

## Нейронные сети: основные положения

*В статье рассмотрены основы теории нейронных сетей, позволяющие в дальнейшем обратиться к конкретным структурам, алгоритмам и идеологии практического применения сетей в компьютерных приложениях.*

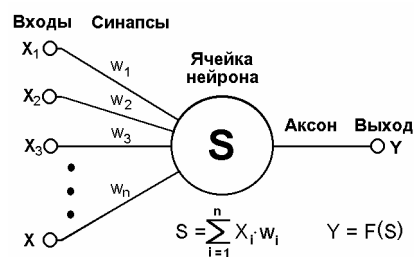
В последние десятилетия в мире бурно развивается новая прикладная область математики, специализирующаяся на искусственных нейронных сетях (НС). Актуальность исследований в этом направлении подтверждается массой различных применений НС. Это автоматизация процессов распознавания образов, адаптивное управление, аппроксимация функционалов, прогнозирование, создание экспертных систем, организация ассоциативной памяти и многие другие приложения. С помощью НС можно, например, предсказывать показатели биржевого рынка, выполнять распознавание оптических или звуковых сигналов, создавать самообучающиеся системы, способные управлять автомашиной при парковке или синтезировать речь по тексту. В то время как на западе применение НС уже достаточно обширно, у нас это еще в некоторой степени экзотика – российские фирмы, использующие НС в практических целях, наперечет [1].

Широкий круг задач, решаемый НС, не позволяет в настоящее время создавать универсальные, мощные сети, вынуждая разрабатывать специализированные НС, функционирующие по различным алгоритмам.

Модели НС могут быть программного и аппаратного исполнения. В дальнейшем речь пойдет в основном о первом типе.

Несмотря на существенные различия, отдельные типы НС обладают несколькими общими чертами.

Во-первых, основу каждой НС составляют относительно простые, в большинстве случаев – однотипные, элементы (ячейки), имитирующие работу нейронов мозга. Далее под нейроном будет подразумеваться искусственный нейрон, то есть ячейка НС. Каждый нейрон характеризуется своим текущим состоянием по аналогии с нервными клетками головного мозга, которые могут быть возбуждены или заторможены. Он обладает группой синапсов – однонаправленных входных связей, соединенных с выходами других нейронов, а также имеет аксон – выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов. Общий вид нейрона приведен на рисунке 1. Каждый синапс характеризуется величиной синаптической связи или ее весом  $w_i$ , который по физическому смыслу эквивалентен электрической проводимости.



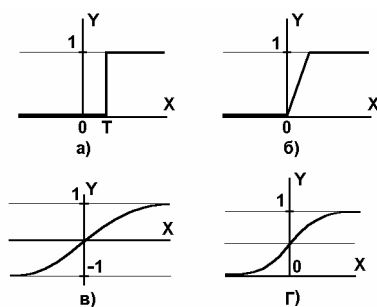
**Рис.1 Искусственный нейрон**

Текущее состояние нейрона определяется, как взвешенная сумма его входов:

$$s = \sum_{i=1}^n x_i \cdot w_i \quad (1)$$

Выход нейрона есть функция его состояния:

$$y = f(s) \quad (2)$$



**Рис.2 а) функция единичного скачка; б) линейный порог (гистерезис); в) сигмоид – гиперболический тангенс; г) сигмоид – формула (3)**

Нелинейная функция  $f$  называется активационной и может иметь различный вид, как показано на рисунке 2. Одной из наиболее распространенных является нелинейная функция с насыщением, так называемая логистическая функция или сигмоид (т.е. функция S-образного вида)[2]:

$$f(x) = \frac{1}{1 + e^{-\alpha x}} \quad (3)$$

При уменьшении  $\alpha$  сигмоид становится более пологим, в пределе при  $\alpha=0$  вырождаясь в горизонтальную линию на уровне 0.5, при увеличении  $\alpha$  сигмоид приближается по внешнему виду к функции единичного скачка с порогом  $T$  в точке  $x=0$ . Из выражения для сигмоида очевидно, что выходное значение нейрона лежит в диапазоне  $[0,1]$ . Одно из ценных свойств сигмоидной функции – простое выражение для ее производной, применение которого будет рассмотрено в дальнейшем.

