

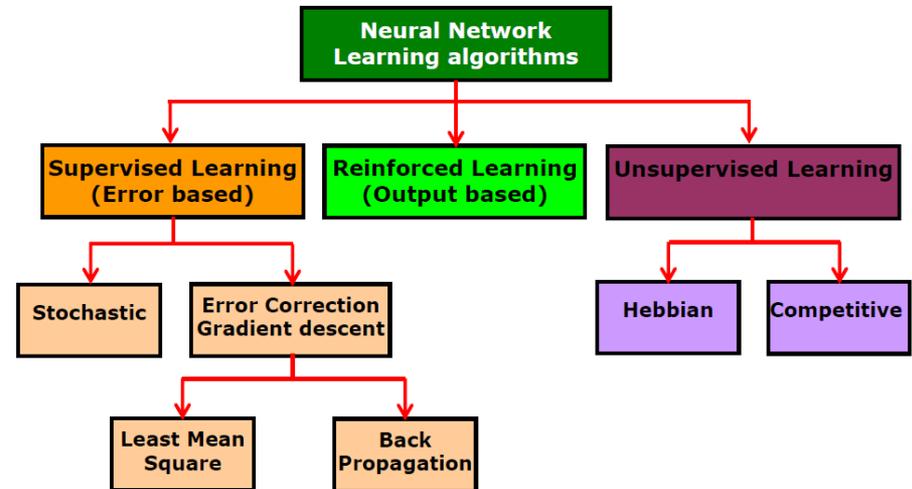
# Нейронные сети и их практическое применение.

## Лекция 4. Алгоритм обратного распространения ошибки.

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

# Классификация алгоритмов обучения

1. Обучение с учителем (Supervised learning). Обучающая выборка содержит входные сигналы, и ожидаемые выходы. Цель подбора весов – близость реальных и ожидаемых выходов сети.
2. Обучение без учителя (Unsupervised learning). Нет знаний о желаемых выходах сети.
  - конкуренция нейронов между собой:
    - Winner takes all (WTA);
    - Winner takes most (WTM).
  - учет корреляции обучающих и выходных сигналов (обучение по Хеббу).
3. Reinforcement learning. Имеется только информация о правильности выхода сети



# Обучение по Хеббу

1. Если два нейрона, соединенных связью, активизируются одновременно, то вес связи возрастает; если – неодновременно, то вес уменьшается.

2. Математическая модель  $\Delta w_{kj}(n) = F(y_k(n), x_j(n)),$

3. Гипотеза Хебба  $\Delta w_{kj}(n) = \eta y_k(n) x_j(n)$

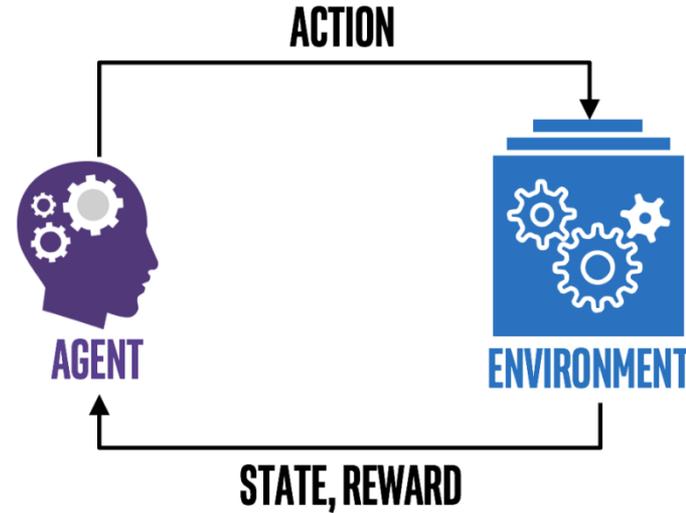
4. Гипотеза ковариации  $\Delta w_{kj} = \eta (x_j - \bar{x})(y_k - \bar{y})$

# Обучение с подкреплением

Обучение с подкрепление

(Reinforcement learning – RL) – обучение агента выбору действий в заданном окружении так, чтобы максимизировать поощрение.

