

Лекция № 10

Искусственные нейронные сети

Artificial neural networks

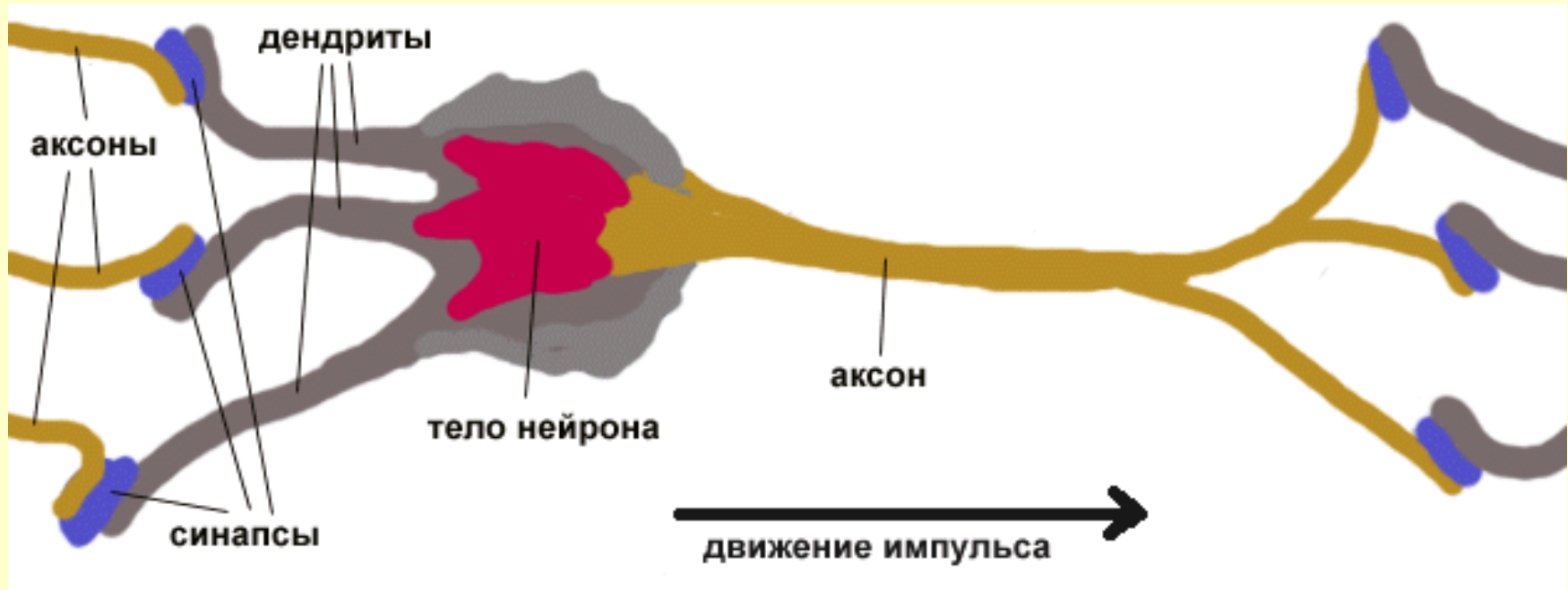
НЕЙРОННЫЕ СЕТИ

1. Основные определения и принципы

Нейронная сеть - совокупность нейронов, связанных, между собой определенным образом, с выделенными входными и выходными нейронами.

Искусственная нейронная сеть - совокупность связанных между собой формальных нейронов.

Схема биологического нейрона



Бионейрон - клетка, имеющая ядро и два вида длинных отростков, связанных с другими нейронами.

Отростки подразделяются на *дендриты*, передающие сигнал к ядру от других нейронов, и *аксоны*, передающие сигнал другим нейронам. В месте соединения дендрита и аксона возникает связь - *синапс*, толщина которого и характеризует соединение.

НЕЙРОННЫЕ СЕТИ

Функционирование бионейрона происходит следующим образом:

В текущий момент нейрон получает сигнал от других нейронов, который, проходя через синапсы и дендриты, изменяет свое значение, в зависимости от толщины связи (умножается на синаптический вес).

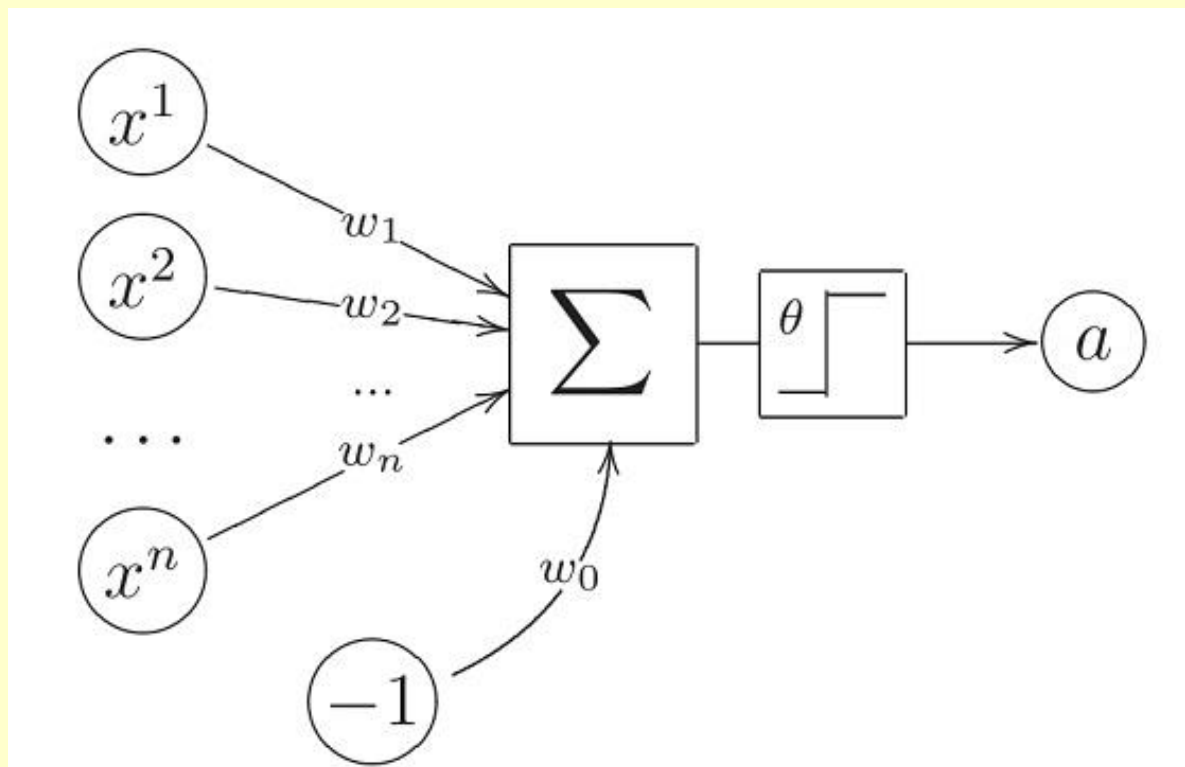
Далее в ядре сигналы со всех дендритов суммируются, и в зависимости от величины полученного значения формируется внешний сигнал, который передается следующим нейронам через аксоны.

