Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота *№3*

із дисципліни ***«Методи і технології штучного інтелекту»***

Тема: ***«Дослідження алгоритму нечіткої кластеризації»***

Виконав:

Студент групи ІА-34

Ястремський Богдан

Перевірив:

старший викладач кафедри ІСТ

Польшакова Ольга Михайлівна

Київ — 2025

Тема: Дослідження алгоритму нечіткої кластеризації.

Мета: Вирішення практичного завдання кластеризації методами нечіткої логіки.

Примітки: Код усіх файлів буде винесено окремо в додаток А наприкінці звіту.

Хід роботи:

1. Необхідно сформулювати завдання в галузі обчислювальної техніки або програмування, для якої була б необхідна автоматична класифікація множини об'єктів, які задаються векторами ознак в просторі ознак.

   На сьогоднішній день, оцінки в школі, хоч і частково поверхнево, але показують рівень навчання в школі, університеті і інших навчальних закладах. Базуючись на них, можна класифікувати «слухачів» - учнів, студентів за оцінками – «відмінник», «середній» та «двієчник».

   Тому завдання полягає в тому, щоб класифікувати учнів за оцінками, до якої категорії він належав би.

2. Вирішити сформульовану задачу з використанням механізму кластеризації методами нечіткої логіки за допомогою програмних засобів моделювання або мови програмування високого рівня.

   Спочатку нам потрібні дані, які зазвичай беруться з сайтів (у форматі .csv, .json і т.д). Але в нашому випадку, заготовлених даних немає.

   Тому гарною ідеєю буде їх згенерувати. Генерувати будемо за допомогою ф-ції нормального Гаусівського розподілу np.random.normal, яка приймає в якості параметрів:

   - 1d-масив центр(-ів) початку(-ів) генерації
   - числове значення ширини кривої Гауса
   - 1d-кортеж бажаного розміру даних (рядки-стовпц).

Далі, пропустимо всі дані, які не входять у наші задані межі (1-12).

Після об’єднання даних в один єдиний масив, виведемо деякі з них з 3 різних кластерів. Наприклад 100-110 (двієчники), 400-410 (середні), 700-710 (відмінники). Варто пам’ятати, що, не дивлячись на те, що ми генерували 800 студентів, після np.clip їхня к-сть може бути меншою, оскільки ми не задаємо жодних обмежень на так званий «шум» в генерації.

В реальності дані фільтруються на етапі додавання їх в сервіс.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import skfuzzy as fuzz

num_students = 800  # Students quantity
num_subjects = 4  # Subjects quantity
num_clusters = 3  # Clusters quantity (can be "excellent student", "good
student", "f student" for example)
min_grade, max_grade = 1, 12

# Setting centers for each cluster (can be edited for best choice)
clusters_centers_ratings = [4, 7, 10]
clusters_centers = [[i] * num_subjects for i in clusters_centers_ratings]

# Data for each cluster
clusters_data = []

for i in range(num_clusters):
    cluster_size = (num_students // num_clusters, num_subjects)

    cluster_data = np.random.normal(clusters_centers[i], 1.1, cluster_size)
    cluster_data = np.clip(cluster_data, min_grade, max_grade)
    clusters_data.append(cluster_data)

clusters_data = np.vstack(clusters_data)

columns = ['Алгебра', 'Англійська', 'Геометрія', 'Фізика']
df = pd.DataFrame(clusters_data, columns=columns)

# Printing test data
print("Двієчники:")
print(df.iloc[100:110])

print("\nСередні:")
print(df.iloc[400:410])

print("\nВідмінники:")
print(df.iloc[700:710])

