Введение в компьютерный и интеллектуальный анализ данных

Воротницкая Т.И.

5. Data Mining

- **Data Mining** — «Добыча данных», интеллектуальный анализ данных, совокупность методов для обнаружения в массивах данных новых знаний: нетривиальных, полезных и доступных для интерпретации.
- Специфика современных требований к обработке данных:
  - Данные имеют неограниченный объем
  - Данные являются разнородными (количественными, качественными, текстовыми)
  - Результаты должны быть конкретны и понятны
  - Инструменты для обработки сырых данных должны быть просты в использовании
- Почему не работает статистика: не работает концепция усреднения по выборке, статистика работает с фиктивными величинами (средняя температура по больнице) и подходит для проверки ранее сформулированных гипотез.

5. Data Mining

	X_1	X_2	…	X_k	Y
n_1	x_{11}	x_{12}	…	x_{1k}	y_1
n_2	x_{21}	x_{22}	…	x_{2k}	y_2
…	…	…	…	…	…
n_N	x_{N1}	x_{N2}	…	x_{Nk}	y_N

5. Data Mining
Задачи

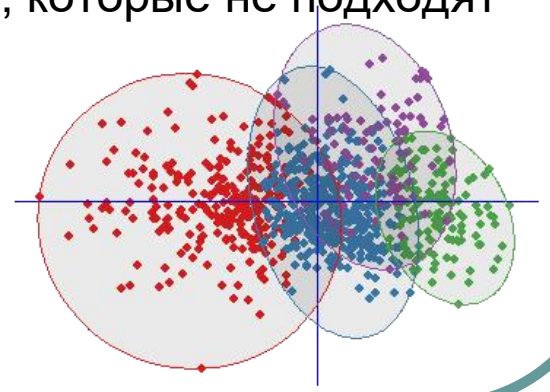
- **Классификация** – это отнесение объектов к одному из заранее известных классов.
- **Кластеризация** – это группировка объектов на основе данных, описывающих сущность этих объектов.
- **Регрессия** (задачи прогнозирования) - установление зависимости непрерывных выходных от входных переменных.
- **Ассоциация** – выявление закономерностей между связанными событиями. Примером такой закономерности служит правило, указывающее, что из события X следует событие Y.
- **Визуализация** – создание графического образа анализируемых данных.
- **Последовательные шаблоны** – установление закономерностей между связанными во времени событиями, т.е. обнаружение зависимости, что если произойдет событие X, то спустя заданное время произойдет событие Y.
- **Анализ отклонений** – выявление наиболее нехарактерных шаблонов.

5. Кластерный анализ.

Кластерный анализ —процедура, выполняющая сбор данных, содержащих информацию о выборке объектов, и затем упорядочивающая объекты в сравнительно однородные группы

Основные цели кластеризации:

- упрощение дальнейшей обработки данных, т.е. разбиение исходного множества объектов на кластеры для работы с каждой группой в отдельности;
- сжатие данных – сокращение объёма хранимых данных (оставить по одному представителю от каждого кластера);
- выделение нетипичных объектов (элементы, которые не подходят ни к одному из кластеров);
- построение иерархии множества объектов.



5. Кластерный анализ.

Кластерный анализ включает в себя два вспомогательных этапа:

- предварительная обработка (pre-processing),
- постобработка данных (post-processing).

pre-processing: нормализация данных (путем вычитания среднего и деления на стандартное отклонение), удаление объектов-выбросов.

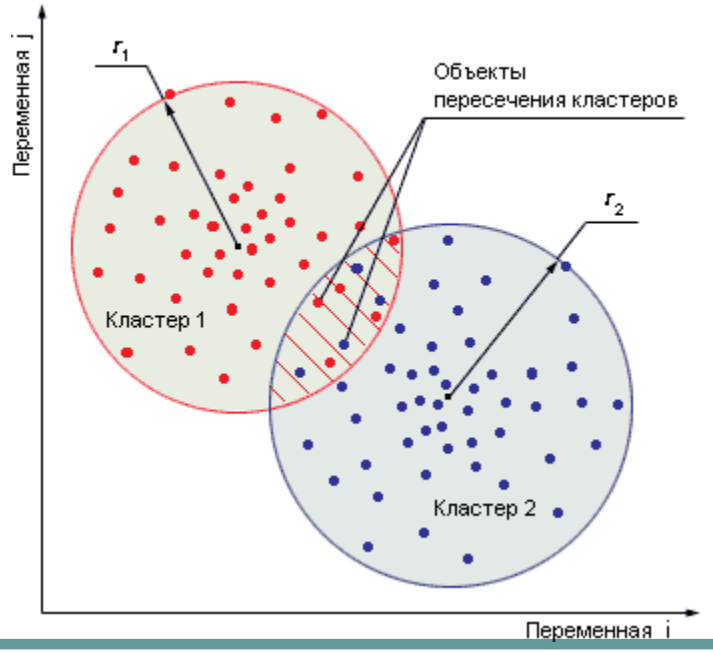
post-processing: удаление малых кластеров и кластеров-выбросов, дробление больших кластеров, объединение близких кластеров.

Для нахождения решения задач кластерного анализа необходимо:

- задать способ сравнения объектов между собой (меру сходства);
- способ кластеризации;
- установить число кластеров;
- $F = \sum_{j=1}^{n} \left(x_j - \bar{x}\right)^2 \rightarrow min$

5. Кластерный анализ.
Характеристики кластера

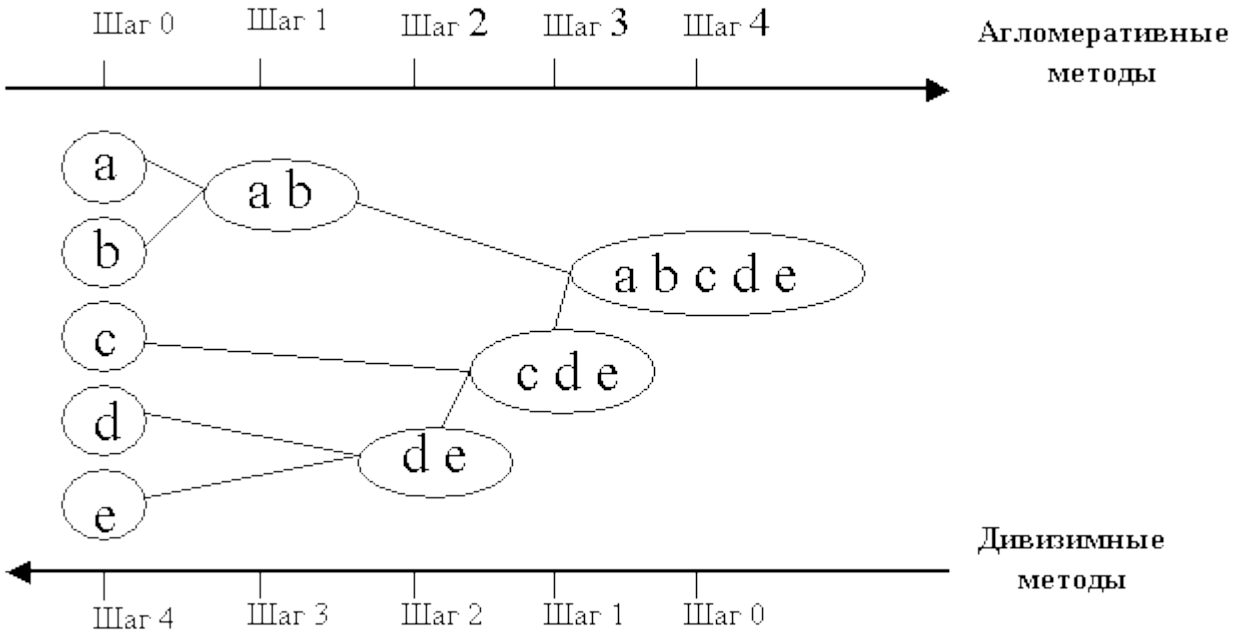
- Центр – среднее геометрическое место точек в пространстве переменных
- Радиус – максимальное расстояние точек от центра кластера
- Размер – радиус кластера или среднеквадратичное отклонение объектов кластера