Считается, что **обучение** нейронов в организме заключается в изменении толщины синапсов и модификации функции зависимости выходных сигналов от входных сигналов.

НЕЙРОННЫЕ СЕТИ

У. Маккалок и У. Питтс предложили модель формального нейрона.

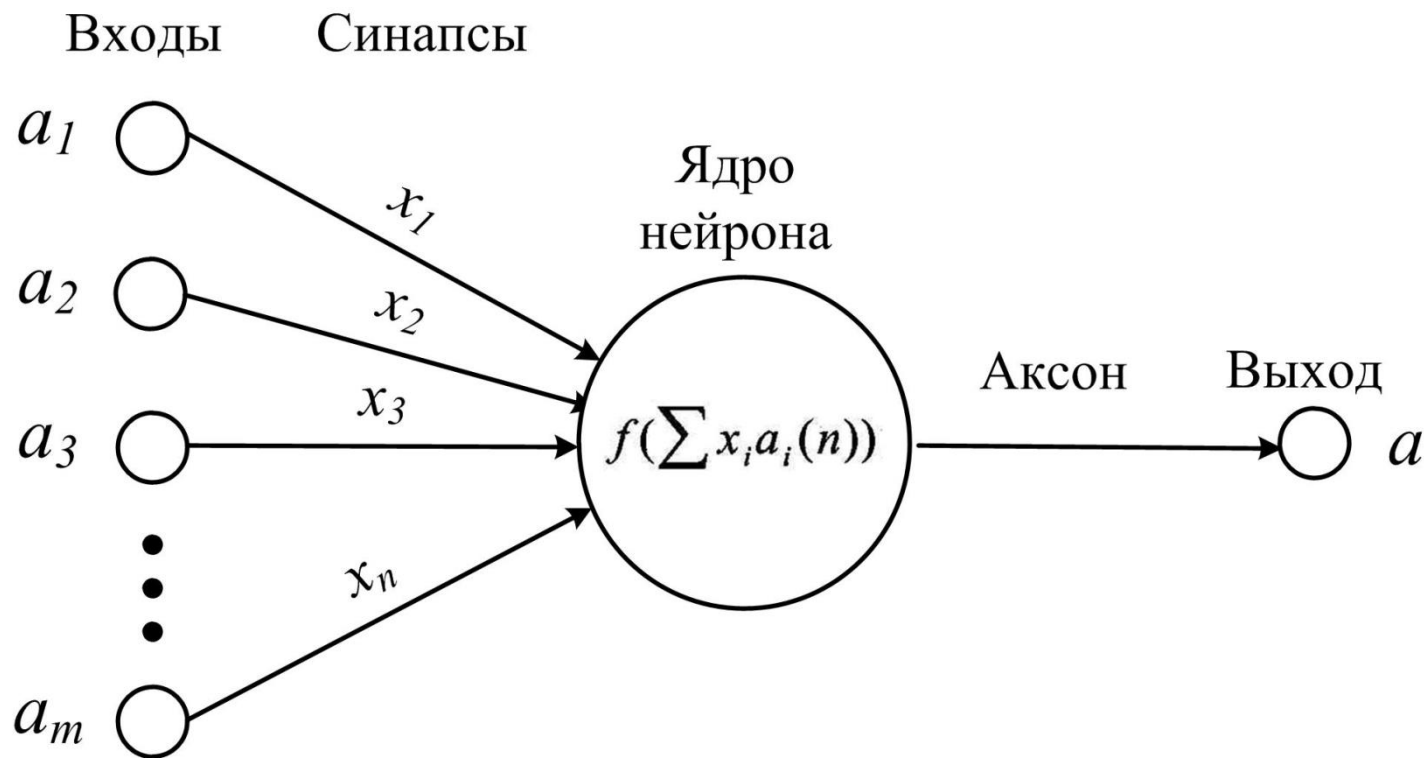
Математически **формальный нейрон** – это пороговый элемент с единственным двоичным выходом, функция активации которого зависит от линейной комбинации всех входных сигналов.



НЕЙРОННЫЕ СЕТИ

Формальный нейрон имеет m входов a_1, \dots, a_m ($m \geq 1$) и один выход a .

Он характеризуется: порогом срабатывания θ и весами x_1, \dots, x_m .
Каждому входу a_i ставится в соответствие синаптический вес x_i .



НЕЙРОННЫЕ СЕТИ

Формальный нейрон работает в дискретном времени, его выход в момент $n+1$ зависит только от входов в момент n .

Эта зависимость формально описывается следующим образом:

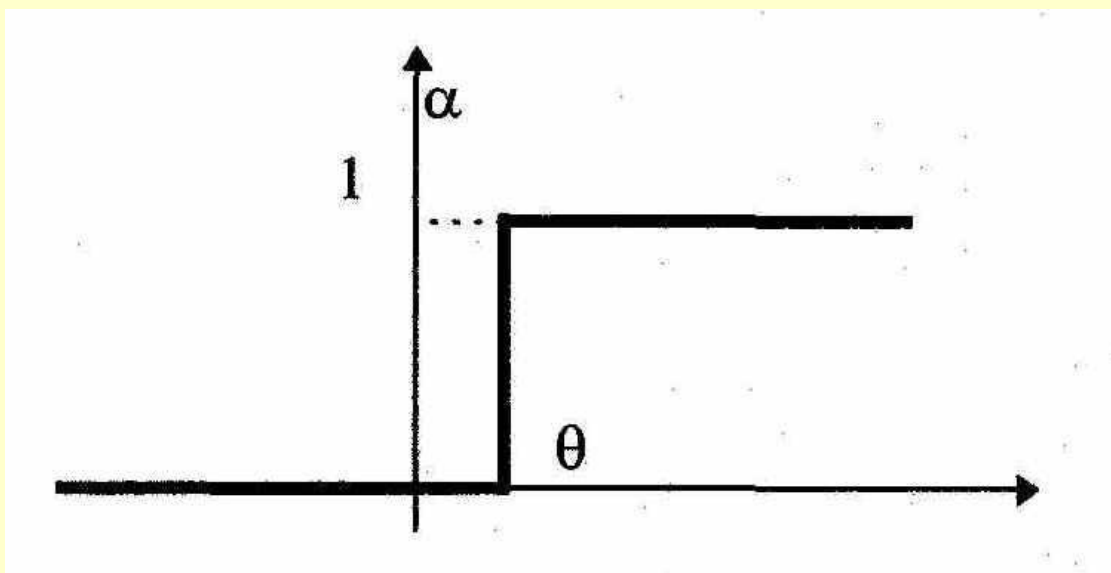
$$\alpha(n+1) = \frac{\text{sign}\left(\sum x_i a_i(n) - \theta\right) + 1}{2}$$

т.е. формальный нейрон в момент $n+1$ передает импульс по аксону в том и только в том случае, когда сумма всех весов возбужденных входов в момент n превышает порог срабатывания θ .