- агент – программа, которая выполняет действия – объект обучения;
- окружение – известные условия, в которых действует агент;
- действия – множество действий, которые может выполнить агент, приводящие к изменению окружения;
- поощрение – награда (положительная или отрицательная) за выполненные действия



# Обучение с подкреплением.

## Примеры

### ❑ Размещение рекламных баннеров на веб-странице.

- агент – программа, принимающая решение, сколько баннеров разместить на странице
- окружение – веб-страница
- действия – {добавить, удалить, ничего не делать}
- поощрение – положительное при увеличении дохода, отрицательное – иначе.

### ❑ Управление роботом.

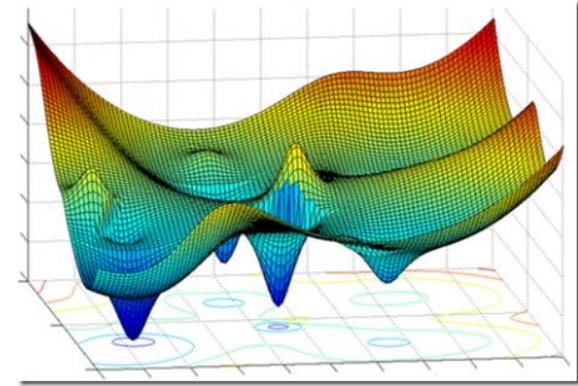
- агент – программа, управляющая роботом;
- окружение – мир, в котором робот перемещается;
- действия – {вперед, назад, влево, вправо}
- поощрение – положительное при достижении пункта назначения, отрицательное при потере времени, выборе неправильного направления или падении.

# Обучение с подкреплением. Свойства

- ❑ Поиск динамических паттернов в данных.
  - обучение с учителем/без учителя: статические паттерны
- ❑ Правильный ответ изначально неизвестен
  - обучение с учителем: ответ есть в обучающей выборке
- ❑ Построение последовательности решений
  - обучение с учителем: один пример - одно решение

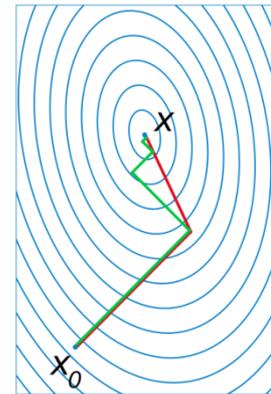
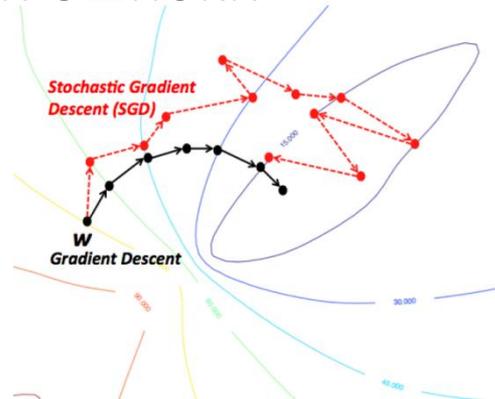
# Алгоритм обратного распространения ошибки (1)

- ❑ Функция ошибки  $E = F(d, y(x, w))$   
 $x$  – известные входные вектора;  
 $d$  – требуемые выходные значения сети;  
 $y(x, w)$  – вычисленные выходные значения сети;  
 $w$  – вектор весов сети.



- ❑ Поиск минимума функции ошибки

$$\hat{w} = \arg \min_w F(d, y(x, w))$$

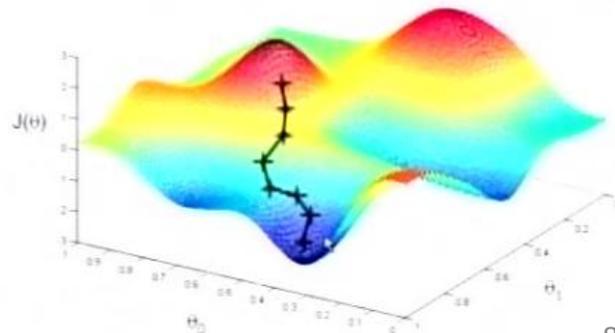


# Алгоритм обратного распространения ошибки (2)

1. Инициализировать веса сети
2. Выбрать обучающую пару из обучающего множества.
3. Подать на вход сети, вычислить выход
4. Вычислить разность между реальным выходом и требуемым выходом
5. Подкорректировать веса сети, так чтобы уменьшить ошибку
6. Повторять шаги 2-5 для каждого вектора из обучающего множества, пока ошибка не достигнет приемлемого уровня

