Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

**ОТЧЕТ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ**

Исследование многослойного персептрона

тема

Преподаватель \_\_\_\_\_\_\_\_\_\_\_ А.С. Михалев

подпись, дата инициалы, фамилия

Студент КИ15–16Б, 031510065 \_\_\_\_\_\_\_\_\_\_\_ Т.В. Радионов

номер группы, зачетной книжки подпись, дата инициалы, фамилия

Красноярск 2018

# Цель исследования

Цель данного задания исследовать характеристики многослойного персептрона с одним скрытым слоем и обучающегося с учителем методом обратного распространения ошибки путем разработки его программной модели.

# Теория

Нейронную сеть (далее – НС) можно представить в качестве взвешенного направленного графа, узлы (нейроны) которого расположены слоями. Кроме того, узел одного слоя имеет связи со всеми узлами предыдущего слоя (рисунок 1).

Каждый узел (нейрон) обладает активационной функцией — функцией, ответственной за вычисление сигнала на выходе узла (нейрона). Также существует понятие смещения, представляющего из себя узел, на выходе которого всегда появляется единица.

Выбор начальных весов окажет влияние на то, сумеет ли сеть достичь глобального (или только локального) минимума ошибки, и насколько быстро этот процесс будет происходить. Изменение весов между двумя нейронами связано с производной активационной функции нейрона из последующего слоя и активационной функции нейрона слоя предыдущего. В связи с этим, важно избегать выбора таких начальных весов, которые обнулят активационную функцию или ее производную. Также начальные веса не должны быть слишком большими (или входные сигнал для каждого скрытого или выходного нейрона скорее всего попадут в регион очень малых значений сигмоида (регион насыщения)). С другой стороны, если начальные веса будут слишком маленькими, то входной сигнал на скрытые или выходные нейроны будет близок к нулю, что также приведет к очень низкой скорости обучения. Стандартная процедура инициализации весов состоит в присвоении им случайных значений в интервале (-0,5; 0,5). Значения могут быть как положительными, так и отрицательными, так как конечные веса, получающиеся после обучения сети, могут быть обоих знаков.

В случае с большинством нейронных сетей цель состоит в обучении сети таким образом, чтобы достичь баланса между способностью сети давать верный отклик на входные данные, использовавшиеся в процессе обучения (запоминания), и способностью выдавать правильные результаты в ответ на входные данные, схожие, но неидентичные тем, что были использованы при обучении (принцип обобщения). Обучение сети методом обратного распространения ошибки включает в себя три этапа: подачу на вход данных, с последующим распространением данных в направлении выходов, вычисление и обратное распространение соответствующей ошибки и корректировку весов. После обучения предполагается лишь подача на вход сети данных и распространение их в направлении выходов. При этом, если обучение сети может являться довольно длительным процессом, то непосредственное вычисление результатов обученной сетью происходит очень быстро. Кроме того, существуют многочисленные вариации метода обратного распространения ошибки, разработанные с целью увеличения скорости протекания процесса обучения.

Также стоит отметить, что однослойная нейронная сеть существенно ограничена в том, обучению каким шаблонам входных данных она подлежит в то время, как многослойная сеть (с одним или более скрытым слоем) не имеет такого недостатка [1].

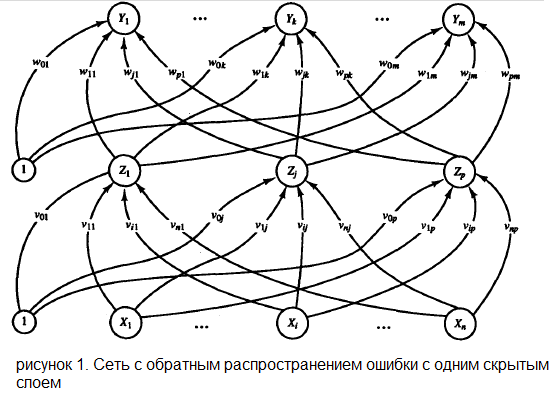


Рисунок 1 – Сеть с обратным распространением ошибки с одним скрытым слоем

В качестве активационной функции в многослойных персептронах, как правило, используется сигмоидальная активационная функция, в частности логистическая:

|  |  |
| --- | --- |
|  | (1) |

Алгоритм обратного распространения ошибки следующий:

1. Инициализировать синаптические веса маленькими случайными значениями.
2. Выбрать очередную обучающую пару из обучающего множества; подать входной вектор на вход сети.
3. Вычислить выход сети.
4. Вычислить разность между выходом сети и требуемым выходом (целевым вектором обучающей пары).
5. Подкорректировать веса сети для минимизации ошибки (как см. ниже).
6. Повторять шаги с 2 по 5 для каждого вектора обучающего множества до тех пор, пока ошибка на всем множестве не достигнет приемлемого уровня.

