

LL3. Использование различных алгоритмов кластеризации для различных типов данных. Визуализация данных.

Основные концепции

- **Кластер** = группа данных, которые похожи друг на друга.
- **Кластеризация (группировка)** = процесс выявления кластеров.
- **Прототип (представитель)** = "объект", который представляет данные в кластере.
- **Центроид** = среднее значение данных в кластере; центроид не обязательно является элементом из набора данных.
- **Медоид** = данные из кластера, которые наиболее близки к среднему значению кластера; медоид всегда принадлежит набору данных.
- **Радиус кластера** = стандартное отклонение данных в кластере (среднее расстояние до центроида).
- **Диаметр кластера** = максимальное расстояние (несходство) между любыми двумя точками в кластере.

Типы кластеризации

Четкая (Crisp) vs нечеткая (Fuzzy) кластеризация

- **Четкая кластеризация** = каждая точка данных принадлежит только к одному кластеру.
- **Нечеткая кластеризация** = одна точка данных может принадлежать нескольким кластерам с определенной степенью принадлежности.

Плоская (Flat) vs иерархическая (Hierarchical) кластеризация

- **Плоская (разбиение на кластеры) кластеризация** = результат – это набор кластеров (разбиение данных).
- **Иерархическая кластеризация** = результат – это иерархическая структура разбиений.

Типы алгоритмов

- **Алгоритмы разбиения** (пример: kMeans, Fuzzy cMeans).
- **Иерархические алгоритмы** (пример: агломеративные алгоритмы, дивизивные алгоритмы).
- **Алгоритмы, основанные на плотности** (пример: DBSCAN).
- **Алгоритмы, основанные на вероятностных моделях** (пример: EM = Expectation Maximization).

Метрики качества Не существует единого индикатора для оценки качества кластеризации. Наиболее распространенный подход заключается в оценке:

- **Компактности кластеров** (внутрикластерная изменчивость – должна быть низкой).
- **Степени разделения данных между разными кластерами** (межкластерная изменчивость – должна быть высокой).

```
from sklearn.metrics import silhouette_score
```

Индивидуальное задание: Выберите для реализации (на языке Python) по одному методу из каждой вышеуказанной категории (минимум четыре алгоритма). Объясните концепцию и основные этапы работы каждого алгоритма. Выберите или создайте набор данных в соответствии с требованиями каждого разработанного алгоритма. Оцените полученные результаты.