

# Методы и средства обработки изображений

Янучкин Алексей Леонидович

Ассистент каф. ЭВМ

# Искусственные нейронные сети

Лекция 7

# Нейронная сеть

- *Искусственная нейронная сеть (ИНС)* – это существенно параллельно распределённый процессор, который обладает способностью к сохранению и репрезентации опытного знания.
- Построена по принципу организации и функционирования биологических нейронных сетей

# НС vs МОЗГ

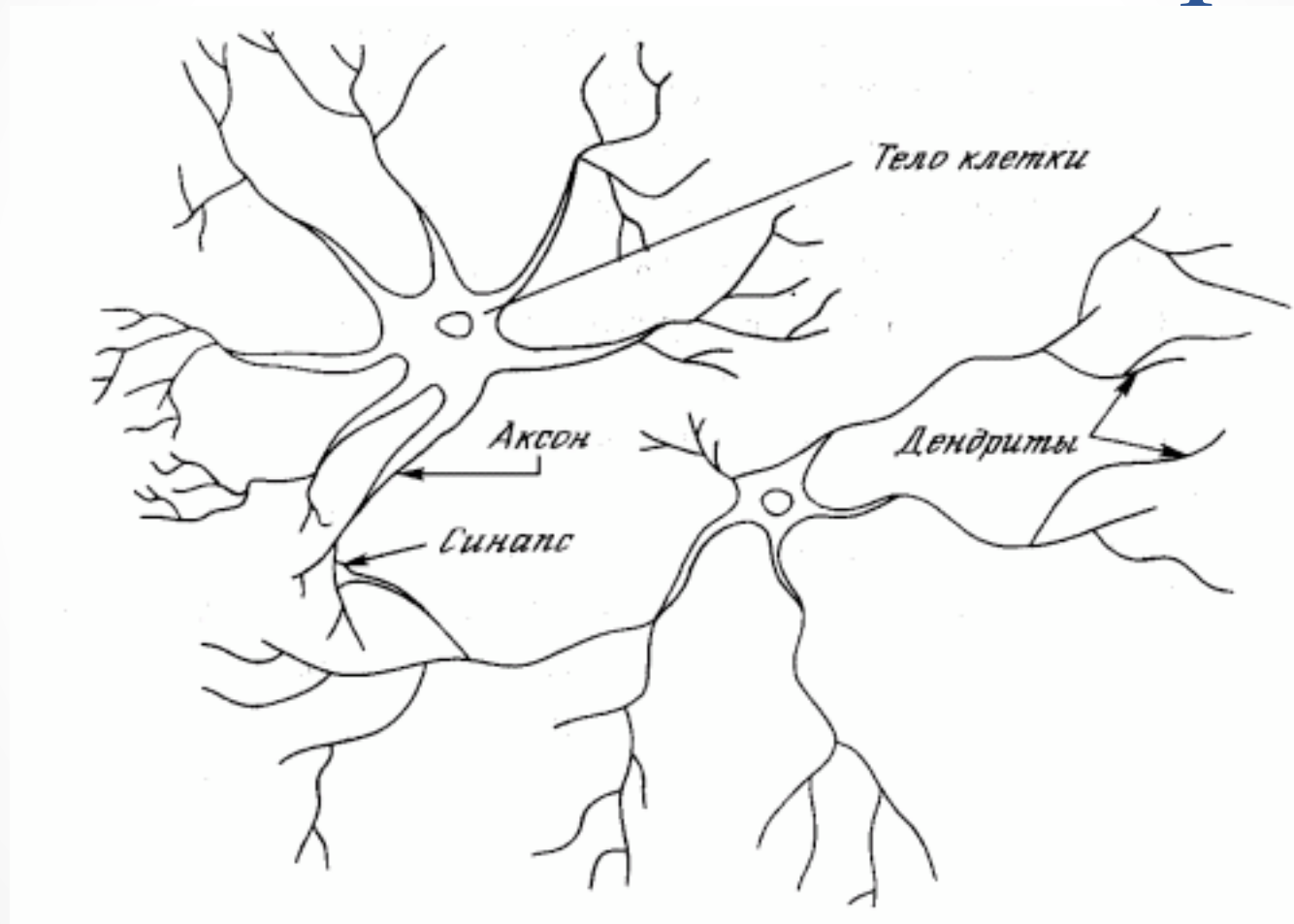
Сходь с мозгом в двух аспектах:

- знание приобретается сетью в процессе обучения;
- для сохранения знания используются силы межнейронных соединений, называемые также синоптическими соединениями.

