

# Нейронные сети

## Обзор

Нейронные сети, или модели соединений, состоят из компьютерных аппаратных и программных средств, с помощью которых предпринимаются попытки копировать модели обработки информации биологическим мозгом. С вычислительной точки зрения нейронные сети представляют собой большое число вычислительных элементов, объединенных с еще большим числом других элементов, а детальные вычисления в нейронных сетях в значительной степени выполняются общими усилиями. Теория построения нейронных сетей включает в себя структуру сети (т. е. топологию нейронной сети), обучающий алгоритм и представление знаний. Нейронные сети уже начинают применять в науке, медицине, машиностроении и бизнесе (Widrow et al., 1994). В менеджменте нейронные сети используются при классификации, оценке регрессии, анализе временных рядов и при решении задач оптимизации (Masson & Wang, 1990).

## 1. Введение в нейронные сети

Один из подходов к созданию искусственного интеллекта состоит в проектировании механизмов имитации процесса передачи сигналов в биологическом мозге. На рис. 1 изображена упрощенная модель биологического мозга. Биологический мозг состоит из огромного числа нейронов. Нейроны действуют как процессоры передачи сигнала, они стимулируют другие нейроны и, в свою очередь, стимулируются ими. Аксоны и дендриты соединяют нейроны друг с другом с целью передачи сигналов. Место их соединения называется синапсом. Эту упрощенную модель можно считать метафорой для развития искусственных нейронных сетей, осуществляемого с использованием электрических и электронных аппаратных средств и программного обеспечения, как показано на рис. 2.

Нейроны (или узлы нейронных сетей) могут быть выполнены в виде компьютеров с высоким уровнем связи, которые действуют в параллельном режиме, и работа ведется одновременно над разными частями одной и той же задачи. Однако и отдельный компьютер может смоделировать нейронную сеть, выполняя алгоритм.

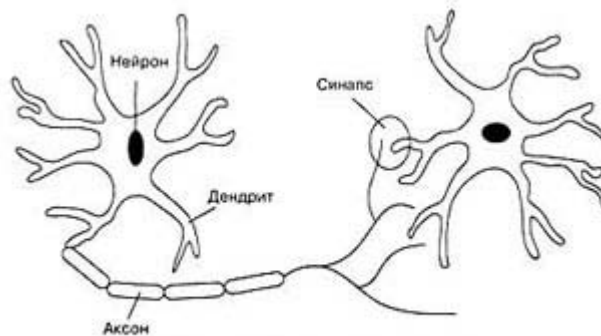


Рис. 1. Упрощенная биологическая модель мозга

Хотя ни одна из существующих сейчас технологий не может даже приблизиться к мощности биологического мозга, достаточно сложные искусственные нейронные сети уже моделировались на компьютерах; такие сети уже могут решать различные типы интеллектуальных задач, например распознавание образа, обработку изображения и управление роботом.

Первые исследования в области нейронных сетей проводились еще в первой половине XX в., когда психологи пытались выявить нейронную основу интеллекта. Работа МакКаллока и Питтса (McCulloch & Pitts, 1943) представила основные достижения в вычислениях и нейронауке за этот период. Уже к 1950 г. Хебб (Hebb, 1949) предложил простые правила (правила Хебба) ассоциативного обучения, которые легли в основу моделирования нейронных сетей в 1950-х гг.,

например перцептрона (в кибернетике – распознающий элемент) Розенблатта (Rosenblatt, 1959) и Аделаина Видроу (Widrow, 1960). В течение 1970-х и 1980-х гг. появилось большое количество моделей нейронных сетей, имеющих практическое применение. Коонен (Kohonen, 1989) постулировал ассоциативную память нейронных сетей. Хопфилд (Hopfield & Tank, 1986) предложил методы энергии для анализа автоматической ассоциативной памяти и сетей оптимизации. Румельхарт и его коллеги (Rumelhart et al., 1986) заново открыли обратную связь при обучении нейронных сетей. Прогресс современных информационных и компьютерных технологий в конце 1980-х гг. в большой степени был обусловлен исследованиями в области нейронных сетей. Начиная с первой международной конференции по нейронным сетям, проведенной IEEE (Некоммерческая техническая профессиональная ассоциация ученых и исследователей) в 1987 г. в Сан-Диего, Калифорния, которая отметила начало новой эры исследований в этой области, интерес к нейронным сетям значительно возрос. Исследования в области нейронных сетей все еще далеки от завершения, однако уже ожидается, что модели нейронных сетей будут использоваться и как модели реальных функций мозга, и как вычислительные устройства.

