Лекция 11

Кластеризация

Машинное обучение

Сергей Муравьёв / Андрей Фильченков

25.03.2020

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

- Слайды доступны: **shorturl.at/ltVZ3**
- Видео доступны: **shorturl.at/hjyAX**

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

Две задачи вероятностной классификации

Первая задачи: **восстановление плотности вероятности**

Дано: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{|\mathcal{D}|}$.

Задача: найти эмпирические оценки $\widehat{\Pr}(y)$ и $\hat{p}(x|y)$, $y \in Y$.

Вторая задачи: **минимизация среднего риска**

Дано:

- Априорные вероятности $\Pr(y)$,
- Правдоподобия $p(x|y), y \in Y$.

Проблема: найти классификатор a, который минимизирует $R(a)$.

Какая из этих двух задач уже решена и каков ответ?

Восстановление смеси распределения

Модель смеси генеративных распределений:

$$p(x) = \sum_{j=1}^{k} w_j p_j(x),$$

где $w_j \geq 0, \sum_{j=1}^{k} w_j = 1; p_j(x) = \varphi(x; \theta_j)$ — функция правдоподобия j-го компонента смеси, w_j — его априорная вероятность, k — количество компонентов смеси.

Две задачи:

1) С заданным набором данных $\mathcal{D}_s \sim p(x)$, числом k и функцией φ оценить вектор параметров $\Theta = (w_1, \dots, w_k, \theta_1, \dots, \theta_k)$.
2) Найти k.

Решения задач

Мы знаем, как решать такие задачи:

путем максимизации логарифма правдоподобия

$$L(\Theta) = \ln \prod_{i=1}^{m} p(x_i) = \sum_{i=1}^{m} \ln \sum_{j=1}^{k} w_j p_j(x_i; \theta_j) \to \max_{\Theta},$$

где $m = |\mathcal{D}_S|$.

Тогда в чем состоит задача?

Решения задач

Мы знаем, как решать такие задачи:

путем максимизации логарифма правдоподобия

$$L(\Theta) = \ln \prod_{i=1}^{m} p(x_i) = \sum_{i=1}^{m} \ln \sum_{j=1}^{k} w_j p_j(x_i; \theta_j) \to \max_{\Theta}.$$

Непонятно, что делать с логарифмом суммы, поэтому мы не можем найти аналитическое решение.

Идея EM алгоритма

Основная идея: добавить скрытые переменные, такие что:

1) они могут быть выражены с помощью Θ;
2) они могут помочь разделить сумму.

$$p(X, H|\Theta) = \prod_{i=1}^{k} p(X|H, \Theta)p(H|\Theta)$$

Схема EM алгоритма

EM алгоритм — повторение двух шагов:

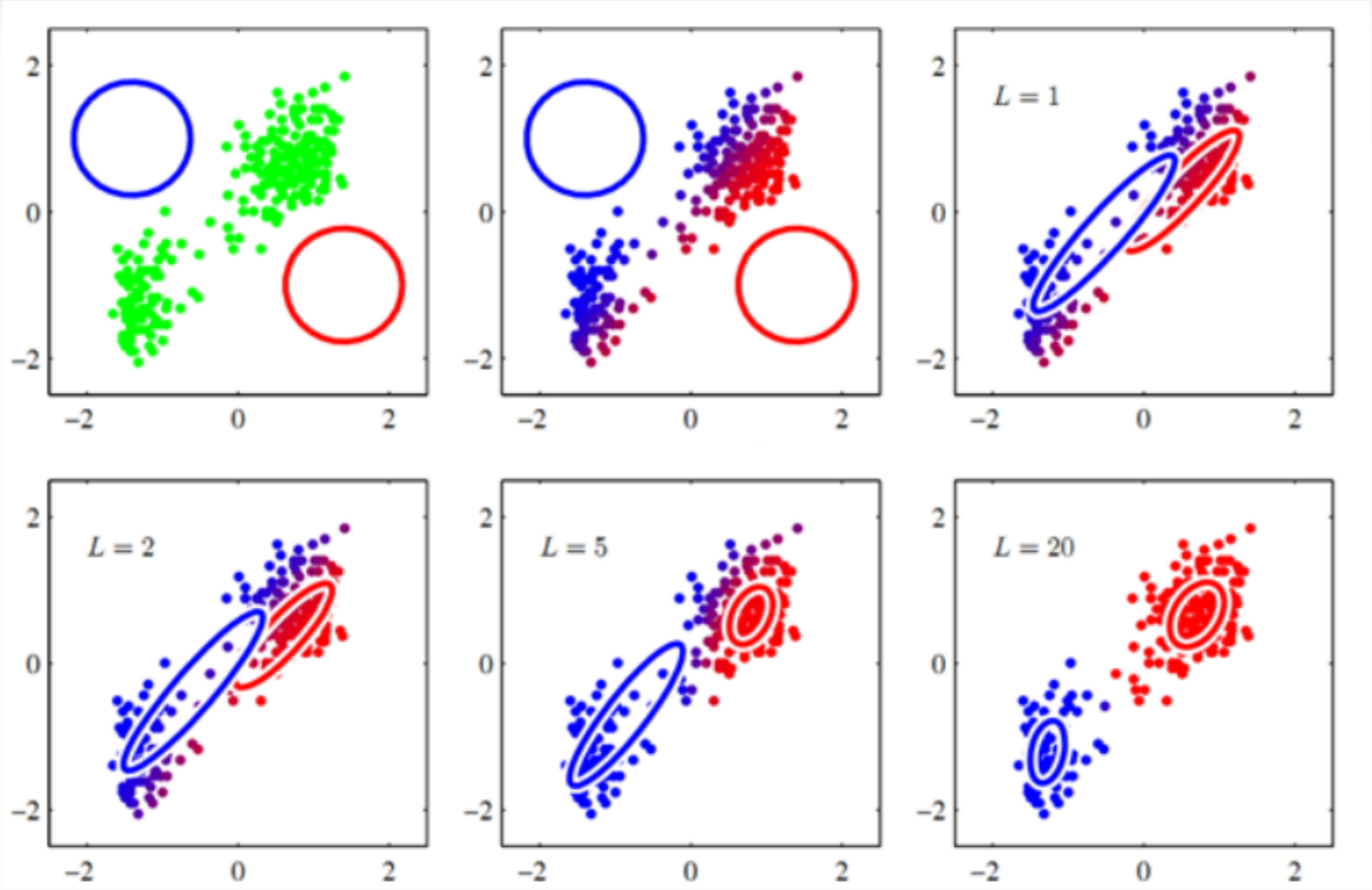
$H \leftarrow$E-STEP(Θ) (ожидание):

поиск наиболее вероятных значений скрытых переменных

$\Theta \leftarrow$M-STEP(H, Θ) (максимизация):

поиск наиболее вероятных параметров с учетом значений скрытых переменных

Примеры (Гауссианы)



Скрытые переменные

Чем являются скрытые переменные?

