

Министерство науки и высшего образования Российской Федерации  
федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Казанский национальный исследовательский технический  
университет им. А.Н. Туполева-КАИ»

На правах рукописи



**Катасёва Дина Владимировна**

**НЕЙРОНЕЧЕТКАЯ МОДЕЛЬ И ПРОГРАММНЫЙ КОМПЛЕКС  
ФОРМИРОВАНИЯ БАЗ ЗНАНИЙ ИНТЕЛЛЕКТУАЛЬНЫХ  
СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ  
ПО ОЦЕНКЕ СОСТОЯНИЯ ОБЪЕКТОВ**

Специальность:

05.13.18 – Математическое моделирование, численные методы  
и комплексы программ

Диссертация на соискание ученой степени  
кандидата технических наук

Научный руководитель:  
доктор технических наук,  
профессор Исмагилов И.И.

Казань – 2022

## ОГЛАВЛЕНИЕ

<b>Введение .....</b>	<b>5</b>
<b>1 Анализ предметной области и постановка задачи формирования и использования баз знаний для систем поддержки принятия решений по оценке состояния объектов .....</b>	<b>15</b>
<b>1.1 Задача оценки состояния объектов .....</b>	<b>15</b>
<b>1.2 Применение интеллектуальных систем поддержки принятия решений для оценки состояния объектов .....</b>	<b>16</b>
<b>1.3 Структура и состав систем поддержки принятия решений .....</b>	<b>18</b>
<b>1.4 Анализ и выбор модели представления знаний для системы поддержки принятия решений .....</b>	<b>20</b>
<b>1.5 Использование нечетких нейронных сетей для формирования баз знаний .....</b>	<b>22</b>
<b>1.6 Анализ особенностей данных для оценки состояния объектов .....</b>	<b>24</b>
<b>1.7 Формирование требований к виду нечетких правил, алгоритму вывода на правилах и нечеткой нейронной сети .....</b>	<b>25</b>
<b>1.8 Постановка задачи формирования и использования баз знаний для оценки состояния объектов .....</b>	<b>25</b>
<b>1.9 Выводы .....</b>	<b>26</b>
<b>2 Разработка нейронечеткой модели формирования баз знаний для оценки состояния объектов .....</b>	<b>28</b>
<b>2.1 Сравнительный анализ основных видов нечетко-продукционных правил для оценки состояния объектов .....</b>	<b>28</b>
<b>2.2 Предложенный вид нечетко-продукционных правил .....</b>	<b>32</b>
<b>2.3 Алгоритм логического вывода на правилах .....</b>	<b>33</b>
<b>2.4 Критерий применения алгоритма в условиях некомплектности входных данных .....</b>	<b>35</b>
<b>2.5 Этапы построения нейронечеткой модели .....</b>	<b>37</b>
2.5.1 Описание этапов построения модели .....	37
2.5.2 Задание состава слоев и определение их функциональности .....	37
2.5.3 Определение состава нейронов в слоях нечеткой нейронной сети .....	39
2.5.4 Формирование из исходных данных выборок для обучения, тестирования и валидации нечеткой нейронной сети .....	41
2.5.5 Обучение и тестирование нечеткой нейронной сети .....	43
2.5.6 Оценка обобщающей способности нейронечеткой модели .....	45
<b>2.6 Постановка задачи разработки методов и алгоритмов построения нейронечеткой модели и формирования баз знаний .....</b>	<b>46</b>
<b>2.7 Выводы .....</b>	<b>47</b>

<b>3</b>	<b>Методы и алгоритмы построения нейронечеткой модели и формирования баз знаний для оценки состояния объектов.....</b>	<b>49</b>
3.1	Метод сэмплинга данных для формирования обучающей, тестовой и валидационной выборок.....	49
3.2	Метод расчета весовых коэффициентов в нечеткой нейронной сети....	52
3.3	Построение нейронечеткой модели .....	54
3.3.1	Этапы параметрической идентификации нейронечеткой модели .....	54
3.3.2	Определение начальных значений параметров функций принадлежности .....	55
3.3.3	Метод обучения нечеткой нейронной сети .....	57
3.3.4	Выбор и кодирование параметров обучения.....	59
3.3.5	Генетический алгоритм настройки значений параметров функций принадлежности .....	61
3.3.6	Алгоритм обучения нечеткой нейронной сети .....	64
3.3.7	Алгоритм оценки обобщающей способности нейронечеткой модели...	66
3.4	Формирование базы знаний на основе построенной модели .....	67
3.4.1	Метод формирования базы знаний .....	67
3.4.2	Алгоритм формирования базы знаний.....	73
3.5	Выводы.....	75
<b>4</b>	<b>Программный комплекс формирования и использования баз знаний для оценки состояния объектов.....</b>	<b>77</b>
4.1	Описание разработанного программного комплекса.....	77
4.1.1	Общее описание программного комплекса .....	77
4.1.2	Структура и состав программного комплекса .....	78
4.1.3	Графический интерфейс программного комплекса .....	80
4.1.4	Алгоритм использования программного комплекса для поддержки принятия решений по оценке состояния объектов.....	87
4.2	Исследования, проведенные на базе программного комплекса.....	89
4.2.1	Постановка задачи исследований.....	89
4.2.2	Описание источника и наборов данных для исследований.....	90
4.2.3	Влияние числа градаций и формы функций принадлежности на результаты инициализации и обучения нечетких нейронных сетей .....	95
4.2.4	Влияние значений параметров генетического алгоритма на результаты обучения нечетких нейронных сетей.....	102
4.2.5	Рекомендации по выбору значений параметров инициализации и обучения нечетких нейронных сетей.....	107
4.2.6	Описание сформированных баз знаний при анализе наборов данных	107
4.2.7	Сравнение точности классификации на основе сформированных баз знаний с точностью других методов классификации .....	113

<b>4.3 Апробация программного комплекса при формировании баз знаний для оценки состояния объектов в различных предметных областях .....</b>	<b>115</b>
4.3.1 Формирование базы знаний для подбора геолого-технических мероприятий на нефтяном месторождении.....	115
4.3.2 Формирование базы знаний для оценки функционального состояния водителей автотранспортных средств.....	126
4.3.3 Формирование базы знаний для определения фишинговых сайтов.....	134
<b>4.4 Выводы.....</b>	<b>141</b>
<b>Заключение.....</b>	<b>144</b>
<b>Список сокращений.....</b>	<b>146</b>
<b>Список литературы.....</b>	<b>147</b>
<b>Приложение 1 Свидетельства о регистрации программ для ЭВМ.....</b>	<b>175</b>
<b>Приложение 2 Акты о внедрении и использовании результатов диссертационного исследования .....</b>	<b>180</b>

## ВВЕДЕНИЕ

**Актуальность темы исследования.** В настоящее время в различных сферах человеческой деятельности для повышения эффективности оценки состояния объектов (проектных скважин на нефтяном месторождении, электронных писем, функциональных систем человека и т.д.) применяются интеллектуальные системы поддержки принятия решений (СППР). Эффективность их практического использования обеспечивается высокой скоростью обработки информации, адекватностью рекомендуемых решений исходным данным об оцениваемых объектах, а также снижением трудоемкости процесса принятия решений для человека. Преимущества применения СППР в задаче оценки состояния объектов обусловлены снижением влияния человеческого фактора на процесс принятия решений и автоматизацией этого процесса.

Основным компонентом интеллектуальной СППР является база знаний (БЗ). От ее адекватности (отражении закономерностей предметной области, формализованных в рамках конкретной модели представления знаний) и от механизма логического вывода во многом зависит способность СППР решать задачи по оценке состояния объектов и обосновывать рекомендуемые решения, особенно в условиях разнотипности входных данных (количественных и качественных) и их некомплектности (отсутствия части значений входных переменных при принятии решений). Для этого требуется выбор соответствующей модели представления знаний, разработка инструмента формирования БЗ в рамках выбранной модели, а также реализация алгоритма вывода.

Для получения точных и обоснованных решений в указанных условиях целесообразно использовать методы нечеткой логики и механизмы нечеткого логического вывода, а в качестве инструмента формирования баз знаний – нечеткие нейронные сети (ННС). Использование такого подхода позволяет получать решения в виде, пригодном для интерпретации человеком, учитывая лингвистическую неопределенность задачи принятия решений и нечеткий характер оцениваемых объектов.

Однако задачи выбора конкретного вида нечетких правил, учитывающего особенности анализируемых данных, разработки алгоритма логического вывода на правилах для обоснованной лингвистической оценки состояния объектов, а также построения нейронечеткой модели для формирования баз знаний являются нетривиальными и требуют соответствующего решения. Следовательно, тема диссертационного исследования, связанная с формированием и использованием в СППР нечетких баз знаний для оценки состояния объектов, является актуальной.

**Степень разработанности темы.** Исследованиям в области нечеткой логики и разработки СППР посвящены работы следующих ученых: Заде Л.А. [68; 232], Мамдани Е.А. [219; 220], Сугено М. [228; 229], Батыршина И.З. [23; 24], Недосекина А.О. [146; 223], Аверкина А.Н. [3; 4], Еремеева А.П. [63; 64], Ковалева С.М. [108; 217], Васильева В.И. [35; 36], Большакова А.А. [31; 198], Исмагилова И.И. [75; 213], Вульфина А.М. [41; 42], Гловы В.И. [46; 47], Аникина И.В. [7; 8], Ажмухамедова И.М. [5; 6], Симоновой Л.А. [164; 166], Новиковой С.В. [148; 149], Кремлевой Э.Ш. [116; 117], Колоденковой А.Е. [109; 216], Верещагиной С.С. [37] и др. Однако их исследования сосредоточены в основном на применении методов инженерии знаний и «ручного» подхода к формированию баз знаний, что требует привлечения экспертов и большой аналитической работы.

В настоящее время для автоматизации формирования баз знаний все чаще применяются инструментальные средства интеллектуального анализа данных, в частности, нечеткие нейронные сети, позволяющие в процессе обучения сформировать систему правил нечетко-продукционного типа. Исследованиям в этой области посвящены работы следующих ученых: Кордона О. [204], Хоффмана Ф. [208], Ротштейна А.П. [158], Финна В.К. [174; 175], Ярушкиной Н.Г. [190; 191], Наместникова А.М. [52; 143], Ходашинского И.А. [178; 179; 181], Бардамовой М.Б. [17], Комарцовой Л.Г. [111; 112], Емалетдиновой Л.Ю. [60; 61], Паклина Н.Б. [151] и др. Однако вопросы нейронечеткого формирования и эффективного использования в интеллектуальных СППР нечетких баз знаний для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных остаются не до конца исследованными.

Таким образом, **научная задача**, решаемая в диссертации, заключается в разработке нейронечеткой модели, методов и алгоритмов, а также комплекса программ формирования и использования нечетких баз знаний для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных. Ее решение имеет научную и практическую ценность для построения интеллектуальных систем поддержки принятия решений по оценке состояния объектов в различных предметных областях.

**Объект исследования:** нечеткие базы знаний интеллектуальных систем поддержки принятия решений по оценке состояния объектов.

**Предмет исследования:** методы и алгоритмы построения нейронечеткой модели, формирования баз знаний и их использования в системах поддержки принятия решений для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных.

**Цель диссертационной работы:** повышение точности оценки состояния объектов и улучшение интерпретируемости принимаемых решений в условиях разнотипности и некомплектности входных данных на основе разработки соответствующего математического, алгоритмического и программного обеспечения.

**Для достижения цели необходимо решить следующие задачи:**

1) предложить вид нечетко-продукционных правил и разработать алгоритм логического вывода на правилах, применяемые для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных;

2) разработать нейронечеткую модель, методы и алгоритмы ее построения, а также метод и алгоритм формирования базы знаний на основе построенной модели;

3) реализовать разработанные методы и алгоритмы в виде программного комплекса для формирования, использования баз знаний и проведения вычислительных экспериментов;

4) произвести апробацию разработанного математического, алгоритмического и программного обеспечения при решении практических задач по оценке состояния объектов в различных предметных областях.

**Методы исследования.** Для решения задач использованы методы сэмплинга данных, математической статистики, математического и нейронечеткого моделирования, нечеткой логики, генетической оптимизации, объектно-ориентированного программирования.

**Достоверность полученных результатов.** Предложенные в диссертационной работе методы, алгоритмы и оригинальная нейронечеткая модель теоретически обоснованы и не противоречат известным положениям других авторов. Достоверность научных положений, выводов и рекомендаций обусловлена математически строгим выполнением расчетов, подтверждена результатами проведенных исследований и практического использования.

**На защиту выносятся следующие результаты:**

1) вид нечетко-продукционных правил и алгоритм вывода на правилах для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных;

2) нейронечеткая модель формирования баз знаний, методы и алгоритмы ее построения для реализации этапов структурной и параметрической идентификации;

3) метод и алгоритм формирования базы знаний на основе построенной нейронечеткой модели;

4) комплекс программ, реализующий предложенные методы и алгоритмы, позволяющий формировать и использовать нечеткие базы знаний для оценки состояния объектов в условиях разнотипности и некомплектности входных данных.

**Научная новизна.**

1. Разработан новый математический метод моделирования состояния объектов на основе формирования нечетких баз знаний, *отличающийся* применением группового сэмплинга с замещением для получения однородных обучающей, тестовой и валидационной выборок данных, использованием нового вида нечетких правил, а также реализацией алгоритма вывода на правилах для обоснованной лингвистической оценки состояния объектов, *что позволяет* применять модель в условиях разнотипности и некомплектности входных данных (п. 1).



2. Разработан численный метод обучения нечеткой нейронной сети, основанный на классическом генетическом алгоритме, *отличающийся* новым способом кодирования значений параметров функций принадлежности в виде соответствующих хромосом, *что позволяет* осуществлять их настройку и по ее результатам определять значения весовых коэффициентов нейронечеткой модели (п. 3).

3. Разработан численный метод формирования базы знаний на основе нейронечеткой модели, *отличающийся* построением оптимальных маршрутов прохождения входных образов по слоям нечеткой нейронной сети, *что позволяет* генерировать нечетко-продукционные правила, соответствующие построенным маршрутам (п. 3).

4. Разработан комплекс программ, реализующий предложенные методы и алгоритмы, *отличающийся* новым составом программных модулей и их взаимодействием, *что позволяет* обеспечить требуемую функциональность по формированию и использованию нечетких баз знаний для оценки состояния объектов и проведения вычислительных экспериментов (п. 4).

**Теоретическая значимость** диссертационного исследования заключается в разработке нейронечеткой модели, методов и алгоритмов ее построения, метода и алгоритма формирования базы знаний, а также алгоритма логического вывода для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных.

**Практическая значимость** диссертации заключается в разработке комплекса программ, реализующего предложенные методы и алгоритмы и позволяющего обеспечить требуемую функциональность при формировании, использовании нечетких баз знаний для оценки состояния объектов и проведении вычислительных экспериментов.

**Соответствие диссертации паспорту научной специальности.** В диссертации разработан комплекс программ для формирования и использования нечетких баз знаний по оценке состояния объектов в условиях разнотипности и некомплектности входных данных. Предложен модифицированный вид нечетких правил и алгоритм вывода на правилах для обоснованной лингвистической оценки состояния объектов. Разработана оригинальная нейронечеткая модель, методы ее построения, а также метод и алгоритм формирования базы знаний на основе построенной модели.

Таким образом, результаты диссертационного исследования соответствуют следующим пунктам паспорта научной специальности 05.13.18 «Математическое моделирование, численные методы и комплексы программ»:

*П1. Разработка новых математических методов моделирования объектов и явлений:* для моделирования объектов предложен новый математический метод, основанный на построении нейронечеткой модели и позволяющий формировать базы знаний нечетко-продукционного типа для оценки состояния объектов в условиях разнотипности и некомплектности входных данных для принятия решений.

*П3. Разработка, обоснование и тестирование эффективных вычислительных методов с применением современных компьютерных технологий:* для получения нечетко-продукционных правил, пригодных для обоснованной лингвистической оценки состояния объектов, разработаны, обоснованы и протестированы методы обучения нечеткой нейронной сети и формирования базы знаний.

*П4. Реализация эффективных численных методов и алгоритмов в виде комплексов проблемно-ориентированных программ для проведения вычислительного эксперимента:* предложенные методы и алгоритмы реализованы в виде комплекса программ, на базе которого проведены вычислительные эксперименты по оценке эффективности разработанного математического и алгоритмического обеспечения и возможности его практического использования для решения поставленных задач.

**По теме диссертационного исследования опубликованы 23 работы, в том числе 8 статей в российских рецензируемых научных журналах, 5 статей в журналах, индексируемых в Scopus и WoS, 10 статей – в других журналах и материалах конференций. Получено 4 свидетельства о регистрации программ для ЭВМ.**

**Основные результаты диссертационной работы докладывались и обсуждались на следующих конференциях:**

- международных научно-технических конференциях «Проблемы техники и технологий телекоммуникаций ПТиТТ-2014» и «Оптические технологии в телекоммуникациях ОТТ-2014» (Казань, 2014);

- седьмой межрегиональной научно-практической конференции «Информационная безопасность и защита персональных данных. Проблемы и пути их решения» (Брянск, 2015);
- одиннадцатой международной Четаевской конференции, посвященной 115-летию со дня рождения Н.Г. Четаева и памяти академика АН РТ Т.К. Сиразетдинова «Аналитическая механика, устойчивость и управление (Казань, 2017);
- международных форумах «Kazan Digital Week» (Казань, 2020, 2021);
- тридцать четвертой международной научной конференции «Математические методы в технике и технологиях – ММТТ-34» (Санкт-Петербург, 2021);
- международной молодежной научной конференции «XXV Туполевские чтения (школа молодых ученых)» (Казань, 2021);
- десятой национальной научно-практической конференции с международным участием «Моделирование энергоинформационных процессов» (Воронеж, 2021).

**Реализация результатов работы. Результаты исследования:**

- внедрены в опытно-промышленную эксплуатацию в бизнес-процесс управления инвестиций центра обслуживания бизнеса ПАО «ТатНефть» в виде интеллектуальной системы поддержки принятия решений для формирования оптимальной программы геолого-технических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяных месторождений;
- использованы в центре интеллектуальных транспортных систем ГБУ «Безопасность дорожного движения» при формировании базы знаний для системы поддержки принятия решений по оценке функционального состояния водителей автотранспортных средств при прохождении процедуры предрейсового медицинского осмотра;
- внедрены в учебный процесс ФГБОУ ВО «Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ» и используются при изучении дисциплин «Интеллектуальные системы обеспечения информационной безопасности», «Системы искусственного интеллекта»,

«Нечеткие нейросети и генетические алгоритмы», «Системы поддержки принятия решений».

**Структура и объем диссертации.** Диссертация изложена на 185 страницах машинописного текста, содержит 55 рисунков, 34 таблицы, состоит из введения, четырех глав, заключения, списка использованной литературы из 232 наименований на 28 страницах и двух приложений на 11 страницах.

**Сведения о личном вкладе автора.** Постановка научной задачи, содержание диссертации и все представленные в ней результаты получены лично автором. Подготовка к публикации некоторых научных статей, а также получение и регистрация результатов интеллектуальной деятельности проводились совместно с соавторами, причем вклад диссертанта определяющий.

**Во введении** обоснована актуальность темы диссертационного исследования, приведены основные научные положения и результаты.

**В первой главе** рассматривается задача оценки состояния объекта. Для ее решения обосновывается необходимость использования интеллектуальных СППР. Рассматривается структура и типовой состав таких систем. Указывается на необходимость анализа данных, формирования базы знаний и организации логического вывода на ее правилах. Анализируются модели представления знаний. При формировании базы знаний обосновывается целесообразность использования нечетко-продукционной модели. В качестве инструмента ее формирования предлагается использование нечеткой нейронной сети. Анализируются особенности исходных данных при решении задачи оценки состояния объектов. В результате анализа формулируются требования к виду нечетких правил, алгоритму вывода на правилах и к ННС. Ставится задача формирования и использования баз знаний для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных.

**Во второй главе** описываются основные виды нечетких правил, используемых для оценки состояния объектов, приводится их сравнительный анализ. Для обработки разнотипных входных данных, ранжирования получаемых решений и выбора наиболее предпочтительного из них с возможностью однозначной интер-

претации предлагается новый вид правил и алгоритм вывода на правилах. Вводится критерий его применения в условиях некомплектности входных данных для принятия решений. Описываются этапы построения нейронечеткой модели, связанные с заданием состава слоев ННС, определением их функциональности и состава нейронов в каждом слое, формированием выборок данных, а также обучением, тестированием и оценкой ее обобщающей способности. Ставится задача разработки методов и алгоритмов ее построения и формирования БЗ для лингвистической оценки состояния объектов.

**В третьей главе** предлагается метод сэмплинга данных, используемых для обучения, тестирования и валидации нечеткой нейронной сети, метод расчета значений ее весовых коэффициентов. Описываются методы и алгоритмы построения нейронечеткой модели: метод и алгоритм обучения ННС, генетический алгоритм настройки значений параметров функций принадлежности, алгоритм оценки обобщающей способности нейронечеткой модели. Кроме того, раскрываются вопросы определения начальных значений параметров функций принадлежности, а также выбора и кодирования параметров задачи в ГА. Описываются метод и алгоритм формирования базы знаний на основе построенной модели.

**В четвертой главе** описывается разработанный программный комплекс: общие характеристики, структура и состав компонентов, графический интерфейс, а также алгоритм использования для поддержки принятия решений. Приводятся результаты проведенных исследований: влияния числа градаций, формы функций принадлежности и значений параметров генетического алгоритма на обучение ННС. Даются рекомендации по построению нейронечетких моделей, описываются сформированные базы знаний, приводится сравнение их точности с точностью других методов классификации. Приводятся результаты апробации при формировании баз знаний для подбора геолого-технических мероприятий на нефтяном месторождении, для оценки функционального состояния водителей автотранспортных средств и для определения фишинговых сайтов.

**В заключении** сформулированы основные научные и практические результаты работы, намечены направления перспективных исследований.

**В приложении 1** представлены свидетельства о регистрации программ для ЭВМ.

**В приложении 2** представлены акты о внедрении и использовании результатов диссертационного исследования.

Диссертация выполнена на кафедре систем информационной безопасности федерального государственного бюджетного образовательного учреждения высшего образования «Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ» (КНИТУ-КАИ).

Диссертационная работа выполнялась в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2014-2020 годы» по выполнению научно-исследовательской работы «Создание программного комплекса поддержки принятия решений по формированию множества предпочтительных вариантов геолого-технических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяного месторождения» по теме «Создание технологии долгосрочного планирования инвестиций для эффективной разработки нефтяных месторождений на основе высокопроизводительных вычислений и машинного обучения» (2019-2020 гг., рег. № АААА-А19-119121990030-0).

Автор выражает искреннюю благодарность своему научному руководителю, доктору технических наук, профессору Исмагилову Ильясу Идрисовичу за оказанную поддержку и ценные советы при проведении исследований.

# **1 Анализ предметной области и постановка задачи формирования и использования баз знаний для систем поддержки принятия решений по оценке состояния объектов**

*В этой главе рассматривается задача оценки состояния объекта. Для ее решения обосновывается необходимость использования интеллектуальных СППР. Рассматривается структура и типовой состав таких систем. Указывается на необходимость анализа данных, формирования базы знаний и организации логического вывода на ее правилах. Анализируются модели представления знаний. При формировании базы знаний обосновывается целесообразность использования нечетко-продукционной модели. В качестве инструмента ее формирования предлагается использование нечеткой нейронной сети. Анализируются особенности исходных данных при решении задачи оценки состояния объектов. В результате анализа формулируются требования к виду нечетких правил, алгоритму вывода на правилах и к ННС. Ставится задача формирования и использования баз знаний для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных.*

## **1.1 Задача оценки состояния объектов**

В различных сферах человеческой деятельности часто возникает необходимость решения задач оценки состояния объектов [84; 130; 136; 150; 160; 206]. В таких задачах у каждого объекта имеется набор возможных дискретных состояний, одно из которых соответствует его текущему состоянию. Объектами могут быть проектные скважины на нефтяном месторождении [70; 145; 226] (оценка целесообразности бурения скважин), электронные письма [80; 88; 101] (оценка принадлежности письма к категории «спам» / «не спам»), функциональные системы человека [21; 159; 171] (оценка состояния усталости человека), сетевой трафик [72; 102; 196] (оценка вредоносности сетевого трафика) и т.д. Оценка текущего состояния объекта и определение достоверности этой оценки составляют предмет решаемой задачи [86].

На рисунке 1.1 представлена обобщенная схема решения этой задачи.

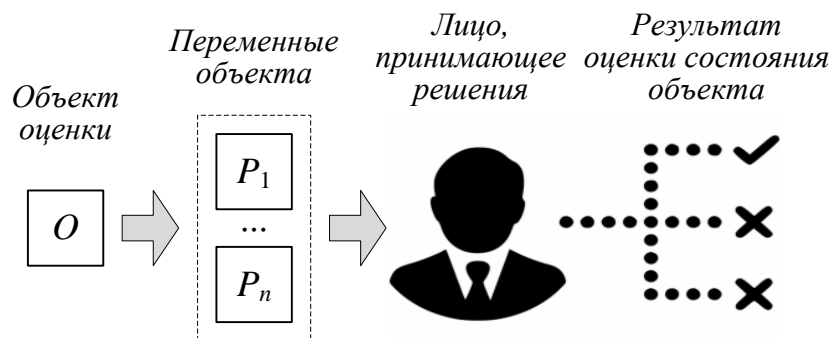


Рисунок 1.1 – Схема решения задачи оценки состояния объекта

В этой задаче ключевую роль играет человек – лицо, принимающее решение (ЛПР) [29; 91; 177]. По значениям переменных объекта  $P_1, \dots, P_n$  он формирует результат оценки его состояния, выбирая оптимальное решение из множества допустимых. При решении данной задачи ЛПР опирается только на свое мнение, подкрепленное знаниями, опытом и интуицией [182].

Однако такая схема оценки состояния объекта сопряжена с рядом трудностей в силу наличия человеческого фактора [91]. В качестве ЛПР может выступать как квалифицированный, так и неквалифицированный пользователь, ЛПР может ошибаться из-за невнимательности, усталости и т.д. Все это может приводить к ошибкам в оценке состояния объекта. Поэтому в настоящее время для повышения эффективности этого процесса актуально использование интеллектуальных СППР [38; 83; 132; 186]. Рассмотрим особенности этого класса интеллектуальных систем, их преимущества, структуру и предъявляемые к ним требования.

## 1.2 Применение интеллектуальных систем поддержки принятия решений для оценки состояния объектов

Существует большое количество примеров использования интеллектуальных систем в различных предметных областях [56; 96; 133; 197; 218; 221; 227; 231]. Как правило, такие системы осуществляют функцию поддержки процесса принятия решений для человека. Применительно к задаче оценки состояния объекта рекомендации, поступающие от СППР, помогают ЛПР принять правильное решение, не ошибиться в выборе текущего состояния объекта [85; 91].



На рисунке 1.2 показана схема оценки состояния объекта с применением интеллектуальной СППР.

