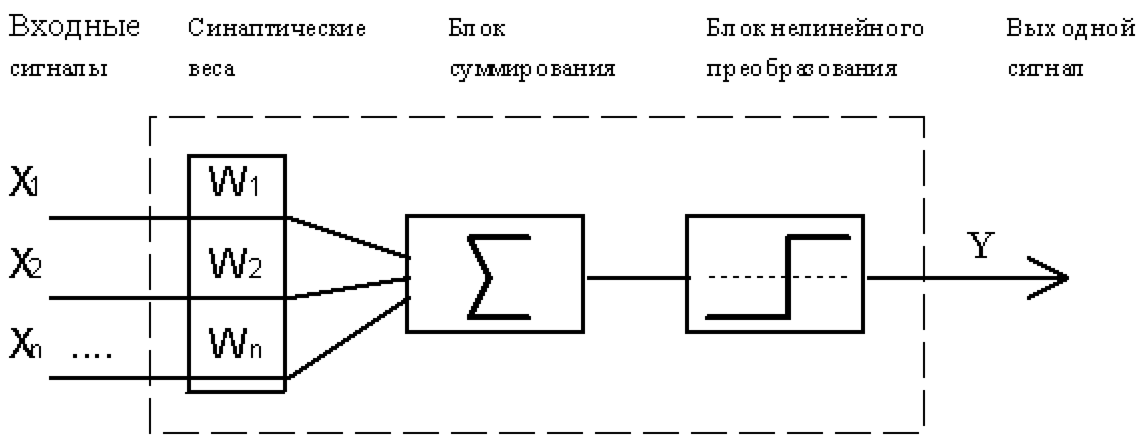
Метод коррекции ошибки. Метод обратного распространения ошибки

Выполнила: Шамустдинова Лилия

Формальный нейрон

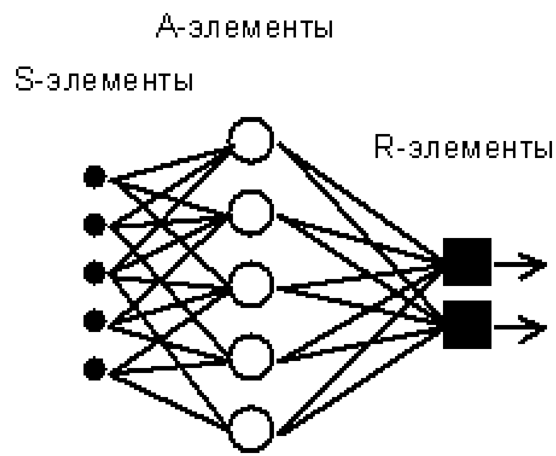


- Вектор локальной памяти содержит информацию о весовых множителях, с которыми входные сигналы будут интерпретироваться нейроном
- В блоке суммирования происходит накопление общего входного сигнала равного взвешенной сумме входов

$$net = \sum_{i=1}^{n} W_i x_i$$

- Отклик нейрон далее описывается по принципу "все или ничего", т. е. переменная подвергается нелинейному пороговому преобразованию, при котором выход (состояние активации нейрона) Y устанавливается равным единице, если net > Q, и Y=0 в обратном случае. Значение порога Q (часто полагаемое =нулю) также хранится в локальной памяти.

Персептрон Розенблатта



- S-элементы формируют сетчатку сенсорных клеток, принимающих двоичные сигналы от внешнего мира.
- Далее сигналы поступают в слой ассоциативных или А-элементов (для упрощения изображения часть связей от входных S-клеток к А-клеткам не показана). Только ассоциативные элементы, представляющие собой формальные нейроны, выполняют нелинейную обработку информации и имеют изменяемые веса связей.
- R-элементы с фиксированными весами формируют сигнал реакции персептрона на входной стимул.

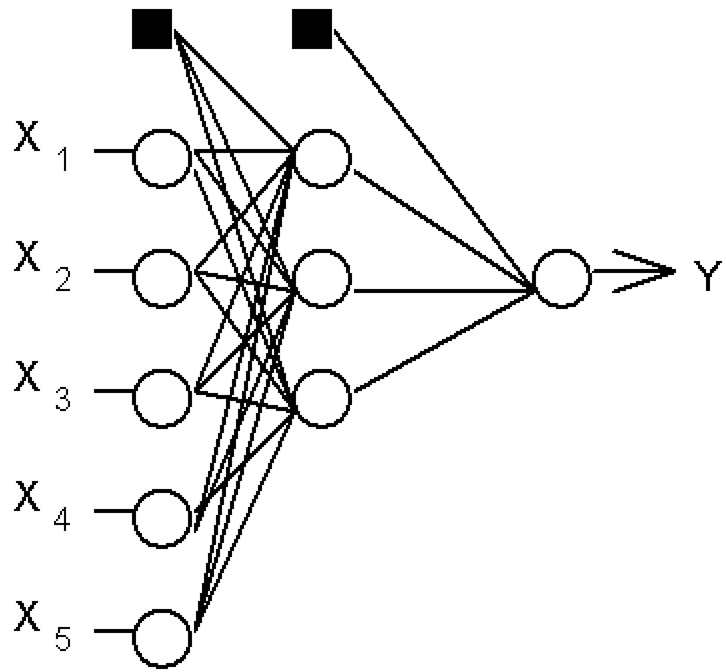
Теорема об обучении персептрона

- Обучение сети состоит в подстройке весовых коэффициентов каждого нейрона
- Пусть имеется набор пар векторов (x^a, y^a), $a = 1..p$, называемый *обучающей выборкой*
- Будем называть нейронную сеть обученной на данной обучающей выборке, если при подаче на входы сети каждого вектора x^a на выходах всякий раз получается соответствующий вектор y^a

Алгоритм коррекции по ошибке

- Шаг 0: Начальные значения весов всех нейронов $W(t=0)$ полагаются случайными
- Шаг 1: Сети предъявляется входной образ x^a, в результате формируется выходной образ $\mathbf{y}^{\alpha} \neq y^{\alpha}$
- Шаг 2: Вычисляется вектор ошибки $\delta^{\alpha} = \left(y^{\alpha} - \mathbf{y}^{\alpha}\right)$, делаемой сетью на выходе. Дальнейшая идея состоит в том, что изменение вектора весовых коэффициентов в области малых ошибок должно быть пропорционально ошибке на выходе, и равно нулю если ошибка равна нулю
- Шаг 3: Вектор весов модифицируется по следующей формуле: $W(t+\Delta t) = W(t) + \eta x^{\alpha} \cdot \left(\delta^{\alpha}\right)^{T}$ Здесь $0 < \eta < 1$ - темп обучения
- Шаг 4: Шаги 1-3 повторяются для всех обучающих векторов. Один цикл последовательного предъявления всей выборки называется *эпохой*. Обучение завершается по истечении нескольких эпох, а) когда итерации сойдутся, т.е. вектор весов перестает изменяться, или б) когда полная просуммированная по всем векторам абсолютная ошибка станет меньше некоторого малого значения.

