ИНТЕЛЛЕКТУАЛЬНОЕ МОДЕЛИРОВАНИЕ

Образовательная программа «6B06103 – Компьютерная инженерия»

Лекция 13

Кластерный анализ: метод k-средних

Неделя 13 · Tan, Steinbach, Kumar: 8.1, 8.2, 8.5

Цель и план лекции

Цель: понять принципы кластерного анализа и научиться применять k-средних, выбирать K и интерпретировать кластеры

1. Кластеризация: виды и типы кластеров
2. Алгоритм k-средних и SSE
3. Ручной расчёт на шести точках
4. Инициализация, k-means++, ограничения
5. Метод локтя, силуэт, scikit-learn

Что такое кластерный анализ

Внутри группы объекты похожи, между группами — различны

- **Классификация без учителя:** меток нет, смысл группам даёт аналитик
- Понятие кластера нечёткое: 20 точек можно разбить на 2, 4 или 6 групп
- **Понимание данных:** сегменты клиентов, темы документов, климатические зоны
- **Преобразования:** сжатие, ускорение поиска соседей

Виды кластеризаций

Плоская / иерархическая

Непересекающиеся группы или вложенные кластеры (дендрограмма)

Исключающая / перекрывающаяся / нечёткая

Один кластер, несколько, или вес от 0 до 1

Полная / частичная

Все объекты в кластерах или шум остаётся вне

Типы кластеров

Тип	Определение	Форма
Хорошо разделённые	Ближе к любому объекту своего кластера	Любая
На прототипах	Ближе к центроиду своего кластера	Шарообразная
На графах	Цепочки близких соседей	Произвольная
На плотности	Плотные области среди разреженных	Произвольная
Концептуальные	Общее свойство	По концепции

Базовые алгоритмы: k-средних, агломеративная иерархическая, DBSCAN

Алгоритм k-средних

1

Выбрать K начальных центроидов

2

Отнести каждую точку к ближайшему центроиду

3

Пересчитать центроиды как средние

↺

Повторять, пока центроиды меняются

- K задаёт пользователь; центроид — среднее точек кластера
- Близость: евклидово расстояние, для документов — косинусное сходство

Целевая функция SSE

$$\mathrm{SSE} = \sum_{i=1}^{K} \sum_{x \in C_i} \mathrm{dist}(c_i, x)^2$$

$$c_i = \frac{1}{m_i} \sum_{x \in C_i} x$$

- Центроид (1, 1), (2, 3), (6, 2) = (3, 2)
- Оба шага уменьшают SSE → алгоритм всегда сходится
- Но находит лишь **локальный минимум**, зависящий от начала
- Часто останавливаются, когда кластер меняют < 1 % точек

Ручной расчёт: итерация 1

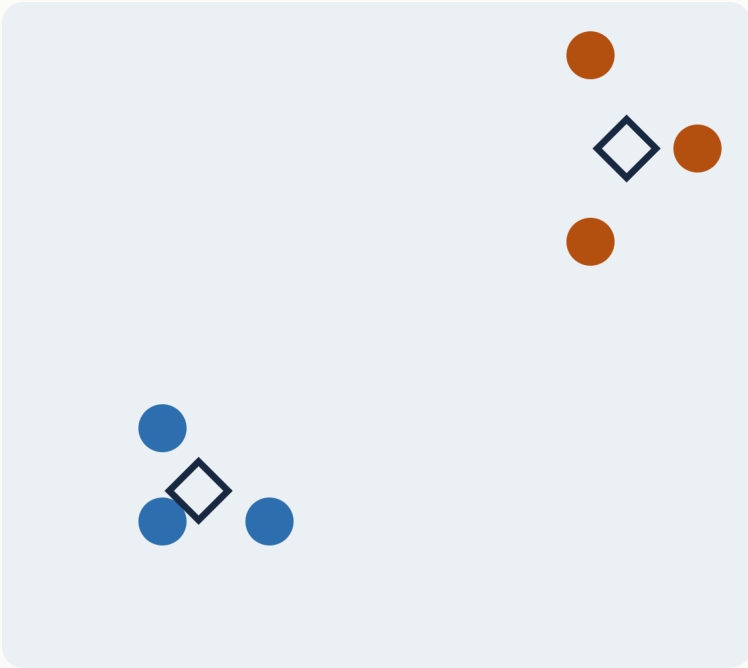
P1 (1, 1), P2 (2, 1), P3 (1, 2), P4 (5, 4), P5 (6, 5), P6 (5, 6); K = 2; C1 = P1, C2 = P2

