

Методы кластеризации

К. В. Воронцов
vokov@forecsys.ru

Этот курс доступен на странице вики-ресурса
<http://www.MachineLearning.ru/wiki>
«Машинное обучение (курс лекций, К.В.Воронцов)»

28 апреля 2010

Постановка задачи кластеризации

Дано:

X — пространство объектов;

$X^\ell = \{x_i\}_{i=1}^\ell$ — обучающая выборка;

$\rho: X \times X \rightarrow [0, \infty)$ — функция расстояния между объектами.

Найти:

Y — множество кластеров и

$a: X \rightarrow Y$ — алгоритм кластеризации, такие, что:

— каждый кластер состоит из близких объектов;

— объекты разных кластеров существенно различны.

Кластеризация — это *обучение без учителя*.

Некорректность задачи кластеризации

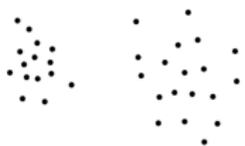
Решение задачи кластеризации принципиально неоднозначно:

- точной постановки задачи кластеризации нет;
- существует много критериев качества кластеризации;
- существует много эвристических методов кластеризации;
- число кластеров $|Y|$, как правило, неизвестно заранее;
- результат кластеризации существенно зависит от метрики ρ , которую эксперт задаёт субъективно.

Цели кластеризации

- Упростить дальнейшую обработку данных, разбить множество X^ℓ на группы схожих объектов чтобы работать с каждой группой в отдельности (задачи классификации, регрессии, прогнозирования).
- Сократить объём хранимых данных, оставив по одному представителю от каждого кластера (задачи сжатия данных).
- Выделить нетипичные объекты, которые не подходят ни к одному из кластеров (задачи одноклассовой классификации).
- Построить иерархию множества объектов (задачи таксономии).

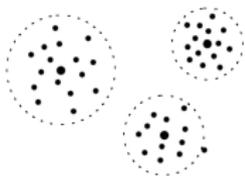
Типы кластерных структур



внутрикластерные расстояния, как правило,
меньше межкластерных



ленточные кластеры



кластеры с центром

Типы кластерных структур



кластеры могут соединяться перемычками



кластеры могут накладываться на разреженный фон из редко расположенных объектов



кластеры могут перекрываться

Типы кластерных структур



кластеры могут образовываться не по сходству, а по иным типам регулярностей

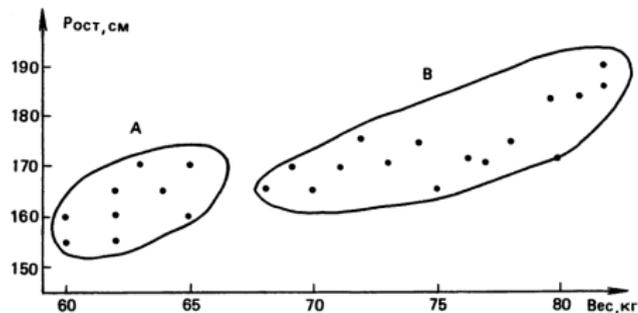


кластеры могут вообще отсутствовать

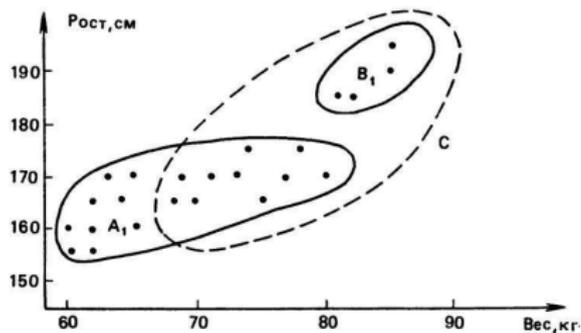
- Каждый метод кластеризации имеет свои ограничения и выделяет кластеры лишь некоторых типов.
- Понятие «тип кластерной структуры» зависит от метода и также не имеет формального определения.

Проблема чувствительности к выбору метрики

Результат зависит от нормировки признаков:



A — студентки,
B — студенты



после перенормировки
(сжали ось «вес» вдвое)

Содержание: методы кластеризации

- 1 Графовые методы кластеризации**
 - Алгоритм выделения связных компонент
 - Алгоритм ФОРЭЛ
 - Функционалы качества кластеризации
- 2 Иерархическая кластеризация (таксономия)**
 - Агломеративная иерархическая кластеризация
 - Дендрограмма и свойство монотонности
 - Свойства сжатия, растяжения и редуktivности
- 3 Статистические методы кластеризации**
 - EM-алгоритм
 - Метод k -средних
- 4 Сети Кохонена**
 - Модели конкурентного обучения
 - Карты Кохонена
 - Гибридные сети: кластеризация + регрессия

Алгоритм выделения связанных компонент

Выборка представляется в виде графа:

— вершины графа — объекты x_i ;

— рёбра — пары объектов с расстоянием $\rho_{ij} = \rho(x_i, x_j) \leq R$.

