

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего  
образования  
«Национальный исследовательский  
Нижегородский государственный университет им. Н.И. Лобачевского»  
(ННГУ)

Институт информационных технологий, математики и механики

## **Отчет по лабораторной работе №1**

На тему:

**«Реализация метода обратного распространения ошибки для  
двуслойной полностью связанной нейронной сети»**

**Выполнил:**  
студент гр. 381603-3м  
Аверин Андрей

Нижний Новгород  
2018

## Оглавление

Постановка задачи.....	3
Метод обратного распространения ошибки.....	4
Общая схема метода обратного распространения ошибки.....	5
Вывод математических формул.....	6
Программная реализация.....	8
Результаты экспериментов.....	9

## Постановка задачи

Целью данной лабораторной работы является изучение метода обратного распространения ошибки для обучения многослойных искусственных нейронных сетей и его реализация на примере полностью связанной нейронной сети.

В качестве демонстрации работоспособности реализованного метода будет решена задача классификации рукописных цифр из набора данных MNIST() с помощью полностью связанной нейронной сети с одним скрытым слоем.

В процессе выполнения лабораторной работы ставятся следующие задачи

1. Изучение общей схемы метода обратного распространения ошибки.
2. Вывод математических формул для всех этапов метода.
3. Разработка программной реализации метода обратного распространения ошибки.
4. Тестирование демонстрационного приложения для классификации рукописных цифр с помощью разработанной программной реализации.

## Метод обратного распространения ошибки

Пусть у искусственной нейронной сети есть множество входов  $Inputs = \{x_1, x_2, \dots, x_N\}$ , множество выходов  $Outputs = \{u_1, u_2, \dots, u_M\}$  и множество внутренних узлов, расположенных на скрытых слоях.

Вес связи соединяющей  $i$ -ый и  $j$ -ый узлы обозначим через  $\omega_{i,j}$ , а через  $z_i$  – выход  $i$ -го узла.

В качестве функции ошибки нейронной сети для задачи классификации рассматривается кросс-энтропия:

$$E(\omega) = \frac{-1}{L} \sum_{k=1}^L \sum_{j=1}^M y_j^{(k)} \cdot \ln(u_j^{(k)}), y_j^{(k)} = 1 \leftrightarrow y^{(k)} \in j\text{-му классу}, \quad (1)$$

Где:

- $L$  – количество примеров в обучающей выборке,
- $M$  – количество классов и нейронов выходном слое,
- $y^{(k)} = (y_j^{(k)})_{j=1, \dots, M} \in Y$  – метка входного примера  $x^{(k)} = (x_j^{(k)})_{j=1, \dots, N} \in X$  из обучающей выборки,
- $u^{(k)} = (u_j^{(k)})_{j=1, \dots, M}$  – выход нейронной сети для того же примера.

Задача обучения нейронной сети может быть сформулирована следующим образом:

$$E(\omega) \rightarrow \min, \forall (x, y) \in (X, Y) \quad (2)$$

Задача (2) может быть решена численными методами, например, методом градиентного спуска или квазиньютоновскими методами.

В данной лабораторной работе рассматривается последовательный (стохастический) режим обучения нейронной сети. В таком режиме корректировка весов выполняется после каждого предъявления примера из обучающей выборки.

В качестве критерия остановки можно использовать фиксированное число предъявлений всей обучаемой выборки (максимально допустимое количество эпох), достигнутая точность поиска минимума функции потерь, либо слабое изменение весов.

## Общая схема метода обратного распространения ошибки

Общая схема метода обратного распространения ошибки может быть представлена следующим образом:

1. Инициализация весов  $\omega$  и установка параметров нейронной сети.
2. Пока не выполнится один из критериев остановки (достигнуто максимальное число эпох или достигнутая точность обучения):
  - 1) Прямой проход нейронной сети
    - (1) Подаем входные параметры системы и вычисляем значения выходных сигналов нейронов поочередно на каждом из слоев.
  - 2) Обратный проход нейронной сети
    - (1) Вычисляем ошибку на каждом из слоев
    - (2) Делаем корректировку весов
3. Полученные значения весов  $\omega$  использовать для дальнейшей классификации.

## Вывод математических формул

Как видно из схемы метода обратного распространения ошибки, описанном в предыдущей секции, необходимым условием реализации и применения данного метода является знание частных производных и градиентов функции потерь на всех слоях нейронной сети.