Многослойный персептрон



- Межнейронные синаптические связи сети устроены таким образом, что каждый нейрон на данном уровне иерархии принимает и обрабатывает сигналы от каждого нейрона более низкого уровня. Таким образом, в данной сети имеется выделенное направление распространения нейроимпульсов - от входного слоя через один (или несколько) скрытых слоев к выходному слою нейронов

Многослойный персептрон

- Персептрон представляет собой сеть, состоящую из нескольких последовательно соединенных слоев формальных нейронов МакКаллока и Питтса
- На низшем уровне иерархии находится *входной* слой, состоящий из сенсорных элементов, задачей которого является только прием и распространение по сети входной информации
- Далее имеются один или, реже, несколько *скрытых* слоев. Каждый нейрон на скрытом слое имеет несколько входов, соединенных с выходами нейронов предыдущего слоя или непосредственно со входными сенсорами X1..Xn, и один выход
- Нейрон характеризуется уникальным вектором весовых коэффициентов w. Веса всех нейронов слоя формируют матрицу, которую мы будем обозначать V или W
- Функция нейрона состоит в вычислении взвешенной суммы его входов с дальнейшим нелинейным преобразованием ее в выходной сигнал: $y = 1 \Big/ \left(1 + \exp(-[\sum_i W_i x_i - \Theta])\right)$

Метод обратного распространения ошибок

- Для упрощения обозначений ограничимся ситуацией, когда сеть имеет только один скрытый слой
- Матрицу весовых коэффициентов от входов к скрытому слою обозначим W, а матрицу весов, соединяющих скрытый и выходной слой - как V
- Для индексов примем следующие обозначения: входы будем нумеровать только индексом i, элементы скрытого слоя – индексом j, а выходы, соответственно, индексом k
- Пусть сеть обучается на выборке (Xa,Ya), a =1..p. Активности нейронов будем обозначать малыми буквами y с соответствующим индексом, а суммарные взвешенные входы нейронов - малыми буквами x

Метод обратного распространения ошибок

- Шаг 0: Начальные значения весов всех нейронов всех слоев V(t=0) и W(t=0) полагаются случайными числами.
- Шаг 1: Сети предъявляется входной образ Xa, в результате формируется выходной образ y!=Ya. При этом нейроны последовательно от слоя к слою функционируют по следующим формулам:

скрытый слой $$x_j = \sum_i W_{ij} X_i^{\alpha}; y_j = f(x_j)$$

выходной слой $$x_k = \sum_j V_{jk} y_j; y_k = f(x_k)$$

Здесь f(x) - сигмоидальная функция

Метод обратного распространения ошибок (2)

- Шаг 2: Функционал квадратичной ошибки сети для данного входного образа имеет вид: $E = 1/2\sum_k \left(y_k - Y_k^{\alpha}\right)^2$

  Данный функционал подлежит минимизации. Классический градиентный метод оптимизации состоит в итерационном уточнении аргумента согласно формуле: $V_{jk}(t+1) = V_{jk}(t) - h \cdot \frac{\partial E}{\partial V_{jk}}$

  Функция ошибки в явном виде не содержит зависимости от веса V_{jk}, поэтому воспользуемся формулами неявного дифференцирования сложной функции:

$$\frac{\partial E}{\partial y_k} = \delta_k = \left(y_k - Y_k^{\alpha}\right);$$

$$\frac{\partial E}{\partial x_k} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_k} = \delta_k \cdot y_k(1 - y_k)$$

$$\frac{\partial E}{\partial V_{jk}} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_k} \cdot \frac{\partial x_k}{\partial V_{jk}} =$$

$$= \delta_k \cdot y_k(1 - y_k) \cdot y_j$$

  Здесь учтено полезное свойство сигмоидальной функции f(x): ее производная выражается только через само значение функции, f'(x)=f(1-f)

Метод обратного распространения ошибок (3)

- Шаг 3: На этом шаге выполняется подстройка весов скрытого слоя. Градиентный метод по-прежнему дает:

$$W_{ij}(t+1) = W_{ij}(t) - h \cdot \frac{\partial E}{\partial W_{ij}}$$

Вычисления производных выполняются по тем же формулам, за исключением некоторого усложнения формулы для ошибки d $_j$.

$$\frac{\partial E}{\partial x_k} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_k} = \delta_k \cdot y_k(1 - y_k)$$

$$\frac{\partial E}{\partial y_j} = \delta_j = \sum_k \frac{\partial E}{\partial x_k} \cdot \frac{\partial x_k}{\partial y_j} =$$

$$= \sum_k \delta_k \cdot y_k(1 - y_k) \cdot V_{jk};$$

$$\frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial y_j} \cdot \frac{\partial y_j}{\partial x_j} \cdot \frac{\partial x_j}{\partial W_{ij}} =$$