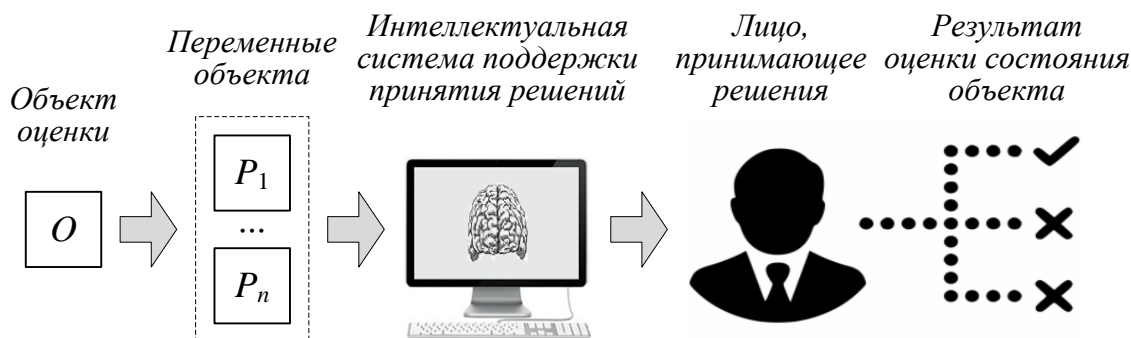


Рисунок 1.2 – Схема решения задачи оценки состояния объекта с использованием интеллектуальной системы поддержки принятия решений

Система помогает ЛПР принимать решения [105]. В данном случае она производит анализ значений переменных объекта  $P_1, \dots, P_n$ , формирует результаты оценки его состояния, предлагает ЛПР варианты решений и выделяет наиболее предпочтительный из них. Человек, опираясь на предложенные системой варианты решений, принимает окончательное решение. Следовательно, в данном случае ЛПР учитывает варианты решений, предложенные системой.

Основными преимуществами применения СППР в задаче оценки состояния объекта являются следующие [91; 115]:

- 1) высокая скорость обработки информации при принятии решений;
- 2) адекватность рекомендуемых решений исходным данным, характеризующим оцениваемый объект;
- 3) снижение трудоемкости процесса принятия решений для человека.

Высокая скорость обработки информации достигается за счет автоматизации вычислений при анализе значений переменных объекта, выполнении процедуры логического вывода, сопоставлении входных данных с имеющимися в базе знаний системы правилами принятия решений [91].

Адекватность рекомендуемых решений исходным данным об объекте оценки обусловлена тем, что интеллектуальная система всегда работает по одному алгоритму логического вывода и применяет для этого заранее сформированную базу знаний, адекватность которой обеспечивается в процессе ее формирования [91].

Снижение трудоемкости выражается в снижении интеллектуальной нагрузки на человека в процессе принятия решений [71; 91].

Таким образом, преимущества применения интеллектуальных СППР в задаче оценки состояния объектов [83] обусловлены главным образом снижением влияния человеческого фактора и трудоемкости процесса принятия решений за счет его автоматизации.

При этом к СППР предъявляются следующие основные требования [91]:

- 1) высокая точность оценки состояния объектов;
- 2) умение представлять рекомендуемые решения в лингвистическом виде, пригодном для дальнейшего анализа ЛПР.

Высокая точность оценки состояния объектов должна достигаться за счет формирования адекватных баз знаний и применения эффективных алгоритмов логического вывода [85]. Под эффективностью алгоритма вывода следует понимать результативность использования в интеллектуальных СППР нечетких баз знаний для обоснованной лингвистической оценки состояния объектов. При этом точность такой оценки должна быть не ниже точности оценки у ЛПР.

Умение обосновывать рекомендуемые решения необходимо для их понимания и интерпретации человеком. Это, в свою очередь, повышает уровень доверия ЛПР к предлагаемым системой решениям, что позволяет сделать выбор окончательного решения по оценке состояния объекта более обоснованным [91].

Рассмотрим структуру и типовой состав интеллектуальных СППР.

### **1.3 Структура и состав систем поддержки принятия решений**

Любая интеллектуальная СППР имеет структуру и состав компонентов, зависящих от особенностей решаемой задачи, от используемых механизмов ее решения, от реализуемых стратегий получения и использования знаний и т.д. Однако в большинстве интеллектуальных СППР имеется обязательный набор компонентов, предназначенных для формирования базы знаний и реализации логического вывода для решения поставленных задач [84].

На рисунке 1.3 представлен пример типовой структуры и состава компонентов СППР [84].

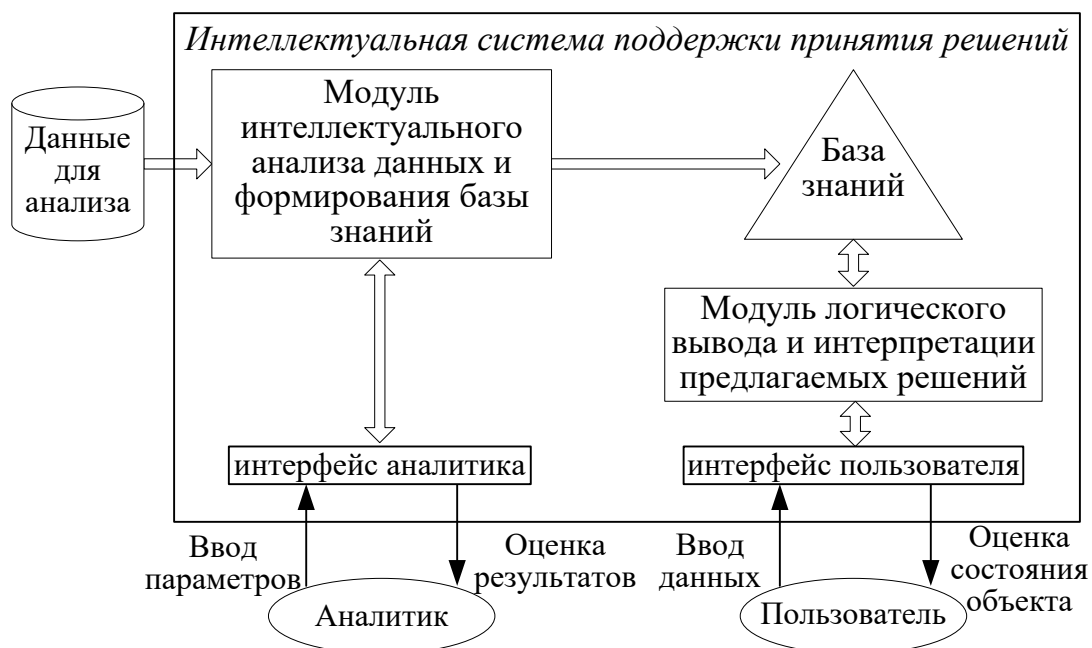


Рисунок 1.3 – Пример структуры системы поддержки принятия

Система включает следующие основные модули [84]:

- 1) модуль интеллектуального анализа данных и формирования БЗ;
- 2) модуль логического вывода и интерпретации предлагаемых решений.

В первом модуле реализуются механизмы интеллектуального анализа исходных данных, в результате которого происходит формирование базы знаний – основного компонента СППР. Сформированная база знаний используется во втором модуле для реализации механизмов логического вывода и интерпретации предлагаемых системой решений.

Система также имеет интерфейсы взаимодействия с аналитиком и с пользователями. Роль аналитика заключается в управлении процессами интеллектуального анализа данных и формирования базы знаний. Он должен корректировать значения параметров (например, выбирать методы анализа данных) и оценивать получаемые результаты на каждом этапе указанных процессов.

Роль пользователя СППР заключается в ее использовании непосредственно для решения задач, связанных с оценкой состояния объектов. Вводя данные об объекте, пользователь получает интерпретируемую оценку его состояния, на основе которой в дальнейшем принимает окончательное решение.

Таким образом, для построения интеллектуальной СППР требуется проанализировать данные, сформировать БЗ и реализовать вывод на ее правилах с интерпретацией предлагаемых решений [85]. Необходимость анализа данных, формирования БЗ и выполнения логического вывода в СППР потребовали выбора соответствующей модели представления знаний [97].

#### **1.4 Анализ и выбор модели представления знаний для системы поддержки принятия решений**

Как было сказано в предыдущем разделе, база знаний является основным компонентом СППР [84]. При этом от модели представления знаний зависит результативность процессов формирования и использования базы знаний [84].

В настоящее время в интеллектуальных системах используются различные модели представления знаний [27; 43; 76; 81; 165; 172]: логические, фреймовые, семантические, продукционные, нечетко-продукционные. Выбор конкретной модели, как правило, определяется особенностями решаемой задачи.

При построении СППР важными критериями выбора модели представления знаний являются [30; 110]:

- 1) гибкость модели;
- 2) обоснованность и возможность однозначной интерпретации получаемых решений;
- 3) наличие инструментов формирования БЗ в рамках выбранной модели [82].

Гибкость модели важна с точки зрения эффективности ее применения для описания знаний и закономерностей в рассматриваемой предметной области. Обоснованность и возможность однозначной интерпретации получаемых решений важны для пользователя СППР, для повышения его доверия к предлагаемым системой решениям. Наличие инструментов формирования базы знаний важно для автоматизации этого процесса с минимальным участием человека (аналитика).

В таблице 1.1 показано сравнение моделей представления знаний [82].

Таблица 1.1 – Сравнение моделей представления знаний

Критерии сравнения моделей Модель представления знаний	Гибкость при описании закономерностей	Обоснованность и интерпретируемость решений	Наличие инструментов формирования БЗ
Логическая	низкая	низкая	нет
Фреймовая	средняя	высокая	нет
Семантическая	средняя	высокая	нет
Продукционная	средняя	высокая	да
Нечетко-продукционная	высокая	высокая	да

Логическая модель представления знаний получила минимальные оценки по указанным критериям из-за ее формального характера. Знания и закономерности в такой модели описываются в виде набора аксиом, что не позволяет эффективно обосновывать и интерпретировать получаемые решения. Кроме того, для такой модели отсутствуют эффективные инструменты автоматизированного построения БЗ. Для использования модели требуется непосредственное участие человека [85].

Фреймовая модель обладает достаточной гибкостью и высокой интерпретирующей способностью. Однако эта модель не имеет развитых инструментов формирования БЗ без участия человека [85].

Семантическая модель представления знаний обладает теми же достоинствами, недостатками и ограничениями по указанным критериям, что и фреймовая модель [2], что сдерживает ее использование в СППР [84].

Продукционная модель представления знаний, по сравнению с предыдущими, наиболее полно соответствует указанным критериям. Однако ее гибкость ограничена четким характером моделируемых объектов.

В нечетких условиях применяется нечетко-продукционная модель, обладающая теми же достоинствами, что и продукционная модель. При этом она характеризуется максимальной гибкостью, так как позволяет учитывать как четкие, так и нечеткие особенности моделируемых объектов. Кроме того, нечетко-продукционная модель позволяет максимально точно интерпретировать и обосновывать получаемые решения и имеет эффективные инструменты формирования базы знаний [82].

Таким образом, для формирования БЗ целесообразно использовать нечетко-продукционную модель. При этом в качестве инструмента формирования БЗ целесообразно использовать нечеткую нейронную сеть [192], позволяющую формировать систему правил для обоснованной лингвистической оценки состояния объектов [84]. Рассмотрим особенности использования ННС [48].

### **1.5 Использование нечетких нейронных сетей для формирования баз знаний**

Особенностью всех ННС является их гибридный характер [95; 157]. Подобно классическим нейронным сетям они требуют обучения. Обученная ННС (нейронечеткая модель) отражает логику функционирования нечетких систем и формирует базу знаний нечетко-продукционного типа. Правила базы знаний могут быть использованы для решения поставленных задач, в данном случае – для задачи оценки состояния объектов [91; 212].

На рисунке 1.4 показана схема формирования базы знаний на основе анализа данных с использованием ННС [48].

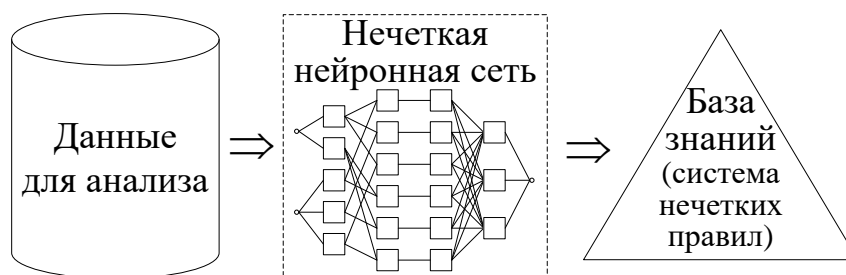


Рисунок 1.4 – Схема формирования базы знаний

Как видно из рисунка, для реализации представленной схемы необходимо иметь набор данных, характеризующих решаемую задачу [91].

Данные для анализа и обучения ННС могут быть представлены в виде, как показано в таблице 1.2 [91; 189].

Таблица 1.2 – Общий вид таблицы данных для анализа

№ п/п	Входные переменные					Выходная переменная
	1	2	3	...	<i>n</i>	
1						
2						
3						
...						
<i>N</i>						

Таблица данных объемом  $N$  содержит  $n$  входных и одну выходную переменные, характеризующие оцениваемый объект. Каждая строка такой таблицы соответствует определенному объекту и содержит значения входных и выходной переменных. Значением выходной переменной является конкретное состояние объекта, в котором он находился при указанных условиях (конкретных значениях входных переменных) [91].

Следовательно, имея характеристики  $N$  объектов (или  $N$  наблюдений за объектами), можно сформировать исходную таблицу данных, пригодных для обучения ННС и формирования базы знаний [84]. При этом ННС выступает в качестве инструмента интеллектуального анализа данных и формирования БЗ [203].

Таким образом, для формирования БЗ требуется:

- 1) определить особенности данных, характеризующих оцениваемые объекты и используемых в процессе принятия решений;
- 2) предложить вид правил с учетом особенностей данных;
- 3) разработать алгоритм вывода на правилах для обоснованной лингвистической оценки состояния объектов с учетом особенностей данных и предложенного вида правил [83];
- 4) разработать структуру ННС с учетом особенностей данных и алгоритма логического вывода;
- 5) обучить ННС и сформировать систему правил выбранного вида.

Рассмотрим особенности данных, характеризующих оцениваемые объекты [84].

## 1.6 Анализ особенностей данных для оценки состояния объектов

Для оценки состояния объектов используются входные данные, характеризующие оцениваемые объекты. Эти данные могут быть получены из различных источников и не всегда обладают свойством полноты. Поэтому для эффективного решения поставленной задачи необходимо понимать и учитывать их особенности (см. рис. 1.5) [91].

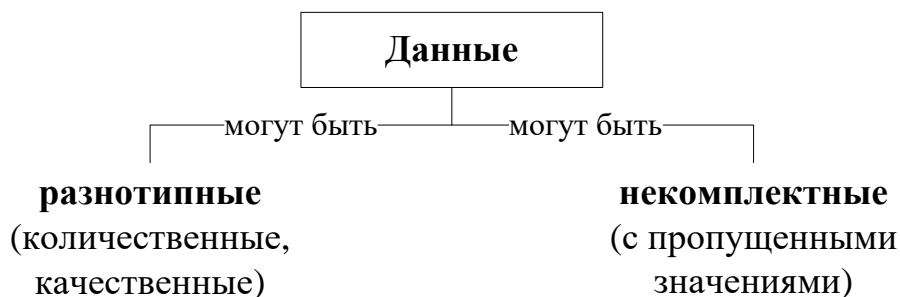


Рисунок 1.5 – Особенности данных для оценки состояния объектов

Данные, описывающие моделируемые объекты и используемые для принятия решений по оценке их состояния, могут быть разнотипными и некомплектными. Разнотипность данных означает [33], что часть входных переменных в таблице для анализа являются количественными и принимают непрерывные или дискретные значения, лежащие на числовой оси [128]. Остальные входные переменные являются качественными и принимают категориальные (номинальные, порядковые или бинарные) значения [167].

Некомплектность данных, используемых для принятия решений, означает ситуацию, когда в процессе оценки состояния объекта на входе СППР появляются данные с пропущенными значениями [91; 187]. Отсутствие части входных данных для принятия решений возможно, например, при возникновении ошибок в процессе их передачи по каналам связи, при неработающих или некорректно работающих датчиках, отвечающих за получение данных, при наличии человеческого фактора во время «ручного» ввода данных и др. В любом случае наличие пропусков в данных предъявляет дополнительные требования к их обработке.

Рассмотренные особенности входных данных необходимо учитывать при формировании соответствующих требований [82].



## **1.7 Формирование требований к виду нечетких правил, алгоритму вывода на правилах и нечеткой нейронной сети**

Исходя из условий решаемой задачи при выборе вида правил, разработке алгоритма вывода на правилах и построении нейронечеткой модели должны учитываться следующие требования [100]:

1) antecedentes нечетко-продукционных правил для обработки разнотипных данных должны включать в себя как количественные, так и качественные входные переменные;

2) алгоритм логического вывода должен учитывать особенности предложенного вида правил и обосновывать предлагаемые решения в лингвистическом виде, пригодном для анализа ЛПР;

3) должна быть обеспечена возможность применения алгоритма логического вывода в условиях некомплектности входных данных для принятия решений;

4) нечеткая нейронная сеть должна реализовывать этапы алгоритма логического вывода и в результате обучения на разнотипных данных формировать базу знаний, как систему правил выбранного вида.

Исходя из указанных требований, рассмотрим постановку решаемой задачи.

## **1.8 Постановка задачи формирования и использования баз знаний для оценки состояния объектов**

Таким образом, с учетом вышеописанного, для реализации технологии нейронечеткого моделирования, формирования и использования БЗ для обоснованной лингвистической оценки состояния объектов в условиях разнотипности и некомплектности входных данных необходимо:

- 1) предложить вид нечетких правил с учетом требования (1);
- 2) разработать алгоритм вывода на правилах с учетом требования (2);
- 3) разработать критерий применения алгоритма логического вывода, удовлетворяющий требованию (3);

4) разработать нейронечеткую модель, удовлетворяющую требованию (4), а также методы ее построения и формирования базы знаний;

5) разработать комплекс программ для формирования и использования БЗ по оценке состояния объектов.

Решение указанных задач отражено в следующих главах диссертации.

## 1.9 Выводы

Для повышения эффективности решения задачи оценки состояния объектов целесообразно использовать интеллектуальные СППР. Эффективность достигается за счет высокой скорости обработки информации при принятии решений, адекватности рекомендуемых решений исходным данным, характеризующим оцениваемые объекты, а также снижения трудоемкости процесса принятия решений для человека. При этом к СППР предъявляются такие требования, как высокая точность оценки состояния объектов и обоснованность рекомендуемых решений.

В состав интеллектуальных СППР должен входить набор компонентов для формирования базы знаний и реализации механизмов логического вывода. Анализ моделей представления знаний показал, что для оценки состояния объектов целесообразно использовать нечетко-продукционную модель. В качестве инструмента формирования базы знаний целесообразно использовать ННС.

Для формирования БЗ на основе ННС требуется определить особенности анализируемых данных, предложить вид нечетких правил с учетом особенностей данных, разработать алгоритм вывода на правилах с учетом особенностей данных и предложенного вида правил, разработать структуру ННС с учетом особенностей данных и алгоритма логического вывода, а также обучить ННС и сформировать систему правил выбранного вида.

Проведенный анализ показал, что данные, описывающие моделируемые объекты и используемые для принятия решений по оценке их состояния, могут быть разнотипными и некомплектными. Исходя из этого, а также учитывая особенности решаемой задачи, сформулированы следующие требования:

1) antecedentes правил для обработки разнотипных данных должны включать в себя как количественные, так и качественные входные переменные;

2) алгоритм логического вывода должен учитывать особенности предложенного вида правил и обосновывать предлагаемые решения в виде, пригодном для лингвистической интерпретации ЛПР;

3) должна быть обеспечена возможность применения алгоритма логического вывода в условиях некомплектности входных данных для принятия решений;

4) нечеткая нейронная сеть должна реализовывать этапы алгоритма логического вывода и в результате обучения на разнотипных данных формировать базу знаний, как систему правил выбранного вида.

С учетом указанных требований произведена постановка задачи формирования и использования баз знаний для оценки состояния объектов.

## 2 Разработка нейронечеткой модели формирования баз знаний для оценки состояния объектов

*В этой главе описываются основные виды нечетких правил, используемых для оценки состояния объектов, приводится их сравнительный анализ. Для обработки разнотипных входных данных, ранжирования получаемых решений и выбора наиболее предпочтительного из них с возможностью однозначной интерпретации предлагается новый вид правил и алгоритм вывода на правилах. Вводится критерий его применения в условиях некомплектности входных данных для принятия решений. Описываются этапы построения нейронечеткой модели, связанные с заданием состава слоев ННС, определением их функциональности и состава нейронов в каждом слое, формированием выборок данных, а также обучением, тестированием и оценкой ее обобщающей способности. Ставится задача разработки методов и алгоритмов ее построения и формирования БЗ для лингвистической оценки состояния объектов.*

### 2.1 Сравнительный анализ основных видов нечетко-продукционных правил для оценки состояния объектов

Как было показано в предыдущей главе, при выборе вида правил, составляющих БЗ, необходимо рассматривать нечеткие продукции. При этом необходимо учесть дискретный характер решаемой задачи, так как предполагается, что число возможных состояний объекта дискретно и счетно [113]. Рассмотрим основные виды нечетко-продукционных правил, пригодные для решения задачи.

Ниже представлена простая нечеткая продукция [22; 51]:

$$\text{If } x_1 = \tilde{A}_1 \ \& \ x_2 = \tilde{A}_2 \ \& \ \dots \ x_i = \tilde{A}_i \ \& \ \dots \ x_n = \tilde{A}_n \ \text{Then } y = B, \quad (2.1)$$

где  $x_i$  ( $i=1..n$ ) – входные переменные;

$\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$  – нечеткие множества, выступающие в роли нечетких ограничений на значения входных переменных;

$\mu_{\tilde{A}_i}(x_i)$  – функции принадлежности (ФП) значений  $x_i$  к  $\tilde{A}_i$ ;

$y$  – выходная переменная;

$B$  – состояние объекта.

Правила (2.1) обладают однозначной интерпретируемостью, но позволяют учитывать на входе значения только количественных входных переменных. Это ограничивает возможности их практического использования, в частности, для обработки разнотипных входных данных.

Разновидностью нечетко-продукционных правил вида (2.1) является взвешенная нечеткая продукция [1; 173]:

$$\text{If } x_1 = \tilde{A}_1 \& x_2 = \tilde{A}_2 \& \dots x_i = \tilde{A}_i \& \dots x_n = \tilde{A}_n \text{ Then } y = B [CF], \quad (2.2)$$

где  $x_i (i=1..n)$  – входные переменные;

$\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$  – нечеткие множества;

$\mu_{\tilde{A}_i}(x_i)$  – функции принадлежности;

$y$  – выходная переменная;

$B$  – состояние объекта;

$CF \in [0;1]$  – вес правила.

Использование весов правил делает нечеткие продукции более универсальными, позволяя их ранжировать в процессе логического вывода. В случае срабатывания нескольких правил при оценке состояния объекта в качестве решающего можно выбрать то, у которого вес больше.

Взвешенные нечетко-продукционные правила получили свое развитие в виде модифицированной нечеткой продукции [46; 47]:

$$\text{If } x_1 = \tilde{A}_1(w_1) \& x_2 = \tilde{A}_2(w_2) \& \dots x_i = \tilde{A}_i(w_i) \& \dots x_n = \tilde{A}_n(w_n) \text{ Then } y = B [CF], \quad (2.3)$$

где  $x_i (i=1..n)$  – входные переменные;

$\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$  – нечеткие множества;

$\mu_{\tilde{A}_i}(x_i)$  – функции принадлежности;

$w_i$  – веса условий в правилах;

$y$  – выходная переменная;

$B$  – состояние объекта;

$CF \in [0;1]$  – вес правила.

Использование весов условий позволяет применять нечеткие продукции в условиях неполноты входных данных. Однако при этом возникают трудности с интерпретацией таких правил, с ее неоднозначностью. При этом у таких правил сохраняется ограничение на обработку разнотипных входных данных.

Для снятия этого ограничения используют усовершенствованную нечеткую продукцию [81; 82]:

$$\text{If } x_1 = \vec{A}_1(w_1) \& x_2 = \vec{A}_2(w_2) \& \dots x_i = \vec{A}_i(w_i) \& \dots x_n = \vec{A}_n(w_n) \text{ Then } y = B[CF], \quad (2.4)$$

где  $x_i (i=1..n)$  – входные переменные;

$\vec{A}_i$  – ограничения на значения входных переменных, которые могут быть как четкими ( $\vec{A}_i = A_i$  для обработки качественных значений входных переменных), так и нечеткими ( $\vec{A}_i = \tilde{A}_i$  для обработки количественных значений);

$A_i$  – четкие значения входных переменных;

$\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$  – нечеткие множества;

$\mu_{\tilde{A}_i}(x_i)$  – функции принадлежности;

$w_i$  – веса условий в правилах;

$y$  – выходная переменная;

$B$  – состояние объекта;

$CF \in [0;1]$  – вес правила.

Использование в таких правилах двойственного ограничения на значения входных переменных позволяет обрабатывать разнотипные входные данные.

Разновидностью нечетко-продукционных правил вида (2.4) является смешанная нечеткая продукция [37; 62]:

$$\text{If } x_1 = A_1 \& x_2 = \tilde{A}_2 \& \dots x_n = \tilde{A}_n([x_{n_{\min}}, x_{n_{\max}}]) \text{ Then } y = B[Rung] \quad (2.5)$$

где  $x_i (i=1..n)$  – входные переменные;

$A_i$  – четкие значения входных переменных;

$\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$  – нечеткие множества;

$\mu_{\tilde{A}_i}(x_i)$  – функции принадлежности;

$\tilde{A}_i([x_{i\min}, x_{i\max}])$  – значения входных переменных, представленные в виде интервалов;

$x_{i\min}, x_{i\max}$  – минимальное и максимальное значение  $i$ -й входной переменной;

$y$  – выходная переменная;

$B$  – четкое и/или нечеткое значение выходной переменной;

$Rung \in [1, 2, \dots, c]$  – важность правила;

$c$  – количество продукционных правил в базе знаний.

Особенностью правил вида (2.5) является то, что они позволяют комбинировать четкие и нечеткие значения переменных в условиях правил и учитывать важность (значимость) правил на основе их ранжирования. При этом значение важности каждого правила определяется лишь количеством входящих в него переменных и не зависит от правильности самого правила, результатов его экспертной оценки или статистической значимости, что может приводить к снижению точности получаемых решений и неоднозначности их интерпретируемости.

Анализ характеристик рассмотренных нечетко-продукционных правил с учетом особенностей решаемой задачи и входных данных для принятия решений позволяет сформулировать следующие критерии выбора вида правил:

- возможность обработки в антецедентах правил разнотипных входных данных;
- интерпретируемость правил и получаемых решений;
- возможность учета весов правил для ранжирования предлагаемых решений.

В таблице 2.1 представлены результаты сравнения видов правил.

Таблица 2.1 – Сравнение видов нечетких правил

Критерии \ Вид правила	Обработка разнотипных входных данных	Интерпретируемость	Возможность учета веса правила при оценке состояния объектов
Простая нечеткая продукция	-	однозначная	-
Взвешенная нечеткая продукция	-	однозначная	+
Модифицированная нечеткая продукция	-	неоднозначная	+
Усовершенствованная нечеткая продукция	+	неоднозначная	+
Смешанная нечеткая продукция	+	неоднозначная	+

Рассмотренные виды правил в полной мере не соответствуют указанным критериям. Следовательно, актуально предложить новый вид правил для обработки разнотипных входных данных, позволяющий однозначно интерпретировать и ранжировать получаемые решения.

## 2.2 Предложенный вид нечетко-продукционных правил

В диссертации для оценки состояния объектов предложен следующий вид нечетко-продукционных правил [91]:

$$\text{If } x_1 = \vec{A}_1 \ \& \ x_2 = \vec{A}_2 \ \& \dots \ x_i = \vec{A}_i \ \& \dots \ x_n = \vec{A}_n \ \text{Then } y = B [CF], \quad (2.6)$$

где  $x_i$  ( $i=1..n$ ) – входные переменные;

$\vec{A}_i$  – четкие ( $\vec{A}_i = A_i$ ) и нечеткие ( $\vec{A}_i = \tilde{A}_i$ ) ограничения на значения входных переменных, позволяющие обрабатывать, соответственно, качественные и количественные входные данные;

$A_i$  – четкие значения входных переменных;

$\tilde{A}_i = \{x_i, \mu_{\tilde{A}_i}(x_i)\}$  – нечеткие множества;

$\mu_{\tilde{A}_i}(x_i)$  – функции принадлежности;

$y$  – выходная переменная;

$B$  – состояние объекта;

$CF \in [0;1]$  – вес правила.

