

Распознавание цифр на основе нейронных сетей в Octave

Маринчук Александр Сергеевич

Приамурский государственный университет им. Шолом-Алейхема

Студент

Баженов Руслан Иванович

Приамурский государственный университет им. Шолом-Алейхема

к.п.н., доцент, зав. кафедрой информационных систем, математики и правовой информатики

Аннотация

В настоящее время развитие искусственных нейронных сетей получило огромное применение в современном мире: распознавание речи, прогнозирование курса валют, создание нейроигроков, распознавание рукописного текста и многое другое. В данной статье рассказывается о реализации системы распознавания цифр при помощи нейронных сетей в программной среде Octave.

Ключевые слова: Octave, нейронные сети, распознавание цифр, нейроны.

Digit recognition based on neural networks in Octave

Marinchuk Alexander Sergeevich

Sholom-Aleichem Priamursky State University

Student

Bazhenov Ruslan Ivanovich

Sholom-Aleichem Priamursky State University

Candidate of pedagogical sciences, associate professor, Head of the Department of Information Systems, Mathematics and Legal Informatics

Abstract

Currently, the development of artificial neural networks has gained immense use in the modern world: speech recognition, currency forecasting, the creation of neuroplayers, handwriting recognition and much more. This article discusses the implementation of the number recognition system using neural networks in the Octave software environment.

Keywords: Octave, neural networks, number recognition, neurons.

Искусственная нейронная сеть (ИНС) — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. Это понятие возникло при

изучении процессов, протекающих в мозге, и при попытке смоделировать эти процессы.

Наиболее часто нейронные сети применяют для распознавания какого-либо изображения или рукописного текста, которое подаётся на вход в систему и которое ИНС необходимо проанализировать и подать на выход результат.

На сегодняшний день множество людей рассмотрело тему, связанную с нейронными сетями. В работе Н. С. Костина предлагается классификация искусственных нейронных сетей в зависимости от структуры связей между нейронами в сети. Особое внимание уделяется модульным нейронным сетям, в частности сетям с ядерной организацией [2]. В статье В. Г. Манжула и Д. С. Федяшова рассмотрена актуальность использования нейронных сетей в интеллектуальном анализе данных, описан метод нейронных сетей в интеллектуальном анализе данных, рассматриваются конкретные типы нейронных сетей, используемых в интеллектуальном анализе данных, а именно нейронные сети Кохонена и нечеткие нейронные сети, выявлены достоинства и недостатки данных сетей [3]. Р. М. Немков рассмотрел сверточную нейронную сеть с динамическими рецептивными полями в качестве средства против взлома нейронных сетей через такое изменение входных изображений, при котором искажённые изображения неотличимы для человеческого восприятия от искомым, но, тем не менее, сетью распознаются неправильно [4]. В своей статье И. С. Дрокин использовал механизм предобучения многослойных персептронов, что позволило значительно улучшить качество и скорость обучения глубоких сетей [5]. Н. И. Червяков и Э. Е. Тихонов показали результаты анализа и сравнения методов прогнозирования с методами прогнозирования на нейронных сетях. Также они рассмотрели вопросы определения структуры и выбора типа нейронной сети для задач прогнозирования, провели сравнительный анализ радиально-базисной нейронной сети и сети типа многослойный персептрон на примере прогнозирования объема экспорта [6]. В работе Н. В. Зубричева рассматривается новый вид нейронной сети для распознавания изображений - капсульная нейронная сеть - архитектура искусственных нейронных сетей, которая предназначена для распознавания изображений [7]. К. О. Потапкин привел обзор существующих разновидностей искусственных программных нейронных сетей, рассмотрел особенности их архитектуры, возможности и способы применения, а также подробно описал работу простейшей нейронной сети - сети Хопфилда [8]. Изучена возможность применения теории нейронных сетей для прогнозирования валютных пар, восстановления динамических моделей курсов акций, найдены методы повышения точности прогнозов, реализован алгоритм динамических нейронных сетей для Forex, построены математические модели для решения задач аппроксимации и классификации в статье Д. Б. Владимировой и А. А. Кокшаровой [9]. Е. С. Михалкин и др. написали статью, где целью их работы является представление мощности хорошего качества для оценки качества электроэнергии на конкретный сетевой адрес с помощью искусственных