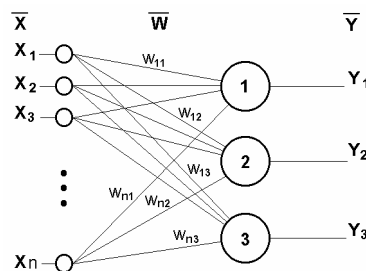
$$f'(x) = \alpha \cdot f(x) \cdot (1 - f(x)) \quad (4)$$

Следует отметить, что сигмоидная функция дифференцируема на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того она обладает свойством усиливать слабые сигналы лучше, чем большие, и предотвращает насыщение от больших сигналов, так как они соответствуют областям аргументов, где сигмоид имеет пологий наклон.

Возвращаясь к общим чертам, присущим всем НС, отметим, во-вторых, принцип параллельной обработки сигналов, который достигается путем объединения большого числа нейронов в так называемые слои и соединения определенным образом нейронов различных слоев, а также, в некоторых конфигурациях, и нейронов одного слоя между собой, причем обработка взаимодействия всех нейронов ведется послойно.

В качестве примера простейшей НС рассмотрим трехнейронный перцептрон (рис.3), то есть такую сеть, нейроны которой имеют активационную функцию в виде единичного скачка\*. На  $n$  входов поступают некие сигналы, проходящие по синапсам на 3 нейрона, образующие единственный слой этой НС и выдающие три выходных сигнала:

$$y_j = f\left[\sum_{i=1}^n x_i \cdot w_{ij}\right], \quad j=1...3 \quad (5)$$



**Рис.3 Однослойный перцептрон**

Очевидно, что все весовые коэффициенты синапсов одного слоя нейронов можно свести в матрицу  $\mathbf{W}$ , в которой каждый элемент  $w_{ij}$  задает величину  $i$ -ой синаптической связи  $j$ -ого нейрона. Таким образом, процесс, происходящий в НС, может быть записан в матричной форме:

$$\mathbf{Y} = \mathbf{F}(\mathbf{XW}) \quad (6)$$

где  $\mathbf{X}$  и  $\mathbf{Y}$  – соответственно входной и выходной сигнальные векторы,  $\mathbf{F}(\mathbf{V})$  – активационная функция, применяемая поэлементно к компонентам вектора  $\mathbf{V}$ .

\* Иногда перцептроном называют любую НС слоистой структуры, однако здесь и далее под перцептроном понимается только сеть, состоящая из нейронов с активационными функциями единичного скачка (бинарная сеть).

Теоретически число слоев и число нейронов в каждом слое может быть произвольным, однако фактически оно ограничено ресурсами компьютера или специализированной микросхемы, на которых обычно реализуется НС. Чем сложнее НС, тем масштабнее задачи, подвластные ей.

Выбор структуры НС осуществляется в соответствии с особенностями и сложностью задачи. Для решения некоторых отдельных типов задач уже существуют оптимальные, на сегодняшний день, конфигурации, описанные, например, в [2],[3],[4] и других изданиях, перечисленных в конце статьи. Если же задача не может быть сведена ни к одному из известных типов, разработчику приходится решать сложную проблему синтеза новой конфигурации. При этом он руководствуется несколькими основополагающими принципами: возможности сети возрастают с увеличением числа ячеек сети, плотности связей между ними и числом выделенных слоев (влияние числа слоев на способность сети выполнять классификацию плоских образов показано на рис.4 из [5]); введение обратных связей наряду с увеличением возможностей сети поднимает вопрос о динамической устойчивости сети; сложность алгоритмов функционирования сети (в том числе, например, введение нескольких типов синапсов – возбуждающих, тормозящих и др.) также способствует усилению мощи НС. Вопрос о необходимых и достаточных свойствах сети для решения того или иного рода задач представляет собой целое направление нейрокомпьютерной науки. Так как проблема синтеза НС сильно зависит от решаемой задачи, дать общие подробные рекомендации затруднительно. В большинстве случаев оптимальный вариант получается на основе интуитивного подбора.