Скрытые переменные

Что в этом случае скрытые переменные?

Скрытые переменные связаны с параметрами k распределений, которые мы пытаемся найти. Но удобнее работать со степенью принадлежности каждой точки каждому компоненту.

E-step

$$p(x_i, \theta_j) = p(x)\Pr(\theta_j|x) = w_j p_j(x)$$

Скрытые переменные $H = (h_{ij})_{m\times k}$,
где $h_{ij} = \Pr(\theta_j|x_i)$ — степени вероятности того, что x_i принадлежит j-му компоненту:

$$h_{ij} = \frac{w_j p_j(x_i)}{p(x_i)} = \frac{w_j p_j(x_i)}{\sum_{s=1}^{k} w_s p_s(x_i)},$$

$$\sum_{j=1}^{k} h_{ij} = 1.$$

M-step

Теорема

Если скрытые переменные известны, то задачу минимизации $\mathcal{L}(\Theta)$ можно свести к k независимым подзадачам

$$\theta_j = \operatorname{argmax}_\theta \sum_{i=1}^{m} h_{ij} \ln \varphi(x_i, \theta),$$

и оптимальные веса равны

$$w_j = \frac{1}{m} \sum_{i=1}^{m} h_{ij}.$$

Будем максимизировать θ_j.

Максимизация ожидания (EM)

Input: $\mathcal{D}_s, k, \Theta^{(0)}$

1. Repeat

2. **E-step**: for all $i = 1, \dots, m; j = 1, \dots, k$

$$h_{ij} = \frac{w_j \varphi(x_i;\theta_j)}{\sum_{s=1}^{k} w_s \varphi(x_i;\theta_j)};$$

3. **M-step**: for all $j = 1, \dots, k$

$$\theta_j = \operatorname{argmax}_\theta \sum_{i=1}^{m} h_{ij} \ln \varphi(x_i, \theta); w_j = \frac{1}{m} \sum_{i=1}^{m} h_{ij};$$

4. Until a **stopping criterion** is satisfied

Return $\Theta = (\theta_j, w_j)_{j=1}^{k}$.

Анализ алгоритма

Преимущества:

- Сходится во многих ситуациях
- Легко превращается в нечувствительность к шуму
- Самый гибкий подход

Questions:

1. Когда останавливаться?
2. Как ускорить сходимость?
3. Как выбрать начальную аппроксимацию?
4. Как выбрать k?

Некоторые ответы (1/2)

1. Когда останавливаться?

Пока результат не стабилизируется.
Рекомендуется делать это относительно h:

$$\max_{i,j} \left| h_{ij} - h_{ij}^{(0)} \right| > \delta_1$$

$$\max_{i} \sum_{j} |h_{ij}^{(t)} - h_{ij}^{(t-1)}| > \delta_2$$

…

2. Как ускорить сходимость?

Ускорить M-step.

Некоторые ответы (2/2)

3. Как выбрать начальную аппроксимацию?

- Равномерно.
- Выбор из отдаленных точечных районов.

…

4. Как выбрать k?

- Итеративно проверять для каждого k.
- Проверять некоторые значения k и восстанавливать график.

Улучшения EM

- **Изменение количества компонентов**

попробовать добавить или удалить компоненты

- **Обобщенный EM-алгоритм(GEM)**

не пытаться найти хорошее решение M-step

- **Стохастический EM-алгоритм(SEM)**

попытаться найти максимум невзвешенного правдоподобия на М-шаге

- **Иерархический EM-алгоритм (HEM)**

Попробовать разделить «плохие» компоненты

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

Постановка задачи

Задача: разделить набор объектов одного типа на группы так, чтобы объекты в этих группах имели похожие свойства.

“Похожесть” формализуется с абстрактной мерой.

$\mathcal{D}$ — набор данных, состоящий из объектов из X

$\rho: X \times X \to [0; +\infty)$ — метрика на X.

Найти алгоритм $a: X \to Y$, где Y — множество кластеров.

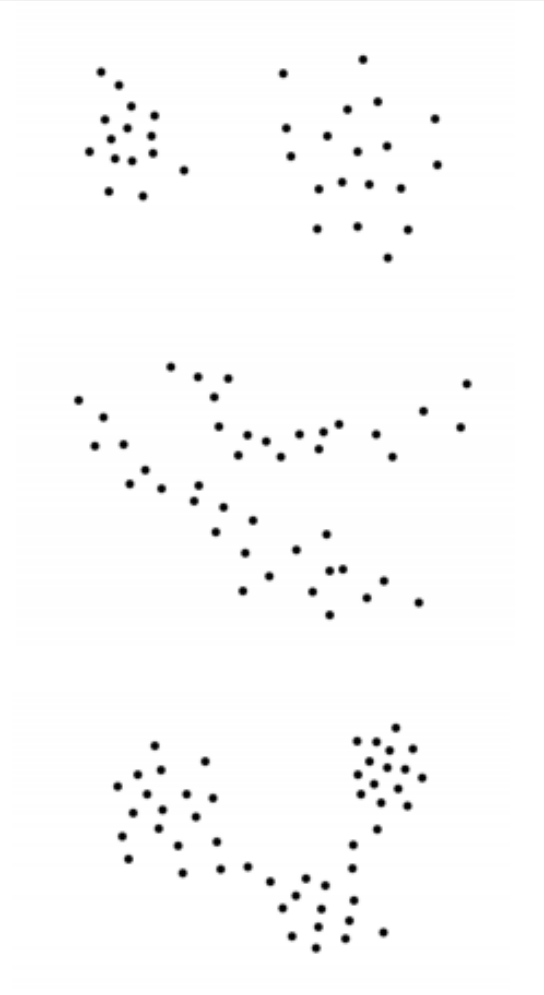
Некорректность постановки задачи

- Нет правильной постановки задачи
- Нет универсального критерия качества
- Нет универсальной меры расстояния между объектами (следствие теоремы Клейнберга)
- Количество кластеров обычно неизвестно

Цели кластеризации

- Уменьшить объем данных
- Найти группы похожих объектов
- Найти необычные объекты
- Найти иерархию объектов (групп)

Примеры кластеров (1/2)



Явно разделимые

Полосы

С «мостами»

Примеры кластеров (2/2)

С шумами

Смесь распределений

Нет кластеров

Приложения

- Биология и медицина
  - Анализ последовательностей
  - Медицинская «визуализация» (КТ снимки)
- Социальные науки
  - Анализ криминала
- Информационные технологии
  - Сегментация изображения
- Маркетинг
  - Целевые группы
- Анализ текста
- Социальные сети

Меры оценки качества

- **Внешние меры** основаны на данных, которые не использовались для кластеризации, таких как известные метки классов и внешние тесты, бенчмарки.
- **Внутренние меры** не используют какой-либо внешней информации и основываются на структуре раздела.

