МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«кубанский государственный университет»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра прикладной математики**

**курсовая работа**

**НЕЙРОННЫЕ СЕТИ В ЗАДАЧАХ ПРОГНОЗИРОВАНИЯ**

Работу выполнила \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Л.С. Дуплеева(подпись)

Направление подготовки 01.03.02 Прикладная математика и информатика

Направленность (профиль) Системный анализ, исследование операций и управление (Математическое и информационное обеспечение экономической деятельности)

Научный руководитель

канд. физ.-мат. наук, доц. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А. В. Письменский  
(подпись)

Нормоконтролер

канд. физ.-мат. наук, доц. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Г.В. Калайдина

(подпись)

Краснодар

2018

**РЕФЕРАТ**

Курсовая работа содержит 30 страниц, 5 рисунков, 16 источников.

ИСКУСТВЕННЫЕ НЕЙРОННЫЕ СЕТИ, НЕЙРОН, ПЕРСЕПТРОН, ПРОГНОЗИРОВАНИЕ, ОБУЧЕНИЕ, ДАННЫЕ, ТЕСТИРОВАНИЕ, АЛГОРИТМЫ, МАТЕМАТИЧЕСКАЯ МОДЕЛЬ, ПРЕДСКАЗАНИЕ

Объектом исследования являются прогнозирующие искусственные нейронные сети и методы работы с ними.

Целью работыявляется ознакомление с материалами про искусственные нейронные сети, что в дальнейшем послужит разработке программного обеспечения в задачах прогнозирования.

В результате были рассмотрены теоретические основы, методы обучения и области применения нейронных сетей. А также проведены тестовые расчеты прогнозирования значений некоторых функций средствами нейросетевого пакета Neural Network Wizard и сделаны выводы о том, какие данные являются наиболее подходящими для более точных результатов для многослойной нейронной сети, в т.ч. подробно рассмотрен метод обратного распространения ошибки, которые используют для обучения таких нейросетей.

# СОДЕРЖАНИЕ

Введение 4

1 Формальный нейрон 6

2 Персептрон 8

2.1 Основные понятия .8

2.2 Классификация персептронов 9

2.3 Правила обучения персептронов 10

3 Применение нейронных сетей 14

4 Реализация нейронных сетей 16

4.1 Общая информация 16

4.2 Выбор средств реализации работы нейронной сети 18

4.3 Пример реализации средствами Neural Network Wizard 19

Заключение 28

Список использованных источников 29

**ВВЕДЕНИЕ**

Ещё с давних времен человечество пыталась найти способ облегчить свою жизнь и автоматизировать некоторые задачи с помощью специализированных устройств. Но если ранее делался упор на приспособлениях, способных выполнять физическую работу, то в современном мире люди стремятся к тому, чтобы создавать машины, которые смогут выполнять действия, идентичные принципам работы человеческого мозга.

Важность данного вопроса возрастала с темпом развития науки. Освоение космоса, робототехника, кибернетика, данные сферы не дают человечеству останавливаться в развитии и подталкивают к изучению новых методов, способных решать новые появляющиеся, сложные задачи. Несмотря на то, что количество скептиков, считающих, что некоторые вопросы неразрешимы с помощью науки, не убавляется, задача создания искусственного интеллекта остается всё также востребованной.

Одним из направлений искусственного интеллекта стали искусственные нейронные сети (ИНС), которые в дальнейшем будем называть просто нейронными сетями или же нейросетями. Они представляют собой математическую модель, работающую по принципу действия биологических сетей нервных клеток животного организма, состоящих из нейронов. Исследование в данной в области связано с тем, что обработка данных человеческим мозгом в корне отличается от того, как работают современные ПК. Если сопоставить человеческий мозг с машиной, то можно сказать, что это сложный, нелинейный компьютер, работающий с параллельными вычислениями. Также он умеет организовать работу нейронов так быстро, как ни способен ни один компьютер в мире. Конечно, ИНС – это довольно упрощенная модель мозга, но она способна такие решать задачи, выполнимость которых невозможно обычными алгоритмами. Ключевая особенность нейросетей заключается в том, что они не программируются, как обычное программное обеспечение, а обучаются.

Области, в которых активно применяются нейронные сети, довольно обширны. Например, распознавание образов, медицина, экономика и т.д. Но также существует область, проработанность которой всё ещё не высока – это область прогнозирования, несмотря на то что эта сфера крайне важна для безопасной жизнедеятельности человека. К примеру, если идет речь о прогнозировании природных катаклизмов.