Очевидно, что процесс функционирования НС, то есть сущность действий, которые она способна выполнять, зависит от величин синаптических связей, поэтому, задавшись определенной структурой НС, отвечающей какой-либо задаче, разработчик сети должен найти оптимальные значения всех переменных весовых коэффициентов (некоторые синаптические связи могут быть постоянными).

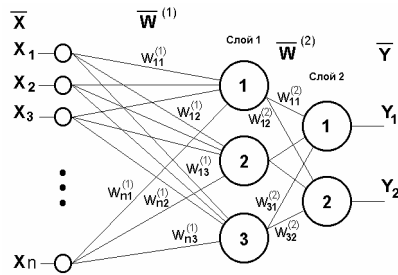
Этот этап называется обучением НС, и от того, насколько качественно он будет выполнен, зависит способность сети решать поставленные перед ней проблемы во время эксплуатации. На этапе обучения кроме параметра качества подбора весов важную роль играет время обучения. Как правило, эти два параметра связаны обратной зависимостью и их приходится выбирать на основе компромисса.

Обучение НС может вестись с учителем или без него. В первом случае сети предъявляются значения как входных, так и желательных выходных сигналов, и она по некоторому внутреннему алгоритму подстраивает веса своих синаптических связей. Во втором случае выходы НС формируются самостоятельно, а веса изменяются по алгоритму, учитывающему только входные и производные от них сигналы.

Существует великое множество различных алгоритмов обучения, которые однако делятся на два больших класса: детерминистские и стохастические. В первом из них подстройка весов представляет собой жесткую последовательность действий, во втором – она производится на основе действий, подчиняющихся некоторому случайному процессу.

Развивая дальше вопрос о возможной классификации НС, важно отметить существование бинарных и аналоговых сетей. Первые из них оперируют с двоичными сигналами, и выход каждого нейрона может принимать только два значения: логический ноль ("заторможенное" состояние) и логическая единица ("возбужденное" состояние). К этому классу сетей относится и рассмотренный выше перцептрон, так как выходы его нейронов, формируемые функцией единичного скачка, равны либо 0, либо 1. В аналоговых сетях выходные значения нейронов способны принимать непрерывные значения, что могло бы иметь место после замены активационной функции нейронов перцептрона на сигмоид.

Еще одна классификация делит НС на синхронные и асинхронные[3]. В первом случае в каждый момент времени свое состояние меняет лишь один нейрон. Во втором – состояние меняется сразу у целой группы нейронов, как правило, у всего слоя. Алгоритмически ход времени в НС задается итерационным выполнением однотипных действий над нейронами. Далее будут рассматриваться только синхронные НС.



**Рис.4 Двухслойный перцептрон**

Сети также можно классифицировать по числу слоев. На рисунке 4 представлен двухслойный перцептрон, полученный из перцептрона с рисунка 3 путем добавления второго слоя, состоящего из двух нейронов. Здесь уместно отметить важную роль нелинейности активационной функции, так как, если бы она не обладала данным свойством или не входила в алгоритм работы каждого нейрона, результат функционирования любой р-слойной НС с весовыми матрицами  $\mathbf{W}^{(i)}$ ,  $i=1,2,\dots,p$  для каждого слоя  $i$  сводился бы к перемножению входного вектора сигналов  $\mathbf{X}$  на матрицу

$$\mathbf{W}^{(\Sigma)} = \mathbf{W}^{(1)} \cdot \mathbf{W}^{(2)} \cdot \dots \cdot \mathbf{W}^{(p)} \quad (7)$$

