УПРАВЛЕНИЕ ОБРАЗОВАНИЯ ГОРОДА ПЕНЗЫ
Муниципальное бюджетное общеобразовательное учреждение
«Гимназия № 53» г. Пензы
(МБОУ «Гимназия № 53» г. Пензы)

V открытый региональный конкурс исследовательских и проектных работ школьников «Высший пилотаж - Пенза» 2023

# ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ ДЛЯ КРЕДИТНОГО СКОРИНГА

Выполнил:
Козлов Фёдор,
учащийся 10 класса
МБОУ «Гимназия № 53» г. Пензы
Научный руководитель:
Артюхина Елена Владимировна, старший
преподаватель кафедры
«Компьютерные технологии» ПГУ
Руководитель:
Святкин Алексей Викторович,
Учитель информатики
МБОУ «Гимназия № 53» г. Пензы

Пенза, 2022

Оглавление

**Введение**

В настоящее время искусственные нейронные сети широко используются при решении самых разнообразных задач особенно там, где обычные алгоритмические решения оказываются неэффективными или вовсе невозможными. Например, при распознавании текстов, игре на фондовых рынках, контекстной рекламе в Интернете, фильтрации спама, проверки проведения подозрительных операций по банковским картам, системы безопасности и видеонаблюдения и др. Решения на основе искусственных нейронных сетей становятся все более совершенными и популярными [1]].

***Актуальность***:

В наши дни возрастает необходимость в системах, которые способны не только выполнять однажды запрограммированную последовательность действий над заранее определенными данными, но и способны сами анализировать вновь поступающую информацию, находить в ней закономерности, производить прогнозирование и т.д. В этой области приложений самым лучшим образом зарекомендовали себя так называемые нейронные сети – самообучающиеся системы, имитирующие деятельность человеческого мозга [2-4].

Нейронные сети широко применяются и в банковской сфере, в частности для решения задач кредитного скоринга. Опыт последнего времени еще раз подтвердил, что в современном мире нельзя полагаться исключительно на экспертный опыт и на старые системы кредитного скоринга. Необходимо учитывать весь объем информации о клиентах-заемщиках и периодически обновлять базы данных, используемые для скоринговых моделей. Только в этом случае в отношении той или иной кредитной заявки возможно принятие оптимального решения, и, как следствие, значительное снижение кредитных рисков. Снижению рисков также способствуют кредитные бюро, в которых хранится кредитная история заемщиков.

***Объектом исследования*** являются нейронные сети.

***Целью*** работы является исследование и создание нейронных сетей, позволяющих эффективно решать задачу кредитного скоринга.

***Задачи:***

1. Изучить основные понятия нейронных сетей.
2. Познакомиться с возможностями платформы Deductor, в частности с инструментарием для работы с нейронными сетями.
3. Исследовать возможность применения многослойного персептрона для решения задачи кредитного скоринга.
4. Провести экспериментальные исследования для выбора лучшей конфигурации нейронной сети для решения поставленной проблемы.

***Ожидаемые результаты и практическая значимость***

Разработанный нейросетевые модели позволят эффективно проводить кредитный скоринг. Полученные в работе результаты могут быть использованы при создании банковских систем на основе искусственных нейронных сетей.

**Теоретическая часть**

В настоящее время много говорят об искусственном интеллекте. В искусственном интеллекте приято выделять *сильный ИИ* — интеллектуальный алгоритм, способный решать широкий спектр интеллектуальных задач как минимум наравне с человеческим разумом" и слабый ИИ, *прикладной ИИ* — интеллектуальный алгоритм, имитирующий человеческий разум в решении конкретных узкоспециализированных задач. Сильный ИИ в настоящее время — это, скорее цель для науки. В области прикладного ИИ достигнуты огромные результаты: распознавание лиц, общение на естественном языке, поиск информации и т.п.).

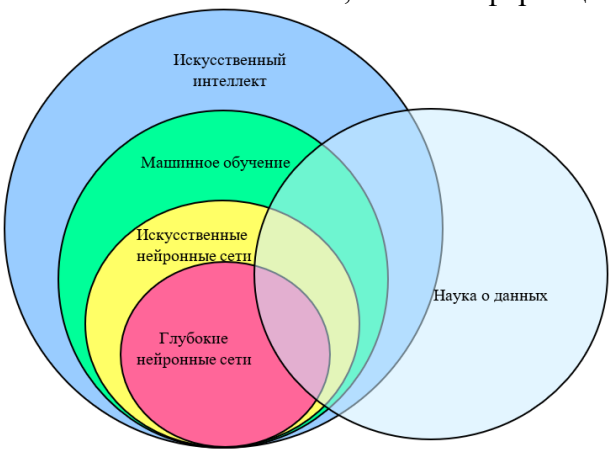
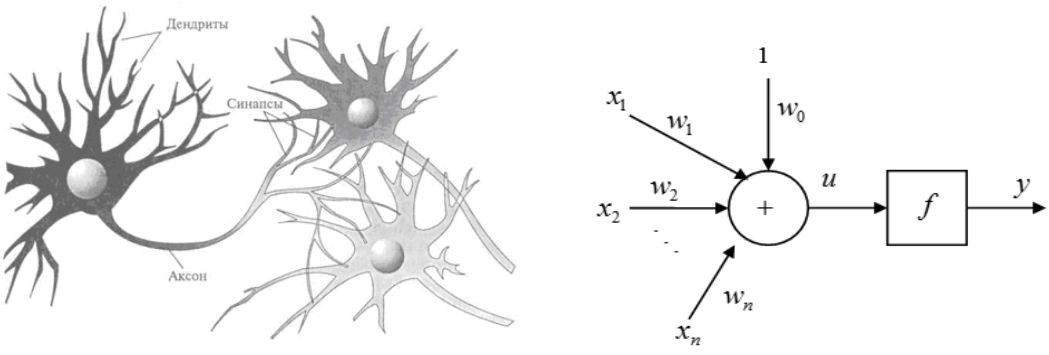



Рис. 1. Структура прикладного искусственного интеллекта

Структура современного прикладного ИИ показана на рис. 1. Основным направление современного искусственного интеллекта является машинное обучение (Machine Learning) — технологии автоматического обучения алгоритмов ИИ на примерах, в результате чего качество работы алгоритмов повышается [3].