5. Кластерный анализ.
Классификация алгоритмов

- По способу обработки данных
  - Иерархические
    - нисходящие
    - восходящие
  - Неиерархические
- По способу анализа данных
  - четкие (непересекающиеся) - каждому объекту выборки ставят в соответствие номер кластера
  - нечеткие (пересекающиеся) - объекту ставят в соответствие набор вещественных значений, показывающих степень отношения объекта к кластерам.
- По возможности расширения объема обрабатываемых данных
  - Масштабируемые
  - Немасштабируемые

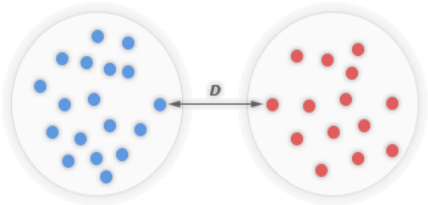
5. Кластерный анализ.
Иерархические методы



5. Кластерный анализ.
Метод поиска ближайшего соседа

Метод одиночной связи

- Расстояние между двумя кластерами определяется как расстояние между ближайшими их представителями



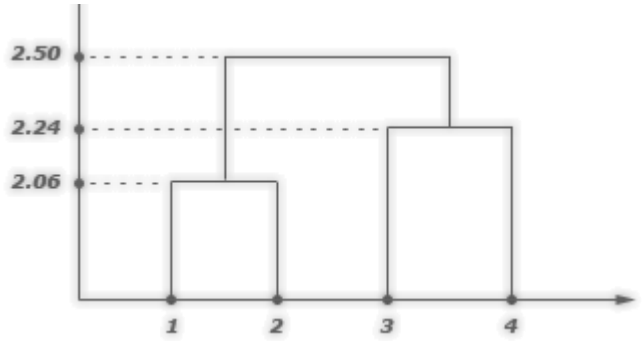
1. рассчитываем матрицу расстояний между объектами;
2. находим минимальное значение, соответствующее расстоянию между двумя наиболее близкими кластерами;
3. объединяем найденные кластеры в новый;
4. повторяем шаги 2-3 до тех пор, пока число объектов (кластеров) не станет меньше заданного

5. Кластерный анализ.
Метод поиска ближайшего соседа

	1	2	3	4
1	0	2,06	4,03	6,32
2	2,06	0	2,5	4,12
3	4,03	2,5	0	2,24
4	6,32	4,12	2,24	0

	1,2	3	4
1,2	0	2,5	4,12
3	2,5	0	2,24
4	4,12	2,24	0

	1,2	3,4
1,2	0	2,5
3,4	2,5	0



5. Кластерный анализ.
Метод k-средних

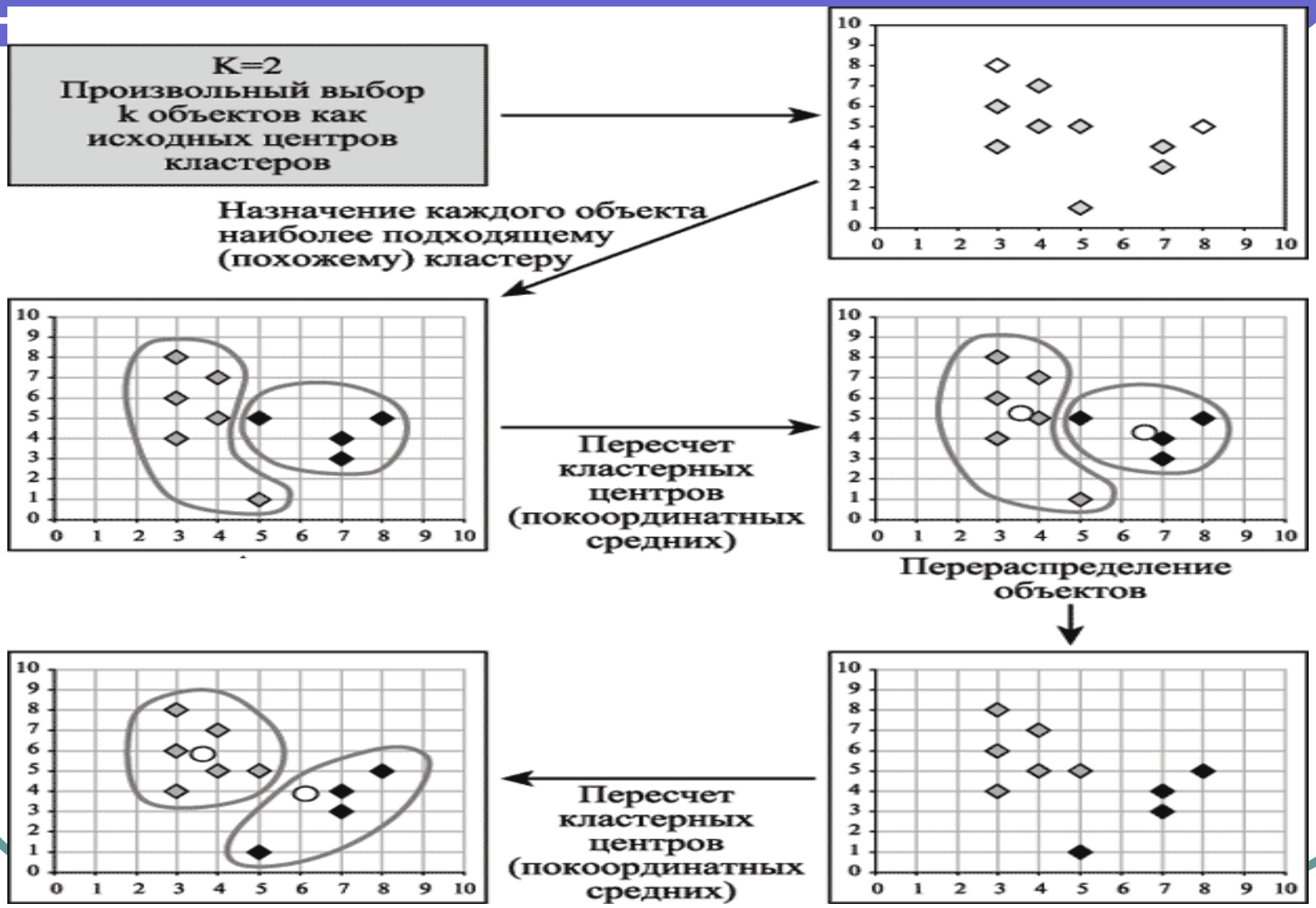
1. Первоначальное распределение объектов по кластерам.

