Лекция 7. Нейронные сети

В последнее время все большую заслуженную популярность приобретают нейронные сети. Семейство алгоритмов, известное как нейронные сети, недавно пережило свое возрождение под названием «глубокое обучение». Несмотря на то что глубокое обучение сулит большие перспективы в различных сферах применения машинного обучения, алгоритмы глубоко обучения, как правило, жестко привязаны к конкретным случаям использования. В данном разделе мы рассмотрим лишь некоторые относительно простые методы, а именно многослойные персептроны для классификации и регрессии, которые могут служить отправной точкой в изучении более сложных методов машинного обучения.

Одним из простейших типов нейронных сетей является персептрон. Нейронная сеть представляет из себя совокупность нейронов, соединенных друг с другом определенным образом. Рассмотрим один нейрон, представленный на рисунке 1.

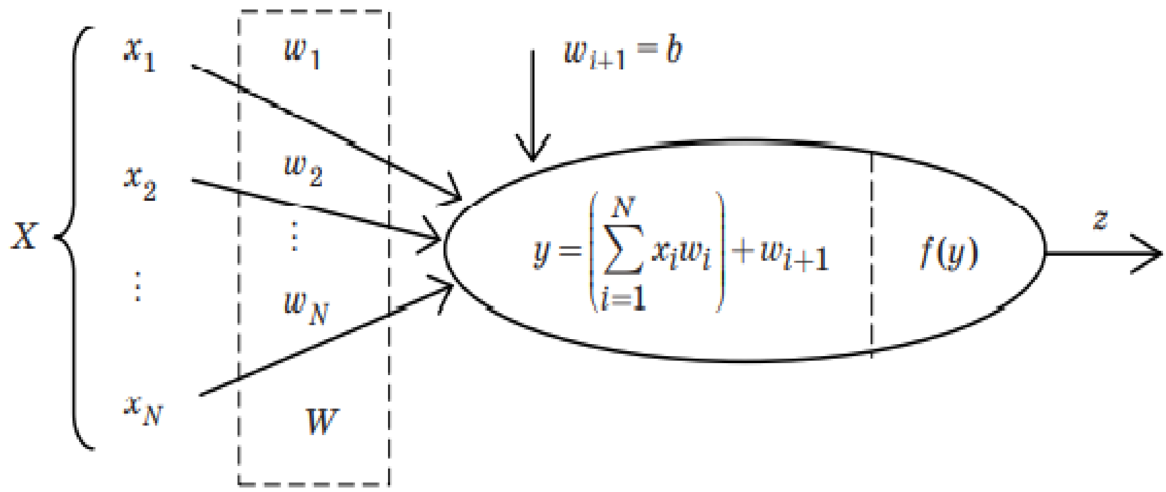


Рисунок 1 –Простейшая нейронная сеть – персептрон

Нейрон – элемент, который вычисляет выходной сигнал (по определенному правилу) из совокупности входных сигналов. На вход нейрона поступает вектор x_i, i=1, 2... N, с соответствующими весами связи w_i, i=1, 2... N. Вес связи определяет, будет ли усилен или ослаблен передаваемый сигнал. Чаще всего структура связей между нейронами представляется в виде матрицы W, которую называют весовой матрицей.

Нейрон вычисляет взвешенную сумму

$$y = \left(\sum_{i=1}^{N} x_i w_i\right) + w_{i+1},$$

здесь w_{i+1} - вес сдвигового нейрона b. Эта сумма передается в функцию активации нейрона $f(y)$, которая вычисляет выходной сигнал нейрона.

Между собой нейроны могут быть соединены абсолютно по-разному, это определяется структурой конкретной сети. Поскольку к каждому нейрону могут приходить несколько входных сигналов, то при моделировании нейронной сети необходимо задать определенное правило комбинирования всех этих сигналов.

При построении нейронных сетей используются различные функции активации некоторые из них представлены в таблице 1.

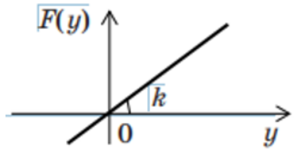
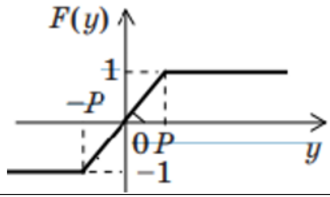
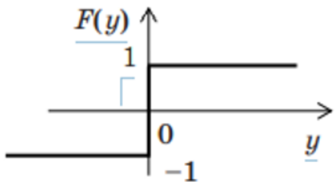
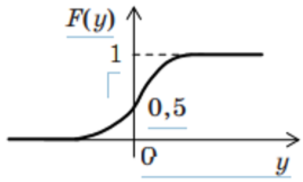
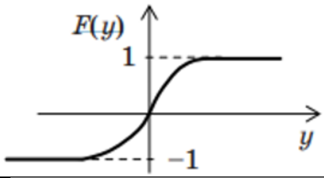
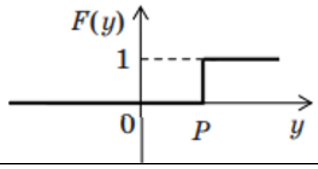
Обучение нейронной сети- это процесс, в котором параметры нейронной сети настраиваются посредством моделирования среды, в которую эта сеть встроена. Тип обучения определяется способом подстройки параметров.

