

1 Теоретическая часть

Для того чтобы обучить 3-х слойную нейронную сеть с 1 скрытым слоем, воспользуемся методом обратного распространения ошибки. Основная идея работы этого метода состоит в распространении сигналов ошибки от выходов сети к её входам. Пусть x_1, \dots, x_n - множество входов. Обозначим через ω_{jk} вес ребра, соединяющего i и j нейроны, o_j - выход j -го нейрона. Так как обучение происходит с учителем, для каждого входного вектора x_j мы знаем правильный ответ сети $t_k, k \in \text{Нейроны выходного слоя}$. Для того, чтобы узнать насколько ответ сети отличается от ожидаемого, будем использовать функцию ошибки - перекрестная энтропия:

$$E = - \sum_{j=1}^n t_j \log(o_j) \quad (1)$$

Метод обратного распространения ошибки можно разделить на 2 части:

Прямой ход:

Перед обучением сети проинициализируем веса ω_{ij} и смещения ω_{0j} случайными малыми значениями

1. Подадим на входной слой значения x_1, \dots, x_n
2. Каждый входной нейрон отправляет полученный сигнал всем нейронам в следующем скрытом слое
3. Каждый скрытый нейрон суммирует полученные значения, умножает на соответствующие веса $s_j = \omega_{0j} + \sum_i x_i \omega_{ij}$ и применяет функцию активации к каждому нейрону:

$$f(s_j) = \frac{1}{1 + e^{-2s_j}}$$

после чего отправляет получившиеся значения на выходной слой

4. Каждый выходной нейрон находит взвешенную сумму, как и нейроны скрытого слоя, после чего применяет функцию активации:

$$f(s_j) = \frac{e^{s_j}}{\sum_{j=1}^n e^{s_j}}$$

Обратный ход:

1. Для выходного слоя на основе полученных и ожидаемых значений рассчитывается ошибка по формуле (1) и нормируется. Одним из критериев остановки обучения сети можно считать полученное значения.
2. Для каждого нейрона выходного слоя рассчитывается ошибка:

$$\frac{\partial E}{\partial s_i} = \sum_{j=1}^n \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial s_i}; \quad \frac{\partial E}{\partial o_j} = -\frac{\partial t_j \log(o_j)}{\partial o_j} = -\frac{t_j}{o_j} \quad (2)$$

$$\frac{\partial o_j}{\partial s_i} = \begin{cases} \frac{\partial \frac{e^{s_j}}{\sum_k e^{s_k}}}{\partial s_j} = \frac{e^{s_j} \sum_k e^{s_k} - (e^{s_j})^2}{(\sum_k e^{s_k})^2} = \frac{e^{s_j}}{\sum_k e^{s_k}} - \frac{(e^{s_j})^2}{(\sum_k e^{s_k})^2} = o_j(1 - o_j), & \text{если } i = j \\ \frac{\partial \frac{e^{s_i}}{\sum_k e^{s_k}}}{\partial s_j} = \frac{-e^{s_i} e^{s_j}}{(\sum_k e^{s_k})^2} = -o_i o_j, & \text{если } i \neq j \end{cases} \quad (3)$$

Из (2) и (3) следует:

$$\delta_i = \frac{\partial E}{\partial s_i} = \sum_j o_i t_j - t_i(1 - o_i) = -t_i + t_i o_i + o_i \sum_j t_j = -t_i + o_i(t_i + \sum_j t_j) = o_i - t_i \quad (4)$$