Актуальностьданной работы обусловлена тем, что нейронные сети на данный момент являются самым прогрессивным программным обеспечением, которое наиболее успешно справляется с задачей прогнозирования.

**1 Формальный нейрон**

Первой работой в истории, на которой были положены теоретические основы искусственного нейрона считается статья, опубликованная в 1943 году Уоррена Мак-Каллока и Вальтера Питтса, где как раз и описывается логический элемент, который способен на деятельность, схожую с деятельностью нейрона человеческого мозга. Этот упрощенный элемент получил название формального нейрона и также является главным функционирующим элементом нейронной сети.

Схематично его строение можно представить так (рис.1):

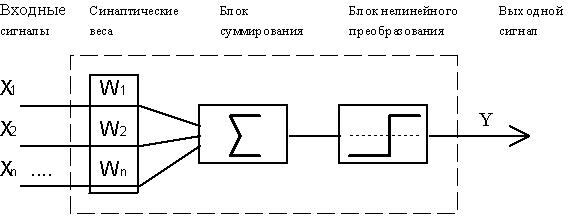


Рисунок 1 – Схема формального нейрона

– это компоненты вектора входных сигналов, которые поступают в нейрон. В дальнейшем, проходя через три функциональных блока, они преобразуются в выходной сигнал Y.

Рассмотрим подробнее эти блоки.

Первым является блок локальной памяти, в котором хранится информация о весовых множителях - значение весов входных сигналов нейрона. Затем идет блок суммирования, где функцией сумматора накапливается общий сигнал, исходящий от входов. Функция сумматора выглядит так *,* где - компоненты вектора входных сигналов. Значение этой функции определяет есть ли реакция нейрона на внешний фактор. И наконец, блок нелинейного преобразования. В локальной памяти также хранится вес порога Q и если значение сумматора превысило значение порога, то выходу нейрона присваивается значение единицы, если наоборот, то нуля. Это и называется нелинейным пороговым преобразованием [15].

**2 Персептрон**

**2.1 Основные понятия**

Персептрон – это компьютерная модель восприятия (perception от лат. восприятие) информации человеческим мозгом. Как известно, реализация данной идеи в 1957 году принадлежит Фрэнку Розенблатту, который позже в 1958-1960х годах создал первый в мире нейрокомпьютер «Марк-1».

Система персептрона довольна проста, но несмотря на это она способна решать довольно тяжелые математические задачи. Она состоит из трех разновидностей элементов и описывается таким образом: на вход поступает сигнал от S-элементов сенсоров, который передается в ассоциативные A-элементы и затем поступает в реагирующие R-элементы. Рассмотрим подробнее работу персептрона [6].

Первым слоем элементов являются сенсоры, биологически они выполняют функцию рецептора и могут иметь только два состояния: активное и не активное. Сравнить их можно с булевским типом в языках программирования, т.е. от них передается сигнал true (1), если сенсор среагировал и false (0) в противном случае. Далее сигнал поступает в ряд ассоциативных элементов, которые реагируют только в том случае если количество сигналов от рецептора превышают определенную так называемую пороговую величину. И наконец, от A-элементов сигнал поступает в R-элементы, выполняющие роль сумматора, но при этом здесь появляется еще одно понятие – веса, которое определяется значением коэффициента от предшествующих А-элементов. Реагирующие элементы подсчитывают сумму входных сигналов, помноженную на веса и если получившиеся значение превысило порог, т.е. персептрон смог что-либо распознать, то на выходе значение 1, а в противном случае –1.

**2.2 Классификация персептронов**

Рассмотрим различные типы персептронов.

1. Однослойный персептрон

Однослойный персептрон является моделью, в которой входные элементы соединены с выходными элементами с помощью системы весов. Каждый S-элемент поставлен в однозначное соответствие с A-элементом, их связи всегда равны единице. Также пороговое значение А-элемента не превосходит единицы. Фактически однослойные персептроны являются формальными нейронами.

Следует отметить, что у них имеются множество ограничений, в частности то, что они не могут идентифицировать ситуацию, в которой на их вход подаются различные сигналы, примером этого может послужить задача XOR. Она заключается в том, что дана логическая функция двух аргументов, которая принимает значение «истина», когда один из аргументов равен 0, а другой 1, и, соответственно, значение «ложь», когда они совпадают. Однослойный персептрон не способен решить данную задачу.