то есть фактически такая р-слойная НС эквивалентна однослойной НС с весовой матрицей единственного слоя  $\mathbf{W}^{(\Sigma)}$ :

$$\mathbf{Y} = \mathbf{XW}^{(\Sigma)} \quad (8)$$

Продолжая разговор о нелинейности, можно отметить, что она иногда вводится и в синаптические связи. Большинство известных на сегодняшний день НС используют для нахождения взвешенной суммы входов нейрона формулу (1), однако в некоторых приложениях НС полезно ввести другую запись, например:

$$s = \sum_{i=1}^n x_i^2 \cdot w_i \quad (9)$$

или даже

$$s = \sum_{i=1}^n x_i \cdot x_{((i+1) \bmod n)} \cdot w_i \quad (10)$$

Вопрос в том, чтобы разработчик НС четко понимал, для чего он это делает, какими ценными свойствами он тем самым дополнительно наделяет нейрон, и каких лишает. Введение такого рода нелинейности, вообще говоря, увеличивает вычислительную мощь сети, то есть позволяет из меньшего числа нейронов с "нелинейными" синапсами сконструировать НС, выполняющую работу обычной НС с большим числом стандартных нейронов и более сложной конфигурации[4].

Многообразие существующих структур НС позволяет отыскать и другие критерии для их классификации, но они выходят за рамки данной статьи.

Теперь рассмотрим один нюанс, преднамеренно опущенный ранее. Из рисунка функции единичного скачка видно, что пороговое значение  $T$ , в общем случае, может принимать произвольное значение. Более того, оно должно принимать некое произвольное, неизвестное заранее значение, которое подбирается на стадии обучения вместе с весовыми коэффициентами. То же самое относится и к центральной точке сигмоидной зависимости, которая может сдвигаться вправо или влево по оси  $X$ , а также и ко всем другим активационным функциям. Это, однако, не отражено в формуле (1), которая должна была бы выглядеть так:

$$s = \sum_{i=1}^n x_i \cdot w_i - T \quad (11)$$

Дело в том, что такое смещение обычно вводится путем добавления к слою нейронов еще одного входа, возбуждающего дополнительный синапс каждого из нейронов, значение которого всегда равняется 1. Присвоим этому входу номер 0. Тогда

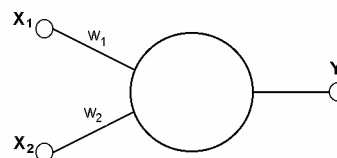
$$s = \sum_{i=0}^n x_i \cdot w_i \quad (12)$$

где  $w_0 = -T$ ,  $x_0 = 1$ .

Очевидно, что различие формул (1) и (12) состоит лишь в способе нумерации входов.

Из всех активационных функций, изображенных на рисунке 2, одна выделяется особо. Это гиперболический тангенс, зависимость которого симметрична относительно оси X и лежит в диапазоне  $[-1, 1]$ . Забегая вперед, скажем, что выбор области возможных значений выходов нейронов во многом зависит от конкретного типа НС и является вопросом реализации, так как манипуляции с ней влияют на различные показатели эффективности сети, зачастую не изменяя общую логику ее работы. Пример, иллюстрирующий данный аспект, будет представлен после перехода от общего описания к конкретным типам НС.

Какие задачи может решать НС? Грубо говоря, работа всех сетей сводится к классификации (обобщению) входных сигналов, принадлежащих  $n$ -мерному гиперпространству, по некоторому числу классов. С математической точки зрения это происходит путем разбиения гиперпространства гиперплоскостями (запись для случая однослойного перцептрона)



**Рис.5 Однонейронный перцептрон**

$$\sum_{i=1}^n x_i \cdot w_{ik} = T_k, \quad k=1...m \quad (13)$$

Каждая полученная область является областью определения отдельного класса. Число таких классов для одной НС перцептронного типа не превышает  $2^m$ , где  $m$  – число выходов сети. Однако не все из них могут быть разделены данной НС.

Например, однослойный перцептрон, состоящий из одного нейрона с двумя входами, представленный на рисунке 5, не способен разделить плоскость (двумерное гиперпространство) на две полуплоскости так, чтобы осуществить классификацию входных сигналов по классам А и В (см. таблицу 1).