Важнейшей (но не единственной) частью современного машинного обучения являются *искусственные нейронные сети* (Artificial Neural Networks) — математические модели, состоящие из слоёв "нейронов", построенных по принципу организации и функционирования биологических нейронных сетей (рис. 2а) [2]. Первая модель нейрона была предложена в 1943 году (рис. 2б).




а б

Рис. 2 Строение биологического нейрона и Модель нейрона Мак-Каллока-Питса

Нейрон описывается зависимостью

$$y = f(w_1x_1 + w_2x_2 + \cdots + w_nx_n + w_0),$$

где $y$ – выход нейрона, $x_i$ – входные сигналы, $w_i$ – синаптические веса, $w_0$ – порог (смещение), $f(u)$ – пороговая функция активации:

$$f(u)=\begin{cases}1, & \text{если } u>0,\\ 0, & \text{если } u\le 0.\end{cases}$$

*Многослойный персептрон* — многослойная сеть, состоящая из нейронов, расположенных на разных уровнях, причем, помимо входного и выходного слоев, имеется еще, как минимум один внутренний, т.е. скрытый слой. Изображенная на рис. 3 сеть содержит M слоев, среди которых выделяют первый (входной) слой, промежуточные (скрытые) слои и выходной слой.

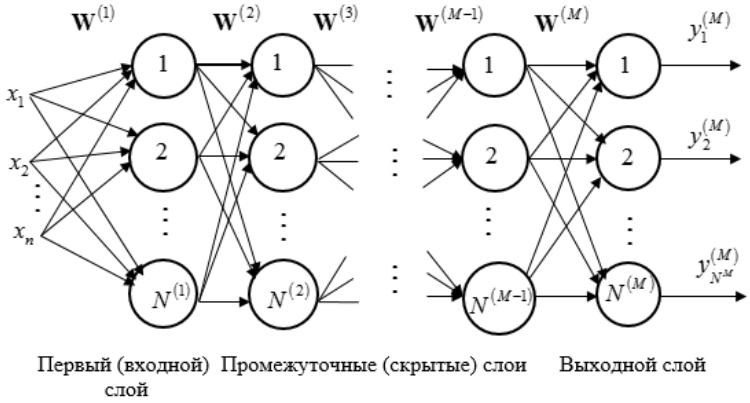



Рис. 3. Структура многослойной сети

Новым направлением нейронных сетей являются *глубокие нейронные сети* (Deep Neural Networks) — сети, содержание большое число слоев, они требуют мощных достаточно сложных алгоритмов обучения. С искусственным интеллектом пересекается наука о данных (Data Science) — раздел информатики, изучающий проблемы анализа, обработки и представления данных в цифровой форме [4].

Машины не учатся как люди [6]. Машинное обучение заключается в поиске математической формулы, применение которой к набору входных данных (обучающему набору) дает желаемые результаты. Эта формула должна также обладать обобщающей способностью — формировать правильные выходные данные для входных данных, отличающихся от обучающих данных, при условии, что входные данные имеют такое же или близкое статистическое распределение, что и обучающие данные. Пока машинное обучение уступает обучению людей. Например, если модель машинного обучения не обучена распознавать изображения при их повороте, то она не распознает изображение при повороте, а человек легко решает эту задачу. Термин "машинное обучение" был придуман в фирме IBM в 1959 году из маркетинговых соображений для привлечения клиентов и талантливых сотрудников.

На основе эмпирических данных программа настраивает математическую модель (обучается). Эмпирические данные могут быть получены самой программой в предыдущие сеансы ее работы или просто предъявлены ей. Обучение делится на следующие основные виды:

*Обучение с учителем* [2-6] называют еще обучением на размеченных данных. При обучении с учителем имеется набор обучающих (тренировочных) примеров (learning set, или training set). Обучающие примеры — это наборы признаков, факторов, описывающих отдельные объекты, например, показатели, описывающие состояние больного, оцифрованная картинка. Каждому примеру соответствует известный ответ (метка, отклик, реакция, целевое значение). Задача обучения — так настроить на имеющихся примерах модель, чтобы на тех данных, на которых модель не обучалась, она выдавала достаточно точный ответ. Естественно, что обучающий набор и данные, на которых будет применяться модель, должны быть из одного множества. Процесс обучения нейронной сети схематично представлен на рис. 4.