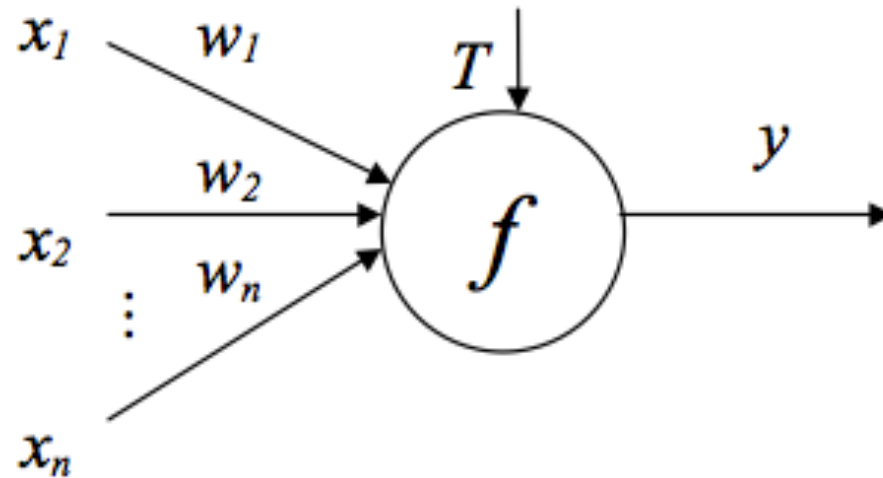
# Свойства НС

- Обучаемость.
  - Выбрав одну из моделей НС, создав сеть и выполнив алгоритм обучения, мы можем обучить сеть решению задачи, которая ей по силам.
- Способность к обобщению.
  - После обучения сеть становится нечувствительной к малым изменениям входных сигналов (шуму или вариациям входных образов) и дает правильный результат на выходе.
- Способность к абстрагированию.
  - Если предъявить сети несколько искаженных вариантов входного образа, то сеть сама может создать на выходе идеальный образ, с которым она никогда не встречалась.

# Биологический нейрон



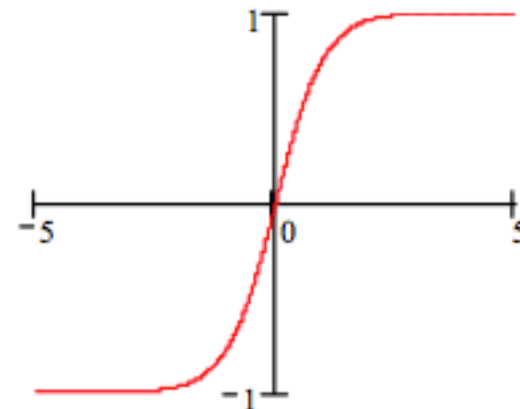
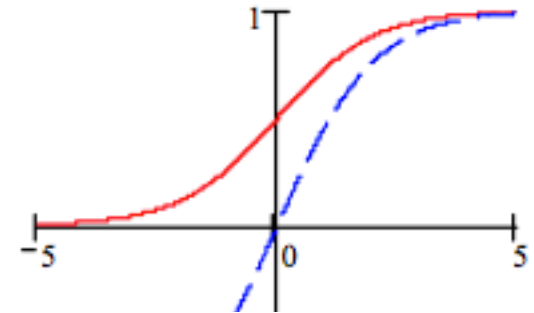
# Искусственный нейрон



