# МНОГОСЛОЙНЫЕ НЕЙРОННЫЕ СЕТИ. МЕТОД ОБРАТНОГО РАСПРОСТРАНЕНИЯ ОШИБКИ

## 1. Теоретическое введение

### *1.1. Основные понятия и определения*

Архитектура многослойной нейронной сети представлена на рис. 1. Все нейроны сети объединены в группы, называемые слоями. За каждый такт дискретного времени сигнал передается от слоя к слою, пока не достигнет выходного (последнего) слоя, на котором фиксируется реакция сети. Входной (нулевой) слой не выполняет какой-либо вычислительной функции и служит исключительно для распределения входного векторного сигнала $x = (x_1, x_2, \ldots, x_{N_0})$ на нейроны первого слоя. В многослойной нейронной сети сигнал распространяется только в прямом направлении: обратные связи и связи между нейронами одного слоя отсутствуют. В связи с этим представленная на рис. 1 сеть называется многослойной нейронной сетью прямого распространения (multilayer neural network).

Рабочие слои сети, расположенные между нулевым (распределительным) и выходным слоями, называются скрытыми (hidden layers). Состояние $q$-го слоя характеризуется вектором $s^q$, $q = \overline{0,K}$. В соответствии с обозначением входного векторного сигнала справедливо $s^0 = x$. Число нейронов в $q$-ом слое принимается равным $N_q$, так что $s^q = (s_1^q, s_2^q, \ldots, s_{N_q}^q)$, $q = \overline{0,K}$. Здесь $s_i^q$, $q = \overline{0,K}$, $i = \overline{1,N_q}$, означает выход $i$- го нейрона $q$-го слоя.

Функционирование каждого слоя сети определяется значениями матрицы $W$ синаптических коэффициентов и вектора $-b$ смещений нейронов. В $q$-ом слое, $q = \overline{1,K}$, матрица $W^q$ имеет размерность $[N_q, N_{q-1}]$, а вектор $b^q = (b_1^q, b_{2,}^q, \ldots, b_{N_q}^q)$. Расчет прохождения сигнала в сети прямого распространения выполняется по следующим формулам:

$$\begin{cases} s_i^q = f_q(h_i^q), \\ h_i^q = \sum\limits_{j=1}^{N_{q-1}} w_{ij}^q \, s_j^{q-1} - b_i^q, \end{cases} \tag{1.1}$$

$$q = \overline{1,K}\,,\ i = \overline{1,N_q}\,,$$

где $h_i^q$ — потенциал $i$-го нейрона $q$-го слоя; $f_q(h)$ — функция активации нейронов $q$-го слоя (предполагается, что все нейроны одного слоя имеют одинаковые функции активации). К функциям активации $f_q(h)$, $q = \overline{1,K}$, предъявляется требование непрерывности и дифференцируемости. “Жесткие” пороговые функции активации не допускаются. Под $w_{ij}^q$ в формуле (1.1) понимается элемент матрицы $W^q$ синаптических коэффициентов $q$-го слоя.

Удобно преобразовать математическую модель (1.1) путем введения одного дополнительного нейрона в каждый слой МНС. Отметим эти нейроны нулевым индексом и положим их состояние тождественно равным 1, так что $s_0^q = 1$ для $q = \overline{0,K}$. За счет введенных нейронов естественно расширяются вектора $s^q$, $q = \overline{0,K}$, состояний нейронов в слоях:

$$\tilde{s}^{\,q} = (\, s_0^q, s_1^q, s_2^q,\ \ldots,\ s_{N_q}^q\,),\ q = \overline{0,K}\,. \tag{1.2}$$

Расширим также матрицу синаптических коэффициентов в $q$-ом слое ($q = \overline{1,K}$), дополнив ее нулевым столбцом, содержащим смещения нейронов $q$-го слоя:

$$\tilde{W}^q = \begin{pmatrix} -b_1^q & w_{11}^q & w_{12}^q & \ldots w_{1N_{q-1}}^q \\ -b_2^q & w_{21}^q & w_{22}^q & \ldots w_{2N_{q-1}}^q \\ . & . & . & . \\ -b_{N_q}^q & w_{N_q1}^q & w_{N_q2}^q & \ldots w_{N_qN_{q-1}}^q \end{pmatrix} \tag{1.3}$$

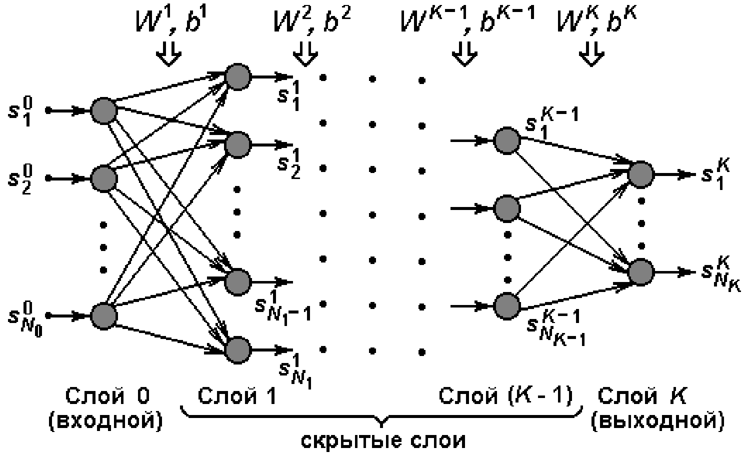


Рис. 1. Архитектура многослойной нейронной сети.