нейронных сетей путем метода машинного обучения [10]. В работе D. F. Specht предлагается четырехслойная нейронная сеть, которая может сопоставить любой входной шаблон с любым количеством классификаций [11]. Рассмотрели проблему обучения на примерах в многослойных линейных нейронных сетях с прямой связью, используя методы оптимизации в своей книге P. Pierre и K. Hornik [12]. В книге S. Haykin содержится полный доступ к нейронным сетям с инженерной точки зрения, рассматриваются все аспекты данной технологии [13]. В диссертации М. В. Алферова разработан оригинальный алгоритм прогнозирования значений технико-экономических показателей работы угольных предприятий на краткосрочную перспективу, основанный на технологии искусственных нейронных сетей [14]. Разработал методику извлечения знаний из эмпирических данных с использованием теории нейронных сетей в своей диссертации П. Е. Родионов [15].

Для разработки собственной системы распознавания цифр на основе нейронных сетей была выбрана программная среда Octave.

Octave – свободная система для математических вычислений, использующая совместимый с MATLAB язык высокого уровня. Представляет из себя интерактивный командный интерфейс для решения линейных и нелинейных математических задач, а также проведения других численных экспериментов. Язык Octave оперирует арифметикой вещественных и комплексных скаляров и матриц, имеет расширения для решения линейных алгебраических задач, нахождения корней систем нелинейных алгебраических уравнений, работы с полиномами, решения различных дифференциальных уравнений, интегрирования систем дифференциальных и дифференциально-алгебраических уравнений первого порядка, интегрирования функций на конечных и бесконечных интервалах [16];

Обучающий набор данных для нейросети целесообразнее взять готовый, т.к. в открытом доступе имеются базы знаний для обучения нейронных сетей. В данной статье будем использовать набор данных, состоящий из черно-белых изображений рукописных цифр формата PNG, написанных различным почерком [17]. Возьмем все изображения и переведем их в формат .mat и размер 20x20 пикселей для удобного чтения в Octave.

Далее загружаем наши изображения, создаем матрицу, определяем слои, где число 400 соответствует количеству пикселей, число 25 количеству нейронов в скрытом слое, а число 10 количеству нейронов на выходе (10 цифр). Здесь же задаем epsilon, лямбду и максимальное количество итераций, предварительно подобрав оптимальные значения.

```

clear; close all; clc;
load('images.mat');
m = size(X, 1);
layers = [400, 25, 10];
e = 0.1;
l = 0.01;
max_iterations = 3000;
image = randperm(size(X, 1));
image = image(1);
display_data(X(image, :));
[nn_params, cost] = neural_network_train(X, y, layers, l, e, max_iterations);
predictions = neural_network_predict(X, nn_params, layers);

```

Рисунок 1 – Инициализация

Функция `neural_network_train` используется для инициализации нейронной сети и ее обучения.

```

function [nn_params, cost] = neural_network_train(X, y, layers, lambda, epsilon, max_iterations)
    nn_params = nn_params_init(layers, epsilon);
    nn_params_unrolled_initial = unroll(nn_params);
    options = optimset('MaxIter', max_iterations);
    gradient_step = @(p) nn_gradient_step(p, layers, X, y, lambda);
    [nn_params_unrolled, cost] = fmincg(gradient_step, nn_params_unrolled_initial, options);
    nn_params = nn_params_roll(nn_params_unrolled, layers);
end

```

Рисунок 2 – Функция `neural_network_train`

Функция `nn_params_init` служит для нарушения симметрии параметров нейронной сети при обучении. Определяем длину слоёв, где каждый слой будет иметь собственную матрицу с входящими и исходящими соединениями.

```

function nn_params = nn_params_init(layers, epsilon)
    L = length(layers);
    nn_params = {};
    for layer_number=1:(L-1)
        L_in = layers(layer_number);
        L_out = layers(layer_number + 1);
        nn_params{layer_number} = rand(L_out, L_in + 1) * 2 * epsilon - epsilon;
    end
end