$$y = f\left(\sum_{i=1}^n w_i x_i + T\right),$$

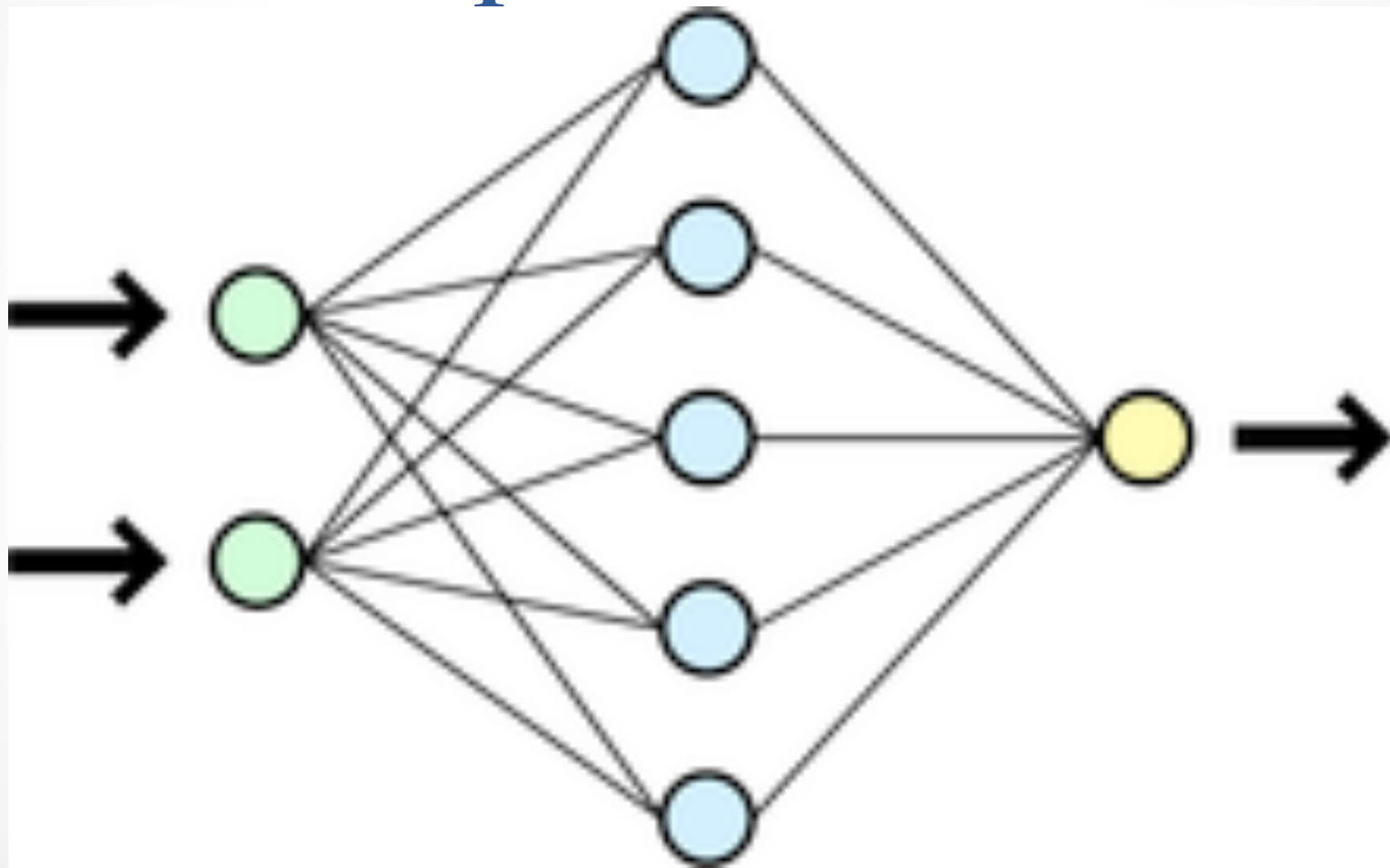
# Функции активации

- Линейная функция  $y = Ax$ ;
- Пороговая функция  $y = \begin{cases} 1, x > 0 \\ 0, x \leq 0 \end{cases}$ ;
- Биполярная пороговая функция  $y$
- Сигмоидная функция  $y = \frac{1}{1 + e^{-x}}$ ;
- Биполярная сигмоидная фу
- Гиперболический тангенс  $y$





# Нейронная сеть



# Классификация НС

По типу входной информации:

- аналоговые НС (используют информацию в форме действительных чисел);
- двоичные НС (оперируют с информацией, представленной в двоичном виде)

# Классификация НС

По характеру обучения:

- с учителем (известно входное пространство решений НС);
- без учителя (НС формирует выходное пространство решений только на основе входных воздействий – самоорганизующиеся сети);

# Классификация НС

По характеру настройки синапсов:

- сети с фиксированными связями (весовые коэффициенты НС выбираются сразу, исходя из условия задачи);
- сети с динамическими связями (в процессе обучения происходит настройка синаптических связей);

# Классификация НС

По методу обучения

- НС с алгоритмом обратного распространения ошибки;
- НС с конкурентным обучением;
- НС, использующие правило Хебба;
- НС с гибридным обучением, в которых используются
- различные алгоритмы обучения;

# Классификация НС

По характеру связей:

- НС с прямыми связями;
- НС с обратным распространением информации;

# Классификация НС

По характеру связей:

- НС с прямыми связями;
- НС с обратным распространением информации;

# Классификация НС

По архитектуре и обучению:

- Персептронные сети с прямыми связями;
- Самоорганизующиеся НС (НС Кохонена, НС адаптивного резонанса, рециркуляционные НС);
- НС с обратными связями (НС Хопфилда, НС Хэмминга, двунаправленная ассоциативная память, рекуррентные НС);
- гибридные НС (НС встречного распространения, НС с радиально-базисной функцией активации).



# Сферы применения

- распознавание зрительных, слуховых образов;
- ассоциативный поиск информации и создание ассоциативных моделей, синтез речи, формирование естественного языка;
- формирование моделей и различных нелинейных и трудно описываемых математически систем, прогнозирование развития этих систем во времени;
- системы управления и регулирования с предсказанием;
- разнообразные конечные автоматы: системы массового обслуживания и коммутации, телекоммуникационные системы;
- принятие решений и диагностика, исключаящие логический вывод, особенно в областях, где отсутствуют четкие математические модели:
  - в медицине, криминалистике, финансовой сфере.