Классификация нейронных сетей по характеру обучения делит их на:

- нейронные сети, использующие обучение с учителем;
- нейронные сети, использующие обучение без учителя.

Нейронные сети, использующие обучение с учителем. Обучение с учителем предполагает, что для каждого входного вектора существует целевой вектор, представляющий собой требуемый выход. Вместе они называются обучающей парой. Обычно сеть обучается на некотором числе таких обучающих пар. Предъявляется выходной вектор, вычисляется выход сети и сравнивается с соответствующим целевым вектором. Далее, веса изменяются в соответствии с алгоритмом, стремящимся минимизировать ошибку. Векторы обучающего множества предъявляются последовательно, вычисляются ошибки и веса подстраиваются для каждого вектора до тех пор, пока ошибка по всему обучающему массиву не достигнет приемлемого уровня. В рамках этой работы будем рассматривать только обучение нейронных сетей с учителем.

Таблица 1 – Функции активации

Название функции	**Описание**	**Графическое представление**
Линейная	$F(y) = ky,$ $k = 0,$ k – коэффициент активации	
Линейная с насыщением	$F(y) = \begin{cases} +1, \lvert y \rvert > P, \\ ky, \lvert y \rvert < 1, \\ -1, y < -P. \end{cases}$	
Определение знака	$F(y) = sign(y)$	
Униполярная сигмоидальная (s-образная)	$F(y) = \frac{1}{1 + \exp(-ky)},$ $k > 0$	
Биполярная сигмоидальная (гиперболический тангенс)	$F(y) = \tanh(ky),$ $k > 0$	
Пороговая	$F(y) = \begin{cases} 1, y \geq P, \\ 0, y < P \end{cases}$	

Успех обучения ИНС во многом зависит от подготовки количественных обучающих данных, которая обычно содержит исключение аномальных наблюдений, заполнение пропущенных данных, а также нормирование входных переменных.

Нормирование означает приведение каждой компоненты входного вектора к интервалу [0,1] или [–1,1]. При известном диапазоне изменения входной переменной $[x_{min}, x_{max}]$ нормирование выполняется по формуле

$$x_N = \frac{x - x_{min}}{x_{max} - x_{min}}.$$

Однослойная нейронная сеть может обучаться по правилу Хебба. Предъявляем на вход персептрона один пример. Если выходной сигнал персептрона совпадает с правильным ответом, то никаких действий предпринимать не надо. В случае ошибки необходимо обучить персептрон правильно решать данный пример. Ошибки могут быть двух типов. Рассмотрим каждый из них.

Первый тип ошибки – на выходе персептрона 0, а правильный ответ – 1. Для того чтобы персептрон выдавал правильный ответ необходимо, чтобы сумма произведений $x \cdot w$ стала больше порогового значения. Поскольку переменные x_i принимают значения 0 или 1, увеличение суммы может быть достигнуто за счет увеличения весов $\mathrm{w_i}$. Однако нет смысла увеличивать веса при переменных x_i, которые равны нулю. Таким образом, следует увеличить веса w_i при тех переменных x_i, которые равны 1. Для закрепления единичных сигналов с x_i, следует провести ту же процедуру и на всех остальных слоях.

Первое правило Хебба. Если на выходе персептрона получен 0, а правильный ответ равен 1, то необходимо увеличить веса связей между одновременно активными нейронами. При этом выходной персептрон считается активным. Входные сигналы считаются нейронами.

Второй тип ошибки – на выходе персептрона 1, а правильный ответ равен нулю. Для обучения правильному решению данного примера следует уменьшить сумму произведений xw. Для этого необходимо уменьшить веса связей w_i при тех переменных x_i, которые равны 1 (поскольку нет смысла уменьшать веса связей при равных нулю переменных x_i). Необходимо также провести эту процедуру для всех активных нейронов предыдущих слоев. В результате получаем второе правило Хебба.

Второе правило Хебба. Если на выходе персептрона получена 1, а правильный ответ равен 0, то необходимо уменьшить веса связей между одновременно активными нейронами.

Таким образом, процедура обучения сводится к последовательному перебору всех примеров обучающего множества с применением правил Хебба для обучения ошибочно решенных примеров. Если после очередного цикла предъявления всех примеров окажется, что все они решены правильно, то процедура обучения завершается.