Шаги 2, 3 – проход вперед

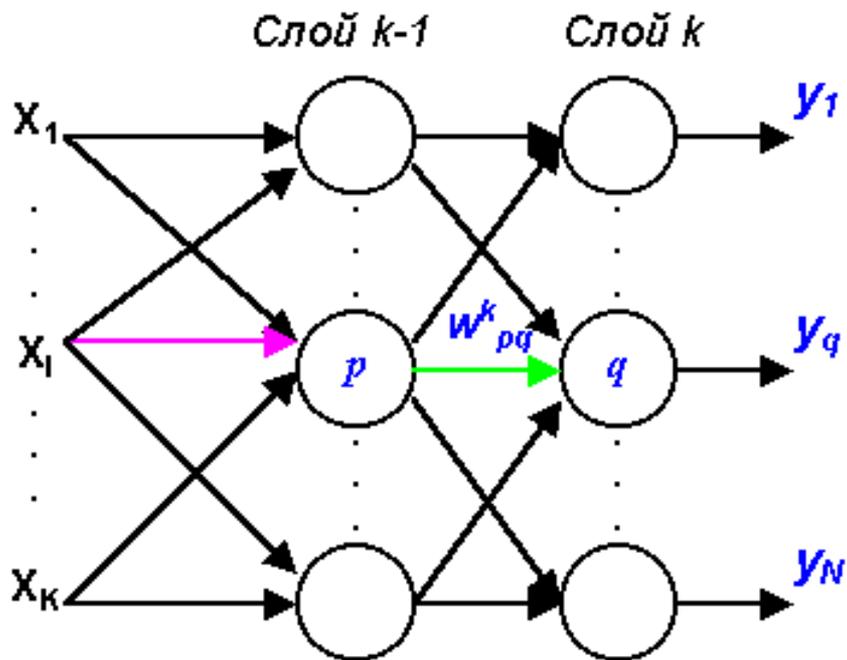
Шаги 4, 5 – обратный проход



# Алгоритм обратного

## распространения ошибки (3)

Необходимое условие: дифференцируемость функции активации.



Выходной слой

$$\delta_q = \frac{dy_q}{du_q} (y_q - d_q)$$

Скрытый слой

$$\delta_p^{k-1} = \left[ \sum_q \delta_q^k \cdot w_{pq}^k \right] \cdot \frac{dv_p^{(k-1)}}{du_p^{(k-1)}}$$

$$\Delta w_{pq}^k = \eta \delta_q^k v_p^{k-1}$$

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

# Алгоритм обратного

## распространения ошибки (4)

Минимизация целевой функции :  $E(w) = \frac{1}{2} \sum_j (y_j - d_j)^2$

$d$  - желаемый выход

$y$  - реальный выход

Метод обучения - градиентный спуск.

$$\Delta w_{lp}^{(k-1)} = -\eta \cdot \frac{\partial E}{\partial w_{lp}^{(k-1)}} \quad \frac{\partial E}{\partial w_{lp}^{(k-1)}} = \frac{\partial E}{\partial v_p^{(k-1)}} \cdot \frac{dv_p^{(k-1)}}{du_p^{(k-1)}} \cdot \frac{\partial u_p^{(k-1)}}{\partial w_{lp}^{(k-1)}}$$