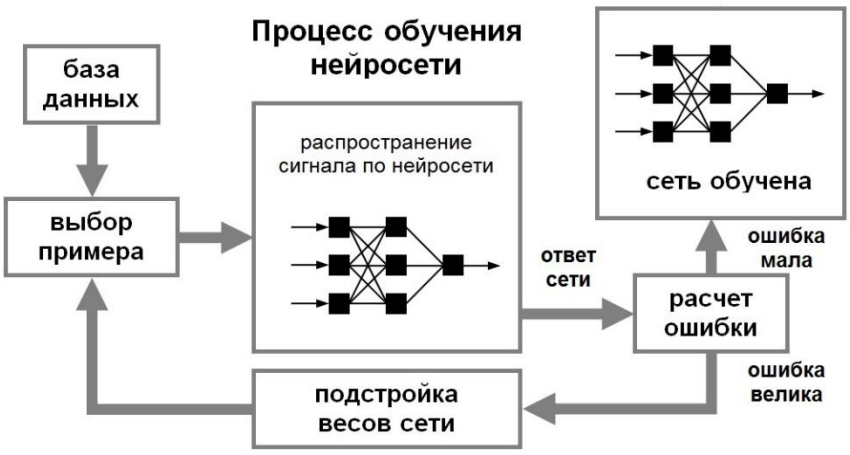



Рис. 4. Процесс обучения НС с учителем

Основными задачами, решаемыми с помощью обучения с учителем, являются задачи классификации и регрессии. При решении задачи классификации необходимо отнести объект к одному из заранее известных классов. Типичные задачи классификации — задачи распознавания образов. Например, множество фотографий необходимо разделить на фотографии определенных объектов; множество клиентов банка разделить на надежных заемщиков, которым можно давать кредит, и ненадежных; разделить пациентов больницы на подозреваемых в заболевании и здоровых. При решении задач регрессии необходимо предсказать значение некоторой функции, например, по характеристикам квартиры оценить ее стоимость, предсказать курс акций.

*Обучение без учителя* — это обучение на неразмеченных данных, когда целевые значения неизвестны. Типичная задача обучения без учителя — это кластеризация, когда необходимо разделить объекты на заранее неизвестные группы (кластеры) в зависимости от близости их признаков. Например, пользователей мобильного оператора можно разделить на кластеры в зависимости от особенностей пользователей.

*Обучение с частичным привлечением учителя* использует обучающий набор данных, включающий как размеченные, так и не размеченные данные (таких данных, обычно, намного больше, чем размеченных). Привлечение к обучению неразмеченных данных вносит больше разнообразия в обучающие данные и повышает точность модели.

В *обучении с подкреплением* модель "живет" в некотором окружении и выполняет некоторые действия. Разные действия приносят разные вознаграждения. Цель алгоритма обучения с подкреплением — построить поведение, приносящее максимальные вознаграждения.

**Практическая часть**

Для реализации методов Data Mining предлагается использовать платформу Deductor, которая разработана компанией BaseGroup Labs [1]. Выбор инструментария обуславливается удобством программного обеспечения, его доступностью и правилами лицензирования. Deductor 5 является аналитической платформой, т.е. основой для создания законченных прикладных решений. Реализованные в Deductor технологии позволяют на базе единой архитектуры пройти все этапы построения аналитической системы: от создания хранилища данных до автоматического подбора моделей и визуализации полученных результатов.

*Исходные данные*. Анализ кредитоспособности производится на основе скоринговых анкет. Анализируемые данные в этом случае представляются в виде обычной таблицы, в которой содержатся прецеденты. В данном случае их используется 1000. По причине секретности банковской информации и отсутствия возможности найти необходимые данные в кредитных бюро из-за высокой стоимости и возможности доступа к ней только юридических лиц, используются тестовые данные компании Basegroup [3]. Имеется файл (**`loans.txt`**,) с информацией о заемщиках банка. Структура файла представлена в табл. 1.

Таблица 1 Структура данных по заемщикам

| № | Поле | Описание | Тип |
|---|---|---|---|
| 1 | Код | Служебный код заявки | Целый |
| 2 | Дата | Дата выдачи кредита | Дата/время |
| 3 | О/Д, % | Коэффициент О/Д («Обязательства/Доход») в % | Вещественный |
| 4 | Возраст | Возраст заемщика (полных лет) на момент принятия решения о выдаче кредита | Целый |
| 5 | Проживание | Основание для проживания: собственник; муниципальное жилье; аренда | Строковый |
| 6 | Срок проживания в регионе | Менее 1 года; от 1 года до 5 лет; свыше 5 лет | Строковый |
| 7 | Семейное положение | Холост/не замужем; женат/замужем; разведен (-а)/вдовство; другое | Строковый |
| 8 | Образование | Среднее; среднее специальное; высшее | Строковый |
| 9 | Стаж работы на последнем месте | Менее 1 года; от 1 года до 3 лет; свыше 3 лет | Строковый |
| 10 | Уровень должности | Сотрудник; руководитель среднего звена; руководитель высшего звена | Строковый |
| 11 | Кредитная история | Информация берется из бюро кредитных историй. Если имеется негативная информация о клиенте (просрочки по прошлым кредитам), то ему присваивается категория «отрицательная» | Строковый |
| 12 | Просрочки свыше 60 дней | Факт наличия просрочек свыше 60 дней: 0 — отсутствовали, 1 — имели место | Целый |
| 13 | Тестовое множество | Служебный признак, отвечающий за то, к какому множеству относится запись. TRUE соответствует тестовому множеству | Логический |

Данная задача решалась с помощью инструмента логистическая регрессия (ошибка 16%) деревья решений (ошибка 10,71) в [3], что позволит провести сравнение результатов.

Примем простейшее правило отнесения заемщика к "плохим" — если у клиента имелась хотя бы одна просрочка свыше 60 дней, то относится к классу плохих. Для классификации заемщиков введем строковое вычисляемое поле *`Класс заемщика`*. Для этого используем инструмент *`Калькулятор`* (рис. 5).