Точка	До C1 (1; 1)	До C2 (2; 1)	Кластер
P1 (1, 1)	0,00	1,00	1
P2 (2, 1)	1,00	0,00	2
P3 (1, 2)	1,00	1,41	1
P4 (5, 4)	5,00	4,24	2
P5 (6, 5)	6,40	5,66	2
P6 (5, 6)	6,40	5,83	2

Новые центроиды: $C_1 = (1; 1{,}5)$, $C_2 = (4{,}5; 4)$

Итерации 2–3 и результат

- Итерация 2: P2 переходит в кластер 1
- $C_1 = \left(\frac{4}{3}, \frac{4}{3}\right) \approx (1{,}33; 1{,}33)$
- $C_2 = \left(\frac{16}{3}, 5\right) \approx (5{,}33; 5)$
- Итерация 3: изменений нет — стоп
- Кластеры {P1, P2, P3} и {P4, P5, P6}
- $\mathrm{SSE} = \mathbf{1{,}333 + 2{,}667 = 4{,}0}$



Выбор начальных центроидов

Разные запуски — разные решения с разной SSE

- **Многократный запуск** и выбор минимальной SSE (n_init)
- **k-means++**: следующий центроид — с вероятностью ∝ квадрату расстояния до ближайшего выбранного (по умолчанию в scikit-learn)
- Центроиды из иерархической кластеризации выборки
- **Бисекционный k-средних**: делить кластер с наибольшей SSE на два

Практические вопросы

Проблема	Решение
Пустой кластер	Взять самую удалённую точку или точку из кластера с max SSE
Выбросы	Удалить до кластеризации или после — точки, сильно растящие SSE
Разный масштаб	Стандартизировать признаки (доход доминирует над возрастом)
Постобработка	Делить кластеры с большой SSE, объединять близкие

Сложность: время $O(I \cdot K \cdot m \cdot n)$ — линейна по числу точек

Сильные и слабые стороны

Плюсы

- Простота
- Эффективность на больших данных
- Понятные центроиды

Минусы

- K нужно задать заранее
- Плохо: разный размер, плотность, не шарообразная форма
- Чувствителен к выбросам и инициализации
- Нужно среднее (иначе k-медоиды, k-моды)

Сложная форма и шум — DBSCAN; иерархия — агломеративная кластеризация

Оценка качества и метод «локтя»

- **Внутренние меры:** связность (cohesion) и разделённость (separation)
- SSE — мера связности: меньше — компактнее
- SSE всегда падает с ростом K (при K = m равна 0)

Локоть: строим SSE против K и ищем точку, после которой SSE убывает медленно

Внешние меры при известных метках: ARI — 1 при полном совпадении, ≈ 0 для случайного разбиения

Коэффициент силуэта

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}$$

- a_i — среднее расстояние до объектов своего кластера
- b_i — минимальное среднее расстояние до другого кластера
- ≈ 1 — хорошо вписан; ≈ 0 — на границе; < 0 — не в том кластере
- Выбирают K с наибольшим средним силуэтом