Выход  $z_j$ ,  $j$ -го нейрона входного слоя:

$$z_j = x_j, j \in \{1, \dots, N\}, \quad (3)$$

где  $x_j$  – значение  $j$ -ой компоненты входного вектора, что в данной лабораторной работе является интенсивностью  $j$ -го пикселя.

Выход  $z_j$   $j$ -го нейрона скрытого слоя вычисляется следующим образом:

$$z_j = f\left(\sum_{k \in \text{Parents}(j)} \omega_{k,j} \cdot z_k\right), \quad (4)$$

где  $f(\cdot)$  – функция активации слоя.

В данной лабораторной работе в качестве функции активации скрытых слоёв рассматривается сигмоидальная функция

$$f(x) = \frac{1}{1 + \exp(-\beta x)}, \quad (5)$$

где  $\beta$  – параметр функции активации.

Выход  $z_j$ ,  $j$ -го нейрона выходного слоя вычисляется аналогично (4), только в качестве функции активации рассматривается функция Softmax:

$$f(x_j) = \frac{\exp(x_j)}{\sum_i \exp(x_i)}. \quad (6)$$

Формулы (4) – (6) позволяют вычислить выход нейронной сети, а именно выполнить прямой ход алгоритма обратного распространения ошибки

Перед началом обратного хода необходимо вычислить градиент функции ошибки (1) по сумматору нейронов выходного слоя с функцией активации (6) для одного предъявленного

примера из обучающей выборки при условии  $\sum_{j=1}^M y_j = 1$ .

$$s_j = \ln \sum_{m=1}^M \exp(s_m)$$

$$y_j() = \sum_{j=1}^M \square$$

$$\frac{\partial E(\omega)}{\partial s_j} = \frac{-\partial}{\partial s_j}$$

$$- \left( -y_1 \frac{\exp(s_j)}{\sum_{m=1}^M \exp(s_m)} - \dots - y_{j-1} \frac{\exp(s_j)}{\sum_{m=1}^M \exp(s_m)} + y_j \left( 1 - \frac{\exp(s_j)}{\sum_{m=1}^M \exp(s_m)} \right) - y_{j+1} \frac{\exp(s_j)}{\sum_{m=1}^M \exp(s_m)} - \dots - y_M \frac{\exp(s_j)}{\sum_{m=1}^M \exp(s_m)} \right) = u_j - y_j, \quad (7)$$

$$\text{где } u_j = \frac{\exp(s_j)}{\sum_i \exp(s_i)}.$$

Ошибка в узлах скрытого слоя является произведением взвешенной ошибки на последующем слое на производную функции активации этого слоя.

Производные выбранных функций активации (5) – (6) могут быть вычислены с использованием значений самой функции.

Таким образом в случае сигмоидальной функции активации (5) получаем:

$$\frac{df(x)}{dx} = \beta f(x)(1 - f(x)) \quad (8)$$

Подстановка полученных формул (7) и (8) в общую схему метода обратного распространения ошибки позволяет вычислить ошибки на всех слоях нейронной сети и произвести коррекцию весов для минимизации функции ошибки.

Коррекция весов производится по следующей формуле:

$$\omega_{i,s}^{(r+1)} = \omega_{i,s}^{(r)} - \frac{\eta \partial E(\omega)}{\partial \omega_{i,s}}. \quad (9)$$

Где:

- $\eta$  – скорость обучения,
- $r$  – порядковый номер эпохи.

## **Программная реализация**

Программная реализация метода, была разработана на языке Python с использованием библиотеки numpy в соответствии с приведённым выше алгоритмом и выведенными формулами для коррекции весов.



## Результаты экспериментов

Точность классификации определялась согласно формуле

$$precision = \frac{correct\ classification}{correct\ classification + false\ classification} \quad (10)$$

При проведении экспериментов в качестве критерия остановки использовался комбинированный подход – максимальное количество эпох (  $max_{epoch} h=20$  ), и порог достигнутой точности на валидационной выборке (  $max_{precision}=0.98$  ).

Начальные веса инициализируются случайными значениями из равномерного распределения с границами 0.003 и 0.007.

### Результаты проведенных экспериментов:

Количество нейронов на скрытом слое	Скорость обучения скрытого слоя	Количество эпох	Достигнутая точность на тестовой выборке
100	0.05	20	0.9762
150	0.08	20	0.9771
100	0.1	20	0.9721
50	0.05	20	0.9660
200	0.1	20	0.9750