$$= \delta_j \cdot y_j(1 - y_j) \cdot X_i^\alpha =$$

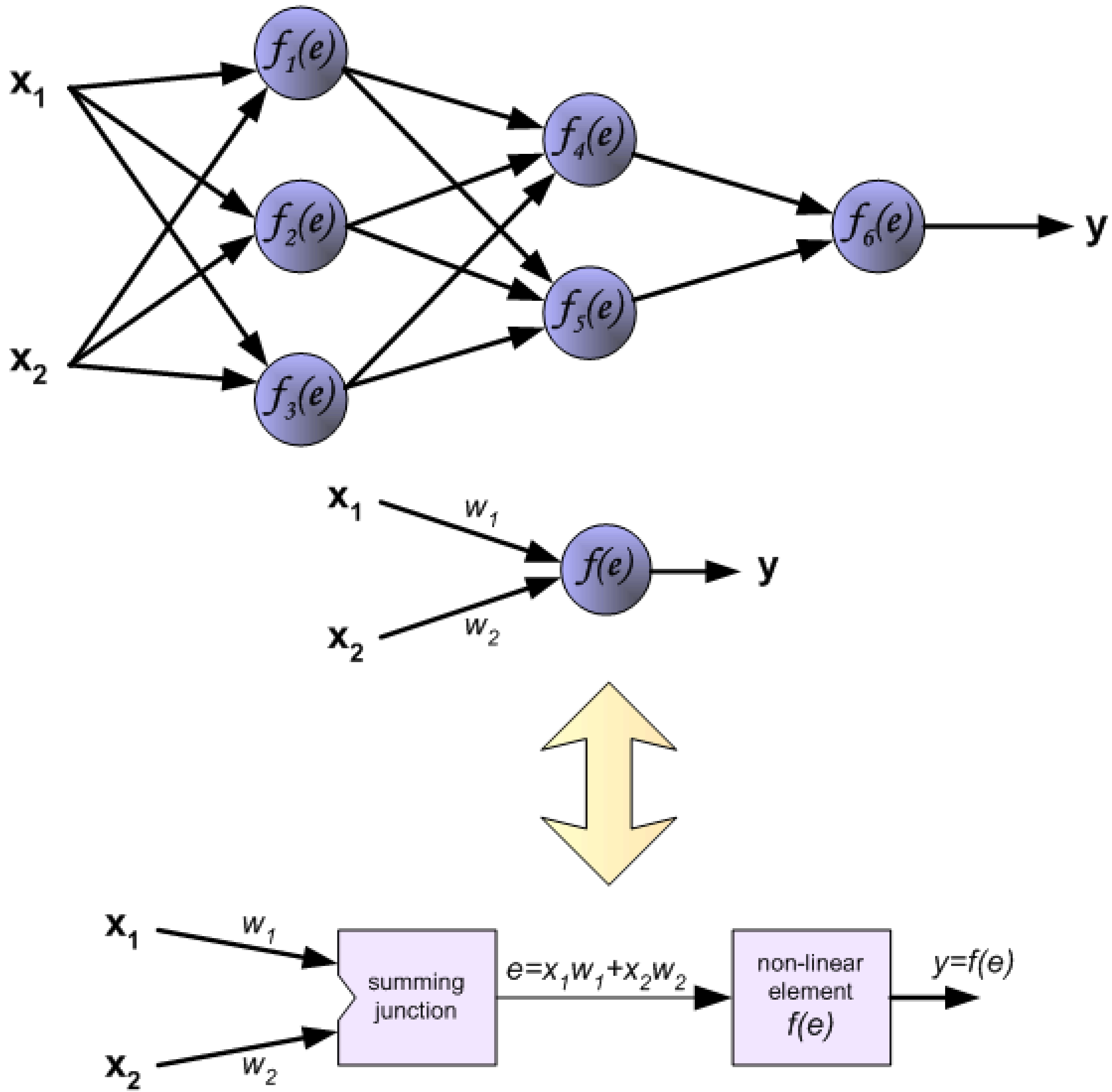
$$= [\sum_k \delta_k \cdot y_k(1 - y_k) \cdot V_{jk}] \cdot [y_j(1 - y_j) \cdot X_i^\alpha]$$

При вычислении d $_j$ здесь и был применен принцип обратного распространения ошибки: частные производные берутся только по переменным *последующего* слоя. По полученным формулам модифицируются веса нейронов скрытого слоя. Если в нейронной сети имеется несколько скрытых слоев, процедура обратного распространения применяется последовательно для каждого из них, начиная со слоя, предшествующего выходному, и далее до слоя, следующего за входным. При этом формулы сохраняют свой вид с заменой элементов выходного слоя на элементы соответствующего скрытого слоя.

Метод обратного распространения ошибок (4)

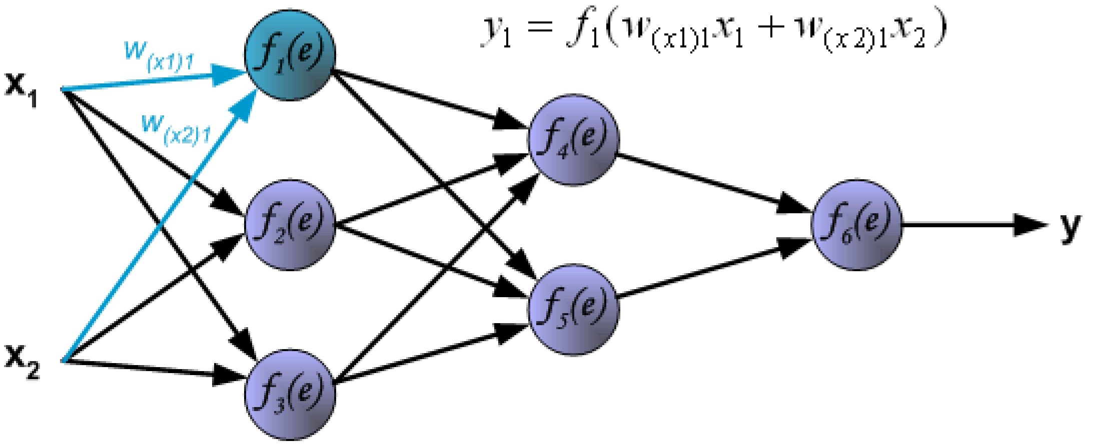
- Шаги 1-3 повторяются для всех обучающих векторов. Обучение завершается по достижении малой полной ошибки или максимально допустимого числа итераций, как и в методе обучения Розенблатта