Для примера с шестью точками средний силуэт = 0,75

Проверка в scikit-learn

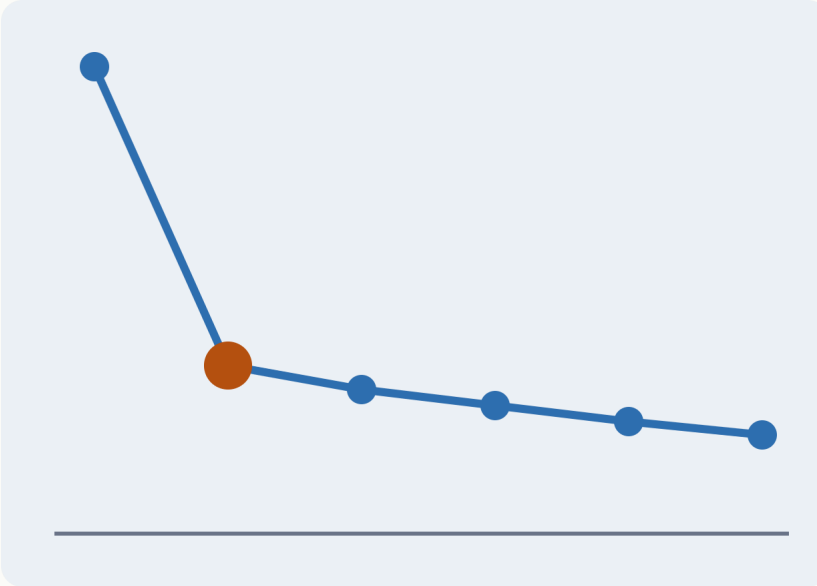
```
P = np.array([[1,1],[2,1],[1,2],[5,4],[6,5],[5,6]])
km = KMeans(n_clusters=2, init=P[:2], n_init=1).fit(P)
km.labels_           # [0 0 0 1 1 1]
km.cluster_centers_ # [[1.33 1.33] [5.33 5.]]
km.inertia_          # SSE = 4.0
km.n_iter_           # 3
silhouette_score(P, km.labels_)  # 0.75
```

inertia_ — это SSE; labels_ — номера кластеров; cluster_centers_ — центроиды

Сегментация 250 клиентов: выбор K

Признаки: возраст, доход, индекс расходов; стандартизация, K = 2…7

K	SSE	Силуэт
2	279,9	0,591
3	100,7	0,648
4	86,5	0,503
5	76,8	0,511
6	67,6	0,430
7	59,2	0,301



Локоть и максимум силуэта — при K = 3

Лекция 13 · Метод k-средних · график: SSE для K = 2…7, оранжевая точка — K = 3

Профили кластеров (K = 3)

Обеспеченные, экономные

Возраст 40

Доход 594

Расходы 28

100 клиентов

Молодые, активно тратят

Возраст 24

Доход 244

Расходы 69

80 клиентов

Старший возраст, умеренные

Возраст 54

Доход 356

Расходы 45

70 клиентов

Номера кластеров произвольны; смысл даёт аналитик по центроидам в исходных единицах

Iris и «полумесяцы»: сравнение с истинными метками

Iris, K = 3

0,73

ARI

setosa — отдельно; versicolor и virginica смешаны (16 цветков)

make_moons, K = 2

0,24

ARI

Режет прямой линией, а не по форме — нужен DBSCAN

Выводы

- Кластеризация — обучение без учителя
- k-средних минимизирует SSE и сходится к локальному минимуму
- Важны k-means++, многократные запуски, масштабирование
- K выбирают методом локтя и по силуэту
- Слабость — разный размер, плотность и сложная форма

Далее: ассоциативные правила

Контрольные вопросы

1. Чем кластеризация отличается от классификации? Три примера применения.
2. Виды кластеризаций и типы кластеров по Tan и соавт.
3. Шаги k-средних и формула SSE; почему сходится, но не всегда оптимально?
4. Центроид точек (2, 4), (4, 6), (6, 2), (8, 8) и SSE кластера.
5. k-means++, метод локтя, силуэт; s_i при $a_i = 2$, $b_i = 5$.
6. Программа: масштабирование, выбор K = 2…8 по силуэту, средние признаков по кластерам.

Литература

1. Tan P.-N., Steinbach M., Kumar V. Introduction to Data Mining. — Pearson, 2014. — 8.1; 8.2; 8.5.
2. VanderPlas J. Python Data Science Handbook. — O’Reilly, 2016. — In Depth: k-Means Clustering.
3. Мюллер А., Гвидо С. Введение в машинное обучение с помощью Python. — 2017. — Глава 3.
4. Грас Дж. Data Science. Наука о данных с нуля. — 2020. — Глава 20.
5. Arthur D., Vassilvitskii S. k-means++: The Advantages of Careful Seeding. — SODA, 2007.