1. Многослойный персептрон

Особенностью этого типа персептрона является то, что у него есть два вида:

а) по Розенблатту;

Этот персептрон содержит более 1-го слоя А-элементов

б) по Румельхату.

Этот вид является частным случаем многослойного персептрона по Розенблатту поскольку на него накладываются дополнительные условия:

– обучение происходит по методу обратного распространения ошибки

– S-A связи имеют произвольные веса, и проходят обучение также, как

и A-R связи

1. Персептрон с одним скрытым слоем

Персептрон с одним скрытым слоем является простейшим из выше представленных типов, поскольку у него есть один слой A, S, R элементов, свое название он получил благодаря тому, что слой А-элементов расположен между слоями S и R.

**2.3 Правила обучения персептронов**

Рассмотрим различные правила обучения нейросетей.

1) Обучение с учителем

Алгоритм относится к данному типу, потому что при обучении известны входные и выходные вектора сети. Сеть учится давать результаты, которые уже известны заранее. Новые результаты могут быть получены вследствие того, что подается вектор, не встречающийся при обучении.

Примером может послужить разделение класса объектов так, чтобы при выходе персептрона объекты класса А давали положительный результат, а объекты класса В – отрицательный.

Для начала выбираем произвольные пороги А-элементов и устанавливаем S-A связи, которые не будут меняться в течение обучения, затем определяем начальные коэффициенты и предъявляем обучающую выборку, т.е. когда на вход подается объект, принадлежащий к классу А, происходит реакция некоторых А-элементов, а соответствующие им коэффициенты мы инкрементируем, в тоже время при подаче на вход объектов класса В при реагировании тех же элементов мы значения соответствующих им элементов декрементируем. Таким образом прогоняем всю обучающую выборку и в результате получаем значение весов связей .

2) Обучение без учителя

Алгоритм относится к данному типу, потому что при обучении известны только входные вектора, а на выходе сеть учится давать «наилучшие» результаты, какие результаты относятся к данной категории определяется алгоритмом.

Рассмотрим метод, организованный по принципу градиентного спуска по поверхности ошибки, называемый дельта-правилом.

Пусть у нас имеется n нейронов, составляющих персептрон. Вектор входных сигналов обозначим также у нас имеется вектор, значения сигналов которого должны быть получены от персептрона под воздействием входного вектора и вектор выходных значений персептрона , значения которого получается взвешиванием и суммированием входного вектора. Исходя из вышеперечисленного можно составить вектор ошибки , вычисляемый по формуле .

Для того чтобы корректировать j-вес i-нейрона расчёты проводятся по формуле , где t - номер текущей итерации обучения.

Также часто вводят коэффициент пропорциональности ƞ называемый скоростью или нормой обучения. В формуле он умножается на величину ошибки [12].

1. Метод обратного распространения ошибки

Метод обратного распространения ошибки является одним из самых популярных методов обучения многослойного персептрона.

Для начала пронумеруем все узлы, включая входы и выходы персептрона. Обозначим через вес, соединяющий i-й и j-й узлы. Чтобы изменять веса в течение обучения сети, т.е. после каждой обучающей выборки значение весов должно меняться, мы будем использовать стохастический градиентный спуск. Таким образом происходит движение в так называемом пространстве весов и для того, чтобы минимизировать значение ошибки мы должны двигаться в сторону противоположную градиенту. Вес увеличивается, когда в нейросеть поступает каждая группа правильных ответов.

, где – это коэффициент, задающий скорость движения  
 ( )

Также у нас имеется функция ошибки , где k принадлежит выходам из сети, – это правильные ответы сети, а – это выходы k-узла.

Если j принадлежит выходам сети и интересующий вес входит в последний уровень ИНС, то функция сумматора выглядит так: исходя из этого можем найти производную .

Сумматор влияет на ошибку только при выходе , поэтому

, где f(s) – это функция активации.

Существуют множество видов функций активации, но наиболее часто используемые сигмоидальные функции:

1) Экспоненциальная функция (функция Ферми)

2) Рациональная сигмоида

3) Гиперболический тангенс

В каждом из которых s – это выход сумматора нейронной сети, а – произвольная константа.

Взяв за функцию активацию функцию Ферми и проделав некоторые математические вычисления получим: . Если у j-го узла есть выходы, т.е. он находится не на последнем уровне, то, обозначив их Children, выполняется , где k Children(j) и . Также введем обозначение .