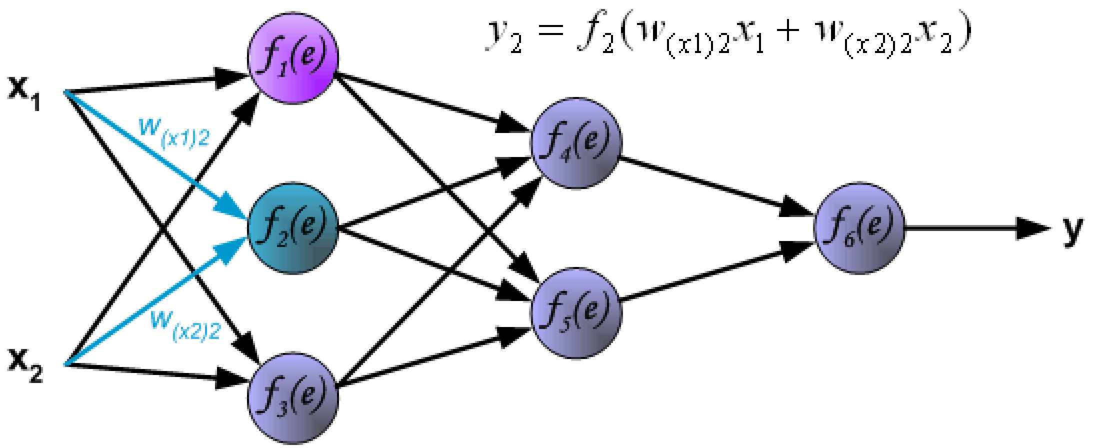
Пример



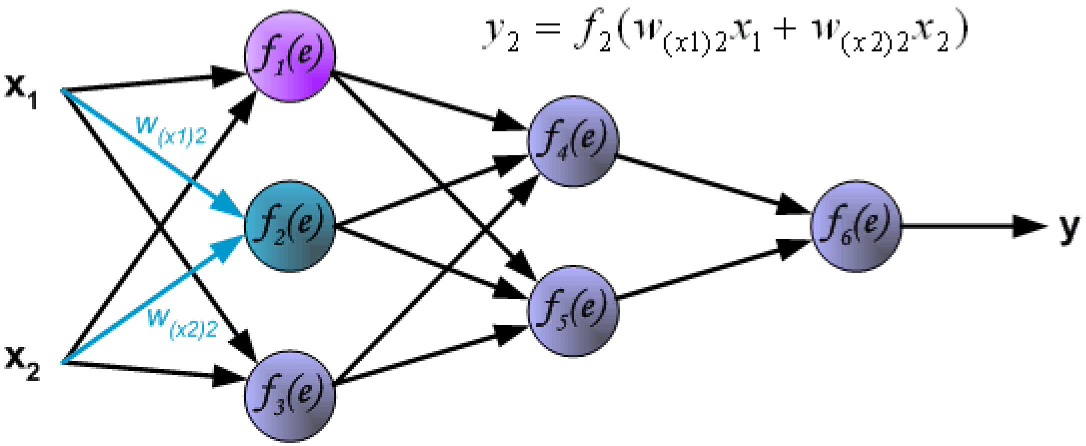
x1
x2
f1(e)
f2(e)
f3(e)
f4(e)
f5(e)
f6(e)
y
x1
w1
f(e)
y
x2
w2
x1
w1
summing
junction
x2
w2
e=x1w1+x2w2
non-linear
element
f(e)
y=f(e)

Пример


$y_1 = f_1(w_{(x1)1}x_1 + w_{(x2)1}x_2)$
$w_{(x1)1}$
$w_{(x2)1}$
x_1
x_2
$f_1(e)$
$f_2(e)$
$f_3(e)$
$f_4(e)$
$f_5(e)$
$f_6(e)$
y

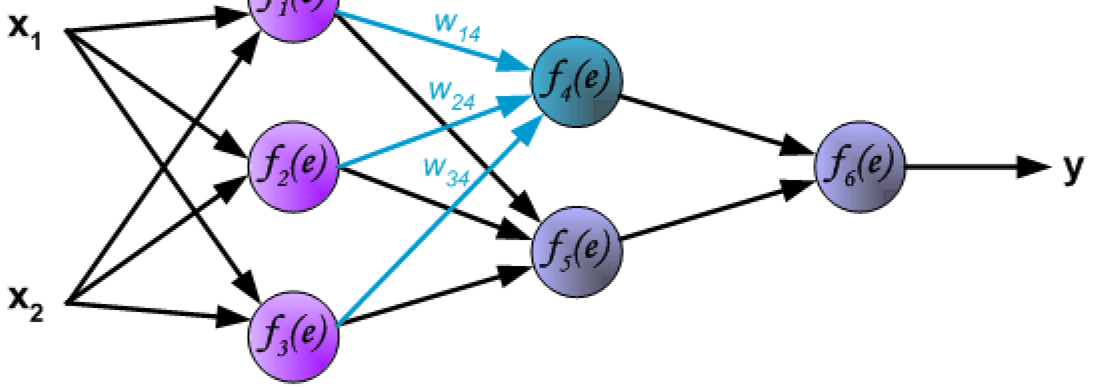

$y_2 = f_2(w_{(x1)2}x_1 + w_{(x2)2}x_2)$
$w_{(x1)2}$
$w_{(x2)2}$
x_1
x_2
$f_1(e)$
$f_2(e)$
$f_3(e)$
$f_4(e)$
$f_5(e)$
$f_6(e)$
y

Пример

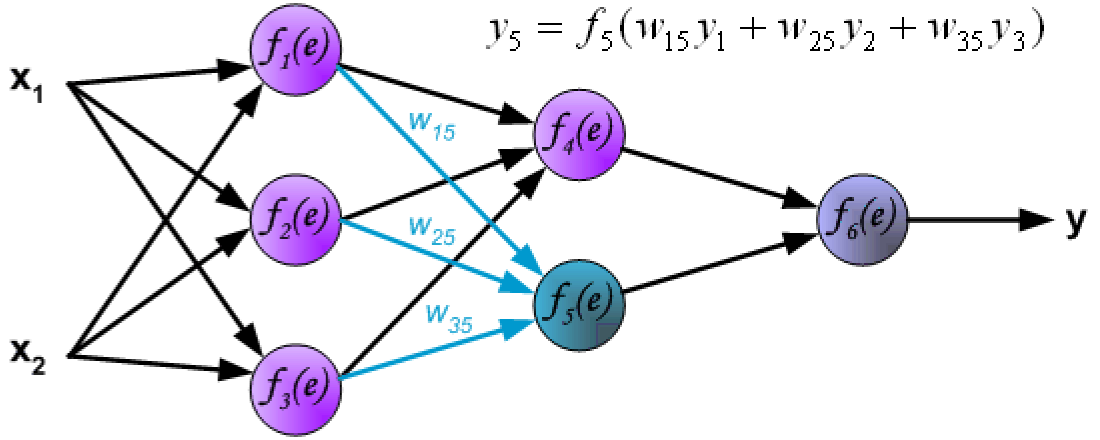


Пример

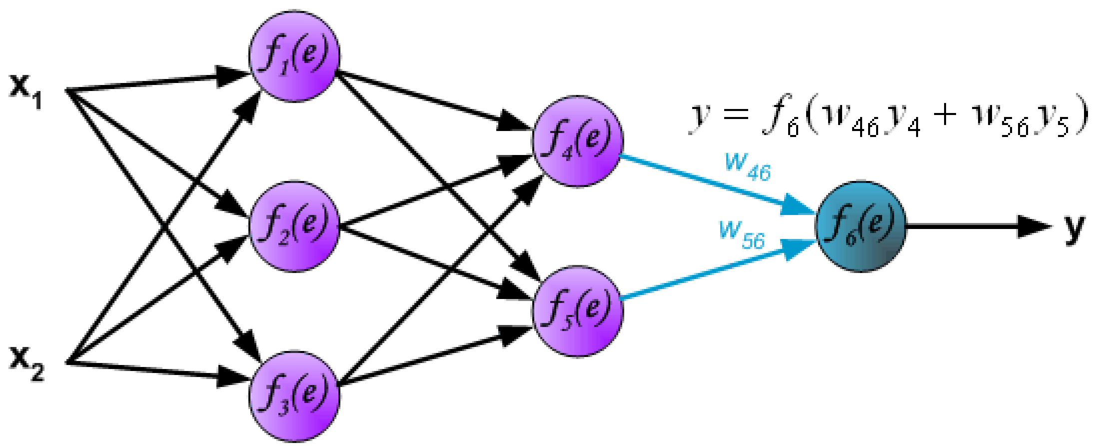
$$y_4 = f_4(w_{14}y_1 + w_{24}y_2 + w_{34}y_3)$$



