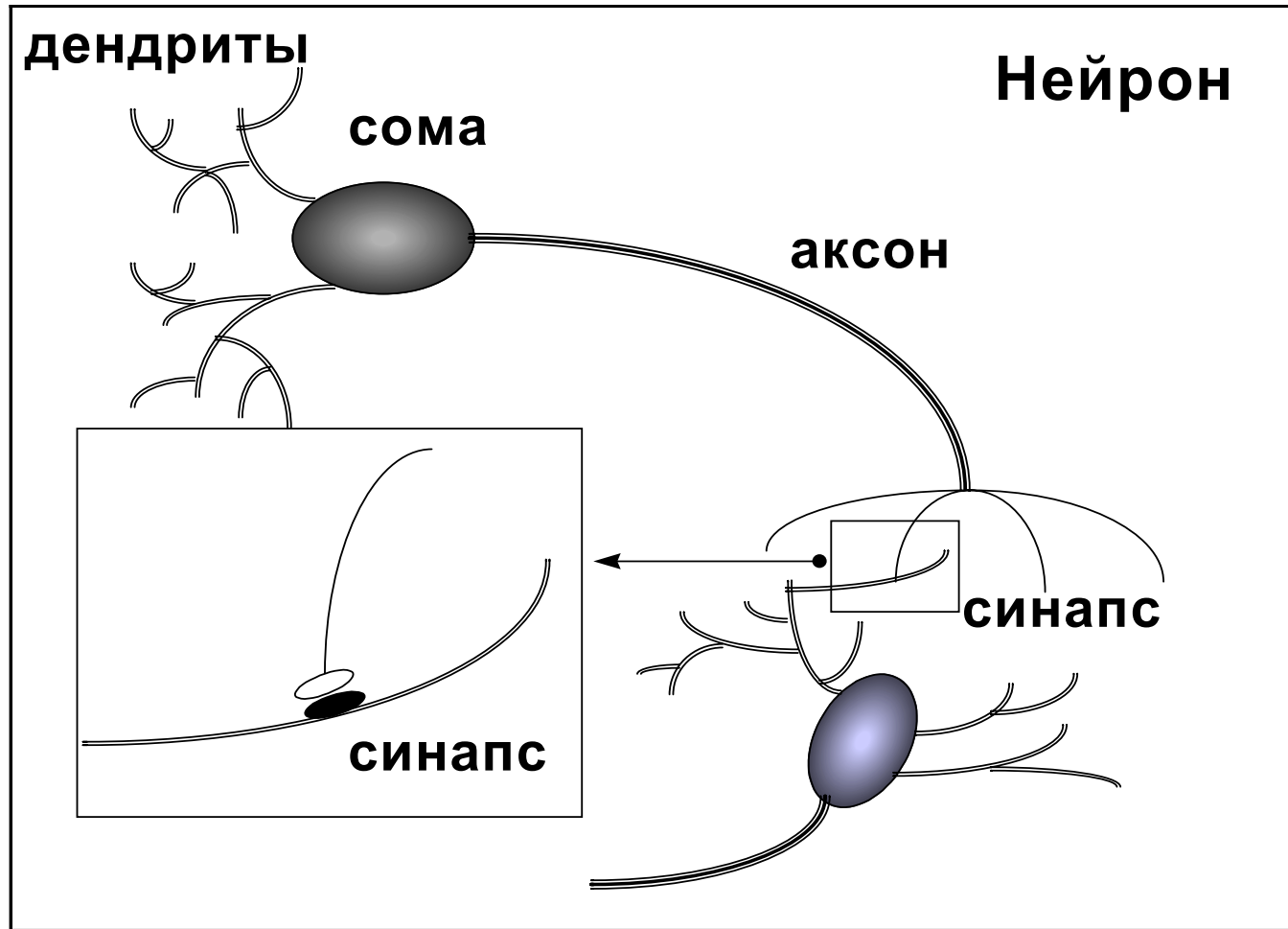


Нейронные сети

Лекция 14 АХТП

Биологический нейрон

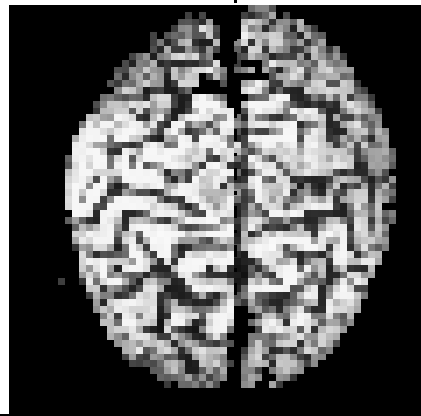


◆ Традиционные ЭВМ

- ◆ Последовательные
- ◆ Заданный алгоритм
- ◆ Иерархическая структура алгоритмов, разбиение сложной задачи на простые



