**Лекція 4. Кластерізація**

1. **Поняття та типи**

Кластеризація — це набір методів без учителя для групування даних за певними критеріями в так звані кластери, що дозволяє виявляти схожості та відмінності між об'єктами, а також спрощувати їх аналіз і візуалізацію. Через часткову схожість у постановці завдань з класифікацією кластеризацію ще називають unsupervised classification. (классифікація без підкріплення).

**Області застосування кластеризації та її різновиди**Кластеризація широко застосовується в машинному навчанні для вирішення різноманітного спектру завдань:

1. класифікація (визначення, до якого класу належить кожен об'єкт або виділення нових класів, які не були відомі заздалегідь);
2. сегментація ринку (поділ потенційних клієнтів на групи за їхніми характеристиками для розробки більш ефективних стратегій у маркетингу та продажах);
3. сегментація зображень (поділ зображення на сегменти або групи пікселів);
4. кластеризація геоданих (групування даних за їхнім географічним розташуванням, наприклад, поділ районів на безпечні та небезпечні, багаті й бідні тощо);
5. зниження розмірності (зменшення кількості ознак шляхом об'єднання схожих в один кластер).

**Існує багато різних типів** кластеризації, які можна розділити за такими критеріями:

За способом формування кластерів:

1. Роздільні (partitioning) — ділять дані на задану кількість кластерів, мінімізуючи відстань всередині кластера і максимізуючи відстань між кластерами (наприклад, K-means).
2. Засновані на щільності (density-based) — групують точки, що знаходяться в областях з високою щільністю, і відокремлюють їх від областей з низькою щільністю (наприклад, DBSCAN).
3. Засновані на сітці (grid-based) — ділять простір на комірки сітки та аналізують щільність даних у кожній комірці (наприклад, STING).
4. Засновані на моделі (model-based) — передбачають, що дані породжені певною статистичною моделлю і намагаються підібрати параметри цієї моделі (наприклад, суміші Гауссіанів).
5. Засновані на графах (graph-based) — використовують графове представлення даних і ділять його на підграфи, що відповідають кластерам (наприклад, спектральна кластеризація).
6. Засновані на підпросторах (subspace-based) — шукають кластери в підпросторах ознак, а не в усьому просторі (наприклад, CLIQUE).
7. Засновані на ансамблі (ensemble-based) — комбінують результати різних алгоритмів кластеризації, щоб отримати більш стабільний та надійний поділ (наприклад, CSPA).

За ступенем вкладеності кластерів:

1. Плоскі (flat) — ділять дані на один рівень кластерів, не враховуючи їхню ієрархію (наприклад, K-means).
2. Ієрархічні (hierarchical) — ділять дані на кілька рівнів кластерів, враховуючи їхню ієрархію. Існують два основні підходи до ієрархічної кластеризації: агломеративний (починається з того, що кожен об'єкт є окремим кластером, а потім поступово найближчі кластери об'єднуються в більші) та дивізивний (починається з того, що всі об'єкти складають один кластер, а потім поступово розділяються на дрібніші кластери).

За ступенем перетину кластерів:

1. Виключні (exclusive) — кожен об'єкт належить лише одному кластеру (наприклад, K-means).
2. Перекривні (overlapping) — кожен об'єкт може належати до кількох кластерів (наприклад, MCOKE).
3. Нечіткі (fuzzy) — кожен об'єкт належить кожному кластеру з певною мірою належності (наприклад, fuzzy K-means).
4. **Кластерізація K-mean**

Сутність k-means (k-середніх) полягає в розбитті набору даних на k кластерів, де кожен кластер представлений своїм центроїдом (середнім значенням). Основна мета алгоритму — мінімізувати суму квадратів відстаней між кожним об'єктом і найближчим центроїдом.

Алгоритм k-means працює наступним чином:

1. Вибір кількості кластерів (k) — користувач задає кількість кластерів, на які потрібно поділити дані.
2. Ініціалізація центроїдів — випадковим чином вибираються початкові центроїди для кожного з кластерів.
3. Призначення об'єктів кластерам — кожен об'єкт даних призначається до найближчого центроїда (згідно з обраною метрикою, зазвичай це евклідова відстань).
4. Оновлення центроїдів — обчислюється новий центроїд для кожного кластера як середнє значення всіх об'єктів, що належать цьому кластеру.
5. Повторення кроків 3 і 4 — процес призначення об'єктів і оновлення центроїдів повторюється до тих пір, поки центроїди не перестануть змінюватися або зміни будуть мінімальними.
6. Зупинка — алгоритм зупиняється, коли більше немає значних змін у положенні центроїдів.

K-means простий у реалізації та застосуванні, але має певні обмеження: алгоритм чутливий до початкового розміщення центроїдів і може потрапити в локальний мінімум.



**Модифікація k-mean++**

Метод **k-means++** — це вдосконалення стандартного алгоритму **k-means**, спрямоване на покращення ініціалізації центроїдів, що дозволяє уникнути проблем із вибором поганих початкових значень і зменшує ймовірність потрапляння в локальні мінімуми.

Основна відмінність полягає в розумному виборі початкових центроїдів, що дозволяє швидше збігатися до оптимального розв'язку.

Процес ініціалізації в k-means++:

1. **Вибір першого центроїда** — перший центроїд вибирається випадковим чином із набору даних.
2. **Вибір наступних центроїдів**:
	* Для кожної точки обчислюється відстань до найближчого вже обраного центроїда.
	* Кожна точка отримує вагу пропорційно квадрату цієї відстані.
	* Наступний центроїд вибирається випадковим чином, але з більшою ймовірністю для точок, які знаходяться на великій відстані від уже обраних центроїдів.
3. **Повторення** — крок 2 повторюється, поки не буде вибрано **k** центроїдів.

Після того як ініціалізація завершена, алгоритм працює так само, як і стандартний k-means: призначення точок до найближчих центроїдів і оновлення положень центроїдів на основі середніх значень точок у кластерах.

Переваги k-means++:

* **Швидша збіжність** — метод зменшує кількість ітерацій, необхідних для збіжності.
* **Краща якість кластерів** — центроїди краще розташовані, що дозволяє отримати більш рівномірний розподіл кластерів.
* **Зниження ризику локальних мінімумів** — правильна ініціалізація зменшує ймовірність того, що алгоритм потрапить у локальний мінімум.