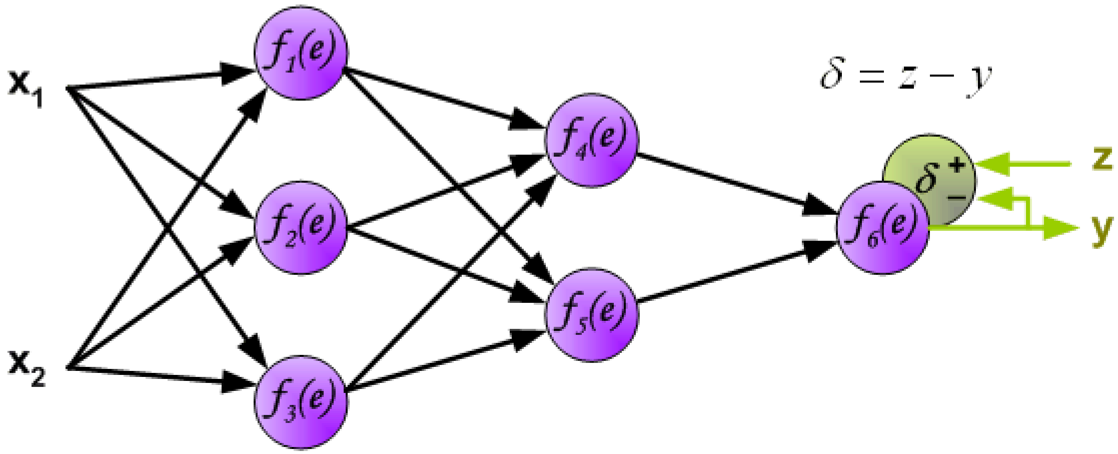
$$y_5 = f_5(w_{15}y_1 + w_{25}y_2 + w_{35}y_3)$$