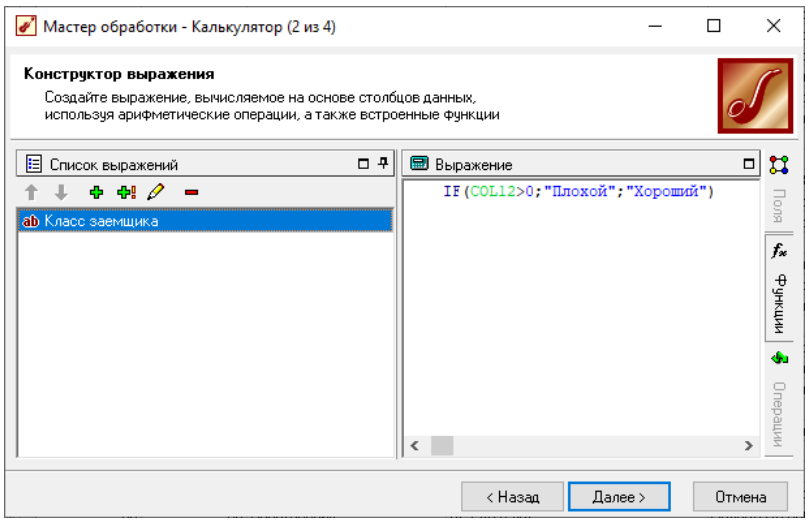



Рис. 5. Создание нового поля Класс заемщика

Откроем *Мастер обработки* и выберем в нем *Нейросеть*. Установим в входные и выходные поля, как показано на рис. 6. Настройку нормализации оставим по умолчанию.

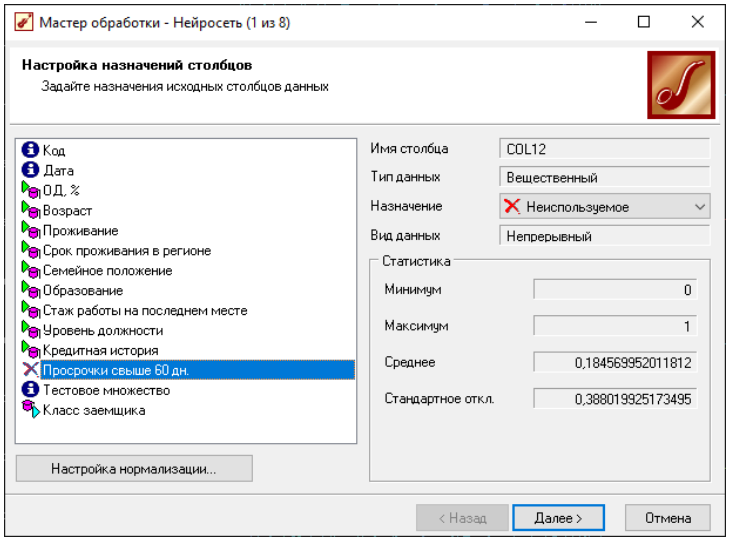



Рис. 6. Настройка данных для нейронной сети

Поля *Код*, *Тестовое множество* и *Дата* будут информационным, *Просрочки свыше 60 дн.* — неиспользуемым, *Класс заемщика* — выходным, остальные поля — входными.

В окне *Разбиение исходного множества* на подмножества (рис. 7) указывается разбиение на тестовое и обучающее множества.

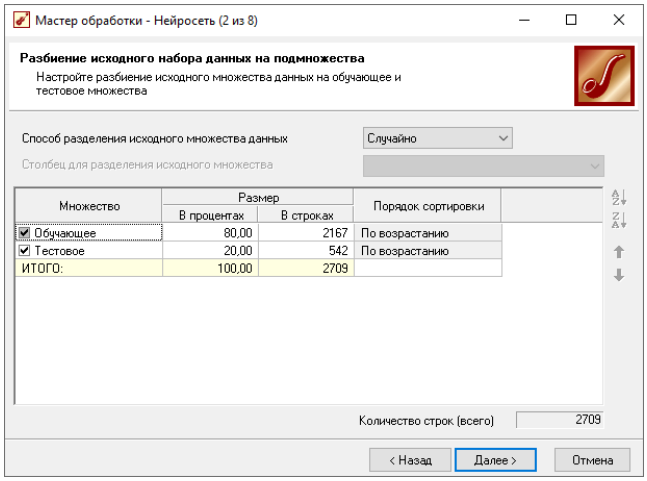



Рис. 7. Разбиение исходного множества на подмножества

Обучающее множество включает записи (примеры), которые будут использоваться в качестве входных данных, а также соответствующие целевые (желаемые) выходные значения. Тестовое множество также включает записи, содержащие входные и желаемые выходные значения, но используемое не для обучения модели, а для проверки его результатов. Используется подход к преодолению эффекта переобучения сети, основанный на организации целенаправленной процедуры прерывания обучения — на обучении с ранним остановом, для этого используется перекрестная проверка.

На следующем шаге задается структура нейронной сети (рис. 8.).