$$\frac{\partial E}{\partial v_p^{(k-1)}} = \sum_q \frac{\partial E}{\partial v_q^{(k)}} \cdot \frac{dv_q^{(k)}}{du_q^{(k)}} \cdot \frac{\partial u_q^{(k)}}{\partial v_p^{(k-1)}} = \sum_q \frac{\partial E}{\partial v_q^{(k)}} \cdot \frac{dv_q^{(k)}}{du_q^{(k)}} \cdot w_{pq}^{(k)} \quad \delta_q^k = \frac{\partial E}{\partial v_q^{(k)}} \cdot \frac{dv_q^{(k)}}{du_q^{(k)}}$$

$$\delta_p^{k-1} = \left[ \sum_q \delta_q^k \cdot w_{pq}^k \right] \cdot \frac{dv_p^{(k-1)}}{du_p^{(k-1)}}$$

Выходной слой

$$\delta_i = (y_i - d_i) \cdot \frac{dy_i}{du_i}$$

$$\Delta w_{pq}^k = -\eta \delta_q^k v_p^{k-1}$$

# Алгоритм обратного распространения ошибки (5)

1. Рассчитать выходы сети.

2. Рассчитать для выходного слоя:

$$\delta_i = (y_i - d_i) \cdot \frac{dy_i}{du_i} \quad \Delta w_{pi}^N = -\eta \delta_i v_p^{N-1}$$

3. Рассчитать для всех остальных слоев:

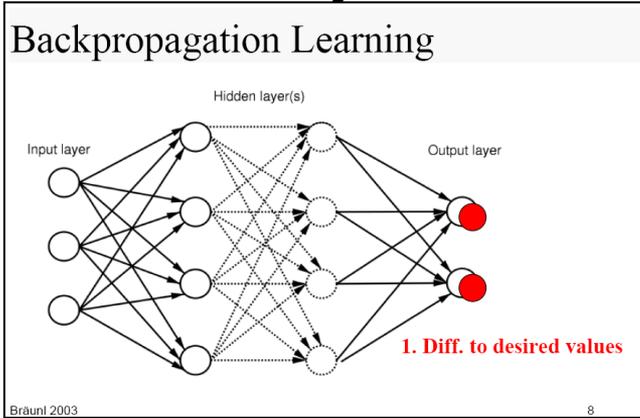
$$\delta_p^{k-1} = \left[ \sum_q \delta_q^k \cdot w_{pq}^k \right] \cdot \frac{dv_p^{(k-1)}}{du_p^{(k-1)}} \quad \Delta w_{pq}^k = -\eta \delta_q^k v_p^{k-1}$$

4. Скорректировать веса:

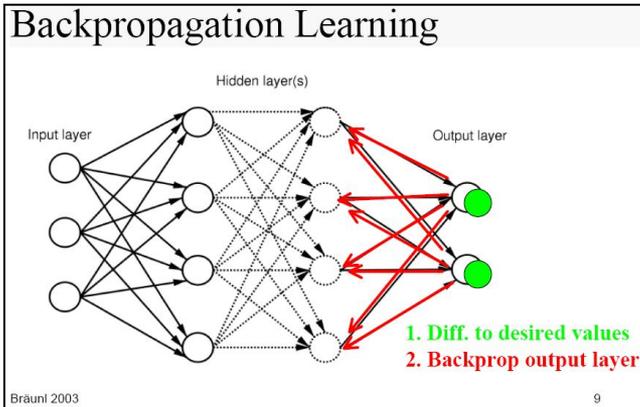
$$w_{pq}^k(t+1) = w_{pq}^k(t) + \Delta w_{pq}^k$$

5. Если ошибка существенна, перейти на шаг 1.

# Визуализация. Выходной слой



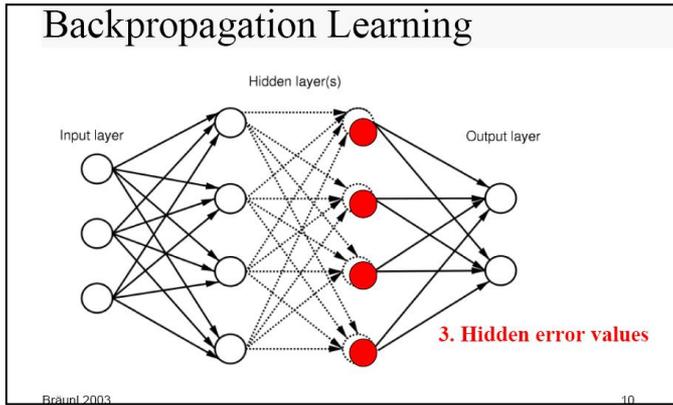
$$\delta_i = (y_i - d_i) \cdot \frac{dy_i}{du_i}$$