Пример

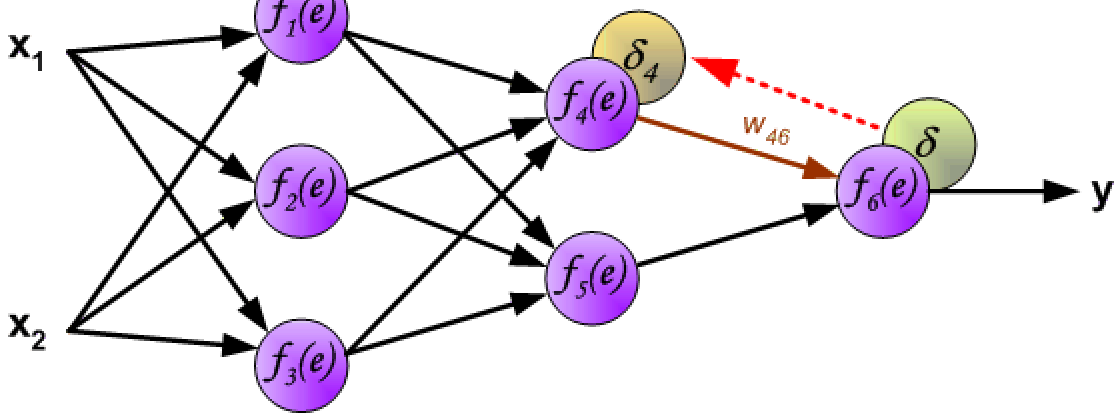


Пример

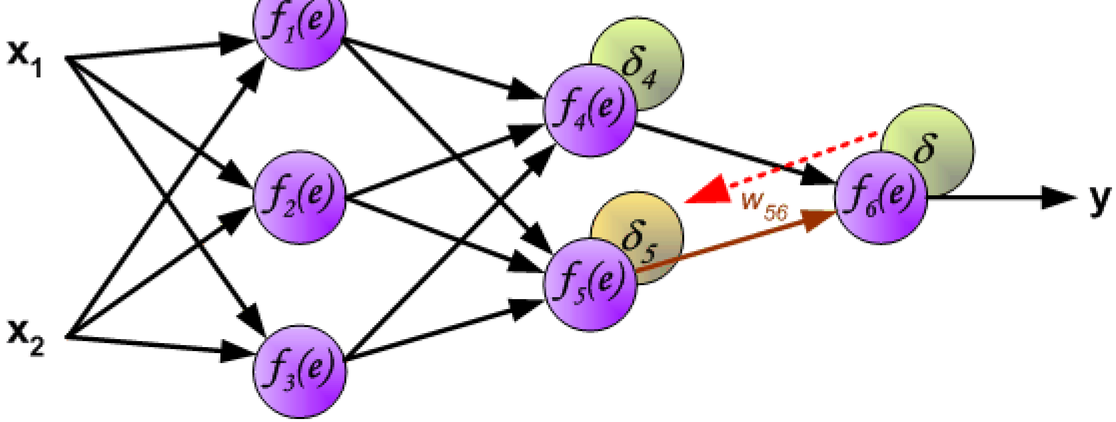


Пример

$$\delta_4 = w_{46}\delta$$

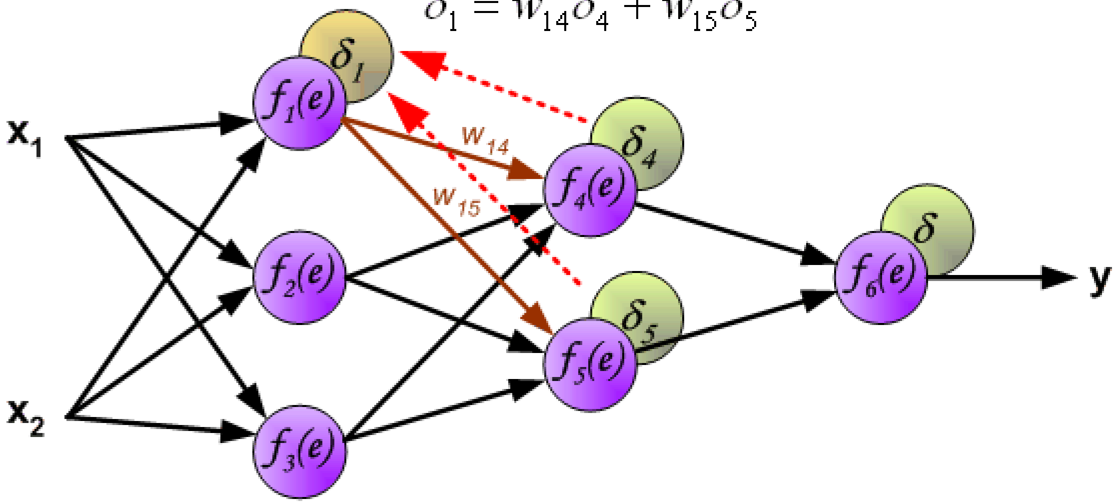


$$\delta_5 = w_{56}\delta$$

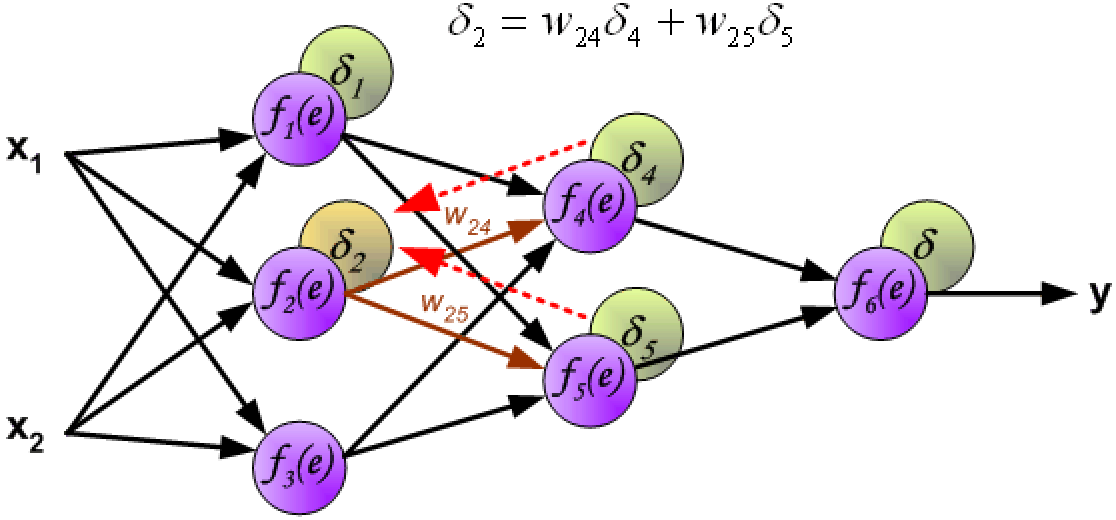


Пример

$$\delta_1 = w_{14}\delta_4 + w_{15}\delta_5$$

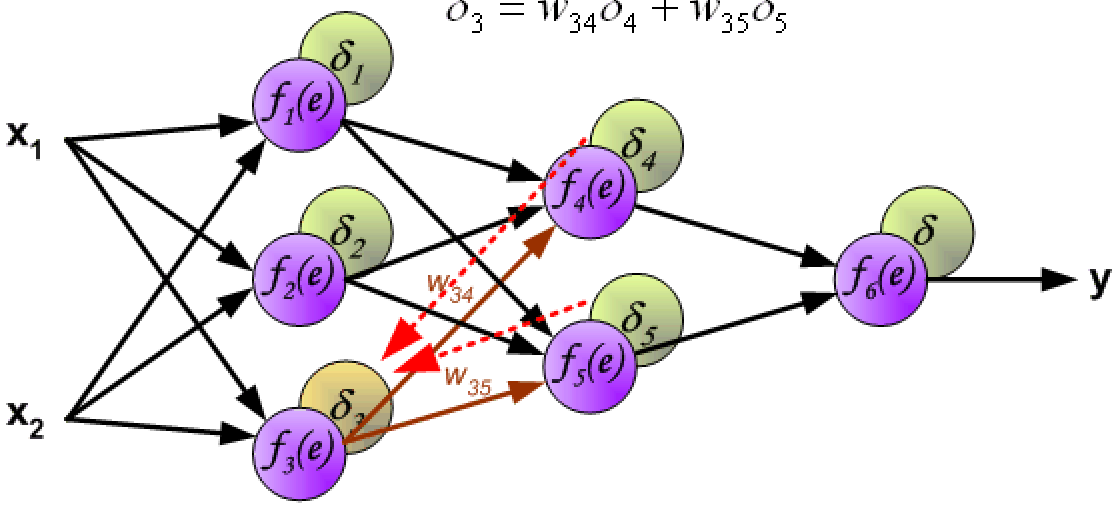


$$\delta_2 = w_{24}\delta_4 + w_{25}\delta_5$$



Пример

$$\delta_3 = w_{34}\delta_4 + w_{35}\delta_5$$

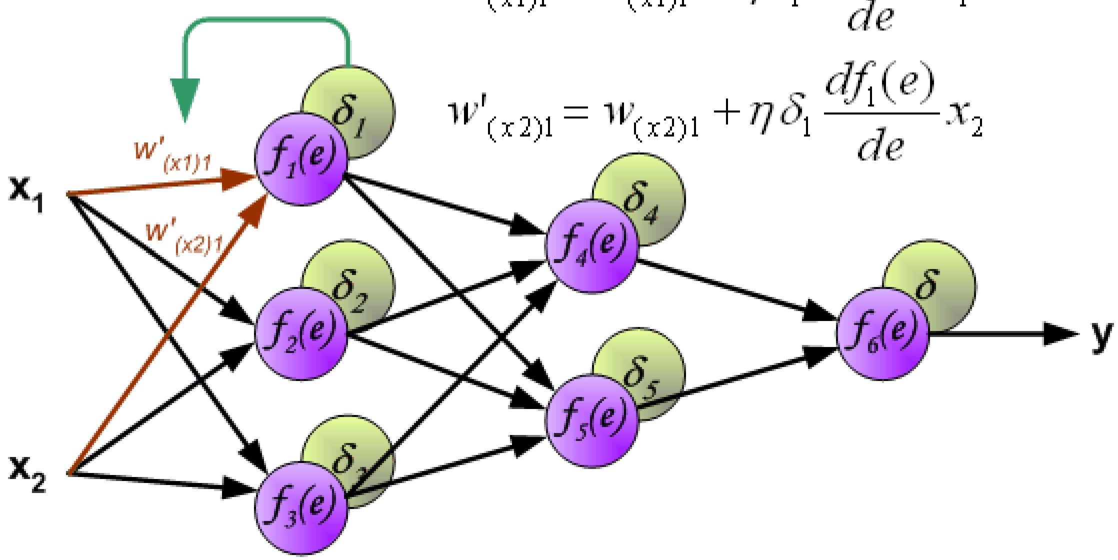


Пример

- $$S(x) = \frac{1}{1 + \exp(-x)}$$
- S'(x) = S(x)*(1 - S(x))