**Таблица 1**

$x_1$	$x_2$	0	1
0		А	В
1		В	А

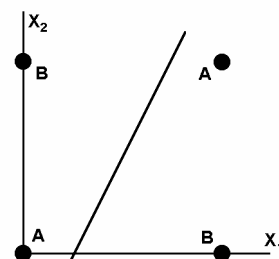
Уравнение сети для этого случая

$$x_1 \cdot w_1 + x_2 \cdot w_2 = T \quad (14)$$

является уравнением прямой (одномерной гиперплоскости), которая ни при каких условиях не может разделить плоскость так, чтобы точки из множества входных сигналов, принадлежащие разным классам, оказались по разные стороны от прямой (см. рисунок 6).

Если присмотреться к таблице 1, можно заметить, что данное разбиение на классы реализует логическую функцию исключающего ИЛИ для входных сигналов. Невозможность реализации однослойным перцептроном этой функции получила название проблемы исключающего ИЛИ.

Функции, которые не реализуются однослойной сетью, называются линейно неразделимыми[2]. Решение задач, подпадающих под это ограничение, заключается в применении 2-х и более слойных сетей или сетей с нелинейными синапсами, однако и тогда существует вероятность, что корректное разделение некоторых входных сигналов на классы невозможно.



**Рис.6 Визуальное представление работы НС с рисунка 5**

Наконец, мы можем более подробно рассмотреть вопрос обучения НС, для начала – на примере перцептрона с рисунка 3.

Рассмотрим алгоритм обучения с учителем[2][4].

1. Проинициализировать элементы весовой матрицы (обычно небольшими случайными значениями).
2. Подать на входы один из входных векторов, которые сеть должна научиться различать, и вычислить ее выход.
3. Если выход правильный, перейти на шаг 4.

Иначе вычислить разницу между идеальным и полученным значениями выхода:

$$\delta = Y_i - Y$$

Модифицировать веса в соответствии с формулой:

$$w_{ij}(t+1) = w_{ij}(t) + v \cdot \delta \cdot x_i$$

где  $t$  и  $t+1$  – номера соответственно текущей и следующей итераций;  $v$  – коэффициент скорости обучения,  $0 < v \ll 1$ ;  $i$  – номер входа;  $j$  – номер нейрона в слое.

Очевидно, что если  $Y_i > Y$  весовые коэффициенты будут увеличены и тем самым уменьшат ошибку. В противном случае они будут уменьшены, и  $Y$  тоже уменьшится, приближаясь к  $Y_i$ .

4. Цикл с шага 2, пока сеть не перестанет ошибаться.

На втором шаге на разных итерациях поочередно в случайном порядке предъявляются все возможные входные вектора. К сожалению, нельзя заранее определить число итераций, которые потребуется выполнить, а в некоторых случаях и гарантировать полный успех. Этот вопрос будет косвенно затронут в дальнейшем.

В завершении данной вводной статьи хотелось бы отметить, что дальнейшее рассмотрение НС будет в основном тяготеть к таким применениям, как распознавание образов, их классификация и, в незначительной степени, сжатие информации. Более подробно об этих и других применениях и реализующих их структурах НС можно прочитать в журналах *Neural Computation*, *Neural Computing and Applications*, *Neural Networks*, *IEEE Transactions on Neural Networks*, *IEEE Transactions on System, Man, and Cybernetics* и других.

## Литература

1. Е. Монахова, "Нейрохирурги" с Ордынки, *PC Week/RE*, №9, 1995.
2. Ф.Уоссермен, *Нейрокомпьютерная техника*, М., Мир, 1992.
3. Итоги науки и техники: физические и математические модели нейронных сетей, том 1, М., изд. ВИНТИ, 1990.
4. *Artificial Neural Networks: Concepts and Theory*, IEEE Computer Society Press, 1992.
5. Richard P. Lippmann, *An Introduction to Computing with Neural Nets*, *IEEE Acoustics, Speech, and Signal Processing Magazine*, April 1987.