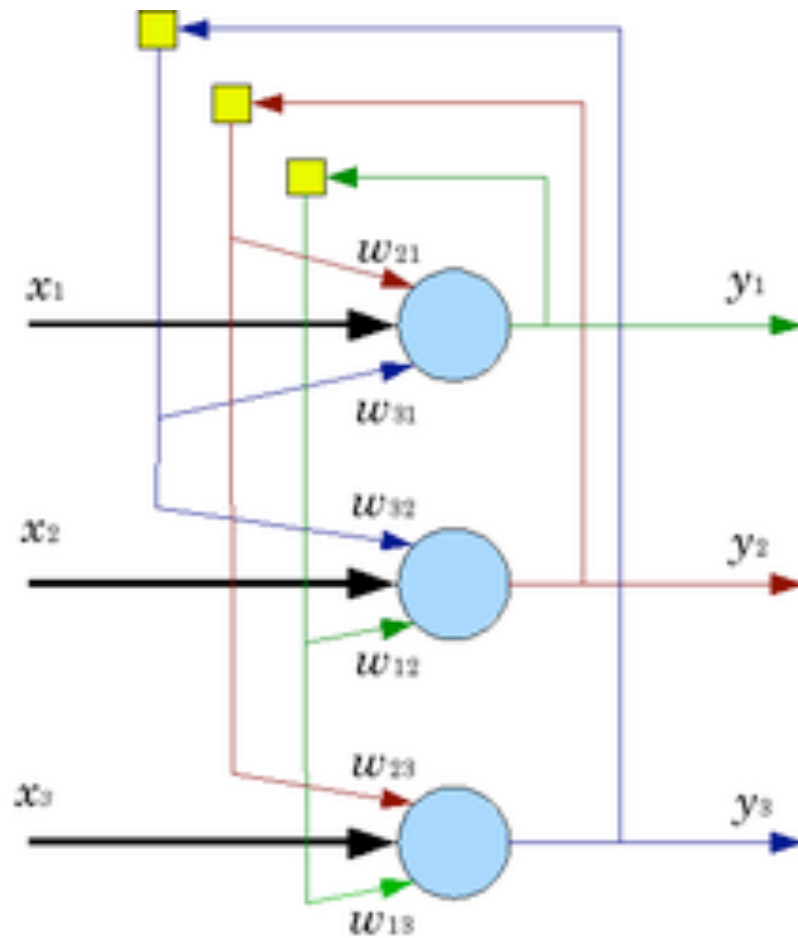
# НЕЙРОННАЯ СЕТЬ ХОПФИЛДА

...

# Сеть Хопфилда

- Сеть Хопфилда – однослойная, симметричная, не линейная автоассоциативная нейронная сеть, которая запоминает бинарные / биполярные образы.

# Топология



# Обучение

$$w_{ij} = \begin{cases} \sum_{k=1}^m a_i^k a_j^k, & i \neq j, \text{ для } i, j = \overline{1, n}, \\ 0, & i = j \end{cases}$$

где  $w_{ij}$  — вес связи от  $i$ -го нейрона к  $j$ -му;

$n$  — количество нейронов в сети;

$m$  — количество образов, используемых для обучения сети;

$a_i^k$  —  $i$ -й элемент  $k$ -го образа из обучающей выборки.

# Обучение

Матрица весовых коэффициентов

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nn} \end{bmatrix}.$$

# Воспроизведение

$$a_i(t+1) = f\left(\sum_{j=1}^n w_{ij} a_j(t)\right),$$

где  $a_j(t)$  – выход  $j$ -го нейрона в момент времени  $t$ , а  $f$  – бинарная / биполярная функция активации;

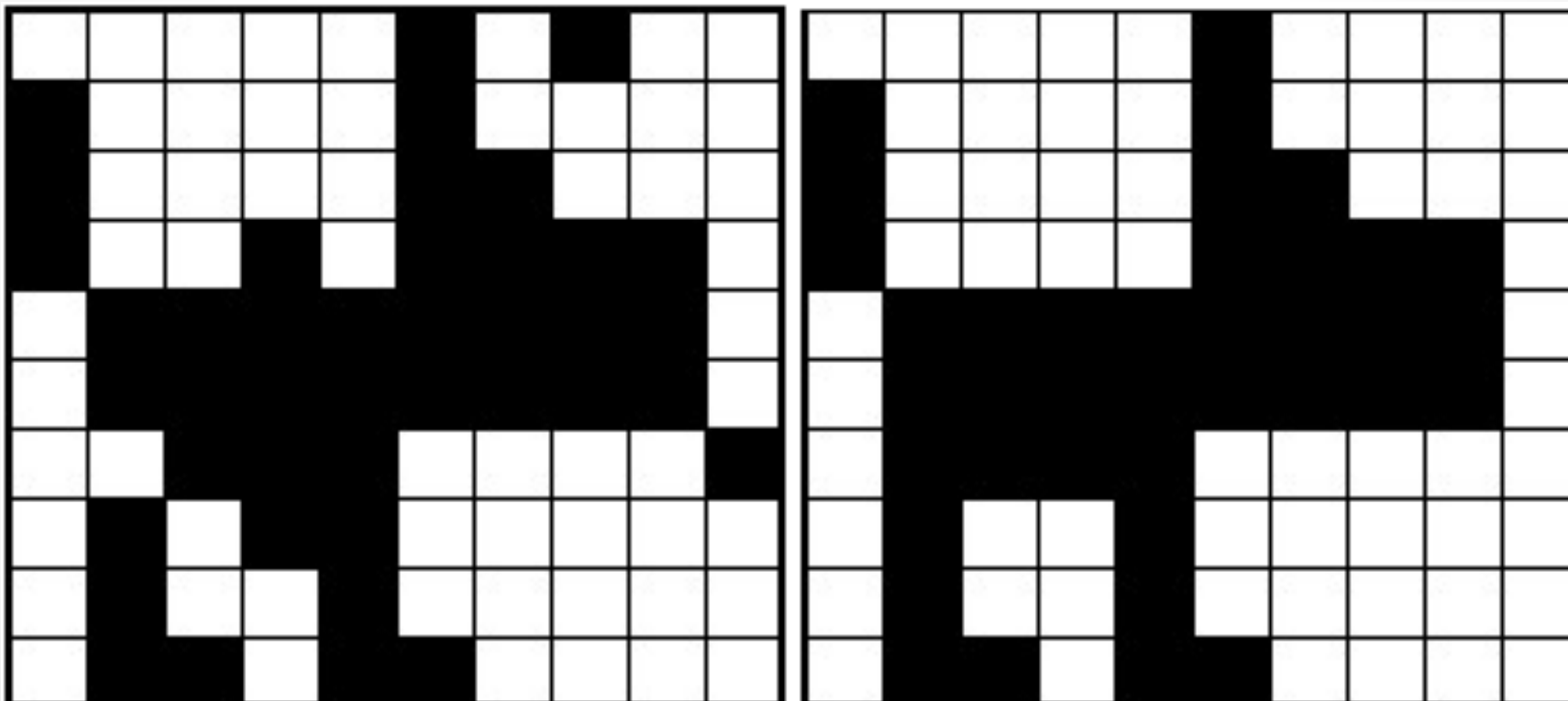
$$f(x) = \begin{cases} 1 & x > 0, \\ -1 & x \leq 0. \end{cases}$$

