

Многослойный персептрон представляет собой ориентированный взвешенный граф. Обозначим *i*-ый нейрон *l*-го слоя . Обозначим – веса на ребре, соединяющим *i*-ый нейрон *l*-го слоя с *j*-ым нейроном (*l-1)*-го слоя. Введем аналогичные обозначения для байеса *b* – дополнительного нейрона, – *j*-ый нейрон на *l*-ом слое.



Для настройки и обучения нейронной сети нам понадобятся веса на ребрах . Изначально, зададим их случайными значениями.



Обозначим – функцию активации *j*-го нейрона на *l*-ом слое. В качестве функции активации, выберем гиперболический тангенс, задаваемый выражением *f(x) = tanh(x)*. Получим,



Используя вышеописанные обозначения, запишем формулу расчета значений *i*-го нейрона на *l*-ом слое, где сумма ведется по всем нейронам (*l-1)*-го слоя.



Для улучшения работы сети, вводится вектор смещений *b* – дополнительный нейрон на каждом слое. Обозначим – смещение на *l*-ом слое. После введения дополнительных нейронов – смещений *b*, формула примет вид



Цель обратного распространения ошибки состоит в вычислении частных производных функции потерь



Введем квадратичную функцию потерь



Где – ответ нейрона на выходе, а – желаемый ответ на *k*-ый тестовый пример.



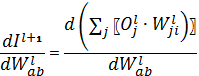
Выпишем алгоритм backpropagation для многослойной сети. Посчитаем производную функции потерь , для этого воспользуемся формулой цепной производной



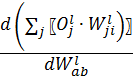
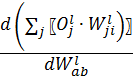
Где и некоторые функции. Применив формулу, получим



Вычислим второй сомножитель



При или при , дробь обращается в ноль, следовательно, мы имеем равенство



По определению

; , при и ,



Подставим выражение в числитель , получим



Где – производная функции



Вычислим первый сомножитель



Распишем второй сомножитель



Осталось вычислить первый сомножитель. Заметим, что мы имеем рекурсивное соотношение



Выпишем базу рекурсии



И рекурсивное соотношение



Мы получили все необходимые расчеты для реализации backpropagation. Осталось обучить нейронную сеть.