Меры качества в метрическом пространстве (1/2)

Среднее внутрикластерное расстояние:

$$F_0 = \frac{\sum_{i<j}[y_i = y_j]\rho(x_i, x_j)}{\sum_{i<j}[y_i = y_j]} \to \min.$$

Среднее межкластерное расстояние:

$$F_1 = \frac{\sum_{i<j}[y_i \neq y_j]\rho(x_i, x_j)}{\sum_{i<j}[y_i \neq y_j]} \to \max.$$

Отношение:

$$F_0/F_1 \to \min.$$

Меры качества в метрическом пространстве (2/2)

Среднее внутрикластерное расстояние :

$$\Phi_0 = \sum_{y \in Y} \frac{1}{|C_y|} \sum_{i: x_i \in C_y} \rho^2(x_i, c_y) \to \min.$$

Сумма межкластерных расстояний:

$$\Phi_1 = \sum_{y \in Y} \rho^2(c_y, c) \to \max.$$

Отношение:

$$\Phi_0 / \Phi_1 \to \min.$$

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

EM

Работают также как и оригинальный EM

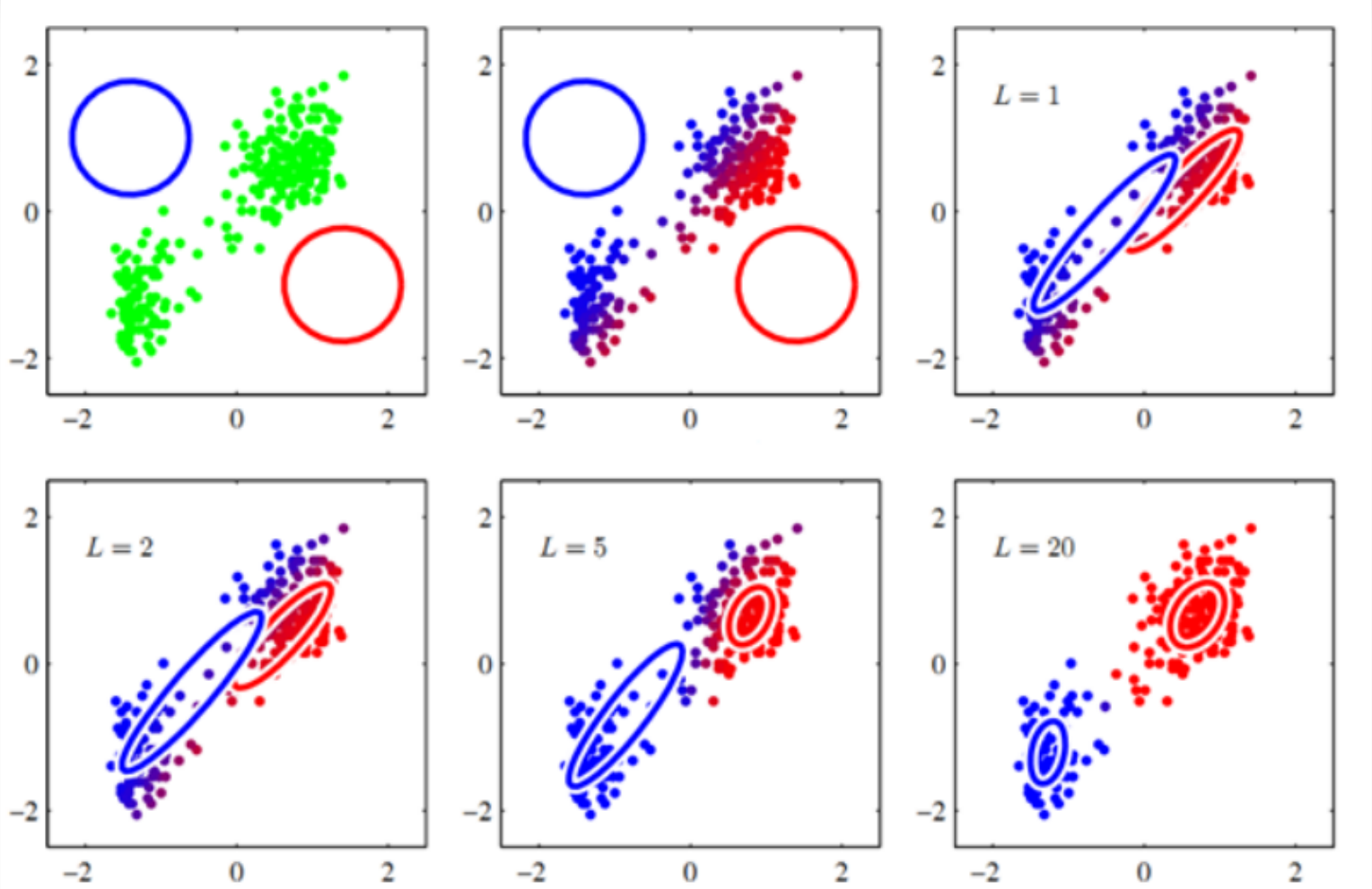
Предположение: выборка простая.

w_y — априорная вероятность принадлежности кластеру y.

Аппроксимировать Гауссинами.

Каждый кластер описывается d-размерной Гауссовской функцией плотности с диагональной ковариационной матрицей.

EM пример



Идея k-средних

- ***k*-средних** представляет собой итерационный алгоритм, который разбивает наборы на k частей.
- Центр масс кластера (среднее внутрикластерное расстояние по каждому признаку) C_j называется *центроидом*

$$c_j = \frac{1}{|C_j|} \sum_{i \in C_j} x_i \ \in \ C_j$$

Алгоритм k-средних

Это упрощение EM-алгоритма с сильной ассоциацией только с одним классом.

1. Выбрать k точек (**центроидов**) $\{c_i\}_{i=1}^{k}$ из набора данных.
2. Повторять
3. Для каждого x найти ближайший центроид $n(x)$.

$$C_i = \{x | n(x) = c_i\}$$

4. Для каждого C_i найти центральную точку и определить её центроидом.
5. Пока центроиды не будут изменяться.

Модифицированный алгоритм k-means++

Как предотвратить произвольно плохие локальные минимумы в k-средних?

Делать то же самое, что и в k-средних, но выбирать новый центр i-го кластера с вероятностью, пропорциональной $||p - c_i||^2$

c-средних (нечёткая кластеризация)

Неточная степень принадлежности кластера $u_i(x)$ объекта x кластеру C_i, при этом $\sum_i u_i(x) = 1$.