# Режимы воспроизведения

- Синхронный
  - При работе в синхронном режиме на один такт работы сети все нейроны одновременно меняют состояние
- Асинхронный
  - В случае асинхронной работы состояние меняет только один случайный нейрон



# Пример



# Недостатки

- Относительно небольшой объём памяти, величину которого можно оценить выражением

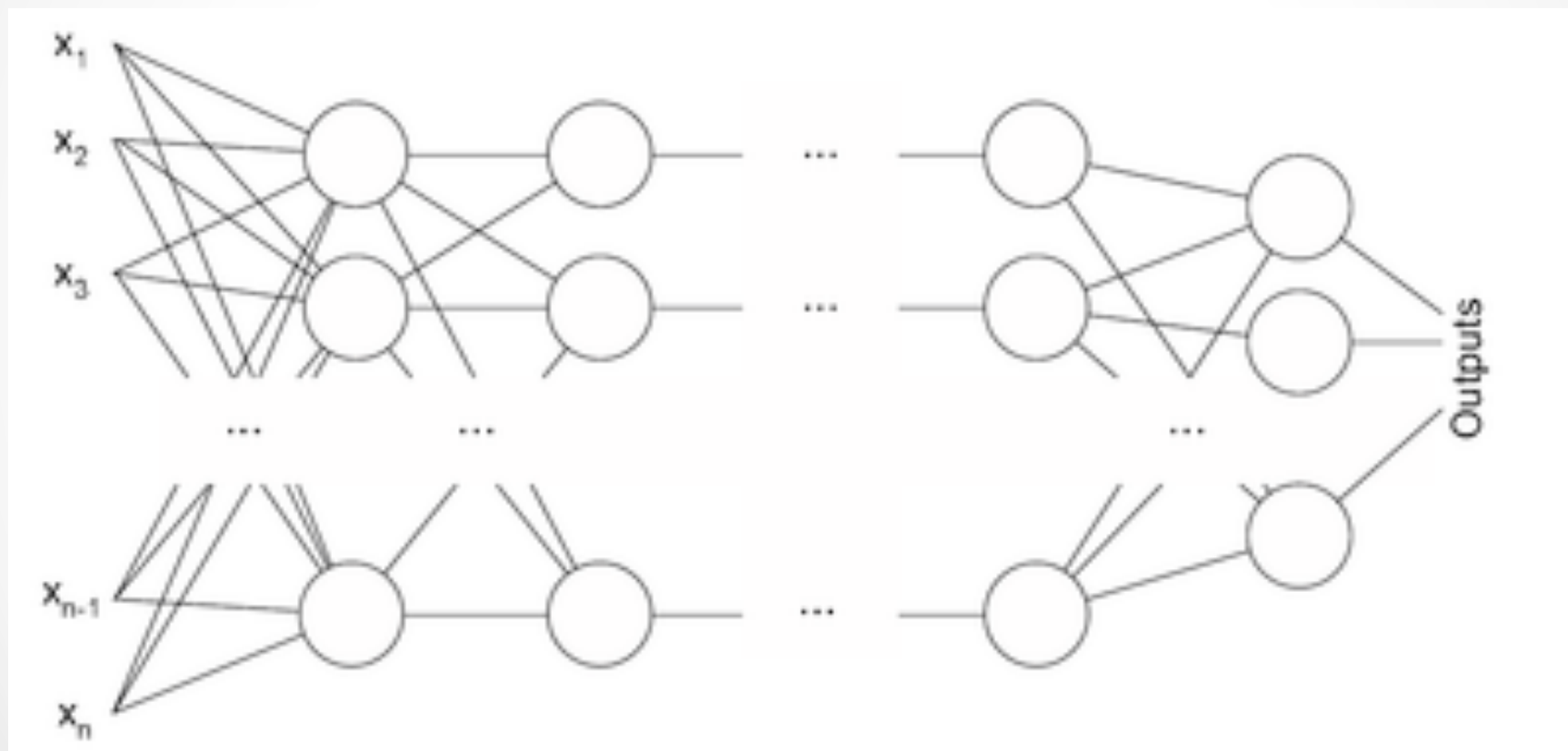
$$m = \frac{n}{2 \ln n + \ln \ln n},$$

- Достижение устойчивого состояния не гарантирует правильный ответ сети.
  - Это происходит из-за того, что сеть может сойтись к так называемым ложным аттракторам, иногда называемым «химерами» (как правило, химеры склеены из фрагментов различных образов).