$$w'_{(x1)1} = w_{(x1)1} + \eta \delta_1 \frac{df_1(e)}{de} x_1$$

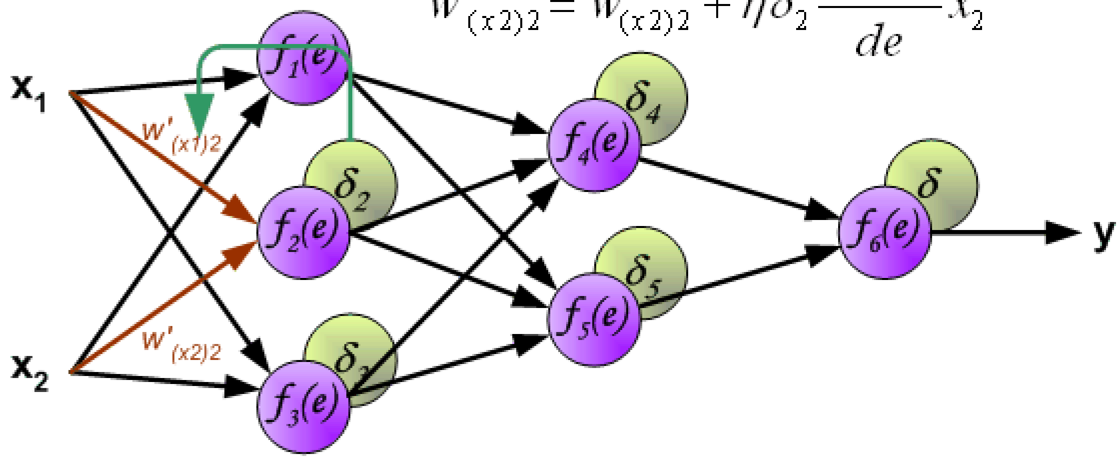
$$w'_{(x2)1} = w_{(x2)1} + \eta \delta_1 \frac{df_1(e)}{de} x_2$$



Пример

$$w'_{(x1)2} = w_{(x1)2} + \eta \delta_2 \frac{df_2(e)}{de} x_1$$

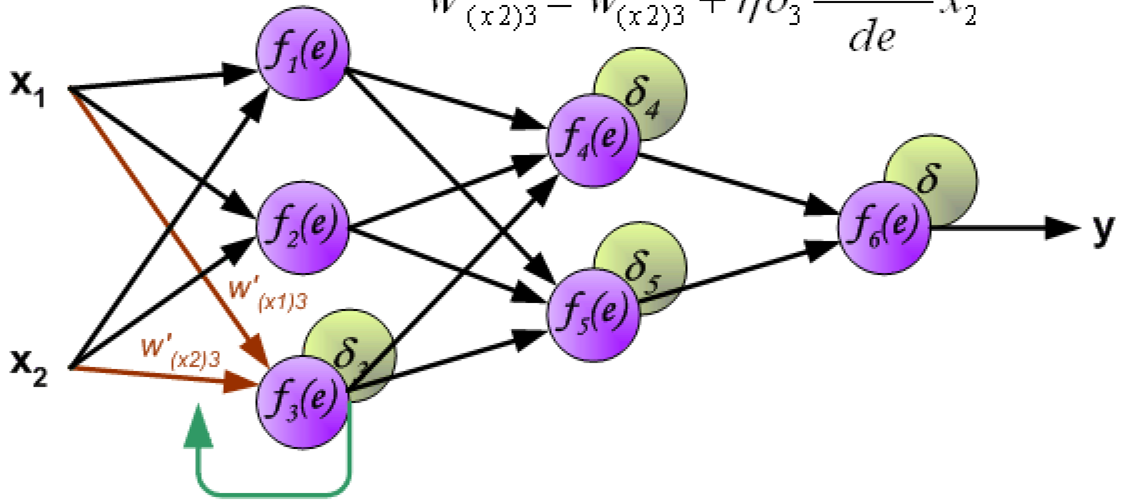
$$w'_{(x2)2} = w_{(x2)2} + \eta \delta_2 \frac{df_2(e)}{de} x_2$$