3. Каждый нейрон скрытого слоя суммирует входящие ошибки (от нейронов в последующем слое) и умножает величину полученной ошибки на производную активационной функции:

$$\begin{aligned}\delta_j &= \frac{\partial E}{\partial s_j} = \sum_{k \in \text{Выходной слой}} \frac{\partial E}{\partial s_k} \frac{\partial s_k}{\partial s_j} \\ \frac{\partial s_k}{\partial s_j} &= \frac{\partial s_k}{\partial o_j} \frac{\partial o_j}{\partial s_j}; \quad \frac{\partial s_k}{\partial o_j} = \omega_{jk}; \\ \frac{\partial o_j}{\partial s_j} &= \frac{\partial \left(\frac{1}{1 + e^{-2s_j}} \right)}{\partial s_j} = -\frac{1}{(1 + e^{-2s_j})^2} \frac{\partial (1 + e^{-2s_j})}{\partial s_j} = \\ &= 2e^{-2s_j} \frac{1}{(1 + e^{-2s_j})^2} = \frac{1 + e^{-2s_j} - 1}{(1 + e^{-2s_j})^2} = 2o_j(1 - o_j) \\ \text{Где } \frac{\partial E}{\partial s_k} &= \delta_k - \text{поправка, вычисленная для } k\text{-го узла} = \quad (4)\end{aligned}$$

Таким образом получаем для нейронов выходного слоя ошибку $\delta_j = o_j - t_j$, а для нейронов скрытого слоя $\delta_j = 2o_j(1 - o_j)(\sum_k \delta_k \omega_{jk})$

4. Каждый выходной и скрытый нейрон изменяет веса своих связей с нейронами предыдущих слоев по формуле:

$$\omega_{ij} = \omega_{ij} - \eta \delta_j o_i$$

Данная процедура повторяется для каждого тренировочного набора. Обучение останавливается после обучения всего тестового набора определенное количество раз или по достижении заданной ошибки между ожидаемым и реальным выходами.

2 Программная реализация

Программа состоит из нескольких модулей:

Библиотека:

1. *Neuron*: хранит информацию о текущем состоянии нейрона, а также ошибку, необходимую для обучения сети. Каждый нейрон имеет свою функцию активации и два списка нейронов: входящие и исходящие, где для каждой пары определен вес ребра между ними.
2. *Layer*: хранит список нейронов, принадлежащих данному слою, а также его тип: входной, скрытый, выходной. Нейроны смещения закреплены за слоем и представлены как вектор весов для каждого нейрона следующего слоя. Реализует сбор значений с предыдущего слоя и их преобразование с помощью функции активации.
3. *LayerBinder*: отвечает непосредственно за логику алгоритма, реализует прямой и обратный ход.
4. *Вспомогательные классы*: отвечают за расчет значений/производных функций; генерацию случайных весов.

Приложение: В методе Main происходит загрузка тестовых и тренировочных MNIST данных, их парсинг; обработка входных параметров: количество нейронов в скрытом слое, необходимая максимальная ошибка, количество эпох, скорость обучения; вызов библиотечных классов для обучения сети и проверки корректности работы.

Тесты: Написано 5 тестов: 3 теста отвечают за работу функций активации, функцию ошибки и их производных, 2 оставшихся теста проверяют корректность работы прямой и обратной части алгоритма. С помощью тестов было найдено несколько ошибок в процессе рефакторинга кода.

3 Результаты

Нейронная сеть была натренирована с помощью MNIST данных (60.000 тренировочных наборов) и проверена с помощью тестовых данных (10.000 наборов). Использовались различные конфигурации сети:

Нейронов в скрытом слое	Макс. ошибка	Кол-во эпох	Скорость обучения	Трен. наборов неправильно	Тест. наборов неправильно
200	0.001	25	0.01	14	247
200	0.001	25	0.005	141	273
200	0.001	25	0.008	34	256
200	0.0005	35	0.01	0	239
200	0.0005	35	0.005	30	257
200	0.0005	35	0.008	3	246
300	0.001	25	0.01	6	228
300	0.001	25	0.005	110	262
300	0.001	25	0.008	19	234
300	0.0005	35	0.01	0	207

Как можно видеть, наиболее хорошо себя показала сеть с последней конфигурацией.

Процент правильных ответов на тренировочном наборе: 100%

Процент правильных ответов на тестовом наборе: 97.93%