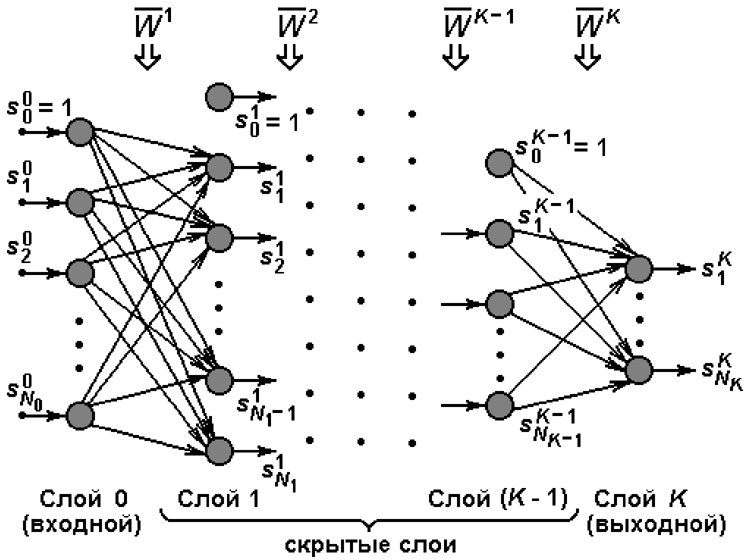


Рис. 2. Эквивалентное преобразование многослойной нейронной сети.

Обозначим $\tilde{w}_{ij}^q$, $i = \overline{1, N_q}$, $j = \overline{0, N_{q-1}}$, элемент матрицы $\tilde{W}^q$. В новых обозначениях математическая модель (1.1) преобразуется к следующему виду:

$$\begin{cases} s_0^q = 1, \quad s_i^q = f_q(h_i^q), \\ h_i^q = \sum\limits_{j=0}^{N_{q-1}} w_{ij}^q \, s_j^{q-1}, \end{cases} \tag{1.4}$$

$$q = \overline{1, K}, \; i = \overline{1, N_q}.$$

Формулам (1.4) соответствует схема функционирования МНС, представленная на рис. 2.

Применяя векторно-матричные операции над данными в слоях МНС, перейдем от уравнений (1.4) к следующей математической модели:

$$\begin{cases} s_0^q = 1, \; s_i^q = f_q(h_i^q), \\ h^q = \tilde{s}^{\,q-1} (\tilde{W}^q)^T, \end{cases} \tag{1.5}$$

$$q = \overline{1, K}, \; i = \overline{1, N_q},$$

где $h^q = (h_1^q, h_2^q, \ldots, h_{N_q}^q)$ — вектор потенциалов нейронов $q$-го слоя, верхний индекс $^T$ – знак транспонирования вектора или матрицы.

### *1.2. Постановка задачи обучения МНС*

Многослойная нейронная сеть осуществляет преобразование входного вектора $x$ размерности $N_0 = M$ в выходной вектор $s^K$ размерности $N_K = N$: $s^K = \varphi(x)$. Это функциональное преобразование для выбранной архитектуры сети (числа слоев, распределения нейронов по слоям и активационных характеристик нейронов) зависит от значений синаптических коэффициентов и смещений всех нейронов. Эти параметры сети собраны в матрицы $\tilde{W}^q$, $q = \overline{1, K}$. Даже в достаточно простых практических приложениях число параметров сети может достигать нескольких десятков тысяч. Например, в задаче распознавания изображения, представленного матрицей значений яркости размером 10*10 ($M$ = 100), с помощью сети, содержащей два рабочих слоя (скрытый слой $N_1$ = 50 и выходной слой $N_2$ = $N$ =2), число параметров превышает 5000. Если при решении практической задачи разумно подобрана архитектура нейронной сети, то настройкой параметров можно добиться близости фактического функционального преобразования, реализуемого нейронной сетью, к желаемому преобразованию в решаемой задаче. Процесс настройки параметров нейронной сети называется ее обучением.