Рассмотрим подробней 5 шаг – корректировка весов сети. Здесь следует выделить два нижеописанных случая.

*Случай 1. Корректировка синаптических весов выходного слоя*

Например, для модели нейронной сети на рис. 1, это будут веса имеющие следующие обозначения:  и . Определимся, что индексом  будем обозначать нейрон, из которого выходит синаптический вес, а  – нейрон в который входит (рисунок 2):

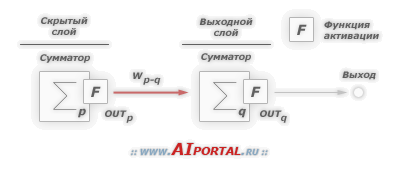


Рисунок 2 – Простая модель многослойной нейронной сети

Введем величину , которая равна разности между требуемым  и реальным  выходами, умноженной на производную логистической функции активации (формулу (1) логистической функции активации см. выше):

|  |  |
| --- | --- |
|  | (2) |

Тогда, веса выходного слоя после коррекции будут равны:

|  |  |
| --- | --- |
|  | (3) |

где:

* – номер текущей итерации обучения;
* – величина синаптического веса, соединяющего нейрон  с нейроном ;
* – коэффициент «скорости обучения», позволяет управлять средней величиной изменения весов;
* – выход нейрона .

*Случай 2. Корректировка синаптических весов скрытого слоя*

Определимся, что индексом  будем обозначать нейрон из которого выходит синаптический вес, а  – нейрон в который входит (обратите внимание на появление новой переменной ) (рисунок 3):

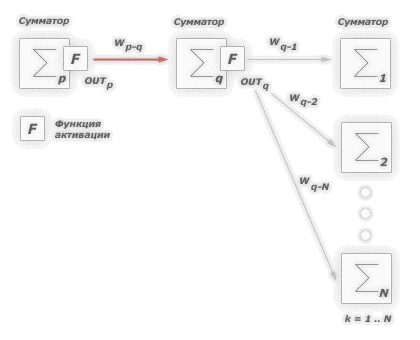


Рисунок 3 - Простая модель многослойной нейронной сети

Введем величину , которая равна:

|  |  |
| --- | --- |
|  | (4) |

где:

* – сумма от 1 по M.

Тогда, веса скрытых слоев после коррекции будут равны:

|  |  |
| --- | --- |
|  | (5) |

Для каждого нейрона в скрытом слое должно быть вычислено delta и подстроены все веса, ассоциированные с этим слоем. Этот процесс повторяется слой за слоем по направлению к входу, пока все веса не будут подкорректированы [2].

Нейрон смещения или bias нейрон — это третий вид нейронов, используемый в большинстве нейросетей (рисунок 4). Особенность этого типа нейронов заключается в том, что его вход и выход всегда равняются 1 и они никогда не имеют входных синапсов. Нейроны смещения могут, либо присутствовать в нейронной сети по одному на слое, либо полностью отсутствовать, 50/50 быть не может (красным на схеме обозначены веса и нейроны которые размещать нельзя). Соединения у нейронов смещения такие же, как у обычных нейронов — со всеми нейронами следующего уровня, за исключением того, что синапсов между двумя bias нейронами быть не может. Следовательно, их можно размещать на входном слое и всех скрытых слоях, но никак не на выходном слое, так как им попросту не с чем будет формировать связь.

