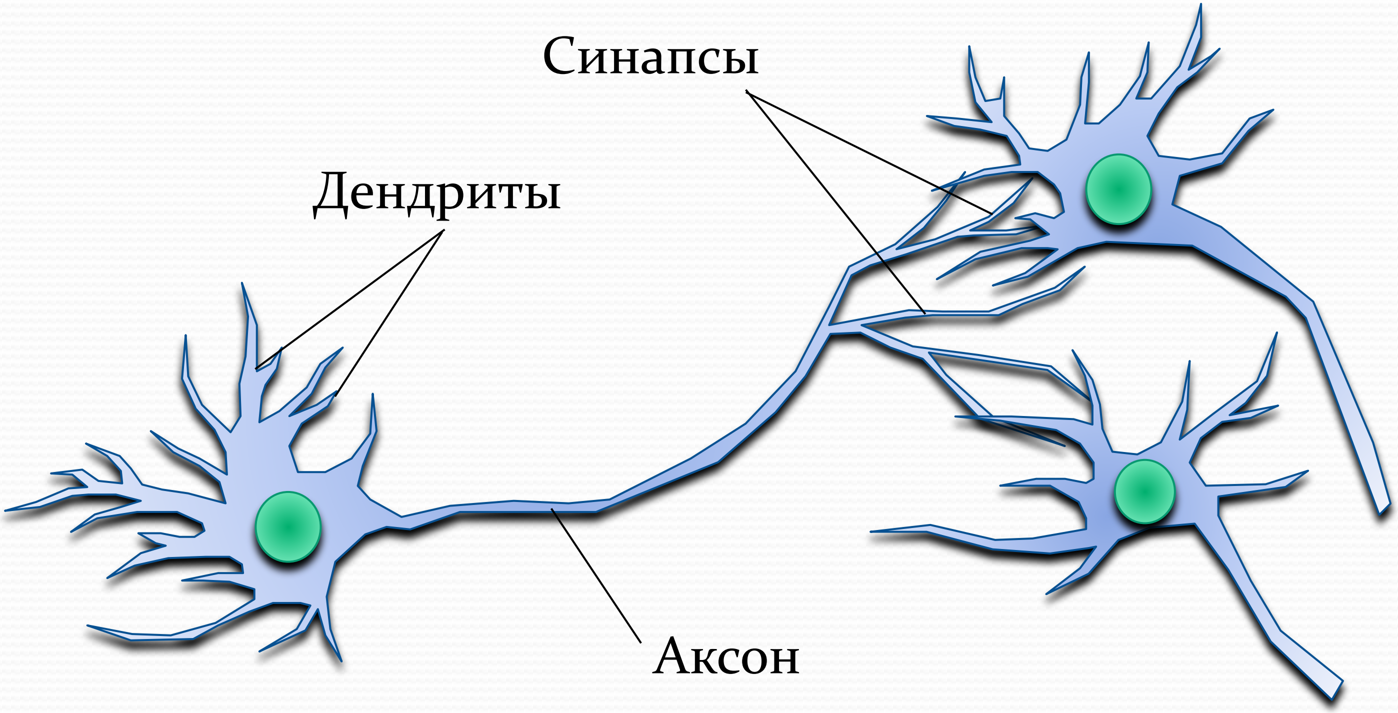
НЕЙРОИНФОРМАТИКА

1. Нейрон
2. Математический нейрон Мак-Каллока – Питтса
3. Персептрон Розенблатта
4. Многослойный персептрон
5. Задачи, решаемые с помощью нейросетей
6. Невербальность и "шестое чувство" нейросетей
7. Методы обучения нейросетей
8. Подготовка входных параметров
9. Рекуррентные сети

1. Нейрон

Нейронные сети и **нейрокомпьютеры** – это одно из направлений компьютерной индустрии, в основе которого лежит идея создания искусственных интеллектуальных устройств по образу и подобию человеческого мозга.