Отметим, что положительный вес $x_i > 0$ указывает на то, что i -й вход является возбуждающим, а отрицательный вес $x_i < 0$ - что i -й вход является тормозящим.

НЕЙРОННЫЕ СЕТИ

Формальный нейрон является примером нейрона с так называемой бинарной (или разрывной) функцией преобразования. Ее можно изобразить на рисунке таким образом:



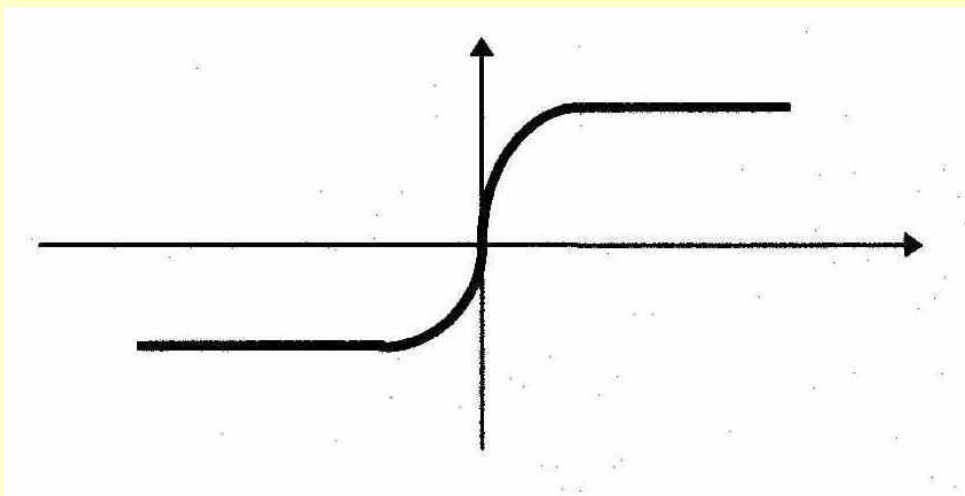
но применение разрывных функций не очень выгодно, из-за невозможности применить аппарат вычислительной математики.

НЕЙРОННЫЕ СЕТИ

Поэтому в настоящий момент используют так называемую сигмоидную функцию преобразования. Тогда функционирование нейрона описывается следующим образом:

$$\alpha(n+1) = f\left(\sum x_i a_i(n)\right),$$

где $f(A)$ имеет вид, представленный на рисунке:



НЕЙРОННЫЕ СЕТИ

Обычно, в качестве f выбирают функции следующего вида:

$$1. \quad f(A) = \frac{A}{\theta + |A|}, \quad (-\infty < A < +\infty, \theta > 0);$$

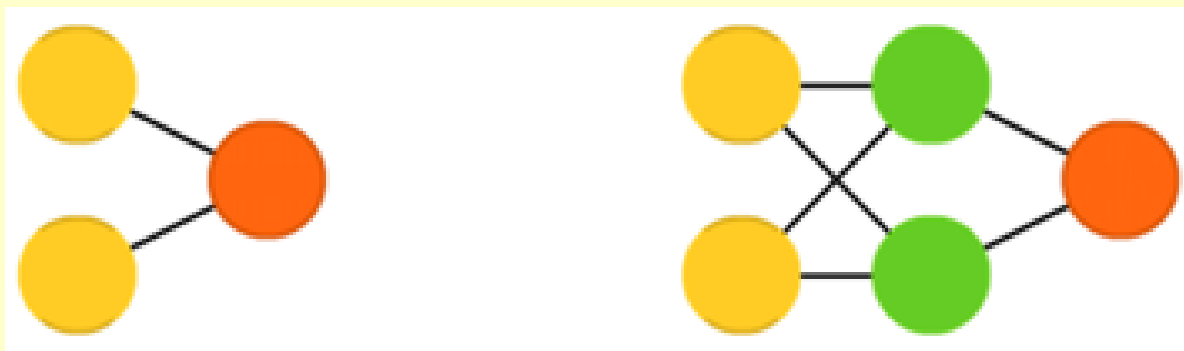
$$2. \quad f(A) = th \frac{A}{\theta}, \quad (-\infty < A < +\infty, \theta > 0);$$

$$3. \quad f(A) = \frac{1}{1 + e^{-(A-\theta)}}, \quad (-\infty < A < +\infty, \theta > 0).$$

Число θ при этом характеризует угол наклона касательных в окрестности нуля и называется параметром пологости нейрона

НЕЙРОННЫЕ СЕТИ

Рассмотрим теперь соединение формальных нейронов в сеть.



Простейшая нейронная сеть состоит из двух входных и одного выходного нейрона и может моделировать логический вентиль — базовый элемент цифровой схемы, выполняющий элементарную логическую операцию.

НЕЙРОННЫЕ СЕТИ

Существует несколько основных типов соединения формальных нейронов в сеть, а также комбинации и модификации.

Самый очевидный тип - *последовательное соединение слоев*.

