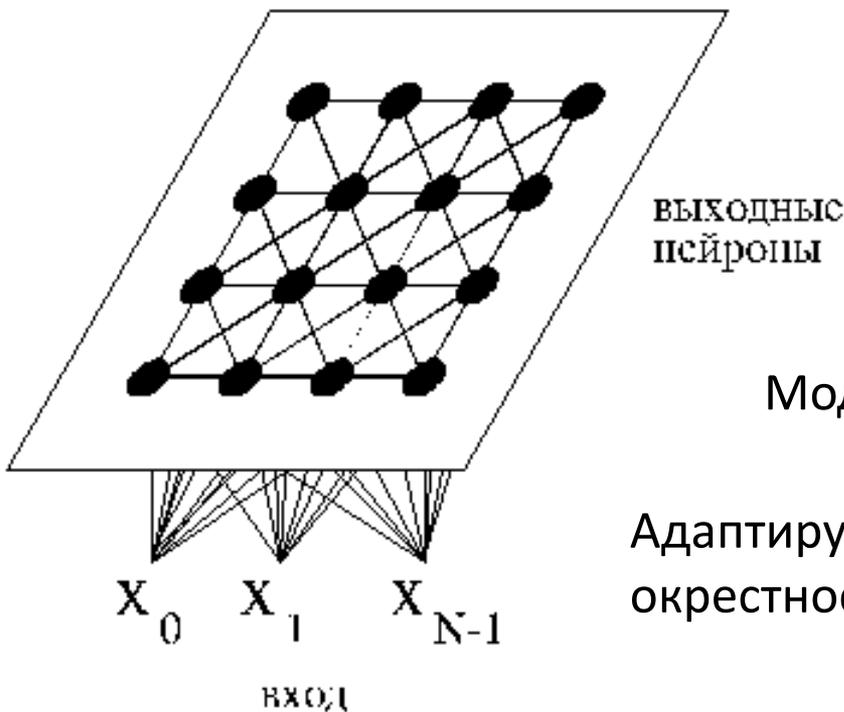


Нейронные сети и их практическое применение.

Лекция 7. Сеть Кохонена.

Дмитрий Буряк
к.ф.-м.н
dyb04@yandex.ru

Сеть с самоорганизацией на основе конкуренции



$$w_i = (w_{i0}, w_{i1}, \dots, w_{i(N-1)})^T$$

$$d(x, w_j) = \min_{0 \leq i < N} d(x, w_i)$$

$$w_i(k+1) = w_i(k) + \eta_i(k)[x - w_i(k)]$$

Модификация весов по правилу Кохонена.

Адаптируются только веса, лежащие в некоторой окрестности нейрона-победителя.

Меры расстояния между векторами

евклидова мера: $d(x, w_i) = \|x - w_i\| = \sqrt{\sum_j (x_j - w_{ij})^2}$

скалярное произведение: $d(x, w_i) = 1 - \frac{\langle x, w_i \rangle}{\|x\| \|w_i\|} = \cos(x, w_i)$