# Plotting generated data (for Algebra and English e.g.)
plt.scatter(clusters_data[:, 0], clusters_data[:, 1])
plt.title("Згенеровані дані")
plt.show()
```

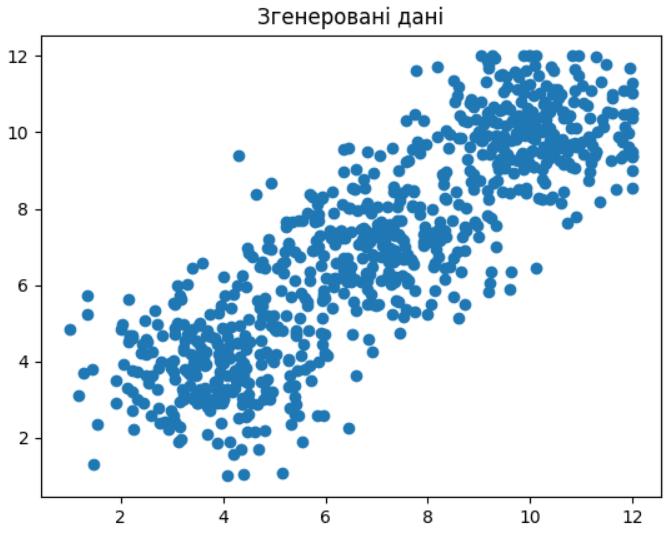

Рис. 3.1 – Згенеровані дані

Деякі дані:

Двієчники:

	Алгебра	Англійська	Геометрія	Фізика
100	5.396627	3.375041	4.944259	4.589545
101	4.807874	5.651000	4.324725	3.779124
102	4.283223	3.646368	1.595308	5.294538
103	3.252789	4.554923	4.108357	3.610279
104	3.964304	3.963457	3.687918	3.877636
105	4.683193	1.701068	4.688306	5.290176
106	4.309067	2.532287	3.866706	4.097322
107	3.668598	4.693384	2.761084	5.124685
108	3.623780	4.083313	3.771672	3.268466
109	4.339259	3.849648	4.263276	3.175859

Середні:

	Алгебра	Англійська	Геометрія	Фізика
400	6.918096	4.249618	7.822323	8.770859
401	4.647767	8.394069	8.062244	7.154621
402	7.469617	6.710195	7.520677	7.938351
403	4.296618	9.402953	6.943547	6.283531
404	7.914900	7.122013	5.472434	5.332276
405	5.967549	5.854987	4.644990	6.546855
406	6.800162	7.105699	7.777861	7.697242
407	7.755642	5.317821	6.276459	9.328567
408	9.184999	5.837059	6.662730	6.846597
409	8.710179	5.503627	5.044920	7.625824

Відмінники:

	Алгебра	Англійська	Геометрія	Фізика
700	9.332755	7.570735	11.298831	11.077122
701	9.808336	10.335231	10.139141	9.609369
702	7.842772	9.452940	10.727295	8.902689
703	10.461485	11.626551	10.830208	9.733981
704	10.890357	9.618781	11.248281	9.406927
705	10.052243	9.087305	9.213253	9.394720
706	10.876744	11.110540	10.920989	8.125194
707	11.803863	9.762216	11.105518	10.717570
708	12.000000	10.493211	9.312409	11.416036
709	10.946497	9.620391	10.291076	10.879400

3. Знайти центри кластерів і побудувати графік зміни значень цільової функції.

За допомогою fuzz.cluster.cmeans ми проводимо алгоритм нечіткої кластеризації. Передаємо у функцію транспонований масив даних, к-сть кластерів, «м’якість» кластеризації (чим більше, тим нечіткіше), похибку і максимальну к-сть ітерацій.

Отримуємо на виході center, u, u0, d, jm, p, fpc:

- center: матриця центрів кластерів
- u: матриця, яка зберігає ступені належності кожного елемента до кожного з кластерів
- u0: початкова матриця, яка зберігає початкові ступені належності кожного елемента до кожного з кластерів
- d: матриця відстаней між кожним елементом (об'єктом) і центром кожного з кластерів
- jm: матриця цільової функції
- p: к-сть проведених ітерацій
- fpc: Fuzzy Partition Coefficient – коефіцієнт розбиття на кластери. 0 – дуже нечітка кластеризація, 1 – чітка кластеризація.

Далі виводимо графік розбиття на кластери з центрами, графік цільової функції та FPC.

Бачимо, що кластеризація успішна. FPC = 0.5, що означає, що кластеризація може бути охарактеризована як середньої чіткості.

```
# FCM for data
center, u, u0, d, jm, p, fpc = fuzz.cluster.cmeans(
    clusters_data.T, num_clusters, 3, error=0.005, maxiter=100
)

fuzzy_labels = np.argmax(u, axis=0)

for i in range(num_clusters):
    cluster_points = clusters_data[fuzzy_labels == i]
    plt.scatter(cluster_points[:, 0], cluster_points[:, 1])

plt.scatter(center[:, 0], center[:, 1], marker="*", color="black")
plt.title("Кластери з центрами")
plt.show()

# Objective function plotting
plt.plot(jm)
plt.xlabel("Кількість ітерацій")
plt.ylabel("Значення цільової функції")
plt.grid(True)
plt.show()