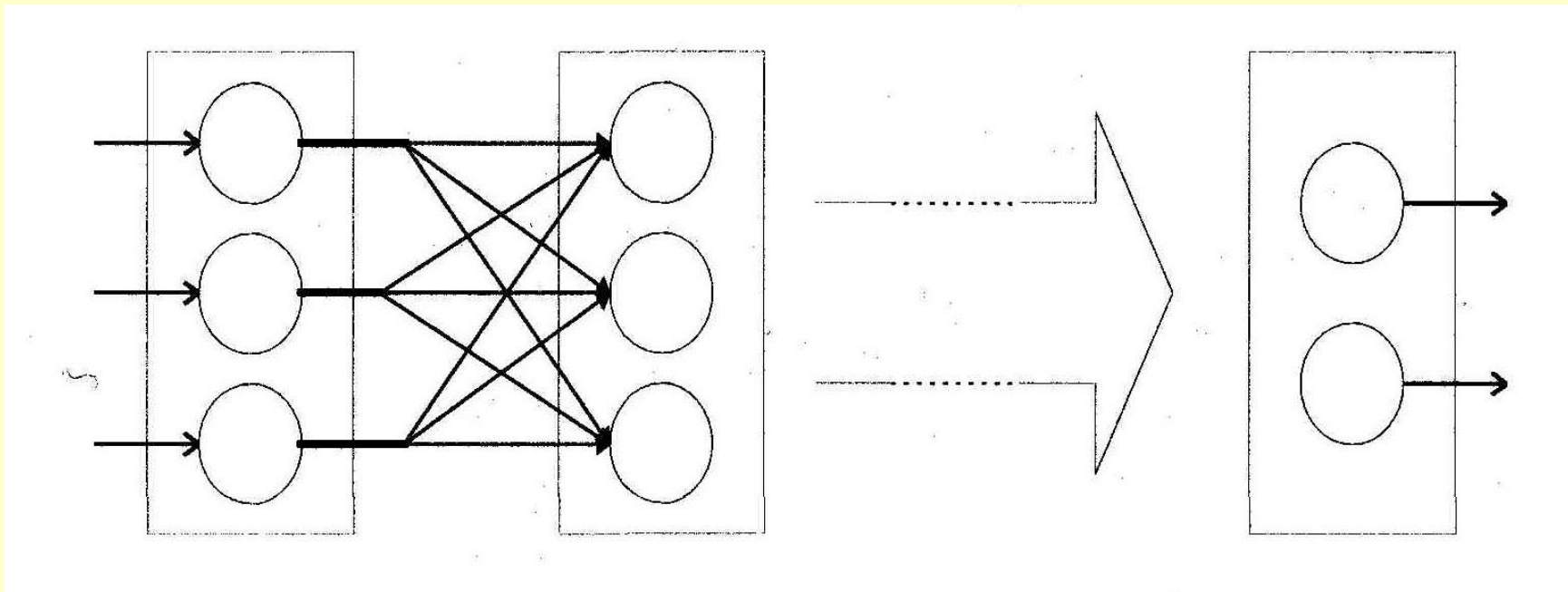
Выбираются нейроны *входного* слоя, причем ровно столько, сколько сигналов собирается обрабатывать сеть. Далее определяется количество *скрытых* или *внутренних* слоев и количество нейронов в них. Эти числа зависят от конкретной задачи и выясняются уже в процессе обучения или формирования сети.

После этого формируются нейроны *выходного* слоя - по количеству выходных сигналов.

Все слои соединяются друг с другом последовательно, по принципу все со всеми, применительно к нейронам соседних слоев. Получается картина, представленная на рисунке:

НЕЙРОННЫЕ СЕТИ

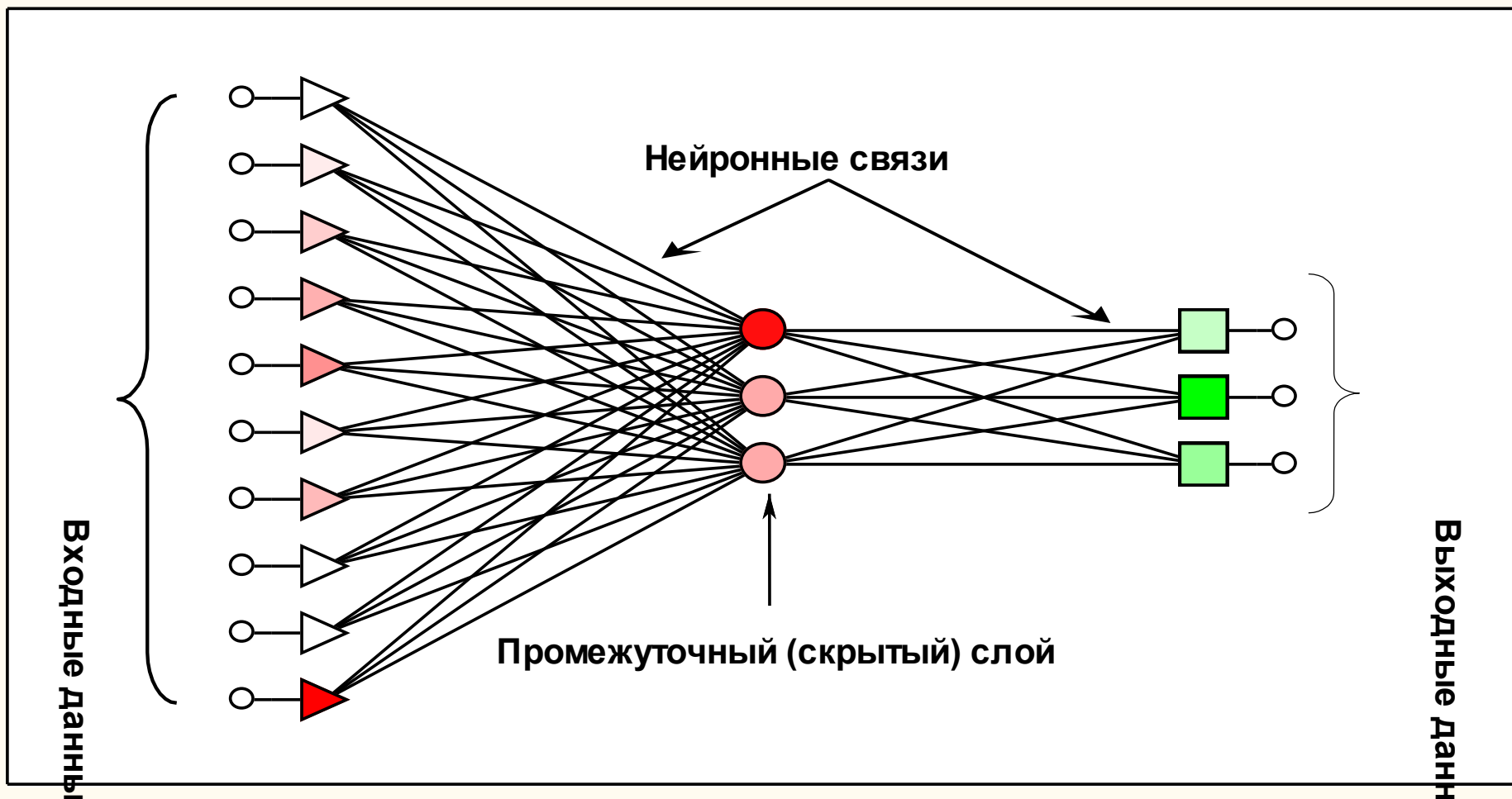
Нейронная сеть с последовательным соединением слоев