Нерассмотренными остались два вопроса. Первый – насколько надо увеличивать (уменьшать) веса связей при применении правила Хебба. Второй – о сходимости процедуры обучения.

Теорема о сходимости персептрона. Если существует вектор параметров α, при котором персептрон правильно решает все примеры обучающей выборки, то при обучении персептрона по правилу Хебба решение будет найдено за конечное число шагов.

Теорема о «зацикливании» персептрона. Если не существует вектора параметров α, при котором персептрон правильно решает все примеры обучающей выборки, то при обучении персептрона по правилу Хебба через конечное число шагов вектор весов начнет повторяться.

Модификацией правила обучения Хебба является дельта-правило, которое базируется на идее непрерывного изменения синоптических весов для уменьшения разности (дельта) между значением желаемого и текущего выходного сигнала. Алгоритм обучения однослойной нейронной сети по дельта-правилу (рисунок 2).

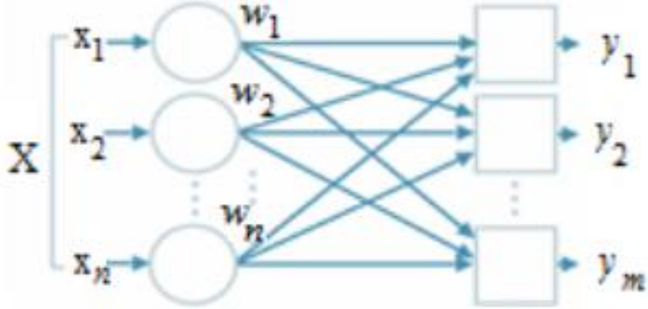


Рисунок 2 – пример обучения нейронной сети

Шаг 1. Инициализация матрицы весов (и порогов, в случае использования пороговой функции активации) случайным образом.

Шаг 2. Предъявление нейронной сети образа (на вход подаются значения из обучающей выборки – вектор X) – берется соответствующий выход (вектор D).

Шаг 3. Вычисление выходных значений нейронной сети (вектор Y).

Шаг 4. Вычисление для каждого нейрона величины расхождения реального результата с желаемым:

$$\varepsilon_i = (d_i - y_i),$$

где d_i – желаемое выходное значение на i-нейроне, y_i – реальное значение на i-нейроне.

Шаг 5. Изменение весов (и порогов при использовании пороговой функции) по формулам

$$w_{ij}(t+1) = w_{ij}(t) - \eta\varepsilon_i x_j,$$

$$\Theta_i(t+1) = \Theta_i(t) - \eta\varepsilon_i$$

где t – номер текущей итерации цикла обучения, w_{ij} – вес связи j-входа с i-нейроном; η – коэффициент обучения (от 0 до 1); x_j – входное значение; Θ_i – пороговое значение i-нейрона.

Шаг 6. Проверка условия продолжения обучения (вычисление значения ошибки и/или проверка заданного количества итераций). Если обучение не завершено, то переходим к шагу 2, иначе заканчиваем обучение.

Пример обучения нейронной сети по дельта-правилу

Для обучения нейронной сети по Δ-правилу необходимо:

1. Графически отобразить структуру нейронной сети. Определить размерность матрицы синоптических весов.

2. Определить обучающую выборку, представив ее в табличном виде.

3. Выбрать входные данные, на которых будет рассматриваться итерация цикла обучения.

4. Следуя алгоритму обучения по Δ-правилу, просчитать итерации цикла и представить новые синоптические веса в матричном виде.

Пример. Просчитать одну итерацию цикла обучения по Δ-правилу однослойной бинарной неоднородной нейронной, состоящей из двух нейронов и имеющей функции активации: гиперболический тангенс (a=1) и пороговую функцию (Θ=0,7). Выход первого нейрона соответствует оператору эквивалентности, а второго – дизъюнкции. В качестве обучающей выборки использовать таблицу истинности для операций эквивалентности и дизъюнкции (не использовать первую строчку таблицы). Синоптические веса задать случайным образом.

Решение.

1. По заданию нейронная сеть состоит из двух нейронов, значит, входов у однослойной нейронной сети будет 2 и выходов 2, а синоптических весов 4. Первый нейрон имеет пороговую функцию активации, второй – гиперболический тангенс.