Весь рассмотренный метод можно привести к следующему алгоритму:

1) Инициализация случайными, достаточно малыми значениями

2) Выполнить в цикле до n для всех d от 1 до m:

– подать данные на вход и подсчитать значения выходов узлов ;

– вычислить для всех k принадлежащим выходам;

– для каждого уровня, начиная с предпоследнего, вычислить значения j узлов , где k Children;

– для каждого ребра сети выполнить:

В итоге получаем:

.

1. **Применение нейронных сетей**

Искусственные нейронные сети относятся к технологиям машинного обучения и спектр задач, решаемых ими значительно широк. Соответственно, в него входят классические задачи машинного обучения такие как: классификация, регрессия, кластеризация, но если говорить обобщённо, то они применяются там, где обычные алгоритмы являются неэффективными или же попросту невозможными. Основными задачами нейронных сетей являются распознавание образов, прогнозирование, принятие решений, оптимизация и анализ данных.

ИНС как направление переживало как взлеты, так и падения. В настоящее время они вновь обрели свою актуальность. Ведутся многочисленные разработки в этой области такими крупными компаниями, как Microsoft, Google и Яндекс. Возникает резонный вопрос, почему именно сейчас? Дело в том, что современные видеокарты позволяют обучать сеть намного быстрее, чем это было в прошлом, также свою роль сыграли технологии ускоренного обучения и предобученные, готовые нейросети, благодаря которым значительно уменьшается время подготовки для работы.

Рассмотрим более подробно возможности нейросетей в нескольких актуальных областях современного развития технологий машинного обучения.

В области распознавания образов искусственные нейронные сети за небольшой промежуток времени сделали большой шаг вперед. То, что ранее казалось невозможным или реализуемым только в отдаленном будущем сейчас активно используется на рынке. Так, например, они научились не только просто распознавать, что находится на картинке, но и учитывать их семантику, т.е. смысл, тем самым описывающее происходящее. К примеру, на картинке изображена не просто «девушка», а «девушка в красной блузе гладит кошку».

При развитии технологий искусственного интеллекта были многочисленные скептики утверждающее, что процесс «создания чего-то нового» может быть присущ только человеку, но и в этой сфере нейросети ждал успех. Они довольно удачно создают свои музыкальные произведения, пишут картины и создают новые тексты.

Также нейросети широко используют для задач прогнозирования.

Прогнозирование – это предсказание будущих событий с целью минимизации риска принятых решений. Наиболее частым в использовании являются задачи в области экономики и финансов, например:

– прогноз курса ценных бумаг;

– предсказание кризиса;

– прогноз развития кризисной ситуации и многие другие [1].

Но и немало примеров предсказания событий в других сферах таких как метеорология (прогноз погоды), политика (предсказание динамики рейтингов или результатов выборов) и т.д.

**4** **Реализация нейронных сетей**

**4.1 Общая информация**

Для реализации искусственной нейронной сети используются два подхода:

1) Программный

2) Аппаратный

В дальнейшем рассмотрим каждый из них.

Программный подход безусловно имеет ряд преимуществ такие как простота в использовании, а также легкость внедрения в различные системы.

Для создания нейросетевой модели используется пятишаговый процесс:

а) определение какими должны быть входные данные и какую задачу решает нейросетевая модель;

б) поиск и сбор информации, оснащенную примерами, а также решение о представлении данных;

в) выбор архитектуры и типа нейронной сети;

г) обучение;

д) тестирование нейросети.

В связи с направлением курсовой работы рассмотрим подробно алгоритм реализации нейронной сети для задач прогнозирования.

При построении прогнозирующей системы необходимо определить три параметра: период прогнозирования (единица времени на которую выполняется прогноз), интервал прогнозирования (частота выполнения нового прогноза) и горизонт прогнозирования (число периодов, покрытых прогнозом). При этом горизонт должен быть равен или больше времени, необходимого для реализации решения, чтобы сам прогноз имел смысл. Также следует учитывать и то, что с увеличением горизонта снижается точность прогноза.

Одним из самых ключевых аспектов в задаче прогнозирования является обучающая выборка.

В настоящее время для решения задачи прогнозирования применяется метод аппроксимации функции.

Несмотря на простоту и популярность использования программной реализации ИНС у аппаратной версии есть также свои преимущества такие как:

а) скорость;

б) стоимость;

Аппаратные средства намного менее зависимы от возможностей ЦП и при этом, как уже отмечалось выше они сохраняют высокие показатели скорости.

в) надежность;

Меньшая вероятность отказа оборудования.

г) безопасность.