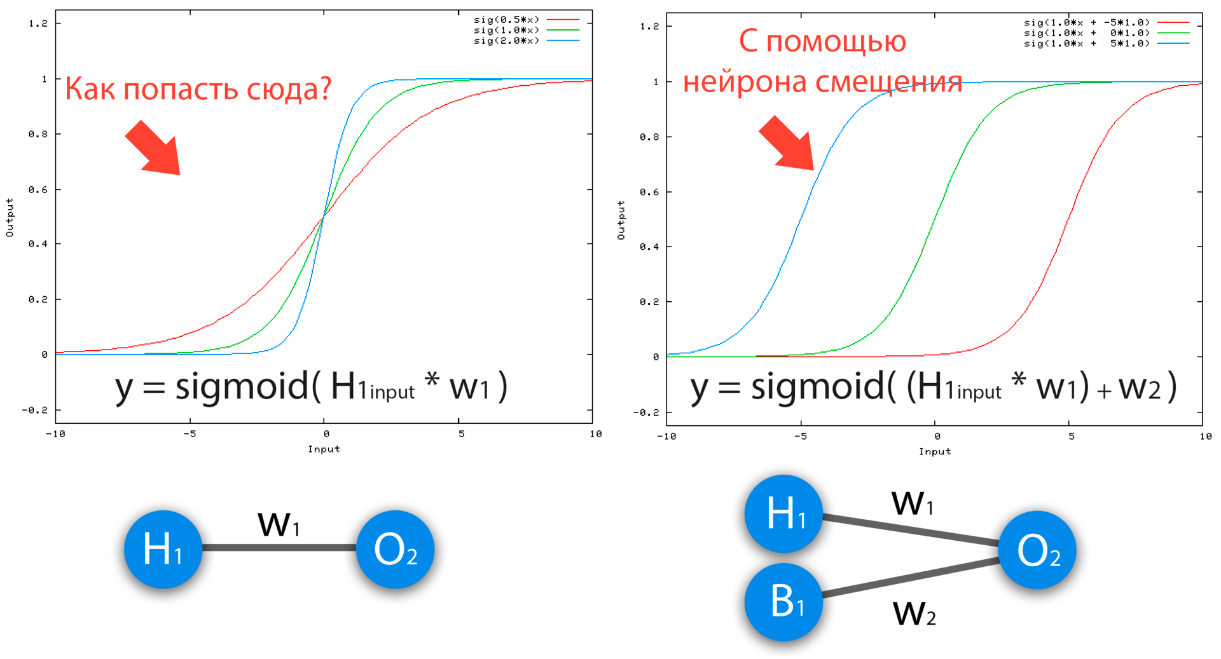


Рисунок 4 – Нейрон смещения

Сходимость говорит о том, правильная ли архитектура НС и правильно ли были подобраны гиперпараметры в соответствии с поставленной задачей (рисунок 5). Допустим наша программа выводит ошибку НС на каждой итерации в лог. Если с каждой итерацией ошибка будет уменьшаться, то мы на верном пути и наша НС сходится. Если же ошибка будет прыгать вверх — вниз или застынет на определенном уровне, то НС не сходится. В 99% случаев это решается изменением гиперпараметров. Оставшийся 1% будет означать, что у вас ошибка в архитектуре НС. Также бывает, что на сходимость влияет переобучение НС [3].



Рисунок 5 – Сходимость НС

# Описание задачи

Для выполнения данного задания взята задача по классификации цифр (от 0 до 9) и их искаженных вариантов. Многослойный персептрон представляет из себя входной слой с 15 входами, скрытый слой с 5 нейронами и выходной слой с 10 нейронами (в дальнейшем опыты будут проводится только с данным персептроном). Данные для входного слоя персептрона берутся из сенсорного поля программы в виде клеточного пространства размерностью 5 на 10.

В качестве активационной функции будет использоваться сигмоидальная функция и также не будет нейрона смещения.

Инициализация начальных весов происходит случайным образом в диапазоне [-0.5; 0.5].

Обучение будет проходит с учителем – будет взята обучающая выборка с правильными значениями цифр 0-9, на которой будет обучаться НС c заданным в программе количеством шагов и скоростью обучения по алгоритму обратного распространения ошибки, описанного в теоретической части. Затем результат обучения проверяется сперва на обучающей выборке (в случае неуспеха происходит переобучение). Основным результатом классификации программы является текстовое сообщение с числом, которое соответствует максимальному значение одного из выходов персептрона.

Также можно будет выполнить переобучение программы и посмотреть на результат обучения с помощью вычисления ошибки классификации искаженных вариантов цифр тестовой выборки (их будет 30 – по 3 варианта на каждую цифру).

С целью исследования различных характеристик будет встроен в программу график с изменяющимися параметрами: общая ошибка классификации тестовой выборки (error) – ось X, скорость обучения (eta) – ломаная на графике, количество шагов обучения (параметр N) – ось X. Ось error будет в диапазоне [0; 1], графиков eta будет 10 (от 0.1 до 1.0 с шагом 0.1), ось X будет в диапазоне от 1000 до 10000 (который можно изменить в программе). Также будет введен дополнительно параметр количества опытов M, который отвечает за количество повторений обучения с текущими значениями eta и N и выводом среднего значения error (данный параметр позволит «сгладить» график eta – чем больше M, тем меньше вероятность ошибки при вычислении параметра error).