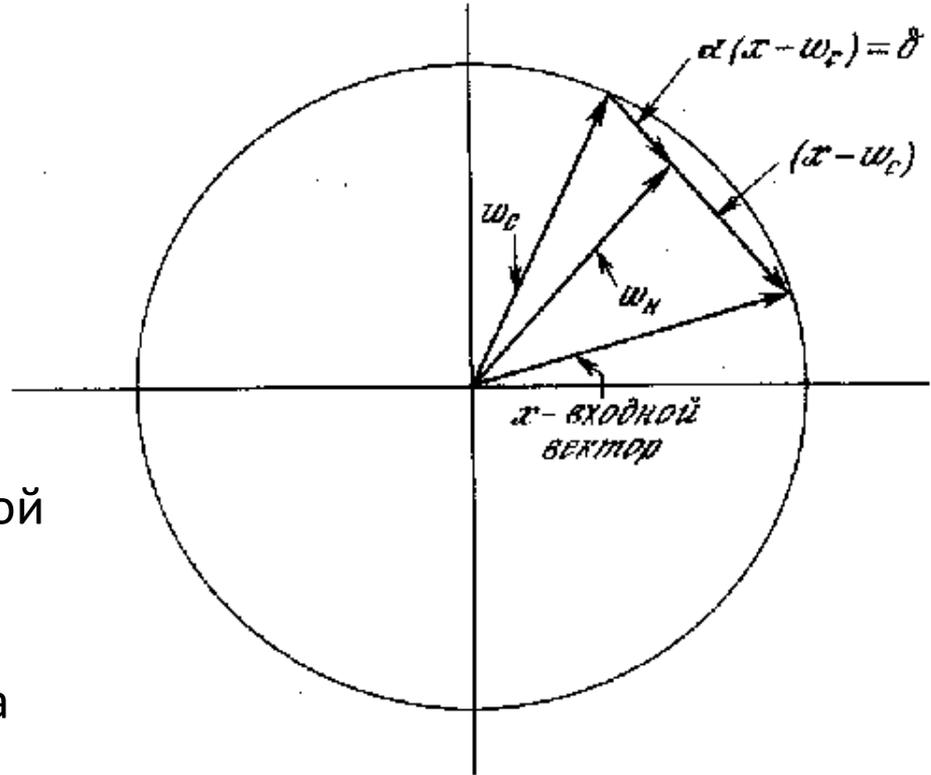
Манхэттен: $d(x, w_i) = \sqrt{\sum_j |x_j - w_{ij}|}$

мера относительно нормы L_∞ : $d(x, w_i) = \max_j |x_j - w_{ij}|$

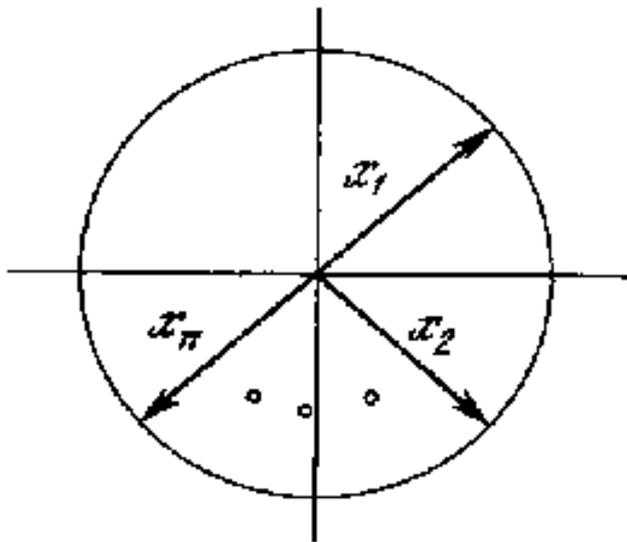
Общий смысл обучения

$$w_H = w_C + \alpha[x - w_C]$$

- В результате обучения веса нейронов определяют центры кластеров, выделенных в обучающих данных.
- После обучения кластер, к которому будет отнесен входной вектор определяется вектором весов нейрона, до которого расстояние от входного вектора минимально.



Инициализация весов



1. Инициализация случайным образом (с нормализацией).
2. Метод выпуклой комбинации:

$$w_{ij} = \frac{1}{\sqrt{N}}$$

$$x_i = \alpha x_i + \frac{1-\alpha}{\sqrt{N}}$$

3. Добавление шума к входным векторам.

Проблема мертвых нейронов

1. Начальная адаптация всех весов.
2. Учет активности нейрона («чувство справедливости»)
 - контролирование частоты выигрыша каждого нейрона;
 - вычисление потенциала нейронов:

$$p_i(k+1) = \begin{cases} p_i(k) + \frac{1}{N} (i \neq h); & \text{При проигрыше потенциал нейрона растет} \\ p_i(k) - p_{\min} (i = h) & \text{При выигрыше потенциал нейрона падает} \end{cases}$$

$$p_{\min} \approx 0.75$$

- модификация расстояния между входным вектором и вектором весов:

$$d(x, w_i) \leftarrow N_i d(x, w_i)$$

Алгоритмы обучения

$$E = \frac{1}{N} \sum_{i=1}^N \|x_i - w_{w(i)}\|^2$$

1. WTA с учетом активности нейронов (CWTA): $w_h = w_h + \eta[x - w_h]$

2. WTM (Winner takes most): $w_i = w_i + \eta_i G(i, x)[x - w_i]$

а) классический алгоритм Кохонена:

- функция G определяет прямоугольную окрестность;

- соседство гауссовского типа:

$$G(i, x) = e^{-\frac{d^2(i, w)}{2\lambda^2}}$$

б) алгоритм нейронного газа:

Сортировка векторов весов по расстояниям до входного вектора;

$G(i, x) = e^{-\frac{m(i)}{\lambda}}$, где $m(i)$ - номер нейрона i в полученной перестановке по расстояниям.

Применение сети Кохонена

1. Компрессия данных.

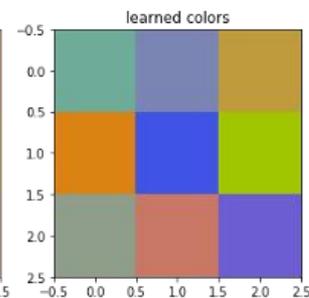
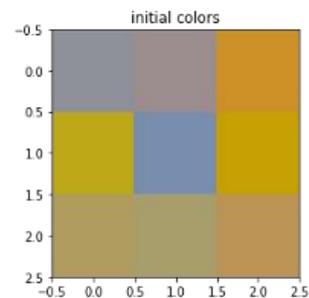
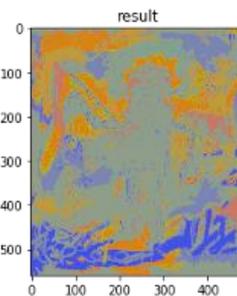
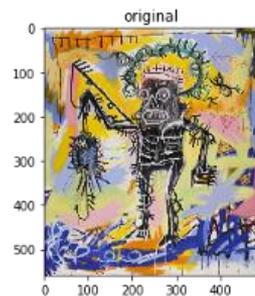
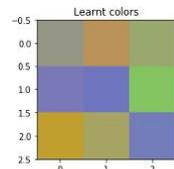
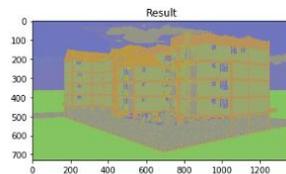
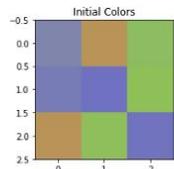
Кадр разбивается на части (входные вектора).

Каждой части сопоставляется нейрон-победитель.

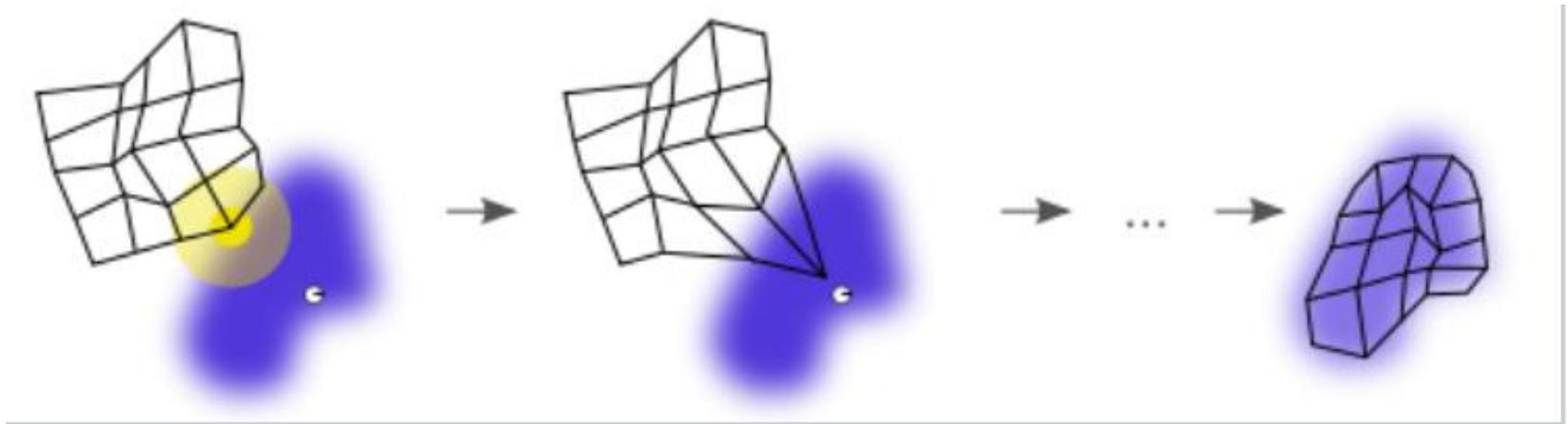
2. Диагностирование неисправностей.