- Выбирается число k, и на первом шаге выбираем k точек, которые считаются начальными центроидами - "центрами" кластеров. Каждому кластеру соответствует один центр.
- Выбор начальных центроидов может осуществляться следующим образом:
  - выбор k-наблюдений для максимизации начального расстояния;
  - случайный выбор k-наблюдений;
  - выбор первых k-наблюдений.
- Для каждого объекта определить ближайший к нему центроид
- В результате каждый объект назначен одному из k начальных кластеров.

2. Итеративный процесс.

- Вычисляются *центры кластеров*, которыми считаются покоординатные средние кластеров.
- Объекты перераспределяются к этим центрам.
- Процесс вычисления центров и перераспределения объектов продолжается до тех пор, пока не выполнено одно из условий:
  - кластерные центры стабилизировались, т.е. все наблюдения принадлежат кластеру, которому принадлежали до текущей итерации;
  - число итераций равно максимальному числу итераций.

5. Кластерный анализ.
Метод k-средних



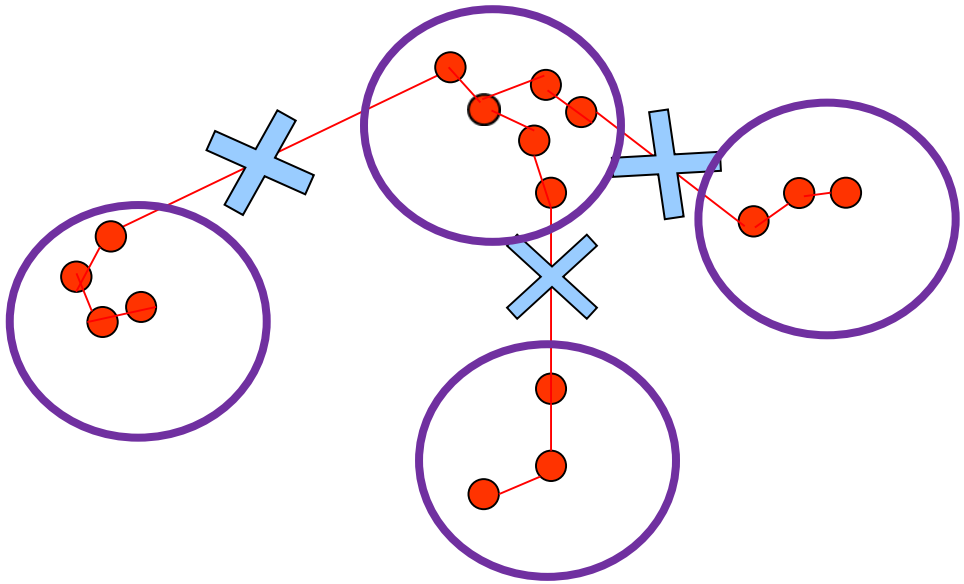
5. Кластерный анализ.
Графовые методы

Алгоритм кратчайшего незамкнутого пути (КНП)
(требует предположения о количестве кластеров k)

Алгоритм:

1. Установить количество кластеров k. Найти две близлежащие точки и соединить их ребром.
2. Найти изолированную точку, ближайшую к некой неизолированной, и соединить эти две точки ребром.
3. Если в выборке остаются изолированные точки, то перейти к п.2, иначе к п.4.
4. Удалить (k-1)самых длинных ребер.
5. Скрепленные ребрами группы точек объединить в кластеры.

5. Кластерный анализ.
Графовые методы



5. Кластерный анализ.
Функции в R

kmeans (x, centers, iter.max = 10, nstart = 1, algorithm = c("Hartigan-Wong", "Lloyd", "Forgy", "MacQueen"))

clara (x, k, metric = "euclidean", stand = FALSE, samples = 5, sampsize = min(n, 40 + 2 * k), trace = 0, medoids.x = TRUE, keep.data = medoids.x, rngR = FALSE)

agnes(x, diss = inherits(x, "dist"), metric = "euclidean", stand = FALSE, method = "average", par.method, keep.diss = n < 100, keep.data = !diss)

6. Дискриминантный анализ.

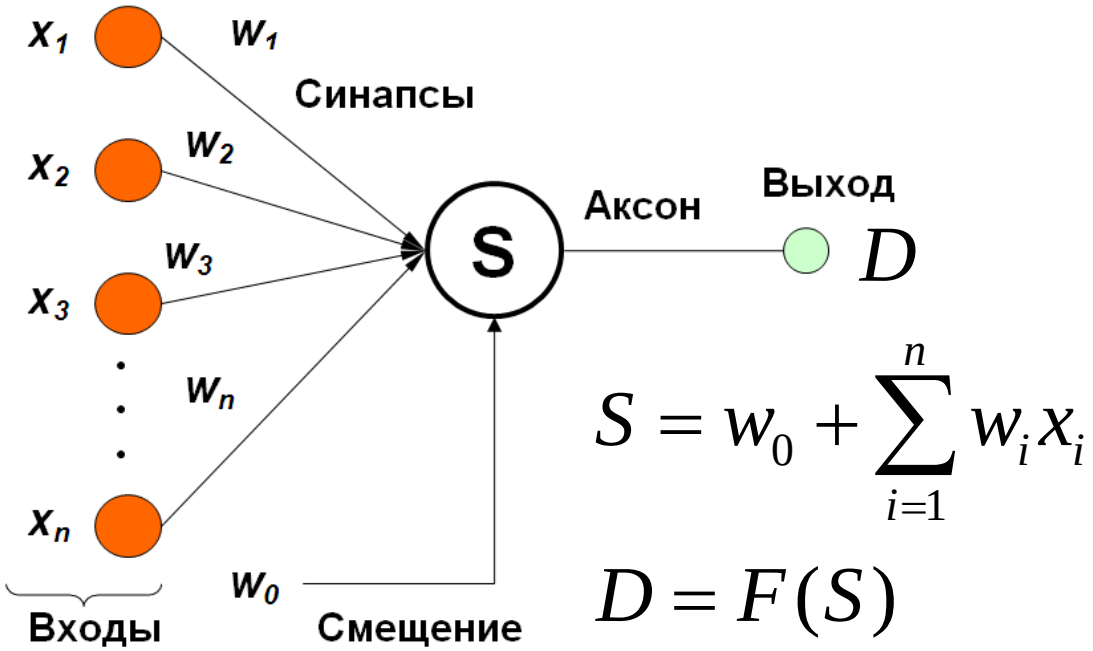
- Дискриминантный анализ – отнесение объекта к одной из заранее заданных групп на основании некоторых признаков (значений независимых переменных)
- Дискриминантная функция:

$$d = \sum_{i=1}^{n} w_i x_i$$

- Необходимо определить такие значения весовых коэффициентов, чтобы по значению дискриминантной функции с максимальной четкостью провести разделение по группам.

6. Дискриминантный анализ.
Нейронная сеть

- Общий вид искусственного нейрона:

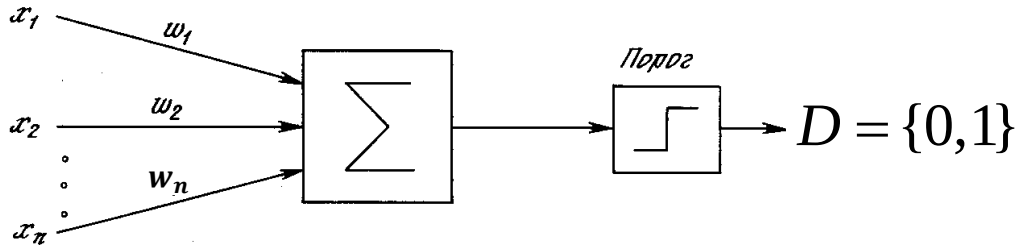


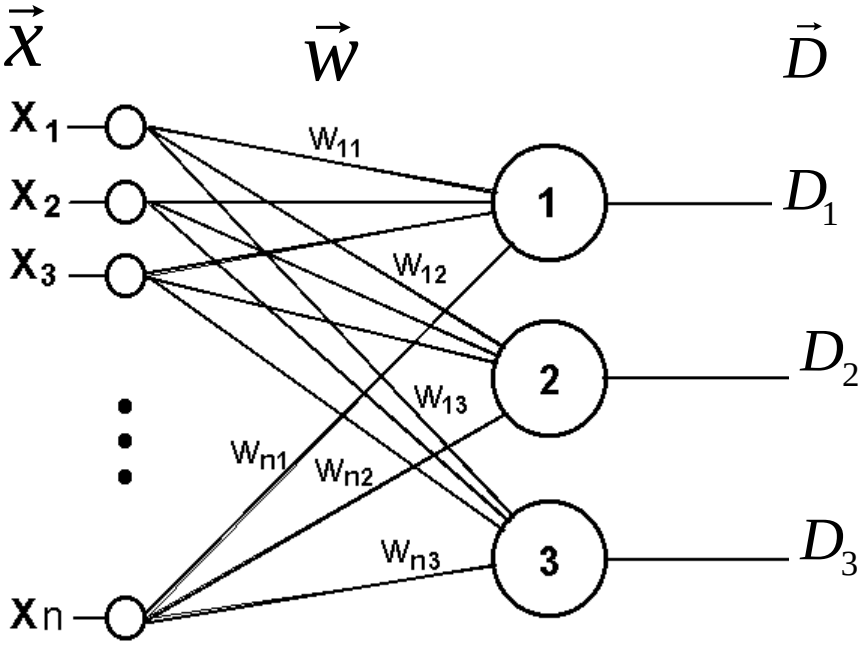
$$S = w_0 + \sum_{i=1}^{n} w_i x_i$$

$$D = F(S)$$

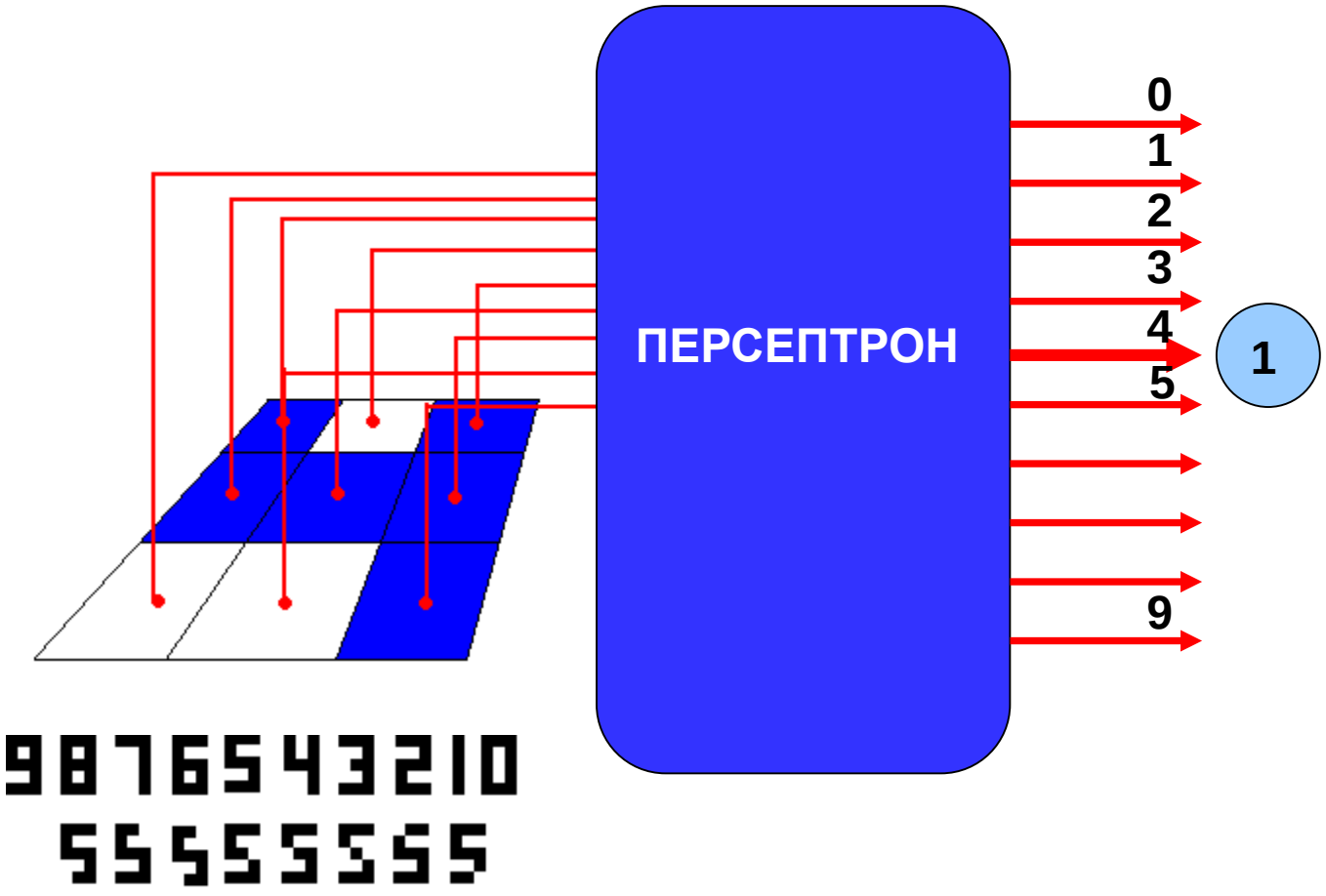
6. Дискриминантный анализ
Однослойный персептрон