В результате можно будет испытывать классификатор на самостоятельно введенных входных данных, оценивать текущий результат обучения и выполнять переобучение, а также проводить исследование персептрона путем изучения графиков и определения наиболее оптимальных параметров для программы с целью улучшить классификатор.

# Ход работы

В ходе исследований было установлено, что сходимость персептрона (НС) наблюдается при параметре N в диапазоне [1000; 10000] при некоторых значениях eta.

Для первого опыта был взят диапазон N [1000; 10000] c шагом 500 при M = 100. В результате сеть произвела переобучение 18000 раз и затратила на это время около 35 минут. Как видно из рисунка 6, персептрон сходится при eta = 0,2; 0,3; 0,4. Наименьшая ошибка сети наблюдается при eta = 0,4 и N = 9000.

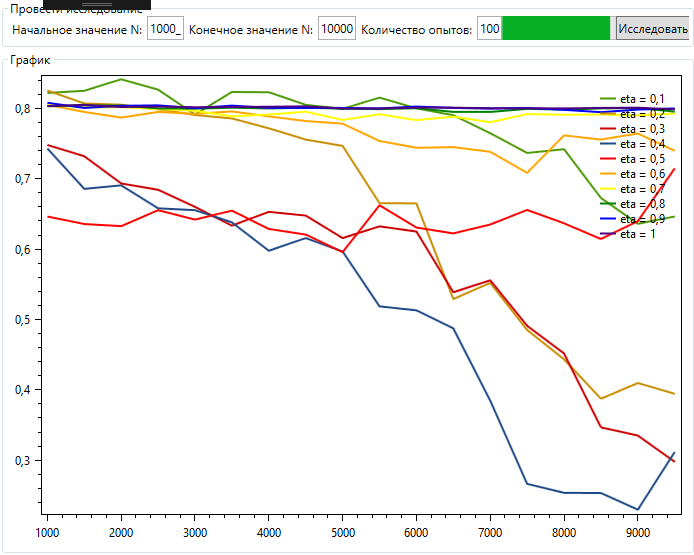


Рисунок 6 – Опыт 1

Второй опыт был проведен при тех же условиях, что и первый с целью изучить то, будет ли результат схож или отличен. Время и количество переобучений сети аналогично первому опыту. Как видно из рисунка 7, персептрон сходится при eta = 0,2; 0,3. Наименьшая ошибка сети наблюдается при eta = 0,3 и N = 8500.

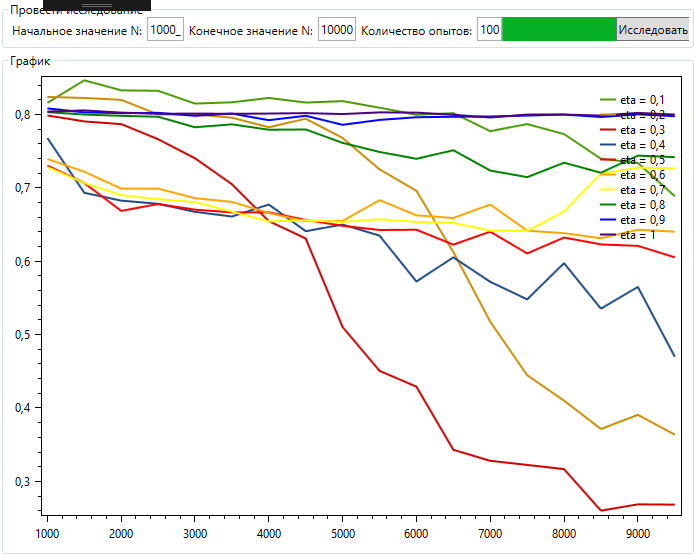


Рисунок 7 – Опыт 2

Третий опыт был взят аналогичный предыдущим диапазон N [1000; 10000] c меньшим шагом 200 при M = 100 (для большего количества точек на графике). В результате сеть произвела переобучение 45000 раз и затратила на это время примерно 85 минут. Как видно из рисунка 8, персептрон сходится при eta = 0,2; 0,3. Наименьшая ошибка сети наблюдается при eta = 0,3 и N = 9000.

