.
* Графік фактичні vs прогнозовані значення.

# Графік MSE в залежності від K

plt.subplot(3, 2, 1)

plt.plot(k\_values, mse\_values, marker='o')

plt.title('MSE в залежності від K')

plt.xlabel('K')

plt.ylabel('Середньоквадратична помилка (MSE)')

plt.xticks(k\_values)

plt.grid()

# Графік R^2 в залежності від K

plt.subplot(3, 2, 2)

plt.plot(k\_values, r2\_values, marker='o', color='orange')

plt.title('R^2 в залежності від K')

plt.xlabel('K')

plt.ylabel('Коефіцієнт детермінації (R^2)')

plt.xticks(k\_values)

plt.grid()

# Графік фактичних і прогнозованих значень

plt.subplot(3, 2, 3)

plt.scatter(range(len(y\_test)), y\_test, color='red', alpha=0.5, label='Фактичні значення')

plt.scatter(range(len(y\_pred\_optimal)), y\_pred\_optimal, color='blue', alpha=0.5, label='Прогнозовані значення')

plt.plot([0, len(y\_test)], [y.min(), y.max()], 'r--', lw=2) # Лінія y=x

plt.title(f'Фактичні vs Прогнозовані значення (K={optimal\_k})')

plt.xlabel('Індекс')

plt.ylabel('Значення')

plt.legend()

plt.grid()

# Графік помилок

plt.subplot(3, 2, 4)

plt.scatter(y\_pred\_optimal, errors, alpha=0.5)

plt.axhline(0, color='red', linestyle='--') # Лінія на нулі

plt.title('Помилки регресії')

plt.xlabel('Прогнозовані значення')

plt.ylabel('Помилка (Фактичні - Прогнозовані)')

plt.grid()

# Графік з крапочками

plt.subplot(3, 2, 5)

plt.scatter(y\_test, y\_pred\_optimal, alpha=0.5, color='purple')