Как можно легко заметить, по своей структуре правила вида (2.6) больше всего соответствуют правилам вида (2.4). Отличие заключается в отсутствии в правилах предложенного вида весов условий. Как было сказано выше, использование таких весов позволяет применять нечеткие продукции в условиях неполноты входных данных. При этом ухудшается интерпретируемость самих правил и, соответственно, получаемых на их основе решений. Для принятия решений в случае некомплектности входных данных целесообразнее использовать не веса условий, а дополнительный критерий применимости алгоритма логического вывода. Указанный критерий будет рассмотрен ниже.



Таким образом, особенности и одновременно достоинства нечетко-продукционных правил вида (2.6) заключаются в том, что [91]:

- 1) позволяют обрабатывать разнотипные входные данные;
- 2) поддаются однозначной лингвистической интерпретации;
- 3) характеризуются весом  $CF$  (для оценки состояния объекта).

### 2.3 Алгоритм логического вывода на правилах

Рассмотрим алгоритм вывода, учитывающий особенности предложенных правил и позволяющий обосновывать предлагаемые решения в виде, пригодном для лингвистической интерпретации ЛПР.

Пусть  $X^* = \{x_1^*, \dots, x_i^*, \dots, x_n^*\}$  – множество входных значений, характеризующих оцениваемый объект и пусть  $S = \{R_1, \dots, R_p, \dots, R_N\}$  – база знаний, состоящая из  $N$  нечетко-продукционных правил вида (2.6) и соответствующая множеству входных значений  $X^*$ . Следует отметить, что база знаний должна существовать и быть адекватной. При этом ее соответствие множеству входных значений означает, что все правила базы знаний имеют соответствующее число  $n$  и соответствующий состав  $x_1, \dots, x_i, \dots, x_n$  входных переменных.

В алгоритме логического вывода для правил базы знаний рассчитываются следующие характеристики [90]:

-  $T \in [0;1]$  – степень истинности антецедента правила:

$$T = \min_i (\mu_{\tilde{A}_i}(x_i^*)), \quad (2.7)$$

где  $x_i^*$ , ( $i=1..n$ ) – четкие входные значения (количественные или качественные).

Для количественных входных значений функция принадлежности принимает значения на отрезке от 0 до 1, т.е.  $\mu_{\tilde{A}_i}(x_i^*) \in [0;1]$ .

Для качественных входных значений значения функции принадлежности рассчитываются по следующим правилам:

- 1) Если  $x_i^* = A_i$  То  $\mu_{A_i}(x_i^*) = 1$ ;
- 2) Если  $x_i^* \neq A_i$ , То  $\mu_{A_i}(x_i^*) = 0$ .

Следовательно, при совпадении входного значения  $x_i^*$  с соответствующим значением  $A_i$  в условии правила значение функции принадлежности принимается равным единице. Соответственно, при несовпадении входного значения  $x_i^*$  с соответствующим значением  $A_i$  в условии правила значение функции принадлежности принимается равным нулю.

-  $C \in [0;1]$  – оценка достоверности решения, полученного на основе применения правила [84]:

$$C = T * CF. \quad (2.8)$$

С учетом введенных обозначений разработанный алгоритм логического вывода имеет следующие этапы [86; 87; 100]:

- 1) ввод входных значений  $x_i^*$ ,  $i = 1..n$ ;
- 2) для каждого правила из базы знаний  $R_p \in S$  ( $p=1..N$ ) вычисление степени истинности его антецедента  $T$  согласно (2.7);
- 3) формирование конфликтного множества правил с ненулевой степенью истинности антецедента:  $S_{conf} = \{R_p \mid T \neq 0, p = 1..N\}$ ;
- 4) для каждого правила из множества  $S_{conf}$  расчет оценки достоверности  $C$  согласно (2.8);
- 5) ранжирование правил: сортировка правил из конфликтного множества по убыванию величины оценки достоверности  $C$ ;
- 6) разрешение конфликта: выбор первого решающего правила  $R^* \in S_{conf}$  с максимальным значением оценки достоверности ( $R^*: \max C$ );
- 7) оценка состояния объекта: определение состояния объекта  $B^*$ , как значения выходной переменной  $y$  выбранного правила  $R^*$ , и степени достоверности  $C^*$  выбранного решения;
- 8) обоснование выбранного решения: вывод ранжированного списка правил в естественно-языковом виде, пригодном для лингвистической интерпретации.

На рисунке 2.1 представлена блок-схема алгоритма.

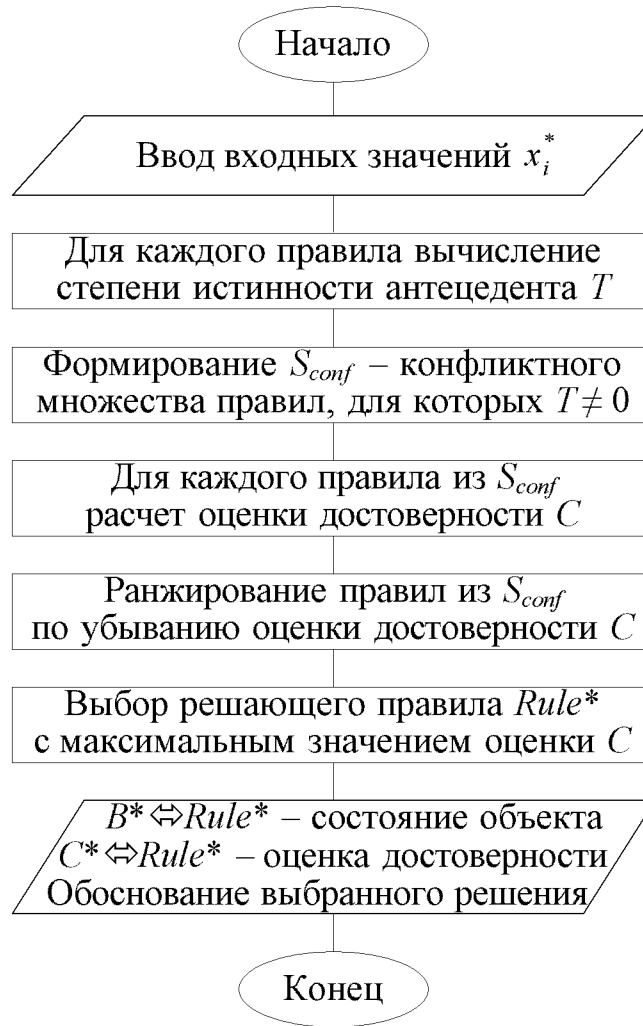


Рисунок 2.1 – Блок-схема алгоритма вывода

Реализация описанных этапов позволяет определять состояния объекта с оценкой достоверности полученного решения.

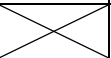
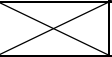
Рассмотрим критерий применения алгоритма в условиях некомплектности входных данных для принятия решений.

## 2.4 Критерий применения алгоритма в условиях некомплектности входных данных

Одним из условий эффективного решения задачи является обеспечение возможности применения разработанного алгоритма логического вывода в условиях некомплектности входных данных для принятия решений. Для этого введем соответствующий критерий.

Пусть  $x_i$  ( $i = 1..n$ ) – все входные переменные, характеризующие конкретный объект оценки. В момент оценки состояния объекта часть значений входных переменных в силу различных причин может отсутствовать. В этом случае наблюдается ситуация некомплектности входных данных для принятия решений. Для применения алгоритма логического вывода в указанных условиях необходимо выбрать базу знаний, соответствующую конкретному составу входных переменных с известными значениями. Это правило является критерием применения алгоритма логического вывода в условиях некомплектности входных данных для принятия решений. Рассмотрим особенности использования этого критерия на конкретном примере (см. таблицу 2.2).

Таблица 2.2 – Соответствие базы знаний конкретному составу входных переменных с известными значениями

Пример состава входных переменных с известными значениями						База знаний
$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$S_{123456}$
$x_1$		$x_3$	$x_4$		$x_6$	$S_{1346}$
$x_1$	$x_2$	$x_3$		$x_5$	$x_6$	$S_{12356}$
...	...	...	...	...	...	...
	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$S_{23456}$

Из таблицы видно, что, полному составу входных переменных объекта (в данном случае  $x_1, x_2, x_3, x_4, x_5, x_6$ ) соответствует база знаний с условным обозначением  $S_{123456}$ . А, например,  $S_{1346}$  – база знаний для оценки состояния объекта и состоящая из набора правил, имеющих входные переменные  $x_1, x_3, x_4, x_6$ . Следовательно, для конкретного состава входных переменных объекта с известными значениями выбирается соответствующая база знаний.

Следует отметить, что необходимым условием для успешного применения введенного критерия является то, что все базы знаний должны быть предварительно сформированы на основе построения нейронечетких моделей для различного состава входных переменных. Кроме того, как было сказано выше, должна быть подтверждена адекватность сформированных баз знаний. Рассмотрим этапы построения нейронечеткой модели для формирования адекватных баз знаний.

## 2.5 Этапы построения нейронечеткой модели

### 2.5.1 Описание этапов построения модели

Как известно [61], нейронечеткая модель является результатом обучения ННС на данных, характеризующих особенности решаемой задачи. После обучения ННС позволяет сформировать систему правил. При этом, от вида правил и алгоритма вывода зависит структура нечеткой нейронной сети и логика ее функционирования. Таким образом, для построения нейронечеткой модели необходимо [89]:

- 1) задать состав слоев и определить функциональность каждого слоя в ННС (состав слоев и их функциональность определяются этапами алгоритма вывода);
- 2) определить состав нейронов в слоях ННС (зависит от анализируемых данных и числа градаций входных нейронов);
- 3) сформировать обучающую, тестовую и валидационную выборки данных (характеристики выборок зависят от метода сэмплинга данных);
- 4) обучить нечеткую нейронную сеть, используя обучающую выборку и контролируя процесс обучения по тестовой выборке (контроль переобучения);
- 5) оценить обобщающую способность обученной нечеткой нейронной сети (нейронечеткой модели) на валидационной выборке данных.

Указанные этапы необходимо реализовать для построения нейронечеткой модели. Рассмотрим описание каждого из этапов более подробно.

### 2.5.2 Задание состава слоев и определение их функциональности

В отличие от классических нейронных сетей [55], для которых возможен произвольный выбор числа скрытых слоев, каждый из которых выполняет функцию преобразования входных сигналов в выходные, для ННС состав слоев и их функциональность однозначно определяются этапами алгоритма вывода. Разработанный алгоритм включает следующие функциональные этапы:

- 1) ввод входных значений;
- 2) вычисление значений функций принадлежности;
- 3) определение степеней истинности и формирование конфликтного множества правил;

4) расчет оценки достоверности принятого решения и ранжирование правил из конфликтного множества;

5) выбор решающего правила и оценка состояния объекта.

В соответствии с указанными этапами разработана схема задания состава слоев в ННС и определения их функциональности (см. рис. 2.2).

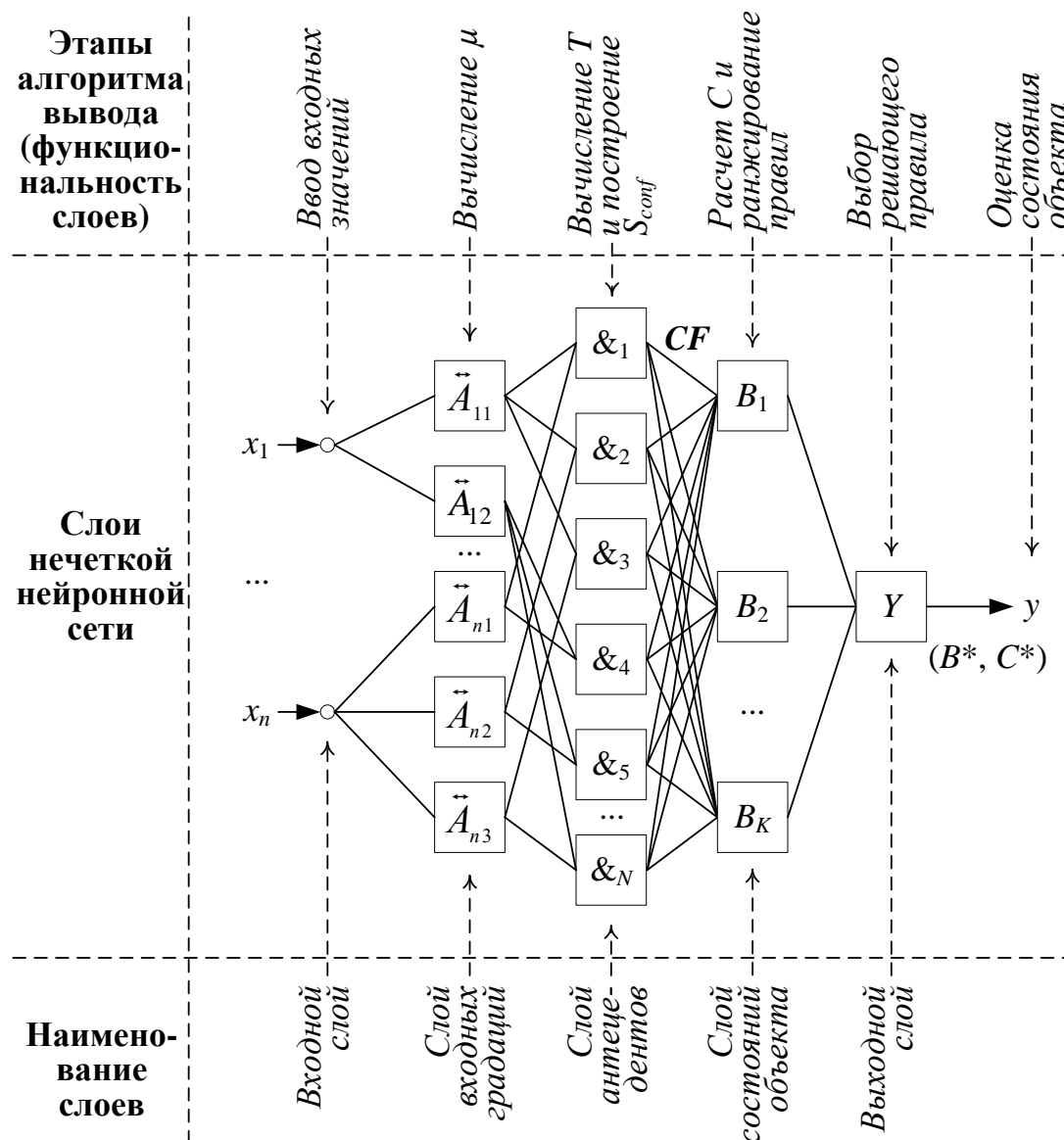


Рисунок 2.2 – Схема задания состава слоев сети и их функциональности

Сеть состоит из следующих функциональных слоев:

- 1) входной слой, выполняющий роль распределения входных значений;
- 2) слой градаций, вычисляющих степень истинности  $\mu$  для каждой градации;
- 3) слой антецедентов, формирующий конфликтное множество правил;
- 4) слой состояний объекта, в котором рассчитываются итоговые оценки достоверности рекомендуемых решений  $C$ ;

5) выходной слой, осуществляющий выбор решающего правила и формирующий результаты решения задачи.

Таким образом, ННС состоит из пяти слоев, функционирующих согласно этапам алгоритма логического вывода.

### 2.5.3 Определение состава нейронов в слоях нечеткой нейронной сети

В каждом слое ННС содержатся нейроны, выполняющие описанные в предыдущем разделе функции. Однако, состав и число нейронов в слоях ННС, в отличие от классических нейронных сетей, не может задаваться произвольно, а однозначно определяется в соответствии с определенными правилами.

На рисунке 2.3 представлена схема определения состава нейронов в ННС.

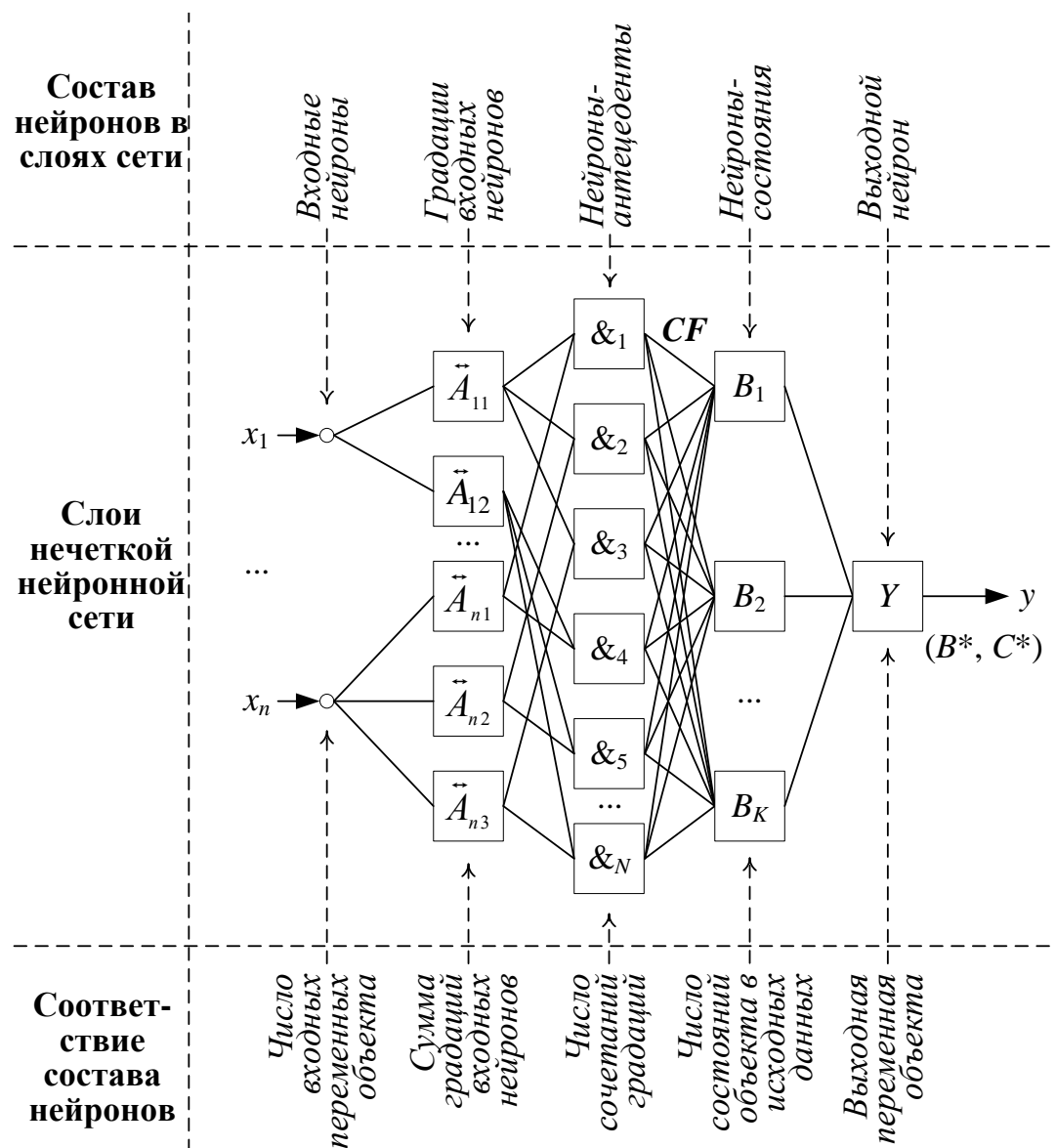


Рисунок 2.3 – Схема определения состава нейронов в слоях нейронной сети

Указанная схема позволяет определить количественный состав нейронов в каждом слое нечеткой нейронной сети.

Число входных нейронов равно числу входных переменных объекта.

Значения количественных входных нейронов разбиваются на определенное число нечетких градаций (как правило, в количестве от 2-х до 9-ти). Для остальных нейронов, соответствующих качественным переменным, градации не задаются. Их значениями являются фактические значения соответствующих входных переменных в обучающей выборке данных. Следовательно, состав нейронов второго слоя определяется составом нечетких градаций количественных входных переменных и значений соответствующих качественных переменных. Общее число нейронов второго слоя равно сумме градаций и значений всех входных нейронов.

В третьем слое нечеткой нейронной сети содержатся нейроны-антецеденты, число которых однозначно определяется числом сочетаний градаций и значений входных нейронов. В частности, как показано на рисунке, если ННС включает два входных нейрона с двумя и тремя градациями соответственно, то число сочетаний этих градаций равно их произведению, то есть в данном случае шести. Для любого другого числа входных нейронов и их градаций (значений) число нейронов третьего слоя определяется аналогично.

Четвертый слой содержит  $K$  нейронов, соответствующих числу состояний объекта в данных.

Единственный нейрон пятого слоя соответствует выходной переменной в обучающей выборке данных.

Следует отметить, что связи между нейронами третьего и четвертого слоев характеризуются соответствующими весами, совокупность которых образует матрицу весовых коэффициентов межнейронных связей  $CF$ .

Таким образом, состав нейронов в ННС зависит от структуры данных, на которых она обучается, а также от числа градаций (значений) входных нейронов.



### 2.5.4 Формирование из исходных данных выборок для обучения, тестирования и валидации нечеткой нейронной сети

Исходные данные еще не предназначены непосредственно для построения нейронечеткой модели. Такая модель является результатом обучения ННС. Необходимо сформировать обучающую выборку данных. Кроме того, всегда необходимо контролировать процесс обучения и следить за тем, чтобы нечеткая нейронная сеть не переобучилась [40]. Для этого потребуется тестовая выборка данных. Следует отметить, что явление переобучения присуще всем методам машинного обучения, в том числе ННС.

На рисунке 2.4 показана схема эффекта переобучения модели.

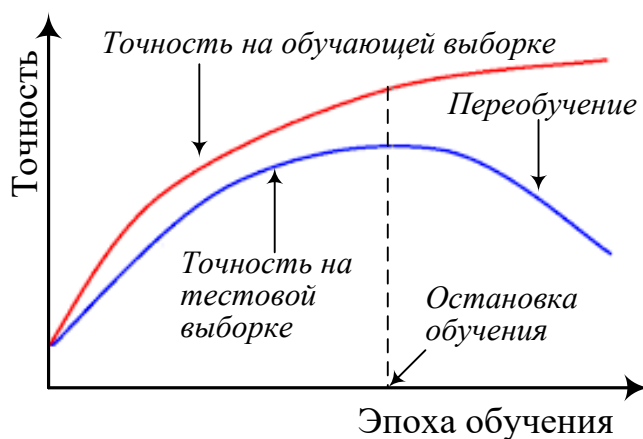


Рисунок 2.4 – Схема эффекта переобучения модели

Из рисунка видно, что в системе координат «Точность» – «Эпоха обучения» в начале обучения модели ее точность на обучающей и тестовой выборках возрастает. Однако на некоторой эпохе обучения (на рисунке показана, как точка остановки обучения) точность на тестовой выборке начинает уменьшаться. Это свидетельствует о начале момента переобучения модели. Поэтому, несмотря на продолжающееся повышение точности на обучающей выборке данных, целесообразно прекратить обучение в этой точке, иначе модель не сможет приобрести способность к обобщению, т.е. к адекватной работе с новыми данными [135; 156].

Результатом успешного обучения ННС является нейронечеткая модель. Однако, перед тем, как ее использовать непосредственно для формирования базы знаний требуется оценить адекватность построенной модели (ее обобщающую

способность), т.е. способность показывать адекватные результаты работы модели на новых данных, которые не использовались в процессе ее построения. Для этого потребуется валидационная выборка данных [153]. При достаточном уровне точности классификации нейронечеткой модели на этой выборке модель будет считаться адекватной. В противном случае потребуется ее переобучение, например, с новыми значениями параметров алгоритма обучения, на обновленных выборках данных или будет принято решение о невозможности построения адекватной нейронечеткой модели на исходных данных. В этом случае может потребоваться обогащение данных для анализа [168].

В любом случае для построения нейронечеткой модели и оценки ее адекватности потребуется предварительное формирование из исходных данных соответствующих выборок. На рисунке 2.5 представлена обобщенная схема формирования обучающей, тестовой и валидационной выборок данных.



Рисунок 2.5 – Схема формирования выборок данных для построения и оценки адекватности нейронечеткой модели

Из исходных данных необходимо сформировать обучающую, тестовую и валидационную выборки. Для этого необходимо выбрать или разработать метод сэмплинга данных [39; 107]. Это может быть случайный сэмплинг, равномерный сэмплинг, стратифицированный сэмплинг, последовательный сэмплинг, сэмплинг

со смещением и другие. В любом случае конкретный метод сэмпинга необходимо выбирать или разрабатывать, исходя из особенностей решаемой задачи оценки состояния объектов. Для построения нейронечеткой модели он должен гарантировать наличие во всех выборках различных примеров из исходных данных, соответствующих всем состояниям объекта.

От метода сэмпинга будет зависеть однородность сформированных выборок и, как следствие, результативность обучения и тестирования ННС, оценки обобщающей способности нейронечеткой модели.

### 2.5.5 Обучение и тестирование нечеткой нейронной сети

Как было сказано выше, сформированные выборки данных будут использованы при обучении, тестировании (контроле переобучения) и валидации (оценке обобщающей способности) ННС. При этом обучение и тестирование являются неразрывно связанными процессами. Рассмотрим эти процессы более подробно.

На рисунке 2.6 показано применение обучающей и тестовой выборок данных для обучения ННС [10].

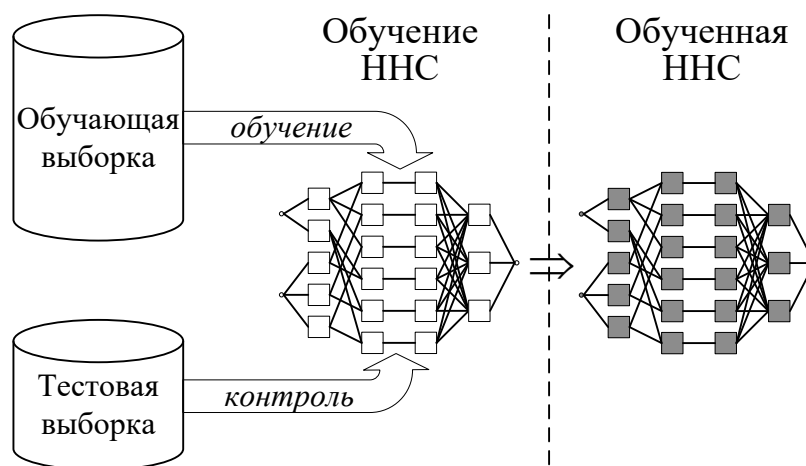


Рисунок 2.6 – Схема обучения сети

Обучающая и тестовая выборки данных используются на этапе обучения нечеткой нейронной сети. Причем тестовая выборка играет вспомогательную функцию, позволяя контролировать ход обучения и не допустить переобучения сети.

Обучение сети происходит за счет настройки значений параметров ее функций принадлежности [143]. В процессе обучения необходимо стремиться максимизировать точность классификации  $A_{train}$  нечеткой нейронной сети на обучающей выборке данных [91]. При этом, как было сказано выше, процесс обучения необходимо контролировать, следя за не уменьшением ее точности классификации  $A_{test}$  на тестовой выборке данных.