Для обучения МНС используются данные обучающей выборки. Выборка состоит из образцов (примеров), содержащих желаемую реакцию $\sigma^p = (\sigma_1^p, \sigma_2^p, ..., \sigma_N^p)$ сети на входной вектор признаков $x^p = (x_1^p, x_2^p, ..., x_M^p)$:

$$\{ x^p, \sigma^p \}, p = \overline{1, P}, \tag{1.6}$$

где $P$ — объем обучающей выборки.

Обозначим фактическую реакцию рассматриваемой МНС на воздействие $x^p$ через $s^{pK}$ (верхний индекс $K$ соответствует номеру выходного слоя). Тогда разность ($\sigma^p$- $s^{pK}$) характеризует ошибку преобразования входного сигнала сетью, а показатель

$$D = \frac{1}{2} \sum_{p=1}^{P} \sum_{m=1}^{N} (\sigma_m^p - s_m^{pk})^2 \tag{1.7}$$

может служить критерием качества настройки параметров нейронной сети. Задача обучения нейронной сети сводится в такой постановке к достижению

$$\min_{\tilde{W}^q, \; q=\overline{1,K}} D(\tilde{W}^1, \tilde{W}^2, \; \ldots, \tilde{W}^K). \tag{1.8}$$

Поскольку при такой формализации задачи обучения предполагаются известными желаемые реакции МНС на входные сигналы из заданной выборки, что равносильно присутствию “учителя” в процессе обучения, сам процесс обучения называется “обучением с учителем” (supervised learning).

### *1.3. Градиентный поиск оптимальных параметров МНС*

Определим закон, по которому эволюционируют значения параметров $\tilde{w}_{ij}^{q}$, $q = \overline{1,K}$, $i = \overline{1,N_q}$, $j = \overline{0,N_{q-1}}$, в процессе минимизации показателя $D$, следующим равенством:

$$\frac{d}{d\tau}\tilde{w}_{ij}^{q} = -\varepsilon \frac{\partial D}{\partial \tilde{w}_{ij}^{q}} \ , \tag{1.9}$$

где $\varepsilon$ — параметр закона обучения; время $\tau$ процедуры настройки параметров полагается непрерывным. Если объединить все настраиваемые параметры МНС в один вектор, то закон (1.9) определяет, что в процессе эволюции этот вектор изменяется в каждый момент времени в направлении антиградиента критерия $D$. Несложно показать, что закон обучения (1.9) обеспечивает невозрастание показателя $D$ в процессе эволюции:

$$\frac{dD}{d\tau} = \sum_{q=1}^{K}\sum_{i=1}^{N_q}\sum_{j=1}^{N_{q-1}} \frac{\partial D}{\partial \tilde{w}_{ij}^{q}} \frac{d\tilde{w}_{ij}^{q}}{d\tau} = -\alpha \sum_{q,i,j}\left(\frac{\partial D}{\partial \tilde{w}_{ij}^{q}}\right)^{2} \le 0 \ .$$

Таким образом, следуя закону (1.9) настройки параметров МНС, из любого начального значения параметров осуществляется спуск к экстремальной точке. В связи с тем, что показатель $D$, рассматриваемый как функция настраиваемых параметров, может содержать много локальных минимумов, нет гарантии спуска из произвольного начального значения вектора параметров в точку глобального минимума. В практических приложениях рекомендуется реализовать процедуру градиентного спуска неоднократно из различных начальных положений для выбора лучшего из полученных решений.

Поскольку нейроны функционируют в дискретном времени, реализация закона (1.9) требует перехода от непрерывного к дискретному времени $\tau$. В этом случае уравнения эволюции синаптических коэффициентов записываются в следующей форме:

$$\tilde{w}_{ij}^{q}(\tau+1) = \tilde{w}_{ij}^{q}(\tau) - \alpha \frac{\partial D(\tau)}{\partial \tilde{w}_{ij}^{q}} \ , \tag{1.10}$$

где $\alpha$ — параметр закона обучения. От выбора параметра $\alpha$ зависит скорость обучения, форма переходного процесса по настраиваемым параметрам, а также сам факт сходимости процесса обучения. Чем меньше значение параметра $\alpha$, тем менее различаются уравнения (1.9) и (1.10), реализующие поиск в непрерывном и дискретном времени. Следовательно, тем выше вероятность сходимости дискретной поисковой процедуры (гарантированное невозрастание критерия $D$ в процессе эволюции доказано выше только для непрерывного времени). В то же время чрезмерное уменьшение параметра $\alpha$ затягивает переходные процессы, увеличивает необходимое время вычислений. Обычно в программных средствах предусматриваются различные возможности адаптивной подстройки значения $\alpha$ в процессе поиска экстремума, а выбор начального значения этого параметра определяется пользователем.