Нейроны человеческого мозга



- простейший нейрон может иметь до 10 000 дендритов
- человеческий мозг содержит примерно 10^{11} нейронов
- каждый нейрон связан с 10^{3}-10^{4} другими нейронами
- мозг человека содержит 10^{14}-10^{15} взаимосвязей

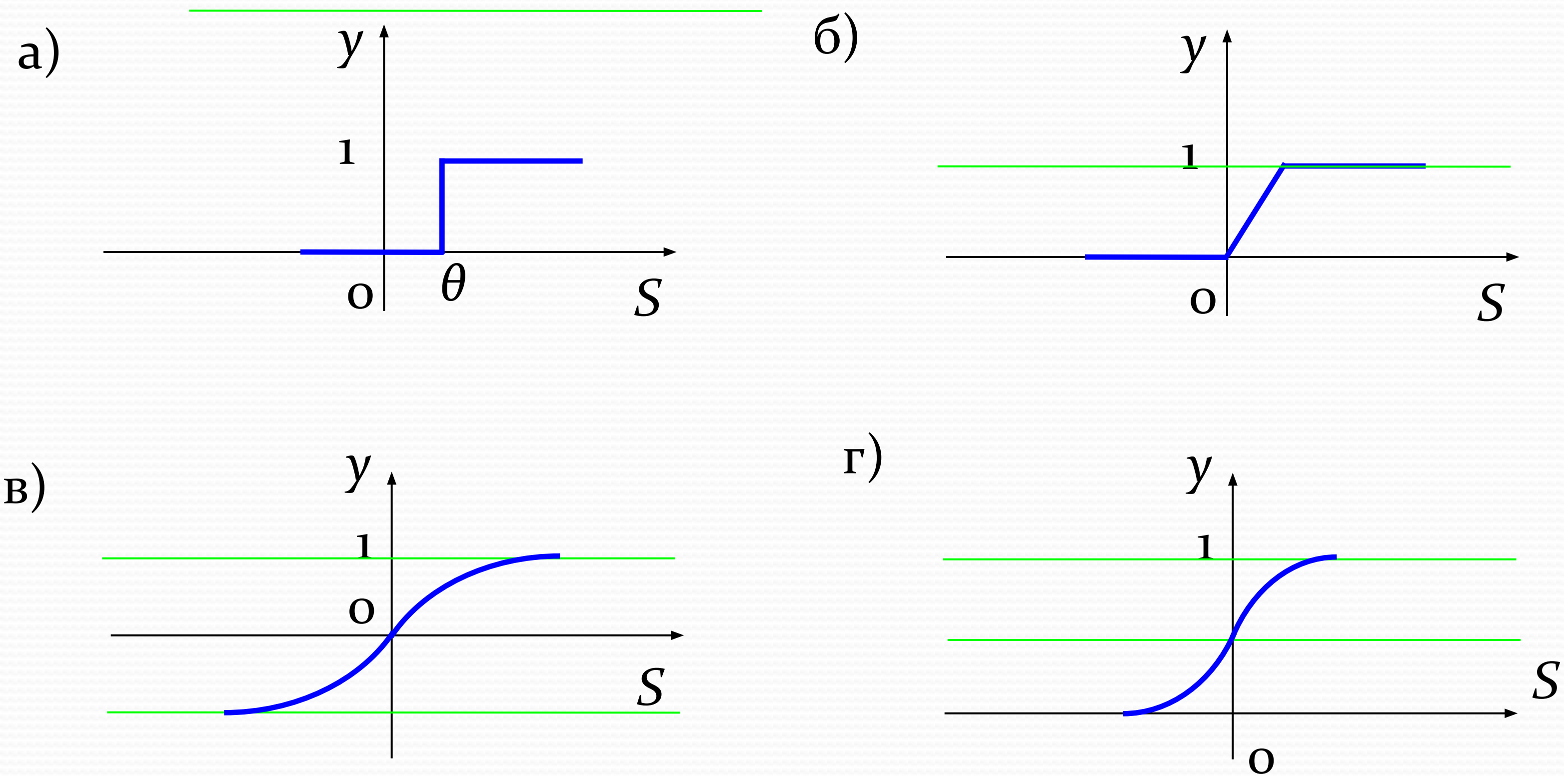
2. Математический нейрон Мак-Каллока – Питтса

- 1943 г. Уоррен Мак-Каллок и Вальтер Питтс выдвинули гипотезу математического нейрона
- математический нейрон имеет несколько входов и один выход
- через входы (j) математический нейрон принимает входные сигналы (x_j)
- входные сигналы умножаются на весовой коэффициент (w_j) и суммируются

$$S = \sum_{j=1}^{J} w_j x_j$$

- выходной сигнал нейрона *y* является нелинейная функция *f*(*S*), которая называется активационной