*Левое
полушарие*



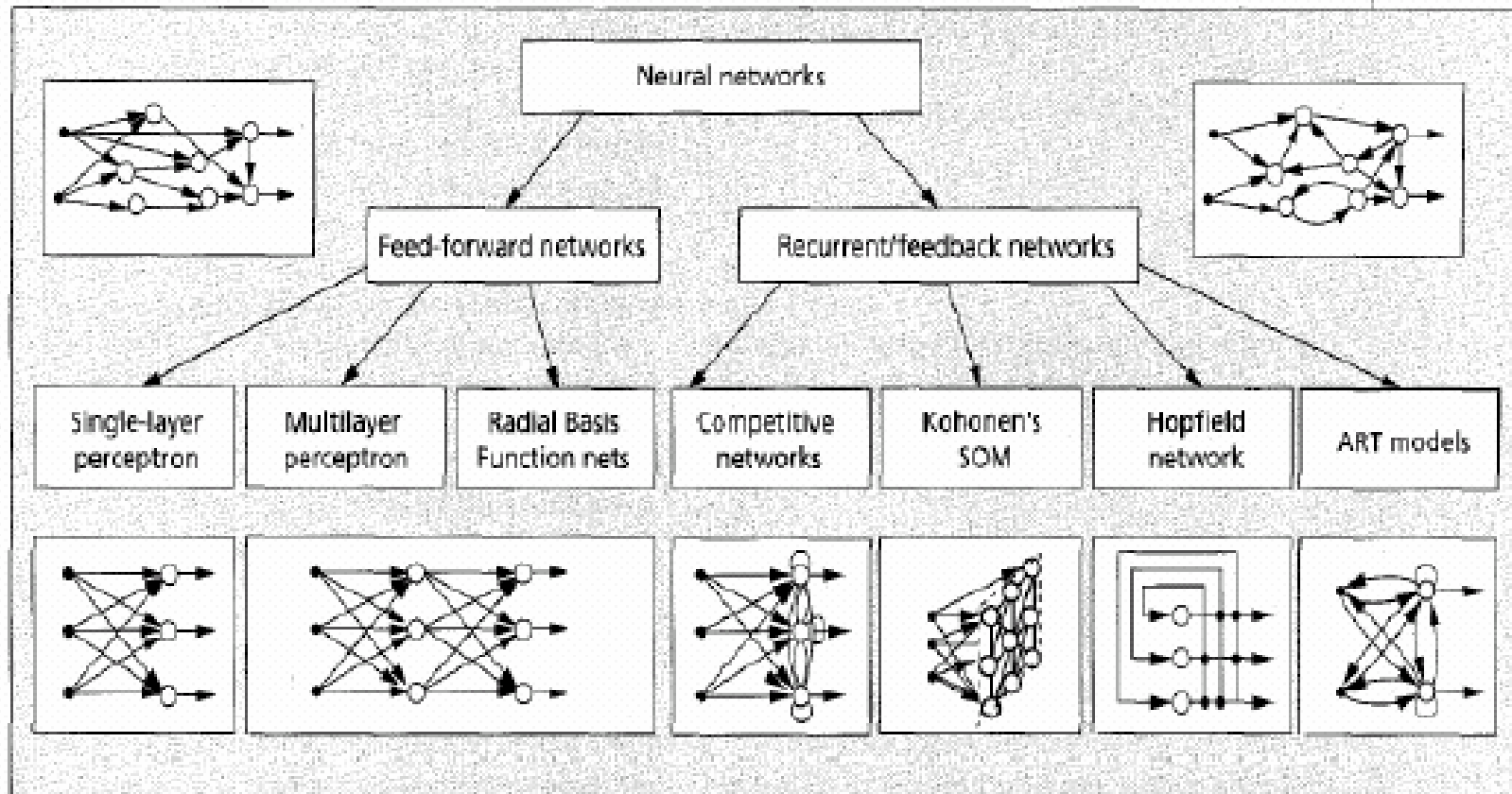
*Правое
полушарие*



◆ Нейрокомпьютеры

- ◆ Параллельные
- ◆ Алгоритм формируется путем обучения на примерах
- ◆ Непосредственные операции с образами

Виды нейронных сетей (прямого распространения и рекуррентных)



Нейронная сеть как «черный ящик»