Лучшая защита от противоправных действий со стороны внешнего воздействия.

Существует множество типов аппаратного обеспечения на основе ИНС, однако, их можно разделить на три класса:

1. Нейрокомпьютеры

Они представляют собой систему, у которой аппаратные составляющие полностью реализованы на нейронных сетях. Такие системы востребованы, когда требуется высокая вычислительная мощность.

1. Чипы

Они предназначены для совместного использования с другими устройства для того, чтобы повысить производительность последних.

1. Клеточные библиотеки

Применяются для построения сложных систем и обеспечивают совместную работу чипов, дополнительных возможностей или функций и т.д.

**4.2 Выбор средств реализации работы нейронной сети**

Для наглядности реализации задач прогнозирования и набора тестовых данных для нейросети проведем сравнительную характеристику различных нейросетевых пакетов.

Будем сравнивать характеристики четырех нейросетевых пакетов в таблице 1. Первым является Neural Network Wizard v1.7 – это пакет, являющийся программной реализацией многослойной нейронной сети по методу обратного распространения ошибки. Вторым **STATISTICA** Neural Networks – это программный пакет для статистического анализа, реализующий функции анализа данных, управления данными, добычи данных, визуализации данных с привлечением статистических методов. Третьим NeuroSolutions – универсальный нейропакет, предназначенный для моделирования широкого круга искусственных нейронных сетей и четвертым NeuroShell 2, который является пакетом, анализующим нейросетевые данные.

Таблица 1 – Сравнительная характеристика средств реализации работы нейросети

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии | Neural Network Wizard | STATISTICA Neural Networks | NeuroSolutions | NeuroShell 2 |
| Возможность бесплатного использования | + | - | - | + |
| Интерфейс пользователя | + | + | + | - |
| Доступность для скачивания | + | - | + | - |
| Визуализация данных | - | + | + | + |
| Изменение структуры и/или критериев обучения нейросети | + | + | - | - |
| Интеграция внешних программных модулей | - | - | + | + |
| Предварительная обработка данных | + | + | - | + |

В связи с проведенным анализом различных характеристик нейропакетов можно сделать вывод о том, что самым оптимальным из них является Neural Network Wizard, который в дальнейшем будем использовать для прогнозирования различных тестовых данных.

**4.3 Пример реализации средствами Neural Network Wizard**

Для того чтобы работать с данной системой необходимо выполнить следующие этапы:

а) собрать статистические данные по процессу;

б) обучить нейросеть на приведенных данных;

в) протестировать полученные результаты.

Мы подберем 4 различных примера функций, по которым будем обучать сеть, прогнозом будут являться значения функций, аргументы которых мы не будем задавать в обучающей выборке, но по которым будем тестировать нейросеть.

В качестве первого примера возьмем периодическую функцию  
 . Прежде чем запускать программу необходимо подготовить текстовый файл с обучающей выборкой, в таблице 2 представлен фрагмент содержания нашего файла в формате \*.txt. Значениями столбца arg являются аргументы функции , а значение столбца res, соответственно, являются значения функции в данной точке .

Таблица 2 – Фрагмент содержание файла с обучающей выборкой для

|  |  |
| --- | --- |
| arg | res |
| 0,50 | 0,479425539 |
| 0,51 | 0,488177247 |
| 0,52 | 0,496880138 |
| 0,53 | 0,505533341 |
| 0,54 | 0,514135992 |

Продолжение таблицы 2

|  |  |
| --- | --- |
| 0,55 | 0,522687229 |
| 0,56 | 0,531186198 |
| 0,57 | 0,539632049 |
| 0,58 | 0,548023937 |
| 0,59 | 0,556361023 |

На рисунке 2 представлены структура и параметры нейросети: количество скрытых слоев, количество нейронов в каждом слое, а также вид сигмоидальной функции.

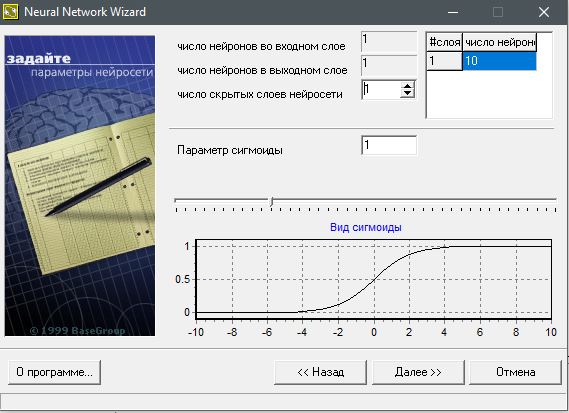


Рисунок 2 – Структура и параметры нейросети в Neural Network Wizard

Мы задали один скрытый слой, в котором 10 нейронов. Считается, что чем более нелинейная задача, тем больше скрытых слоев требуется указывать. Также не существует определенных правил для того, чтобы задавать количество нейронов, но полагают, что количество примеров в обучающей выборке должно быть значительно больше, чем количество нейронов, поскольку иначе сеть может «переобучиться», т.е. не будет обобщать данные, а просто «запомнит» их.