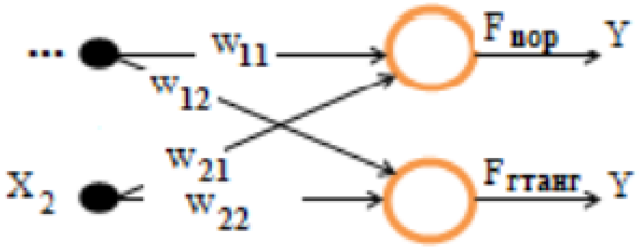


Рисунок 3 – Схема нейронов

2. По заданию нейронная сеть бинарная, поэтому на ее входы могут подаваться только нули и единицы, так как входа 2, то возможных комбинаций входных значений будет 4 (обучающая выборка будет состоять из 4-х

векторов). Выход первого нейрона согласно задания соответствует оператору эквивалентности, а второго – дизъюнкции. Таблица с обучающей выборкой представлена ниже.

Таблица 1 – Обучающая выборка

x_1	x_2	D_1	D_2
0	0	1	0
0	1	0	1
1	0	0	1
1	1	1	1

3. Пусть в качестве вектора обучения будет рассматриваться 3-я строка таблицы.

4. Следуя алгоритму обучения по Δ-правилу, выполним 6 шагов:

Шаг 1. Зададим матрицу весов случайным образом из интервала [0,1] (таблица 2).

Таблица 2 – Матрица весов

$w_{ij}(1)$	1	2
1	0,7	0,9
2	0,5	0,2

Шаг 2. Вектор $X_3 = \{1, 0\}$, вектор $D_3 = \{0, 1\}$.

Шаг 3. Вычисление выходов нейронной сети Y:

$\Theta = 0{,}7$;

$$y_1 = x_1 w_{11} + x_2 w_{21} \ = \ 1 \cdot 0{,}7 + 0 \cdot 0{,}5 \ = \ 0{,}7;$$

$$Y_1 = F(y_1) = \begin{cases} 1 \text{ при } y_1 \geq \Theta \\ 0 \text{ при } y_1 < \Theta \end{cases} = \begin{cases} 1 \text{ при } 0{,}7 \geq 0{,}7 \\ 0 \text{ при } 0{,}7 < 0{,}7 \end{cases} = 1$$

$\alpha = 1$;

$$y_2 = x_1 \cdot w_{12} + x_2 \cdot w_{22} \ = \ 1 \cdot 0{,}9 + 0 \cdot 0{,}2 \ = \ 0{,}9;$$

$Y_2 = F(y_2) = \frac{e^{0,9} + e^{-0,9}}{e^{0,9} - e^{-0,9}} \approx 1{,}39.$

Шаг 4. Вычисление величины расхождения реального результата с желаемым:

$$\varepsilon_1 = (d_1 - y_1) = (0 - 1) = -1;$$

$$\varepsilon_2 = (d_2 - y_2) = (1 - 1{,}39) = -0{,}39.$$

Шаг 5. Зададим η – коэффициент обучения от 0 до 1 и изменим веса:

$$\eta = 0{,}8;$$

$$w_{11}(2) = w_{11}(1) - 0{,}8 \cdot \varepsilon_1\, x_1 = 0{,}7 - 0{,}8(-1)\,1 = 1{,}5;$$

$$w_{21}(2) = w_{21}(1) - 0{,}8 \cdot \varepsilon_1\, x_2 = 0{,}5 - 0{,}8(-1)\,0 = 0{,}5;$$

$$\Theta_1(2) = \Theta_1(1) - 0{,}8 \cdot \varepsilon_1 = 0{,}7 - 0{,}8(-1) = 1{,}5;$$

$$w_{12}(2) = w_{12}(1) - 0{,}8 \cdot \varepsilon_2\, x_1 = 0{,}9 - 0{,}8(-0{,}39)\,1 = 1{,}212;$$

$$w_{22}(2) = w_{22}(1) - 0{,}8 \cdot \varepsilon_2\, x_2 = 0{,}2 - 0{,}8(-0{,}39)\,0 = 0{,}2.$$

Матрица весов представлена в таблице 3.

Таблица 3 – Итоговая матрица весов

$w_{ij}(2)$	1	2
1	1,5	1,2112
2	0,5	0,2

Шаг 6. Вычислим среднеквадратичную ошибку:

$$\varepsilon = \sum_{i=1}^{N}(d_i - Y_i)^2 = \sum_{i=1}^{N}\varepsilon^2 = \varepsilon_1^2 + \varepsilon_2^2 = (-1)^2 + (-0{,}39)^2 = 1{,}156869.$$

Здесь N – количество нейронов. Аналогично выполняются последующие итерации, пока среднеквадратичная ошибка не станет меньше заданной величины.

Контрольные вопросы по теме:

1. Области применения нейросетевого моделиования.

2. Понятие (искусственного) нейрона.

3. Понятие (искусственной) нейронной сети.

4. Понятие функции активации.

5. Требования к функции активации. Виды функций активации

7. Понятие обучения (настройки) нейронной сети.

8. Понятие обучающего и тестового множеств.

9. Обучение нейронной сети по дельта-правилу.

10. Многослойный персептрон.