Указанные условия можно формализовать в виде следующего критерия:

$$A_{train} = \frac{N_{train\_t}}{N_{train}} \rightarrow \max \mid A_{test} = \frac{N_{test\_t}}{N_{test}} \neq \downarrow, \quad (2.9)$$

где  $N_{train}$  – объем обучающей выборки;

$N_{train\_t}$  – количество правильно классифицированных обучающих примеров ( $N_{train\_t} \leq N_{train}$ );

$N_{test}$  – объем тестовой выборки;

$N_{test\_t}$  – количество правильно классифицированных тестовых примеров ( $N_{test\_t} \leq N_{test}$ );

« $\neq \downarrow$ » – условие не уменьшения точности классификации ННС на тестовой выборке данных.

Выражение (2.9) можно прочесть как «максимизация точности классификации нечеткой нейронной сети на обучающей выборке данных при условии не уменьшения ее точности классификации на тестовой выборке». Этот критерий лежит в основе обучения ННС.

При  $N_{train\_t} = N_{train}$  точность классификации ННС на обучающей выборке данных будет равна единице. Аналогично, при  $N_{test\_t} = N_{test}$  точность классификации ННС на тестовой выборке данных также будет равна единице. Однако в реальных задачах редко достигается 100%-я точность классификации. Поэтому в критерии обучения ННС указано стремление к максимальной точности. При этом предполагается, что в предельном случае максимум будет равен единице, что соответствует 100%-й точности классификации обучающих данных.

### 2.5.6 Оценка обобщающей способности нейронечеткой модели

Перед практическим использованием (формированием базы знаний) обученную ННС (нейронечеткую модель) необходимо протестировать на валидационной выборке данных, т.е. оценить ее адекватность (обобщающую способность).

На рисунке 2.7 показана валидационная выборка данных и ее применение для оценки адекватности обученной нечеткой нейронной сети.

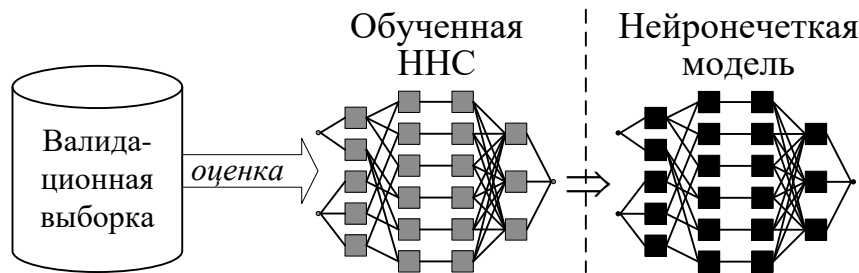


Рисунок 2.7 – Схема валидации обученной нечеткой нейронной сети

Из рисунка видно, что для оценки адекватности обученной нечеткой нейронной сети используется валидационная выборка данных. Результатом этого этапа является искомая нейронечеткая модель, которую можно использовать для формирования БЗ.

Для валидации обученной нечеткой нейронной сети необходимо рассчитать ее точность  $A_{val}$  при классификации данных из валидационной выборки по следующей формуле:

$$A_{val} = \frac{N_{val\_t}}{N_{val}}, \quad (2.10)$$

где  $N_{val}$  – объем валидационной выборки;

$N_{val\_t}$  – количество правильно классифицированных валидационных примеров ( $N_{val\_t} \leq N_{val}$ ).

Полученное значение точности, выраженное в процентах, можно считать оценкой обобщающей способности нечеткой нейронной сети. На этом этапе завершается процесс построения нейронечеткой модели (см. рис. 2.8).

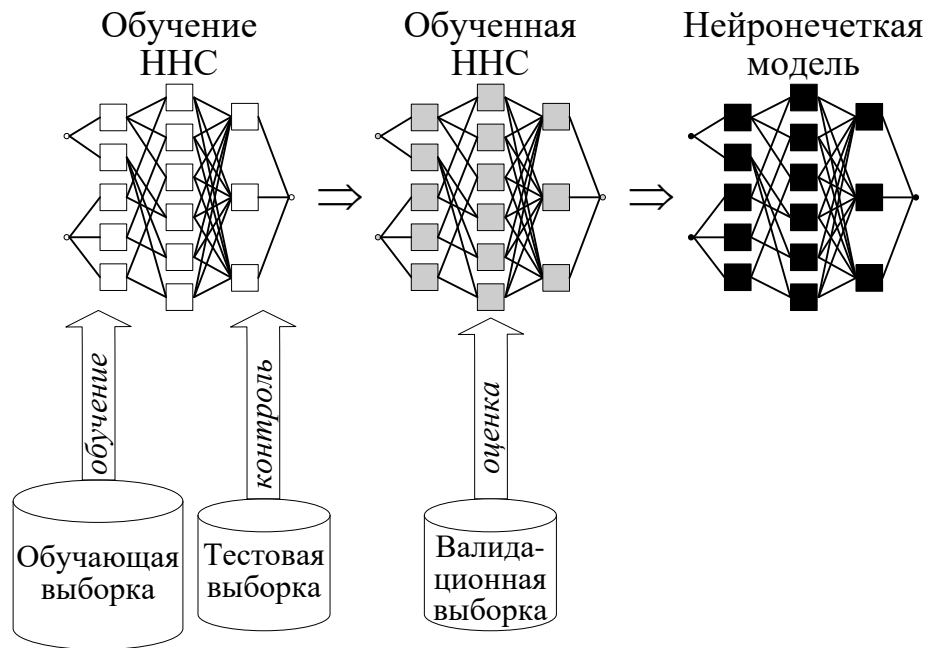


Рисунок 2.8 – Обобщенная схема построения нейронечеткой модели

Таким образом, для построения нейронечеткой модели необходимо реализовать этапы обучения, тестирования и валидации нечеткой нейронной сети. Причем на каждом этапе необходимо применять соответствующую выборку данных.

Для практической реализации описанных этапов необходимо соответствующее математическое и алгоритмическое обеспечение. Рассмотрим постановку задачи разработки методов и алгоритмов построения нейронечеткой модели и формирования соответствующих баз знаний.

## 2.6 Постановка задачи разработки методов и алгоритмов построения нейронечеткой модели и формирования баз знаний

Таким образом, с учетом вышеизложенного, для построения нейронечеткой модели и формирования БЗ требуется разработать [91]:

- 1) метод сэмплинга для формирования обучающей, тестовой и валидационной выборок данных;
- 2) метод расчета значений весов межнейронных связей  $CF$ ;
- 3) метод и алгоритм обучения ННС, максимизирующие ее точность классификации  $A_{train}$  на обучающей выборке данных в условиях контроля ее переобуче-

ния, т.е. не уменьшения точности классификации  $A_{test}$  на тестовой выборке, удовлетворяющие выражению (2.9);

4) метод и алгоритм формирования базы знаний на основе построенной нейронечеткой модели.

Решению указанных задач посвящена следующая глава.

## 2.7 Выводы

Анализ основных видов нечетких правил оценки состояния объектов показал, что для обработки разнотипных входных данных, однозначной интерпретации и ранжирования получаемых решений актуальна разработка нового вида правил. Предложен вид правил, удовлетворяющий указанным требованиям. Разработан алгоритм вывода, позволяющий обосновывать предлагаемые решения в виде, пригодном для лингвистической интерпретации ЛПР. Его применение в условиях некомплектности входных данных для принятия решений потребовало разработки критерия, заключающегося в предварительном формировании и выборе при принятии решения такой базы знаний, которая соответствует фактическому набору входных переменных объекта. При этом все базы знаний должны быть заранее сформированы на основе построения нейронечеткой модели и быть адекватны.

Построение нейронечеткой модели потребовало выполнения этапов:

- 1) задания состава слоев ННС и определения их функциональности;
- 2) определения состава нейронов в каждом слое ННС;
- 3) формирования выборок данных для обучения, тестирования и оценки обобщающей способности ННС;
- 4) обучения ННС при условии контроля переобучения;
- 5) оценки обобщающей способности ННС (нейронечеткой модели) на валидационной выборке данных.

Показано, что разработанная ННС состоит из пяти слоев, функционирующих согласно этапам алгоритма логического вывода. Состав нейронов в каждом слое ННС зависит от структуры данных, на которых она обучается, а также от

числа градаций (значений) входных нейронов. Для формирования однородных выборок данных требуется разработать метод сэмпинга. При обучении ННС необходимо стремиться максимизировать точность классификации на обучающей выборке данных при условии не уменьшения точности на тестовой выборке. Этот критерий лежит в основе обучения ННС. Для валидации обученной ННС необходимо рассчитать точность ее классификации на валидационной выборке данных. Полученное значение точности, выраженное в процентах, является оценкой обобщающей способности нейронечеткой модели. Для практической реализации этапов построения нейронечеткой модели и формирования БЗ требуется разработать соответствующие методы и алгоритмы.



### **3 Методы и алгоритмы построения нейронечеткой модели и формирования баз знаний для оценки состояния объектов**

*В этой главе предлагается метод сэмплинга данных, используемых для обучения, тестирования и валидации нечеткой нейронной сети, метод расчета значений ее весовых коэффициентов. Описываются методы и алгоритмы построения нейронечеткой модели: метод и алгоритм обучения ННС, генетический алгоритм настройки значений параметров ФП, алгоритм оценки обобщающей способности нейронечеткой модели. Кроме того, раскрываются вопросы определения начальных значений параметров ФП, а также выбора и кодирования параметров задачи в ГА. Описываются метод и алгоритм формирования базы знаний на основе построенной модели.*

#### **3.1 Метод сэмплинга данных для формирования обучающей, тестовой и валидационной выборок**

Рассмотрим разработанный метод сэмплинга, позволяющий из исходных данных для анализа сформировать обучающую, тестовую и валидационную выборки, используемые соответственно для обучения и оценки обобщающей способности нечеткой нейронной сети. При разработке метода учитывались особенности решаемой задачи оценки состояния объекта, а именно ее дискретный характер, соответствующий состояниям  $V_1, V_2, \dots, V_k, \dots, V_K$  оцениваемого объекта. Кроме того, при разработке метода сэмплинга учитывалась необходимость гарантированного включения во все выборки различных примеров из исходных данных, соответствующих всем состояниям объекта.

На рисунке 3.1 показана схема разработанного метода сэмплинга данных.



Рисунок 3.1 – Схема разработанного метода сэмплинга данных

Согласно рисунку, разработанный метод сэмплинга данных предполагает выполнение следующих этапов:

- 1) разбиение исходных данных для анализа на группы в соответствии с состояниями объекта  $B_k, k=1..K$ ;
- 2) формирование из каждой  $k$ -й группы подвыборок  $ОБ_k, ТВ_k, ВВ_k, k=1..K$ , на основе процедуры случайного сэмплинга с замещением [13; 84; 152] (здесь  $ОБ$  – обучающая выборка,  $ТВ$  – тестовая выборка,  $ВВ$  – валидационная выборка);
- 3) объединение соответствующих подвыборок для получения итоговых выборок данных:

$$ОБ = ОБ_1 \cup \dots \cup ОБ_k \cup \dots \cup ОБ_K;$$

$$ТВ = ТВ_1 \cup \dots \cup ТВ_k \cup \dots \cup ТВ_K;$$

$$ВВ = ВВ_1 \cup \dots \cup ВВ_k \cup \dots \cup ВВ_K.$$

Следует отметить, что в основу разработанного метода положена процедура формирования выборок данных на основе случайного сэмплинга с замещением [152]. Рассмотрим этапы ее реализации на примере формирования подвыборок  $OB_k$ ,  $TB_k$ ,  $VB_k$  из  $k$ -й группы данных объемом  $N_k$ .

- 1) из  $k$ -й группы данных  $N_k$  раз случайным образом выбирается строка и копируется в подвыборку  $OB_k$ ;
- 2) из  $k$ -й группы данных удаляются все строки, помещенные в  $OB_k$ ;
- 3) оставшиеся в  $k$ -й группе строки случайным образом равномерно разделяются на два непересекающихся подмножества, образуя подвыборки  $TB_k$  и  $VB_k$ .

Следует отметить, что при формировании подвыборки  $OB_k$  каждая строка из  $k$ -й группы данных может быть случайно выбрана несколько раз. При этом объем подвыборки  $OB_k$  будет равен  $N_k$ , что совпадает с объемом исходной  $k$ -й группы данных. Легко показать, что подвыборка  $OB_k$  будет содержать примерно 63,2% уникальных строк из  $k$ -й группы данных, а каждая из подвыборок  $TB_k$  и  $VB_k$  – примерно по 18,4% уникальных строк. При этом объемом каждой подвыборки  $TB_k$  и  $VB_k$  будет равен примерно 18,4% объему  $k$ -й группы данных.

Действительно, при объеме  $k$ -й группы, равном  $N_k$ , вероятность выбора случайной строки  $p_1 = \frac{1}{N_k}$ , а вероятность ее не выбора  $p_2 = 1 - \frac{1}{N_k}$ . Умножая  $p_1$  на

$p_2 N_k$  раз, получим  $p = \left(1 - \frac{1}{N_k}\right)^{N_k} \approx e^{-1} = 0,368$  (или 36,8%). Это вероятность не

выбора строки за  $N_k$  раз. Вероятность выбора строки за  $N_k$  раз равна  $(1 - p) = 0,632$  (или 63,2%). Следовательно, подвыборка  $OB_k$  будет содержать примерно 63,2% уникальных строк из  $k$ -й группы данных, а подвыборки  $TB_k$  и  $VB_k$ , соответственно, по  $36,8\% * 0,5 = 18,4\%$  уникальных строк.

Рассуждая аналогично для всех остальных групп данных, приходим к выводу, что итоговая обучающая выборка будет иметь объем, равный 100% объему исходных данных для анализа. При этом она будет содержать примерно 63,2% уникальных строк из исходных данных. Соответственно, итоговые тестовая и

валидационная выборки будут иметь примерный объем по 18,4% от объема исходных данных и содержать примерно по 18,4% его уникальных строк.

Таким образом, разработанный метод можно назвать методом группового случайного сэмпинга с замещением. Его использование позволяет сформировать обучающую, тестовую и валидационную выборки данных соответствующего объема. Его достоинство заключается в гарантии того, что во всех формируемых выборках будут присутствовать данные всех классов, соответствующих состояниям оцениваемого объекта. Это достигается за счет предварительного разбиения исходных данных для анализа на группы в соответствии с состояниями объекта  $B_k$ ,  $k=1..K$ . Кроме того, применение промежуточной процедуры случайного сэмпинга с замещением позволяет сформировать обогащенную обучающую выборку объемом, равным объему исходных данных. Это особенно актуально в случае малого объема имеющихся данных для анализа [121; 214].

### 3.2 Метод расчета весовых коэффициентов в нечеткой нейронной сети

Как было показано во 2-й главе, в структуре нечеткой нейронной сети имеются веса связей между всеми нейронами 3-го и 4-го слоев. Эти же веса являются весами соответствующих нечетко-продукционных правил.

Зададим матрицу весовых коэффициентов  $CF$  в следующем виде [86]:

$$CF = \begin{pmatrix} CF_{11} \dots CF_{1k} \dots CF_{1K} \\ \dots \\ CF_{p1} \dots CF_{pk} \dots CF_{pK} \\ \dots \\ CF_{N1} \dots CF_{Nk} \dots CF_{NK} \end{pmatrix}, \quad (3.1)$$

где  $CF_{pk}$  – веса связей нейронов 3-го и 4-го слоев в ННС ( $p = 1..N$ ,  $k = 1..K$ );  
 $N$  – число нейронов 3-го слоя (число правил для состояния объекта  $B_k$ );  
 $K$  – число нейронов 4-го слоя (число состояний объекта).

Рассмотрим предложенный в диссертации метод расчета значений элементов матрицы (3.1). Пусть имеется обучающая выборка и необходимо определить вес связи  $CF_{pk}$  в ННС между нейронами  $\&_p$  и  $B_k$ . Данная связь соответствует маршруту в сети, задающему следующее правило:

$$\text{If } x_1 = \vec{A}_{12} \ \& \dots x_n = \vec{A}_{n1} \ \text{Then } y = B_k [CF_{pk}]. \quad (3.2)$$

На рисунке 3.2 показан маршрут в ННС, соответствующий указанному правилу с весом  $CF_{pk}$ , который требуется определить.

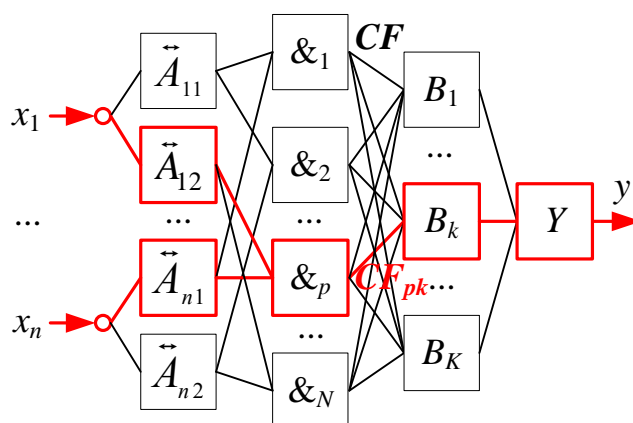


Рисунок 3.2 – Маршрут в нечеткой нейронной сети

Для расчета значений  $CF_{pk} \in CF$  ( $p=1..N$ ,  $k=1..K$ , где  $N$  – число нейронов-антецедентов в ННС,  $K$  – число нейронов-состояний) необходимо выполнить следующие этапы [84]:

1) определить  $N_k$  – число примеров в обучающей выборке со значением выходной переменной, равным  $B_k$ ;

2) определить  $N_R$  – число реализаций шаблона правила (3.2) в этой выборке ( $N_R \leq N_k$ );

3) для каждой реализации определить значение выражения  $\min_i(\mu_{\vec{A}_{ij_i}}^c(x_i^*))$ , где  $i = 1..n$ ,  $c = 1..N_R$ ,  $j_i = 1..n_i$ ,  $n_i$  – число градаций  $i$ -го входного нейрона,  $n$  – число входных нейронов;

4) рассчитать значение веса  $CF_{pk}$  по следующей эмпирической формуле:

$$CF_{pk} = \frac{\sum_{c=1}^{N_R} \min_i(\mu_{\vec{A}_{ij_i}}^c(x_i^*))}{N_k}. \quad (3.3)$$

Следует отметить, что выражение « $\min_i(\mu_{A_{ij}}^c(x_i^*))$ » можно интерпретировать, как степень истинности antecedента для  $c$ -й реализации шаблона правила (3.2) в обучающей выборке данных ( $c = 1..N_R$ ) [84]. Применяя формулу (3.3), можно рассчитать значения всех элементов  $CF_{pk} \in CF$  ( $p=1..N, k=1..K$ ).

### 3.3 Построение нейронечеткой модели

#### 3.3.1 Этапы параметрической идентификации нейронечеткой модели

Для построения любой интеллектуальной модели на основе методов машинного обучения [91] (в том числе нейронечеткой модели) требуется выполнить ряд последовательных этапов, связанных с ее структурной и параметрической идентификацией [66; 104; 178; 179]. Реализация этапа структурной идентификации нейронечеткой модели описана во 2-й главе диссертации. На данном этапе задается состав слоев, их функциональность и состав нейронов в каждом слое нечеткой нейронной сети. Это однозначно определяет структуру нейронечеткой модели.

Рассмотрим этапы построения нейронечеткой модели, связанные с ее параметрической идентификацией (см. рис. 3.3).

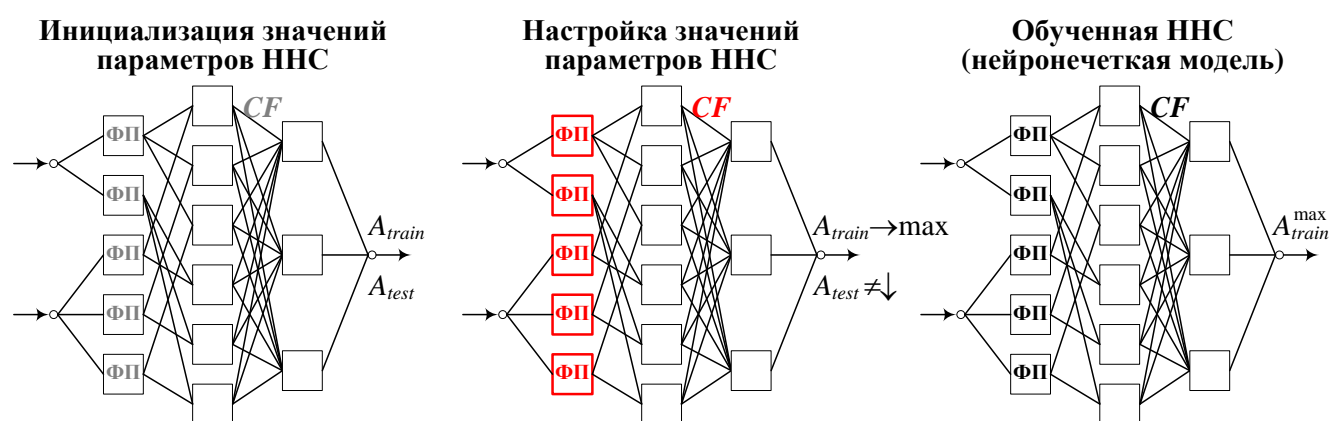


Рисунок 3.3 – Этапы параметрической идентификации нейронечеткой модели

Как видно из рисунка, для параметрической идентификации нейронечеткой модели требуется выполнить следующие этапы [11; 66]:

- 1) инициализация значений параметров ННС;
- 2) настройка значений параметров (обучение) ННС.

На первом этапе рассчитываются начальные значения параметров ФП в ННС и весов  $CF_{pk} \in CF$  по формуле (3.3). При этом также требуется рассчитать начальные значения точности классификации ННС на обучающей ( $A_{train}$ ) и тестовой ( $A_{test}$ ) выборках данных [91].

На втором этапе производится так называемый «тюнинг» модели, в результате которого определяются оптимальные значения параметров ФП и весов  $CF_{pk} \in CF$  в ННС с учетом критерия оптимизации (2.9). Следовательно, при реализации этого этапа необходимо максимизировать точность классификации  $A_{train}$  нечеткой нейронной сети на обучающей выборке данных в условиях контроля ее переобучения (не уменьшения точности классификации ННС  $A_{test}$  на тестовой выборке) [91].

Результатом выполнения этих этапов является построенная нейронечеткая модель. Рассмотрим каждый из этапов более подробно.

### **3.3.2 Определение начальных значений параметров функций принадлежности**

Этот этап является первичным этапом построения нейронечеткой модели. Рассмотрим его реализацию на примере треугольных и трапецеидальных ФП [91]. Выбор этих функций обусловлен широтой и эффективностью их практического использования [161; 207], а также простотой задания и лингвистической интерпретации человеком, что требуется для принятия обоснованных решений [155].

На рисунке 3.4 показаны примеры инициализации параметров треугольных ФП для 2-х, 3-х и 4-х градаций.

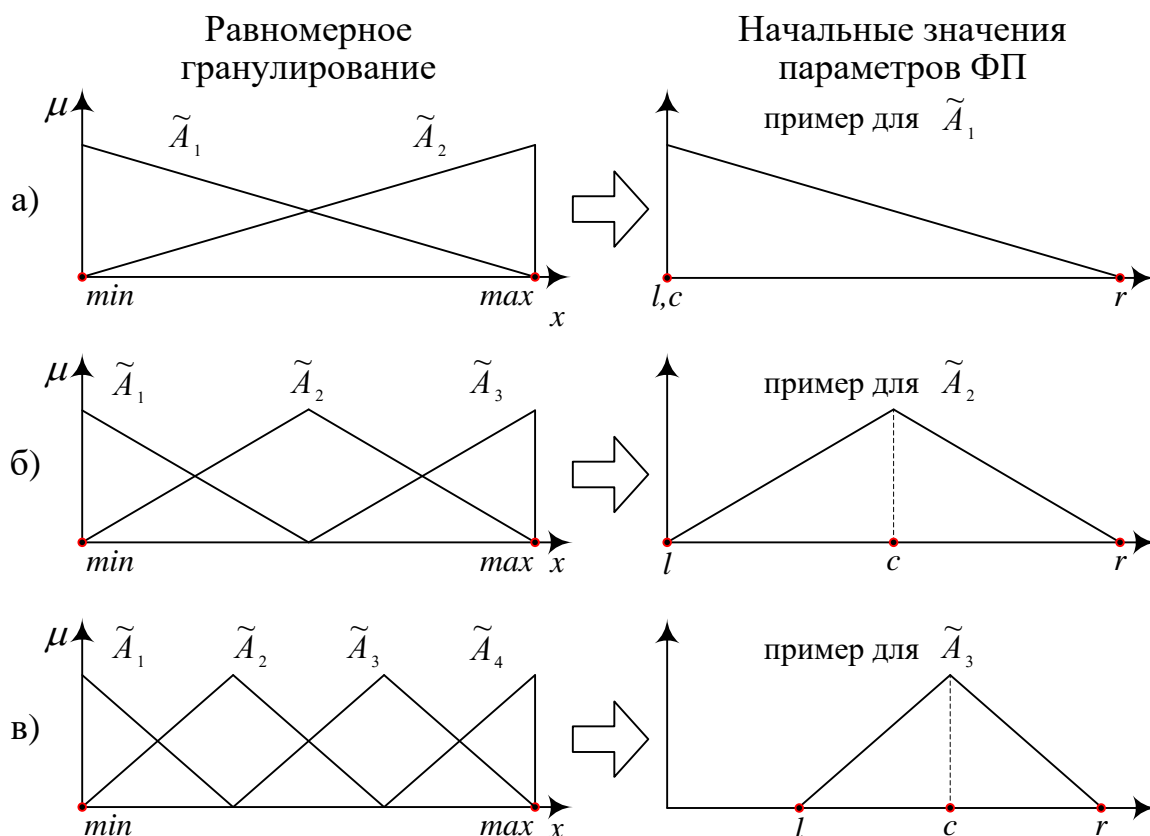


Рисунок 3.4 – Инициализация параметров треугольных функций принадлежности

При любом количестве выбираемых нечетких градаций используется равномерный метод гранулирования [59]. Такой подход является классическим и позволяет максимально просто определить начальные значения параметров каждой функции принадлежности [91].

Так, например, в случае 2-х нечетких градаций задаются две треугольные ФП на отрезке  $[min; max]$ , как показано на рисунке 3.4 (а). Границы этого отрезка соответствуют минимальному и максимальному значениям переменной  $x$  в обучающих данных. Это позволяет произвести инициализацию параметров ФП [84]: левого основания ( $l$ ), моды ( $c$ ) и правого основания ( $r$ ).

На рисунке 3.4 (а) справа показан пример определения начальных значений параметров для первой функции принадлежности. Аналогично определяются начальные значения параметров треугольных функций принадлежности для 3-х (рис. 3.4 (б)), 4-х (рис. 3.4 (в)) и большего числа градаций.



Следует отметить, что трапециевидальные ФП однозначно задаются следующими четырьмя параметрами [84]: левым основанием ( $l$ ), левой границей моды ( $c_1$ ), правой границей моды ( $c_2$ ) и правым основанием ( $r$ ) (см. рис. 3.5).

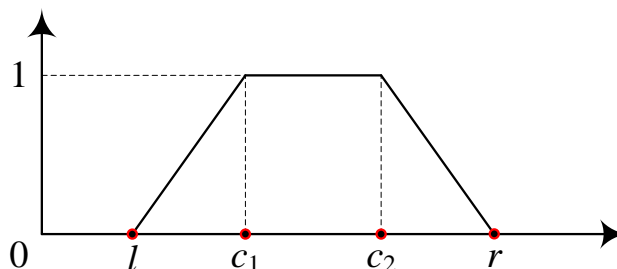


Рисунок 3.5 – Параметры трапециевидальной функции принадлежности

В предельном случае (при  $c_1 = c_2$ ) трапециевидальная функция принадлежности становится треугольной. Следовательно, треугольные ФП являются частным случаем трапециевидальных. В этом смысле функции принадлежности в форме трапеции, в отличие от треугольников, потенциально обладают большими аппроксимационными возможностями [15; 199].