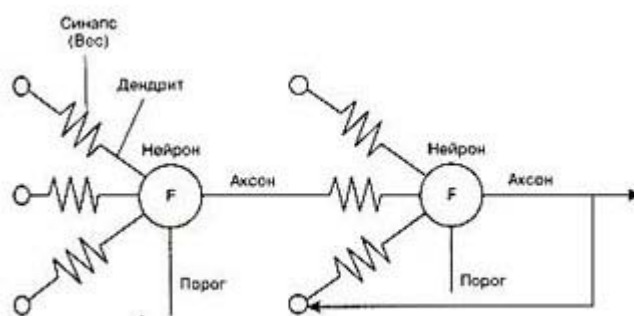


Рис. 2. Искусственная модель нейронной сети подражает биологическому мозгу

## 2. Контекст: представление знаний и обучение машины

Нейронные сети используются для представления знаний. В отличие от обычного вычисления представление знания в нейронных сетях выполняет поиск по содержанию, а не по адресу сохраненных данных. Кроме того, представление знаний в нейронных сетях осуществляется через приблизительное, а не абсолютно точное соответствие. Представление знаний в нейронных сетях состоит из сети, весов связей и семантических интерпретаций, присоединенных к активациям узлов. Например, в контексте управленческой классификации при использовании обученной нейронной сети можно предугадать, выберет ли клиент новый продукт, основываясь на выраженных в числах данных о клиенте, таких как последняя купленная марка, интерес к предварительному экспонированию, возможность дополнительного экспонирования и интерес к нему. Эти кванторные признаки атрибутов являются входами в обученную нейронную сеть. Активация «+1», полученная от нейронной сети, может указывать на то, что клиент выберет новое изделие, а «-1» — наоборот.

Обобщение знаний в нейронных сетях достигается путем обучения. Процесс обучения в нейронных сетях стимулирует желательные образцы активации и блокирует нежелательные, основываясь на доступных данных. Для достижения определенного обобщения знаний в нейронной сети разрабатывается алгоритм обучения. Функция ошибки, определенная на выходе нейронной сети, или энергетическая функция, определенная при активации элементов сети, характеризует качество нейронной сети в обобщении знаний. Обучающий набор данных в этом случае должен состоять из образцов представления знаний, которым предполагается обучить нейронную сеть. Алгоритм обучения действует методом изменения либо весов (т. е. силы связей между узлами), либо выходов нейронной сети, либо структуры нейронной сети, стремясь к минимальным ошибкам или энергии, основываясь на обучающих данных.

В системах нейронных сетей большое количество парадигм обучения. Обучение с учителем (контролируемое обучение) и обучение без учителя (неконтролируемое обучение) — вот две главные парадигмы, обычно используемые в проектировании обучающих алгоритмов. В парадигме обучения с учителем желаемый выход определяется обучающими образцами данных для каждого

образца входа. Процесс обучения пытается минимизировать «дистанцию» между фактическими и желаемыми выходами нейронной сети. Противоположностью обучения с учителем является обучение без учителя. Когда используется такая парадигма, подразумевается несколько образцов входа. Предполагается, что в процессе обучения нейронная сеть обнаруживает существенные особенности входов. В отличие от обучения с учителем здесь не существует априорного набора желаемых значений выхода. Нейронная сеть должна развить собственное представление стимулов входа без помощи учителя.