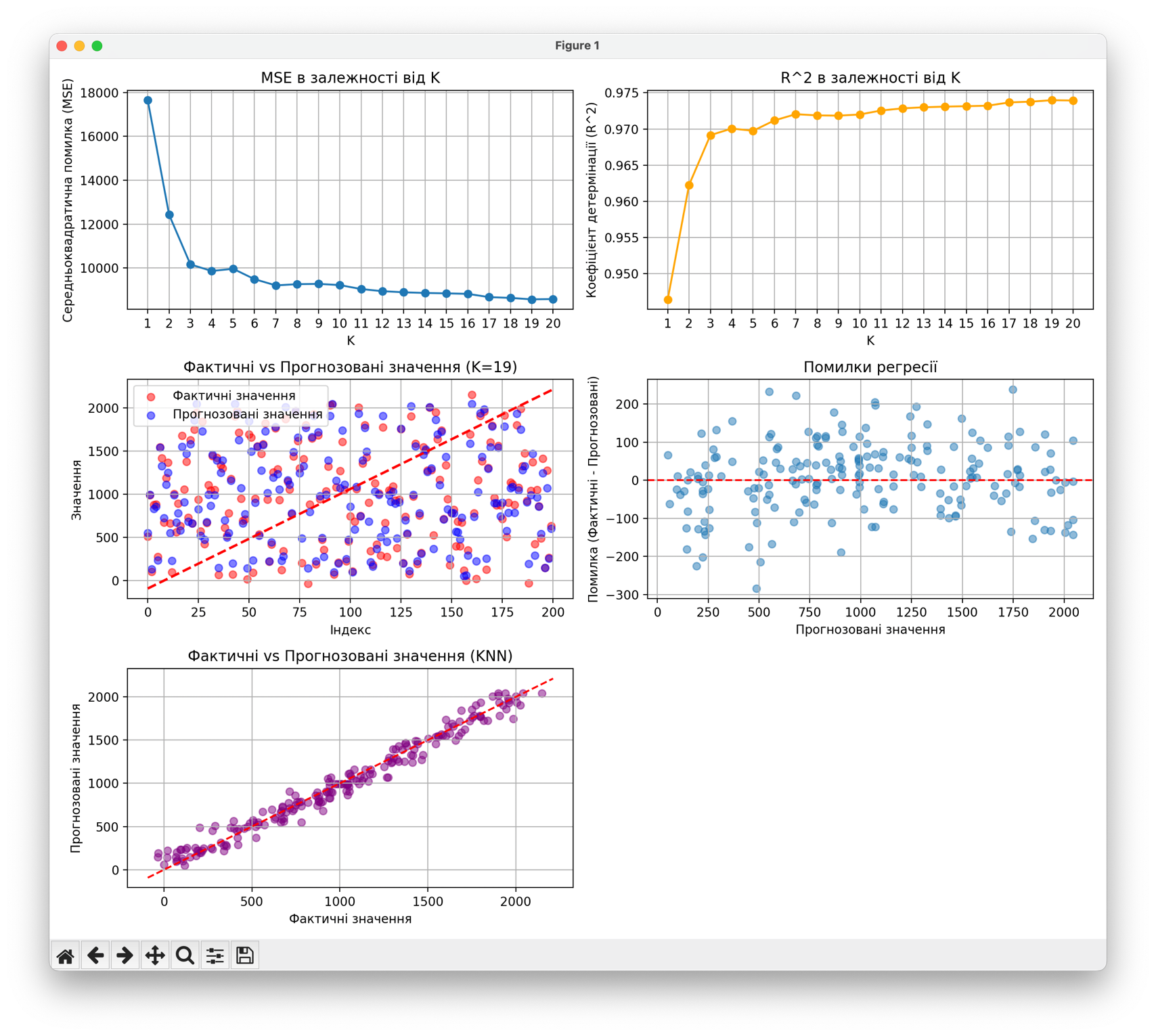
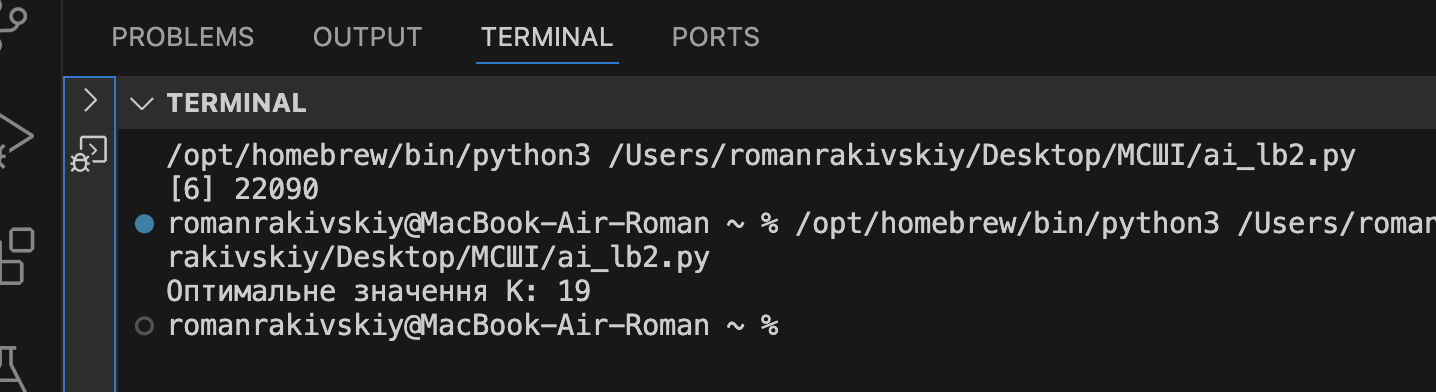
plt.plot([y.min(), y.max()], [y.min(), y.max()], 'r--') # Лінія y=x

plt.title('Фактичні vs Прогнозовані значення (KNN)')

plt.xlabel('Фактичні значення')

plt.ylabel('Прогнозовані значення')

plt.grid()



**Висновки**

Модель KNN показала хороші результати з мінімальною середньоквадратичною помилкою для оптимального значення K. Візуалізації підтверджують адекватність моделі, а також показують, що прогнози моделі наближені до фактичних значень.