Следует отметить, что инициализация параметров трапециевидальных ФП выполняется аналогично и принципиально ничем не отличается от описанного примера. После выполнения этого этапа необходимо обучить нечеткую нейронную сеть путем настройки значений ее параметров [91]. Рассмотрим разработанные в диссертации метод и алгоритмы, используемые для обучения ННС.

### 3.3.3 Метод обучения нечеткой нейронной сети

Существует большое количество методов оптимизации [28; 44; 209], используемых в машинном обучении при построении интеллектуальных моделей [91]. Так, например, при обучении нейронных сетей часто используются градиентные методы [74] и алгоритм обратного распространения ошибки [50], который стал классическим для построения таких моделей. Данный алгоритм предполагает вычисление и минимизацию среднеквадратичной ошибки выхода сети на основе вычисления градиентов по настраиваемым параметрам модели.

Однако в предложенной в диссертации нечеткой нейронной сети функция ошибки принимает дискретные значения и зависит от дискретных параметров, поэтому применение понятия «градиент» для нее будет некорректным. Кроме того, градиент вычисляется для гладких функций. Используемые в диссертации треугольные и трапецеидальные функции принадлежности кусочно-линейны, т.е. не являются гладкими на всей области определения [91].

Поэтому в диссертации для обучения ННС использован генетический алгоритм (ГА) [49; 181]. В настоящее время такие алгоритмы получили широкое распространение при построении интеллектуальных моделей [32; 65]. Они не требуют использования гладких функций и могут быть реализованы с использованием различных эвристических приемов, связанных с кодированием параметров задачи, выполнением генетических операторов и выбором критериев для нахождения квазиоптимальных решений поставленных задач [91].

На рисунке 3.6 показан принцип работы предложенного метода обучения ННС на основе ГА.

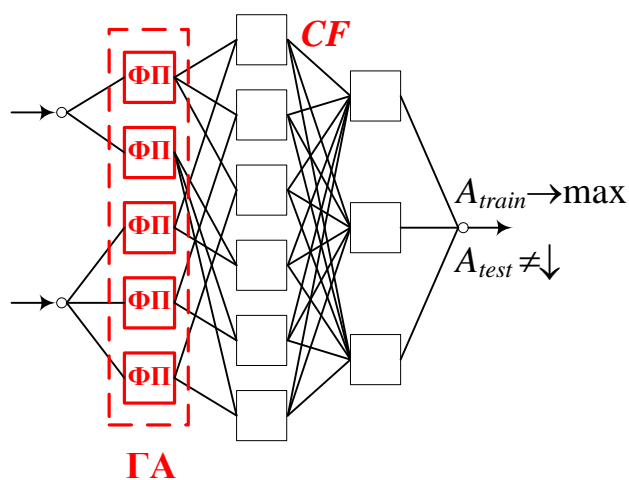


Рисунок 3.6 – Принцип обучения нечеткой нейронной сети

Для обучения ННС используется ГА. При этом на каждой итерации генетической оптимизации происходит уточнение весов  $CF$ . Целью обучения является максимизация  $A_{train}$  (точности классификации обучающих данных) при условии не уменьшения  $A_{test}$  (точности классификации тестовых данных) [91].

Таким образом, для обучения ННС разработан метод итерационной генетической оптимизации [91]. Рассмотрим его особенности.

### 3.3.4 Выбор и кодирование параметров обучения

Несмотря на то, что все генетические алгоритмы предполагают выполнение достаточно стандартных операций, в каждом конкретном случае требуется решение вопросов выбора настраиваемых параметров функций принадлежности, кодирования их значений в виде бинарных хромосом, а также задания фитнес-функции для количественной оценки полезности хромосом.

Рассмотрим вопросы выбора и кодирования параметров обучения на примере треугольных ФП [91] (для трапецеидальных функций принадлежности все рассуждения будут аналогичны), представленных на рисунке 3.7.

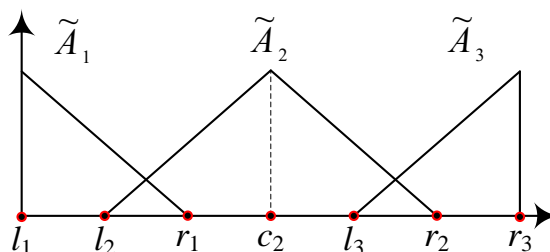


Рисунок 3.7 – Треугольные функции принадлежности и их параметры

На рисунке представлены три треугольные функции принадлежности [91]:

- $\tilde{A}_1$  с параметрами  $(l_1, r_1)$ ;
- $\tilde{A}_2$  с параметрами  $(l_2, c_2, r_2)$ ;
- $\tilde{A}_3$  с параметрами  $(l_3, r_3)$ .

Следует отметить, что для задания крайних ФП ( $\tilde{A}_1$  и  $\tilde{A}_3$ ) достаточно иметь два параметра (в данном случае это  $l$  – левое основание и  $r$  – правое основание).

Очевидно, что не все из представленных параметров  $\{l_1, l_2, r_1, c_2, l_3, r_2, r_3\}$  являются настраиваемыми. Следовательно, при обучении ННС необходимо выбирать такие параметры  $p$ , значения которых можно настраивать [91].

С одной стороны, обучение ННС заключается в настройке значений параметров ее ФП. С другой, значения крайних параметров (в данном случае это параметры  $l_1$  и  $r_3$ ) не могут изменяться, иначе нарушится интерпретируемость соответствующих крайних ФП. Следовательно,  $p \in \{l_2, r_1, c_2, l_3, r_2\}$ . Значения именно этих параметров могут и должны изменяться при обучении ННС [91].

Следующим вопросом является определение интервалов значений для настраиваемых параметров. Поскольку функции принадлежности должны быть интерпретируемы, упорядочены и не иметь точек разрыва, то в качестве указанных интервалов параметров  $p \in (min; max)$  целесообразно выбрать следующие [91]:

$$l_2 \in (l_1; r_1), r_1 \in (l_2; l_3), c_2 \in (l_2; r_2), l_3 \in (r_1; r_2), r_2 \in (l_3; r_3).$$

Выбранные интервалы, ограничивающие значения настраиваемых параметров, легко поддаются интерпретации. Так, например, выражение  $l_2 \in (l_1; r_1)$  означает, что левое основание ФП не может лежать левее, чем левое основание располагающейся слева от нее ФП, и правее, чем правое основание указанной ФП. Аналогично интерпретируются все другие интервалы.

Следующим вопросом, требующим решения, является кодирование значений настраиваемого параметра в заданном интервале. Для этого каждый интервал равномерно разобьем на  $2^m + 1$  отрезков ( $m$  – целое число). Тогда параметр сможет принимать одно из  $2^m$  значений [86], соответствующих границам между отрезками. Значение параметра можно закодировать хромосомой из множества [83]:

$$H = \{H_1, \dots, H_z, \dots, H_Z\} = \{(\underbrace{0, 0, \dots, 0}_m), (\underbrace{0, 0, \dots, 1}_m), \dots, (\underbrace{1, 1, \dots, 1}_m)\}, z = 1..Z, Z = 2^m,$$

где  $H_1 = (0, 0, \dots, 0)$  – хромосома для наименьшего значения параметра;

$H_Z = (1, 1, \dots, 1)$  – хромосома для его наибольшего значения;

$m$  – длина хромосомы (количество используемых в ней генов).

На рисунке 3.8 показан пример кодирования значений параметра (для  $m=4$ ).

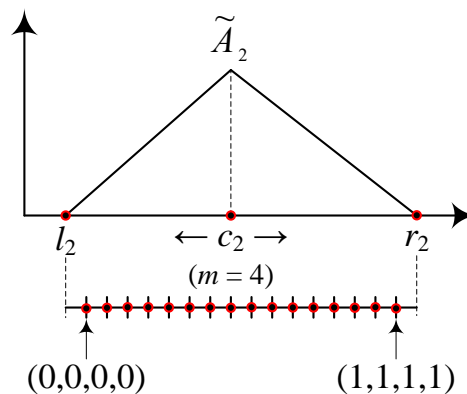


Рисунок 3.8 – Пример кодирования значений параметра

Интервал  $(l_2; r_2)$  ограничивает возможные значения для настраиваемого параметра  $c_2$ . В данном случае, этот параметр может принимать одно из 16 значений (так как  $m = 4$ , то  $2^m = 2^4 = 16$ ).

Завершающим вопросом, требующим решения, является задание фитнес-функции для оценки значений настраиваемого параметра. Так как целью обучения ННС является максимизация точности ее классификации на обучающей выборке данных, то в качестве фитнес-функции целесообразно задать следующую [91]:

$$F(H_z) = \frac{N_{train\_t}}{N_{train}}, \quad (z = 1..Z, Z = 2^m), \quad (3.4)$$

где  $N_{train}$  – объем обучающей выборки;

$N_{train\_t}$  – число верно распознанных обучающих примеров ( $N_{train\_t} \leq N_{train}$ ).

Критерий выбора лучшей хромосомы: максимизация точности классификации ННС на обучающей выборке (в соответствии с выражением (3.4)) при условии не уменьшения ее точности классификации  $A_{test}$  на тестовой выборке:

$$F(H_z) \rightarrow \max_{\forall H_z} | A_{test}(H_z) \neq \downarrow, A_{test}(H_z) = \frac{N_{test\_t}}{N_{test}}, \quad (3.5)$$

где  $N_{test}$  – объем тестовой выборки;

$N_{test\_t}$  – число верно распознанных тестовых примеров ( $N_{test\_t} \leq N_{test}$ ).

Таким образом, выбор настраиваемых параметров ФП, определение интервалов и кодирование их возможных значений, а также задание фитнес-функции с критерием выбора лучшей хромосомы позволяют реализовать разработанный ГА.

### 3.3.5 Генетический алгоритм настройки значений параметров функций принадлежности

Рассмотрим этапы разработанного ГА:

1) создать популяцию хромосом  $h = \{h_1, \dots, h_x, \dots, h_X\} \in H$  объемом  $X < Z$  для значений параметра ФП (хромосому для текущего значения параметра и хромосому, полученные от случайной мутации ее генов с вероятностью  $p_1 = 0,5$  [83]);

2) для каждой хромосомы  $h_x$  определить значения элементов  $CF$  согласно (3.3);

3) оценить приспособленность хромосом  $h_x$  на основе фитнес-функции  $F(h_x)$  согласно (3.4);

4) произвести селекцию 2-х родительских хромосом по методу рулетки [139] (см. рис. 3.9):

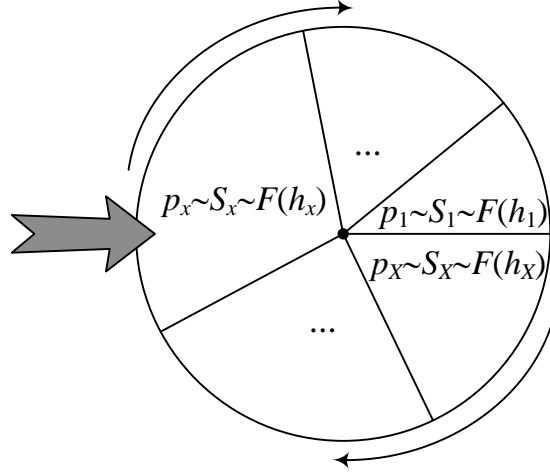


Рисунок 3.9 – Пример работы метода рулетки

где 
$$p_x = \frac{F(h_x)}{\sum_{x=1}^X F(h_x)}$$
 – вероятность выбора хромосомы  $h_x$ ;

$S_x$  – площадь  $x$ -го сектора;

5) произвести скрещивание родительских хромосом для получения 2-х дочерних [84] и включить их в популяцию (для скрещивания используется двухточечный кроссинговер с плавающими точками [83; 137]);

6) произвести мутацию генов дочерних хромосом с вероятностью  $p_2 = 0,05$ ;

7) для дочерних хромосом определить значения элементов  $CF$  согласно (3.3);

8) оценить приспособленность дочерних хромосом согласно (3.4);

9) удалить 2 хромосомы с минимальными значениями фитнес-функции [84];

10) для хромосомы  $h^*$  с лучшей приспособленностью рассчитать соответствующую точность классификации ННС  $A_{test}(h^*)$  согласно (3.5);

11) повторять этапы 4÷10 до тех пор, пока в течение определенного числа поколений происходит улучшение приспособленности хромосом при условии не уменьшения точности классификации ННС на тестовой выборке, вычисляемой согласно (3.5) и соответствующей хромосоме с лучшей приспособленностью;

12) выбрать лучшую хромосому  $h^*$  ( $F(h^*) = \max_{\forall h_x} F(h_x)$ ) и считать значение параметра выбранной функции принадлежности настроенным.

Следует отметить, что описанный ГА применяется для настройки значений одного параметра ФП. Итерационная генетическая оптимизация значений всех параметров каждой ФП позволяет полностью обучить ННС.

На рисунке 3.10 показана блок-схема работы ГА.



Рисунок 3.10 – Блок-схема работы генетического алгоритма

Таким образом, применение разработанного генетического алгоритма позволяет настраивать значение параметра выбранной ФП и уточнять значения весов в нечеткой нейронной сети.

### 3.3.6 Алгоритм обучения нечеткой нейронной сети

Рассмотрим этапы алгоритма обучения ННС:

- 1) получить исходную ННС с начальными значениями параметров;
- 2) последовательно подавая на вход сети данные из обучающей выборки, рассчитать точность их классификации  $A_{train}$ ;
- 3) проверить условие: если  $A_{train} = 1$ , то завершить работу алгоритма, иначе перейти на этап 4;
- 4) произвести обучение нечеткой нейронной сети:
  - 4.1) последовательно выбрать в нечеткой нейронной сети входной нейрон с нечеткими градациями;
  - 4.2) последовательно выбрать нечеткую градацию выбранного входного нейрона для настройки значений параметров ее функции принадлежности;
  - 4.3) последовательно выбрать настраиваемый параметр функции принадлежности у выбранной нечеткой градации;
  - 4.4) применить ГА для настройки значений выбранного параметра функции принадлежности;
  - 4.5) повторять этапы 4.1÷4.4 до тех пор, пока в течение определенного числа циклов обучения дальнейшая настройка параметров ФП не будет приводить к повышению точности классификации  $A_{train}$  на обучающей выборке данных или значение  $A_{train}$  не станет равным единице.

Следует отметить, что алгоритм за конечное число шагов позволяет настроить значения параметров каждой ФП. Его сходимость обеспечена свойствами генетического алгоритма.



На рисунке 3.11 показана блок-схема работы алгоритма.



Рисунок 3.11 – Блок-схема работы алгоритма обучения нечеткой нейронной сети

После обучения нечеткой нейронной сети требуется оценить ее обобщающую способность. Рассмотрим алгоритм, используемый в диссертации для решения этой задачи.

### 3.3.7 Алгоритм оценки обобщающей способности нейронечеткой модели

Как было отмечено во 2-й главе диссертации, для оценки обобщающей способности обученной нечеткой нейронной сети (нейронечеткой модели) используется валидационная выборка данных. Для получения этой оценки используется простой алгоритм. Рассмотрим его этапы:

- 1) получить обученную ННС с настроенными значениями параметров;
- 2) последовательно подавая на вход сети данные из валидационной выборки, рассчитать точность их классификации  $A_{val}$  по формуле (2.10);
- 3) считать значение точности, выраженное в процентах, оценкой обобщающей способности обученной нечеткой нейронной сети (нейронечеткой модели).

На рисунке 3.12 показана блок-схема работы алгоритма.

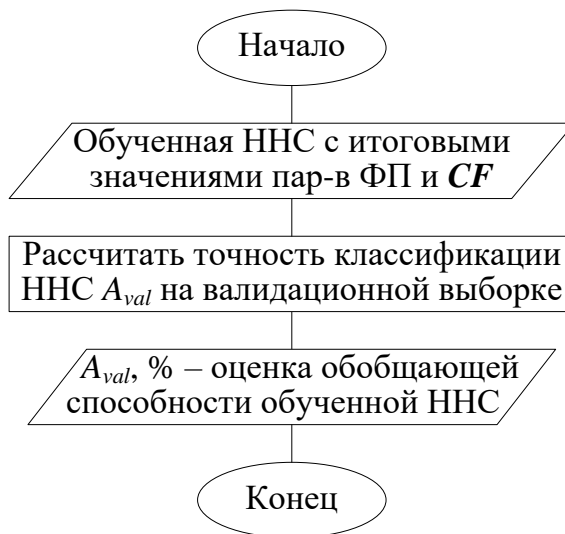


Рисунок 3.12 – Блок-схема алгоритма валидации  
нечеткой нейронной сети

По величине полученной оценки можно судить об адекватности построенной нейронечеткой модели и возможности ее практического использования. Рассмотрим разработанные в диссертации метод и алгоритм формирования базы знаний.

### 3.4 Формирование базы знаний на основе построенной модели

#### 3.4.1 Метод формирования базы знаний

Для формирования БЗ на основе построенной нейронечеткой модели разработан специальный метод (метод маршрутов), позволяющий для каждого входного образа из исходных данных построить оптимальный маршрут его прохождения по слоям обученной нечеткой нейронной сети, сгенерировать нечетко-продукционное правило, соответствующее построенному маршруту, и сформировать искомую базу знаний, содержащую уникальные правила [98].

На рисунке 3.13 представлена обобщенная схема разработанного метода формирования базы знаний на основе построенной нейронечеткой модели.

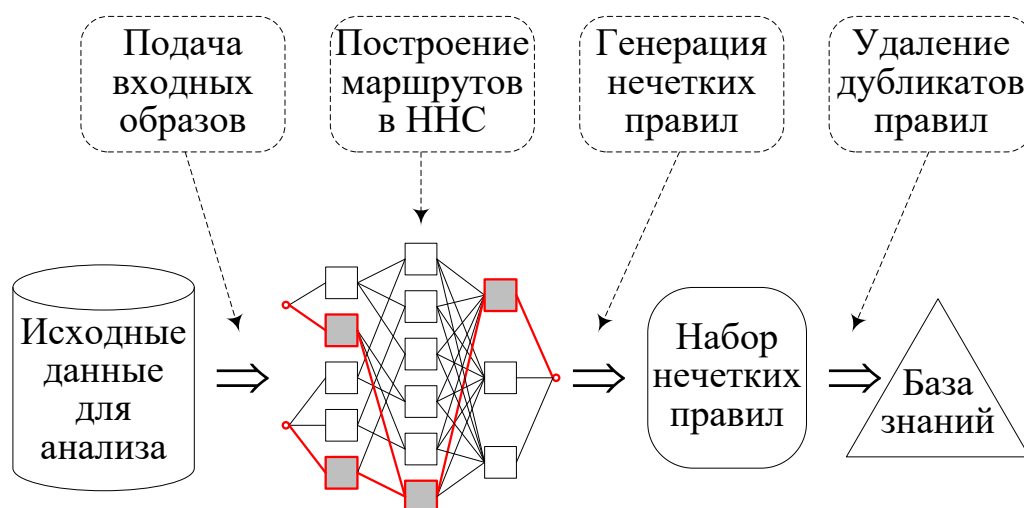


Рисунок 3.13 – Обобщенная схема работы метода формирования базы знаний

Как видно из рисунка, для практического использования разработанного метода маршрутов потребуется наличие всех исходных данных для анализа, из которых были сформированы обучающая, тестовая и валидационная выборки. Использование в методе маршрутов всех исходных данных гарантирует, что сформированная на его основе база знаний будет включать полный состав нечетко-продукционных правил, необходимых для оценки состояния объектов.

В соответствии с представленным рисунком разработанный метод формирования БЗ включает следующие этапы:

- 1) поочередно подать все входные образы из исходных данных на вход нейронечеткой модели;
- 2) для каждого входного образа построить оптимальный маршрут его прохождения по слоям обученной нечеткой нейронной сети (нейронечеткой модели);
- 3) для каждого построенного маршрута сгенерировать соответствующее ему нечетко-продукционное правило;
- 4) в наборе сгенерированных нечетких правил найти и удалить дубликаты (одинаковые правила с идентичными значениями параметров).

Таким образом, после удаления дубликатов оставшиеся нечетко-продукционные правила образуют искомую базу знаний. Рассмотрим описанные этапы разработанного метода формирования базы знаний более подробно.

Пусть  $\{X_1^*, \dots, X_j^*, \dots, X_m^*\}$  – множество входных образов,  $j = 1..m$ , где  $m$  – объем исходных данных. Каждый входной образ  $X_j^* = \{x_{1j}^*, \dots, x_{ij}^*, \dots, x_{nj}^*\}$  представляет собой часть строки в таблице исходных данных, а именно конкретные значения входных переменных в  $j$ -й строке таблицы. При этом каждому набору входных значений соответствует выходное значение  $V_j^*$  (конкретное состояние объекта).

На вход нейронечеткой модели поочередно подаются входные образы  $X_j^*$  из исходных данных. При этом выходное значение модели не рассчитывается. Вместо него подставляется фактическое значение  $V_j^*$  из входного образа  $X_j^*$ . Следовательно, прохождение входного сигнала по слоям обученной нечеткой нейронной сети однозначно задается сочетанием входных значений с выходным результатом. Это позволяет для каждого входного образа построить оптимальный маршрут его прохождения по слоям ННС.

Рассмотрим логику построения оптимального маршрута прохождения входного образа по слоям нейронечеткой модели (см. рис. 3.14).

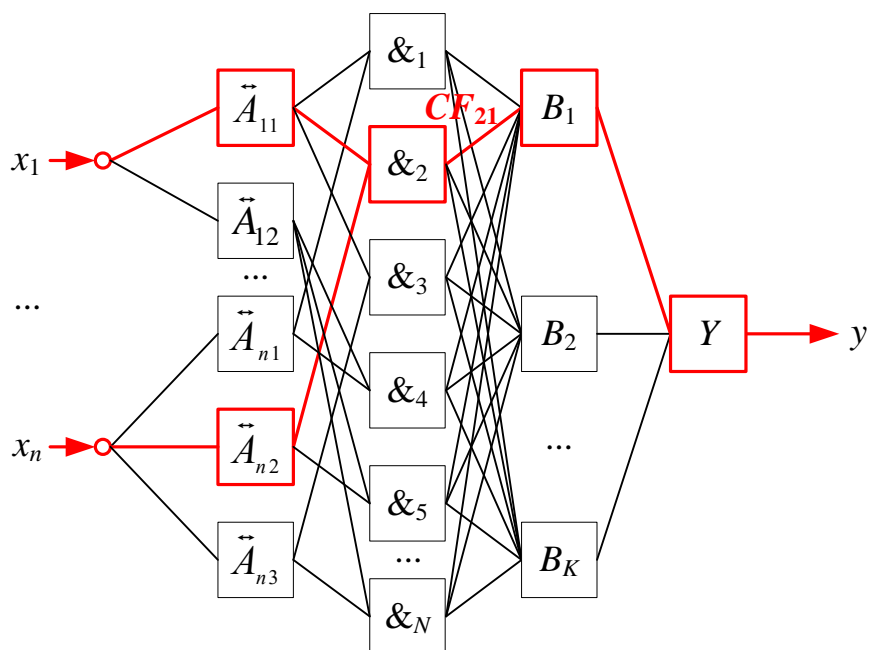


Рисунок 3.14 – Пример прохождения входного образа по слоям нечеткой нейронной сети

Построение маршрута всегда начинается с включения в него всех входных нейронов (левая часть рисунка) и заканчивается включением в него выходного нейрона (правая часть рисунка). Промежуточная часть маршрута между выходным и входными нейронами (строится от выхода ко входам модели) зависит от следующих факторов:

1) фактического выходного значения в таблице данных из множества  $\{B_1, B_2, \dots, B_K\}$ , соответствующего поступившему на вход нейронечеткой модели входного образа;

2) силе связей между выбранным  $B$ -нейроном со всеми  $\&$ -нейронами.

Выбор  $B$ -нейрона производится из таблицы исходных данных и однозначно соответствует вектору входных значений этой же таблицы. В примере на рисунке выше показано, что в качестве выходного значения выбран нейрон  $B_1$ .

Далее необходимо «прогнать» входной сигнал по слоям нейронечеткой модели и рассчитать значения силы связей между выбранным нейроном  $B_1$  и всеми нейронами-антецедентами предыдущего слоя ННС. От нейрона  $B_1$  в направлении слоя антецедентов выбирается связь с максимальным рассчитанным значением. В примере на рисунке показано, что нейрон  $B_1$  связан с нейроном  $\&_2$ . Фактически это означает, что величина силы связи между этими нейронами максимальна.

Построение оставшейся части маршрута производится без дополнительных расчетов и однозначно определяется связями выбранного нейрона-антецедента с нейронами предыдущих слоев. В примере на рисунке показано, что нейрон  $&_2$  связан с нейронами  $\vec{A}_{11}$  и  $\vec{A}_{n2}$ , которые, в свою очередь, связаны с соответствующими входными нейронами  $x_1$  и  $x_n$ .

В итоге, соединив все имеющиеся фрагменты, получим следующий маршрут, представленный на рисунке 3.15.

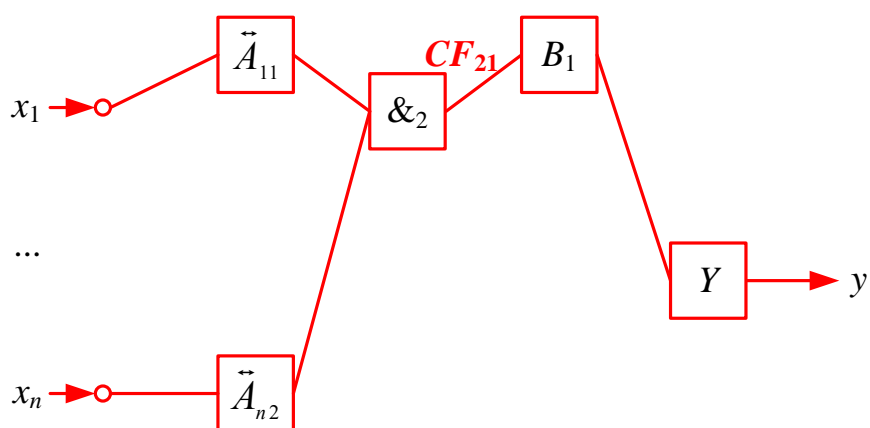


Рисунок 3.15 – Пример построенного маршрута

Таким образом, применение описанного подхода позволяет однозначно построить оптимальный маршрут прохождения каждого входного образа по слоям нейронечеткой модели. Рассмотрим схему генерации нечетко-продукционных правил, соответствующих построенным маршрутам.

Каждый построенный маршрут в нейронечеткой модели однозначно соответствует правилу в БЗ. Следовательно, на основе построенного маршрута можно сгенерировать единственное правило. Рассмотрим процесс генерации правил на примере построенного маршрута.

Для генерации нечетких правил необходимо выполнить следующие этапы:

- 1) декомпонировать построенный маршрут на составляющие его фрагменты;
- 2) для каждого фрагмента сформировать соответствующий ему элемент нечетко-продукционного правила вида (2.6);
- 3) объединить сформированные элементы правил и получить окончательное нечетко-продукционное правило.

Рассмотрим каждый из указанных этапов более подробно.

На рисунке 3.16 представлены отдельные фрагменты построенного маршрута в результате его декомпозиции.