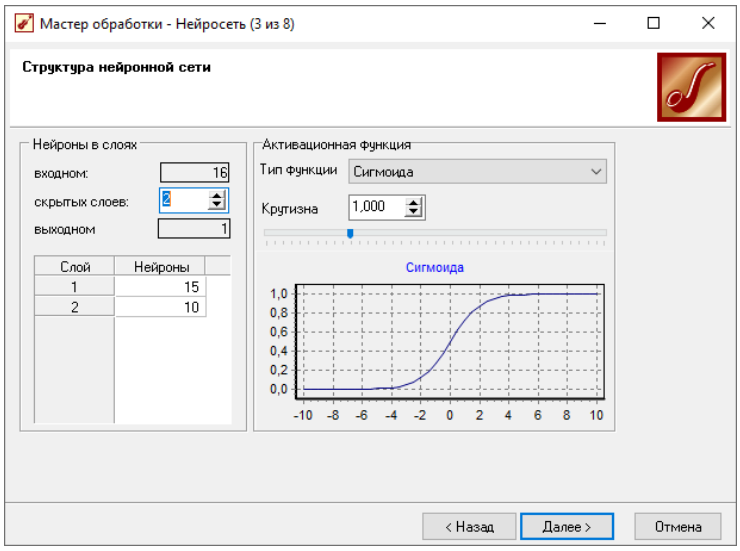



Рис. 8. Задание структуры сети

На этом шаге задаются количество скрытых слоев и нейронов в них, а также функция активации нейронов. Число нейронов во входном и выходном слоях автоматически устанавливается в соответствии с числом входных и выходных полей обучающей выборки и здесь изменить его нельзя.

В секции *Нейроны в слоях* необходимо указать количество скрытых слоев, т.е. слоев нейронной сети, расположенных между входным и выходным слоями. Количество скрытых слоев и количество нейронов в каждом скрытом слое определяются экспериментально. Как правило, достаточно сети с 1 или 2 скрытыми слоями. При выборе количества нейронов следует руководствоваться следующим правилом [6]: "Количество связей между нейронами должно быть примерно на порядок меньше количества примеров в обучающем множестве". Количество связей рассчитывается как связь каждого нейрона со всеми нейронами соседних слоев, включая связи на входном и выходном слоях. Слишком большое количество нейронов может привести к так называемому "переобучению" сети, когда она выдает хорошие результаты на примерах, входящих в обучающую выборку, но практически не работает на других примерах.

В секции *Активационная функция* определим тип функции активации нейронов и ее крутизну. Для этого в списке *Тип функции* следует выбрать нужную функцию активации:

— *Сигмоида* (сигмоидальная функция) $f(s)=\frac{1}{1+e^{-as}}$,

где $a$ — задаваемая крутизна, $s$ — выход адаптивного сумматора нейрона. В нижней части окна отображается график выбранной функции в соответствии с установленной крутизной.

Далее выбирается алгоритм обучения нейронной сети (рис. 9) *Resilient Propagation (RPROP)* — "эластичный" Back Propagation. Для данного алгоритма указываются следующие параметры:

— *Шаг спуска* — коэффициент увеличения скорости обучения, который определяет шаг увеличения скорости обучения при не достижении алгоритмом оптимального результата.

— *Шаг подъема* — коэффициент уменьшения скорости обучения. Задается шаг уменьшения скорости обучения в случае пропуска алгоритмом оптимального результата.

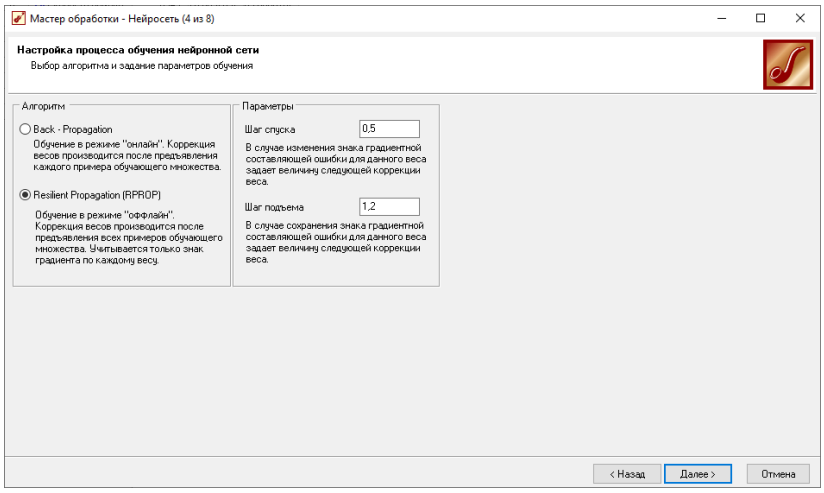



Рис. 9. Настройка процесса обучения нейронной сети

Выберем алгоритм *Resilient Propagation* с параметрами, показанными на рис. 8, и перейдем к следующему шагу — настройке параметров остановки обучения (рис. 10).