$x_i = \{0,1\}$





6. Дискриминантный анализ
Однослойный персептрон



6. Дискриминантный анализ
Обучение персептрона (обучение с учителем)

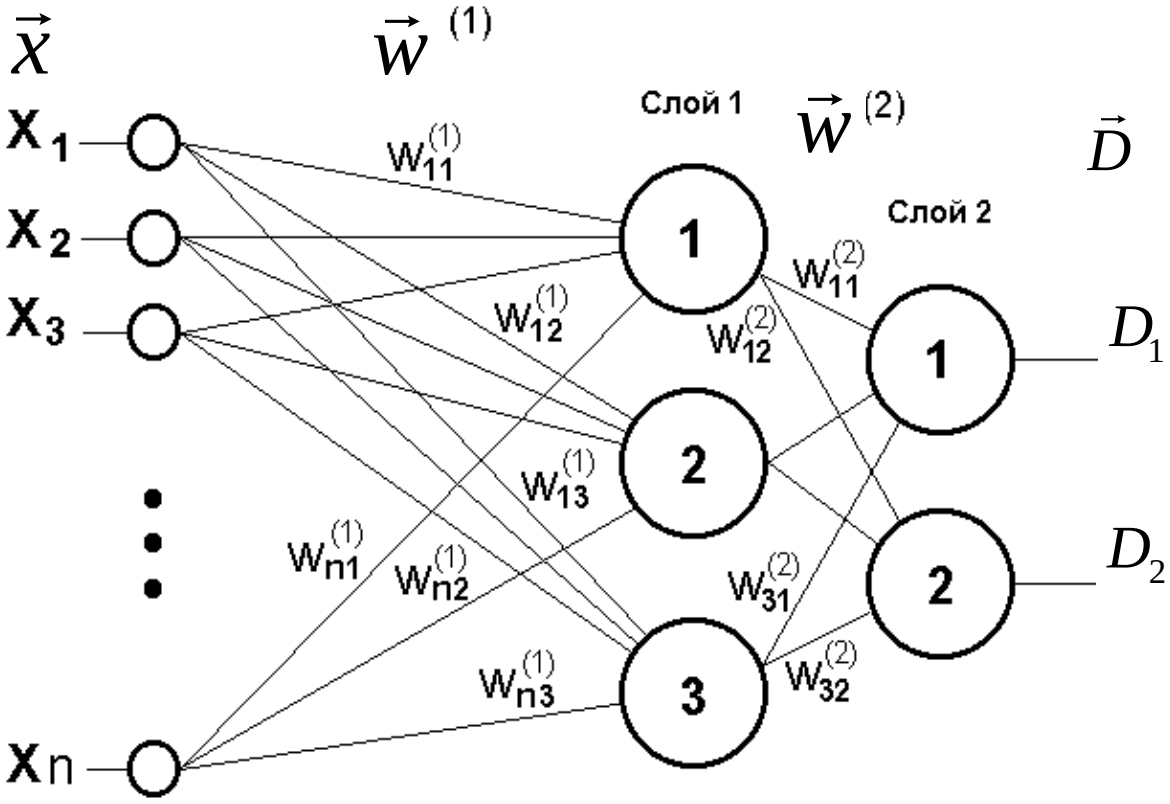
- Создаем обучающую выборку (известно, к какому классу относится каждый элемент)
- Присваиваем весам w_{ji} некоторые значения.
- Повторять, пока для всех элементов обучающей выборки не будет достигнуто правильное распознавание
  1. Для всех элементов обучающей выборки выполнить:
     2. Подать на вход вектор $\vec{x} = (x_1, x_2, ..., x_i, ..., x_n)^T$
     3. Если на выходе D_j неправильное значение, тогда:
        4. $\delta = T_j - D_j$ (здесь T_j – ожидаемое правильное значение выхода D_j)
        5. Изменяем веса: $w_{ji} = w_{ji} + \eta \delta x_i$ (веса, для которых на входе $x_i=0$, не меняются)

Свойства персептрона:

- **доказана сходимость алгоритма обучения;**
- **применение ограничено (например, нельзя воспроизвести функцию XOR)**

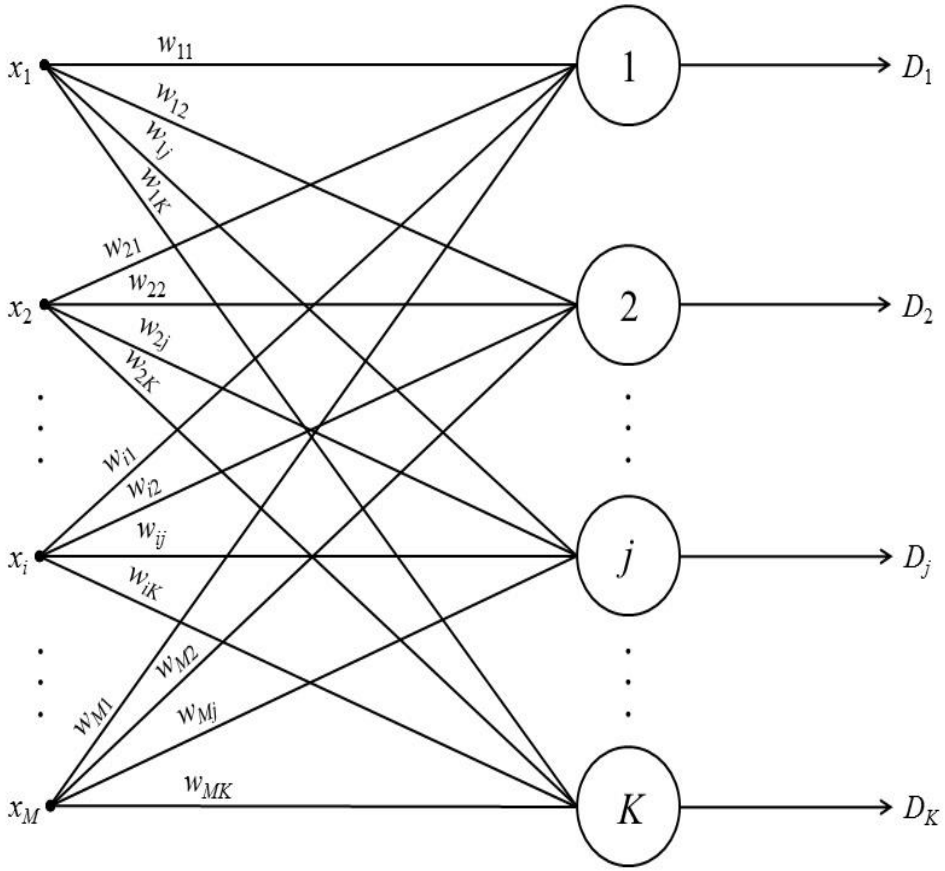
6. Дискриминантный анализ
Двухслойный персептрон