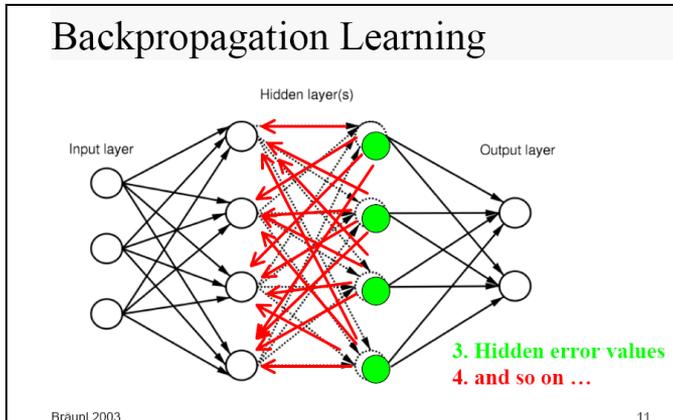
$$\delta_p^{N-1} = \left[ \sum_i \delta_i \cdot w_{pi}^N \right] \cdot \frac{dv_p^{(N-1)}}{du_p^{(N-1)}}$$

$$\Delta w_{pi}^N = -\eta \delta_i v_p^{N-1}$$

# Визуализация. Внутренний слой



$$\delta_p^{k-1} = \left[ \sum_q \delta_q^k \cdot w_{pq}^k \right] \cdot \frac{dv_p^{(k-1)}}{du_p^{(k-1)}}$$

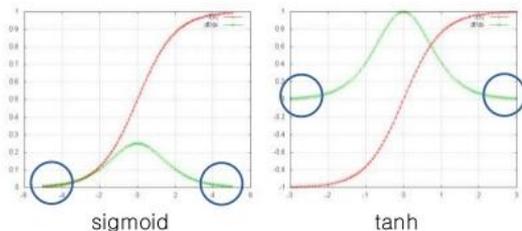


$$\delta_p^{k-2} = \left[ \sum_q \delta_q^{k-1} \cdot w_{pq}^{k-1} \right] \cdot \frac{dv_p^{(k-2)}}{du_p^{(k-2)}}$$

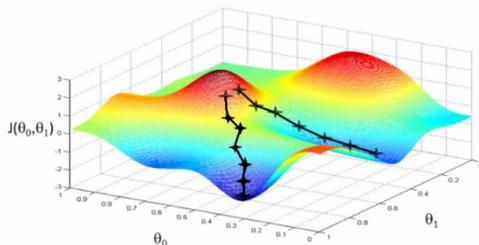
$$\Delta w_{pq}^{k-1} = -\eta \delta_q^{k-1} v_p^{k-2}$$

# Проблемы алгоритма обратного распространения ошибки

1. Градиент стремится к 0.

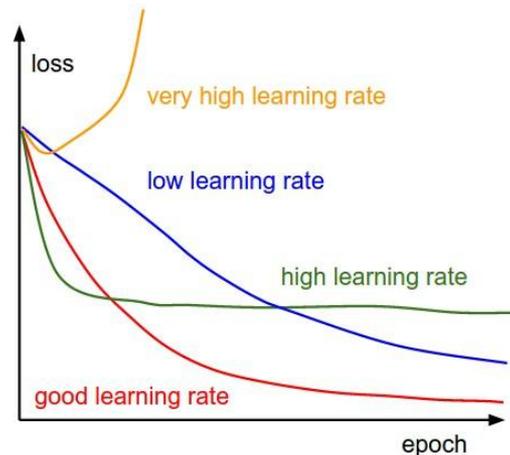


2. Обнаружение локального минимума.