### 3. Анализ: типовые структуры нейронных сетей и обучающие алгоритмы

Модель нейронных сетей определяется структурой (или топологией) сети и обучающим алгоритмом. Существуют сотни видов структур нейронных сетей и соответствующих обучающих алгоритмов, и каждый из них предназначен для решения определенных типов проблем. Ниже представлены три типичные структуры нейронной сети и ее обучающего алгоритма.

#### (1) Многоуровневые нейронные сети с обратной связью при обучении

Одна из наиболее популярных нейронных сетей — нейронная сеть с обратным распространением ошибок с обучением на минимум среднеквадратической ошибки (BPLMS — back-propagation least mean square) (Rumelhart et. al., 1986). Ее топология показана на рис.3. Ребра графа соединяют элементы обработки. Каждому соединению в нейронной сети соответствует вес, указывающий на силу связи.

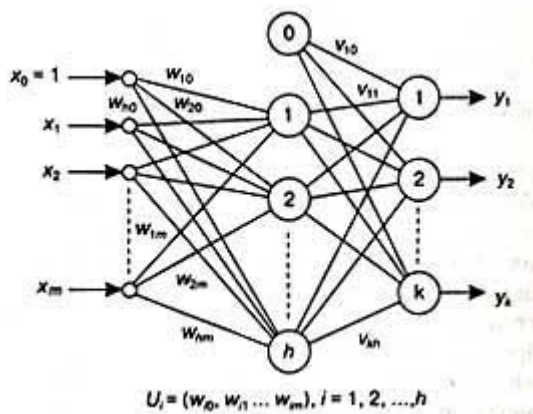


Рис.3

Обучающий алгоритм BPLMS - это интегральный градиентный алгоритм, предназначенный для минимизации среднеквадратической ошибки (разницы) между фактическим и желаемым выходами путем изменения весов сети. При использовании алгоритма BPLMS каждый узел принимает значение сигмоидной логистической функции:

$$\phi(I) = (1 + \exp[-(I - \theta)])^{-1}, \quad (1)$$

Где  $I$  — это сумма взвешенных входов в узел;  $\theta$  — порог — является произвольным числом, отличным от нуля. Нейронная сеть выполнит преобразование от вектора входа

$$X, X \in \Omega^n \rightarrow \text{до вектора выхода } Y, Y \in \mathfrak{R}^k \rightarrow \text{такое, что } Y = \Phi(X).$$

$$(2)$$

Используя обозначения, показанные на рис. 3, будем иметь

$$y_r = \left\{ \left[ 1 + \exp \left( - \left( v_{0r} + \sum_{i=1}^k v_{ir} (1 + \exp(-U_i X^r))^{-1} \right) \right) \right]^{-1} \right\} \quad (r=1 \dots k)$$

(3)

Обучающий набор данных обычно представляет собой набор примеров некоторого функционального преобразования данных, которым можно обучить нейронные сети с алгоритмом BPLMS. Полное объяснение механизма обучения BPLMS дают Румельхарт и его коллеги (Rumelhart et al., 1986). При использовании алгоритма BPLMS, многослойных нейронных сетей только с одним скрытым слоем и использовании сигмоидной функции узлы могут бесконечно приближаться к любой непрерывной функции (Cybenko, 1989). То есть в принципе для генерации произвольной функции не требуется более одного скрытого слоя. Алгоритм обратной связи при обучении нейронной сети приведен ниже.

Шаг 0. Установить топологию нейронной сети с обратной связью при обучении, как показано на рис. 3. Каждый узел принимает значение сигмоидной логистической функции. Установить параметр обучения  $\eta = \rho$ , где  $0 < \rho < 1$ .

Шаг 1. Инициализировать все веса маленькими (0-1) случайными значениями.

Шаг 2. Определить вектор входа  $x$ , который представляет собой точку обучающих данных в  $t$ -мерном пространстве, плюс порог. Определить желаемый выход  $y$  в соответствии с точкой обучающих данных.