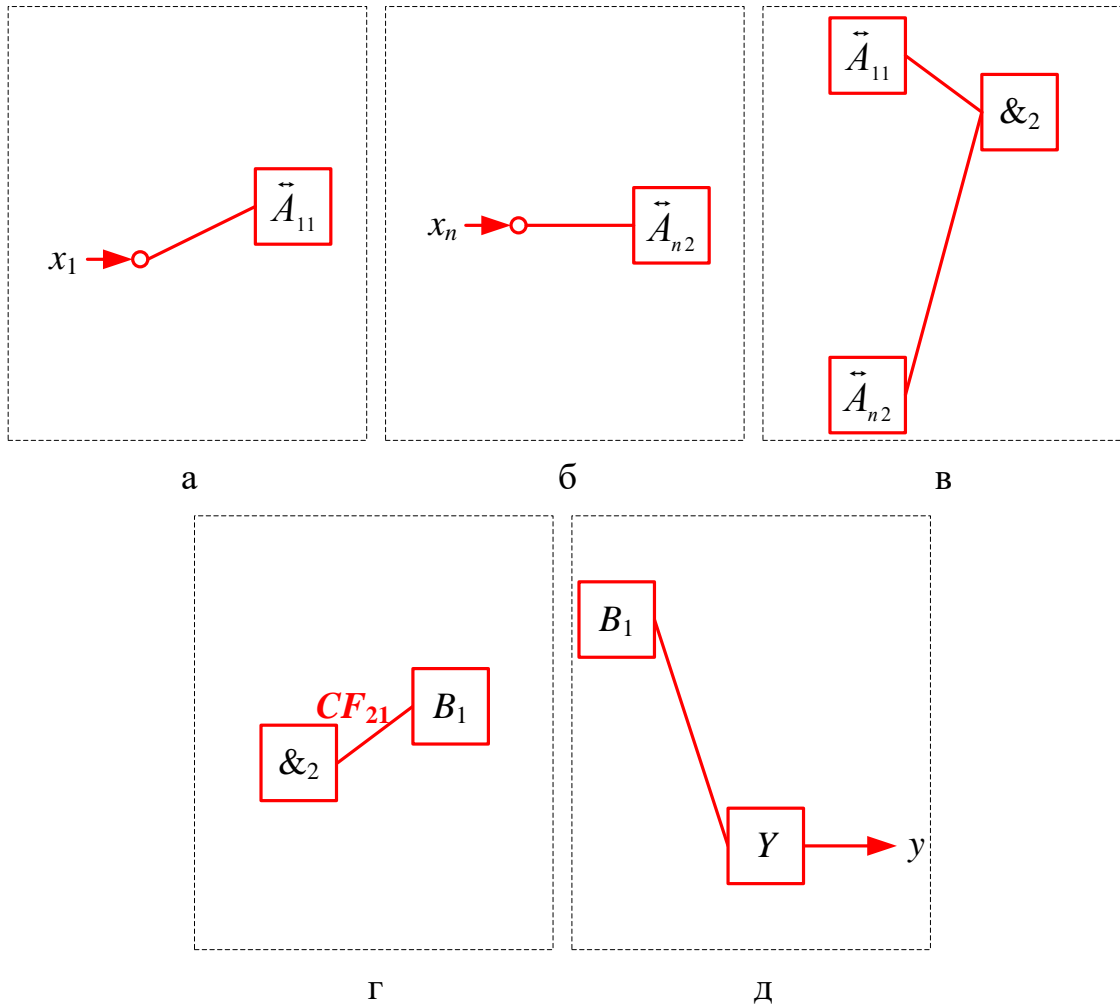


Рисунок 3.16 – Фрагменты построенного маршрута

Первому фрагменту, представленному на рисунке 3.16 (а), соответствует условие нечетко-продукционного правила « $x_1 = \vec{A}_{11}$ ». Второму фрагменту, представленному на рисунке 3.16 (б), соответствует условие правила « $x_n = \vec{A}_{n2}$ ». Третий фрагмент (см. рис. 3.16 (в)) задает antecedent правила « $x_1 = \vec{A}_{11} \&\dots x_n = \vec{A}_{n2}$ ». Четвертому фрагменту (рис. 3.16 (г)), соответствует вес правила « $CF_{21}$ ». Пятый фрагмент, представленный на рисунке 3.16 (д), задает консеквент правила « $y = B_1$ ».

Объединяя сформированные фрагменты, получим следующее нечетко-продукционное правило:

$$\text{If } x_1 = \vec{A}_{11} \&\dots x_n = \vec{A}_{n2} \text{ Then } y = B_1 [CF_{21}].$$

Поступая аналогично для всех построенных маршрутов в нечеткой нейронной сети, соответствующих входным образам из исходных данных, получим соответствующий набор нечетко-продукционных правил.

Следует отметить, что построенные для различных входных образов маршруты, в силу их ограниченного количества, равного значению произведения  $N * K$ , могут повторяться. Соответственно, в этом случае в полученном наборе нечетко-продукционных правил будут встречаться идентичные правила (дубликаты). Такой набор правил, содержащий дубликаты, будет избыточным и, следовательно, неоптимальным. Для сокращения избыточности набора сгенерированных нечетко-продукционных правил необходимо произвести поиск и удалить всех найденных дубликатов. Рассмотрим этап поиска и удаления дубликатов из набора сгенерированных нечетко-продукционных правил.

Отметим, что под дубликатами следует понимать структурно и параметрически совпадающие нечетко-продукционные правила. Структурно совпадающие правила имеют одинаковые антецеденты и консеквенты. При этом в антецедентах таких правил содержатся одни и те же условия, а в консеквентах — одинаковые заключения, соответствующие одному и тому же состоянию объекта. Параметрически совпадающие правила имеют одинаковые значения параметров ФП и весов.

Совпадение указанных факторов позволяет осуществлять поиск дубликатов в наборе сгенерированных нечетко-продукционных правил. Все найденные дубликаты необходимо сгруппировать в отдельные подмножества правил и в каждом подмножестве оставить только одно нечетко-продукционное правило, а остальные удалить. Оставшиеся правила будут составлять искомую БЗ.

На рисунке 3.17 представлена блок-схема работы алгоритма поиска и удаления дубликатов в наборе нечетких правил для получения искомой базы знаний.



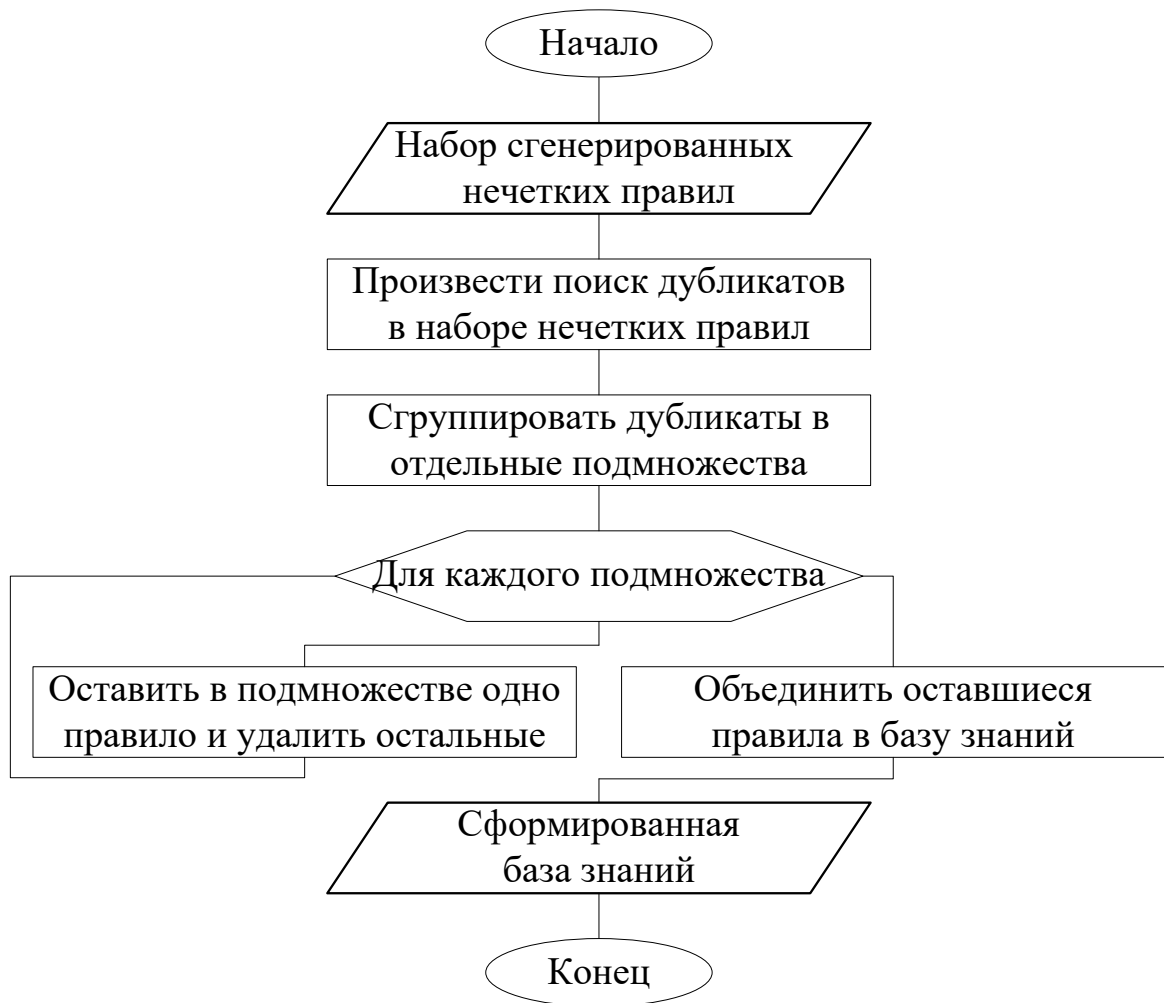


Рисунок 3.17 – Блок-схема алгоритма поиска и удаления дубликатов в наборе сгенерированных нечетких правил

Таким образом, рассмотренный алгоритм позволяет осуществлять поиск и удаление дубликатов в исходном наборе сгенерированных нечетко-продукционных правил, формируя искомую БЗ.

### 3.4.2 Алгоритм формирования базы знаний

Рассмотрим алгоритм формирования БЗ на основе построенной нейронечеткой модели. Данный алгоритм включает следующие этапы:

- 1) получить нейронечеткую модель и все исходные данные для анализа;
- 2) подать очередной входной образ из исходных данных на вход нейронечеткой модели:

2.1) построить оптимальный маршрут прохождения входного образа по слоям нечеткой нейронной сети;

2.2) для построенного маршрута сгенерировать соответствующее ему нечетко-продукционное правило;

3) из набора сгенерированных нечетких правил удалить дубликаты;

4) из оставшихся уникальных нечетко-продукционных правил сформировать искомую БЗ.

На рисунке 3.18 показана блок-схема алгоритма.



Рисунок 3.18 – Блок-схема алгоритма формирования базы знаний на основе построенной нейронечеткой модели

Таким образом, в результате применения метода маршрутов формируется база знаний, состоящая из набора уникальных нечетко-продукционных правил.

### 3.5 Выводы

Обучение, тестирование и валидация нейронечеткой модели потребовали разработки метода сэмплинга данных с учетом особенностей решаемой задачи. Разработанный метод основан на процедуре случайного сэмплинга с замещением и позволяет формировать однородные выборки данных.

Для расчета значений весов нечеткой нейронной сети и, соответственно, нечетко-продукционных правил, разработан эвристический метод. Его особенность заключается в расчете значений весовых коэффициентов, как частоты реализации шаблонов правил в анализируемых данных.

Для построения нейронечеткой модели требуется выполнение этапов инициализации и настройки значений параметров ННС (функций принадлежности и весов). Это потребовало разработки методов и алгоритмов.

Для инициализации значений параметров ФП предложено использовать классический метод равномерного гранулирования. В качестве формализации нечетких гранул предложено использовать треугольные и трапецеидальные ФП. Выбор таких ФП обусловлен широтой и эффективностью их практического использования, а также простотой задания и лингвистической интерпретации человеком, что требуется для принятия обоснованных решений.

Для обучения ННС разработан метод итерационной генетической оптимизации, основанный на применении классического ГА. Для его реализации предложены способы выбора и кодирования значений параметров ФП, участвующих в обучении. Фитнесс-функция и критерий выбора лучшей хромосомы в ГА основаны на максимизации точности классификации ННС на обучающей выборке при условии не уменьшения ее точности классификации на тестовой выборке. Применение генетического алгоритма позволяет настраивать значение параметра одной выбранной ФП и уточнять значения весов в ННС. На основе ГА разработан алгоритм обучения ННС, позволяющий за конечное число шагов настраивать значения параметров каждой ФП. Его сходимость обеспечена свойствами ГА.

После обучения ННС требуется оценить ее обобщающую способность. Для этого предложен алгоритм, основанный на вычислении точности ННС на валидационных данных. Значение точности позволяет судить об адекватности построенной нейронечеткой модели и возможности ее практического использования.

Для формирования БЗ на основе построенной нейронечеткой модели разработан метод маршрутов, позволяющий для каждого входного образа из исходных данных построить оптимальный маршрут его прохождения по слоям обученной ННС, сгенерировать нечеткое правило, соответствующее построенному маршруту, и сформировать искомую БЗ, содержащую уникальные правила. Для его практического использования требуется наличие всех исходных данных. Это гарантирует, что база знаний будет включать полный состав нечетких правил, необходимых для оценки состояния объектов.

## **4 Программный комплекс формирования и использования баз знаний для оценки состояния объектов**

*В этой главе описывается разработанный программный комплекс: общие характеристики, структура и состав компонентов, графический интерфейс, а также алгоритм использования для поддержки принятия решений. Приводятся результаты проведенных исследований: влияния числа градаций, формы функций принадлежности и значений параметров генетического алгоритма на обучение ННС. Даются рекомендации по построению нейронечетких моделей, описываются сформированные базы знаний, приводится сравнение их точности с точностью других методов классификации. Приводятся результаты апробации при формировании баз знаний для подбора геолого-технических мероприятий (ГТМ) на нефтяном месторождении, для оценки функционального состояния водителей автотранспортных средств и для определения фишинговых сайтов.*

### **4.1 Описание разработанного программного комплекса**

#### **4.1.1 Общее описание программного комплекса**

Разработанные методы и алгоритмы реализованы в программном комплексе [91]. Его создание осуществлялось на базе интегрированной среды разработки Visual Studio 2019 [34]. Код программного комплекса написан на объектно-ориентированном языке программирования C# [9].

Следует отметить, что в процессе выполнения диссертационного исследования были реализованы две версии программного комплекса. Первая версия, зарегистрированная в Роспатенте в 2020 году, называется «Программа для формирования баз знаний интеллектуальных систем поддержки принятия решений по оценке состояния объектов» [78]. Однако в процессе дальнейшей работы над диссертацией ряд методов и алгоритмов, реализованных в первой версии программного комплекса, претерпел некоторые изменения. Это потребовало внесения из-

менений в логику работы программы и корректировки ее пользовательского интерфейса. Вторая версия программного комплекса, зарегистрированная в феврале 2022 года, получила уточненное название «Нейронечеткая система формирования баз знаний для оценки состояния объектов» [94]. В этой версии программы в полном объеме реализованы все представленные в диссертации методы и алгоритмы. В дальнейшем при описании разработанного программного комплекса речь пойдет о его текущей версии.

В таблице 4.1 представлено соответствие разработанных методов и алгоритмов задачам, решаемым в диссертации.

Таблица 4.1 – Соответствие методов и алгоритмов решаемым задачам.

Решаемые задачи	Разработанные методы и алгоритмы
Формирование выборок данных для построения нейронечеткой модели	Метод сэмплинга данных
Расчет значений весов связей в нечеткой нейронной сети	Метод расчета значений $CF$
Настройка значений параметров ФП	ГА для настройки значений параметров ФП
Обучение ННС	Метод и алгоритм обучения ННС
Формирование базы знаний на основе построенной нейронечеткой модели	Метод и алгоритм формирования базы знаний
Использование БЗ	Алгоритм логического вывода

Таким образом, программный комплекс позволяет решать задачи, соответствующие разработанному математическому и алгоритмическому обеспечению.

#### 4.1.2 Структура и состав программного комплекса

На рисунке 4.1 показан схема программного комплекса, состоящего из следующих взаимосвязанных программных модулей:

- 1) модуля сэмплинга данных;
- 2) модуля построения нейронечеткой модели;
- 3) модуля формирования БЗ;
- 4) модуля оценки состояния объектов;
- 5) модуля исследований и экспериментов.

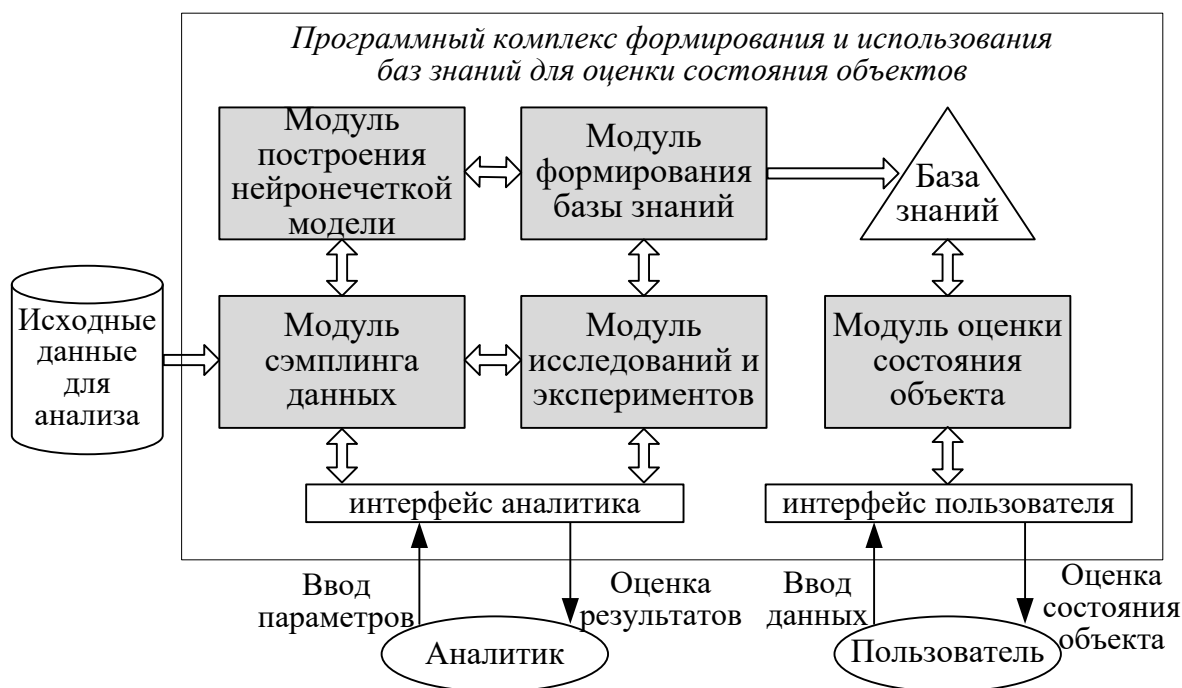


Рисунок 4.1 – Структурная схема разработанного программного комплекса

Рассмотрим программные модули и соответствующие им функциональные задачи (см. табл. 4.2).

Таблица 4.2 – Модули программного комплекса и соответствующие им функциональные задачи

Название программного модуля	Функциональные задачи программного модуля
Модуль сэмплинга данных	- загрузка исходных данных - визуализация загруженных данных - формирование выборок данных
Модуль построения нейронечеткой модели	- построение структуры ННС - инициализация параметров ННС - обучение и валидация ННС - визуализация структуры ННС
Модуль формирования базы знаний	- формирование БЗ - визуализация правил БЗ
Модуль оценки состояния объектов	- использование правил БЗ для оценки состояния объекта - визуализация результатов оценки
Модуль исследований и экспериментов	- исследование нейронечеткой модели и методов ее построения - визуализация результатов исследований

Как следует из таблицы, каждый программный модуль обладает соответствующими функциональными возможностями, в совокупности составляющими функциональные возможности всего программного комплекса [91].

Кроме того, программный комплекс имеет графический интерфейс для работы с ним аналитика и конечного пользователя [91]. Под управлением аналитика производится обучение ННС и формирование БЗ. Пользователь выступает в качестве лица, принимающего решения. Рассмотрим графический интерфейс программного комплекса и логику его функционирования.

### 4.1.3 Графический интерфейс программного комплекса

Интерфейс программного комплекса будем рассматривать с точки зрения заложенной в него логики решения основных задач.

На рисунке 4.2 показано главное окно программы.

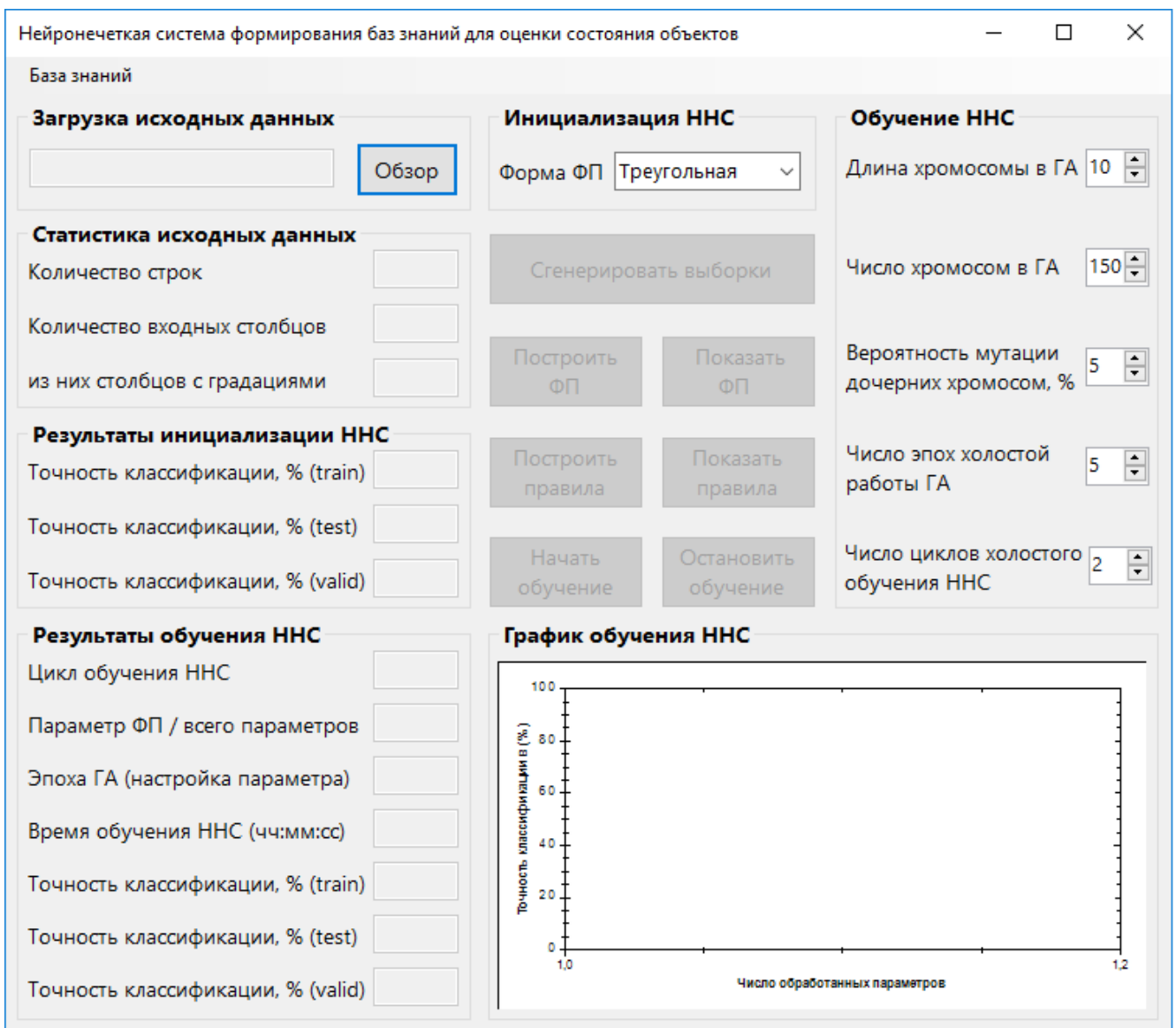


Рисунок 4.2 – Главное окно программного комплекса



Работа программы начинается с загрузки данных для анализа. Для этого необходимо нажать кнопку «Обзор», после чего выбрать нужный файл.

На рисунке 4.3 представлено окно выбора файла с данными для анализа.

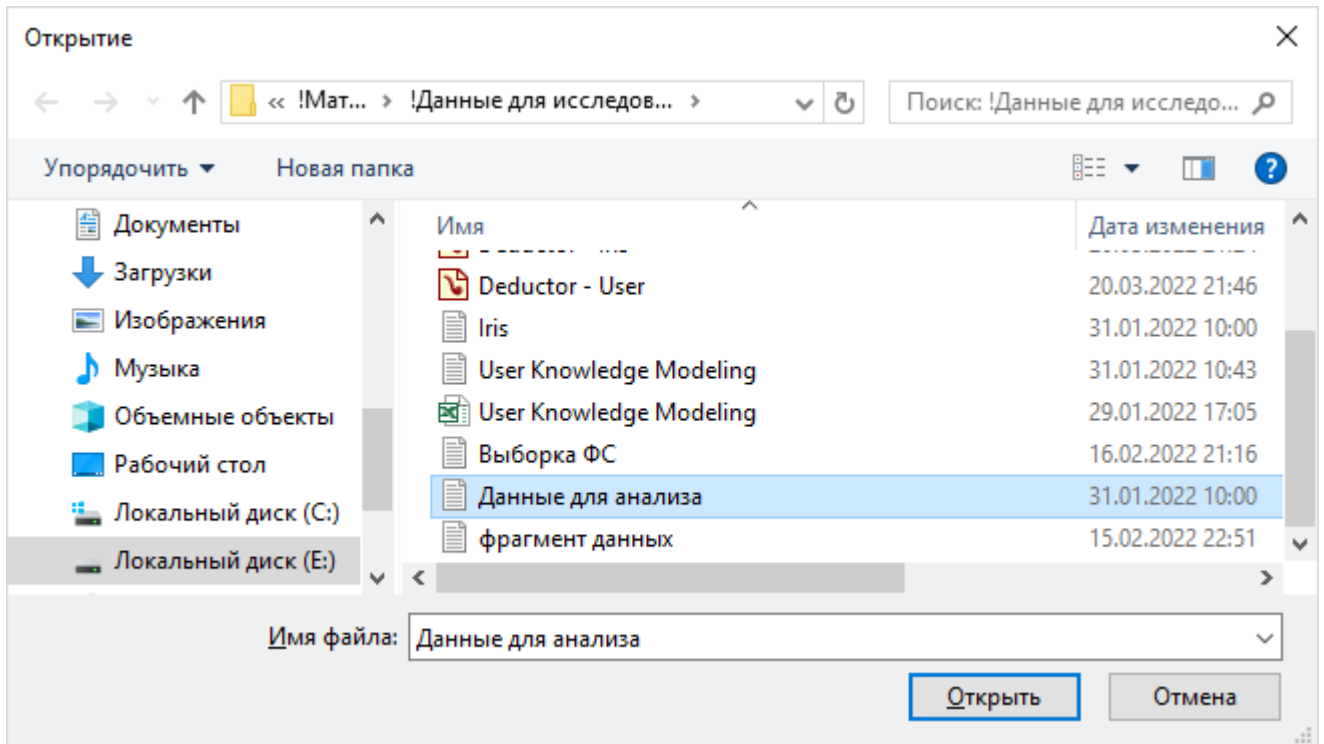


Рисунок 4.3 – Окно выбора файла с исходными данными

После выбора файла появляется следующее окно (см. рис. 4.4).