Для реализации закона обучения (1.10) необходимо определить алгоритм вычисления частных производных критерия $D$ по искомым параметрам: $\dfrac{\partial D}{\partial \tilde{w}_{ij}^{q}}$, $q = \overline{1,K}$, $i = \overline{1,N_q}$, $j = \overline{0,N_{q-1}}$. Этому вопросу посвящены следующие два подраздела.

### *1.4. Алгоритм обучения однослойной нейронной сети с непрерывной передаточной функцией*

Рассмотрим частный случай $K = 1$, когда сеть содержит один рабочий слой. Опустим верхний индекс номера слоя в обозначениях настраиваемых синаптических коэффициентов $\tilde{w}_{ij}^{q}$, $i = \overline{1,N}$ $(N = N_1)$, $j = \overline{1,M}$ $(M = N_0)$, а также векторов потенциалов $h_i$, $i = \overline{1,N}$, и состояний нейронов $s_i$, $i = \overline{1,N}$, выходного слоя. Опустим также индекс номера слоя в обозначении передаточной функции нейронов $f(h)$.

В принятых обозначениях критерий точности $D$ на основании выражений (1.7) и (1.4) может быть представлен в следующем виде:

$$D = \frac{1}{2}\sum_{p=1}^{P}\sum_{m=1}^{N}(\sigma_m^p - s_m^p)^2 = \frac{1}{2}\sum_{p=1}^{P}\sum_{m=1}^{N}(\sigma_m^p - f(h_m^p))^2, \qquad (1.11)$$

где $h_m^p = \sum_{j=0}^{M} \tilde{w}_{mj}\, x_j^p$, $(x_j = s_j^0,\ j = \overline{1, M})$ ; $x_j^p$ – $j$-я составляющая входного вектора $x$ в примере $p$ обучающей выборки.

Вычислим с использованием выражения (1.11) производную показателя $D$ по аргументу $\tilde{w}_{ij}$, $i = \overline{1, N}$, $j = \overline{1, M}$ :

$$\frac{\partial D}{\partial \tilde{w}_{ij}} = -\sum_{p=1}^{P}(\sigma_i^p - f(h_i^p))\frac{\partial}{\partial \tilde{w}_{ij}} f(h_i^p) =$$

$$= -\sum_{p=1}^{P}(\sigma_i^p - f(h_i^p)) f'(h_i^p) x_j^p . \qquad (1.12)$$

В выводе выражения (1.12) было учтено, что в сумме слагаемых по индексу $m$ в выражении (1.11) только одно слагаемое, соответствующее значению $m = i$, зависит от $\tilde{w}_{ij}$ . В связи с этим окончательное выражение (1.12) содержит лишь сумму по примерам обучающей выборки. Введем обозначение:

$$\Delta_i^p = (\sigma_i^p - f(h_i^p)) f'(h_i^p) = (\sigma_i^p - s_i^p) f'(h_i^p), \qquad (1.13)$$

$$i = \overline{1, N} .$$

Тогда выражение (1.12) преобразуется к следующей краткой форме:

$$\frac{\partial D}{\partial \tilde{w}_{ij}} = -\sum_{p=1}^{P}\Delta_i^p x_j^p, \ i = \overline{1, N}, \ j = \overline{0, M} . \qquad (1.14)$$

Заметим, что в силу предположения о непрерывности и дифференцируемости активационной характеристики нейронов производная $f\,'(h)$ существует (см. выражения (1.12) и (1.13)). В частности, для $f(h) = \mathrm{th}(\beta\, h)$