Шаг 3. Вычислить фактический выход нейронной сети

$$y_r = \left\{ \left[ 1 + \exp \left( - \left( v_{0r} + \sum_{i=1}^k v_{ir} (1 + \exp(-U_i X^r))^{-1} \right) \right) \right]^{-1} \right\} \quad (r=1 \dots k)$$

(3)

Шаг 4. Если  $|y_{r-A} - y_{r-D}| < \epsilon$  при  $r = 1, \dots, k$ , где  $\epsilon$  — это заранее определенная малая положительная константа (например, 0,001), для всех точек данных обучения, то СТОП. В противном случае перейти к следующему шагу.

**Шаг 5.** Для узла выхода  $r$  вычислить  $\delta_D = y_{r-A} (1 - y_{r-A})(y_{r-D} - y_{r-A})$ . Изменить веса в соответствии с  $v_{ir}(t+1) = v_{ir}(t) + \eta \delta_D g_{ir}$ , где  $t$  — это время, а  $g_{ir}$  — это выход скрытого узла  $i$  к узлу выхода  $r$ .

Для каждого скрытого узла  $i$  вычислить

$$\delta_{iA} = \sum_r g_{ir} (1 - g_{ir}) \delta_D v_{ir}$$

Изменить веса

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_{iA} g_{ij}$$

где  $w_{ij}$  — это вес связи от скрытого узла  $i$  до входа узла  $j$  во время  $t$ , а  $g$  — это компонент входа в нейронную сеть. Переход к шагу 2.

## (2) Самоорганизующиеся карты

Самоорганизующиеся карты (SOFM - self-organizing feature map) (Kohonen, 1989) используются для обучения некоторым полезным свойствам, полученным с эталонных входов с помощью неконтролируемого или конкурентного обучения. Нейронная сеть, изображенная на рис. 4, представляет собой двухуровневую SOFM. Узлы нижнего уровня (узлы входа) получают входящие значения, представленные точками выборочных данных. Узлы верхнего уровня (узлы выхода)

будут давать карту организации моделей входа после неконтролируемого процесса обучения. На этом рисунке верхний уровень имеет вид одномерного множества, хотя в принципе он может быть представлен и в виде множества двумерного. Каждый узел нижнего уровня связан с каждым узлом верхнего уровня через связи переменного веса.

Неконтролируемый процесс обучения в SOFM вкратце можно описать следующим образом. Сначала веса связей принимают значения небольших случайных чисел. Поступающий входной вектор, представленный точками выборочных данных, образован узлами входа. Затем входной вектор через связи передается к выходным узлам. Активация узлов выхода зависит от входов. В отборе по принципу «победитель получает все» становится активным выходной узел с весами, наиболее близкими к входному вектору. На стадии обучения веса обновляются согласно следующему правилу:

$$W(\text{новый}) = W(\text{старый}) + [\eta \cdot X - w(\text{старый})] \quad (4)$$

где  $W$  — это матрица весов;  $X$  — вектор входа, а  $\eta$  — параметр обучения ( $0 < \eta < 1$ ), который уменьшается с течением времени.

Обновление весов происходит только для активного выходного узла и его топологических соседей (рис. 4). Вначале окрестность велика, а затем она постепенно уменьшается. Когда параметр обучения  $\eta$  уменьшится до нуля при неконтролируемом обучении SOFM, процесс обучения остановится.

После представления достаточного количества входных векторов с помощью весов можно будет определить такие кластеры, в которых локальная функция плотности центров кластеров будет иметь тенденцию бесконечно приближаться к вероятностной функции плотности входных векторов. Веса будут организованы таким образом, что узлы, участие которых в топологическом сходстве чувствительно к значению входа, будут подобными. Выходные узлы в SOFM, таким образом, будут организованы и представлены в виде кластеров в самоорганизующейся карте.