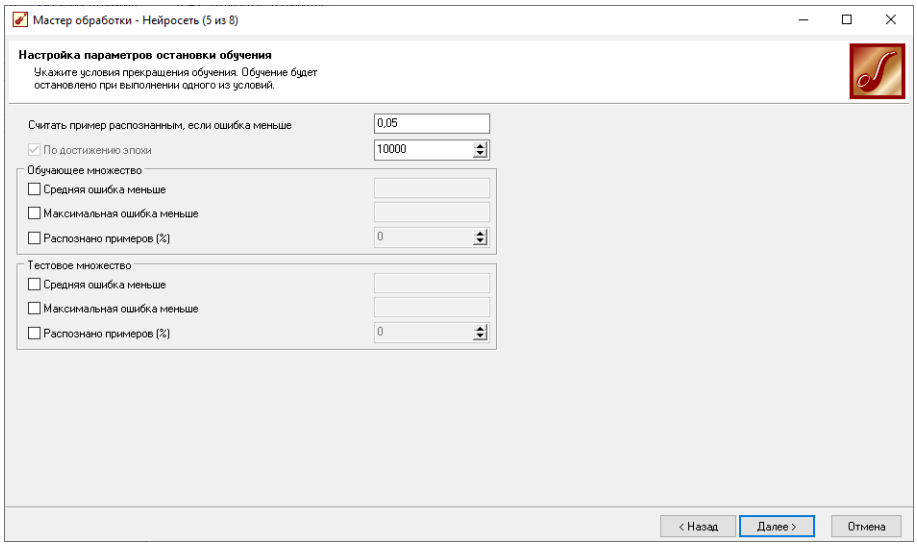



Рис. 10. Настройка остановки обучения

На данном шаге необходимо задать условия, при выполнении которых обучение будет прекращено. Выберем *Считать пример распознанным, если ошибка меньше* — критерием останова является условие, что рассогласование между эталонным и реальным выходом сети становится меньше заданного значения. При выборе нескольких условий останов процесса обучения происходит по достижении хотя бы одного из них.

Установим параметры остановки обучения, показанные на рис. 10, и перейдем к шагу обучения нейронной сети (рис. 11).

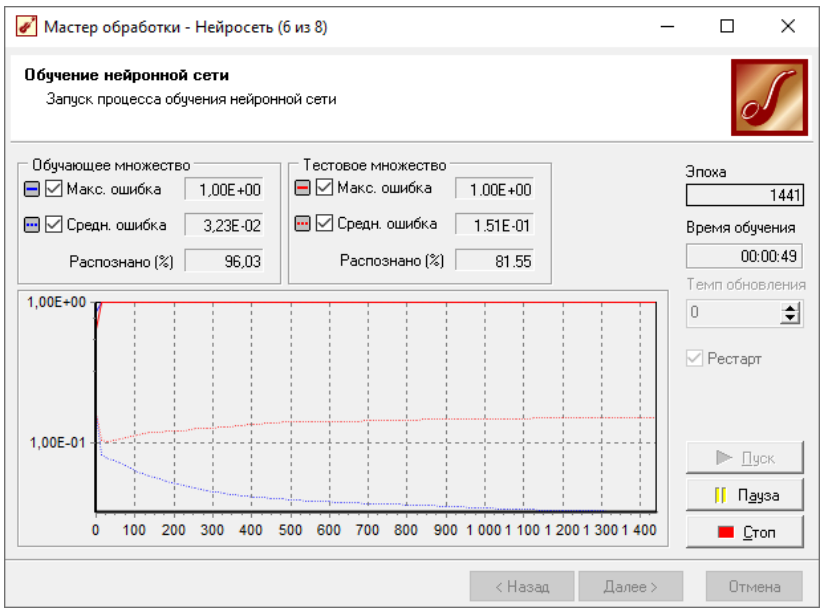



Рис. 11. Обучение нейронной сети

На данном шаге производится собственно процесс обучения нейронной сети. Запускается процесс обучения кнопкой *Пуск*. Остановить процесс обучения можно с помощью кнопки *Стоп*.

В процессе обучения в секциях *Обучающее множество* и *Тестовое множество* отображаются максимальная квадратичная ошибка и средняя квадратичная ошибка на обучающем множестве и тестовом множестве соответственно, а также процент распознанных примеров. В процессе обучения в окне отображаются графики хода обучения для обучающего (синяя линия) и тестового (красная линия) множеств. В правой части окна постоянно отображаются номер текущей эпохи и время, прошедшее с начала обучения.

Флажок *Рестарт* позволяет включить режим инициализации начальных весов сети случайными значениями. Если флажок сброшен, то при повторном запуске обучения после остановки будет иметь место так называемое "дообучение сети", когда обучение будет начато с текущими весами.

После построения модели для просмотра результатов обучения представим полученные данные в виде графа нейросети, диаграммы и диаграммы рассеяния. Зададим также вывод графа нейронной сети.

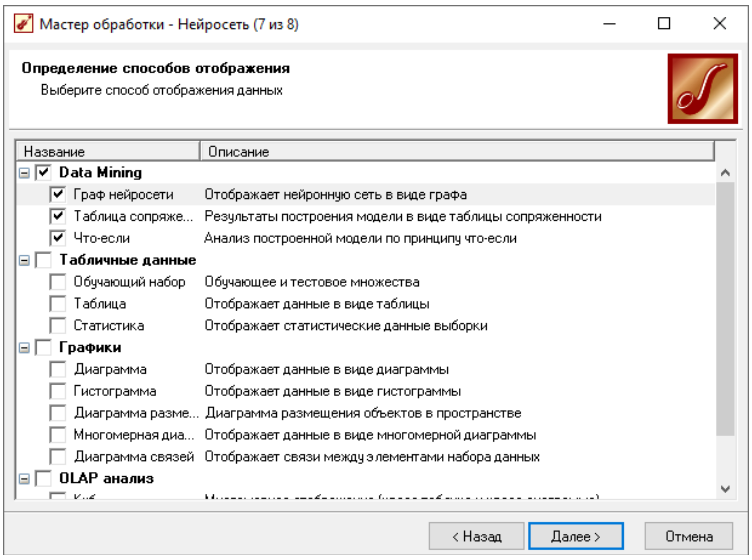



Рис. 12. Определение способов отображения данных

Граф нейронной сети (рис. 13) графически представляет нейронную сеть со всеми ее нейронами и синаптическими связями. При этом отображается не только структура нейронной сети, но и значения весов, которые принимают те или иные нейроны. В зависимости от значения веса он отображается определенным цветом, а соответствующее значение можно определить по цветовой шкале, расположенной внизу окна.