1: **повторять**

2: удалить все рёбра (i, j) , для которых $\rho_{ij} > R$;

3: $K :=$ число связанных компонент
(алгоритм Дейкстры или поиск в глубину);

4: **если** $K < K_1$ **то** уменьшить R ;

5: **если** $K > K_2$ **то** увеличить R ;

6: **пока** $K \notin [K_1, K_2]$

Недостатки:

- задаётся неудобный параметр R ;
- высокая чувствительность к шуму.

Алгоритм КНП — «Кратчайший Незамкнутый Путь»

- 1: Найти пару вершин (i, j) с наименьшим ρ_{ij} и соединить их ребром;
- 2: **пока** в выборке остаются изолированные точки
- 3: найти изолированную точку, ближайшую к некоторой неизолированной;
- 4: соединить эти две точки ребром;
- 5: удалить $K - 1$ самых длинных рёбер;

Достоинство:

- задаётся число кластеров K .

Недостаток:

- высокая чувствительность к шуму.

Алгоритм ФОРЭЛ — «ФОРмальные Элементы»

[Загоруйко, Ёлкина, 1967]

- 1: $U := X^\ell$ — множество некластеризованных точек;
- 2: **пока** в выборке есть некластеризованные точки, $U \neq \emptyset$;
- 3: взять случайную точку $x_0 \in U$;
- 4: **повторять**
- 5: образовать кластер с центром в x_0 и радиусом R :
 $K_0 := \{x_i \in U \mid \rho(x_i, x_0) \leq R\}$;
- 6: переместить центр x_0 в центр масс кластера:
$$x_0 := \frac{1}{|K_0|} \sum_{x_i \in K_0} x_i$$
;
- 7: **пока** состав кластера K_0 не стабилизируется;
- 8: пометить все точки K_0 как кластеризованные:
 $U := U \setminus K_0$;
- 9: применить алгоритм КНП к множеству центров кластеров;
- 10: каждый $x_i \in X^\ell$ приписать кластеру с ближайшим центром;

Замечание к шагу 6:

если X не является линейным векторным пространством, то

$$x_0 := \frac{1}{|K_0|} \sum_{x_i \in K_0} x_i \quad \longrightarrow \quad x_0 := \arg \min_{x \in K_0} \sum_{x' \in K_0} \rho(x, x');$$

Преимущества ФОРЭЛ:

- получаем двухуровневую структуру кластеров;
- кластеры могут быть произвольной формы;
- варьируя R , можно управлять детальностью кластеризации.

Недостаток ФОРЭЛ:

- чувствительность к R и начальному выбору точки x_0 .

Способ устранения:

сгенерировать несколько кластеризаций и выбрать лучшую по заданному *функционалу качества*.

Функционалы качества кластеризации

Случай 1: X — метрическое (не линейное векторное) пространство

- Среднее внутрикластерное расстояние:

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

- Среднее межкластерное расстояние:

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

- Отношение пары функционалов:

$$F_0 / F_1 \rightarrow \min .$$

Функционалы качества кластеризации

Случай 2: X — линейное векторное пространство

- Сумма средних внутрикластерных расстояний:

$$\Phi_0 = \sum_{y \in Y} \frac{1}{|K_y|} \sum_{i: y_i=y} \rho^2(x_i, \mu_y) \rightarrow \min,$$

$K_y = \{x_i \in X^\ell \mid y_i = y\}$ — кластер y ,
 μ_y — центр масс кластера y .

- Сумма межкластерных расстояний:

$$\Phi_1 = \sum_{y \in Y} \rho^2(\mu_y, \mu) \rightarrow \max,$$

где μ — центр масс всей выборки.

- Отношение пары функционалов:

$$\Phi_0 / \Phi_1 \rightarrow \min.$$

Агломеративная иерархическая кластеризация

Алгоритм Ланса-Уильямса [1967]

1: сначала все кластеры одноэлементные:

$$t := 1; \quad C_t = \{\{x_1\}, \dots, \{x_\ell\}\};$$

$$R(\{x_i\}, \{x_j\}) := \rho(x_i, x_j);$$

2: **для всех** $t = 2, \dots, \ell$ (t — номер итерации):

3: найти в C_{t-1} два ближайших кластера:

$$(U, V) := \arg \min_{U \neq V} R(U, V);$$

$$R_t := R(U, V);$$

4: слить их в один кластер:

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

5: **для всех** $S \in C_t$

6: вычислить $R(W, S)$ по формуле Ланса-Уильямса;

Формула Ланса-Уильямса

Как определить расстояние $R(W, S)$
между кластерами $W = U \cup V$ и S ,
зная расстояния $R(U, S)$, $R(V, S)$, $R(U, V)$?

Формула, обобщающая большинство разумных способов
определить это расстояние [Ланс, Уильямс, 1967]:

$$\begin{aligned}R(U \cup V, S) = & \alpha_U \cdot R(U, S) + \\ & + \alpha_V \cdot R(V, S) + \\ & + \beta \cdot R(U, V) + \\ & + \gamma \cdot |R(U, S) - R(V, S)|,\end{aligned}$$

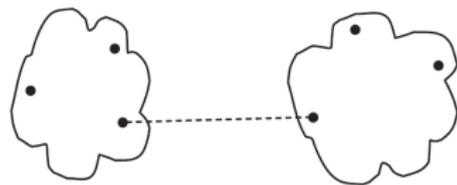
где α_U , α_V , β , γ — числовые параметры.