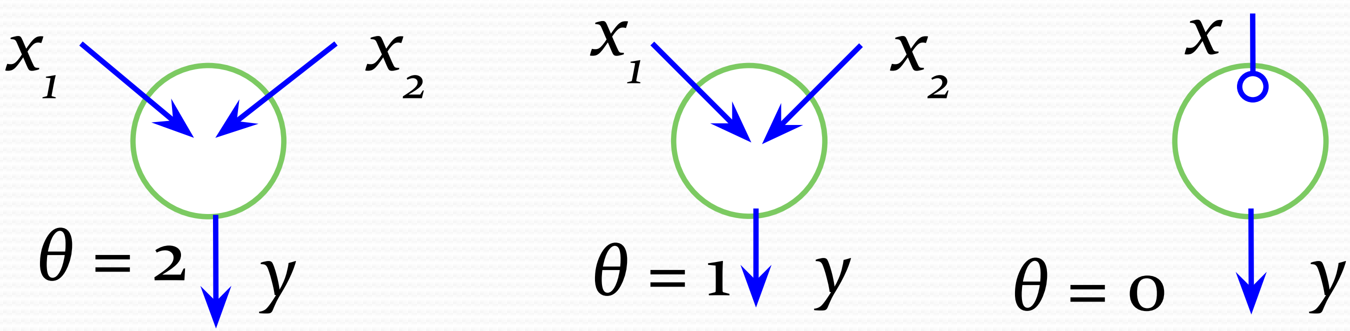
Активационная функция



а) функция единого скачка (θ – порог чувствительности нейрона); б) линейный порог (гистерезис); в) и г) сигмоид (логистическая функция).

Математические нейроны, реализующие логические функции

- математический нейрон изображают кружочком
- возбуждающий вход – стрелкой
- тормозящий вход – маленьким кружочком
- рядом может записываться число, показывающее значение порога θ