НЕЙРОННЫЕ СЕТИ

Нейронная сеть с последовательным соединением слоев

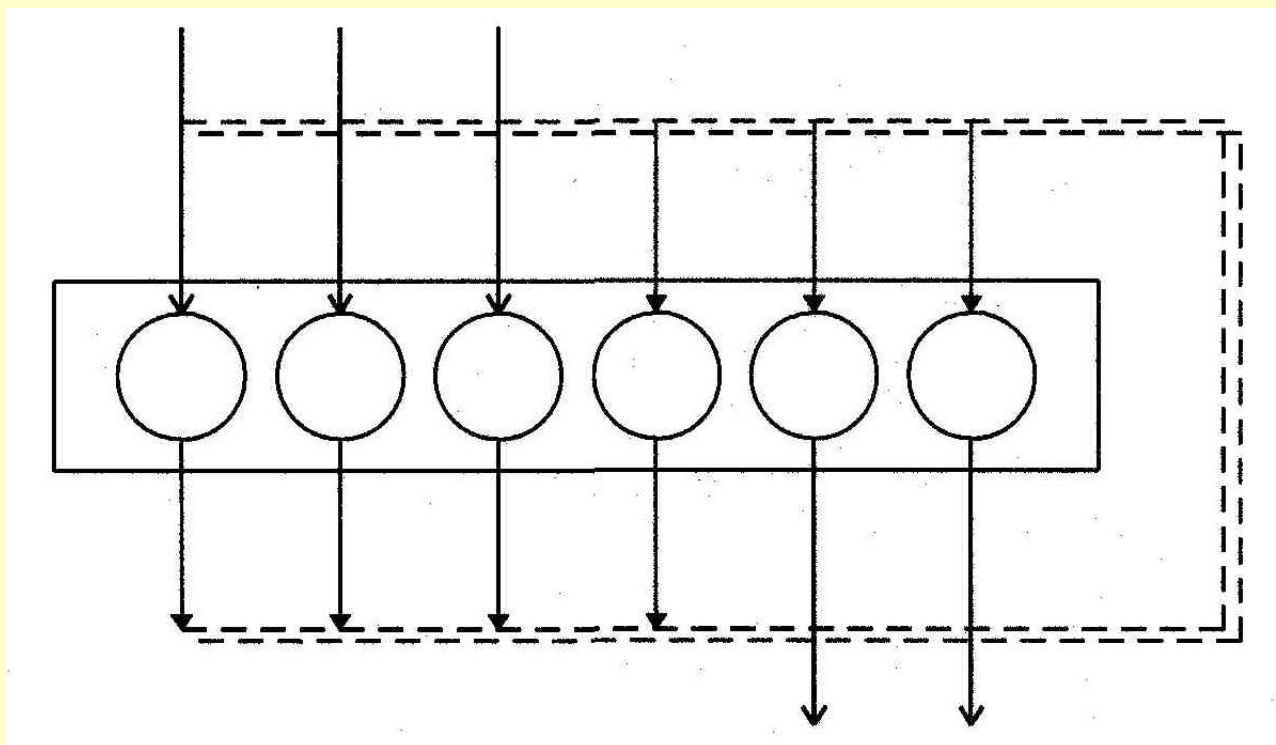
Архитектура сети



НЕЙРОННЫЕ СЕТИ

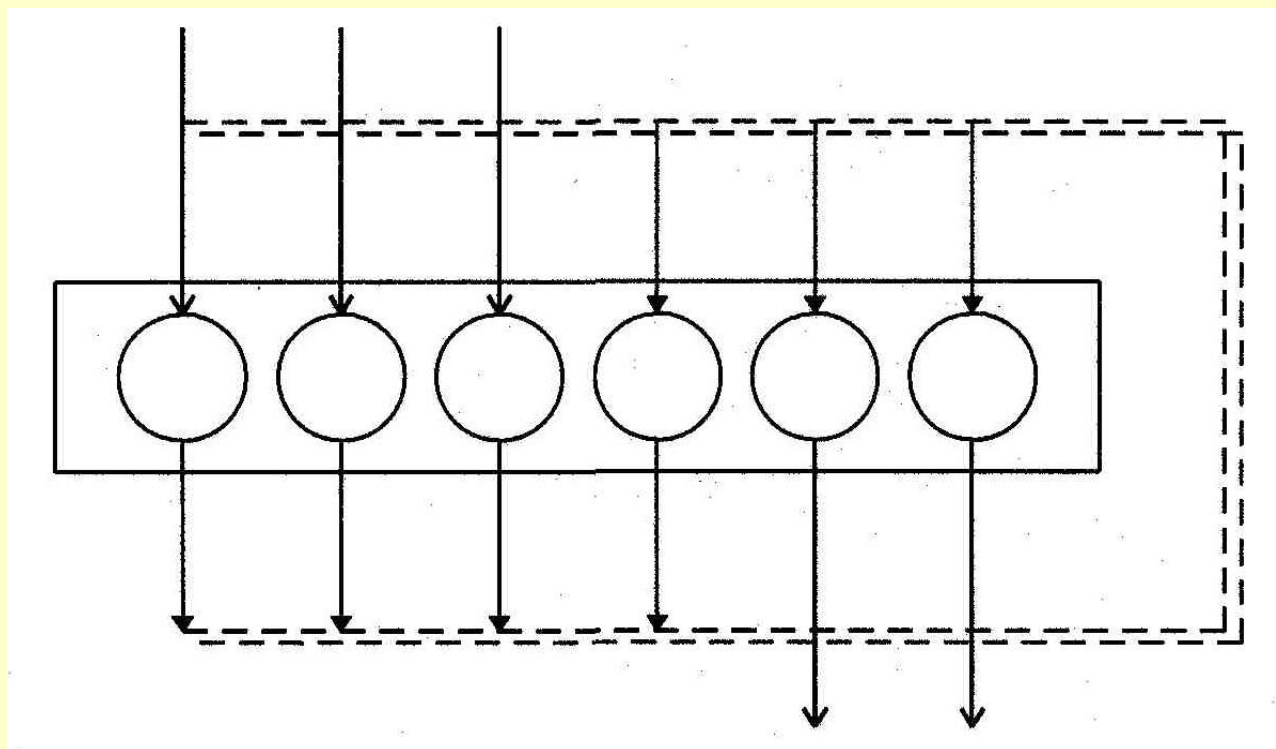
Также распространен и другой тип сетей: *сети с обратной связью*, или так называемые *полносвязные сети*, или *сети Хопфилда*.