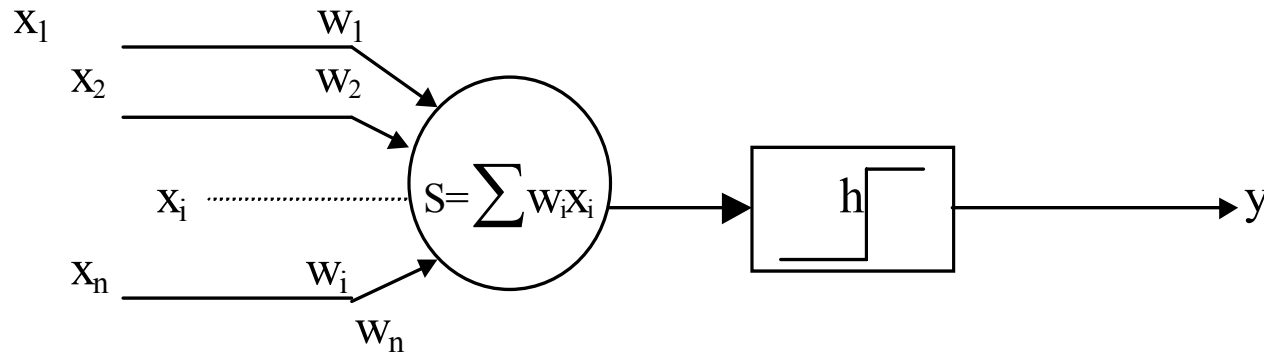
- Сигналы на входах - входной вектор
- Сигналы на выходах – выходной вектор
- НС учится отображению входов на выходы (или, другими словами, вычислению некоторой неизвестной нелинейной функции от многих переменных)

Виды обучения нейронных сетей

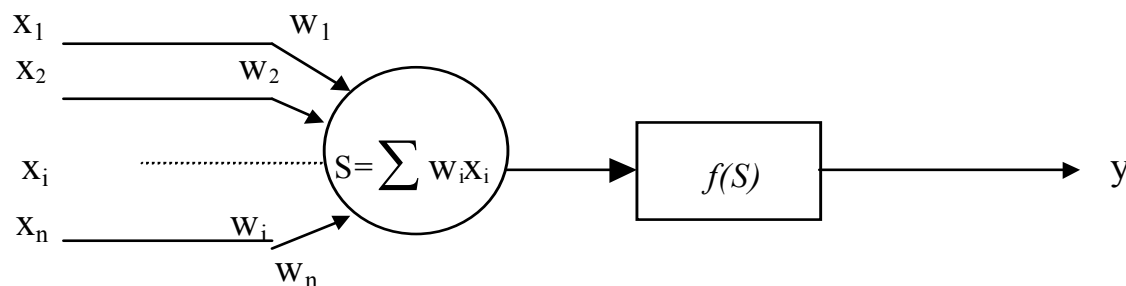
- С учителем (классификация, регрессия)
 - Пример – входной и выходной вектор
- Без учителя (кластеризация)
 - Пример – входной вектор
- С поощрением и наказанием или Reinforcement learning (обучение поведению)
 - Поведение (выход) оценивается и НС получает сигнал поощрения или наказания

Модели нейронов.

Формальный нейрон Мак-Каллока-Питтса (пороговый нейрон)