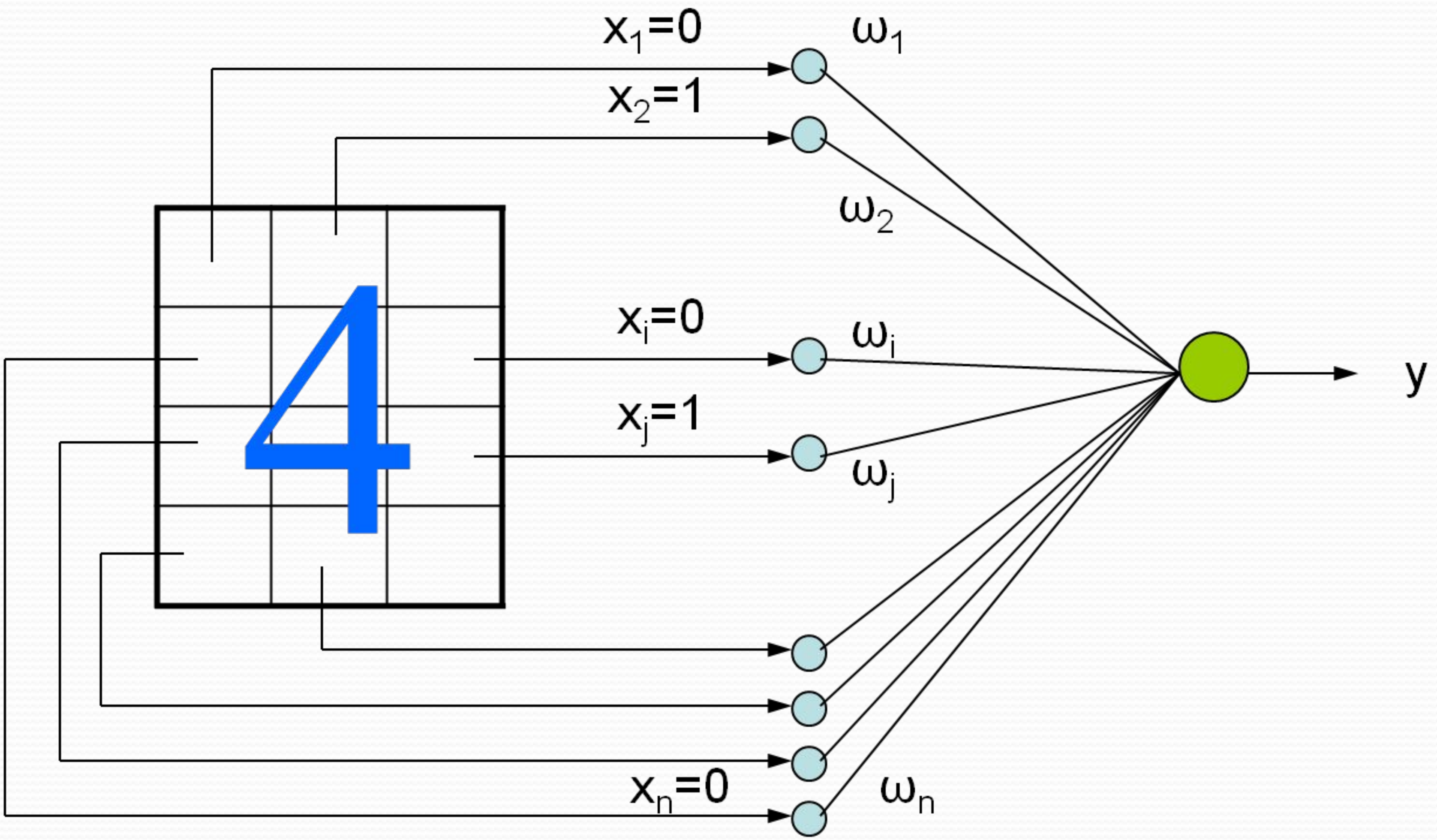


"И" "ИЛИ" "НЕТ"

3. Персептрон Розенблатта

Идея Мак-Каллока – Питтса была реализована Фрэнком Розенблаттом:

- 1958 г. в виде компьютерной программы
- 1960 г. в виде электронного устройства, моделирующего человеческий глаз



$x_1=0$
ω_1
$x_2=1$
ω_2
4
$x_i=0$
ω_i
y
$x_j=1$
ω_j
$x_n=0$
ω_n

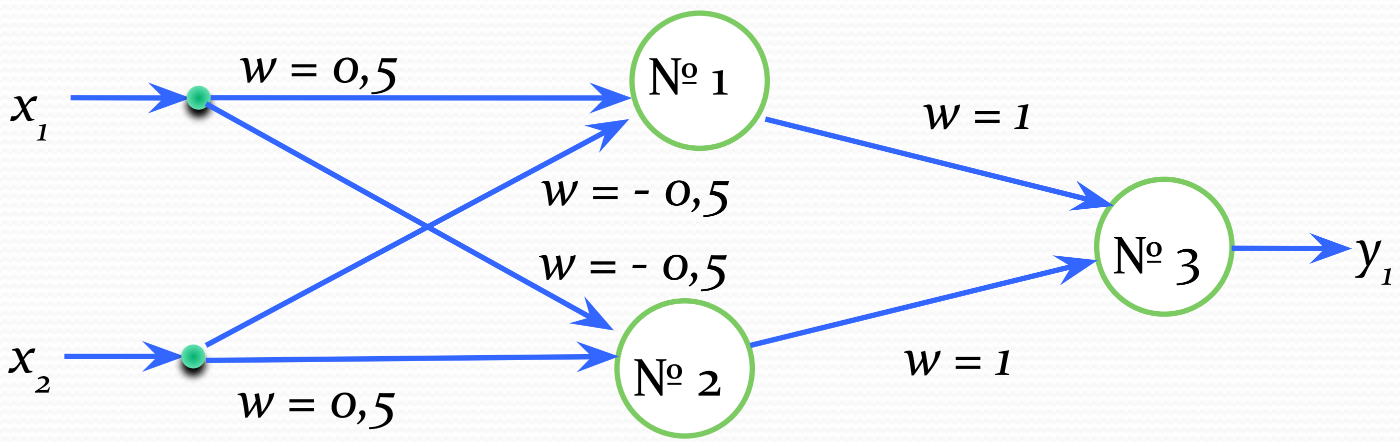
Итерационный алгоритм корректировки весовых коэффициентов