Здесь среди нейронов также выделяются входные и выходные, но сигнал передается не последовательно со слоя на слой, а последовательно, относительно времени.

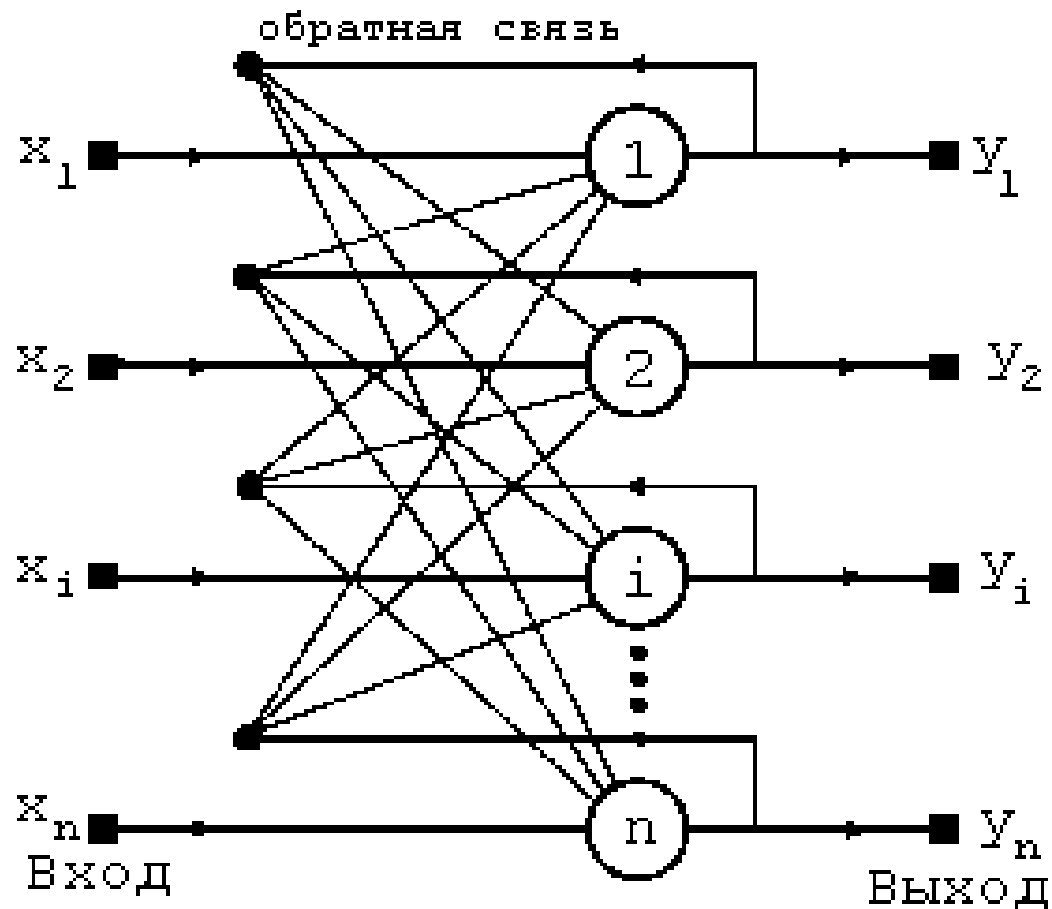


НЕЙРОННЫЕ СЕТИ

На первом шаге функционирования сети Хопфилда на дендриты входных нейронов подается сигнал, а на остальные - значения по умолчанию. Затем нейроны передают сигнал всем дендритам, включая входные нейроны. Как видно, при этом образуется обратная связь. По прошествии нескольких тактов такого функционирования, сигнал снимается с выделенных выходных нейронов.

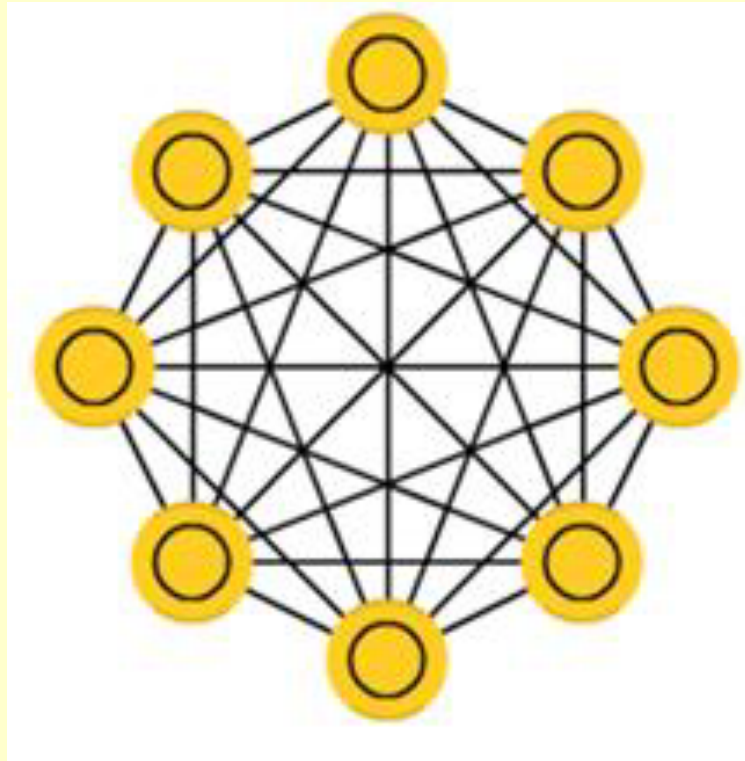


НЕЙРОННЫЕ СЕТИ



Структурная схема сети Хопфилда

НЕЙРОННЫЕ СЕТИ



В нейронной сети Хопфилда каждый нейрон выступает в роли *входного* нейрона до обучения, *скрытым* во время обучения и *выходным* после обучения.

НЕЙРОННЫЕ СЕТИ

При необходимости рассмотренные выше два вида нейронных сетей комбинируют, вводя *обратную связь во внутренние слои последовательных сетей*.

Это дает существенную экономию ресурсов в задачах с большим количеством обрабатываемых параметров, без значительных потерь информативности сети.

НЕЙРОННЫЕ СЕТИ

1. Обучение нейронных сетей

Выше было рассмотрено так называемое *прямое функционирование сети*.

Заключается это действие в формировании ответа (вектора выходных сигналов) при предъявлении вопроса (вектора входных сигналов).

Ответ и вопрос представлены в виде векторов, состоящих из рациональных чисел, лежащих в области $(-\acute{h}, +\acute{h})$, где $\acute{h} = \pm \lim f(A)$, при $A \rightarrow \infty$.