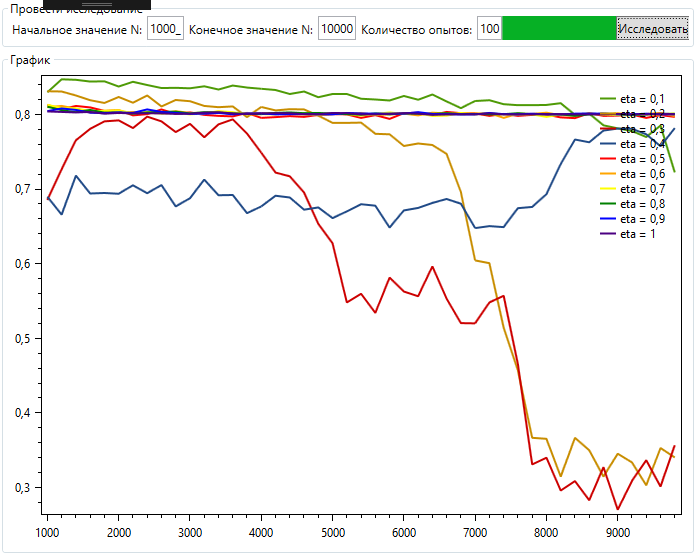
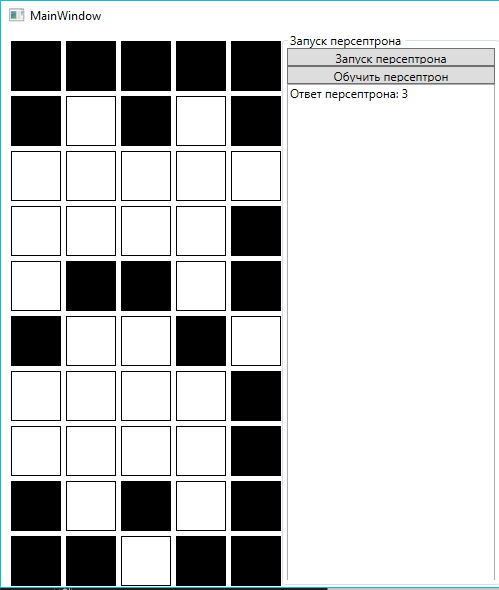
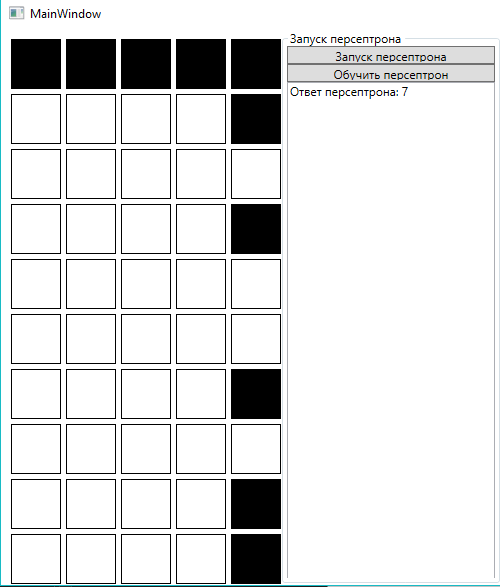


Рисунок 8 – Опыт 3

Также проводилось опыты с отдельно вручную вводимыми числами, с которыми можно ознакомиться на рисунке 9. Примечательно, что не всегда сеть отвечает так, как хочет наблюдающий: это связано с особенностью мышления человека и работы алгоритма. Например, числа «3» и «8» похожи и мы можем искаженный вариант этих чисел отнести сразу к двум классам, однако персептрон отдает предпочтение только одному числу согласно алгоритму. Также «лишний» пиксель в центре «0» у правильного числа может не повлиять на решение нейросети отдать предпочтение числу «8», так как это может быть недостаточно для преодоления порога сумматора выходного нейрона числа «8», а для человека кажется очевидным, что это уже «8».