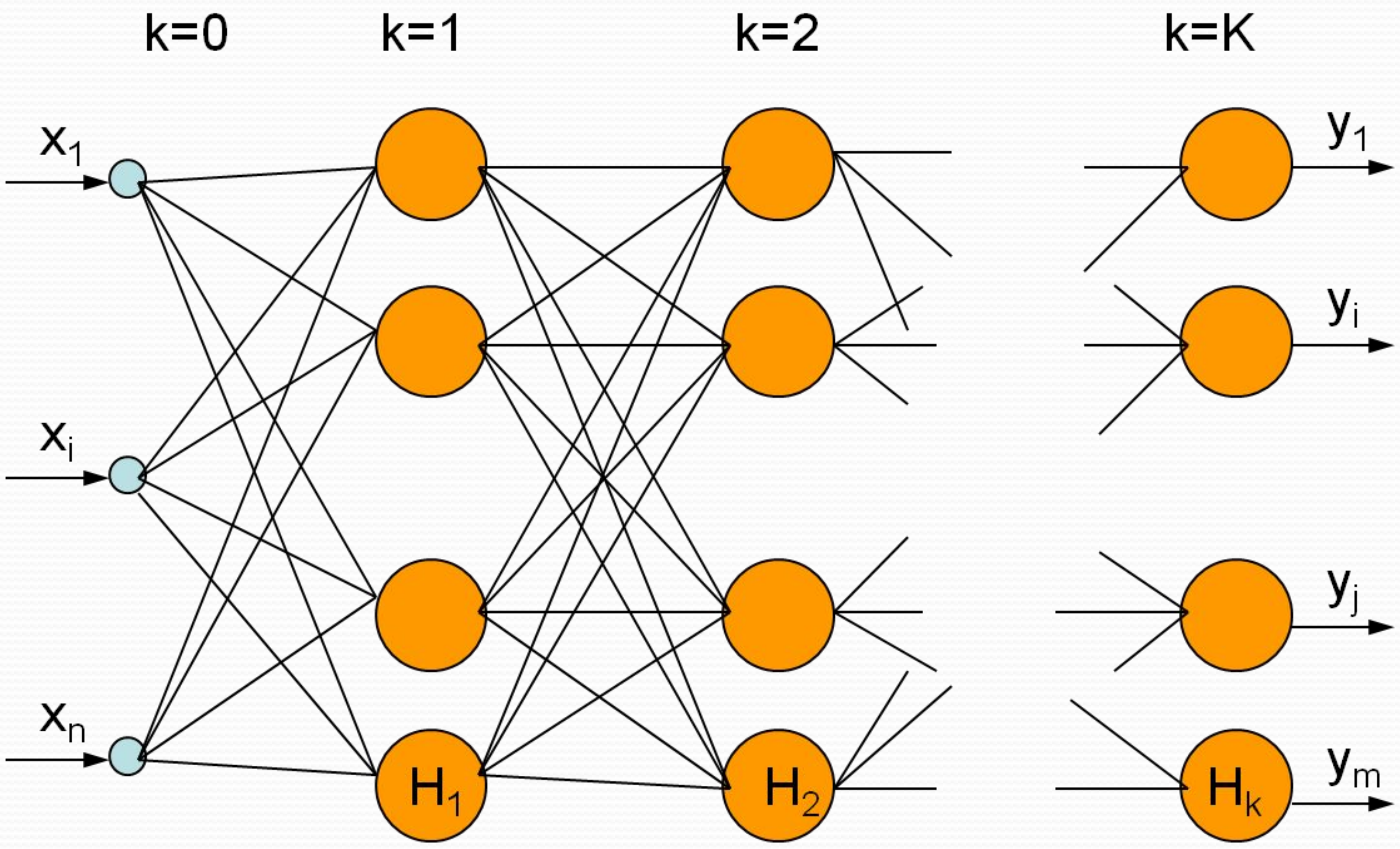
1. Шаг 1. Подать входной образ и вычислить выход персептрона *y*
2. Шаг 2, а. Если выход правильный, то перейти на шаг 1
3. Шаг 2, б. Если выход неправильный и равен нулю, то увеличить веса активных входов, например:
$$w_j(t+1) = w_j(t) + x_j$$
4. Шаг 2, в. Если выход неправильный и равен единице, то уменьшить веса активных входов, например:
$$w_j(t+1) = w_j(t) - x_j$$
5. Шаг 3. Перейти на шаг 1 или завершить процесс обучения.