# Fuzzy partition coefficient
print(f"Якість кластеризації: {fpc}")
```

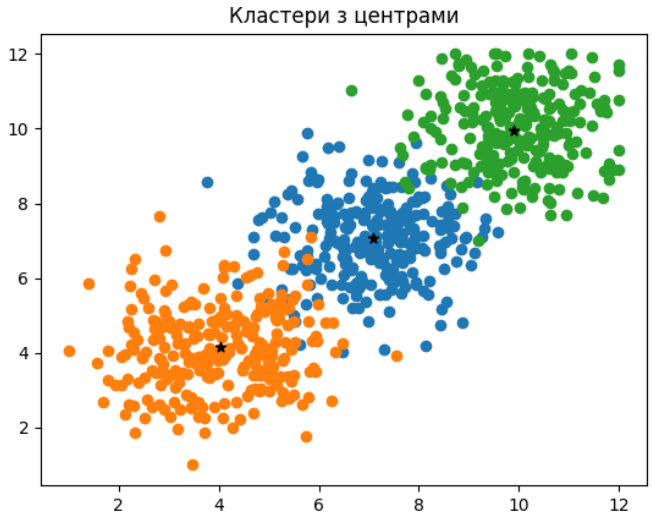

Рис. 3.2 – Кластери з центрами

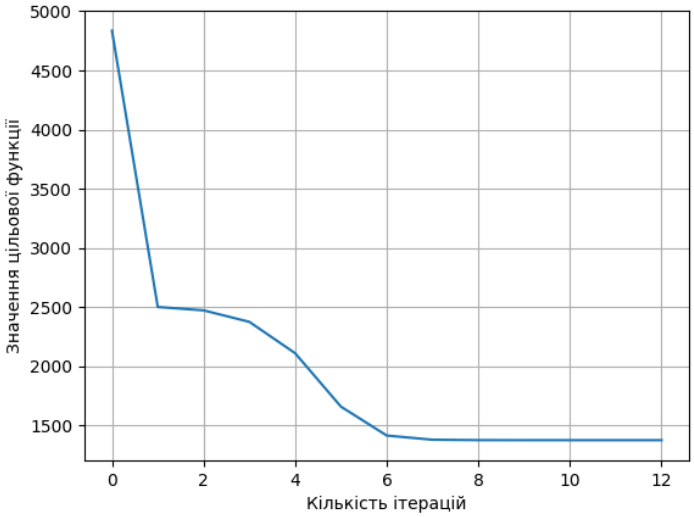

Рис. 3.3 – Графік цільової функції

Якість кластеризації:

```
Якість кластеризації: 0.5010077779211003
```

Висновок: на даному лабораторному занятті я познайомився з алгоритмом нечіткої кластеризації, дізнався, яким чином він розділяє дані на кластери. Спочатку я визначив тип даних для ознак, який підходить найкраще для роботи з їхньої кластеризації – такі дані, які є числовими, і є дуже схожими: такі як оцінки в школі з предметів, зарплата робітників, ціна товару і т.д. Далі я визначив по яким ознакам робити кластеризацію – оцінки з двох предметів, оскільки саме 2D графік легше відобразити і легше оцінити. У висновку я отримав графік, де дані були доволі точно розбиті на кластери. Це видно і зі значення fpc = 0.5, який каже, що кластеризація була чіткою на 50% - дані належать своїм кластерам, але є такі, що знаходяться між 2 кластерами одночасно, не маючи конкретної належності.

Відповіді на контрольні питання:

1. Що таке кластеризація і яке її основне призначення?

   Кластеризація — це статистичний метод і техніка машинного навчання для групування схожих об'єктів (точок даних) у множини, які

називаються кластерами. Головна мета полягає в тому, щоб об'єкти всередині одного кластера були максимально схожими між собою, а об'єкти з різних кластерів — максимально відмінними.

2. У чому відмінність кластеризації від класифікації?

   Головна відмінність кластеризації від класифікації полягає в тому, що перелік груп чітко не заданий і визначається в процесі роботи алгоритму.

3. Які основні підходи до кластеризації існують?
   - **Жорстка кластеризація** (K-means).
   - **М'яка кластеризація** (Fuzzy C-means).
   - **Ієрархічна кластеризація**. Створюється деревоподібна структура (дерево кластерів або дендрограма), де кожен об'єкт спочатку розглядається як окремий кластер, а потім ці кластери об'єднуються в більші кластери, поки всі об'єкти не об'єднаються в один великий кластер.
   - **Щільнісна кластеризація**. Кластери визначаються як області високої щільності, де об'єкти щільно розташовані один біля одного. Кластери розділяються областями низької щільності.
   - **Кластеризація на основі моделей**. Кластери моделюються як розподіли ймовірностей і алгоритм намагається знайти найбільш ймовірний набір таких моделей для даних.
   - **Кластеризація на основі графів**. Об'єкти з'єднуються як вузли в графі, а подібність між ними визначається за вагою. Кластери визначаються як сильно з'єднані компоненти графу
   - **Когнітивні методи кластеризації**. Використовує когнітивні моделі для розуміння або імітації людського сприйняття кластерів. Цей підхід може використовувати нейронні мережі або глибоке навчання для класифікації і кластеризації

4. Що таке жорстка (чітка) кластеризація?

   Кожен об’єкт або належить кластеру, або ні.

5. Які недоліки має жорстка кластеризація в аналізі даних?

   Зазвичай важко конкретно віднести якісь дані до конкретного кластеру.

6. У чому полягає суть нечіткої кластеризації?

   Кожен об'єкт може належати до кількох кластерів з різним ступенем належності. Наприклад, об'єкт може належати до кластера з 70% й до іншого з 30%.

7. Чим відрізняється нечітка кластеризація від класичної (жорсткої)?

   Жорстка - або належить кластеру, або ні. Нечітка - може належати до кількох кластерів з різним ступенем належності.

8. Що таке матриця приналежності у нечіткій кластеризації?

   Матриця приналежності у нечіткій кластеризації (наприклад, в алгоритмі Fuzzy C-Means) - це матриця, яка описує ступінь належності кожного елемента (об'єкта) до кожного з кластерів. Вона є основним елементом нечіткої кластеризації, де кожен об'єкт може належати до кількох кластерів одночасно з різними ступенями належності.

9. Які основні етапи алгоритму Fuzzy C-Means (FCM)?

   Крок 1. Ініціалізація. Вибираються наступні параметри:

   – необхідну кількість кластерів N, 2 <N <K;
   – тип відстаней (наприклад, відстань по Евклиду);
   – фіксований параметр q (зазвичай 1,5);
   – початкова (на нульовій ітерації) матриця функцій приналежності $U^{(0)} = (\mu_{jk})^{(0)}$ об'єктів x_k $(k = \overline{1,K})$ з урахуванням заданих початкових центрів кластерів c_j $(j = \overline{1,N})$.

Крок 2. Регулювання позицій c (jt) центрів кластерів. На t-м ітераційне кроці при відомій матриці $\mu^{(Jkt)}$ обчислюється c (jt) відповідно до викладеного вище рішенням системи рівнянь.

Крок 3. Коригування значень приналежності μ_{jk}. З огляду на відомі c (jt), обчислюються $\mu^{(Jkt)}$, Якщо xk □ *cj*, в іншому випадку:

$$\mu_{jk}^{(t+1)} = \begin{cases} 1, & \text{если } k = j, \\ 0, & \text{если } k \neq j. \end{cases}$$

Крок 4. Зупинка алгоритму.

Алгоритм нечіткої кластеризації зупиняється при виконанні наступної умови:

$$\left\| U^{(t+1)} - U^{(t)} \right\| \leq \varepsilon,$$

де ‖ ‖ - матрична норма (наприклад, Евклідова норма); ε - заздалегідь задається рівень точності.

10. У чому відмінність між алгоритмами k-means та Fuzzy C-Means?

    k-means є жорсткою кластеризацією, FCM – нечітка.

11. Як інтерпретується значення ступеня приналежності об’єкта до кластера у FCM?

    У нечіткій кластеризації, зокрема в Fuzzy C-Means (FCM), кожен об'єкт має ступінь приналежності до кожного з кластерів. Це значення вказує на те, наскільки сильно об'єкт належить до певного кластеру.

12. Які переваги надає застосування нечіткої логіки в задачах кластеризації?

    - Коли об'єкти не можна чітко віднести до одного кластера.
    - Нечітка логіка дозволяє зручно працювати з неоднорідними даними, де кластери можуть бути перекритими або мати складну структуру.
    - Нечіткі алгоритми кластеризації здатні ефективно працювати з шумом і викидами.

13. У яких сферах застосовується нечітка кластеризація (наприклад, медицина, обробка зображень)?
    - Аналіз тексту та обробка природної мови
    - Фінансові та економічні аналізи
    - Географічне та просторове планування
    - Біоінформатика
    - Маркетинг та поведінковий аналіз
    - Аналіз соціальних мереж
    - Системи рекомендацій

14. Які обмеження чи труднощі можуть виникати при використанні нечіткої кластеризації?
    - Вибір параметрів: Параметри нечіткої кластеризації, такі як кількість кластерів (K) або параметр «м'якості» (m в Fuzzy C-Means), часто потрібно вибирати вручну.
    - Нечітка кластеризація, зокрема алгоритм Fuzzy C-Means, може мати високу обчислювальну складність, особливо для великих наборів даних або великої кількості кластерів. Алгоритми потребують виконання багатьох ітерацій для оновлення центрів кластерів та належності.
    - Алгоритм FCM вимагає визначення функції належності між об'єктами та кластерами, що може бути складним у випадках, коли дані мають складну структуру або є надто неоднорідними.

15. Як можна комбінувати нечітку логіку з іншими методами аналізу даних для підвищення точності кластеризації?
    - Поєднання з алгоритмами машинного навчання (класифікація)
    - Поєднання з методами генетичних алгоритмів (найкращі параметри кластеризації)