Именно такими числами и оперируют нейроны и нейронная сеть. Остальные виды данных получаются путем приведения с помощью функций соответствия.

НЕЙРОННЫЕ СЕТИ

Основное свойство нейронных сетей - способность к обучению, или же накапливание опыта.

Пусть мы хотим, чтоб на вопрос *In* сеть выдала ответ *Out*.

Для этого мы подаем *In* на вход сети и проводим прямое функционирование в результате, которого получаем ответ *Out**.

Теперь нужно оценить, на сколько полученный ответ *Out** отличается от *Out*.

Вводится функция оценки $H = H(Out, Out^*)$ со следующими свойствами: $H \geq 0$ и чем меньше H , тем меньше различий между *Out* и *Out**.

В качестве оценки можно выбрать множество функций, к примеру:

$$H = \sum_i \left(Out_i^* - Out_i \right)^2$$

НЕЙРОННЫЕ СЕТИ

Считается, что сеть обучилась примеру, если функция оценки для него меньше заданного числа s .

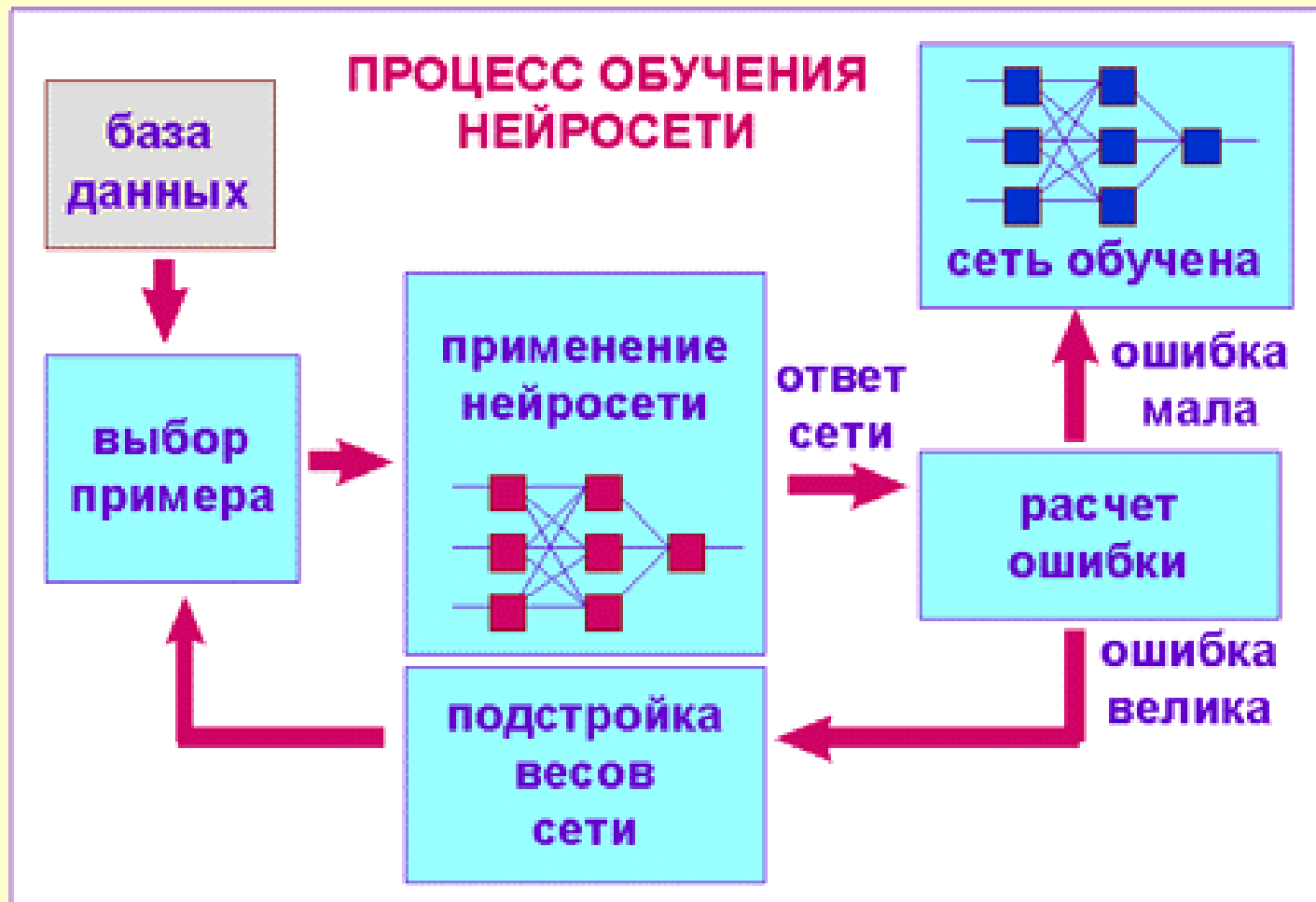
То есть нужно стремиться уменьшить H до заданного значения. Вспомним, что ответ Out^* есть функция от In и от множества синаптических весов X .

Теперь видно, что на самом деле $H = H(In, Out, X)$, но In и Out - фиксированные величины, поэтому изменять и подстраивать можно только множество весовых коэффициентов X .

В этой подстройке и заключается суть обучения.

Обучение нескольким примерам (обучающей выборке) заключается в минимизации функции H , зависящей уже не от одного предъявляемого примера, а от нескольких. В качестве нее обычно выбирают среднее арифметическое оценок примеров. Процесс функционирования с последующим вычислением оценки называется прямым функционированием с оценкой.

НЕЙРОННЫЕ СЕТИ



НЕЙРОННЫЕ СЕТИ

Рассмотрим теперь метод настройки весов.

Первые модели, к примеру, персептрон, не являлись обучаемыми в чистом виде. Так как это были аналого-цифровые схемы, то в качестве синаптических связей у них выступали резисторы с постоянными, а затем и переменными сопротивлениями. Величина их вычислялась путем решения большого числа уравнений.

Скачком развития стали попытки подстройки весов напрямую, без решения уравнений и появления соответствующего метода обратного распространения ошибки (**back - propagation**).

НЕЙРОННЫЕ СЕТИ

Рассмотрим метод обратного распространения ошибки, применимый к нейронам с сигмоидной функцией преобразования.

Нам необходимо минимизировать функцию многих переменных H , которая является дифференцируемой по всем параметрам (ввиду того, что $f(A)$ - дифференцируема по A).

В вычислительной математике известен итерационный метод градиентного спуска для решения подобных задач. Он успешно применяется для вычисления минимума функции. Напомним его вид:

$$x_i^{m+1} = x_i^m - S \frac{\partial H}{\partial x_i}$$

где переменная S называется шагом обучения и выбирается, исходя из соображений оптимизации.