Частные случаи формулы Ланса-Уильямса

1. Расстояние ближнего соседа:

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

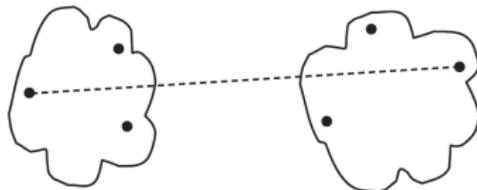
$$\alpha_U = \alpha_V = \frac{1}{2}, \quad \beta = 0, \quad \gamma = -\frac{1}{2}.$$



2. Расстояние дальнего соседа:

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

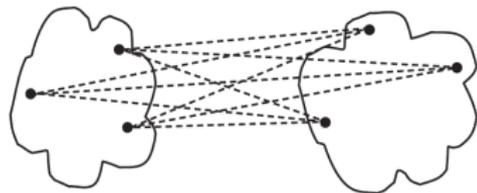
$$\alpha_U = \alpha_V = \frac{1}{2}, \quad \beta = 0, \quad \gamma = \frac{1}{2}.$$



3. Групповое среднее расстояние:

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

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



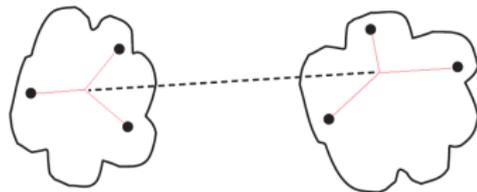
Частные случаи формулы Ланса-Уильямса

4. Расстояние между центрами:

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

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

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



5. Расстояние Уорда:

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

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

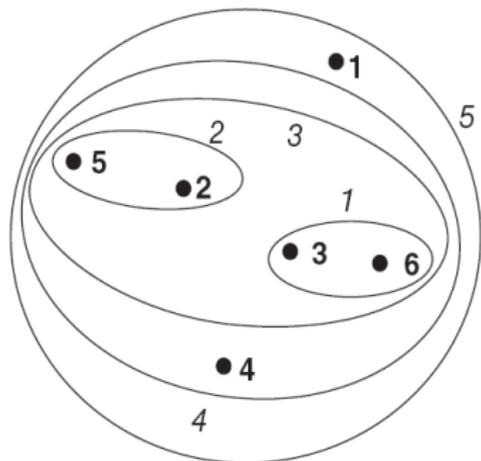
Проблема выбора

Какой тип расстояния лучше?

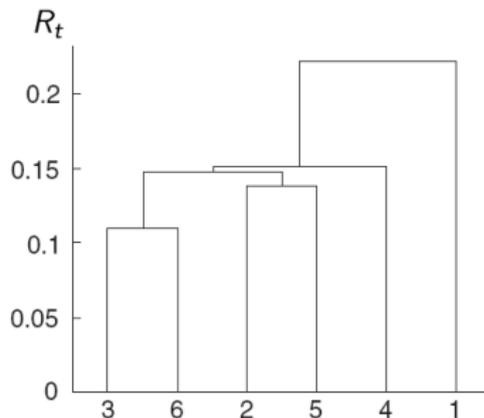
Визуализация кластерной структуры

1. Расстояние ближнего соседа:

Диаграмма вложения



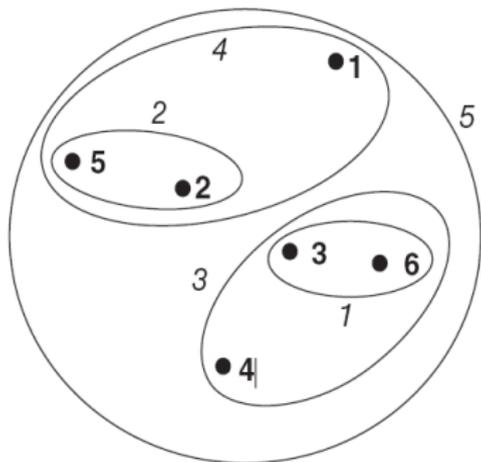
Дендрограмма



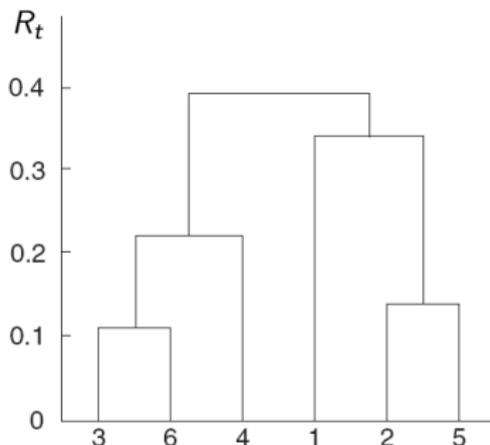
Визуализация кластерной структуры

2. Расстояние дальнего соседа:

Диаграмма вложения



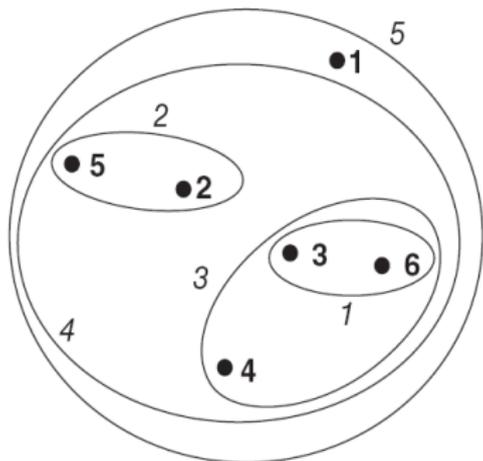
Дендрограмма



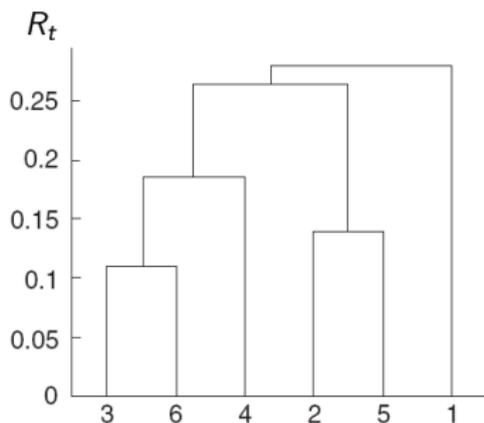
Визуализация кластерной структуры

3. Групповое среднее расстояние:

Диаграмма вложения



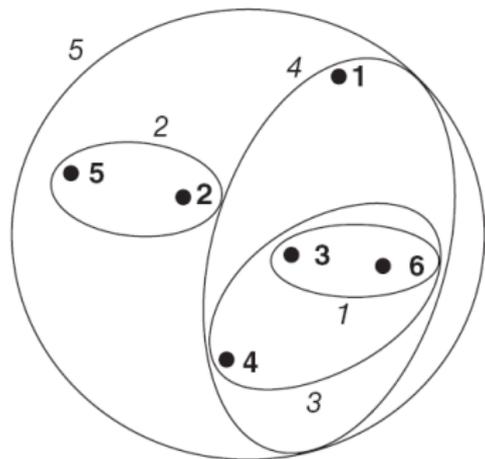
Дендрограмма



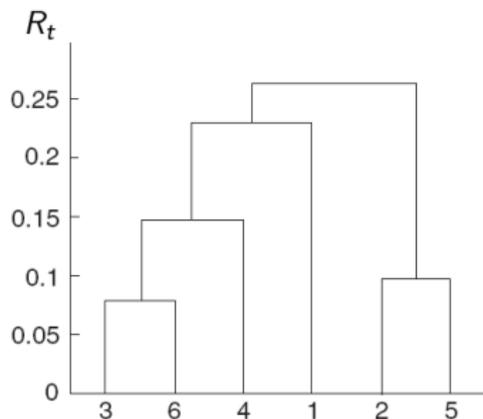
Визуализация кластерной структуры

5. Расстояние Уорда:

Диаграмма вложения



Дендрограмма



Свойство монотонности

Определение

Кластеризация *монотонна*, если при каждом слиянии расстояние между объединяемыми кластерами только увеличивается: $R_2 \leq R_3 \leq \dots \leq R_\ell$.

Теорема (Миллиган, 1979)

Кластеризация монотонна, если выполняются условия

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

Если кластеризация монотонна, то дендрограмма не имеет самопересечений.

R^C не монотонно; R^b , R^d , R^g , R^y — монотонны.

Свойства сжатия и растяжения

Определение

Кластеризация *сжимающая*, если $R_t \leq \rho(\mu_U, \mu_V)$, $\forall t$.

Кластеризация *растягивающая*, если $R_t \geq \rho(\mu_U, \mu_V)$, $\forall t$.

Иначе кластеризация *сохраняет метрику пространства*.

Свойство **растяжения** наиболее желательно, так как оно способствует более чёткому отделению кластеров.

R^b — сильно сжимающее;

R^d , R^y — растягивающие;

R^r , R^c — сохраняют метрику пространства.

Проблема повышения эффективности алгоритма

Проблема эффективности:

- самая трудоёмкая операция в алгоритме Ланса-Уильямса — поиск ближайших кластеров — $O(\ell^2)$ операций:

$$\text{шаг 3: } (U, V) := \arg \min_{U \neq V} R(U, V).$$

- значит, построение всего дерева — $O(\ell^3)$ операций.

Идея повышения эффективности:

- перебирать лишь наиболее близкие пары:

$$\text{шаг 3: } (U, V) := \arg \min_{R(U, V) \leq \delta} R(U, V).$$

- периодически увеличивать параметр δ .

Быстрый (редуктивный) алгоритм Ланса-Уильямса

- 1: сначала все кластеры одноэлементные:
 $t := 1; C_t = \{\{x_1\}, \dots, \{x_\ell\}\};$
 $R(\{x_i\}, \{x_j\}) := \rho(x_i, x_j);$
- 2: выбрать начальное значение параметра δ ;
 $P(\delta) := \{(U, V) \mid U, V \in C_t, R(U, V) \leq \delta\};$
- 3: для всех $t = 2, \dots, \ell$ (t — номер итерации):
- 4: если $P(\delta) = \emptyset$ то увеличить δ так, чтобы $P(\delta) \neq \emptyset$;
- 5: $(U, V) := \arg \min_{(U, V) \in P(\delta)} R(U, V);$
 $R_t := R(U, V);$
- 6: $C_t := C_{t-1} \cup \{W\} \setminus \{U, V\};$
- 7: для всех $S \in C_t$
- 8: вычислить $R(W, S)$ по формуле Ланса-Уильямса;
- 9: если $R(W, S) \leq \delta$ то $P(\delta) := P(\delta) \cup \{(W, S)\};$

Свойство редуktivности

Всегда ли быстрый алгоритм строит ту же кластеризацию?