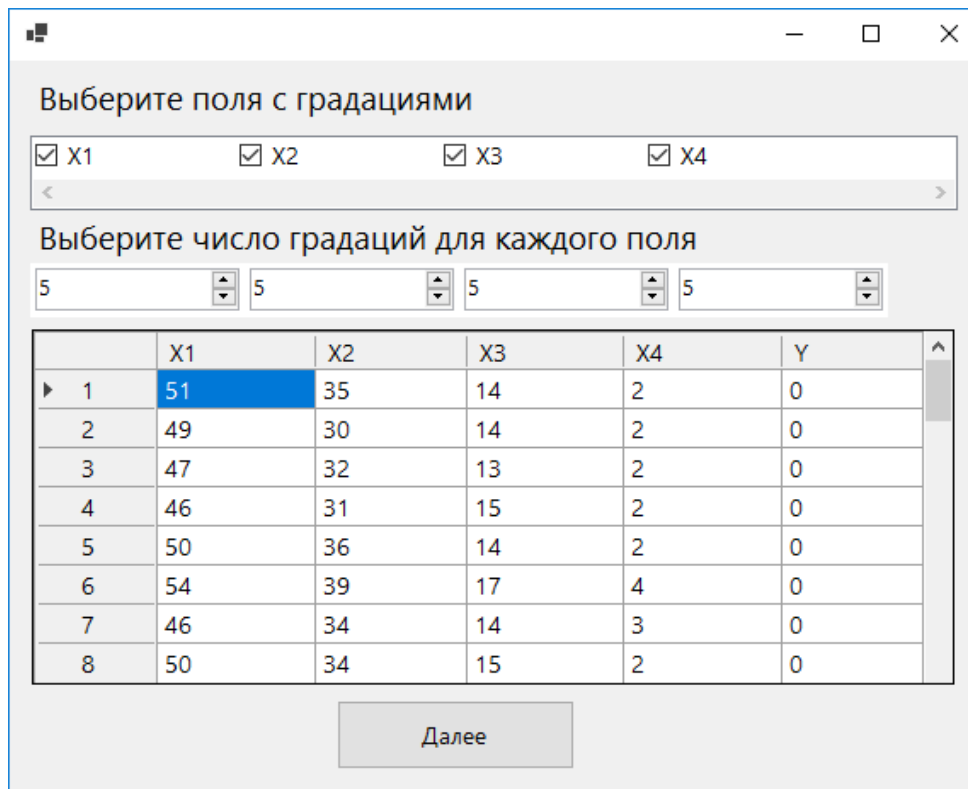


Рисунок 4.4 – Окно выбора градаций и просмотра файла с данными

В этом окне для каждого входного поля (количественной переменной) необходимо выбрать число соответствующих ему нечетких градаций. В данном случае все четыре входных переменных являются количественными (отмечены «галочкой») и для каждой из них выбрано по пять градаций. Кроме того, в окне доступен просмотр содержимого файла. Такой просмотр помогает визуально оценить тип входных данных (количественные или качественные) и принять решение о необходимости их гранулирования.

После выбора числа градаций и нажатия кнопки «Далее» в главном окне программы появляется статистика исходных данных, и становится активной кнопка «Сгенерировать выборки» (см. рис. 4.5).

**Статистика исходных данных**

Количество строк	150	Сгенерировать выборки
Количество входных столбцов	4	
из них столбцов с градациями	4	

Построить ФП      Показать ФП

Рисунок 4.5 – Окно выбора градаций и просмотра файла с данными

При нажатии кнопки «Сгенерировать выборки» происходит формирование обучающей, тестовой и валидационной выборок данных по предложенному в диссертации методу. При этом файлы с выборками автоматически сохраняются в текстовом виде на диске с программой, и становится активной кнопка «Построить ФП», по нажатию которой кнопки «Показать ФП» и «Построить правила» также становятся активными (см. рис. 4.6).

Построить ФП      Показать ФП

Построить правила      Показать правила

Рисунок 4.6 – Активация кнопок «Показать ФП» и «Построить правила»

При нажатии кнопки «Показать ФП» вызывается окно просмотра ФП (см. рис. 4.7).

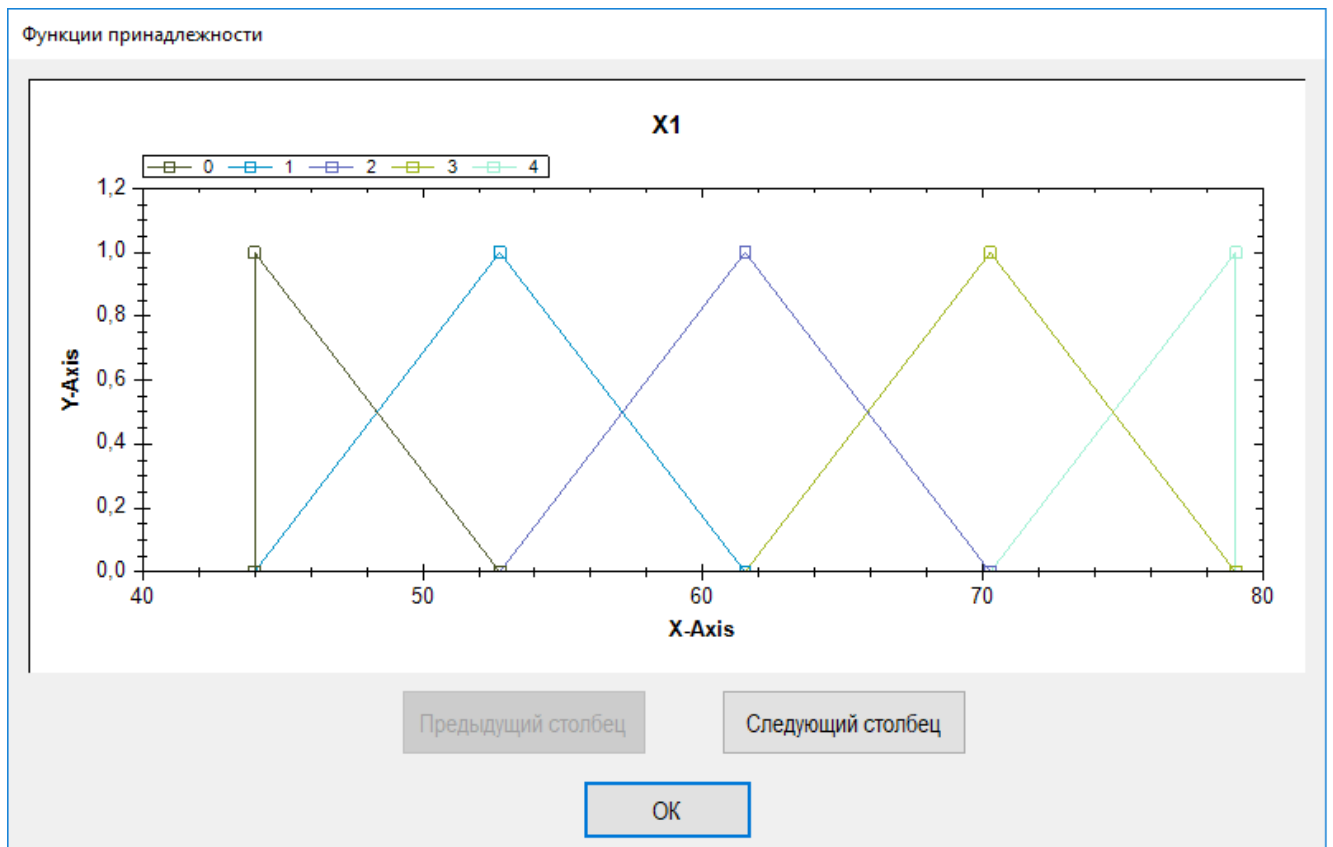


Рисунок 4.7 – Окно просмотра функций принадлежности входных переменных на этапе инициализации нечеткой нейронной сети

В окне представлены пять функций принадлежности для первой входной переменной, равномерно покрывающие область ее значений. Просмотр функций принадлежности для остальных переменных доступен по нажатию кнопок «Следующий столбец» и «Предыдущий столбец».

При нажатии кнопки «Построить правила» появляются результаты инициализации ННС, и становятся активными кнопки «Показать правила» и «Начать обучение» (см. рис. 4.8).

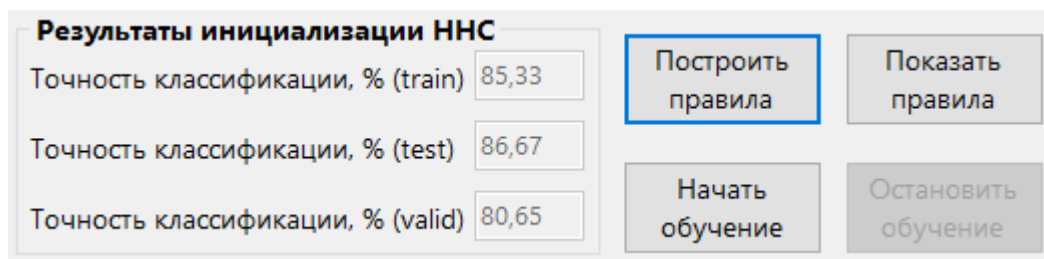


Рисунок 4.8 – Результаты нажатия кнопки «Построить правила»

Результаты инициализации ННС представляют собой значения точности классификации данных из обучающей, тестовой и валидационной выборок, выраженные в процентах.

Нажатие на кнопку «Показать правила» вызывает окно просмотра правил базы знаний (см. рис. 4.9).

The screenshot shows a window titled 'Rules' with a toolbar containing 'Сохранить' and 'Загрузить'. Below the toolbar is a table with 16 rows and 7 columns. The columns are labeled X1, X2, X3, X4, Y, and CF. The first row is highlighted in blue. Below the table is a 'Выход' button.

	X1	X2	X3	X4	Y	CF
1	2	4	1	1	0	0,258
2	3	2	3	3	1	0,255
3	2	2	3	3	1	0,246
4	4	3	4	4	2	0,238
5	3	3	3	3	1	0,221
6	1	3	1	1	0	0,216
7	2	3	1	1	0	0,207
8	4	3	4	5	2	0,189
9	3	2	4	4	2	0,173
10	3	3	4	4	2	0,156
11	2	4	2	1	0	0,149
12	2	3	2	1	0	0,136
13	2	4	1	2	0	0,136
14	4	2	4	4	2	0,129
15	3	3	4	3	1	0,123
16	1	3	2	1	0	0,119

Выход

Рисунок 4.9 – Окно просмотра правил при инициализации нечеткой нейронной сети

В указанном окне представлены правила с четырьмя входными переменными (X1, X2, X3, X3) и одной выходной (Y). Каждому правилу сопоставлен его вес (CF). Значения входных переменных «1», «2», «3», «4», «5» соответствуют номерам их нечетких градаций, а значения выхода «0», «1», «2» – классам решений.

Следует отметить, что сформированный набор правил является предварительным. В процессе обучения ННС эти правила будут уточняться.

Для запуска обучения необходимо нажать кнопку «Начать обучение». Процесс обучения ННС сопровождается соответствующим графиком (см. рис. 4.10).

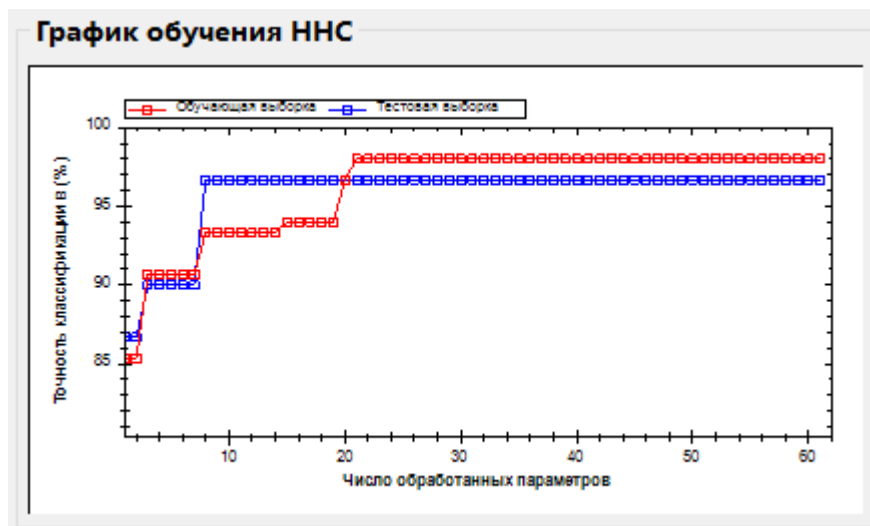


Рисунок 4.10 – График обучения нечеткой нейронной сети

Результаты обучения ННС отображаются в соответствующем разделе главного окна программы (см. рис. 4.11).

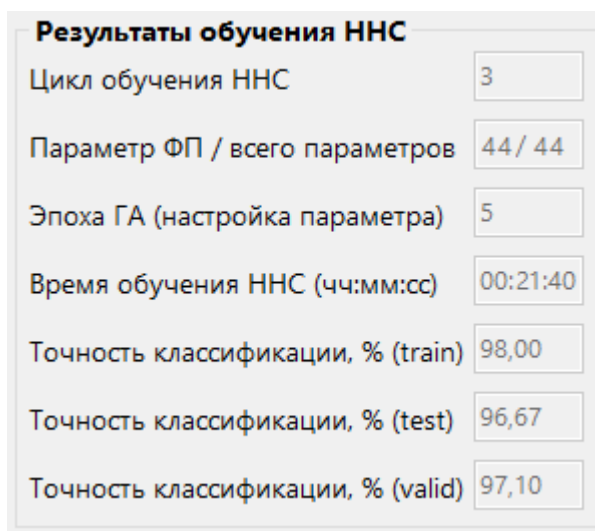


Рисунок 4.11 – Результаты обучения нечеткой нейронной сети

Основными результатами обучения ННС являются достигнутые значения точности классификации на обучающей, тестовой и валидационной выборках данных. Остальные характеристики являются информационными.

На рисунке 4.12 показан пример функций принадлежности с настроенными значениями параметров в результате обучения ННС.

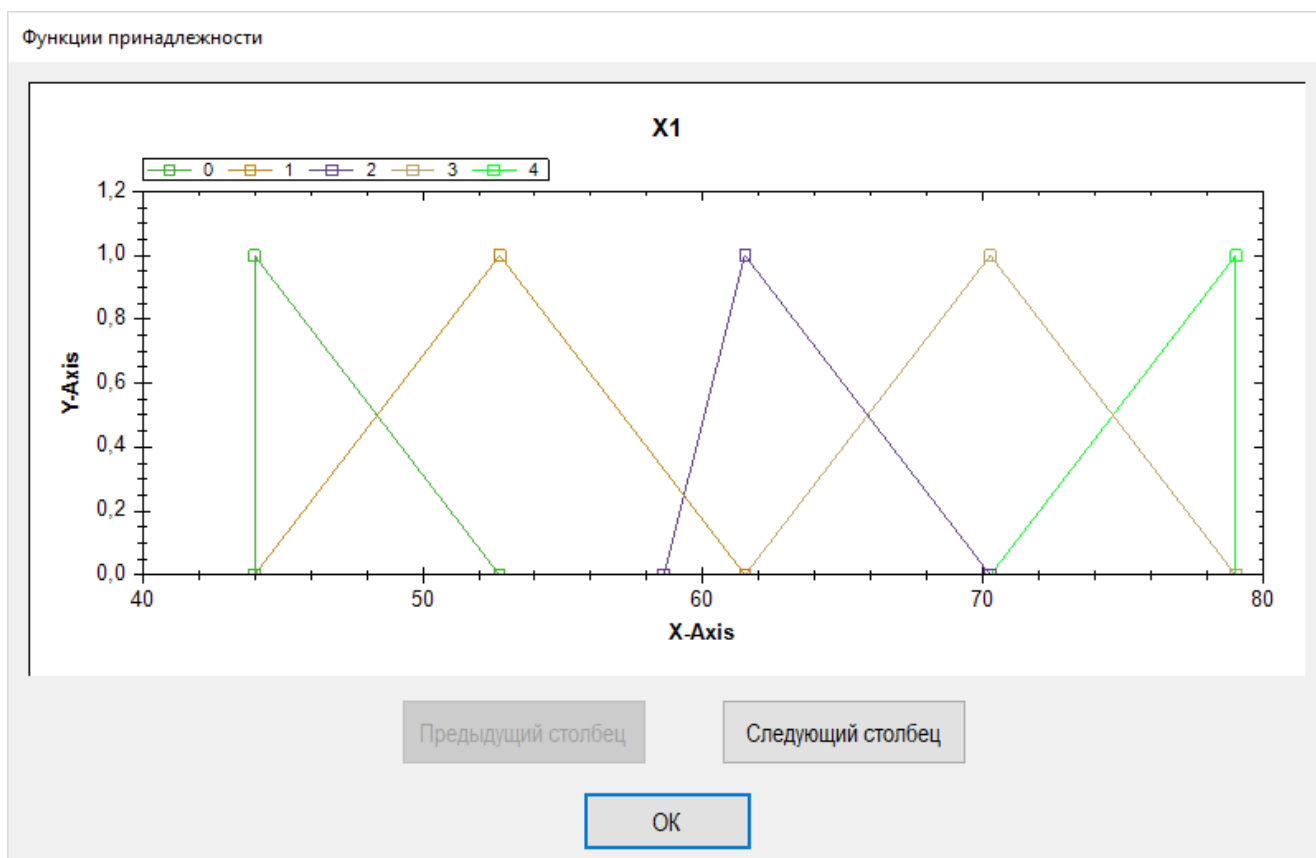


Рисунок 4.12 – Окно просмотра функций принадлежности входных переменных после обучения

После обучения ННС формируется готовая к использованию база знаний, просмотр которой доступен по нажатию кнопки «Показать правила». Для сохранения базы знаний во внешний файл необходимо выбрать соответствующий пункт меню в главном окне программы «База знаний» → «Сохранить базу знаний и функции принадлежности».

Для использования сформированной базы знаний предназначен модуль логического вывода «Logical Conclusion». Данный компонент программного комплекса реализован в виде отдельного программного модуля и позволяет загружать сохраненную базу знаний из файла.

На рисунке 4.13 представлен внешний вид основного окна этого модуля.

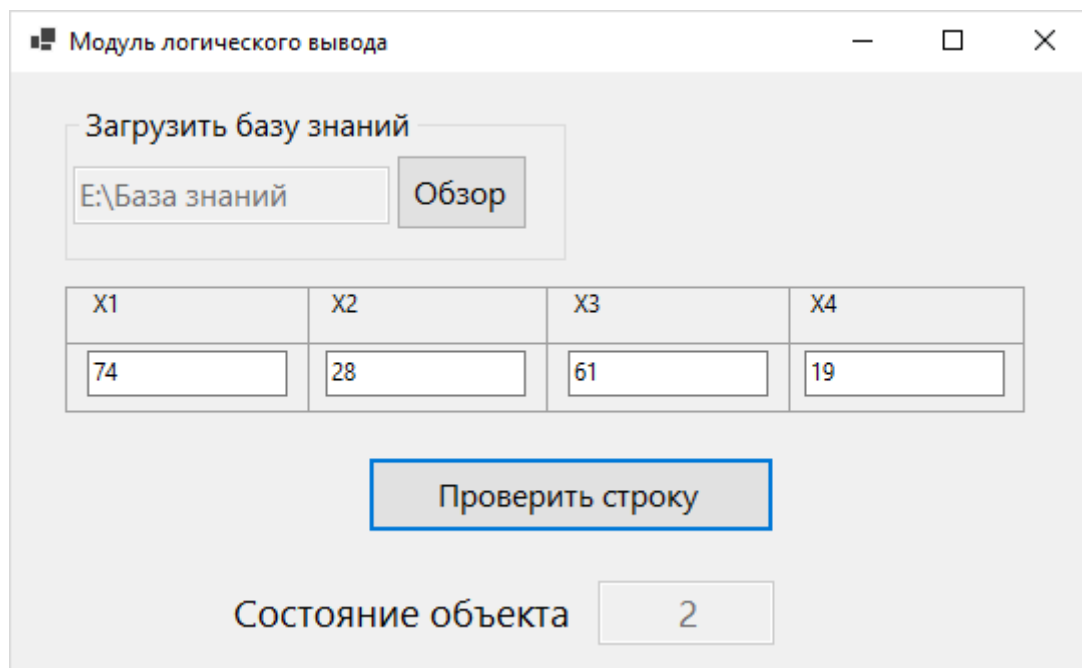


Рисунок 4.13 – Пример работы модуля логического вывода

Таким образом, разработанный программный комплекс имеет дружелюбный интерфейс, позволяющий формировать и использовать БЗ.

#### **4.1.4 Алгоритм использования программного комплекса для поддержки принятия решений по оценке состояния объектов**

Поддержка принятия пользовательских решений на базе программного комплекса потребовала отдельной постановки задачи и разработки соответствующего алгоритма.

Пусть в результате анализа исходных данных и построения нейронечеткой модели сформирована нечеткая БЗ. Требуется разработать обобщенный алгоритм использования программного комплекса и сформированной базы знаний для поддержки принятия решений.

На рисунке 4.14 представлена блок-схема алгоритма.



Рисунок 4.14 – Блок-схема алгоритма поддержки принятия решения по оценке состояния объекта



Рассмотрим этапы работы предложенного алгоритма.

На первом этапе пользователь программного комплекса, выступающий в роли лица, принимающего решения, вводит в программу данные для оценки состояния объекта. На втором этапе пользователь производит запуск модуля оценки состояния объекта. На третьем этапе после выполнения алгоритма логического вывода формируется ранжированный перечень из  $K$  вариантов состояний объекта с соответствующими комплексными оценками их достоверности и нечетко-продукционными правилами для их интерпретации и обоснования предложенных решений. На четвертом этапе для каждого  $k$ -го предложенного варианта ( $k=1..K$ ) производится его экспертная оценка пользователем (ЛПР), который производит оценку данного варианта как на основе анализа комплексных оценок достоверности предлагаемых решений, так и на основе сопоставления выбранных нечетких правил со своим мнением. При этом ЛПР, оценивая предлагаемые варианты решений, опирается на свои знания, опыт и интуицию [58; 67]. На пятом этапе, если вариант выбора решения совпадает с мнением ЛПР, то происходит переход к оценке следующего варианта, иначе пользователь производит корректировку текущего варианта решения. На шестом этапе формируется итоговый перечень вариантов решений с учетом возможных корректировок промежуточных вариантов. На седьмом этапе ЛПР принимает решение – выбирает наиболее достоверный вариант оценки состояния объекта из итогового перечня вариантов.

## **4.2 Исследования, проведенные на базе программного комплекса**

### **4.2.1 Постановка задачи исследований**

Для оценки эффективности разработанных в диссертации и реализованных в программном комплексе методов и алгоритмов, используемых для обучения ННС и формирования нечетких баз знаний, потребовалось проведение следующих исследований:

- исследование влияния числа градаций и формы функций принадлежности на результаты инициализации и обучения ННС;

- исследование влияния значений параметров генетического алгоритма на результаты построения нейронечетких моделей и формирования БЗ;
- оценка точности классификации на основе формируемых баз знаний;
- сравнение точности различных методов классификации.

Следует отметить, что все исследования проводились на компьютере с 4-хядерным процессором Intel(R) Core(TM) i5-7200U с частотой 2.50 ГГц, оперативной памятью 8 Гб, 64-разрядной операционной системой MS Windows 10.

Для проведения исследований потребовалось наличие общедоступных наборов данных. Рассмотрим выбранные наборы данных.

#### 4.2.2 Описание источника и наборов данных для исследований

Существует большое количество общедоступных источников, содержащих данные для анализа [77; 106; 141; 193; 194; 201; 205]. Одним из наиболее известных источников данных является репозиторий UCI [194] (см. рис. 4.15).

The screenshot shows the Machine Learning Repository website. The header includes the title 'Machine Learning Repository' and a link 'View ALL Data Sets'. Below the header, it says 'Browse Through: 622 Data Sets' and 'Table View List View'. A sidebar on the left contains various filters like 'Default Task', 'Attribute Type', 'Data Type', 'Area', '# Attributes', and '# Instances'. The main content is a table listing data sets with columns: Name, Data Types, Default Task, Attribute Types, # Instances, # Attributes, and Year.

Name	Data Types	Default Task	Attribute Types	# Instances	# Attributes	Year
Abalone	Multivariate	Classification	Categorical, Integer, Real	4177	8	1995
Adult	Multivariate	Classification	Categorical, Integer	48842	14	1996
Annealing	Multivariate	Classification	Categorical, Integer, Real	798	38	
Anonymous Microsoft Web Data		Recommender-Systems	Categorical	37711	294	1998
Arrhythmia	Multivariate	Classification	Categorical, Integer, Real	452	279	1998
Artificial Characters	Multivariate	Classification	Categorical, Integer, Real	6000	7	1992
Audiology (Original)	Multivariate	Classification	Categorical	226		1987
Audiology (Standardized)	Multivariate	Classification	Categorical	226	69	1992
Auto MPG	Multivariate	Regression	Categorical	398		1993

Рисунок 4.15 – Главная страница репозитория машинного обучения UCI

Из рисунка видно, что в настоящее время репозиторий машинного обучения UCI содержит 622 наборов данных. Все наборы сгруппированы по задачам исследований, по типу и числу представленных в них входных переменных, по типу данных, по предметной области, по объему данных и др.

Выбор наборов данных для исследований производился на основании следующих критериев, определенных экспертным путем:

- 1) решаемая задача – классификации;
- 2) входные переменные – количественные;
- 3) тип данных – многомерные;
- 4) предметная область – любая;
- 5) число входных переменных – менее 10;
- 6) объем данных – от 100 записей.

На основании указанных критериев выбраны следующие наборы данных, представляющие интерес с точки зрения проводимых исследований:

- Iris Data Set [210];
- Banknote Authentication Data Set [195];
- User Knowledge Modeling Data Set [230].

В таблице 4.3 представлены характеристики наборов данных.

Таблица 4.3 – Характеристики наборов данных для исследований

Название набора данных	Число входных переменных в наборе данных	Объем набора данных	Число классов решений (состояний объектов)
Iris Data Set	4	150	3
Banknote Authentication Data Set	4	1372	2
User Knowledge Modeling Data Set	5	403	4

Набор данных «Iris Data Set» является классическим и часто встречается в литературных источниках [69; 200; 222]. Этот набор содержит характеристики 3-х классов Ирисов Фишера по 50 экземпляров в каждом классе.

На рисунке 4.16 представлен фрагмент данных из этого набора.

X1	X2	X3	X4	Y
51	35	14	2	Iris-setosa
49	30	14	2	Iris-setosa
47	32	13	2	Iris-setosa
46	31	15	2	Iris-setosa
50	36	14	2	Iris-setosa
70	32	47	14	Iris-versicolor
64	32	45	15	Iris-versicolor
69	31	49	15	Iris-versicolor
55	23	40	13	Iris-versicolor
65	28	46	15	Iris-versicolor
63	33	60	25	Iris-virginica
58	27	51	19	Iris-virginica
71	30	59	21	Iris-virginica
63	29	56	18	Iris-virginica
65	30	58	22	Iris-virginica

Рисунок 4.16 – Фрагмент данных из набора «Iris Data Set»

Входными переменными в наборе данных являются:

- 1) X1 = «Длина чашелистика», в мм;
- 2) X2 = «Ширина чашелистика», в мм;
- 3) X3 = «Длина лепестка», в мм;
- 4) X4 = «Ширина лепестка», в мм.

Значениями выходной переменной Y являются следующие классы решений:

- Iris-setosa;
- Iris-versicolor;
- Iris-virginica.

Набор данных «Banknote Authentication Data Set» соответствует задаче определения фальшивых банкнот [185; 202]. Данные этого набора извлечены из изображений, снятых с подлинных банкнот и поддельных образцов, похожих на банкноты. С помощью промышленной видеокамеры получены полутоновые изображения с разрешением около 660 dpi. Для извлечения признаков из изображений использовались Вейвлет-преобразования [54].

На рисунке 4.17 представлен фрагмент данных из этого набора.