$$f\,'(h) = \beta\,[1 - (f(h))^2],$$

а для логистической функции $f(h) = 1 / (1 + \exp(-\beta h))$

$$f\,'(h) = \beta\, f(h)\,(1 - f(h)).$$

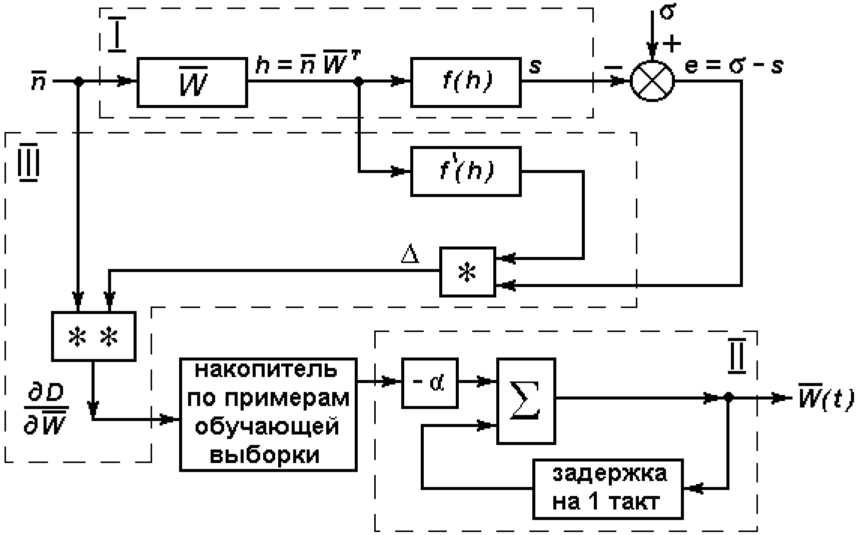


Рис. 3. Схема настройки параметров однослойной нейронной сети

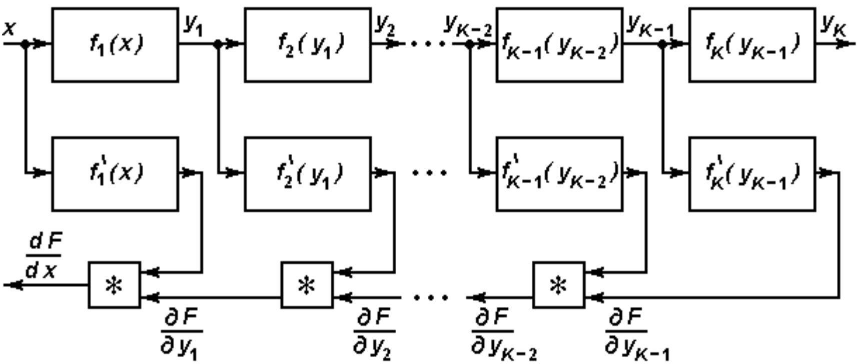


Рис.4. Схема вычисления производной при последовательном соединении нелинейных преобразователей

На рис. 3 представлена схема вычислений в соответствии с формулами (1.10), (1.14), определяющими алгоритм настройки параметров нейронной сети для рассматриваемого случая $K = 1$. В целях упрощения схемы применяются векторно-матричные обозначения для переменных. Блок I представляет собой сеть прямого распространения входного сигнала. Блок II отражает правило эволюции матрицы $\tilde{W}$ в процессе градиентного спуска. Блок III служит для вычисления матрицы частных производных $\frac{\partial D}{\partial \tilde{w}_{ij}}$, $i = \overline{1,N}$, $j = \overline{0,M}$. В составе блока III звено, отмеченное знаком “*”, функционирует согласно выражению (1.13), а звено, отмеченное знаком “**”, вычисляет все парные произведения $\Delta_i\, x_j$ элементов входных векторов $\tilde{x}$ и $\Delta$ ($i = \overline{1,N}$, $j = \overline{0,M}$).

Перед началом процесса обучения устанавливаются некоторые произвольные значения элементов матрицы $\tilde{W}(0)$. Эта операция называется инициализацией нейронной сети. Обычно для этих целей используется датчик случайных чисел, распределенных равномерно на отрезке [-$c$; $c$], где $c$ — параметр, устанавливаемый пользователем. Параметр $c$ должен быть достаточно малым. В противном случае нейрон может оказаться в зоне насыщения сигмоидальной характеристики, когда он нечувствителен к любым малым изменениям синаптических коэффициентов. Эволюция соответствующих коэффициентов прекращается, хотя их значения далеки от оптимальных. Этот эффект называется “параличом” сети.

После установки значений $\tilde{W}(0)$ блоки I и III функционируют $P$ раз в соответствии с объемом обучающей выборки, что позволяет вычислить и накопить сумму (1.14), определяющую значение $\frac{\partial D}{\partial \tilde{w}_{ij}}$ во всех обучающих примерах. Этот цикл обычно называют эпохой. Далее в блоке II реализуется подстройка элементов матрицы $\tilde{W}$, в результате чего формируется $\tilde{W}(1)$:

$$\tilde{w}_{ij}(1) = \tilde{w}_{ij}(0) - \alpha \frac{\partial D(0)}{\partial \tilde{w}_{ij}}.$$

Это значение устанавливается в блоке I, после чего реализуется следующая эпоха и производится следующий такт подстройки коэффициентов в блоке III. Процедура повторяется до тех пор, пока не будет достигнут заданный уровень ошибки $D$, которая вычисляется в соответствии с выражением (1.11) в процессе обучения сети. Если требуемый уровень по ошибке $D$ не достигается, то это может быть обусловлено несколькими причинами. Возможно, в рамках выбранной архитектуры сети принципиально невозможно достигнуть заданной точности. Другая причина – “паралич” сети. Возможно также, что текущее значение матрицы $\tilde{W}$ соответствует положению точки в области обширного “плоскогорья” на поверхности функции $D(\tilde{W})$ (эта поверхность обычно называется адаптивным рельефом), когда продвижение точки сильно замедлено. В этом случае можно увеличить значение параметра $\alpha$, чтобы ускорить продвижение к “обрыву” на адаптивном рельефе. В программных пакетах, реализующих общую идею обучения, обычно применяют различные модификации градиентного спуска, ускоряющие процесс поиска минимума показателя $D$.