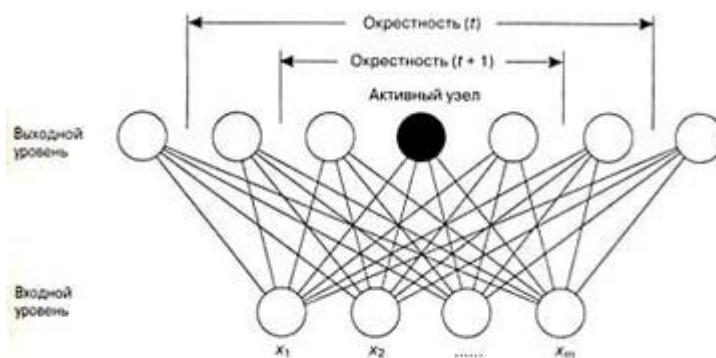


Рис. 4. Самоорганизующиеся карты Коопенга (Kohonen)

Самоорганизующиеся карты являются динамическими системами, которые изучают топологические отношения и абстрактные структуры в многомерных входных векторах с помощью представления в пространстве меньшей размерности. Такое представление можно назвать приближением локальной (вместо глобальной) плотности вероятности входных векторов. То есть хотя маломерная самоорганизующаяся карта не может дать реальную плотность вероятности векторов входа в полном многомерном пространстве, она может показать относительную плотность вероятности ограниченной области по сравнению с ее окрестностью.

Типичный алгоритм самоорганизующихся карт можно представить следующим образом.

Шаг 0. Инициализировать веса от  $M$  входов к  $N$  выходным узлам малыми случайными значениями. Установить начальную функцию окрестности  $nei(t=0)=k$ , где  $t$  — время, а  $k$  — произвольное число (например,  $k=N$ ). Установить начальный параметр обучения  $\eta(t=0)=p$ , где  $0 < p < 1$ .

Шаг 1. Представить наблюдение.

Шаг 2. Вычислить

$$D_j = \sum_{i=1}^M (x_i(t) - w_{ij}(t)),$$

где  $x(t)$  является входом в узел  $i$  в момент времени  $t$ , а  $w(t)$  — вес связи входного узла  $i$  и выходного узла  $j$  в момент времени  $t$ .

Шаг 3. Выбрать победителя, т. е. такой выходной узел  $J$ , что  $D = \min(D)$

Шаг 4. Определить множество

$$nei_j(t) = \{j \mid \max[1, (J - nei(t))] \leq j \leq [(J + nei(t)), k]\},$$

Обновить веса следующим образом:

$$w_{ij}(t+1) = [w_{ij}(t) + \eta(t)(x_i(t) - w_{ij}(t))],$$

для  $j \in nei_j(t)$ , и  $1 < i < M$ ;  $w_{ij}(t+1) = w_{ij}(t)$  в противном случае.

**Шаг 5.** Если  $\eta(t) < \varepsilon$ , где  $\varepsilon$  — это заранее определенная малая положительная константа (например, 0.0001), тогда СТОП; в противном случае вычислить

$$\eta(t) = \delta_1 \eta(t) \text{ и } nei(t) = \delta_2 nei(t),$$

где  $0 < \delta_1 < 1$  и  $0 < \delta_2 < 1$ , а  $\delta_1$  и  $\delta_2$  определены так, что  $nei(t) \rightarrow 0$ , если  $\eta(t) \rightarrow \varepsilon$ .

Установить  $t = t + 1$  и перейти к шагу 2.

### (3) Нейронные сети Хопфилда

Топология нейронных сетей Хопфилда показывается на рис. 5. Существует множество версий сетей Хопфилда (Hopfield, 1982, 1984). Нейронные сети Хопфилда широко использовались в качестве ассоциативной памяти или для решения задач оптимизации. Нейронная сеть Хопфилда состоит из  $N$  узлов. Выход каждого узла  $i$  передает значения всем другим узлам  $j$  ( $j=1 \dots i-1, i=1 \dots N$ ) через веса  $W$ . В первоначальной версии нейронной сети Хопфилда каждый узел формирует взвешенную сумму  $N - 1$  входов и пропускает результат выхода через жесткую ограничивающую нелинейную функцию, т. е.

$$f_n(I) = \begin{cases} +1 & \text{если } I \geq 0; \\ -1 & \text{в противном случае,} \end{cases}$$

где  $I$  является суммой взвешенных входов в узел.