Параметр сигмоиды применяется для преобразования нелинейных данных. Чем выше её параметр, тем больше функция похожа на пороговую. Фактически, этот параметр определяется эмпирически, поэтому мы оставим его без изменений.

В следующем окне (см. рисунок 3) задаются параметры обучения и критерии остановки обучения, если они необходимы.

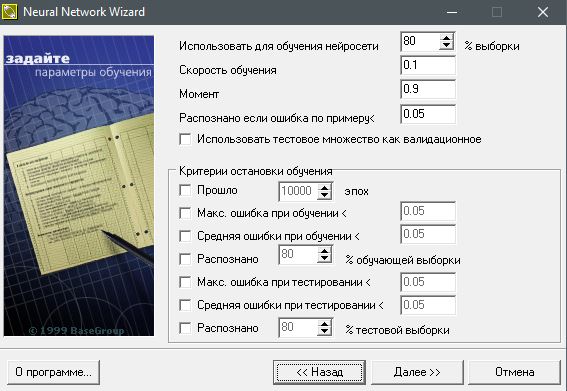


Рисунок 3 – Параметры обучения нейросети в Neural Network Wizard

Рассмотрим все параметры, которые используются для обучения нейросети.

Для обучения мы используем 80% выборки, которое выбирается из файла произвольно, остальная же часть (20%) идет на тестирование нейросети в приложении. Параметр скорости обучения определяет амплитуду коррекцию весов на каждом шаге, моментом является степень воздействия i-ой коррекции веса на i+1-ю. Если результат прогнозирования отличается от значения множества меньше указанной величины в параметре распознано если ошибка по примеру, то пример считается распознанным. Также мы поставили флаг на использовать тестовое множество как валидационное. Это необходимо для того, чтобы предотвратить переобучения нейросети, когда значение ошибки на тестовом множестве начнет увеличиваться. Ещё можно использовать критерии остановки обучения нейросети, если одно из представленных ниже параметров нас устраивает для того, чтобы получить результаты.

На рисунке 4 можно увидеть сам процесс обучения нейросети.

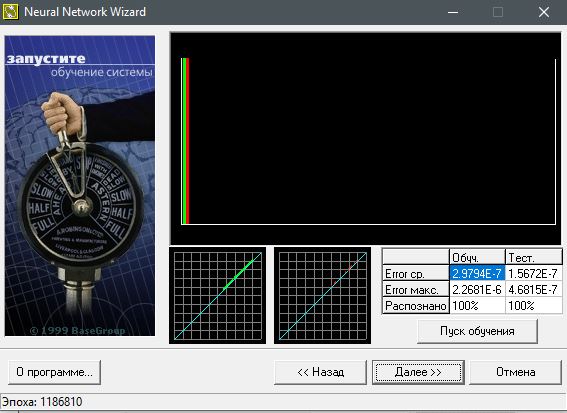


Рисунок 4 – Процесс обучения нейросети по примеру 1 в Neural Network Wizard

На верхней диаграмме показано распределение ошибки обучения: по горизонтали – значение ошибки (чем правее столбец, тем больше ошибка), по вертикали – количество примеров из выборки с данной ошибкой (чем выше столбец, тем больше примеров с указанной ошибкой). Зеленые столбцы – ошибка на рабочей обучающей выборке, красные – на тестовой. В процессе обучения столбцы должны стремиться в левую часть диаграммы [7].

Ниже диаграммы отображается распределение примеров на рабочей и тестовой выборках в которой каждый пример изображается точкой. Чем ближе точка к диагонали, тем точнее нейросеть предсказала ее значение.

В дальнейшем мы протестируем сеть на трех значениях и вычислим абсолютную погрешность вычислений.

Воспользуемся формулами для вычисления тестовых значений:

(1)

(2)

(3)

где – это последнее из значений аргумента обучающей выборки, а , соответственно, первое.