Сигмоидальный нейрон



- Рациональная сигмоида

$$f(S) = S / (|S| + a)$$

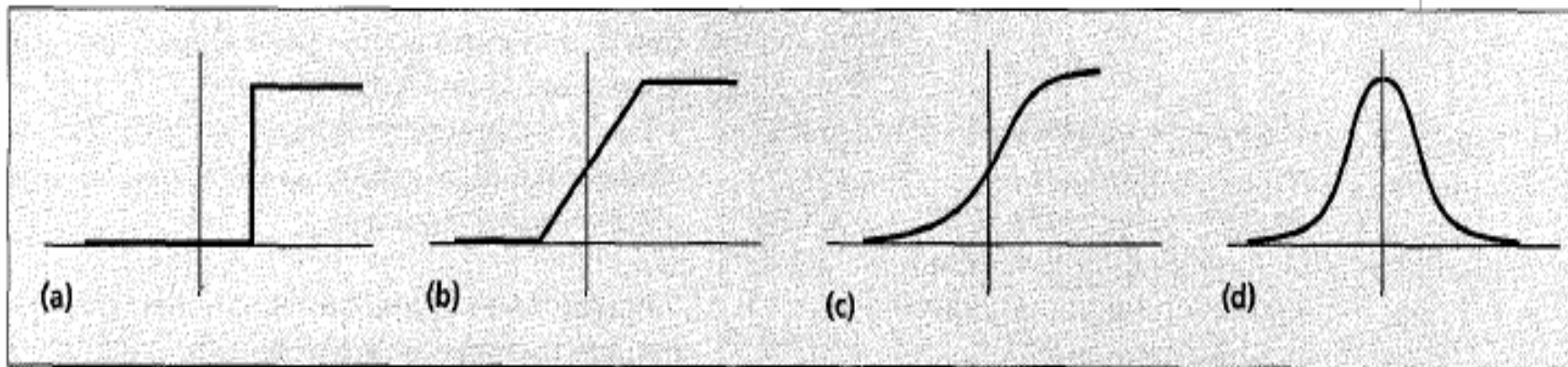
- Экспоненциальная сигмоида

$$f(S) = 1 / (1 + \exp(-aS))$$

- Гиперболический тангенс

$$f(s) = \operatorname{th} \frac{s}{\alpha} = \frac{e^{-\frac{s}{\alpha}} - e^{\frac{s}{\alpha}}}{e^{-\frac{s}{\alpha}} + e^{\frac{s}{\alpha}}}$$

Виды активационной функции



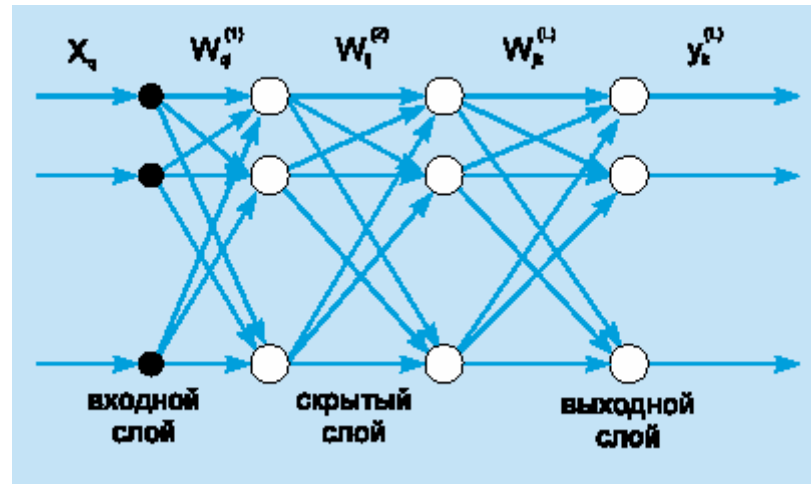
Пороговая,

Линейная,

Сигмоидная,

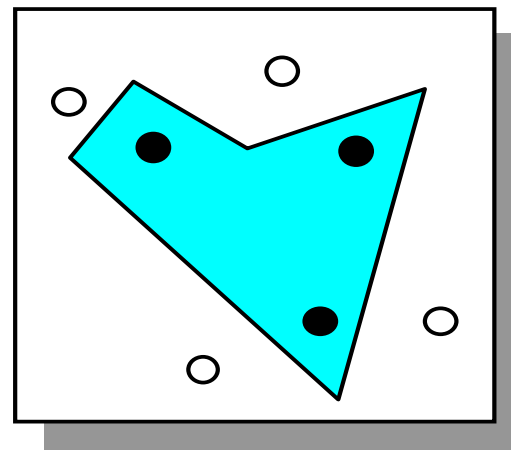
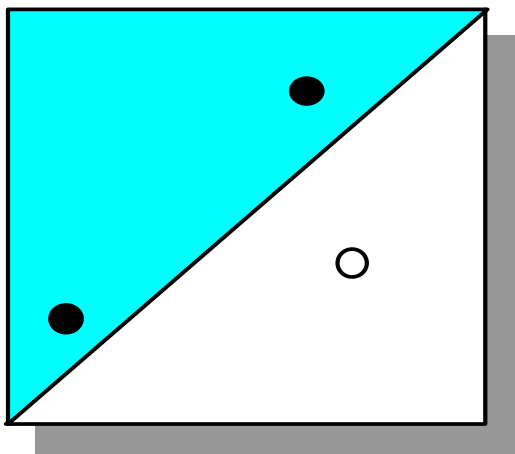
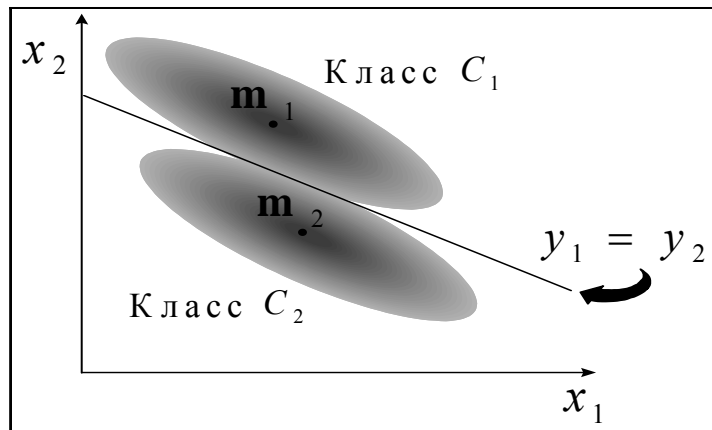
Гауссовская