# МНОГОСЛОЙНЫЙ ПЕРСЕПТРОН

...

# Топология



# Воспроизведение

- Скрытый слой

$$g_j = f\left(\sum_{i=1}^n v_{ij}x_i + Q_j\right),$$

- Выходной слой

$$y_k = f\left(\sum_{j=1}^h w_{jk}g_j + T_k\right).$$

# Обучение

Обучение персептрона проводится с помощью алгоритма обратного распространения ошибки, который минимизирует среднеквадратичную ошибку нейронной сети. Для этого с целью настройки синоптических связей используется метод градиентного спуска в пространстве весовых коэффициентов и порогов нейронной сети.

# Алгоритм обратного распространения ошибки

1. На первом этапе происходит начальная инициализация знаний сети.

- Простейший вариант такой инициализации – присвоить всем весам и порогам случайные значения из диапазона  $[-1,1]$

2. Далее для каждой пары векторов  $(x_r, y_r)$  выполняется следующее:

- Для входного вектора рассчитываются выходы нейронов скрытого слоя и выходы сети
- Происходит коррекция знаний сети, при этом главное значение имеет отклонение реально полученного выхода сети  $y$  от идеального вектора  $y_r$
- Корректируем выходной слой
- Корректируем скрытый слой

3. После того, как коррекция знаний произведена для каждой пары векторов, можно оценить степень успешности обучения сети для определения момента завершения алгоритма.

# Алгоритм обратного распространения ошибки

- При этом главной трудностью является определение ошибки нейрона скрытого слоя
- Эту ошибку явно определить по формуле, аналогичной скрытому слою нельзя, однако существует возможность рассчитать ее через ошибки нейронов выходного слоя (отсюда произошло название алгоритма обратного распространения ошибки)

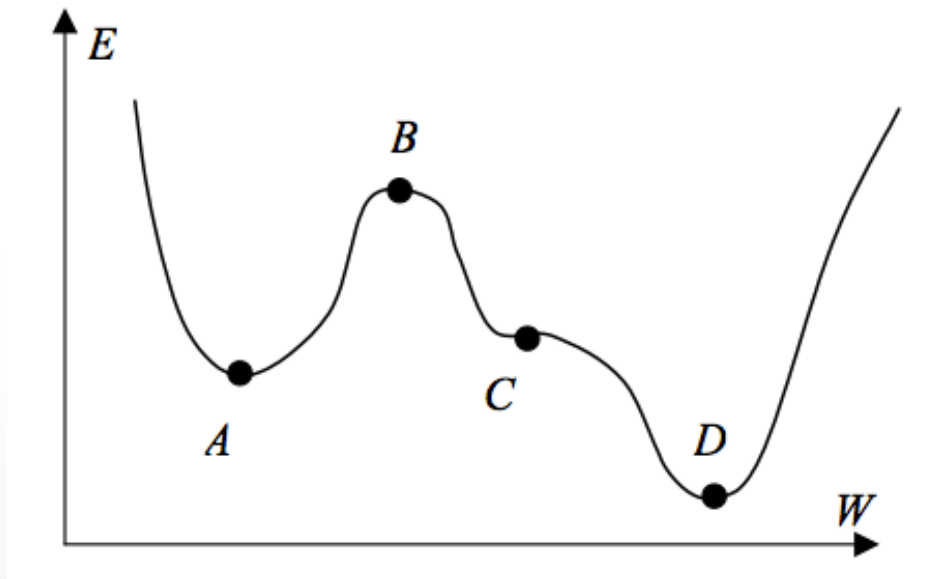


# Скрытые слои

- Способность персептрона-классификатора разделять образы в пространстве признаков прежде всего зависит от его скрытого слоя.
- Очевидно, что чем больше нейронов в скрытом слое, тем большее количество примеров этот слой может разделять.
- Персептрон упрощает функцию через эталонные точки, однако при этом обобщающая способность сети ухудшается.