4. Многослойный персептрон

- М. Минский и С. Пайперт в своей книге "Персептроны" строго математически доказали, что однослойные персептроны в принципе не способны решать многие простые задачи
- Многие понимали, что надо усложнять структуру персептронов

Двухслойный персептрон, реализующий функцию "Исключающее ИЛИ"




k=0
k=1
k=2
k=K
x_1
x_i
x_n
H_1
H_2
H_k
y_1
y_i
y_j
y_m

Алгоритм обратного распространения ошибки

- Шаг 1. Инициализация синаптических весов и смещений
- Шаг 2. Представление из обучающей выборки входного вектора $X_q = (x1, x2, \ldots, x_N)_q$ и соответствующего ему выходного вектора $D_q = (d1, d2, \ldots, d_M)_q$
- Шаг 3. Прямой проход

$$y_i^{(k)} = f_\sigma\left(j\sum_{j=0}^{H_{k-1}} w_{ij}^{(k)} y_j^{(k-1)} \right)$$

- Шаг 4. Обратный проход

$$w_{ij}^{(k)}(t+1) = w_{ij}^{(k)}(t) + \Delta w_{ij}^{(k)}(t+1)$$

$$\Delta w_{ij}^{(k)}(t+1) = \eta \delta_i^{(k)} y_j^{(k-1)}$$

$$\delta_i^{(K)} = (d_i - y_i) y_i (1 - y_i)$$

$$\delta_i^{(k)} = y_i^{(k)}\left(1 - y_i^{(k)}\right)\sum_{l=1}^{H_{k+1}} \delta_l^{(k+1)} w_{li}^{(k+1)}$$

- Шаг 5. Повторение шагов 2–4 необходимое число раз

- на 5 шаге алгоритма вычисляется среднеквадратичная ошибка, усредненная по всем обучающим примерам:

$$\varepsilon = \frac{1}{Q}\sum_{q=1}^{Q}\sum_{i=1}^{M}(d_i - y_i)^2$$

- также вычисляется максимальная разность между желаемым и фактическим выходами персептрона:

$$\varepsilon = \max\left(\left|d_i - y_i\right|\right)_q$$

- итерационный процесс заканчивается после того, как погрешность ε, достигнет заданной величины, либо при достижении предельного числа эпох обучения.

Задачи, решаемые с помощью нейросетей

- если есть математическая модель какого-то процесса, то изучая влияние входных параметров на выходные, можно решить задачу оптимизации моделируемого процесса
- если математическая модель является нестационарной, то её можно использовать для решения задач прогнозирования
- если математическая модель работает в реальном режиме времени, то результаты математического моделирования могут быть оперативно переданы оператору, управляющему объектом, или могут быть непосредственно введены в приборы, что позволяет решать задачи управления моделируемым объектом или процессом
- нейронные сети могут решать задачи распознавания и классификации образов, причем под образами понимаются зрительные изображения, символы, тексты, запахи, звуки, шумы

Невербальность и "шестое чувство" нейросетей

Как и человеческий мозг, нейросеть способна выводить закономерности, делать догадки, открывать законы природы. Но, так же, как и человек, нейросеть не способна к чёткой формулировке алгоритма, позволившего сделать то или иное умозаключение.

Известны случаи, когда нейросети демонстрируют феномен, называемый в жизни шестым чувством. Они с успехом извлекают знания из анализа информации, из которой, казалось бы, эти знания извлечь невозможно.