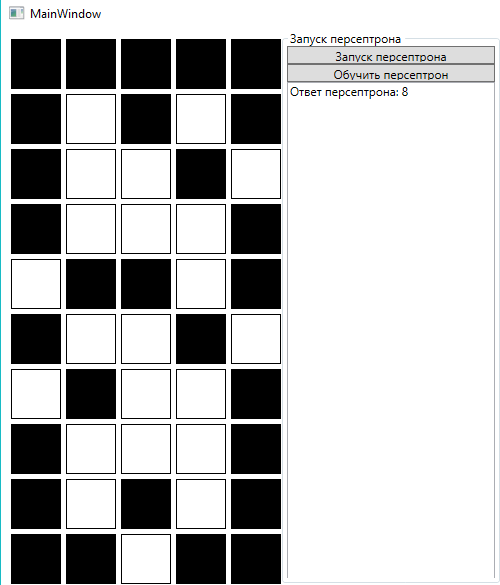
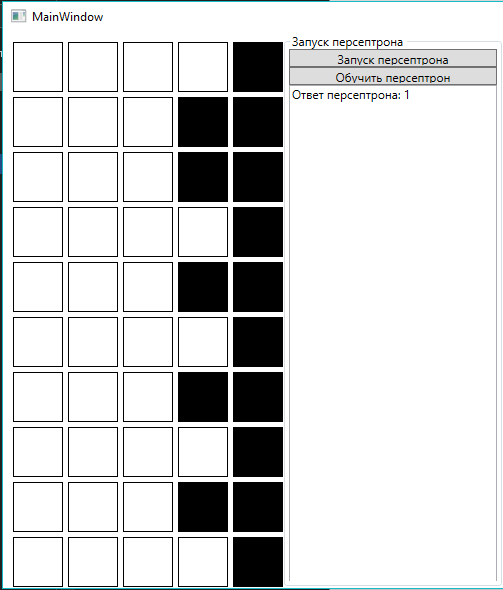


Рисунок 9 – Опыт с вручную вводимыми числами

# Выводы

В ходе исследования были установлены оптимальные параметры для персептрона с поставленной задачей: 0.2 < eta < 0.4 и 8000 < N < 9000. С данными параметрами удалось добиться хороших результатов на тестовой выборке (рисунок 10):

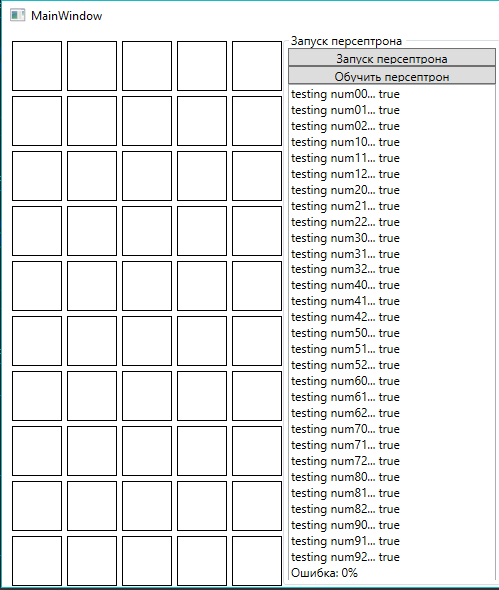


Рисунок 10 – Результат на тестовой выборке

Примечательно, что в опыте 1 eta = 0.4 был оптимальный, однако в остальных опытах при таком значении персептрон не сходился: объяснение этому особенность случайного обучения нейросетей, что наталкивает на вывод, что нет постоянного оптимального параметра, однако можно выделить диапазон, где высока вероятность подобрать его. Такой подход можно использовать в других нейронных сетях для определения оптимальных параметров под конкретные задачи, но разработанная программа не универсальна и не подходит для исследования иных задач, однако ее можно сделать таковой, если добавить возможность изменять количество входов и выходов и количество нейронов скрытого слоя (можно и задавать несколько скрытых слоев).

Также единичное обучение нейросети не занимает более чем 115 миллисекунд, однако при исследовании из-за большого количества циклов обучения это время существенно увеличилось.

Необходимо отметить, что во многом точность исследования нейросетей зависит от обучающей и тестовой выборок: чем больше, тем лучше.

# Список использованных источников

x

|  |  |
| --- | --- |
| 1. | Алгоритм обучения многослойной нейронной сети методом обратного распространения ошибки (Backpropagation) [Электронный ресурс] // habr: [сайт]. [2013]. URL: https:/​/​habr.com/​post/​198268/ (дата обращения: 25.04.2018). |
| 2. | Алгоритм обратного распространения ошибки [Электронный ресурс] // Портал искусственного интеллекта: [сайт]. URL: http:/​/​www.aiportal.ru/​articles/​neural-networks/​back-propagation.html (дата обращения: 25.04.2018). |
| 3. | Нейронные сети для начинающих. Часть 2 [Электронный ресурс] // habr: [сайт]. [2017]. URL: https:/​/​habr.com/​post/​313216/ (дата обращения: 25.04.2018). |

x