### *1.5. Метод обратного распространения ошибки для обучения МНС*

Метод обратного распространения ошибки (error backpropagation), который является одним из самых распространенным в практических приложениях нейронных сетей, был сформулирован независимо друг от друга несколькими русскими и зарубежными учеными в 80-е годы.

Для иллюстрации излагаемого далее принципа рассмотрим следующий пример. Пусть нелинейный преобразователь $F(x)$ представляет собой последовательное соединение нелинейных элементов $f_1$, $f_2$, …, $f_K$ (см. рис. 4). В соответствии со схемой преобразователя можно записать $F(x)= f_K(f_{K-1}(\ldots f_1(x)))$.

Поставим задачу вычисления производной $\frac{dF}{dx}$. Согласно правилу дифференцирования сложной функции

$$\frac{dF(x)}{dx} = f'_K(y_{K-1})\Big|_{y_{K-1}(x)} \; f'_{K-1}(y_{K-2})\Big|_{y_{K-2}(x)} \cdots \; f'_1(x)$$

На рисунке 4 показано, что этот результат формируется на выходе дополнительной цепочки, в которой последовательно соединены блоки перемножения. В этой цепочке реализуется обратное движение сигнала с использованием результата прямого распространения входного воздействия x.

Этот пример наводит на мысль о возможности вычисления частных производных, необходимых для обучения МНС по закону (1.10), с использованием известных правил дифференцирования сложной функции и реализации обратного распространения сигнала ошибки (роль $F$ играет показатель $D$ точности обучения сети).

В качестве первого шага рассмотрим выражение для частных производных $\frac{\partial D}{\partial \tilde{w}_{ij}^K}$ функционала $D$ по настраиваемым параметрам последнего слоя:

$$\frac{\partial D}{\partial \tilde{w}_{ij}^K} = \frac{\partial}{\partial \tilde{w}_{ij}^K}\left(\frac{1}{2}\sum_{p=1}^{P}\sum_{m=1}^{N=N_k}(\sigma_m^p - s_m^{pK})^2\right) =$$

$$= -\sum_{p=1}^{P}\sum_{m=1}^{N=N_k}(\sigma_m^p - s_m^{pK})\frac{\partial s_m^{pK}}{\partial \tilde{w}_{ij}^K} \qquad (1.15)$$

$$i = \overline{1, N_K}\,,\; j = \overline{0, N_{K-1}}\,.$$

Вычисление частной производной $\frac{\partial s_m^{pK}}{\partial \tilde{w}_{ij}^K}$ опирается на уравнение функционирования $K$-го слоя МНС:

$$s_m^{pK} = f_K(h_m^{pK}) = f_K\left(\sum_{r=0}^{N_{K-1}} \tilde{w}_{mr}^K\, s_r^{p(K-1)}\right),\; m = \overline{1, N_K}\,. \qquad (1.16)$$

Из уравнения (1.16) следует, что

$$\frac{\partial s_m^{pK}}{\partial \tilde{w}_{ij}^K} = \begin{cases} 0, & m \neq i \\ f'(h_i^{pK}) s_j^{p(K-1)}, & m = i \end{cases}, \tag{1.17}$$

$$i = \overline{1, N_K}\,,\ j = \overline{0, N_{K-1}}\,.$$

После подстановки выражения (1.17) в (1.15) получим:

$$\frac{\partial D}{\partial \tilde{w}_{ij}^K} = -\sum_{p=1}^{P} (\sigma_i^p - s_i^{pK})\, f_K'(h_i^{pK}) s_j^{p(K-1)}\,.$$

С использованием обозначения

$$\Delta_i^{pK} = (\sigma_i^{pK} - s_i^{pK})\, f_K'(h_i^{pK}), \quad i = \overline{1, N_K}\,, \tag{1.18}$$

последнее выражение для частной производной преобразуется к следующему виду:

$$\frac{\partial D}{\partial \tilde{w}_{ij}^K} = -\sum_{p=1}^{P} \Delta_i^{pK} s_j^{p(K-1)}\,, \quad i = \overline{1, N_K}\,, j = \overline{0, N_{K-1}}\,. \tag{1.19}$$