Центр кластера:

$$c_i = \frac{\sum_{x \in X^m} u_i{}^d(x)x}{\sum_{x \in X^m} u_i{}^d(x)}.$$

Переопределить степень принадлежности:

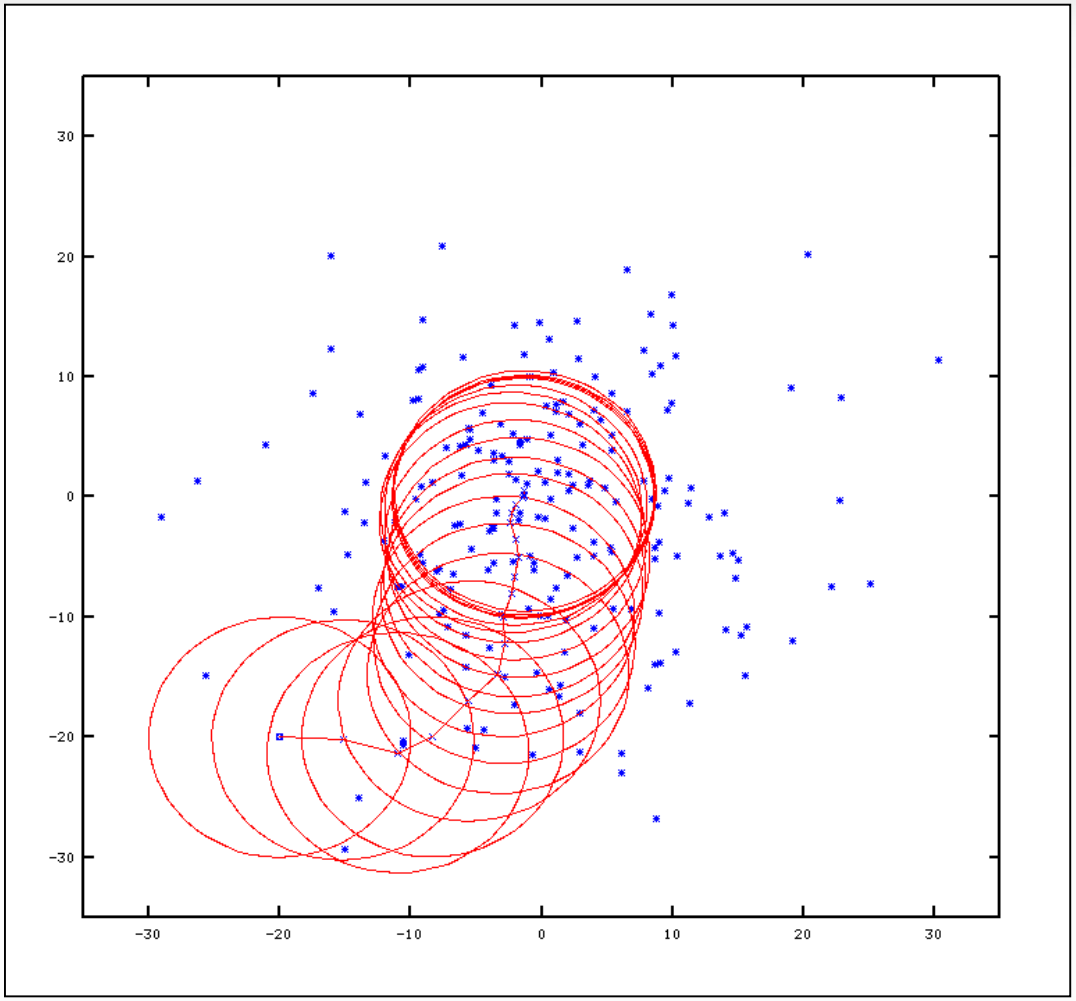
$$u_i(x) = \frac{1}{\sum_j \left(\frac{\rho(c_i, x)}{\rho(c_j, x)}\right)^{2/(d-1)}}.$$

Метод среднего сдвига (Mean-shift)

- Установить шар вокруг каждой точки
- Найти центроид каждой сферы
- Переместить центр сферы к центроиду

После каждой итерации центроиды перемещаются в более «плотные» сферы до тех пор, пока не сойдутся в **модах плотности**.

Метод среднего сдвига



Градиентный подъем для мод плотности

Алгоритм использует градиентный подъём:

$$\nabla \hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} \frac{\partial}{\partial x} K\left(\frac{x - x_i}{h}\right)$$

$$\nabla \hat{f}(x) = 0$$

Гауссово ядро

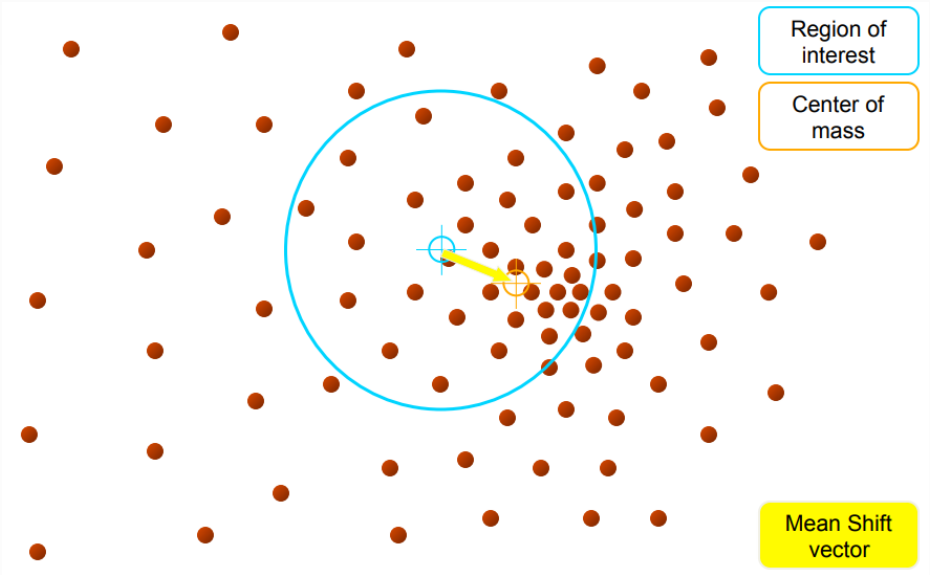
$$\frac{\partial}{\partial x} K\left(\frac{x - x_i}{h}\right) = K\left(\frac{x - x_i}{h}\right) \frac{x - x_i}{h} \frac{1}{h}$$

$$\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) x = \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) x_i$$

“Возрастающее” направление

Вектор возрастающей функции ядра

$$m(x) = \frac{\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) x_i}{\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)}$$



Средний сдвиг

$$m(x) - x = \frac{\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) x_i}{\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)} - x$$

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

Подход на основе графов

Основная идея: будем работать с графом, его вершины являются объектами, а его длины ребер равны расстояниям между соответствующими объектами.