# Метод градиентного спуска

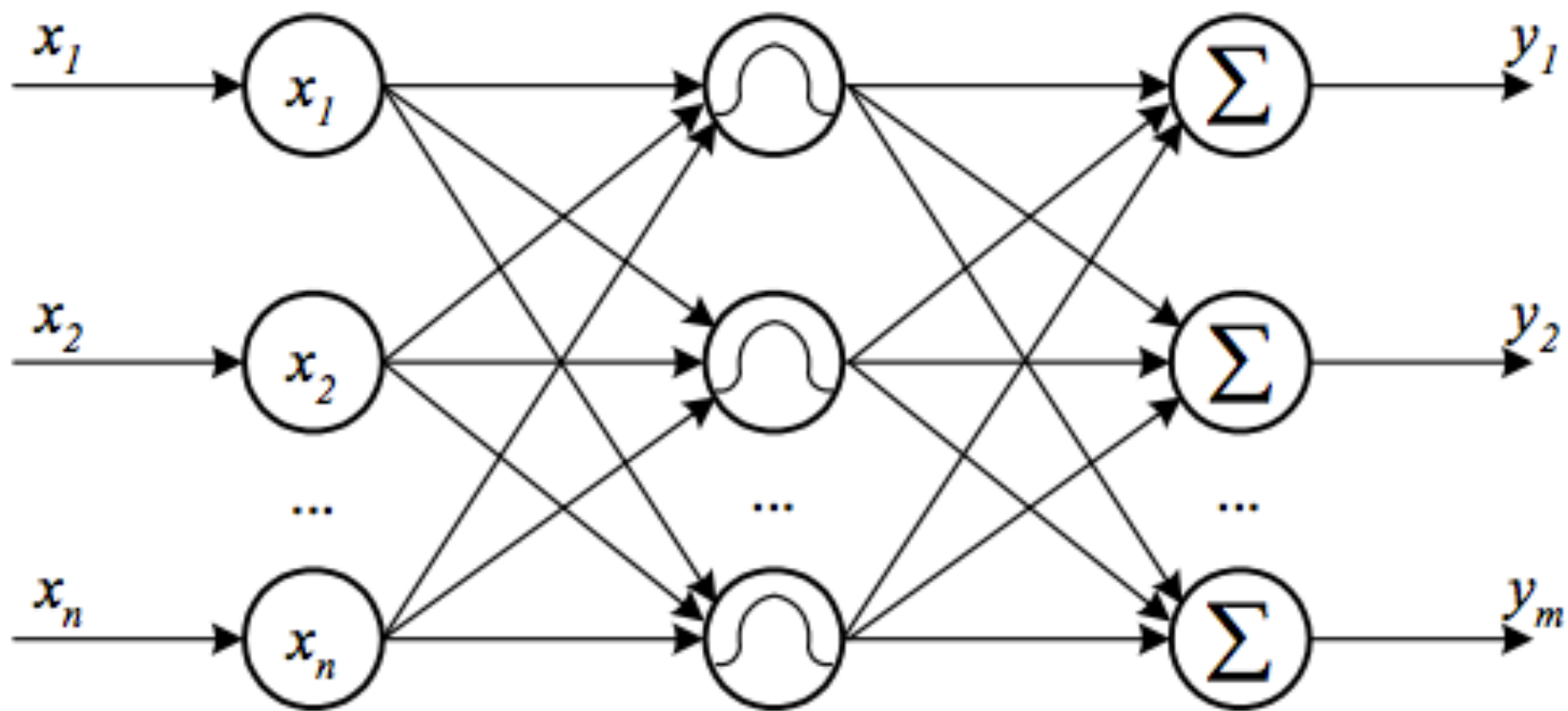
- Другой проблемой является то, что алгоритм градиентного спуска не гарантирует нахождение глобального минимума среднеквадратичной ошибки сети, а гарантируется определение только локального минимума.



СЕТЬ РБФ

...

# Топология



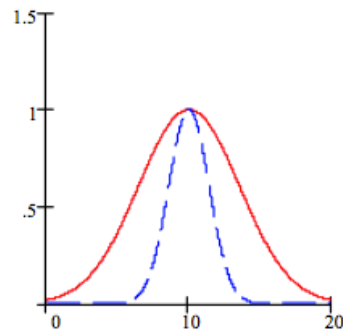
# РБФ-Ячейка

- введения вместо скрытого слоя нейронов слоя РБФ ячеек. Классический закон, по которому такая ячейка функционирует, определяется формулой гауссового колокола

$$g_j = \exp\left(\frac{-|x - t^j|^2}{\sigma_j^2}\right),$$

# Обучение

- РБФ ячейки обучаются путем подбора центра и отклонения каждой из них. Для классификатора в качестве центра выбирается центр кластера в пространстве признаков, компактно содержащего образы одного и того же класса
- Количество РБФ ячеек выбирается таким образом, чтобы покрыть гауссовыми колоколами все классы.