```

Рисунок 3 – Функция `nn_params_init`

Функция `unroll` нужна для преобразования матрицы для каждого слоя в один плоский вектор. Рассчитывается общее количество ячеек, инициализируется вектор ячеек и затем матрица разворачивается в один вектор.

```

function unrolled_cells = unroll(cells)
    cells_count = length(cells);
    unrolled_cells = [];
    for cell_number=1:cells_count
        unrolled_cells = [unrolled_cells; cells{cell_number}(:)];
    end
end

```

Рисунок 4 – Функция `unroll`

Функция `nn_gradient_step` предназначена для выполнения шага градиента. Здесь же рассчитывается стоимость нейронной сети для конкретных параметров модели и выполняется обратное распространение ошибки.

```
function [J gradients] = nn_gradient_step(nn_params_unrolled, layers, X, y, lambda)
    nn_params = nn_params_roll(nn_params_unrolled, layers);
    J = nn_cost_function(nn_params, layers, X, y, lambda);
    gradients = nn_backpropagation(nn_params, layers, X, y, lambda);
end
```

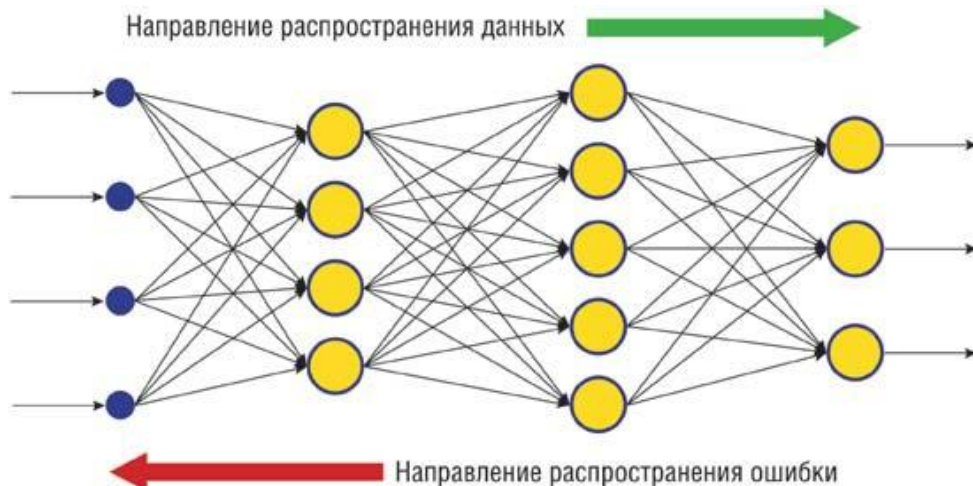
Рисунок 5 – Функция `nn_gradient_step`

Рисунок 6 – Обратное распространение ошибки

Далее следует функция `nn_params_roll`, которая преобразует плоский вектор в матрицы для каждого NN-слоя. Здесь также определяется длина слоёв и уже по входным и исходным соединениям строится матрица.

```
function nn_params_rolled = nn_params_roll(nn_params_unrolled, layers)
    L = length(layers);
    nn_params_rolled = {};
    unrolled_shift = 0;
    for layer_number=1:(L-1)
        L_in = layers(layer_number);
        L_out = layers(layer_number + 1);
        params_width = L_in + 1;
        params_height = L_out;
        params_volume = params_height * params_width;
        layer_params_unrolled = nn_params_unrolled((unrolled_shift + 1):(unrolled_shift + params_volume), 1);
        nn_params_rolled(layer_number) = reshape(layer_params_unrolled, params_height, params_width);
        unrolled_shift = unrolled_shift + params_volume;
    end
end
```