Кластеры могут быть хорошо представлены в графическом описании.

Выбор связных компонент

Зафиксируем радиус R.

Удалим рёбра $\{x, y\}$: $\rho(x, y) > R$.

Кластеры соответствуют связанным компонентам.

Зафиксируем K_1, K_2.

Будем изменять R, пока число кластеров в интервале $[K_1, K_2]$.

Кратчайший путь

Зафиксируем число кластеров K.

Будем искать минимальное остовное дерево (Kruskal, Boruvka, MST).

Удалим $K - 1$ рёбер с максимальными длинами.

FOREL

Input: $U = X^m$, a set of unclusterized points.

1. Repeat
2. Choose a random point x from U
3. Repeat
4. $B \leftarrow$ sphere with radius R and center x
5. $c \leftarrow$ mass center of B
6. Until the sphere does not change
7. $U \leftarrow U \backslash B$
8. Until $U \neq \emptyset$

Return set of clusters

FOREL свойства

Зависит от R

Как выбирать центр масс?

- Центр масс в векторном пространстве
- Такой объект, что сумма расстояний от него до всех остальных объектов минимальна.
- Объект, который в сфере радиуса R содержит максимальное количество объектов из выборки
- Объект, который в сфере радиуса r содержит максимальное количество объектов из сферы радиуса R

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

Плотностный подход

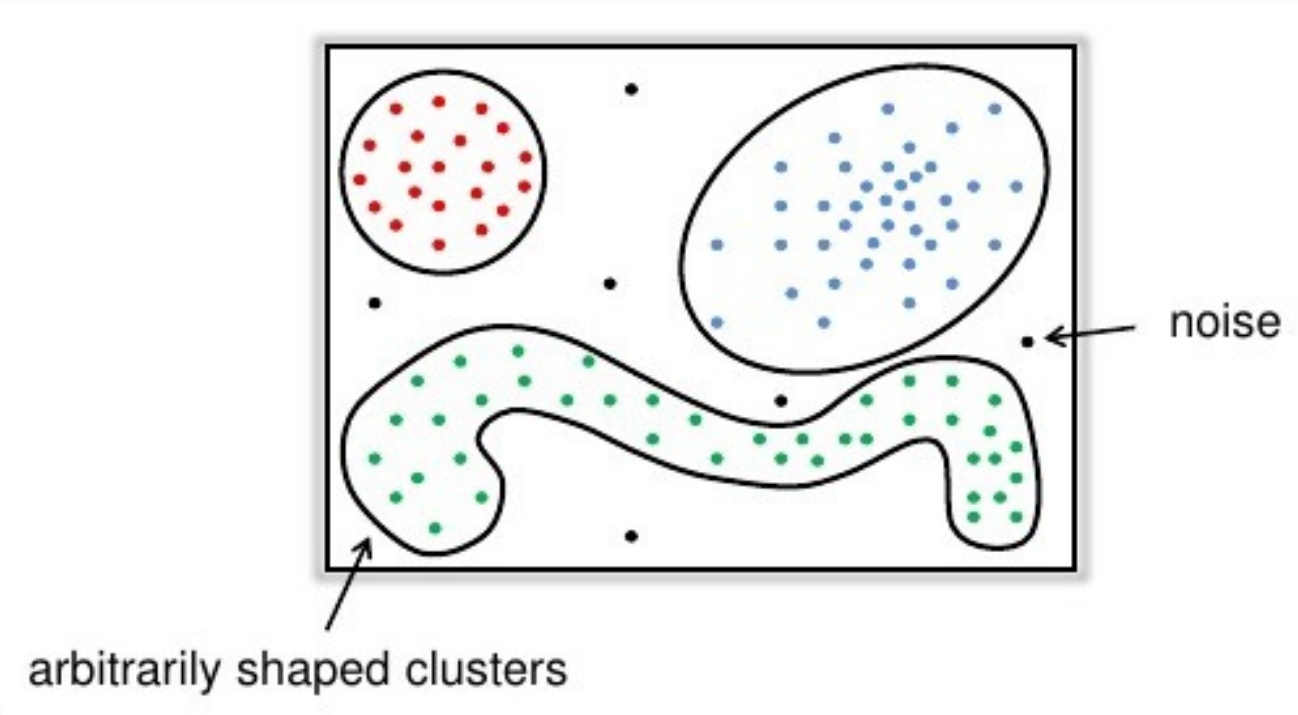
Идея: каждая точка p кластера содержит более M точек в радиусе ε :

$N_\varepsilon(p) -$ множество точек вокруг p в радиусе ε. $|N_\varepsilon(p)| \geq M$.

Проблема с граничными точками.

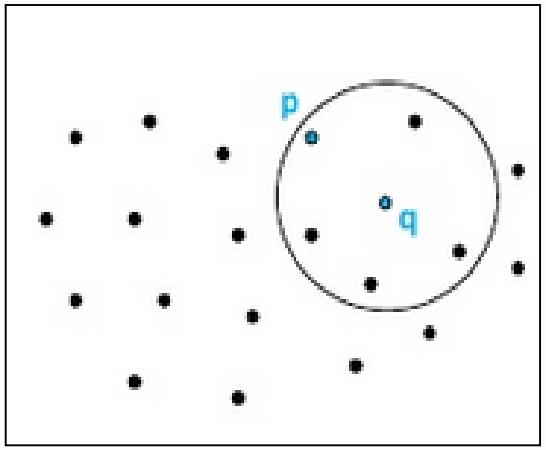
DBSCAN

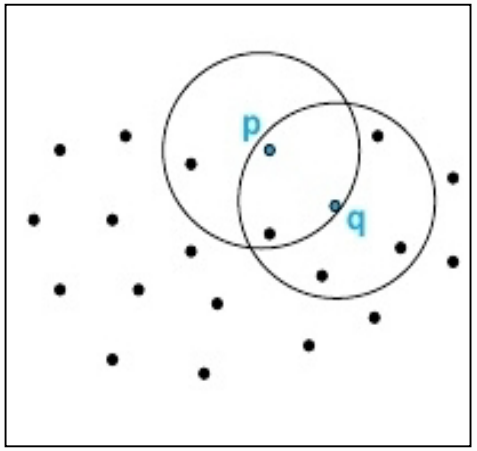
DBSCAN (Density Based Spatial Clustering of Applications with Noise)



Достижимая точка

p является **непосредственно достижимой** из q (с заданными ε и M),
если $p \in N_\varepsilon(q)$ и $|N_\varepsilon(q)| \geq M$.