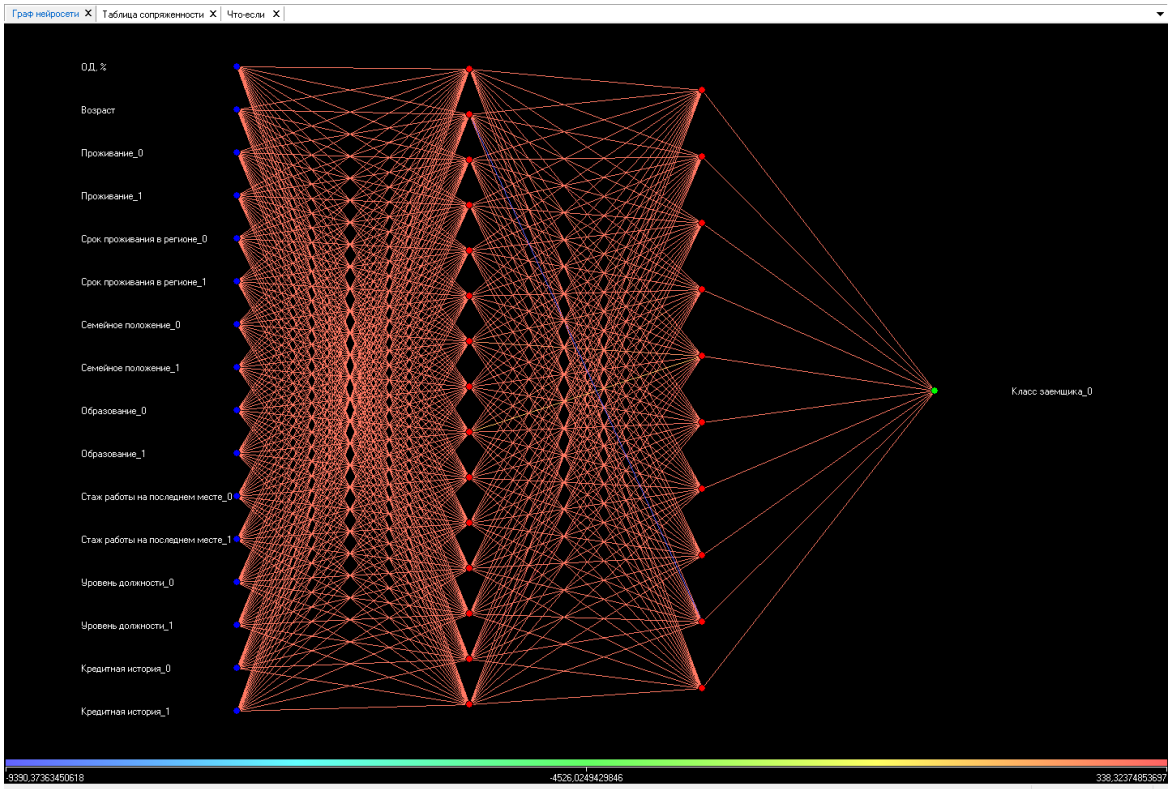



Рис. 13. Граф нейронной сети

На рис. 13 представлен граф нейронной сети с 15 нейронами в первом скрытом слое и 10 во втором. Входной слой 16 нейронов, выходной 1. Цвет ребер показывает значение весов в результате обучения сети (видно на шкале в нижней части рисунка).

Вкладка *Что-если* позволяет ответить на вопрос, что получим в качестве следствия, если выберем данные условия, например, изменяя входные поля, например при получении очередной анкеты, мы сможем получить оценку класса заемщика (рис. 14).

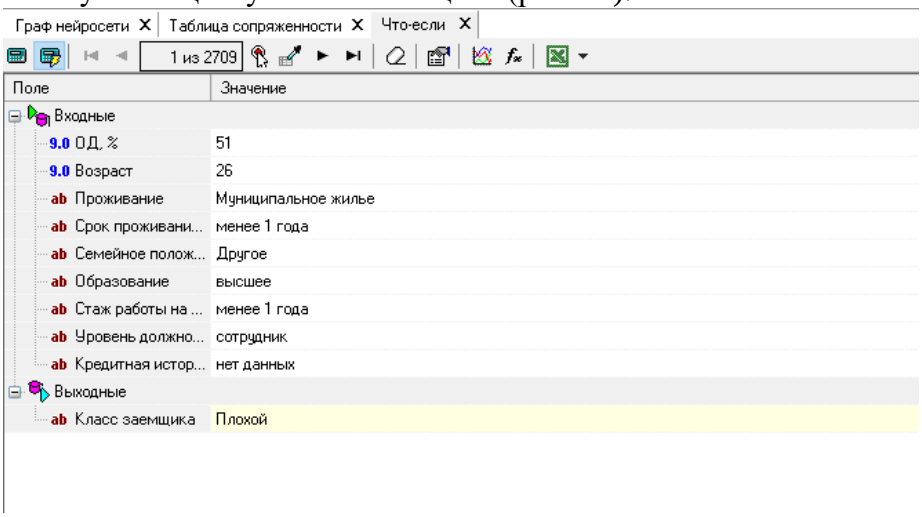



Рис. 14. Вкладка *Что-если*

Провели серию экспериментов, с различными архитектурами сети, рассматривались однослойные и двухслойны сети, лучшие результаты приведены в таблице 2. Ошибка 1 рода (отказ «хорошему» заемщику), ошибка 2 рода (выдача кредита «плохому» заемщику).