3. Выбор скорости обучения.

4. Исчезающий градиент.



# Использование момента

Один из способов ускорить сходимость алгоритма обратного распространения ошибки.

$$\Delta w_{pq}^k(n) = -\eta \delta_q^k(n) v_p^{k-1}(n) + \alpha \Delta w_{pq}^k(n-1)$$

Сделаем рекуррентные подстановки.

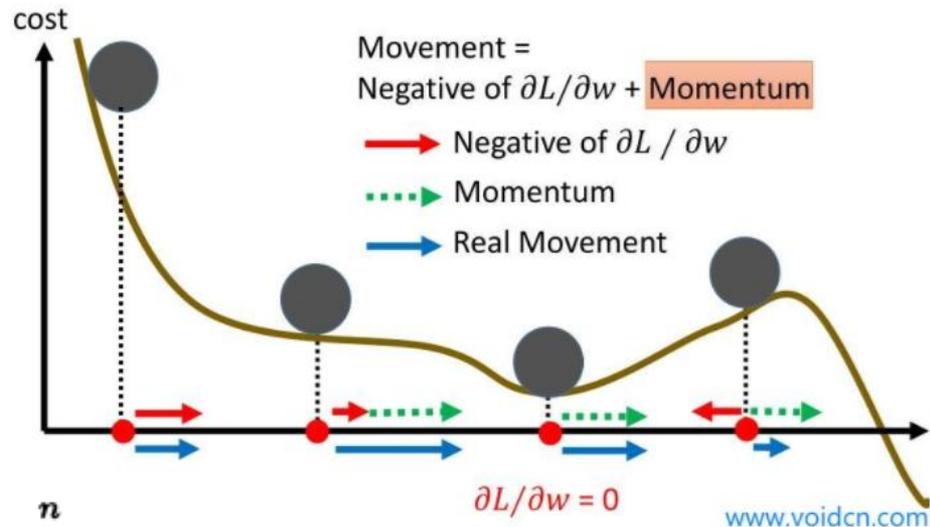
Введем ряд по  $t$ .

Решим разностное уравнение.

$$\Delta w_{ji}(n) = \eta \sum_{t=0}^n \alpha^{n-t} \delta_j(t) y_i(t).$$

$$\delta_i(n) y_i(n) = -\partial \mathbf{E}(n) / \partial w_{ii}(n)$$

$$\Delta w_{ji}(n) = -\eta \sum_{t=0}^n \alpha^{n-t} \frac{\partial \mathbf{E}(t)}{\partial w_{ji}(t)}$$



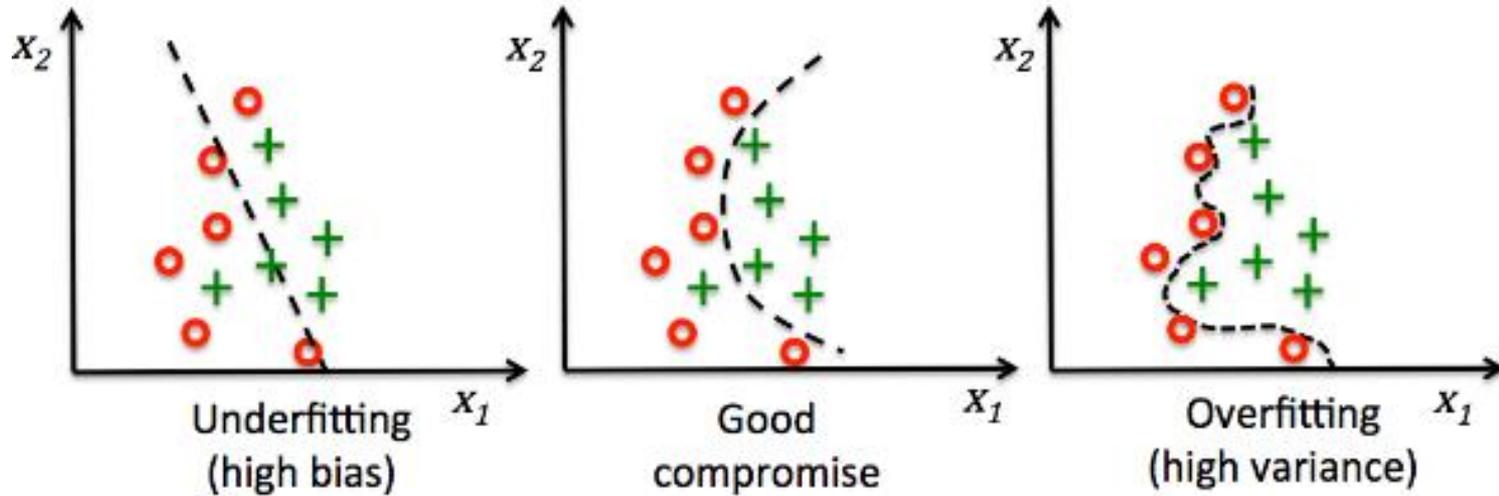
# Использование момента (2)