Определение (Брюинош, 1978)

Расстояние R называется *редуктивным*, если для любого $\delta > 0$ и любых δ -близких кластеров $R(U, V) \leq \delta$ объединение δ -окрестностей U и V содержит δ -окрестность объединения $W = U \cup V$:

$$\{S: R(U \cup V, S) < \delta\} \subseteq \{S: R(S, U) < \delta\} \cup \{S: R(S, V) < \delta\}.$$

Теорема

Если расстояние R редуktivно, то быстрый алгоритм приводит к той же кластеризации, что и исходный алгоритм.

Свойство редуктивности

Теорема (Диде и Моро, 1984)

Расстояние R является редуктивным, если

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

Утверждение

Всякое редуктивное расстояние является монотонным.

R^c не редуктивное; R^b, R^d, R^r, R^y — редуктивные.

Рекомендации и выводы

Стратегия выбора параметра δ на шагах 2 и 4:

- Если $|C_t| \leq n_1$, то $P(\delta) := \{(U, V) : U, V \in C_t\}$.
- Иначе выбрать n_2 случайных расстояний $R(U, V)$;
 $\delta :=$ минимальное из них;
- n_1, n_2 влияют только на скорость, но не на результат кластеризации; сначала можно положить $n_1 = n_2 = 20$.

Общие рекомендации по иерархической кластеризации:

- лучше пользоваться R^y — расстоянием Уорда;
- лучше пользоваться быстрым алгоритмом;
- определение числа кластеров — по максимуму $|R_{t+1} - R_t|$, тогда результирующее множество кластеров $:= C_t$.

Гипотеза (о вероятностной природе данных)

Выборка X^ℓ случайна, независима, из смеси распределений

$$p(x) = \sum_{y \in Y} w_y p_y(x), \quad \sum_{y \in Y} w_y = 1,$$

$p_y(x)$ — плотность, w_y — априорная вероятность кластера y .

Гипотеза (о пространстве объектов и форме кластеров)

$X = \mathbb{R}^n$, $x_i \equiv (f_1(x_i), \dots, f_n(x_i))$; кластеры n -мерные гауссовские:

$$p_y(x) = (2\pi)^{-\frac{n}{2}} (\sigma_{y1} \cdots \sigma_{yn})^{-1} \exp\left(-\frac{1}{2} \rho_y^2(x, \mu_y)\right),$$

$\mu_y = (\mu_{y1}, \dots, \mu_{yn})$ — центр кластера y ;

$\Sigma_y = \text{diag}(\sigma_{y1}^2, \dots, \sigma_{yn}^2)$ — диагональная матрица ковариаций;

$$\rho_y^2(x, x') = \sum_{j=1}^n \sigma_{yj}^{-2} |f_j(x) - f_j(x')|^2.$$

EM-алгоритм (повторение)

- 1: начальное приближение w_y , μ_y , Σ_y для всех $y \in Y$;
- 2: **повторять**

- 3: E-шаг (expectation):

$$g_{iy} := \frac{w_y p_y(x_i)}{\sum_{z \in Y} w_z p_z(x_i)}, \quad y \in Y, \quad i = 1, \dots, \ell;$$

- 4: M-шаг (maximization):

$$w_y := \frac{1}{\ell} \sum_{i=1}^{\ell} g_{iy}, \quad y \in Y;$$

$$\mu_{yj} := \frac{1}{\ell w_y} \sum_{i=1}^{\ell} g_{iy} f_j(x_i), \quad y \in Y, \quad j = 1, \dots, n;$$

$$\sigma_{yj}^2 := \frac{1}{\ell w_y} \sum_{i=1}^{\ell} g_{iy} (f_j(x_i) - \mu_{yj})^2, \quad y \in Y, \quad j = 1, \dots, n;$$

- 5: $y_i := \arg \max_{y \in Y} g_{iy}$, $i = 1, \dots, \ell$;

- 6: **пока** y_i не перестанут изменяться;

Метод k -средних (k -means)

Упрощённый аналог EM-алгоритма:

1: начальное приближение центров μ_y , $y \in Y$;

2: **повторять**

3: аналог E-шага:

отнести каждый x_i к ближайшему центру:

$$y_i := \arg \min_{y \in Y} \rho(x_i, \mu_y), \quad i = 1, \dots, \ell;$$

4: аналог M-шага:

вычислить новые положения центров:

$$\mu_{yj} := \frac{\sum_{i=1}^{\ell} [y_i = y] f_j(x_i)}{\sum_{i=1}^{\ell} [y_i = y]}, \quad y \in Y, \quad j = 1, \dots, n;$$

5: **пока** y_i не перестанут изменяться;