Вычисляя по формулам (1), (2), (3) получим:

В таблице 3 написаны результаты вычислений нейронной сети по данным значениям , и .

Таблица 3 – Результаты первого теста нейросети Neural Network Wizard

|  |  |
| --- | --- |
|  |  |
| 1,005 | 0,842442534907251 |
| 1,05 | 0,862254813550101 |
| 1,5 | 0,930928779757228 |

Вычислим абсолютную погрешность по формуле и занесем результаты в таблицу 4:

(4)

Таблица 4 – Абсолютная погрешность вычислений первого теста нейросетью

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 1,005 | 0,844161967 | 0,001719432 |
| 1,05 | 0,867423226 | 0,005168412 |
| 1,5 | 0,997494987 | 0,066566207 |

Таким образом можно сделать вывод о том, что чем дальше значения аргументов от обучающей выборки, тем менее точнее предсказание результатов.

Рассмотрим второй пример – линейную функцию . Значения аргумента зададим в диапазоне [0; 30] с шагом 0,1, а параметры настройки оставим такими же, как и в предыдущем примере (см. рис 5).

Мы остановим обучение, когда пройдет 1000000 эпоха. Заметно, что данный набор для обучения наиболее прост для нейросети, так как ошибки обучения и тестов значительно меньше, чем в предыдущем примере.

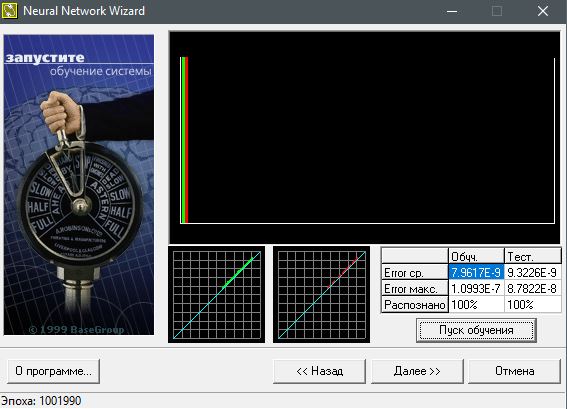


Рисунок 5 – Процесс обучения нейросети по примеру 2 в Neural Network Wizard

Произведем расчеты по формулам (1), (2), (3):

В связи с тем, что диапазон более широк, чем в 1 примере, отклонение от обучающей выборки получилось выше (см. таблицы 5 и 6).

Таблица 5 – Результаты второго теста нейросети Neural Network Wizard

|  |  |
| --- | --- |
|  |  |
| 35,35 | 70,2969021854567 |
| 38,5 | 74,6551577658862 |
| 70 | 86,8630869362652 |

Таблица 6 – Абсолютная погрешность вычислений второго теста нейросетью

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 35,35 | 71,7 | 1,403097815 |
| 38,5 | 78 | 3,344842234 |
| 70 | 141 | 54,13691306 |

Также можно протестировать нейросеть с тем же отклонением от значения , что и в предыдущем примере. Нетрудно заметить, что эта разность равна соответственно: 0,005, 0,05 и 0,5. Составим новые значения , и и вычислим их абсолютную погрешность в таблице 7.

Таблица 7 – Абсолютная погрешность вычислений второго теста нейросетью на основе отклонения от сети из набора первого теста

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 35,005 | 71,01 | 69,7624206931733 | 1,247579307 |
| 35,05 | 71,1 | 69,8327553194583 | 1,267244681 |
| 35,5 | 72 | 70,5258544974459 | 1,474145503 |

По итогу можем сказать, что погрешность вычислений получилась вышее, чем у периодической функции, даже с учетом отклонения.

Рассмотрим третий пример , где c единичным шагом. В этот раз мы не будем прерывать обучение на миллионной эпохе, а закончим его полностью, но при этом поставим галочку, что будем использовать тестовое множество как валидационное, чтобы избежать «переобучения» сети.

Вычислим тестовые параметры (1), (2), (3): , , .

Вынесем результаты теста и абсолютную погрешность вычислений в таблицу 8.

Таблица 8 – Результаты и абсолютная погрешность третьего теста нейросетью

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 100,99 | 4,624874769 | 4,62199152788928 | 0,002883241 |
| 109,9 | 4,708628894 | 4,69622317602088 | 0,012405718 |
| 199 | 5,298317367 | 4,94874269457581 | 0,349574672 |

Результаты логарифмической функции оказались лучше, чем у линейной и хуже, чем у периодической. Это можно объяснить тем, что многослойная нейронная сеть на основе которой работает Neural Network Wizard, используется для работы с нелинейными вычислениями, следовательно, для линейной функции результат работы будет намного хуже, чем с остальными. Это следует учитывать при работе с данным приложением и нейросетями в целом.