$$\Delta w_{ji}(n) = -\eta \sum_{t=0}^n \alpha^{n-t} \frac{\partial \mathbf{E}(t)}{\partial w_{ji}(t)}$$

1. Для сходимости ряда необходимо  $0 \leq |\alpha| < 1$
2. Ускоряет спуск, если знак градиента не меняется
3. Замедляет спуск, если знак градиента изменяется – стабилизирующий эффект.

Помогает избежать локальных минимумов

# Проблема переобучения НС



Необходимо контролировать процесс обучения и регулировать число связей в НС.

# Обучающая и подтверждающая выборки

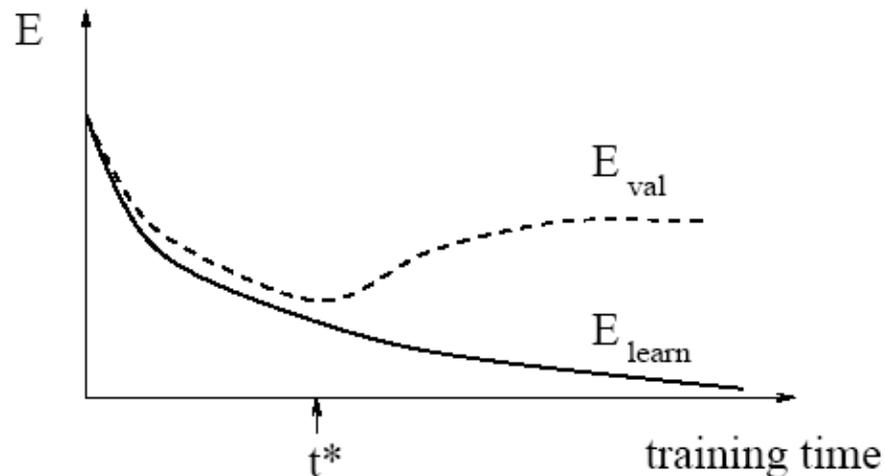
$S_L$  – обучающая выборка;

$E_{\text{learn}}$  – ошибка на обучающей выборке;

$S_V$  – подтверждающая выборка;

$E_{\text{val}}$  – ошибка на подтверждающей выборке;

Необходимо, чтобы  $E_{\text{learn}}$  и  $E_{\text{val}}$  в конце обучения достигли минимума



# Регуляризация

$$E(w) = \frac{1}{2} \sum_j (y_j - d_j)^2 + \lambda \{ \text{penalty} \}$$

- метод Weight Decay;

$$\text{penalty} = \frac{1}{2} \sum (w_{pq}^k)^2$$

$$\Delta w_{pq}^k = -\eta \delta_q^k v_p^{k-1} - \lambda w_{pq}^k$$

- метод Weight Elimination

$$\text{penalty} = \sum \frac{(w_{pq}^k)^2}{c + (w_{pq}^k)^2}$$

# Вопросы

1. Основные отличия обучения с подкреплением от обучения с учителем?
2. Почему нейроны с сигмоидальными функциями активации могут замедлять процесс обучения?
3. В чем выражается эффект переобучения НС?