Перцептроны



- **Классификация** (дискретный набор выходных значений)
- **Регрессия** (непрерывные выходные значения)

Классификация



Классификация

- **Сеть с одним скрытым слоем, содержащим H нейронов со *ступенчатой* функцией активации, способна осуществить произвольную классификацию Nd точек d -мерного пространства (т.е. классифицировать Nd примеров).**
- **Одного скрытого слоя нейронов с *сигмоидной* функцией активации достаточно для аппроксимации любой границы между классами со сколь угодно высокой точностью.**

Регрессия (Аппроксимация)

- **Одного скрытого слоя нейронов с *сигмоидной* функцией активации достаточно для аппроксимации любой функции со сколь угодно высокой точностью. (Более того, такая сеть может одновременно аппроксимировать и саму функцию и ее производные.)**

Точность аппроксимации возрастает с числом нейронов скрытого слоя.

При H нейронах ошибка оценивается как $O(1/H)$.

Алгоритм обучения нейрона Маккалока-Питтса в персептроне Розенблатта (бинарного)

- Случайно выбираются веса w_{ij}
- На входы подается обучающий вектор x и рассчитывается выходной сигнал y_i с использованием пороговой функции
- Если ожидаемое значение d_i совпадает с y_i , то веса не изменяются
- Если $y_i=0$ и $d_i=1$, то $w_{ij}(t+1)=w_{ij}(t)+x_j$
- Если $y_i=1$ и $d_i=0$, то $w_{ij}(t+1)=w_{ij}(t)-x_j$
- Повторяются шаги, начиная со второго, для новых примеров

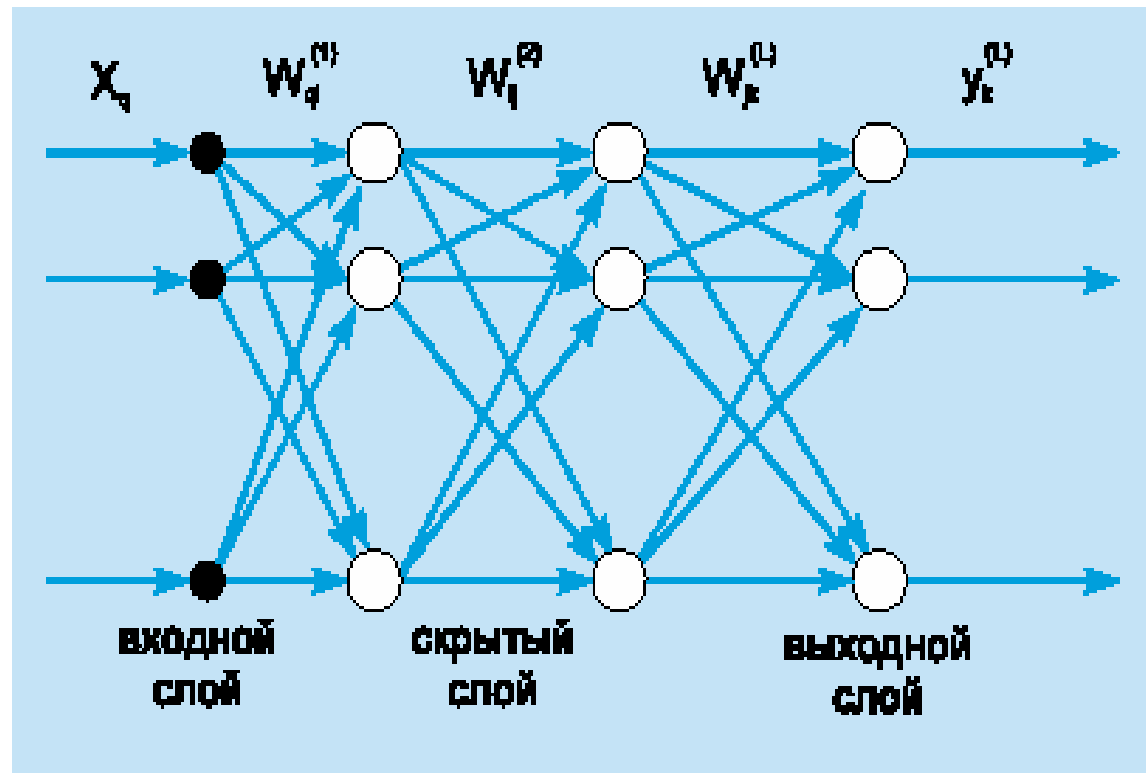
Правило обучения Видроу-Хоффа

$$w_{ij}(t+1) = w_{ij}(t) + x_j(d_i - y_i)$$

Задача минимизации целевой функции

$$E = \sum_{k=1}^p (y^{(k)}_i - d^{(k)}_i)^2$$

Многослойный персептрон (сеть прямого распространения)