3. Визуализация многомерной информации.

Цветное квантование

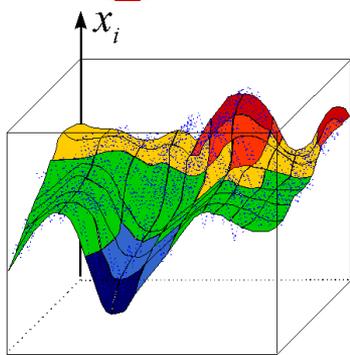
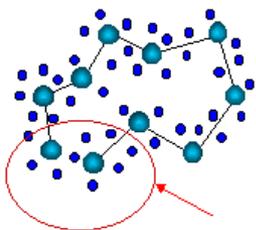
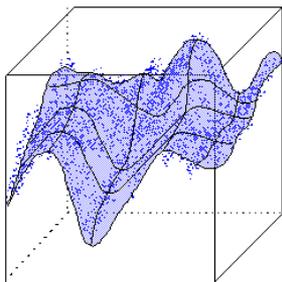


Аппроксимация



Пример: <https://vimeo.com/171493634>

Визуализация многомерной информации



- ❑ Двумерная топографическая карта набора трехмерных данных.
- ❑ Каждая точка в трехмерном пространстве попадает в свою ячейку сетки имеющую координату ближайшего к ней нейрона из двумерной карты.
- ❑ **Свойство локальной близости: близкие на карте области близки в исходном пространстве. Наоборот в общем случае не верно.**
- ❑ Раскраска по i -му признаку.

Карты всех признаков образуют топографический атлас.

Анализ сортов вин

Исходные данные.

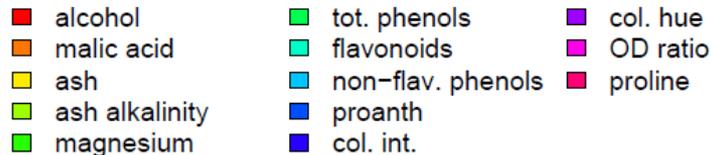
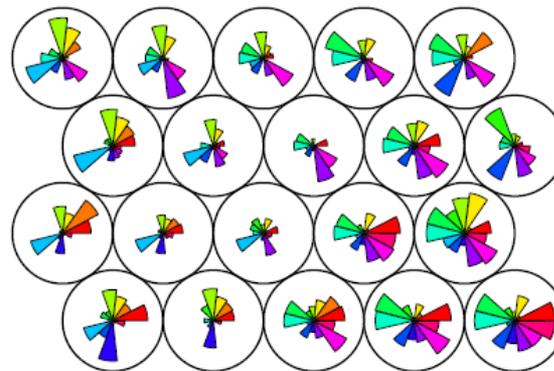
177 наименований вин

13 параметров (алкоголь, яблочная кислота, магний и т.п.).

Задача

Выделить кластеры

Wine data



Достоинства и недостатки

Достоинства:

1. Быстрое обучение;
2. Устойчивость к помехам.

Недостатки:

Решение задач кластеризации только при известном числе кластеров.

Вопросы

1. Какая архитектура у сети Кохонена?
2. Основной принцип обучения сети Кохонена.
3. Какую базовую задачу решает сеть Кохонена?