**ЗАКЛЮЧЕНИЕ**

Развитие нейронных сетей вызвало неоднозначную реакцию со стороны учёных. Некоторые положения вызывали как критику, так и энтузиазм в работе с данной областью, но по прошествии десятилетий стало ясно, что для выполнения некоторых задач и алгоритмов использование ИНС является самым оптимальным решением. В тоже время данной области необходимы нововведения, которые в дальнейшем помогут в развитии интеллектуальных систем, а также смогут улучшить качество выполнения различной работы нейросетями.

В ходе данной курсовой работы были рассмотрены основные теоретические основы и методы работы с искусственными нейронными сетями в области прогнозирования, а также сферы их применения. По результатам работы нейросетевого пакета Neural Network Wizard, который работает на основе многослойной нейронной сети и позволяет осуществлять задачу прогнозирования, были проведены тестовые прогнозы значений некоторых функций и сделаны выводы о том, какие данные необходимо задавать, а также какие из них приведут к наиболее точному прогнозу значений нейросети. Также были изучены основные параметры при настройке и критерии остановки обучения. Тем самым была достигнута цель курсовой работы.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1 Баллод Б.А., Елизарова Н.Н. Методы и алгоритмы принятия решений в экономике / – 2-е изд., перераб. – СПб.: Издательство «Лань», 2018. – 272 с.

2 Ежов А. Шумский С. Нейрокомпьютинг и его применение в экономике и бизнесе / М.: МИФИ, 1998. – 224 с.

3 Уоссермен Ф. Нейросетевая техника: теория и практика / Пер. с английского Ю.А.Зуев. – М.:Мир, 1992. – 184 с.

4 Хайкин С. Нейронные сети. Полный курс / 2-e изд. Пер. с англ. – М.: Издательский дом "Вильямс", 2006. – 1104 с.

5 Каллан Р. Основные концепции нейронных сетей / Пер. с англ. – М.: Издательский дом "Вильямc", 2001. – 287 с.

6 Вьюгин В. Математические основы теории машинного обучения и прогнозирования / М.: МЦНМО, 2013. – 390 с.

7 Персептроны, Однослойный персептрон – Искусственные нейронные сети: [Электронный ресурс] // Студенческая библиотека онлайн (info{aт}studbooks.net), 2013 - 2018. URL: <https://studbooks.net/1160948/informatika/perseptrony>. (10.11.2018)

8 Порядок работы с Neural Network Wizard: [Электронный ресурс] // studopedia.ru. URL: <https://studopedia.ru/18_2723_poryadok-raboti-s-Neural-Network-Wizard.html>. (12.12.2018)

9 Прогнозирование на основе нейронных сетей, Общий подход к прогнозированию с помощью нейронных сетей - Прогнозирующие системы: [Электронный ресурс] // Vuzlit - архив студенческих работ (info{aт}vuzlit.ru) © 2017 - 2018 URL: <https://vuzlit.ru/1146271/prognozirovanie_osnove_neyronnyh_setey#248>. (10.12.2018)

10 С. Осовский. Нейронные сети для обработки информации: Пер. с польского И.Д. Рудинского. –М.: Финансы и статистика, 2002. –344 с.

11 С. А. Терехов Лекции по теории и приложениям искусственных нейронных сетей [Электронный ресурс] // 1998, С. А. Терехов, электронная версия LiveInternet. URL: <http://alife.narod.ru/lectures/neural/Neu_index.htm>. (03.12.2018)

12 Боровиков В.П. Нейронные сети. Statistica Neural Networks. Методология и технологии современного анализа данных / 2-е изд., перераб. и доп. – М.: Горячая линия - Телеком, 2008. – 392 с.

13 Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы / М.: Горячая линия - Телеком, 2006. – 452 с.

14 Назаров А.В., Лоскутов А.И. Нейросетевые алгоритмы прогнозирования и оптимизации систем / СПб.: Наука и Техника, 2003. – 384 с.

15 Горбань А.Н. Обучение нейронных сетей / М.: ПараГраф, 1990. – 160 с.

16 Галушкин А.И. Нейронные сети: основы теории / М.: Горячая линия-Телеком, 2010. – 496 с.