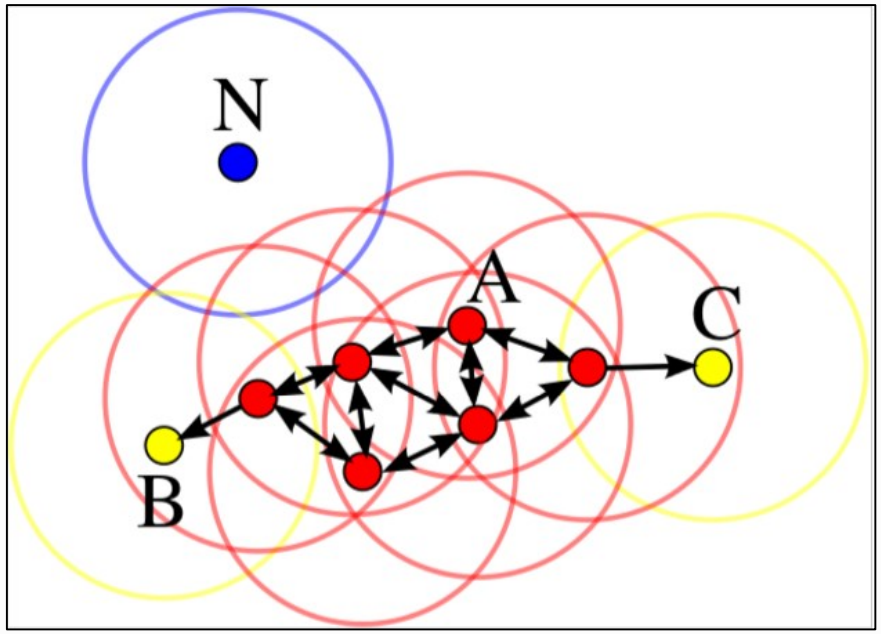




p **достижима** из q (с заданными ε и M), если $\exists \{a_i\}, a_i$ непосредственно достижима из a_{i-1}.

Связные точки

B **связна** с C (с заданными ε и M), если $\exists A$ такая, что B и C достижимы из A (с заданными ε и M).



Определение кластера в терминах DBSCAN

Кластер C_j (с заданными ε и M) является непустым множеством точек:

- $\forall p, q : \ p \in C_j, q$ – достижима из $p \Rightarrow \ q \in C_j$
- $\forall p, q \in C_j$: p связна с q.

DBSCAN алгоритм

Input: $\mathcal{D}, \varepsilon, M$.

foreach $d_i \in \mathcal{D}$: $V[d_i] = \text{false}$, $j = 0$, $Noise = \emptyset$

for all $d_i \in \mathcal{D}$:

    if $V[d_i]$ is false **then**

        $V[d_i] = \text{true}$, $N_i = N_\varepsilon(d_i)$

        if $|N_i| < M$ **then**

            $Noise = Noise + \{d_i\}$

        else

            $j = j + 1$, EXPAND$(d_i, N_i, C_j, \varepsilon, M)$

return $C = \{C_j\}$

Функция EXPAND

Input: $d_i, N_i, C_j, \varepsilon, M$.

$C_j = C_j + \{d_i\}$

for all $d_k \in N_i$:

  if $V[d_k]$ is false **then**

    $V[d_k] = \text{true}, N_{ik} = N_\varepsilon(d_k)$

    if $|N_{ik}| \geq M$ **then**

      $N_i = N_i + N_{ik}$

  if $\nexists p : d_k \in C_p$ **then**

    $C_j = C_j + \{d_k\}$

return $C = \{C_j\}$

План лекции

- EM алгоритм
- Задача кластеризации
- EM-подобные алгоритмы кластеризации
- Графовые алгоритмы
- Плотностные алгоритмы
- Иерархические алгоритмы

Иерархический подход

Идея: строить иерархию кластеров.

Будем строить **дендрограмы**. Количество кластеров рассматривается с точки зрения высоты дерева.

Два подхода:

Разделяющий («дробить» кластеры)

Агломеративный (объединять кластеры)

Алгоритм Lance-Williams

1. 1-element clusters:

   $t = 1, C_t = \{x_1, \dots, x_l\}$ a set of clusters on iteration $t = 1$;

   $R(\{x_i\}, \{x_j\}) = \rho(x_i, x_j)$ relationship between clusters;

2. For all $t = 2 \dots l$:

3. In C_{t-1} find 2 *closest* (most related) clusters:

   $(U, V) = \text{argmin}_{U \neq V} R(U, V)$;

4. Merge them into a single cluster:

   $W = U \cup V$;

   $C_t = C_{t-1} \cup \{W\} \setminus \{U, V\}$;

5. For all $S \in C_t$ count $R(W, S)$.

Расстояние Lance-Williams

Расстояние $R(W, S)$ между кластерами $W = U \cup V$ и S

Расстояние Lance-Williams:

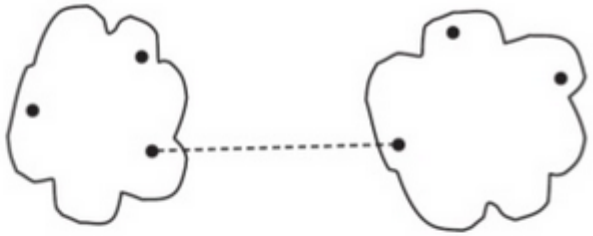
$$R(U \cup V, S) = \alpha_U R(U, S) + \\ + \alpha_V R(V, S) + \\ + \beta R(U, V) + \\ + \gamma |R(U, S) - R(V, S)|$$

Варианты $R(W,S)$ (1/2)

1. Nearest neighbor distance

$$R^N(W,S) = \min_{w\in W, s\in S} \rho(w,s);$$

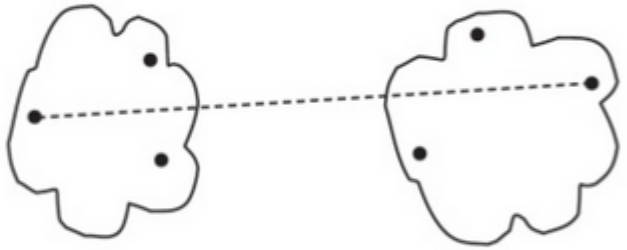
$$\alpha_U = \alpha_V = \frac{1}{2},\ \beta = 0,\ \gamma = -\frac{1}{2}.$$



2. Most distant neighbor distance

$$R^D(W,S) = \max_{w\in W, s\in S} \rho(w,s);$$

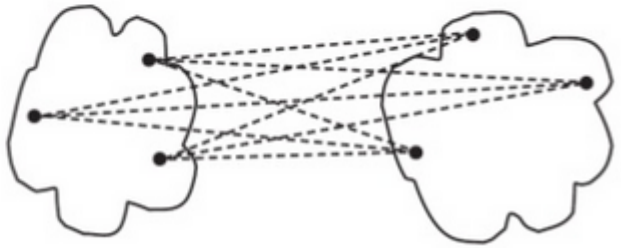
$$\alpha_U = \alpha_V = \frac{1}{2},\ \beta = 0,\ \gamma = \frac{1}{2}.$$



