

Back Propagation Method

Описание структуры используемой нейронной сети

Используемая в данной задаче нейронная сеть состоит из двух слоёв (один скрытый). Сеть решает задачу классификации на десять классов – определяет по изображению, на котором рукописная цифра, что это за цифра. Количество входов равно $784 = 28 * 28$ (по количеству пикселей). Количество выходов равно 10 (по количеству классов – цифр). Количество нейронов скрытого слоя задаётся пользователем приложения.

В качестве функций активаций используются:

- на нейронах скрытого слоя – логистическая функция

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

- на выходных нейронах – функция softmax

$$f(x)_i = \frac{e^{x_i}}{\sum_k e^{x_k}}$$

В качестве функции потерь используется функция cross-entropy:

$$E(t, o) = - \sum_k t_k \log(o_k)$$

, где t – массив правильных (известных из тренировочной разметки) значений на выходах нейронов, а o – массив, того что реально получилось на выходе сети.

Пошаговый алгоритм

1. Инициализация всех весов малыми значениями
2. **Цикл** по всем картинкам из тренировочной (размеченной) выборки:
3. Прямой проход
 - a. Умножение всех входов на матрицу весов скрытого слоя
 - b. Суммирование взвешенных входов и смещения для каждого нейрона скрытого слоя
 - c. Вычисление функции активации для каждого нейрона скрытого слоя (логистическая функция)
 - d. Умножение всех выходов скрытого слоя на матрицу весов выходного слоя
 - e. Суммирование взвешенных выходов скрытого слоя и смещения для каждого выходного нейрона
 - f. Вычисление функции активации (softmax) для каждого выходного нейрона (значение выхода равно вероятности соответствующей цифры быть ответом)
4. Обратный проход
 - a. Вычисление функции потерь (cross-entropy) на основе выходов и правильного ответа к текущему изображению
 - b. Вычисление поправок к весам и смещениям

$$\Delta w_{i,j} = -n \frac{\partial E}{\partial w_{i,j}}$$

, где $0 < n < 1$ – параметр «скорости обучения», $w_{i,j}$ – вес ребра, соединяющего i -ый узел «исходящего» слоя нейронов и j -ый узел – «входящего». При этом,

$$\frac{\partial E}{\partial w_{i,j}} = \frac{\partial E}{\partial S_j} \frac{\partial S_j}{\partial w_{i,j}} = o_i \frac{\partial E}{\partial S_j}$$

, где $o_i = f(S_i)$ – выход i -ого узла «исходящего» слоя нейронов, S_j – взвешенная сумма всех «нейронов-доноров» j -ого узла.

$$\begin{aligned} \frac{\partial E}{\partial S_j} &= \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial S_j} = \left(\frac{\partial}{\partial o_j} \left(- \sum_k t_k \log(o_k) \right) \right) \left(\frac{\partial f(S)}{\partial S} \Big|_{S=S_j} \right) \\ &= \left(- \frac{\partial}{\partial o_j} (t_j \log(o_j)) \right) \left(\frac{e^{S_j}(S - e^{S_j})}{S^2} \right) \\ &= \left(- \frac{t_j}{o_j \ln(2)} \right) (o_j(1 - o_j)) = \frac{o_j - t_j}{\ln(2)} = \delta_j \end{aligned}$$

Для нейронов выходного слоя:

$$\delta_j = \frac{o_j - t_j}{\ln(2)}$$

Для нейронов скрытого слоя:

$$\delta_j = o_j(1 - o_j) \sum_{k \in \text{Children}(j)} \delta_k w_{j,k}$$

Для всех нейронов:

$$\Delta w_{i,j} = -n\delta_j o_i$$

Для всех смещений:

$$\Delta b_j = -n\delta_j$$

с. Изменение весов и смещений, переход к новой итерации цикла

Работа с приложением

Пользователь может вызывать приложение со следующими параметрами:

- Количество нейронов скрытого слоя (-*hu* 800)
- Параметр скорости обучения (-*n* 0.01)
- Путь к директории с выборками (-*p* "data")
- Количество эпох (-*e* 10)

Если один из параметров не задан пользователем, будет использовано значение по умолчанию (указано в скобках).

Результаты обучения

Разработанная реализация показывает следующие результаты:

Hidden Units	Learning Rate	Epochs	Accuracy
300	0.0005	10	0.8212
300	0.001	10	0.8229
300	0.005	10	0.7207
800	0.0005	10	0.8404
800	0.001	10	0.8346
800	0.005	10	0.7222
300	0.001	25	0.9009
800	0.0005	25	0.8546