НЕЙРОННЫЕ СЕТИ

Основной задачей для данного метода является быстрое нахождение градиента:

$$\frac{\partial H}{\partial x_i} = \frac{\partial H}{\partial f} * \frac{df}{\partial A} * \frac{\partial A}{\partial x_i} = H' * a(n)$$

Заметим, что $a(n)$, f' можно вычислить при проведении прямого функционирования. Этот процесс, собственно, и получил название нагруженного прямого функционирования. После того как происходит получение результата, вычисляется ошибка и параметр H' . Далее происходит обратное функционирование, со сверткой вычисленных при нагруженном функционировании параметров, и вычисление градиента.

НЕЙРОННЫЕ СЕТИ

Существует два **положительных фактора** в обучении сети градиентными методами.

Во-первых, изменяются веса сразу всех синапсов, в полном соответствии с анализом чувствительности к изменению. Это позволяет полностью распараллелить обучение, что дает существенное сокращение времени.

Во-вторых, шаг обучения делается всегда в оптимальном направлении, ведь небольшой шаг в направлении градиента всегда влечет уменьшение оценки.

Существует также **небольшой минус**: система становится критичной к появлению локальных минимумов. Но на сегодня известно несколько способов их избежать, что позволяет активно применять градиентные методы в приложениях.

НЕЙРОННЫЕ СЕТИ

Итак, алгоритм обучения градиентными методами включает следующую последовательность действий:

1. Подать на вход вектор сигналов.
2. Провести нагруженное прямое функционирование.
3. Произвести оценку функционирования (для одного примера или же для страницы).
4. Подать на выход вектор ошибок.
5. Провести обратное функционирование и вычислить градиент.
6. Оптимизировать шаг обучения и изменить синаптические веса.

Применение нейронных сетей

На сегодняшний день, существует множество систем, основанных на нейронных сетях.

Дадим обзор функциональных типов нейронных сетей и описание прикладных задач, решаемых ими.

Первый тип сетей – **классификаторы**. Основная задача классификатора - попытаться определить принадлежность предложенного примера к одному или другому классу. (Например, в медицине это попытка определить принадлежность симптомов к классу того или иного заболевания.) Количество распознаваемых классов обычно влияет на число нейронов в последнем слое и, более того, обычно является важной характеристикой сети. Наибольшее применение нейросети этого типа нашли в экспертных системах и системах диагностики.

Применение нейронных сетей

Второй по значимости тип нейросетей – распознаватели.

Небольшое отличие от классификаторов заключается в уменьшении числа классов ответов до двух. То есть специализация распознавателей заключается в умении ответить на вопрос типа «свой - чужой». «Своими» считаются примеры, которым сеть предварительно обучили, а все оставшиеся автоматически считаются «чужими».

Третий тип – предикторы или предсказатели. Основной задачей сетей этого типа является предсказание поведения объектов на основе определенной известной о них информации. Если первые типы сетей работают в основном со статическими данными, то этот тип - с данными реального времени. Предикторы - наиболее «сложные» по своей структуре сети из-за огромной величины обучающей выборки и из-за необходимости постоянного обучения новым знаниям.

Применение нейронных сетей

Четвертый тип – **ассоциативная память**. Здесь используется способность сетей к автоматическому выявлению закономерностей в представляемых выборках и возможность изменения сетью параметров представленного примера в сторону увеличения степени распознавания. На этом принципе работают, например, системы удаления шума в обработке изображений и сигналов.

Применение нейронных сетей

Часто сети нескольких типов *интегрируют* в одну систему.

К примеру, в системах управления могут быть как предикторы, так и распознаватели. Объясняется это тем, что предиктор все время пытается найти значение предсказываемых величин, но иногда встречаются случаи, непредусмотренные и ранее не встречающиеся.

Предиктор в любом случае вычислит какой-либо результат, который с большой вероятностью окажется неправильным.

Распознаватель же, поняв, что ситуация является непредусмотренной, попытается применить другие методы управления.

Применение нейронных сетей

Перечисленные функциональные типы нейронных сетей применимы к следующим задачам:

- восстановление и распознавание изображений (ассоциативные массивы, классификаторы);
- экспертные системы и системы диагностики (классификаторы, fuzzy нейроны);
- системы управления и моделирования в реальном времени (классификаторы, предикторы, ассоциативная память);
- защита информации, шифрация и средства безопасности (предикторы, распознаватели);
- информационные и поисковые системы (предикторы, ассоциативная память, fuzzy нейроны). Построение так называемых интеллектуальных агентов для поиска нужной информации.

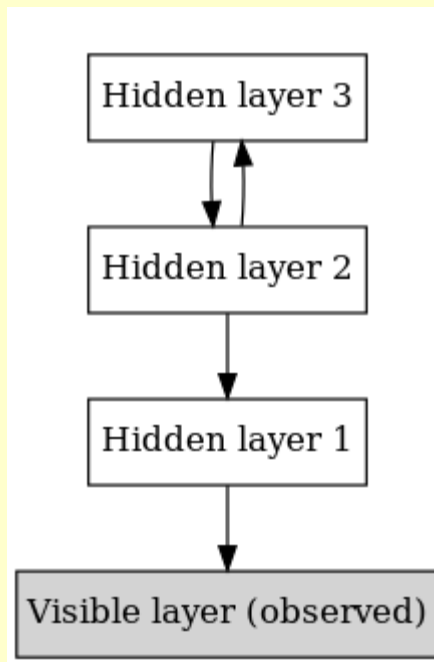
СОВРЕМЕННЫЕ АРХИТЕКТУРЫ НЕЙРОННЫХ СЕТЕЙ

В настоящее время существует несколько десятков топологий нейронных сетей:

- ❖ Глубокие сети доверия,
- ❖ Сверточные нейронные сети,
- ❖ Развертывающие нейронные сети,
- ❖ Глубокие сверточные обратные глубинные сети,
- ❖ Рекуррентные нейронные сети
- ❖ и др.

Глубокие сети доверия

Глубокая сеть доверия



Глубокая сеть доверия (deep belief network) — это глубинная нейронная сеть, состоящая из нескольких скрытых слоев, в которых нейроны внутри одного слоя не связаны друг с другом, но связаны с нейронами соседнего слоя.

Сверточные нейронные сети

Сверточная нейронная сеть



Идея сверточных нейронных сетей заключается в чередовании сверточных слоев, субдискретизирующих слоев (усреднения) и наличии полносвязных слоев на выходе.

Чередование слоев позволяет составлять карты признаков из карт признаков, что на практике означает способность распознавания сложных иерархий признаков.