3. Mean group distance

$$R^G(W,S) = \frac{1}{|W||S|} \sum_{w\in W} \sum_{s\in S} \rho(w,s);$$

$$\alpha_U = \frac{|U|}{|W|},\ \alpha_V = \frac{|V|}{|W|},\ \beta = \gamma = 0.$$



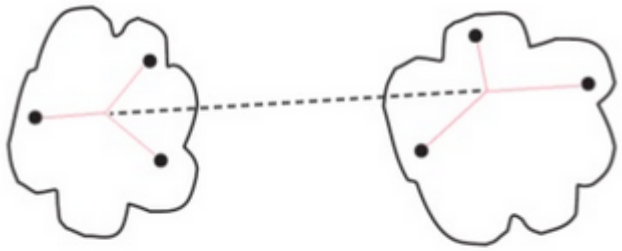
Варианты $R(W,S)$(2/2)

4. Distance between centres

$$R^c(W,S) = \rho^2\Big(\sum_{w\in W}\frac{w}{|W|}, \sum_{s\in S}\frac{s}{|S|}\Big);$$

$$\alpha_U = \frac{|U|}{|W|}, \quad \alpha_V = \frac{|V|}{|W|},$$

$$\beta = -\alpha_U\alpha_V, \quad \gamma = 0.$$



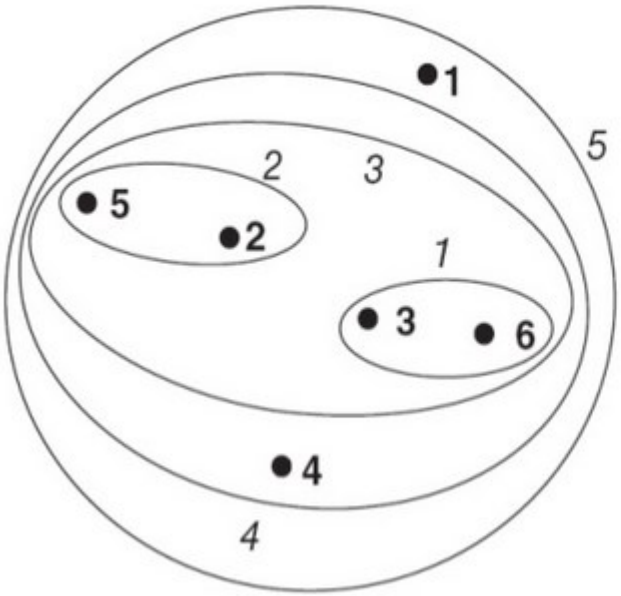
5. Ward's distance

$$R^w(W,S) = \frac{|S||W|}{|S|+|W|}\,\rho^2\Big(\sum_{w\in W}\frac{w}{|W|}, \sum_{s\in S}\frac{s}{|S|}\Big);$$

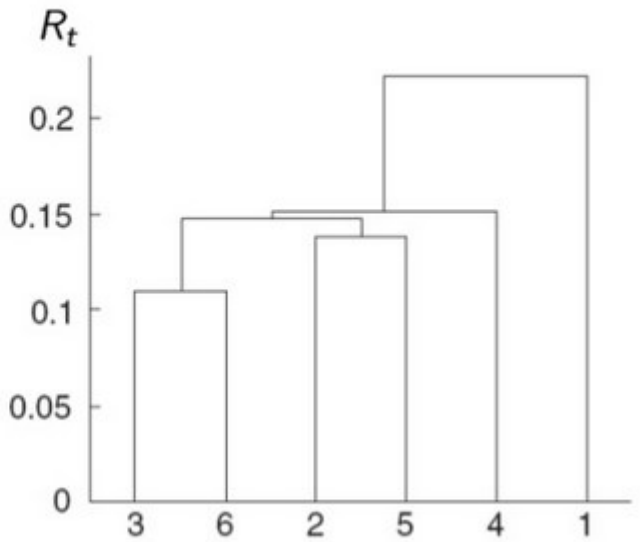
$$\alpha_U = \frac{|S|+|U|}{|S|+|W|}, \quad \alpha_V = \frac{|S|+|V|}{|S|+|W|}, \quad \beta = \frac{-|S|}{|S|+|W|}, \quad \gamma = 0.$$