Два режима работы:

- 1) Режим обучения
- 2) Режим использования обученной сети

Обучение сети многослойного персептрона

- Цикл обучения (на множестве примеров)
 - Предъявление сети очередного примера (входного и выходного векторов)
 - Вычисление фактического выходного вектора (прямое распространение сигналов)
 - Изменение весов связей с целью минимизации ошибки на выходе
- Изменение весов – итерационный процесс. Критерий останова – исчерпание лимита количества итераций или достижение заданной величины ошибки E

Обучение методом обратного распространения
ошибки (Error Back Propagation) для
многослойных персептронов с сигмоидными
функциями активации

$$w_{i,j}(t+1) = w_{i,j}(t) + r g_j x_i'$$

Для выходного слоя $g_j = y_j(1 - y_j)(d_j - y_j)$

Для других слоев $g_j = x_j'(1 - x_j') \sum_k g_k w_{jk}$

Модифицированный алгоритм

$$w_{i,j}(t+1) = w_{i,j}(t) + r g_j x_j' + \alpha (w_{i,j}(t) - w_{i,j}(t-1))$$

Обобщение

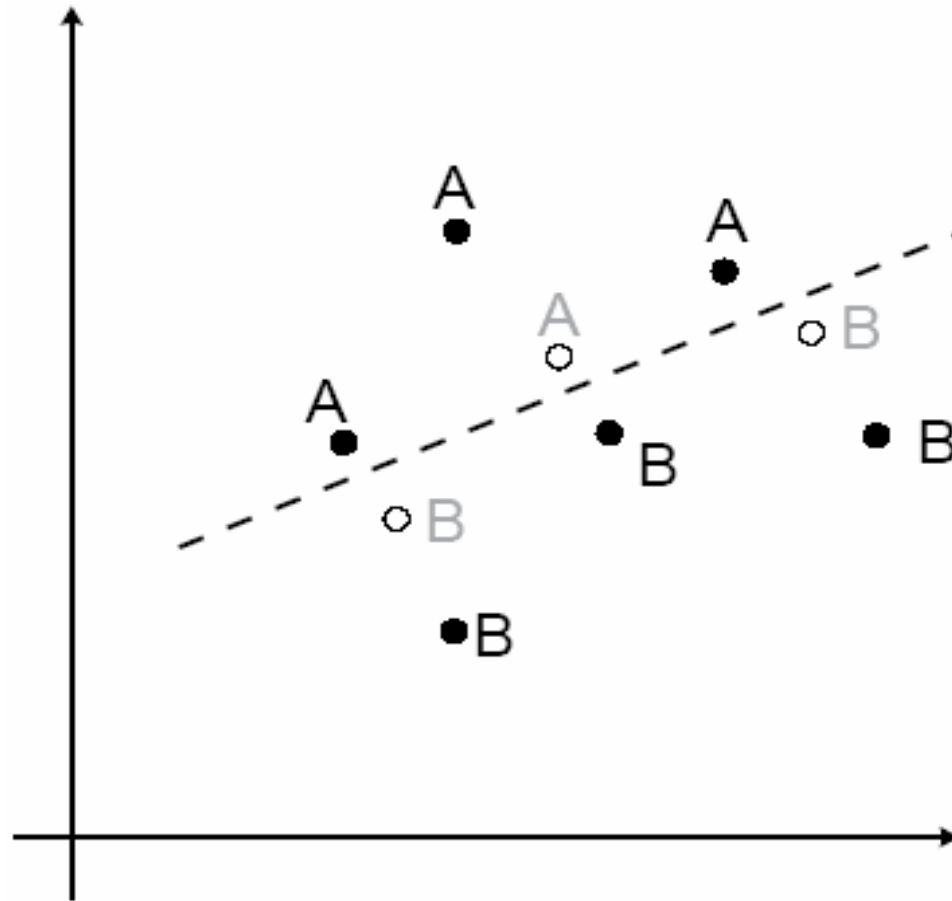
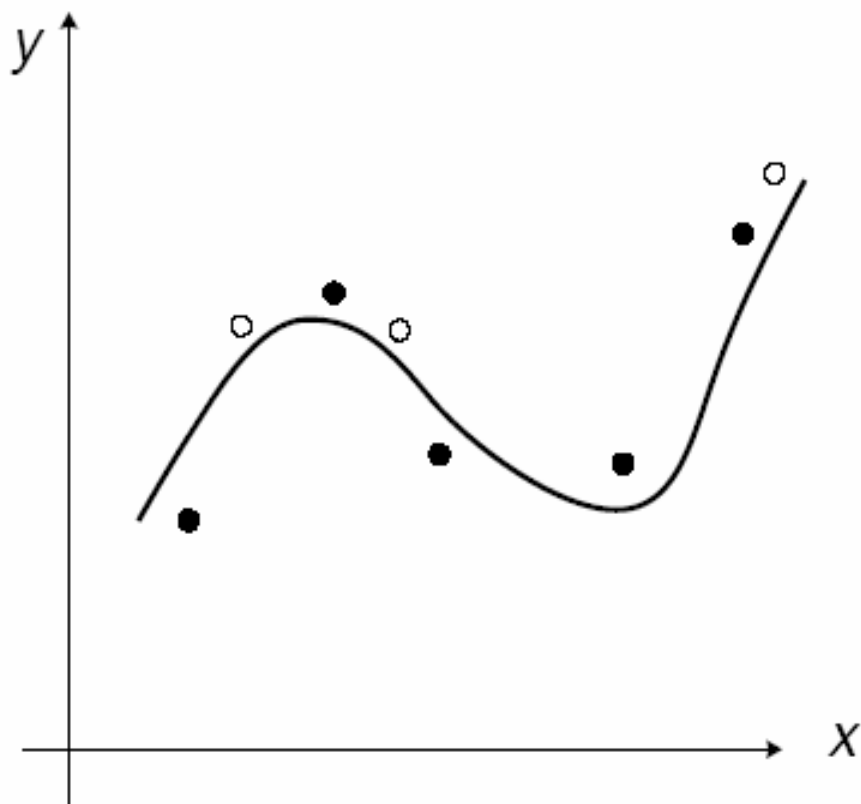
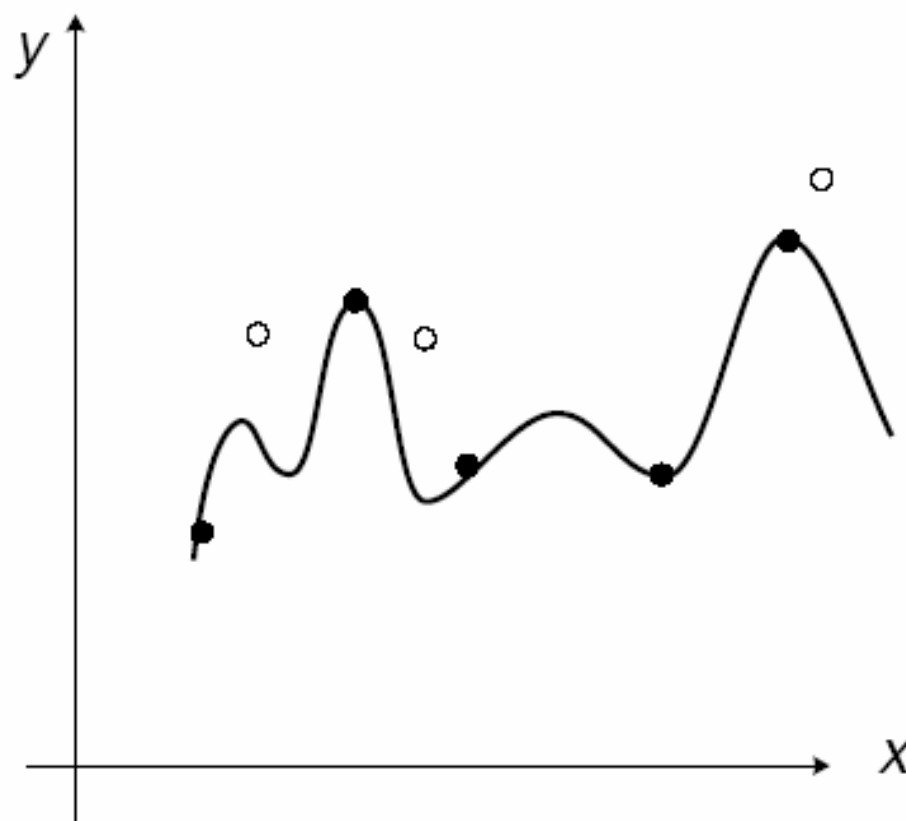


diagram sparse training - two classes

Переобучение (Overfitting)