Перейдем к рассмотрению ($K$-1)-го слоя с настраиваемыми коэффициентами $\tilde{w}_{ij}^{K-1}$, $i = \overline{1, N_{K-1}}$, $j = \overline{0, N_{K-2}}$:

$$\frac{\partial D}{\partial \tilde{w}_{ij}^{K-1}} = -\sum_{p=1}^{P} \sum_{m=1}^{N=N_K} (\sigma_m^p - s_m^{pK}) f_K'(h_m^{pK}) \frac{\partial}{\partial s_i^{p(K-1)}} h_m^{pK} \frac{\partial s_i^{p(K-1)}}{\partial \tilde{w}_{ij}^{K-1}} =$$

$$= -\sum_{p=1}^{P} \sum_{m=1}^{N=N_K} (\sigma_m^p - s_m^{pK})\ f_K'(h_m^{pK})\ \tilde{w}_{mi}^K\ f_{K-1}'(h_i^{p(K-1)})\ s_j^{p(K-2)} =$$

$$= -\sum_{p=1}^{P} \Delta_i^{p(K-1)} s_j^{p(K-2)}\,, \tag{1.20}$$

где использовано обозначение:

$$\Delta_i^{p(K-1)} = \sum_{m=1}^{N=N_K} (\sigma_m^p - s_m^{pK})\, f_K'(h_m^{pK}) \tilde{w}_{mi}^K\ f_{K-1}'(h_i^{p(K-1)}) =$$

$$= \left( \sum_{m=1}^{N=N_K} \Delta_m^{pK}\ \tilde{w}_{mi}^K \right) f_{K-1}'(h_i^{p(K-1)}),\ i = \overline{1, N_{K-1}}\,. \tag{1.21}$$

В последнем выражении была применена формула (1.18).

Для последующих (с конца) слоев ($K$-2), ($K$-3), … вычисления частных производных функционала $D$ по элементам расширенной матрицы синаптических коэффициентов в слое выполняются аналогичным образом. В итоге таких вычислений получается следующая общая формула:

$$\frac{\partial D}{\partial \tilde{w}_{ij}^{q}} = -\sum_{p=1}^{P} \Delta_i^{pq} s_j^{p(q-1)}, \; q = \overline{1,K}, \; i = \overline{1,N_q}, \; j = \overline{0,N_{q-1}}, \quad (1.22)$$

где

$$\Delta_i^{pq} = \left( \sum_{m=1}^{N=N_{q+1}} \Delta_m^{p(q+1)} \, \tilde{w}_{mi}^{q+1} \right) f_q'(h_i^{pq}), \quad q = \overline{1,K-1}, \quad (1.23)$$

$$\Delta_i^{pK} = (\sigma_i^p - s_i^{pK}) f_K'(h_i^{pK}), \quad i = \overline{1,N_K}.$$

Переменные $\Delta_i^{pq}$, $q = \overline{1,K}$, $i = \overline{1,N_q}$, получили название двойственных по отношению к потенциалам нейронов $h_i^{pq}$, $q = \overline{1,K}$, $i = \overline{1,N_q}$, в сети прямого распространения входного сигнала.

На рисунке 5 представлена схема вычислений в соответствии с формулами (1.22), (1.23). Схема содержит цепь обратного распространения , которая возбуждается сигналом ошибки $e = \sigma - s^K$. В целях упрощения обозначений в схеме опущен индекс $p$ примера обучающей выборки. Цепь обратного распространения формирует двойственные переменные $\Delta^q$, $q = \overline{1,K}$, являющиеся векторами размерности $N_q$. Знаками “*” и ”**” отмечены звенья, которые осуществляют то же преобразование данных, что и в схеме на рис. 3.

Для управления процессом настройки параметров МНС согласно системе уравнений (1.10) следует активизировать вычисления (см. рис. 5) $P$ раз по числу обучающих примеров и провести накопление результатов подобно тому, как это показано на рис. 3 для простейшего случая однослойной сети.