Визуализация ближайших соседей

Сюжет включения

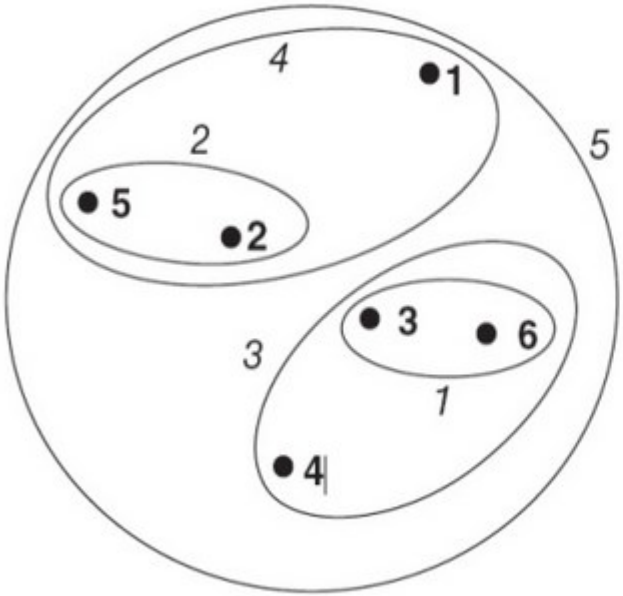


Дендрограма

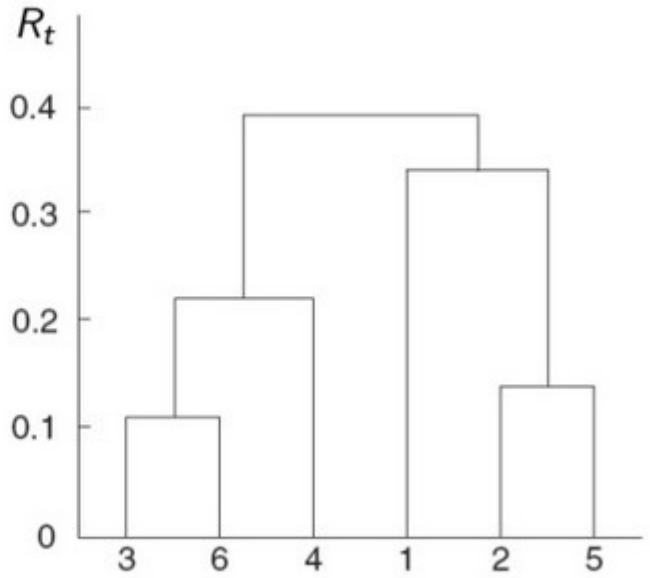


Визуализация самого дальнего соседа

Сюжет включения

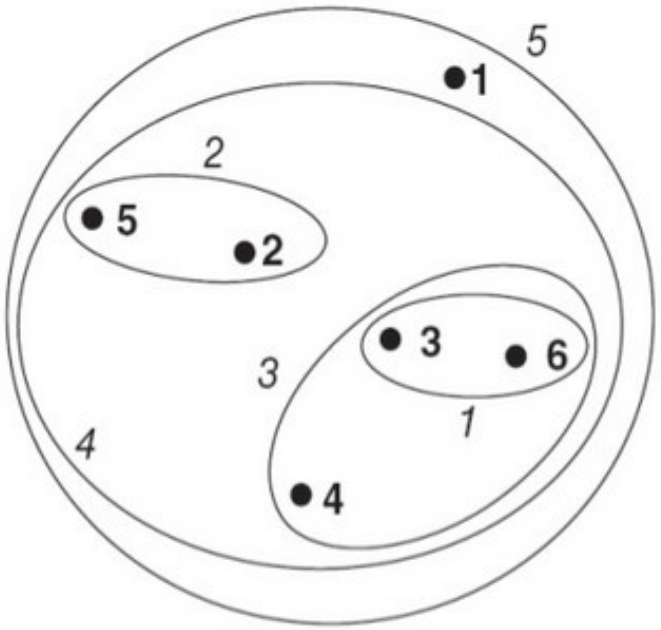


Дендрограма

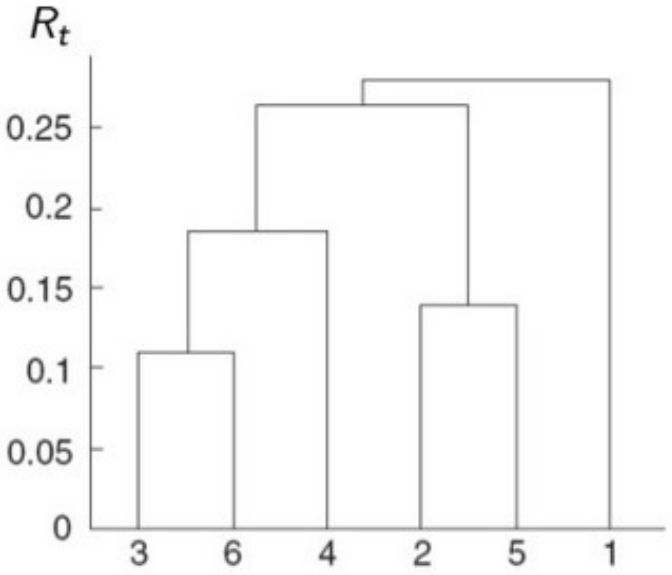


Групповая средняя визуализация

Сюжет включения

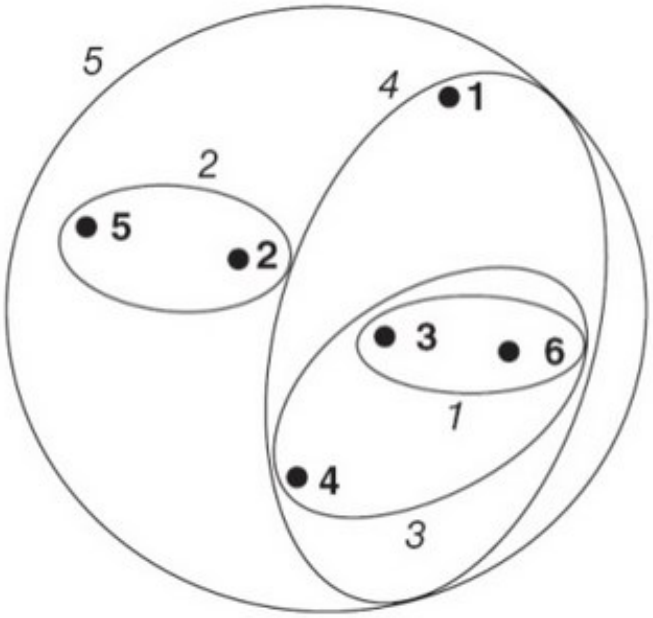


Дендрограма

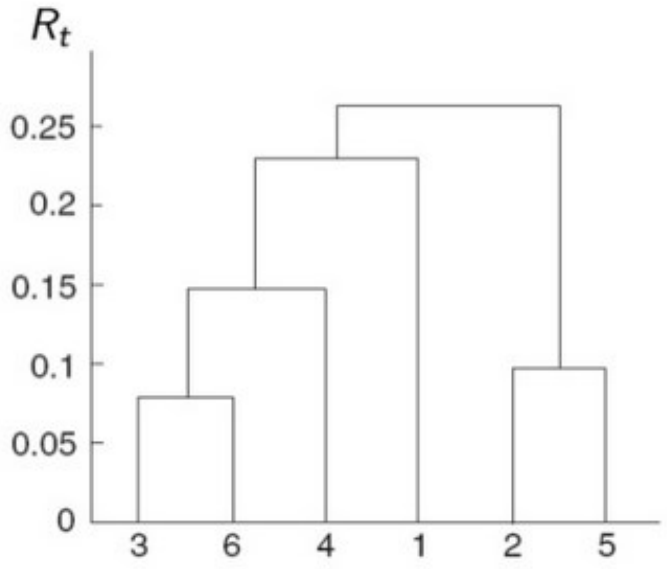


Визуализация расстояния Ward

Сюжет включения



Дендрограма



Монотонная кластеризация

Кластеризация называется **монотонной,** если межкластерное расстояние не уменьшается после объединения.

Теорема (Milligan, 1979)

Кластеризация является монотонной, если

$\alpha_U \geq 0, \alpha_V \geq 0, \alpha_U + \alpha_V + \beta \geq 1, \min\{\alpha_U, \alpha_V\} + \gamma \geq 0.$

Если кластеризация монотонная, дендрограма не имеет самопересечений.

Монотонная кластеризация

Кластеризация называется **монотонной,** если межкластерное расстояние не уменьшается после объединения.

Теорема (Milligan, 1979)

Кластеризация является монотонной, если

$\alpha_U \geq 0, \alpha_V \geq 0, \alpha_U + \alpha_V + \beta \geq 1, \min\{\alpha_U, \alpha_V\} + \gamma \geq 0.$

Если кластеризация монотонная, дендрограма не имеет самопересечений.

R^C не монотонная.

Общие рекомендации

- Расстояние Ward наиболее предпочтительно;
- Ускорение алгоритмов: объединение локально близких кластеров.
- Выбор числа кластеров исходя из минимизации $|R_{t+1} - R_t|$.