Рисунок 7 – Функция `nn_params_roll`

Функция `display_data` служит для итогового отображения цифр, которые были распознаны нейронной сетью в соответствии со сделанным запросом.

```

function [n, display_array] = display_data(X)
    example_width = round(sqrt(size(X, 2)));
    colormap(gray);
    [m n] = size(X);
    example_height = (n / example_width);
    display_rows = floor(sqrt(m));
    display_cols = ceil(m / display_rows);
    pad = 1;
    display_array = -ones(pad + display_rows * (example_height + pad), pad + display_cols * (example_width + pad));
    curr_ex = 1;
    for j = 1:display_rows
        for i = 1:display_cols
            if curr_ex > m,
                break;
            end
            max_val = max(abs(X(curr_ex, :)));

            row_shift = pad + (j - 1) * (example_height + pad) + (1:example_height);
            column_shift = pad + (i - 1) * (example_width + pad) + (1:example_width);
            display_array(row_shift, column_shift) = reshape(X(curr_ex, :), example_height, example_width) / max_val;
            curr_ex = curr_ex + 1;
        end
        if curr_ex > m,
            break;
        end
    end
    h = imagesc(display_array, [-1 1]);
    axis image off;
    drawnow;
end

```

Рисунок 8 – Функция nn_params_roll

На основе обучающих данных система учится, запоминая расположение пикселей для каждой цифры и благодаря этому может при загрузке «незнакомого» изображения, предварительно разбитого на пиксели, распознать его и определить какая перед ней цифра.

Для обучения нейронной сети потребовалось около 5 часов, достигнута точность предсказания в распознавании цифр в 75%. Для проверки системы следует загрузить изображение размером 20x20 пикселей и распознать его с помощью нейронной сети, которая определит, что за цифра подана на вход.

Загружаем изображение в функции `neural_network_predict` с помощью команды `imread` и передаём его в нейронную сеть для распознавания, где с помощью функции `display_data` на выход получаем цифру в виде изображения, которая соответствует нашему запросу.

```

function predictions = neural_network_predict(X, nn_params, layers)
    A=imread("cifral");
    m = size(X, 1);
    num_labels = size(nn_params{end}, 1);
    h = nn_feedforward_propagation(X, nn_params, layers);
    [dummy, predictions] = max(h, [], 2);
end