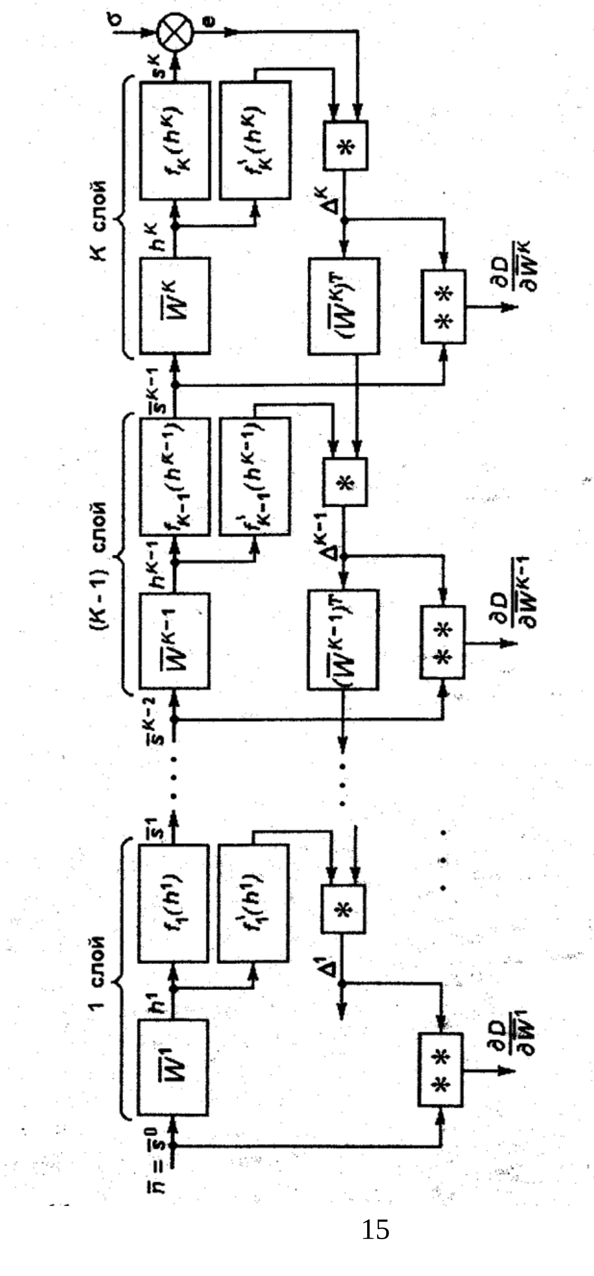


Рис. 5 Схема сетей прямого распространения и обратного распространения ошибки

Сопоставление схем прямого и обратного распространения в их полном (а не сжатом векторно-матричном) представлении показывает, что схема обратного распространения может быть построена по заданной прямой схеме путем применения к последней следующих правил:

1. Направление стрелок, указывающих прохождение сигнала, меняется на обратное.
2. Фрагменты схемы с активационными характеристиками нейронов $f_q(h_q^i),\ q=\overline{1,K},\ i=\overline{1,N_q}$ заменяются нелинейным преобразователем $f_q'(h_q^i)$ и блоком перемножения.
3. Матрицы синаптических коэффициентов $\tilde{W}^q,\ q=\overline{1,K}$, транспонируются.
4. Сумматоры заменяются точками разветвления, а точки разветвления – сумматорами.

В [8] показано, что сформулированные правила построения схемы формирования двойственных переменных справедливы и в том случае, когда нейронная сеть содержит прямые связи не только рядом расположенных слоев, но и более удаленных (связь “перепрыгивает” несколько ближайших слоев).

Практические исследования показывают, что обученная МНС обладает высокой робастностью: при установке значений синаптических коэффициентов, отличающихся от оптимальных (ошибки реализации), сеть продолжает выполнять свою функциональную задачу. Даже разрыв некоторых синаптических связей (технический отказ отдельных элементов вычислительной сети) может не приводить к потере работоспособности сети (отказоустойчивость). Следует заметить, что указанные свойства проявляются только в том случае, когда нейронная сеть обладает некоторой информационной “избыточностью” по отношению к решаемой задаче.

Принципиальной является способность МНС к обобщению, то есть способность формировать “разумную” реакцию на входные воздействия, которых не было в составе обучающей выборки.

Именно благодаря этому свойству МНС успешно применяется для интерполяции функций многих переменных, экстраполяции временных рядов, классификации объектов по их признакам и в других практических приложениях.

## Контрольные вопросы.

1. Нарисуйте схему и объясните особенности архитектуры многослойной нейронной сети.
2. Напишите уравнения функционирования многослойной нейронной сети.
3. Объясните состав данных таблицы обучающей выборки, используемой для настройки параметров нейронной сети.
4. Какие параметры многослойной нейронной сети настраиваются в процессе ее обучения?
5. Какой критерий используется для организации обучения многослойной нейронной сети?
6. Какой метод применяется для обучения многослойной нейронной сети в используемом в работе нейроэмуляторе?
7. Чем характеризуется эффект «паралича» при обучении многослойной нейронной сети?
8. В чем состоит и как реализуется процесс инициализации при обучении нейронной сети?
9. В чем состоит и как проверяется эффект «генерализации данных» в нейронной сети?
10. Какие параметры режима обучения многослойной нейронной сети доступны пользователю для настройки?
11. В чем состоит процедура тестирования обученной многослойной нейронной сети?
12. Какие переменные вычисляются с помощью метода обратного распространения ошибки и как они используются в процессе обучения многослойной нейронной сети?
13. Приведите примеры активационных характеристик нейронов, используемых в многослойных нейронных сетях.
14. Что называется «эпохой» в процессе обучения нейронной сети?

15. Почему разные реализации процесса обучения многослойной нейронной сети из разных начальных условий не приводят к одному и тому же финальному результату?
16. В чем состоит формальное правило построения структурной схемы сети обратного распространения ошибки по заданной схеме прямого распространения сигнала?
17. По какому правилу производится модификация значений параметров многослойной нейронной сети при ее обучении методом обратного распространения ошибки?