Таблица 2 Результаты обучение НС

| Количество нейронов в слоях | Ошибка, % | Ошибка 1 рода, % | Ошибка 2 рода, % |
|---|---|---|---|
| 16x5x1 | 8,55 | 4,61 | 3,94 |
| 16x8x1 | 7,16 | 4,39 | 2,77 |
| 16x10x1 | 5,91 | 3,21 | 2,70 |
| 16x12x1 | 6,13 | 3,14 | 2,99 |
| 16x15x1 | 5,87 | 3,40 | 2,47 |
| **16x20x1** | **5,54** | **2,58** | **2,96** |
| 16x5x7x1 | 7,27 | 3,54 | 3,73 |
| **16x15x10x1** | **4,84** | **2,62** | **2,22** |

Из результата экспериментов видно, что для многослойного персептрона с одним скрытым слоем (рис. 15) лучший результат удалость достигнуть с 20 нейронами в скрытом слое, ошибка сети составила 5,54% (ошибка 1 рода 2,58%, ошибка 2 рода 2,96).

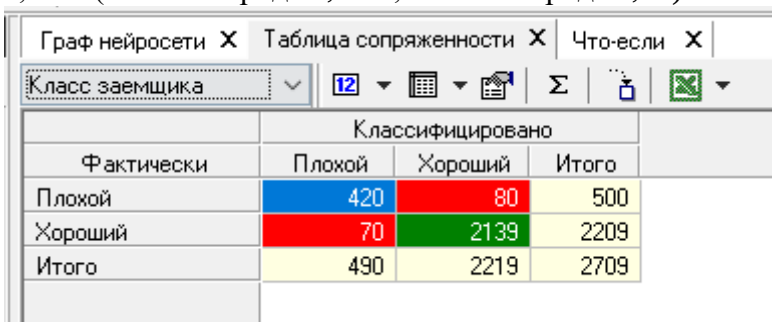



Рис. 15 – Таблица сопряженности для сети **16x20x1**

Для двухслойной сети лучшие результаты (рис. 16) достигнуты для сетей с 15 нейронами в первом скрытом слое и 10 нейронами во втором: ошибка сети составила 4,84% (ошибка 1 рода 2,62%, ошибка 2 рода 2,22%).

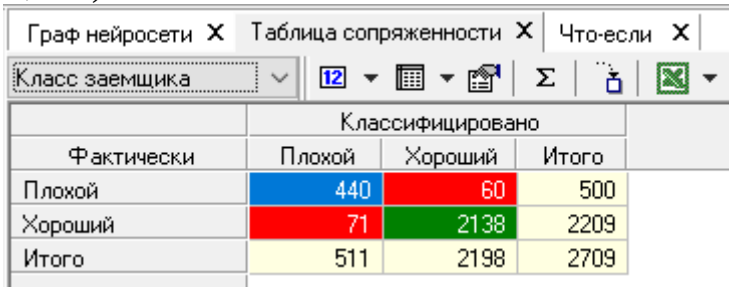



Рис. 16 – Таблица сопряженности для сети **16x15x10x1**

Сравнивая полученные результаты с имеющимися в работе [3] результатами, можно сделать вывод что нам удалось получить лучший результат используя нейронные сети с различной архитектурой. Нейронные сети позволяют эффективно строить нелинейные зависимости, более точно описывающие наборы данных.

## Заключение

В ходе теоретических исследований были изучены основные понятия: искусственный нейрон, нейронная сеть, многослойный персептрон, обучение нейронной сети и другие.

Для моделирования работы указанных алгоритмов была выбрана платформа Deductor, разработанная компанией BaseGroup Labs. Познакомились с возможностями платформы Deductor.

В ходе практической части были исследованы нейронные сети с одним и двумя скрытыми слоями. Лучший результат достигается для сети с 15 нейронами в первом слое и 10 нейронами во втором, ошибка сети составила 4,84%.

В процессе работы были выполнены все поставленные задачи и достигнута цель.

Следует отметить, что данная сеть будет нуждаться в переобучении с течением времени, на новых данных, так как меняется экономическая ситуация, изменяются данные, указанные в анкетах заемщиков, сеть должна обучаться на актуальных данных. Что сделает сеть адаптируемой к изменяющимся параметрам внешней среды.

**Список литературы и источников**

1. Цаунит, А. Н. Перспективы развития и применения нейронных сетей / А. Н. Цаунит. — Текст : непосредственный // Молодой ученый. — 2021. — № 23 (365). — С. 114-117. — URL: https://moluch.ru/archive/365/81791/ (дата обращения: 09.12.2022)

2. Нейронные сети — математический аппарат URL: https://moluch.ru/archive/365/81791/ (дата обращения: 09.11.2022)

3. Паклин Н. Б., Орешков В. И. Бизнес-аналитика: от данных к знаниям.— СПб.: Питер, 2013. — 704 с.

4. Искусственная нейронная сеть [Электронный ресурс]. URL: http://ru.wikipedia.org/wiki/Искусственная_нейронная_сеть / (дата обращения: 09.12.2022)

5. Элементарное введение в технологию нейронных сетей с примерами программ / Р. Тадеусевич, Б. Боровик, Т. Гончаж, Б. Леппер. — М: Горячая линия-Телеком, 2011. — С. 408.

6. Бурков Андрей. Машинное обучение без лишних слов.— СПб.: Питер, 2020. — 192 с.