$x_i = \{0,1\}$ $D = \{0,1\}$



6. Дискриминантный анализ.
Слой Кохонена

- Слой Кохонена



$$S_j = w_{j0} + \sum_{i=1}^{n} w_{ji} x_i$$

$$D_j = F(S_j)$$

$$j_{\max} = \arg\max_j \{D_j\}$$

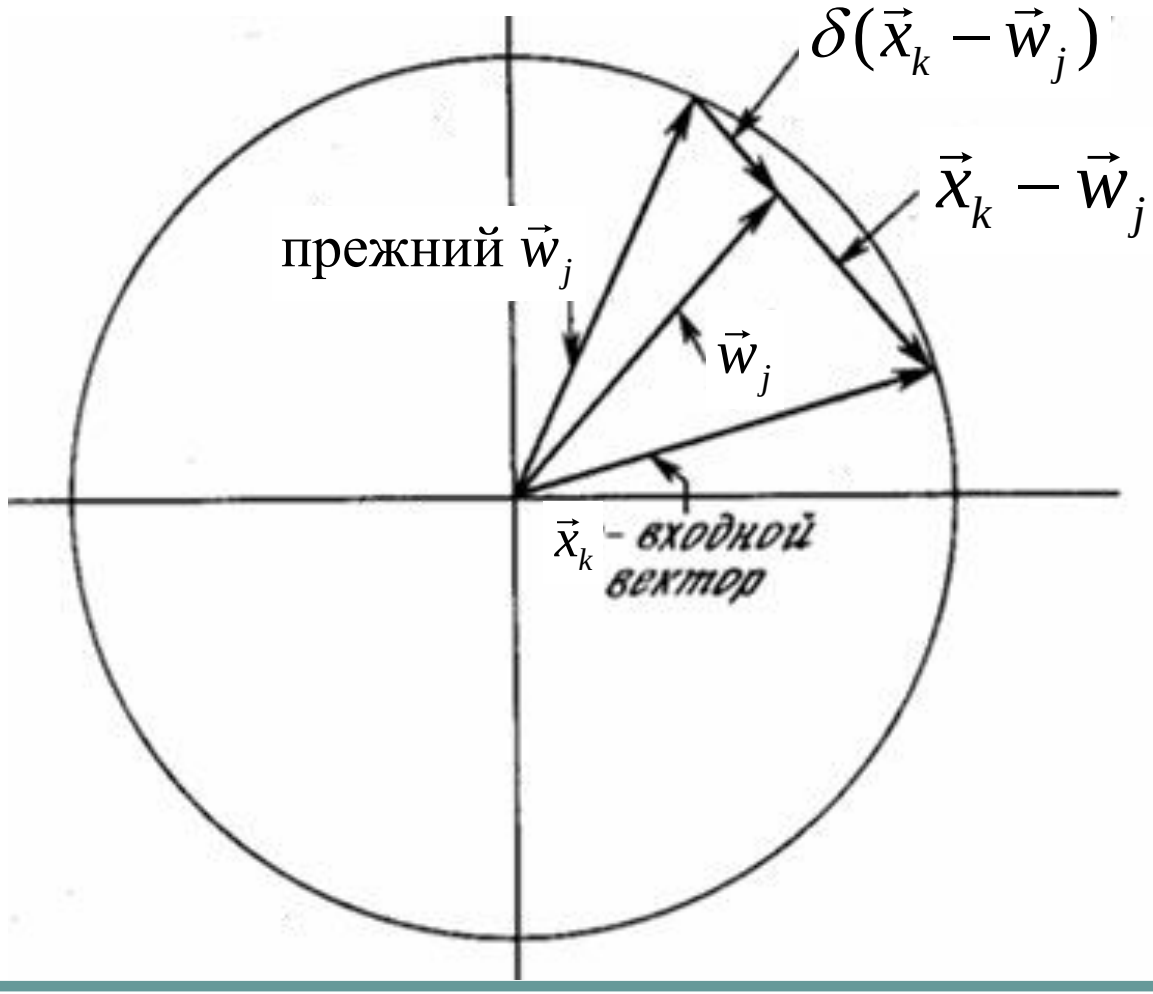
6. Дискриминантный анализ.
Обучение слоя Кохонена (обучение без учителя)

- По сути, решается задача классификации, где априори задано число классов: близкие вектора должны давать один и тот же результат

  - Присваиваем весам w_{ji} некоторые значения.
  - Повторять, пока веса не перестанут изменяться:
    1. Для всех элементов обучающей выборки выполнить:
       2. Подать на вход вектор $\vec{x}_k$ из обучающей выборки X
       3. Рассчитать выход слоя Кохонена, определить выигравший нейрон j
       4. Корректировать веса j-го нейрона: $\vec{w}_j = \vec{w}_j + \eta(\vec{x}_k - \vec{w}_j)$ (в скалярной форме: $w_{ji} = w_{ji} + \eta(x_{ki} - w_{ji})$

. Дискриминантный анализ.