Методы обучения нейросетей

детерминистские

- подстройка весов представляет собой жёсткую последовательность действий

стохастические

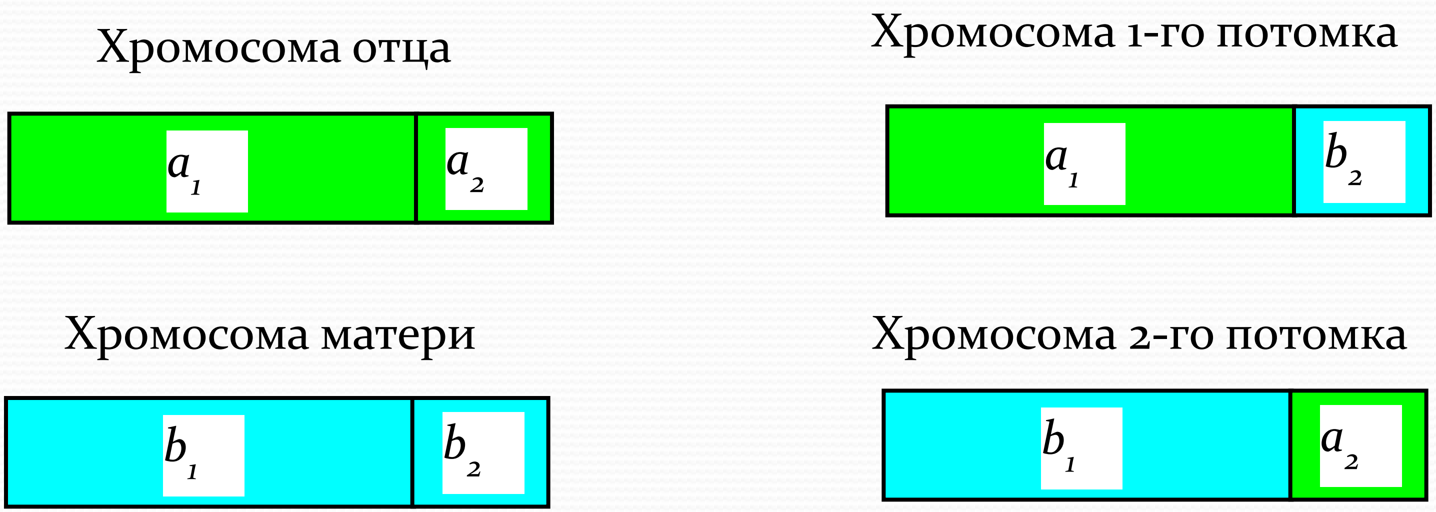
- подстройка весов производится на основе действий, подчиняющихся некоторому случайному процессу

- основная проблема обучения персептронов состоит в том, что поверхность функции ошибок обычно имеет очень сложную форму со множеством локальных минимумов.
- актуальным является развитие методов **глобальной оптимизации**, которые позволяют найти глобальный минимум многоэкстремальной целевой функции

Генетический алгоритм

- предложен Дж. Холландом в 1970-х годах
- имитирует природный оптимизационный процесс, происходящий при эволюции живых организмов
- основные идеи теории Чарльза Дарвина: естественный отбор и генетическое наследование
- мутация – изменение генов

Операция скрещивания, применяемая в генетических алгоритмах



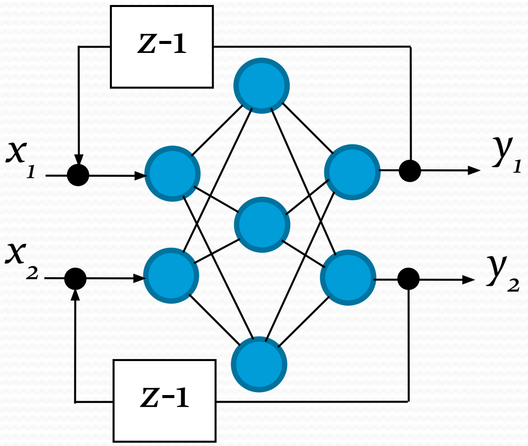
Подготовка входных параметров

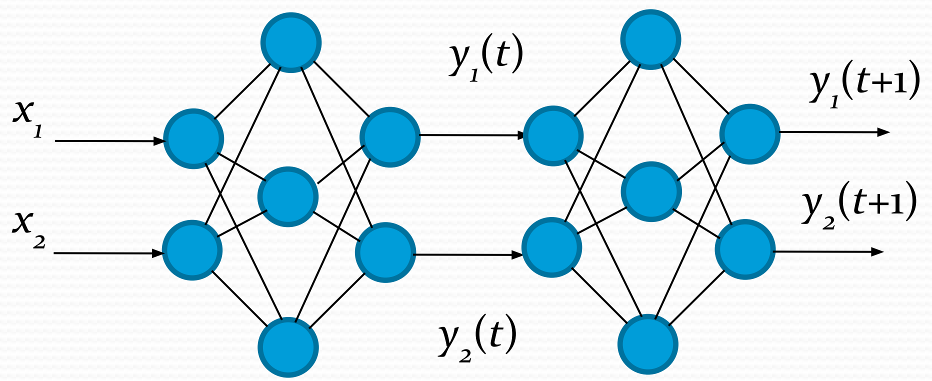
- успех создания нейронной сети во многом зависит от удачного подбора обучающих примеров
- следует учитывать, что не все параметры предметной области влияют на выходной вектор Y
- незначимые параметры не следует включать в список параметров входного вектора X
- на первом этапе в вектор X включают как можно больше параметров