Контролируемый процесс обучения в этом типе нейронной сети Хопфилда можно представить так. Сначала с использованием данной формулы на основе известной модели устанавливаются веса. Затем на сеть подается неизвестный входной образ, что вынуждает выход сети соответствовать неизвестной модели входа. Затем сеть, используя данную формулу, выполняет дискретные временные шаги. Когда выходные узлы остаются неизменными, сеть рассматривается как сходящаяся. После этого выходные узлы представляют собой модель, которая наилучшим образом соответствует неизвестной модели входа.

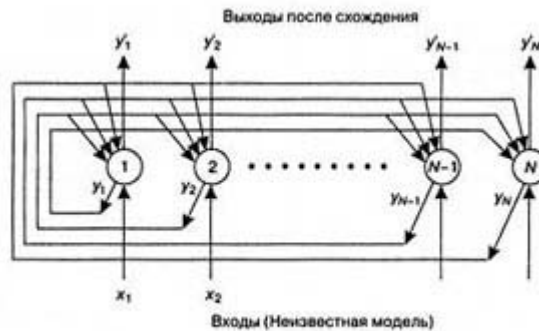


Рис. 5. Модель нейронной сети Хопфилда

Алгоритм действий первоначальной версии нейронных сетей Хопфилда можно объяснить следующим образом.

Шаг 0. Определить топологию нейронной сети Хопфилда, как показано на рис. 5. Каждый узел принимает значение жесткой ограничивающей нелинейной функции.

Шаг 1. Назначить веса связи следующим образом:

$$w_{ij} = \begin{cases} \sum_{p=1}^M x_i^p x_j^p & i \neq j \\ 0 & i = j \end{cases} \quad (1 \leq i, j \leq N).$$

где  $w$  - вес связи между узлом  $i$  и узлом  $j$ , а  $x$ , который может принимать значения  $+1$  или  $-1$ , является измерением  $i$  экземпляра модели  $p$ .

Шаг 2. Установить неизвестную модель входа следующим образом:  $y(t=0) = x(1 < i < N)$

Шаг 3. Повторять следующее действие до схождения:

$$y_j(t+1) = f_{Ht} \left[ \sum_{i=1}^N w_{ij} y_i(t) \right], \quad (1 \leq j \leq N).$$

где  $f$  - жесткая ограничивающая нелинейная функция. Фактически сеть Хопфилда сходится, когда веса симметричны (т. е.  $W=W^T$ ). Хопфилд (Hopfield, 1984) показал, что сеть сходится также в том случае, если градуированная функция нелинейности является сигмоидной.

#### 4. Оценки: типичные примеры применения нейронных сетей в менеджменте

В течение последнего десятилетия в журналах и газетах, таких как «Management Science» («Наука менеджмента»), «IEEE Transactions on Systems» («Транзакции IEEE в системах»), «Man and Cybernetics» («Человек и кибернетика»), «Decision Sciences» («Наука принятия решений»), «Computers & Operations Research» («Компьютерные и операционные исследования»), «European Journal of Operational Research» («Европейский журнал исследования операций»), «AI Expert» («ИИ-эксперт»), «AI Magazine» («ИИ-журнал») и «Business Week» («Неделя бизнеса»); были напечатаны бесчисленные предложения по применению нейронных сетей в бизнесе и исследовании операций. Большинство вариантов применения нейронных сетей в менеджменте касаются задач, попадающих в следующие четыре категории: классификация (распознавание образца), построение эмпирической кривой и анализ временных рядов, кластеризация и оптимизация. Ниже приведены примеры каждой категории.

1. Классификация. Должным образом разработанная нейронная сеть может использоваться как классификатор. После обучения историческим данным нейронная сеть может определять класс принадлежности некоторой характерной черты. Нейронные сети можно использовать при анализе кредитоспособности, чтобы предсказать банкротство фирмы. Нейронные сети могут также оценить активы и обязательства. Во многих банках нейронные сети можно использовать для обнаружения подделки кредитной карточки.