```

Рисунок 9 – Функция neural_network_predict

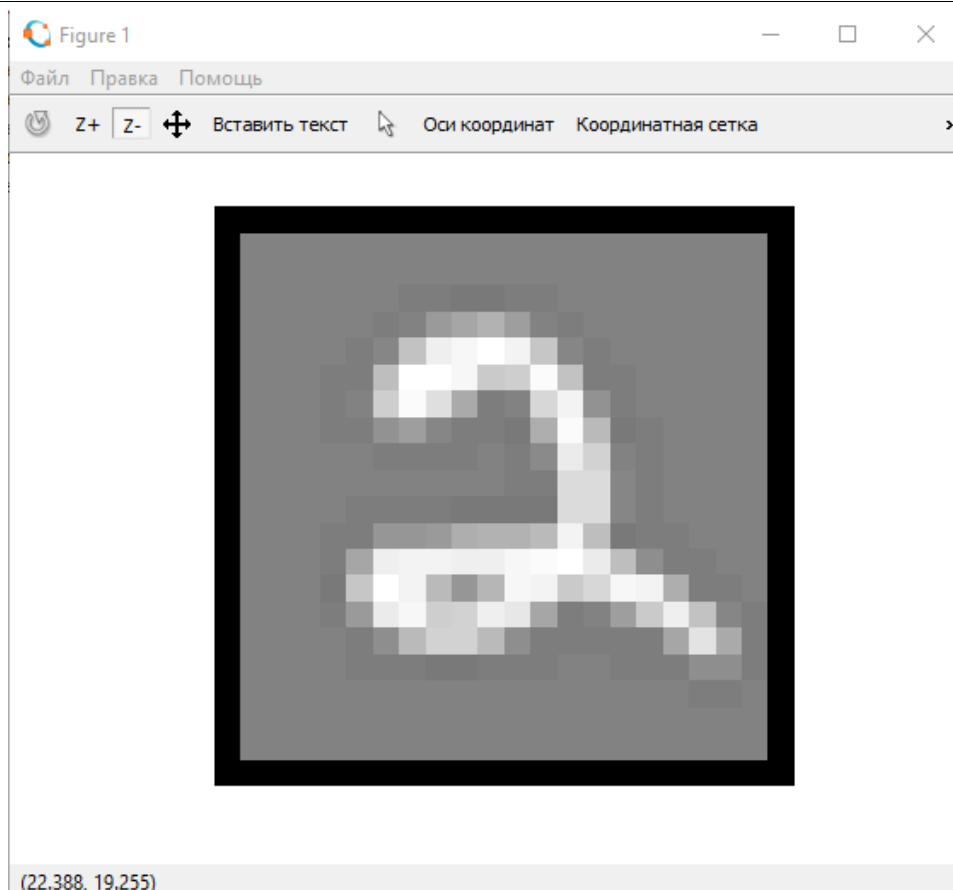


Рисунок 10 – Пример выходного изображения

В современном мире IT-технологии активно внедряются в повседневную жизнь. Уже трудно представить жизнь без смартфонов, компьютеров и различного рода ПО, ориентированного на различные виды работ. В данной сфере успешно развиваются и нейронные сети, которые получили свое применение в распознавание лиц, речи, текстов, составление различного рода прогнозов и многое другое.

В результате нашего исследования была реализована система распознавания цифр с точностью 75%, которая может быть применима для распознавания рукописных документов, содержащих большие объемы вычислений, с последующим их переводом в электронный формат.

Библиографический список

1. Ибрафиров Х. С. Применение нейронных сетей в распознавании рукописного текста // Молодой ученый. 2016. №29. С. 24-27. URL <https://moluch.ru/archive/133/37372/> (дата обращения: 11.12.2018).
2. Костин Н.С. Место модульных нейронных сетей в классификации искусственных нейронных сетей // Интеллектуальный потенциал XXI века: ступени познания. 2013. №19.
3. Манжула В.Г., Федяшов Д.С. Нейронные сети Кохонена и нечеткие нейронные сети в интеллектуальном анализе данных // Фундаментальные исследования. 2011. №4.

4. Немков Р.М. Сверточная нейронная сеть с динамическими рецептивными полями как средство против взлома нейронных сетей низко зашумлёнными паттернами // Информационные системы и технологии. Курск: Закрытое акционерное общество "Университетская книга", 2018.
5. Дрокин И. С. Об одном алгоритме последовательной инициализации весов глубоких нейронных сетей и обучении ансамбля нейронных сетей // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2016. №4.
6. Червяков Н.И., Тихонов Э.Е. Применение нейронных сетей для задач прогнозирования и проблемы идентификации моделей прогнозирования на нейронных сетях // Нейрокомпьютеры: разработка, применение. 2003. №10-11.
7. Зубричев Н. В. Новый вид нейронной сети для распознавания изображений - капсульная нейронная сеть // Современные технологии: актуальные вопросы, достижения и инновации. Пенза: Наука и просвещение, 2017.
8. Потапкин К.О. Искусственные нейронные сети. Нейронная сеть Хопфилда // XLVI Огарёвские чтения. Саранск: Национальный исследовательский мордовский государственный университет им. Н.П. Огарёва, 2018.
9. Владимирова Д.Б., Кокшарова А.А. Прогнозирование финансовых рынков искусственными нейронными сетями // Наука и бизнес: пути развития. 2014. №3.
10. Михалин Е.С., Чипко А.Г., Липкин С.М. Анализ общего качества электроэнергии для электрических сетей используя нейронные сети // Наука сегодня: история и современность. Новосибирск: ООО "Маркер", 2016.
11. Specht D. F. Probabilistic neural networks // Neural networks. 1990. Т. 3. №. 1. С. 109-118.
12. Baldi P., Hornik K. Neural networks and principal component analysis: Learning from examples without local minima // Neural networks. 1989. Т. 2. №. 1. С. 53-58.
13. Haykin S. Neural networks: a comprehensive foundation. Prentice Hall PTR, 1994.
14. Алферов М.В. Разработка алгоритмов комплексного анализа деятельности угольных предприятий с применением метода нейронных сетей: дис. ... канд. техн. наук: 05.13.18. Кемерово, 2004.
15. Родионов П. Е. Методика извлечения знаний в задачах анализа рядов динамики с использованием нейронных сетей: дис. ... канд. техн. наук: 05.13.17. Москва, 2003.
16. GNU Octave URL: <https://www.gnu.org/software/octave/> (дата обращения: 11.12.2018).
17. База данных рукописных цифр URL: <http://yann.lecun.com/exdb/mnist/> (дата обращения: 11.12.2018).