

## Задание № Б-2.1.2. Нейронные сети

### 1. Введение

В этом упражнении вы реализуете нейронную сеть для распознавания рукописных цифр.

Описание файлов задания:

- ex3\_nn.m - главный скрипт для выполнения упражнения,
- ex3data1.mat – обучающая выборка с картинками рукописных цифр,
- ex3weights.mat – значения весов для обученной нейронной сети,
- displayData.m – функция для визуализации обучающей выборки,
- fmincg.m – минимизирующий функционал,
- sigmoid.m - сигмоида,
- \* predict.m – функция прогноза нейронной сети.

(\*) – файлы, с которыми вам необходимо работать.

### 3. Нейронные сети

В предыдущем упражнении вы реализовали мультиклассовую логистическую регрессию для распознавания рукописных цифр. Однако логистическая регрессия не может формировать более сложные гипотезы, поскольку это всего лишь линейный классификатор.

В этой части упражнения вы реализуете нейронную сеть для распознавания рукописных цифр, используя ту же обучающую выборку, что и раньше. Для этого вы будете использовать параметры нейронной сети, которую мы уже обучили. Ваша цель – реализовать алгоритм прямого распространения, использующий уже готовые веса для работы сети. В следующем упражнении вы реализуете алгоритм обратного распространения ошибок для обучения нейронной сети.

Предоставленный скрипт ex3\_nn.m поможет вам выполнить это упражнение.

#### 3.1 Представление модели

Наша нейронная сеть показана на Рисунке 2. Она состоит из 3 слоев (входной, скрытый и выходной). Напомним, что входные данные представляют собой значения пикселей изображений. Поскольку изображения имеют размер 20x20, это дает нам 400 узлов<sup>1</sup> входного слоя (не считая специального узла, который всегда выдает +1). Как и прежде, обучающие данные будут загружены в переменные  $X$  и  $y$ .

Вам даются матрицы весов  $\theta_1$  и  $\theta_2$  уже обученной нейронной сети. Они хранятся в файле ex3weights.mat и загружаются скриптом ex3\_nn.m в переменные Theta1 и Theta2. Они имеют размерности, соответствующие нейронной сети с 25 нейронами скрытого слоя<sup>2</sup> и 10 нейронами выходного слоя.

---

<sup>1</sup> Строго говоря, элементы входного слоя нельзя называть нейронами, потому что в них не происходит никаких вычислений. Они просто являются своего рода регистрами (входными переменными), куда мы загружаем элементы входного вектора. Поэтому подобные элементы нейронной сети мы будем называть *узлами*, а не нейронами. Соответственно, у входного слоя нет никаких весов, потому что  $i$ -тый элемент входного вектора просто «копируется» в  $i$ -тый узел входного слоя. Как правило, слои нейронной сети нумеруются с нуля, из-за чего слой с номером 1 – это уже первый скрытый слой с настоящими нейронами, а матрица весов  $\theta^{[i]}$  относится к  $i$ -му слою. Всюду далее мы будем использовать эту нотацию и под *первым* слоем понимать *первый скрытый* слой (слой с номером 1).

<sup>2</sup> Обращаем ваше внимание на то, что узел, выдающий +1, присутствует на всех слоях, кроме выходного, и он *не принимает никаких входных данных*. Это влияет на размерности матриц весовых коэффициентов. Например, если входной слой имеет 400 узлов для элементов входного вектора плюс один узел для +1, а

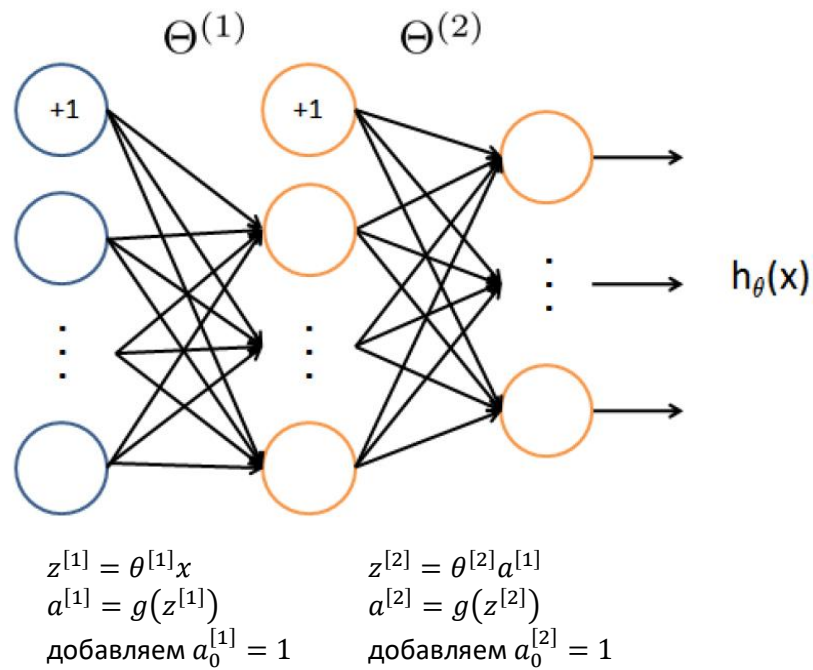


Рисунок 2: Модель нейронной сети

### 3.2 Прямое распространение и вычисление прогноза нейронной сети

Вам необходимо реализовать прямое распространение для нейронной сети. Завершите код в `predict.m`, чтобы он возвращал значение гипотезы нейронной сети  $h_{\theta}(x^{(i)})$  для каждого  $i$ -го вектора. Как и в первой части упражнения значением прогноза будет метка  $k$ , соответствующая наибольшему значению  $h_{\theta}(x)_k$ .

Когда вы закончите, `ex3_nn.m` вызовет вашу функцию `predict`, используя загруженные матрицы весов `Theta1` и `Theta2`. Вы должны увидеть точность прогнозирования около 97,5%. После этого запустится демонстрация работы сети. Чтобы ее остановить, нажмите `Ctrl-C`.

---

первый слой составлен из 25 нейронов и одного узла, выдающего +1, то матрица весов  $\theta^{[1]}$  будет иметь размерность  $25 \times 401$ .