Модификации и обобщения

Варианты k -means:

- вариант Болла-Холла (на предыдущем слайде);
- вариант МакКина: при каждом переходе объекта из кластера в кластер их центры пересчитываются;

Основные отличия EM и k -means:

- EM: мягкая кластеризация: $g_{iy} = P\{y_i = y\}$;
 k -m: жёсткая кластеризация: $g_{iy} = [y_i = y]$;
- EM: форма кластеров эллиптическая, настраиваемая;
 k -m: форма кластеров жёстко определяется метрикой ρ ;

Гибридные варианты по пути упрощения EM:

- EM с жёсткой кластеризацией на E-шаге;
- EM без настройки дисперсий (сферические гауссианы);

Частичное обучение (Semi-supervised learning)

Дано:

Y — множество кластеров;

$\{x_i\}_{i=1}^{\ell}$ — обучающая выборка;

$\{x_i, y_i\}_{i=1}^m$ — размеченная часть выборки, обычно $m \ll \ell$.

Найти:

$a: X \rightarrow Y$ — алгоритм кластеризации.

Как приспособить EM-алгоритм:

E-шаг: $g_{iy} := [y = y_i]$, $y \in Y$, $i = 1, \dots, m$;

Как приспособить k -means:

E-шаг: $y_i := \arg \min_{y \in Y} \rho(x_i, \mu_y)$, $i = m + 1, \dots, \ell$.

Недостатки k -means

- Чувствительность к выбору начального приближения.
- Необходимость задавать k ;

Способы устранения этих недостатков:

- Несколько случайных кластеризаций;
выбор лучшей по функционалу качества.
- Постепенное наращивание числа кластеров k
(аналогично EM-алгоритму)

Постановка задачи

Дано:

$X = \mathbb{R}^n$ — пространство объектов;

$Y = \{1, \dots, M\}$ — множество кластеров, M фиксировано;

$X^\ell = \{x_j\}_{j=1}^\ell$ — обучающая выборка объектов;

$\rho: X \times X \rightarrow \mathbb{R}_+$ — метрика.

Требуется:

построить алгоритм кластеризации $a: X \rightarrow Y$.

Предлагается:

- ввести центры кластеров $w_m \in \mathbb{R}^n$, $m = 1, \dots, M$;
- относить объект $x \in X$ к ближайшему кластеру (правило жёсткой конкуренции WTA — Winner Takes All):

$$a(x) = \arg \min_{m \in Y} \rho(x, w_m).$$

Метод стохастического градиента

Минимизация среднего внутрикластерного расстояния:

$$Q(w; X^\ell) = \frac{1}{2} \sum_{i=1}^{\ell} \rho^2(x_i, w_{a(x_i)}) \rightarrow \min_w, \quad w = (w_1, \dots, w_M);$$

Пусть метрика евклидова, $\rho^2(x_i, w_m) = \|w_m - x_i\|^2$.

$$\frac{\partial Q(w; X^\ell)}{\partial w_m} = \sum_{i=1}^{\ell} (w_m - x_i) [a(x_i) = m].$$

Градиентный шаг в методе SG: для случайного $x_i \in X^\ell$

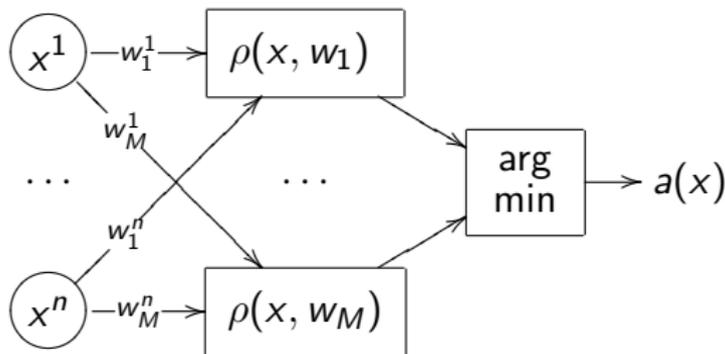
$$w_m := w_m + \eta(x_i - w_m) [a(x_i) = m]$$

(если x_i относится к кластеру m , то w_m сдвигается в сторону x_i).

Сеть Кохонена (сеть с конкурентным обучением)

Структура алгоритма — двухслойная нейронная сеть:

$$a(x) = \arg \min_{m \in Y} \rho(x, w_m)$$



Градиентное правило обучения напоминает перцептрон:

$$\text{если } a(x_i) = m, \text{ то } w_m := w_m + \eta(x_i - w_m).$$

Алгоритм SG (Stochastic Gradient)

Вход: выборка X^ℓ ; темп обучения η ; параметр λ ;

Выход: центры кластеров $w_1, \dots, w_M \in \mathbb{R}^n$;

1: инициализировать центры w_m , $m = 1, \dots, M$;

2: инициализировать текущую оценку функционала:

$$Q := \sum_{i=1}^{\ell} \rho(x_i, w_{a(x_i)});$$

3: **повторять**

4: выбрать объект x_i из X^ℓ (например, случайно);

5: вычислить кластеризацию: $m := \arg \min_{m \in Y} \rho(x_i, w_m)$;

6: градиентный шаг: $w_m := w_m + \eta(x_i - w_m)$;

7: оценить значение функционала:

$$Q := (1 - \lambda)Q + \lambda \rho(x_i, w_m);$$

8: **пока** значение Q и/или веса w не стабилизируются;

Жёсткая и мягкая конкуренция

Правило жёсткой конкуренции WTA (winner takes all):

$$w_m := w_m + \eta(x_i - w_m) [a(x_i) = m], \quad m = 1, \dots, M,$$

Недостатки правила WTM:

- медленная скорость сходимости;
- некоторые w_m могут никогда не выбираться.