# Воспроизведение

- РБФ-слой

$$g_j = \exp\left(\frac{-|x - t^j|^2}{\sigma_j^2}\right),$$

- Выходной слой

$$y_k = \sum_{j=1}^h w_{jk} g_j.$$

# РБФ vs многослойный персептрон

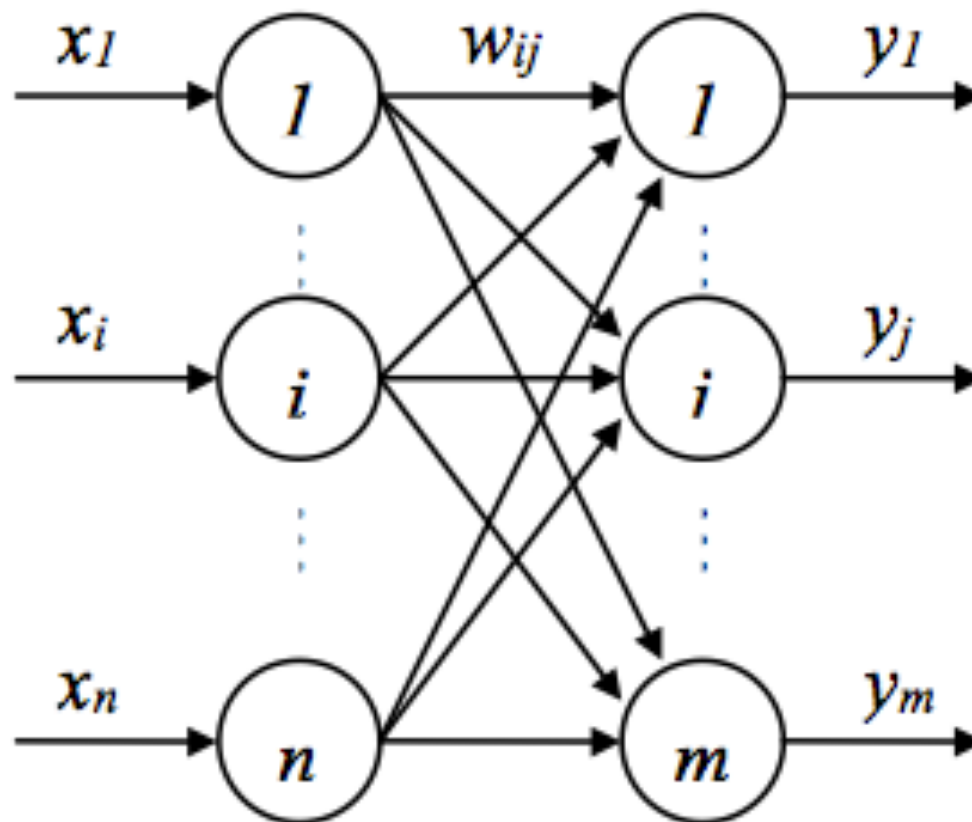
- Скорость обучения такой РБФ-сети гораздо выше, причем допускается полностью аналитический подход к расчету весовых коэффициентов.
- РБФ - ухудшение точности аппроксимации



# КОНКУРЕНТНАЯ НЕЙРОННАЯ СЕТЬ

...

# Топология



# Воспроизведение

$$y_j = \sum_{i=1}^n w_{ij} x_i = |w^j| |x| \cos \alpha,$$

где  $x = (x_1, x_2, \dots, x_i, \dots, x_n)$  – входной вектор;  $w^j = (w_{1j}, w_{2j}, \dots, w_{ij}, \dots, w_{nj})$  – вектор весовых коэффициентов нейрона, а  $|x|$  и  $|w^j|$  – их модули,  $\alpha$  – угол между ними.

# Обучение

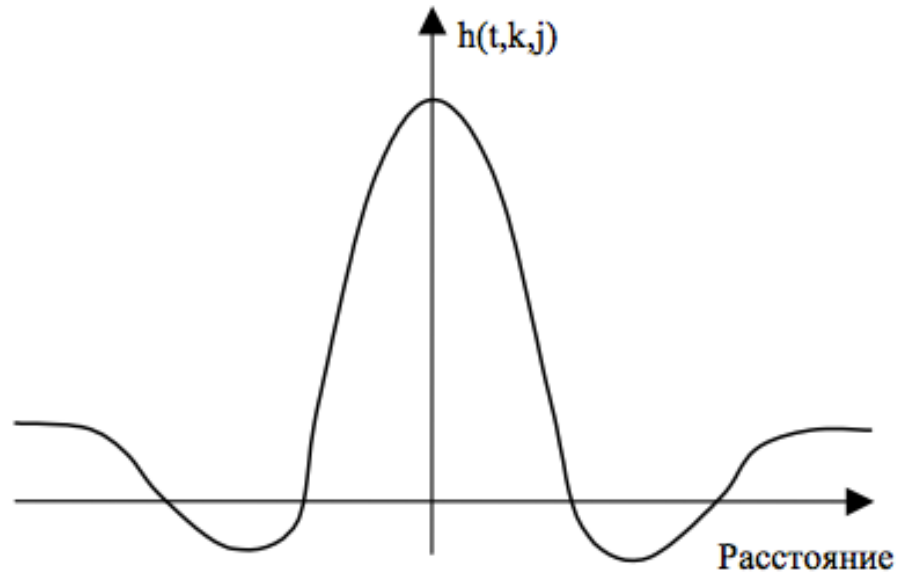
$$w_{ij}(t+1) = w_{ij}(t) + \beta(x_i - w_{ij}(t)),$$

где  $\beta$  – скорость обучения.

- Случайное начальное распределение весовых коэффициентов может привести к тому, что некоторые нейроны никогда не станут победителями, так как их весовые векторы окажутся удаленными от всех входных векторов.
- Хорошие результаты на практике показало частотно–зависимое конкурентное обучение

$$d_v = \min_j (|x - w^j| f_j).$$

# Сеть Кохонена



$$w_{ij}(t+1) = w_{ij}(t) + \beta h(t, k, j) (x_i - w_{ij}(t)).$$

# Резюме лекции

- Искусственная нейронная сеть – моделирует наш МОЗГ
- Нейронные сети
  - Сеть Хопфилда
  - Многослойный персептрон
  - Сеть РБФ
  - Конкурентная нейронная сеть
  - Сеть Кохонена