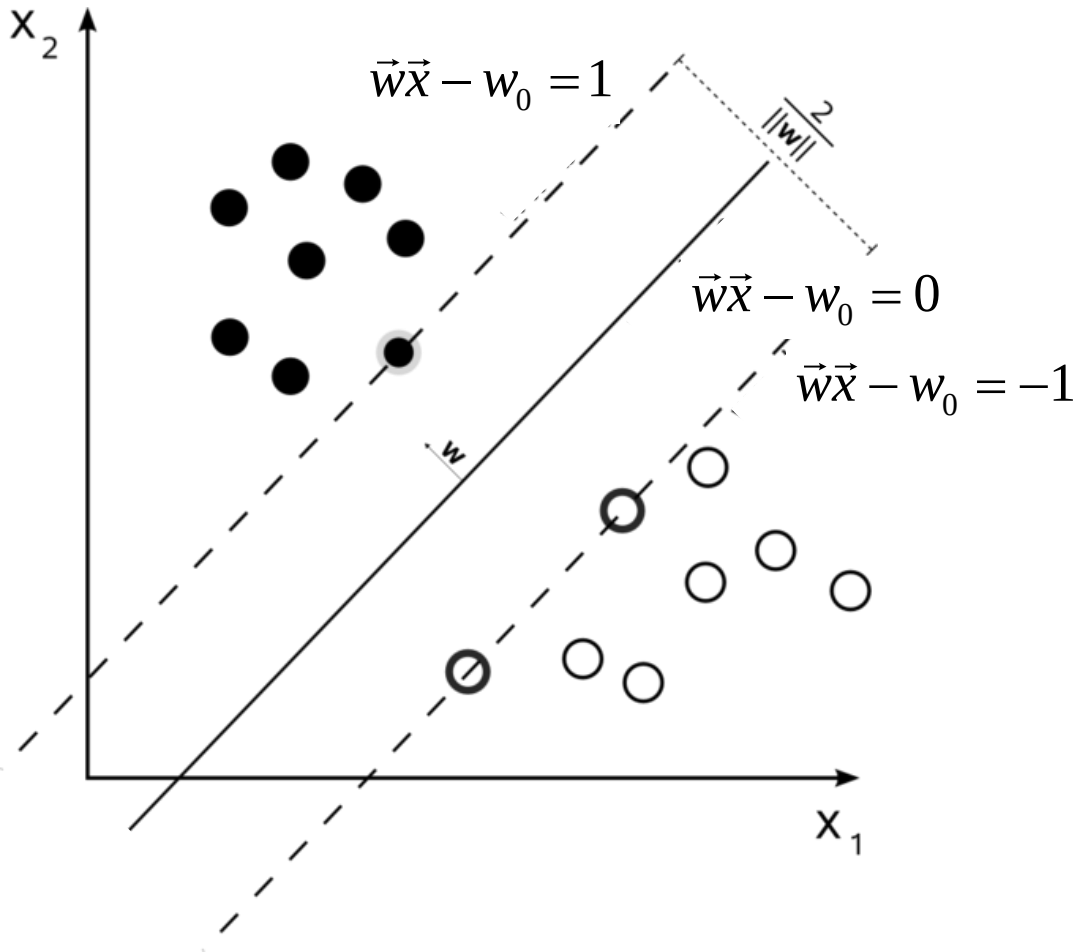
Обучение слоя Кохонена (обучение без учителя)



6. Дискриминантый анализ.
Метод опорных векторов (Support Vector Machine)

- Задача – бинарная классификация
- Классы определяются с помощью границ областей
- Плоскость решения – плоскость (гиперплоскость), разделяющая объекты, принадлежащие к разным классам
- Оптимальная разделяющая плоскость – плоскость решения, максимально далеко отстоящая от ближайших объектов обоих классов
- Опорные вектора – вектора признаков объектов, лежащих на границе между классами

6. Дискриминантный анализ.
Метод опорных векторов (Support Vector Machine)



6. Дискриминантный анализ.
Линейный SVM

- Необходимо построить линейный пороговый классификатор:

$$D(\vec{x}) = sign\left(\sum_{j=1}^{n} w_j x_j - w_0\right) = sign(\vec{w}\vec{x} - w_0)$$

- Уравнение $\vec{w}\vec{x} - w_o = 0$ описывает гиперплоскость, разделяющую классы в пространстве $\mathbf{R^n}$
- Предположим, что выборка линейно разделима: $\exists\ \vec{w}$ и w_0, для которых функционал числа ошибок на тестовой выборке

$$Q(\vec{w}, w_0) = \sum_{k=1}^{t} \left[Z_k (\vec{w}\vec{x}_k - w_0) < 0 \right] = 0$$

- Здесь t – число объектов в тестовой выборке, Z_k принимает значение 1 при правильной классификации и -1 – при неправильной.

6. Дискриминантный анализ.
Линейный SVM

- Нормировка:

$$D(\vec{x}) = sign\left(\sum_{j=1}^{n} w_j x_j - w_0\right) = sign(\vec{w}\vec{x} - w_0) = \begin{Bmatrix} 1 \\ -1 \end{Bmatrix}$$

- В уравнении $\vec{w}\vec{x} - w_o = 0$ выберем $\vec{w}$ и w_0 так, чтобы для всех пограничных объектов $\vec{x}_i^+$ и $\vec{x}_i^-$ тестовой выборки $\vec{x}_k \in X^t$ выполнялось условие нормировки:

$$\vec{w}\vec{x}_i^+ - w_o = D(\vec{x}_i^+) = y_i = 1; \qquad \vec{w}\vec{x}_i^- - w_o = D(\vec{x}_i^-) = y_i = -1;$$

(оптимальная плоскость находится на равных расстояниях от объектов на границе с обоих сторон)

- Тогда для всех объектов тестовой выборки получаем

$$\vec{w}\vec{x}_k - w_o \begin{cases} \le -1, \text{если } D(\vec{x}_k) = -1 \\ \ge 1, \text{если } D(\vec{x}_k) = +1 \end{cases}$$

6. Дискриминантный анализ.
Линейный SVM

- Ширина разделяющей полосы:

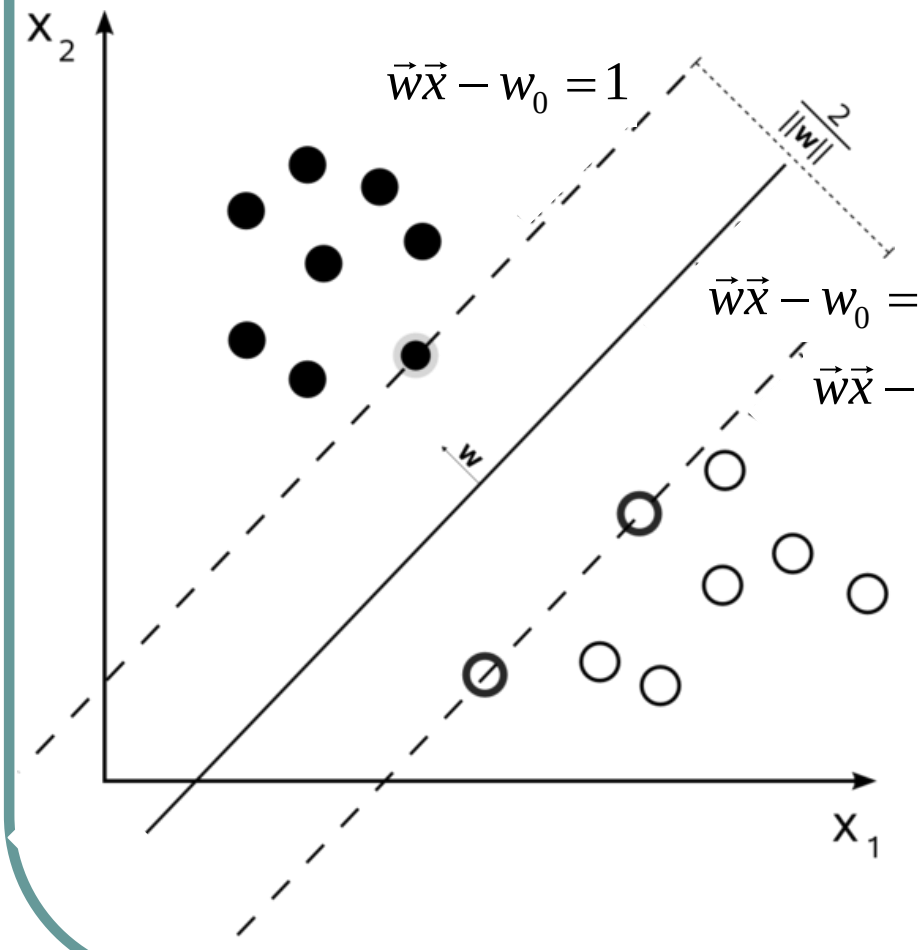
$$\left((\vec{x}^{+} - \vec{x}^{-}) \frac{\vec{w}}{\|\vec{w}\|} \right) = \frac{\vec{w}\vec{x}^{+} - \vec{w}\vec{x}^{-}}{\|\vec{w}\|} = \frac{(w_0 + 1) - (w_0 - 1)}{\|\vec{w}\|} = \frac{2}{\|\vec{w}\|}$$

(учли, что для пограничных точек $\vec{x}_{+}$ и $\vec{x}_{-}$

$$\vec{w}\vec{x}_i^{+} - w_o = y_i = 1; \qquad \vec{w}\vec{x}_i^{-} - w_o = y_i = -1;$$

$$\Rightarrow \begin{cases} \vec{w}\vec{x}^{+} = w_0 + 1 \\ \vec{w}\vec{x}^{-} = w_0 - 1 \end{cases}$$

6. Дискриминантный анализ.
Линейный SVM



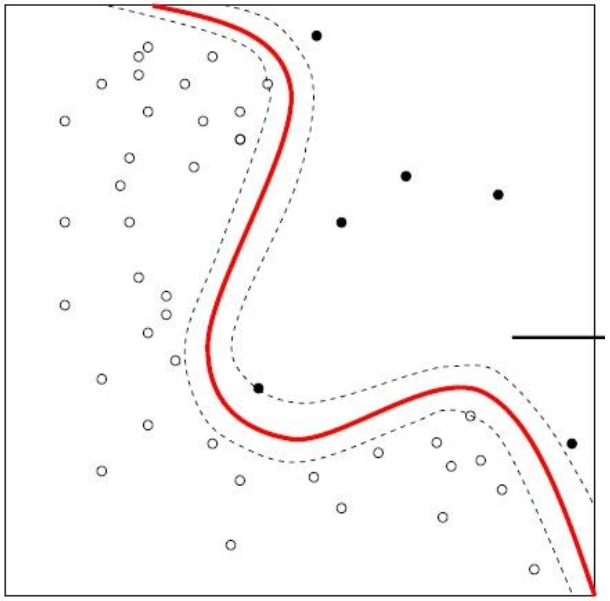
Задача квадратичного программирования

$$\begin{cases} \min \|\vec{w}\| \qquad (\min \sum_{i=1}^{n} w_i^2) \\ y_i \cdot (\vec{w}\vec{x}_i - w_o) \geq 1 \\ i = 1, 2, \ldots, t \end{cases}$$

6. Дискриминантный анализ.
Проблемы SVM

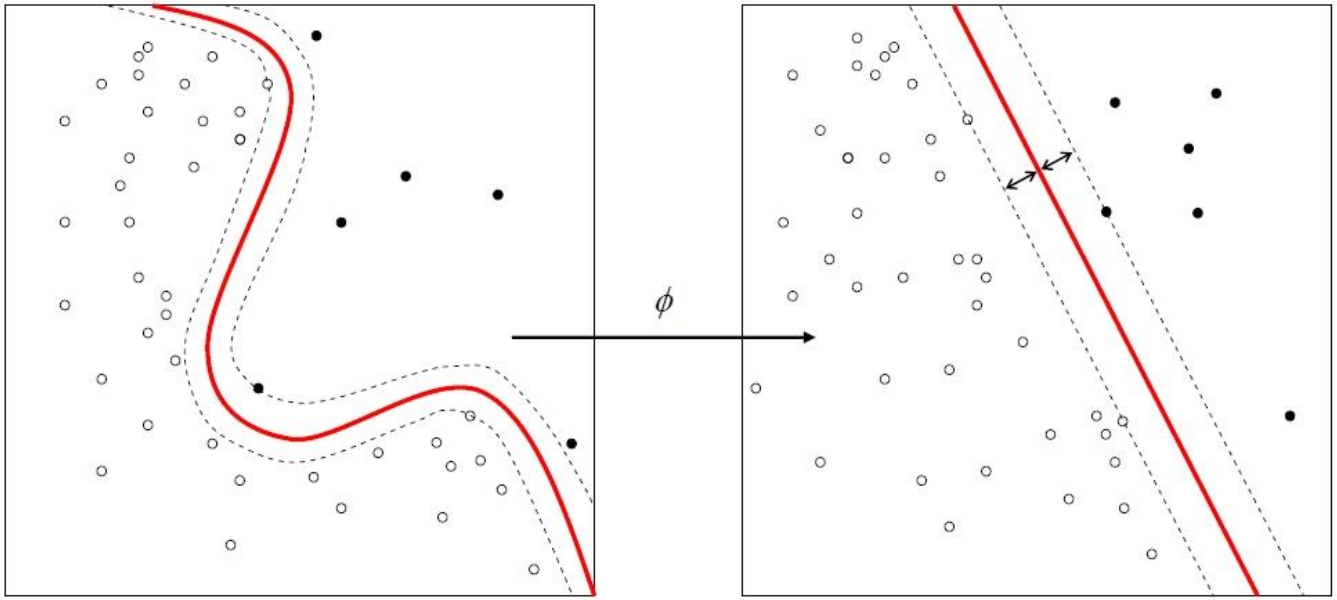
Проблемы:

- Решение задачи линейного программирования (метод, основанный на применении теоремы Куна-Таккера, неконструктивен
- Линейная неразделимость



6. Дискриминантный анализ.
Проблемы SVM

- Линейная неразделимость



6. Дискриминантный анализ.
Проблемы SVM