2. Построение эмпирической кривой и анализ временных рядов. Процесс обучения во многих типах нейронных сетей может рассматриваться как построение эмпирической кривой. Кроме того, нейронные сети могут использоваться для определения модели колебаний временного ряда. Аналитики сферы маркетинга, используя нейронные методы сети, могут определять рыночные функции отклика, основанные на временных данных. Руководители производства могут предсказывать производительность фирмы, основываясь на кривых, представленных обученными нейронными сетями. Многие финансовые учреждения используют нейронные сети для финансового прогноза и управления инвестициями.

3. Кластеризация. Неконтролируемые обучающиеся нейронные сети обычно используются в кластерном анализе для группировки объектов без априорного знания классов. Идентификация потребительских сегментов и группировка технологических деталей могут служить примером в этой прикладной категории.

4. Оптимизация. Поскольку процесс обучения в нейронных сетях должен минимизировать заранее определенную ошибку или энергию, нейронные сети могут использоваться для решения задач оптимизации. Задачи вроде оптимального планирования работ, оптимального планирования работы магазина и минимизации потерь могут быть решены с использованием нейронных сетей.

## 5. Заключение

Вообще говоря, человек все свои знания первоначально получает из реального мира. Нейронные сети способны помочь людям в генерации знаний, которые основывались бы на всех первоначальных данных. Исследования в области нейронных сетей в основном достаточно наглядны. По сравнению с другими вычислительными методами в статистике и науке менеджмента они имеют значительные преимущества. Так, у моделей на основе нейронных сетей очень гибкие теоретические требования; кроме того, им необходимы совсем небольшие объемы предварительных знаний относительно формирования задачи.

Как мощный механизм обучения нейронные сети могут широко применяться в различных областях. Существует, однако, возможность недоразумений в оценке методик машинного обучения. Они никогда не смогут полностью заменить людей в процессе решения задачи. Нейронные сети должны использоваться для обобщения данных, а не для определения, атрибуты и критерии которого весьма важны при сборе данных. Нейронные сети адаптивны по своей природе, они могут подражать решению проблемы человеком, но они не сообщают нам, какой из критериев решения задачи должен быть принят во внимание перед сбором данных. Кроме того, обучающиеся машины часто используются при формализации знаний из данных реального мира, но сами обучающиеся машины не могут генерировать принципы формализации.



## Литература

- Cybenko, G (1989), Approximation by superpositions of a sigmoidal function, Mathematics of Control, Signals and Systems 2:303-14.
- Hebb, D.O. (1949) The Organization of Behavior, New York: John Wiley & Sons.
- Hopfield, J.J. (1982) "Neural networks and physical systems with emergent collective computational abilities", in Proceedings of National Academy of Sciences, USA, 79: 2554-8.
- Hopfield, J.J. (1984) "Neurons with graded response have collective computational properties like those of two-state neurons", in Proceedings of National Academy of Sciences, USA, 81: 3088-92.
- Hopfield, J.J. and Tank, D.W. (1986) "Computing with neural circuits", Sciences 233: 625-33.
- Kohonen, T. (1989) Self-Organization and Associative Memory, 3rd edn, Berlin: Springer-Verlag.
- Masson, E. and Wang, Y. (1990) "Introduction to computation and learning in artificial neural networks", European Journal of Operational Research 47:1-28.
- McCulloch, W.S. and Pitts, W. (1943), A logical calculus of the ideas imminent in nervous activity, Bulletin of Mathematical Biophysics 5:115-33.
- Rosenblatt, R. (1959) Principles of Neurodynamics, New York: Spartan Books.
- Rumelhart, D. McClelland, J. and PDP Research Group (1986) Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations, Cambridge, MA: The MIT Press.
- Widrow, B. and Hoff, M.E. (1960), Adaptive switching circuits, 1960 IRE WESCON Convention Record, Part 4:96 -104.
- Widrow, B. Rumelhart, D.E. and Lehr, M.A. (1994) "Neural networks: Application in industry, business and science", Communications of the ACM 37 (3) : 93-105