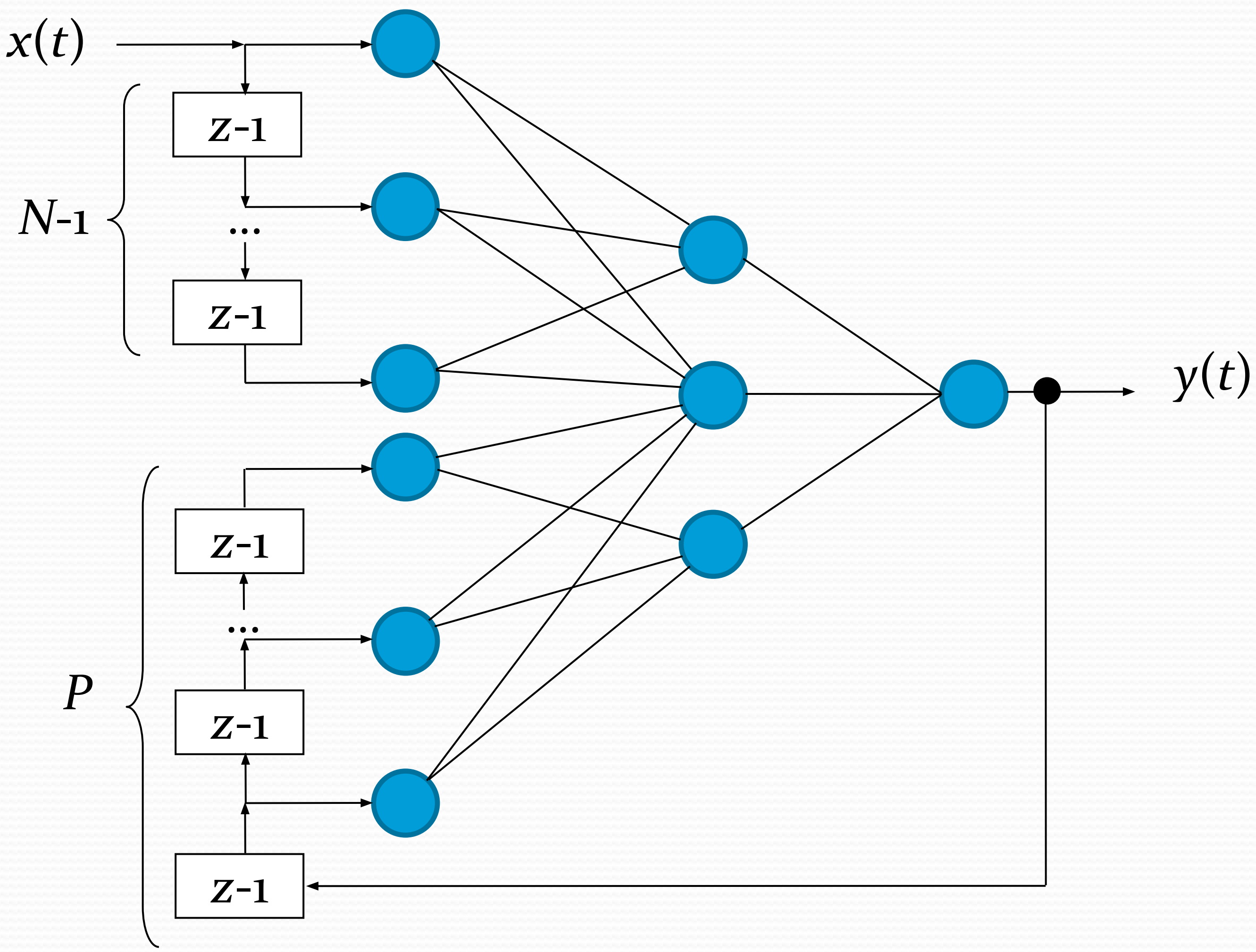
Определение незначимых параметров

1. **анализа значений весовых коэффициентов входных нейронов**. Если у какого-либо входного нейрона синаптические веса значительно меньше, чем у других нейронов, то этот входной нейрон скорее всего соответствует незначимому параметру вектора X
2. **возмущения значений входных параметров и анализа реакции сети на эти возмущения**. Если сеть не реагирует или слабо реагирует на изменения значения какого-либо входного параметра, то этот параметр не является значимым

Рекуррентные сети







$x(t)$
N-1
z-1
z-1
$y(t)$
P
z-1
z-1
z-1