Правило мягкой конкуренции WTM (winner takes most):

$$w_m := w_m + \eta(x_i - w_m) K(\rho(x_i, w_m)), \quad m = 1, \dots, M,$$

где ядро $K(\rho)$ — неотрицательная невозрастающая функция.

Теперь центры всех кластеров смещаются в сторону x_i , но чем дальше от x_i , тем меньше величина смещения.

Обоснование правила мягкой конкуренции

Жёсткая кластеризация WTA:

$$a(x) = \arg \min_{m \in Y} \rho(x, w_m).$$

Мягкая кластеризация WTM:

объект x «размазывается» по всем кластерам,

$$a_m(x) = K(\rho(x, w_m)) \quad m = 1, \dots, M;$$

Минимизация среднего внутрикластерного расстояния:

$$Q(w; X^\ell) = \frac{1}{2} \sum_{i=1}^{\ell} \sum_{m=1}^M a_m(x_i) \rho^2(x_i, w_m) \rightarrow \min_w;$$
$$\frac{\partial Q(w)}{\partial w_m} = \sum_{i=1}^{\ell} (w_m - x_i) a_m(x).$$

Карта Кохонена (Self Organizing Map, SOM)

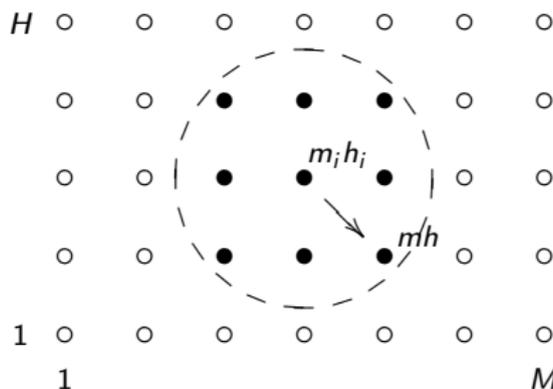
$Y = \{1, \dots, M\} \times \{1, \dots, H\}$ — прямоугольная сетка кластеров.

Каждому узлу (m, h) приписан нейрон Кохонена $w_{mh} \in \mathbb{R}^n$.

Наряду с метрикой $\rho(x_i, x)$ на X вводится метрика на сетке Y :

$$r((m_i, h_i), (m, h)) = \sqrt{(m - m_i)^2 + (h - h_i)^2}.$$

Окрестность (m_i, h_i) :



Обучение карты Кохонена

Вход: X^ℓ — обучающая выборка; η — темп обучения;

Выход: $w_{mh} \in \mathbb{R}^n$ — векторы весов, $m = 1..M$, $h = 1..H$;

- 1: $w_{mh} := \text{random} \left(-\frac{1}{2MH}, \frac{1}{2MH} \right)$ — инициализация весов;
- 2: **повторять**
- 3: выбрать объект x_i из X^ℓ случайным образом;
- 4: WTA: вычислить координаты кластера:
$$(m_i, h_i) := a(x_i) \equiv \arg \min_{(m,h) \in Y} \rho(x_i, w_{mh});$$
- 5: **для всех** $(m, h) \in \text{Окрестность}(m_i, h_i)$
- 6: WTM: сделать шаг градиентного спуска:
$$w_{mh} := w_{mh} + \eta(x_i - w_{mh}) K(r((m_i, h_i), (m, h)));$$
- 7: **пока** кластеризация не стабилизируется;

Интерпретация карт Кохонена

Два типа графиков — цветных карт $M \times H$:

- Цвет узла (m, h) — локальная плотность в точке (m, h) — среднее расстояние до k ближайших точек выборки;
- По одной карте на каждый признак:
цвет узла (m, h) — значение j -й компоненты вектора $w_{m,h}$.

Пример:

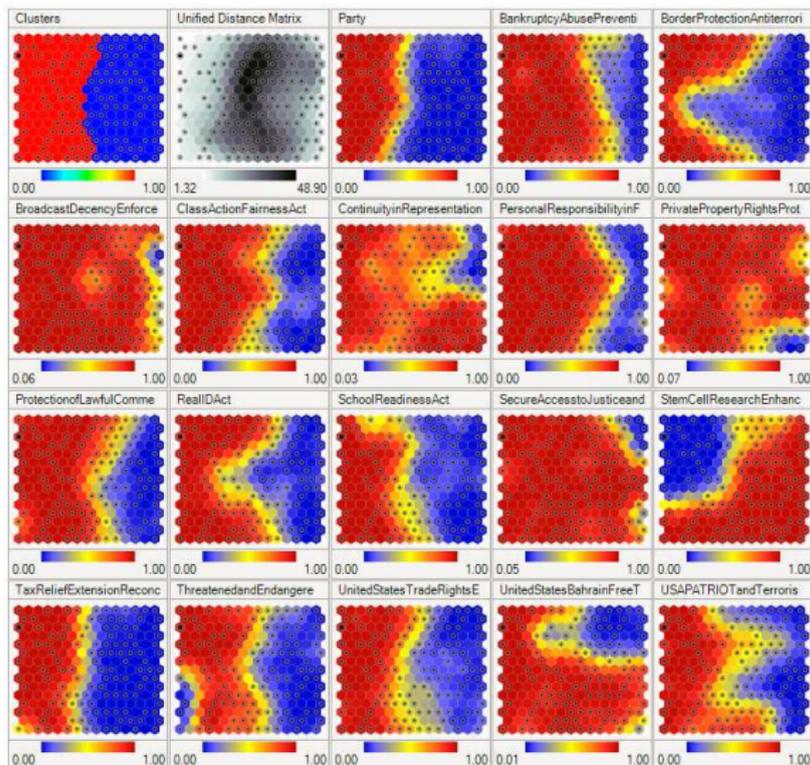
Задача UCI house-votes (US Congress voting patterns)

Объекты — конгрессмены;

Признаки — вопросы, выносившиеся на голосование;

Есть целевой признак {демократ, республиканец}.

Интерпретация карт Кохонена (пример)



Достоинства и недостатки карт Кохонена

Достоинства:

- Возможность визуального анализа многомерных данных.

Недостатки:

- **Субъективность.** Карта зависит не только от кластерной структуры данных, но и...
 - от свойств сглаживающего ядра;
 - от (случайной) инициализации;
 - от (случайного) выбора x_i в ходе итераций.
- **Искажения.** Близкие объекты исходного пространства могут переходить в далёкие точки на карте, и наоборот.

Рекомендуется только для разведочного анализа данных.

Кусочно-постоянная непараметрическая регрессия

Задача восстановления регрессии:

$$X^\ell = \{x_i, y_i\}_{i=1}^\ell, \quad y_i \in \mathbb{R}.$$

Основная идея — применить WTA-кластеризацию:

1) разбить выборку на M кластеров с центрами w_m :

$$c(x) = \arg \min_{m=1, \dots, M} \rho(x, w_m);$$

2) на каждом кластере построить регрессию-константу:

$$a(x) = v_{c(x)} = \sum_{m=1}^M v_m [c(x) = m].$$

Требуется: по выборке X^ℓ найти центры w_m и уровни v_m .

Кусочно-постоянная непараметрическая регрессия

Настройка центров w_m сетью Кохонена — WTA.

Задача настройки уровней v_m решается аналитически:

$$Q(v) = \frac{1}{2} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2 \rightarrow \min_v;$$
$$\frac{\partial Q}{\partial v_m} = \sum_{i=1}^{\ell} (a(x_i) - y_i) [c(x_i) = m] = 0.$$

Подставляя сюда $a(x_i) = v_m$, получаем среднее y_i по кластеру:

$$v_m = \frac{\sum_{i=1}^{\ell} y_i [c(x_i) = m]}{\sum_{i=1}^{\ell} [c(x_i) = m]}.$$

Кусочно-гладкая непараметрическая регрессия

Усложнение: теперь функция $a(x)$ должна быть гладкой.

Основная идея — применить **WTM-кластеризацию**:

1) «мягкая» кластеризация на M кластеров с центрами w_m :

$$c_m(x) = K(\rho(x, w_m)), \quad m = 1, \dots, M;$$

2) формула Надарая-Ватсона сглаживает ответы v_m , но не по ℓ объектам, а по M центрам кластеров w_m :

$$a(x) = \frac{\sum_{m=1}^M v_m c_m(x)}{\sum_{m=1}^M c_m(x)}.$$

Требуется: по выборке X^ℓ найти центры w_m и уровни v_m .

Кусочно-гладкая непараметрическая регрессия

Настройка центров w_m сетью Кохонена — WTM:

$$Q(w) = \frac{1}{2} \sum_{i=1}^{\ell} \sum_{m=1}^M c_m(x_i) \|w_m - x_i\|^2 \rightarrow \min_w;$$

$$\frac{\partial Q(w)}{\partial w_m} = \sum_{i=1}^{\ell} c_m(x_i) (w_m - x_i);$$

Настройка уровней v_m линейным нейроном:

$$Q(v) = \frac{1}{2} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2 \rightarrow \min_v;$$

$$\frac{\partial Q(v)}{\partial v_m} = \sum_{i=1}^{\ell} \frac{c_m(x_i)}{\sum_{s=1}^M c_s(x_i)} (a(x_i) - y_i);$$

Кусочно-гладкая непараметрическая регрессия

Веса слоя Кохонена w_m и веса линейного слоя v_m на каждой итерации SG обновляются независимо:

$$\begin{cases} w_m := w_m + \eta_1(x_i - w_m)K(\rho(x, w_m)); \\ v_m := v_m - \eta_2(a(x_i) - y_i)K(\rho(x, w_m)); \end{cases}$$

Достоинства этого метода:

- регрессия учитывает кластерную структуру выборки;
- повышается эффективность вычисления $a(x)$.