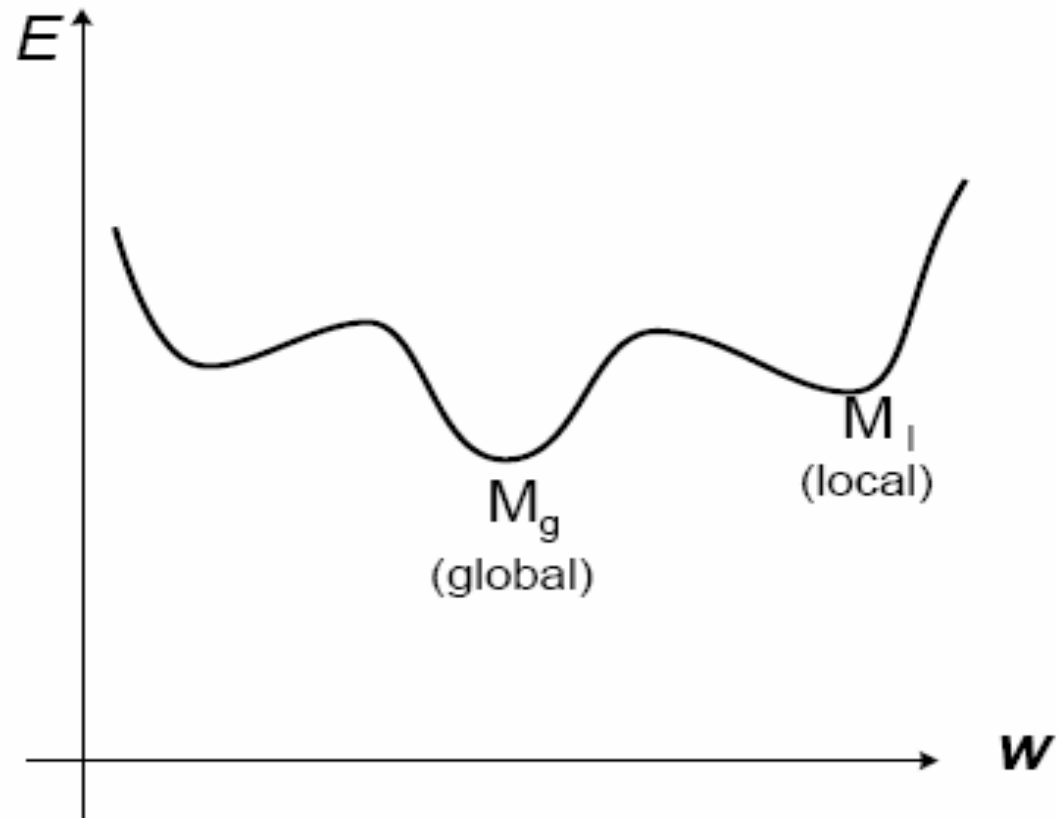


y against x for a binary classifier



overfitting in x-y space

Локальный минимум



local minimum

Достоинства и недостатки многослойного персептрона с обучением обратным распространением ошибки

- Достоинства:
 - Гарантирована возможность решения задачи
- Недостатки:
 - Низкая скорость обучения
 - Возможность переобучения
 - Невозможность переобучения
 - Необходимо выбирать структуру сети под конкретную задачу (обычно это проблема)

Различные подходы к обучению

- Обучение основанное на коррекции ошибки
 - Используется ошибка (d-y) между желаемым и реальным выходом для изменения весов связей с целью уменьшения ошибки
- Вероятностное обучение (машина Больцмана)
 - Адаптация весов видимых нейронов так, чтобы состояния видимых нейронов соответствовали желаемому распределению вероятностей
- Правило Хебба – усиливаются связи между нейронами с похожими (или одинаковыми) состояниями и наоборот

$$w_{ij}(t + 1) = w_{ij}(t) + \eta y_j(t) x_i(t)$$

- Соревновательное обучение
 - Только один нейрон (победитель) в выходном слое активизируется (*winner-take-all*, «победитель получает все»)

Подходы к решению задач с использованием нейронных сетей

- Использование консультационную фирму-исполнителя для решения задачи
- Использование оболочку, ориентированную на решение определенного типа задач
 - Metastock, Predictor, Deductor
- Использование универсальную оболочку
 - Toolbox of MATLAB, Statistica Neural Networks, Brain Maker, Neural Planner, Neural Bench,
- Разработка программной модели нейронной сети как специализированной программы (с использованием или без каких-либо компонентов для программирования нейронных сетей – SNNS, NeuralBase)
- Разработка программной модели как компоненты какого-либо прикладного ПО
 - G2
- Программирование аппаратной реализации нейронной сети