Пример

$$w'_{(x1)3} = w_{(x1)3} + \eta \delta_3 \frac{df_3(e)}{de} x_1$$

$$w'_{(x2)3} = w_{(x2)3} + \eta \delta_3 \frac{df_3(e)}{de} x_2$$

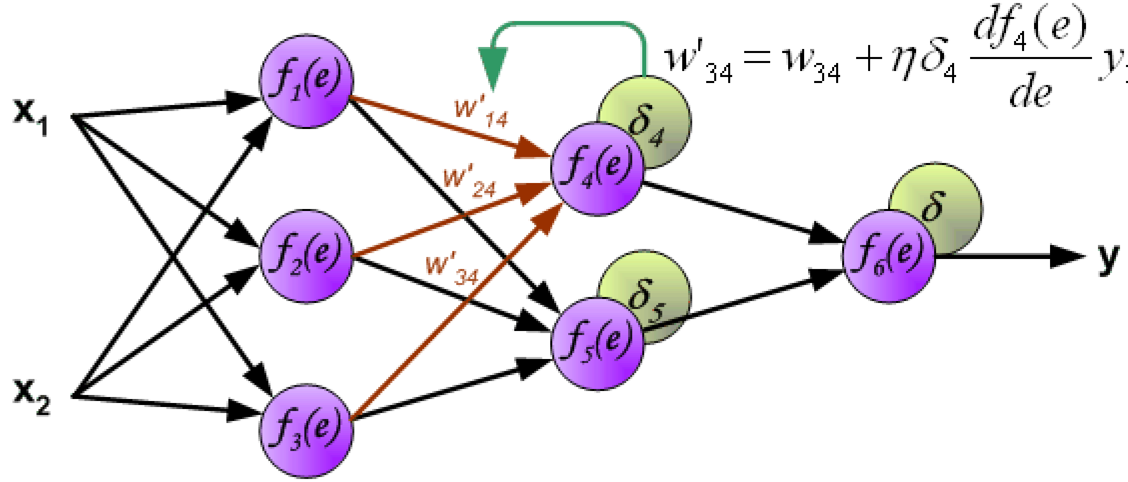


Пример

$$w'_{14} = w_{14} + \eta \delta_4 \frac{df_4(e)}{de} y_1$$

$$w'_{24} = w_{24} + \eta \delta_4 \frac{df_4(e)}{de} y_2$$

$$w'_{34} = w_{34} + \eta \delta_4 \frac{df_4(e)}{de} y_3$$

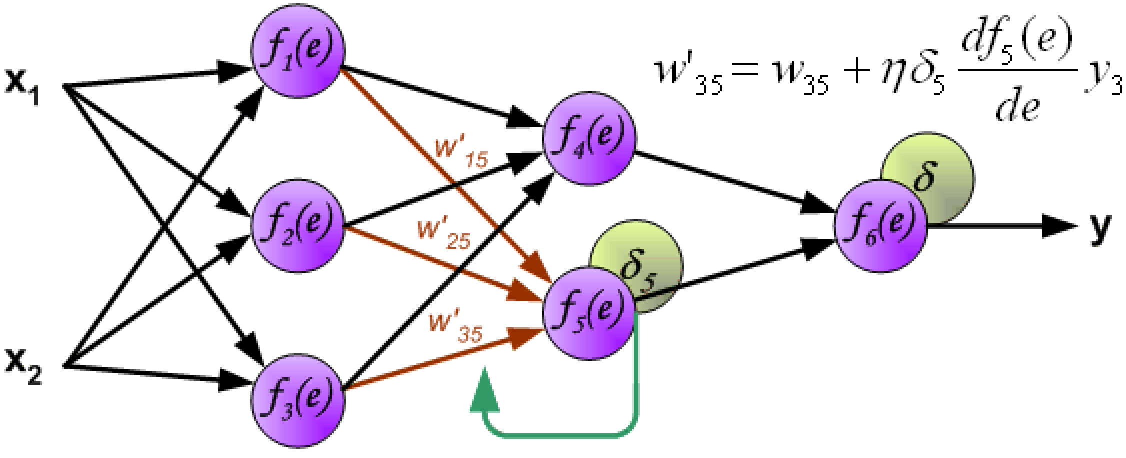


Пример

$$w'_{15} = w_{15} + \eta\delta_5 \frac{df_5(e)}{de} y_1$$

$$w'_{25} = w_{25} + \eta\delta_5 \frac{df_5(e)}{de} y_2$$

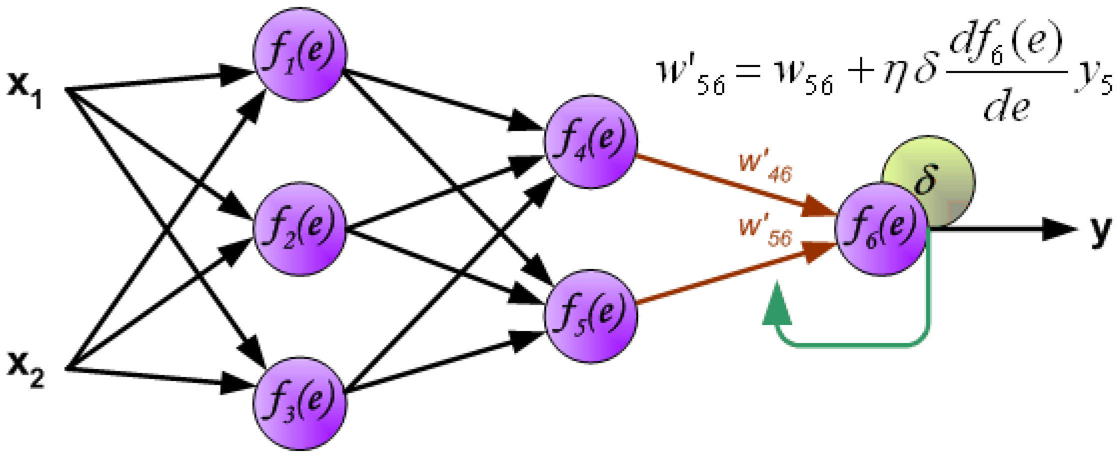
$$w'_{35} = w_{35} + \eta\delta_5 \frac{df_5(e)}{de} y_3$$



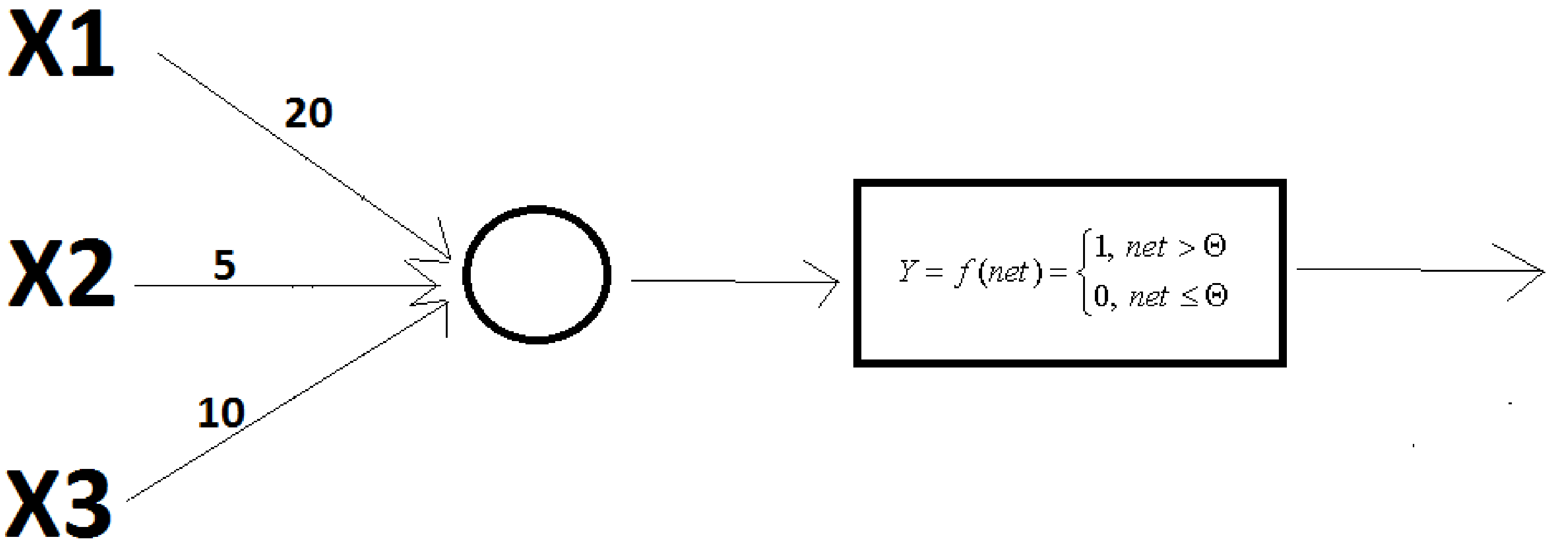
Пример

$$w'_{46} = w_{46} + \eta\,\delta \frac{df_6(e)}{de} y_4$$

$$w'_{56} = w_{56} + \eta\,\delta \frac{df_6(e)}{de} y_5$$



Практика


X1
20
X2
5
X3
10
$Y = f(net) = \begin{cases} 1, & net > \Theta \\ 0, & net \le \Theta \end{cases}$

Практика

Описательные шкалы и градации	Значение
Длина волос:	
– длинные	1
– короткие	0
Наличие брюк:	
– да:	1
– нет	0
Использование духов или одеколона:	
– да:	1
– нет	0

- Пол: 1-юноша, 0-девушка

$$\begin{pmatrix}0\\1\\0\end{pmatrix} -> 1 \qquad \begin{pmatrix}1\\0\\0\end{pmatrix} -> 0 \qquad \begin{pmatrix}1\\1\\1\end{pmatrix} -> 0 \qquad \begin{pmatrix}0\\1\\1\end{pmatrix} -> 1$$