X1	X2	X3	X4	Y
3,6216	8,6661	-2,8073	-0,44699	0
4,5459	8,1674	-2,4586	-1,4621	0
3,866	-2,6383	1,9242	0,10645	0
3,4566	9,5228	-4,0112	-3,5944	0
0,32924	-4,4552	4,5718	-0,9888	0
4,3684	9,6718	-3,9606	-3,1625	0
3,5912	3,0129	0,72888	0,56421	0
-3,8167	5,1401	-0,65063	-5,4306	1
-1,9555	0,20692	1,2473	-0,3707	1
-2,1786	-6,4479	6,0344	-0,20777	1
-2,3299	-9,9532	8,4756	-1,8733	1
0,0031201	-4,0061	1,7956	0,91722	1
1,3518	1,0595	-2,3437	0,39998	1
1,2309	3,8923	-4,8277	-4,0069	1

Рисунок 4.17 – Фрагмент данных из набора «Banknote Authentication Data Set»

Входными переменными в наборе данных являются:

- 1) X1 = «Дисперсия вейвлет-преобразованного изображения»;
- 2) X2 = «Асимметрия вейвлет-преобразованного изображения»;
- 3) X3 = «Экссесс вейвлет-преобразованного изображения»;
- 4) X4 = «Энтропия изображения».

Значениями выходной переменной Y являются следующие классы решений:

- подлинная банкнота (0);
- поддельная банкнота (1).

Набор данных «User Knowledge Modeling Data Set» соответствует задаче классификации знаний студентов (на примере оценивания знаний учащихся по теме «Электрические машины постоянного тока») [215].

На рисунке 4.18 представлен фрагмент данных из этого набора.

X1	X2	X3	X4	X5	Y
0	0	0	0	0	Very Low
0,08	0,08	0,1	0,24	0,9	High
0,06	0,06	0,05	0,25	0,33	Low
0,1	0,1	0,15	0,65	0,3	Middle
0,08	0,08	0,08	0,98	0,24	Low
0,09	0,15	0,4	0,1	0,66	Middle
0,1	0,1	0,43	0,29	0,56	Middle
0,15	0,02	0,34	0,4	0,01	Very Low
0,2	0,14	0,35	0,72	0,25	Low
0	0	0,5	0,2	0,85	High
0,18	0,18	0,55	0,3	0,81	High
0,06	0,06	0,51	0,41	0,3	Low
0,1	0,1	0,52	0,78	0,34	Middle
0,1	0,1	0,7	0,15	0,9	High
0,2	0,2	0,7	0,3	0,6	Middle

Рисунок 4.18 – Фрагмент данных из набора «User Knowledge Modeling Data Set»

Входными переменными в наборе данных являются:

- 1) X1 = «Доля времени на изучение материала»;
- 2) X2 = «Доля числа повторений основного учебного материала»;
- 3) X3 = «Степень изучения дополнительного учебного материала»;
- 4) X4 = «Результат проверки знаний по дополнительному материалу»;
- 5) X5 = «Результат проверки знаний по основному учебному материалу».

Значениями выходной переменной Y являются следующие классы решений:

- очень низкий уровень знаний (Very Low);
- низкий уровень знаний (Low);
- средний уровень знаний (Middle);
- высокий уровень знаний (High).

Таким образом, выбрано три набора данных для исследований, один из которых является классическим, а другие содержат в себе реальные данные из различных предметных областей.

### 4.2.3 Влияние числа градаций и формы функций принадлежности на результаты инициализации и обучения нечетких нейронных сетей

Как известно, ННС являются эффективными функциональными аппроксиматорами [126; 127; 188]. При построении нейронечетких моделей выбранные для исследования наборы данных аппроксимировались нечеткими градациями входных переменных с заданной формой функций принадлежности.

Рассмотрим для каждого набора данных, как влияет число градаций входных переменных и формы функций принадлежности на результаты инициализации ННС (начальная точность классификации на обучающей, тестовой и валидационной выборках данных), а также на результаты обучения ННС (время обучения и итоговая точность классификации на указанных выборках данных).

На этом этапе исследований при вариации числа градаций входных переменных от 2-х до 9-ти и поочередном выборе формы функций принадлежности для каждой вариации (треугольной или трапецеидальной) производилось обучение нечеткой нейронной сети и фиксирование значений соответствующих параметров точности и времени ее обучения. При этом экспертным путем были выбраны следующие значения параметров генетического алгоритма:

- длина хромосомы: 10;
- число хромосом в ГА: 50;
- вероятность мутации генов дочерних хромосом: 5%;
- число эпох холостой работы ГА: 5;
- число циклов холостого обучения ННС: 2.

Длина хромосомы  $m$  определяет максимальное число возможных значений для настраиваемых параметров ФП, как  $2^m$ . В данном случае это означает, что любой настраиваемый параметр сможет принимать одно из  $2^{10}=1024$  значений.

Число хромосом в генетическом алгоритме задает размер начальной популяции хромосом и всех последующих текущих популяций.

Вероятность мутации генов дочерних хромосом влияет на мутацию новых хромосом в популяции.

Число эпох холостой работы генетического алгоритма является критерием для его остановки: алгоритм будет пытаться найти лучшее значение параметра для выбранной функции принадлежности до тех пор, пока в течение заданного числа эпох лучшее найденное значение не будет улучшаться.

Число циклов холостого обучения нечеткой нейронной сети также является одним из критериев остановки обучения: если в течение заданного числа циклов обучения не будет повышаться точность классификации на обучающей выборке данных, то процесс обучения сети завершится.

Выбранные и зафиксированные значения параметров генетического алгоритма не являются оптимальными и используются только на текущем этапе исследования по оценке влияния числа градаций и формы функций принадлежности на результаты инициализации и обучения ННС. По результатам дальнейших исследований значения указанных параметров ГА будут уточнены.

Следует отметить, что для повышения достоверности результатов все исследования проводились многократно (не менее 10 раз для каждой комбинации выбираемых значений параметров), а получаемые результаты усреднялись.

В таблицах 4.4÷4.6 представлены зависимости влияния числа градаций входных переменных и формы ФП на результаты инициализации ННС.

Таблица 4.4 – Влияние числа градаций и формы функций принадлежности на инициализацию нечетких нейронных сетей для набора данных «Iris Data Set»

Форма функции принадлежности	Число градаций входных переменных							
	2	3	4	5	6	7	8	9
Точность классификации на обучающих данных, %								
треугольная	77,3	79,3	80	90,7	93,3	94,7	96	96
трапецеидальная	78,6	82	82,7	91,3	95,3	96,3	96,6	96,6
Точность классификации на тестовых данных, %								
треугольная	79,3	80,8	82,6	88	93,1	95,8	96,6	96,6
трапецеидальная	79,3	81,5	83,3	92,9	96,7	96,7	97,3	98
Точность классификации на валидационных данных, %								
треугольная	77,8	80	83,3	87,1	92,9	93,8	96,3	96,3
трапецеидальная	79,2	81,5	84,6	90	96,3	96,4	96,6	96,8



Таблица 4.5 – Влияние числа градаций входных переменных и формы функций принадлежности на результаты инициализации нечетких нейронных сетей для набора данных «Banknote Authentication Data Set»

Форма функции принадлежности	Число градаций входных переменных							
	2	3	4	5	6	7	8	9
Точность классификации на обучающих данных, %								
треугольная	73,6	75,1	86,2	93,6	95,3	97,1	97,7	98,3
трапецеидальная	74,9	75,4	86,5	93,7	95,9	97,2	98,6	98,8
Точность классификации на тестовых данных, %								
треугольная	72,2	73	85,6	91,7	93,8	96,1	96,7	97,2
трапецеидальная	72,8	74,9	87,3	92,8	95,2	97,2	98,6	98,8
Точность классификации на валидационных данных, %								
треугольная	70,4	75	81,6	91,5	94,2	94,8	97,2	98,6
трапецеидальная	72,8	75,6	85,8	93,7	96,1	97,6	97,7	98,6

Таблица 4.6 – Влияние числа градаций входных переменных и формы функций принадлежности на результаты инициализации нечетких нейронных сетей для набора данных «User Knowledge Modeling Data Set»

Форма функции принадлежности	Число градаций входных переменных							
	2	3	4	5	6	7	8	9
Точность классификации на обучающих данных, %								
треугольная	52	59,1	63,6	69	73,5	76,7	83	89,1
трапецеидальная	52,1	60,1	67,1	73,3	75,6	77,1	83,8	90,3
Точность классификации на тестовых данных, %								
треугольная	52,3	55,6	60,5	68,9	70,2	75,6	84,2	87,2
трапецеидальная	54,3	57,4	61,7	70,5	74,5	78,3	88	89,5
Точность классификации на валидационных данных, %								
треугольная	52	56,8	61,2	66	69,2	75,6	84,6	89,1
трапецеидальная	52,3	58,5	62,5	68,2	73,9	78,7	82,4	89,5

Из таблиц видно, что увеличение числа градаций, а также использование трапецеидальных ФП приводит к повышению точности классификации на всех наборах данных при инициализации ННС.

На наборе данных «Iris Data Set» при использовании треугольных ФП точность классификации в среднем по всем выборкам изменялась от 78,1% (для 2-х

градаций) до 96,6% (для 9-ти градаций). При использовании трапецеидальных ФП точность классификации, соответственно, изменялась от 79% до 97,1%.

На наборе данных «Banknote Authentication Data Set» при использовании треугольных ФП точность классификации в среднем по всем выборкам изменялась от 72,1% (для 2-х градаций) до 98% (для 9-ти градаций). При использовании трапецеидальных ФП – от 73,5% до 98,7%.

На наборе данных «User Knowledge Modeling Data Set» при использовании треугольных ФП точность классификации в среднем по всем выборкам изменялась от 52,1% (для 2-х градаций) до 88,5% (для 9-ти градаций). При использовании трапецеидальных ФП – от 52,9% до 89,8%.

В таблицах 4.7÷4.9 представлены зависимости влияния числа градаций входных переменных и формы функций принадлежности на результаты обучения ННС для различных наборов данных.

Таблица 4.7 – Влияние числа градаций и формы функций принадлежности на результаты обучения для набора данных «Iris Data Set»

Форма функции принадлежности	Число градаций входных переменных							
	2	3	4	5	6	7	8	9
Точность классификации на обучающих данных (повышение точности), %								
треугольная	80 (2,7)	94,7 (15,4)	97,3 (17,3)	<b>98,7</b> <b>(8)</b>	98,7 (5,4)	98,6 (3,9)	98,7 (2,7)	98,7 (2,7)
трапецеидальная	82,7 (4,1)	96,7 (14,7)	98,5 (15,8)	<b>99,1</b> <b>(7,8)</b>	99,1 (3,8)	99,1 (2,8)	99,1 (2,5)	99,1 (2,5)
Точность классификации на тестовых данных (повышение точности), %								
треугольная	82,6 (3,3)	92,9 (12,1)	95,8 (13,2)	<b>96,7</b> <b>(8,7)</b>	96,7 (3,6)	96,6 (0,8)	96,7 (0,1)	96,7 (0,1)
трапецеидальная	84,6 (5,3)	96,2 (14,7)	98,7 (15,4)	<b>99,3</b> <b>(6,4)</b>	99,3 (2,6)	99,3 (2,6)	99,3 (2)	99,3 (1,3)
Точность классификации на валидационных данных (повышение точности), %								
треугольная	83,3 (5,5)	93,1 (13,1)	96,2 (12,9)	<b>96,4</b> <b>(9,3)</b>	96,4 (3,5)	96,4 (2,6)	96,4 (0,1)	96,4 (0)
трапецеидальная	87,1 (7,9)	96,7 (15,2)	97,8 (13,2)	<b>98,1</b> <b>(8,1)</b>	98,1 (1,8)	98 (1,6)	98,1 (1,5)	98,1 (1,3)
Время обучения								
треугольная	00:00:38	00:01:11	00:02:59	<b>00:06:06</b>	00:13:14	00:28:47	00:51:14	01:18:12
трапецеидальная	00:01:05	00:01:49	00:03:52	<b>00:06:58</b>	00:14:08	00:31:14	00:58:33	01:33:59

Таблица 4.8 – Влияние числа градаций и формы функций принадлежности на результаты обучения для набора данных «Banknote Authentication Data Set»

Форма функции принадлежности	Число градаций входных переменных							
	2	3	4	5	6	7	8	9
Точность классификации на обучающих данных (повышение точности), %								
треугольная	86,5 (12,9)	94,1 (19)	96,1 (9,9)	98,6 (5)	<b>98,7</b> <b>(3,4)</b>	98,7 (1,6)	98,7 (1)	98,6 (0,3)
трапецидальная	87,3 (12,4)	95,5 (20,1)	98,3 (11,8)	99,1 (5,4)	<b>99,5</b> <b>(3,6)</b>	99,5 (2,3)	99,5 (0,9)	99,5 (0,7)
Точность классификации на тестовых данных (повышение точности), %								
треугольная	85,8 (13,6)	90,2 (17,2)	95,4 (9,8)	97,6 (5,9)	<b>97,9</b> <b>(4,1)</b>	97,9 (1,8)	97,9 (1,2)	97,9 (0,7)
трапецидальная	86,7 (13,9)	96,1 (21,2)	97,2 (9,9)	98,5 (5,7)	<b>98,8</b> <b>(3,6)</b>	98,7 (1,5)	98,8 (0,2)	98,8 (0)
Точность классификации на валидационных данных (повышение точности), %								
треугольная	85,8 (15,4)	94,1 (19,1)	97,2 (15,6)	98,4 (6,9)	<b>98,8</b> <b>(4,6)</b>	98,8 (4)	98,8 (1,6)	98,7 (0,1)
трапецидальная	86,5 (13,7)	95,2 (19,6)	98,3 (12,5)	98,8 (5,1)	<b>99,1</b> <b>(3)</b>	99,1 (1,5)	99,1 (1,4)	99,1 (0,5)
Время обучения								
треугольная	00:02:19	00:05:24	00:23:25	00:44:55	<b>01:06:14</b>	01:52:03	03:29:39	07:12:37
трапецидальная	00:03:07	00:08:09	00:28:36	00:53:48	<b>01:18:24</b>	02:12:23	03:42:08	07:31:15

Таблица 4.9 – Влияние числа градаций и формы функций принадлежности на результаты обучения для набора данных «User Knowledge Modeling Data Set»

Форма функции принадлежности	Число градаций входных переменных							
	2	3	4	5	6	7	8	9
Точность классификации на обучающих данных (повышение точности), %								
треугольная	65,6 (13,6)	80,2 (21,1)	86,5 (22,9)	<b>94,6</b> <b>(25,6)</b>	94,6 (21,1)	94,6 (17,9)	94,5 (11,5)	н/д
трапецидальная	67,1 (15)	83,1 (23)	89,1 (22)	<b>96,8</b> <b>(23,5)</b>	96,8 (21,2)	96,8 (19,7)	96,8 (13)	н/д
Точность классификации на тестовых данных (повышение точности), %								
треугольная	61,7 (9,4)	79,2 (23,6)	84,4 (23,9)	<b>93,8</b> <b>(24,9)</b>	93,8 (23,6)	93,7 (18,1)	93,8 (9,6)	н/д
трапецидальная	64,8 (10,5)	80,1 (22,7)	86,7 (25)	<b>95,3</b> <b>(24,8)</b>	95,3 (20,8)	95,3 (17)	95,3 (7,3)	н/д
Точность классификации на валидационных данных (повышение точности), %								
треугольная	62,5 (10,5)	74,2 (17,4)	82,6 (21,4)	<b>94,3</b> <b>(28,3)</b>	94,3 (25,1)	94,3 (18,7)	94,3 (9,7)	н/д
трапецидальная	65,2 (12,9)	77,1 (18,6)	84,7 (22,2)	<b>95,3</b> <b>(27,1)</b>	95,3 (21,4)	95,3 (16,6)	95,2 (12,8)	н/д
Время обучения								
треугольная	00:02:25	00:05:44	00:21:23	<b>01:51:02</b>	03:52:12	05:39:01	08:13:54	н/д
трапецидальная	00:02:49	00:07:06	00:33:10	<b>02:32:18</b>	04:38:29	06:14:32	08:40:18	н/д

Обучение ННС позволило повысить точность классификации на всех наборах данных по сравнению с точностью на этапе инициализации. При этом наблюдается тенденция повышения точности классификации и времени обучения сети от увеличения числа градаций и использования формы функций принадлежности.

В частности, при обучении ННС на данных из набора «Iris Data Set», выборе 5 градаций входных переменных и использовании треугольных ФП точность классификации по сравнению с точностью, достигнутой на этапе инициализации, в среднем по всем выборкам повысилась на 8,7%. При тех же условиях, но, используя трапецеидальные ФП, средняя точность повысилась на 7,4%. При этом достигнутая точность классификации при 5 градациях и использовании треугольных ФП в среднем составила 97,3% против 98,8% для трапецеидальных ФП.

Следует отметить, что, исходя из полученных результатов, для набора данных «Iris Data Set» целесообразно использование 5 градаций. В этом случае достигается максимальная точность классификации при минимальном времени обучения ННС (от 6 до 7 минут в зависимости от формы ФП). Дальнейшее увеличение числа градаций не приводит к повышению точности классификации, но требует значительно большего времени на обучение ННС: от 13÷14 минут для 6 градаций до полутора часов для 9 градаций.

Кроме того, увеличение числа градаций приводит к существенному увеличению правил в базе знаний. Так, для набора данных «Iris Data Set» при 5 градациях для каждой из 4 входных переменных и 3 состояний объекта предельное число правил составляет  $5^4 \cdot 3 = 1875$ . А для 9 градаций –  $9^4 \cdot 3 = 19683$  правила. При этом реальное число правил в базе знаний составляет не более 10% от их предельного количества, так как большинство сформированных правил имеют нулевую степень достоверности. В любом случае увеличение числа градаций приводит к «взрывному» росту числа правил, что ухудшает интерпретируемость базы знаний и увеличивает время на выполнение логического вывода.

При обучении ННС на данных из набора «Banknote Authentication Data Set», выборе 6 градаций входных переменных и использовании треугольных ФП точность классификации по сравнению с точностью, достигнутой на этапе инициали-

зации, в среднем по всем выборкам повысилась на 2,9%. При тех же условиях, но, используя трапецеидальные ФП, средняя точность повысилась на 3,4%. При этом достигнутая точность классификации при 6 градациях и использовании треугольных ФП в среднем составила 98,5% против 99,1% для трапецеидальных ФП.

Исходя из полученных результатов, для набора данных «Banknote Authentication Data Set» целесообразно использование 6 градаций. В этом случае достигается максимальная точность классификации при минимальном времени обучения ННС (от 1 часа 6 минут до 1 часа 18 минут в зависимости от формы ФП). Дальнейшее увеличение числа градаций не приводит к повышению точности классификации, но требует значительно большего времени на обучение ННС: от двух часов для 7 градаций до семи с половиной часов для 9 градаций.

При обучении ННС на данных из набора «User Knowledge Modeling Data Set», выборе 5 градаций входных переменных и использовании треугольных ФП точность классификации по сравнению с точностью, достигнутой на этапе инициализации, в среднем по всем выборкам повысилась на 26,3%. При тех же условиях, но, используя трапецеидальные ФП, средняя точность повысилась на 25,1%. При этом достигнутая точность классификации при 5 градациях и использовании треугольных ФП в среднем составила 94,2% против 95,8% для трапецеидальных ФП.

Для набора данных «User Knowledge Modeling Data Set» целесообразно использование 5 градаций. В этом случае достигается максимальная точность классификации при минимальном времени обучения ННС (от двух до двух с половиной часов в зависимости от формы ФП). Дальнейшее увеличение числа градаций не приводит к повышению точности классификации, но требует значительно большего времени на обучение ННС: от четырех часов для 6 градаций до восьми с половиной часов для 8 градаций. Следует отметить, что при использовании 9 градаций ННС обучить не удалось из-за ограниченности вычислительных ресурсов.

Таким образом, полученные результаты на текущем этапе исследований позволяют сделать вывод, что при инициализации и обучении ННС целесообразно выбирать не более 5÷6 градаций входных переменных, а для формализации градаций использовать трапецеидальные ФП.

#### 4.2.4 Влияние значений параметров генетического алгоритма на результаты обучения нечетких нейронных сетей

Рассмотрим для каждого набора данных, как влияют значения параметров ГА на результаты обучения ННС (точность классификации на обучающей, тестовой и валидационной выборках данных, а также время обучения сети).

При инициализации ННС для входных количественных переменных выбиралось по 5 или 6 градаций (в зависимости от набора данных) с трапецеидальными ФП. При обучении ННС экспертным путем выбирались следующие значения параметров ГА (при фиксированных значениях остальных параметров):

- длина хромосомы: 8, 9, 10, 11, 12;
- число хромосом в ГА: 50, 100, 150, 200, 250;
- вероятность мутации генов дочерних хромосом: 1%, 2%, 3%, 4%, 5%;
- число эпох холостой работы ГА: 5, 10, 15, 20;
- число циклов холостого обучения ННС: 2, 4, 6, 8.

В таблице 4.10 представлены зависимости влияния длины хромосомы на результаты обучения ННС для различных наборов данных.

Таблица 4.10 – Влияние длины хромосомы на результаты обучения нечетких нейронных сетей

Набор данных	Длина хромосомы				
	8	9	10	11	12
Точность классификации на обучающих данных, %					
Iris Data Set	92,8	98,3	<b>99,1</b>	99,1	99,1
Banknote Authentication Data Set	93,7	98,8	<b>99,5</b>	99,5	99,5
User Knowledge Modeling Data Set	89,6	92,1	<b>96,8</b>	96,8	96,6
Точность классификации на тестовых данных, %					
Iris Data Set	92,3	96,7	<b>99,3</b>	99,3	99,2
Banknote Authentication Data Set	93,5	97,2	<b>98,8</b>	98,8	98,8
User Knowledge Modeling Data Set	89,2	92,3	<b>95,3</b>	95,3	95,2
Точность классификации на валидационных данных, %					
Iris Data Set	93,1	96,4	<b>98,1</b>	98,1	98,1
Banknote Authentication Data Set	93,4	97,6	<b>99,1</b>	99,1	99,1
User Knowledge Modeling Data Set	88,7	93,1	<b>95,3</b>	95,2	95,3
Время обучения					
Iris Data Set	00:04:17	00:05:57	<b>00:06:58</b>	00:07:40	00:08:04
Banknote Authentication Data Set	01:07:43	01:13:18	<b>01:18:24</b>	01:24:19	01:29:06
User Knowledge Modeling Data Set	02:20:08	02:26:15	<b>02:32:18</b>	02:36:49	02:40:53

Видно, что увеличение длины хромосомы в ГА приводит к увеличению времени обучения ННС, а также к увеличению точности классификации. Так, для набора данных «Iris Data Set» время обучения увеличилось примерно на 100%, для набора данных «Banknote Authentication Data Set» – примерно на 31%, а для набора «User Knowledge Modeling Data Set» – примерно на 15%. При этом максимальная точность классификации данных из каждого набора достигается при длине хромосомы, равной 10. Дальнейшее увеличение ее длины не приводит к повышению точности классификации, но требует большего времени на обучение ННС. Следовательно, для имеющихся наборов данных при обучении ННС целесообразно использовать хромосомы длиной не более 10. Указанное значение длины хромосомы было использовано при проведении дальнейших исследований.

В таблице 4.11 представлены зависимости влияния числа хромосом в ГА на результаты обучения ННС для различных наборов данных.

Таблица 4.11 – Влияние числа хромосом в генетическом алгоритме на результаты обучения нечетких нейронных сетей

Набор данных	Число хромосом в генетическом алгоритме				
	50	100	<b>150</b>	200	250
Точность классификации на обучающих данных, %					
Iris Data Set	99,1	99,2	<b>99,2</b>	99,2	99,2
Banknote Authentication Data Set	99,5	99,5	<b>99,6</b>	99,6	99,6
User Knowledge Modeling Data Set	96,8	96,8	<b>96,8</b>	96,8	96,8
Точность классификации на тестовых данных, %					
Iris Data Set	99,3	99,3	<b>99,4</b>	99,4	99,4
Banknote Authentication Data Set	98,8	98,8	<b>98,9</b>	98,9	98,8
User Knowledge Modeling Data Set	95,3	95,3	<b>95,6</b>	95,6	95,6
Точность классификации на валидационных данных, %					
Iris Data Set	98,1	98,1	<b>98,2</b>	98,2	98,2
Banknote Authentication Data Set	99,1	99,2	<b>99,2</b>	99,2	99,2
User Knowledge Modeling Data Set	95,3	95,3	<b>95,4</b>	95,3	95,4
Время обучения					
Iris Data Set	00:06:58	00:07:03	<b>00:07:09</b>	00:07:13	00:07:18
Banknote Authentication Data Set	01:18:24	01:18:53	<b>01:19:22</b>	01:19:47	01:20:08
User Knowledge Modeling Data Set	02:32:18	02:32:48	<b>02:33:25</b>	02:33:53	02:34:29

Изменение числа хромосом в ГА не приводит к существенному изменению времени обучения ННС. При этом незначительно увеличивается точность классификации при увеличении числа хромосом до  $100 \div 150$ . Дальнейшее увеличение их числа не приводит к повышению точности классификации. Эта закономерность проявилась для всех наборов анализируемых данных. Следовательно, 150 хромосом в ГА можно считать оптимальным. При таком количестве хромосом достигается максимальная точность классификации. Указанное значение числа хромосом было использовано при проведении дальнейших исследований.

В таблице 4.12 представлены зависимости влияния вероятности мутации генов дочерних хромосом на результаты обучения ННС.

Таблица 4.12 – Влияние вероятности мутации генов дочерних хромосом на результаты обучения нечетких нейронных сетей

Набор данных	Вероятность мутации генов дочерних хромосом, %				
	1	2	3	4	5
Точность классификации на обучающих данных, %					
Iris Data Set	99,1	99,2	99,2	99,2	<b>99,2</b>
Banknote Authentication Data Set	99,6	99,6	99,5	99,6	<b>99,6</b>
User Knowledge Modeling Data Set	96,7	96,8	96,8	96,8	<b>96,8</b>
Точность классификации на тестовых данных, %					
Iris Data Set	99,4	99,3	99,4	99,4	<b>99,4</b>
Banknote Authentication Data Set	98,8	98,9	98,9	98,9	<b>98,9</b>
User Knowledge Modeling Data Set	95,6	95,6	95,6	95,6	<b>95,6</b>
Точность классификации на валидационных данных, %					
Iris Data Set	98,2	98,2	98,2	98,2	<b>98,2</b>
Banknote Authentication Data Set	99,2	99,2	99,2	99,2	<b>99,2</b>
User Knowledge Modeling Data Set	95,3	95,4	95,4	95,4	<b>95,4</b>
Время обучения					
Iris Data Set	00:07:12	00:06:58	00:07:03	00:07:05	<b>00:07:09</b>
Banknote Authentication Data Set	01:19:20	01:19:25	01:19:28	01:19:25	<b>01:19:22</b>
User Knowledge Modeling Data Set	02:33:21	02:33:26	02:33:25	02:33:22	<b>02:33:25</b>

Изменение вероятности мутации генов дочерних хромосом в ГА практически не повлияло на точность классификации и время обучения ННС. При снижении вероятности мутации до  $1 \div 3$  процентов наблюдалось незначительное однократное уменьшение точности классификации на 0,1 на каждом из анализируемых наборов данных. Однако, это не является закономерностью, а, скорее, проявлением случайного характера обучения ННС на основе ГА. Кроме того, не выявлена



закономерность влияния вероятности мутации на время обучения ННС. Следовательно, при проведении дальнейших исследований целесообразно не изменять вероятность мутации генов дочерних хромосом в ГА и оставить ее, равной 5%.

В таблице 4.13 представлены зависимости влияния числа эпох холостой работы ГА на результаты обучения ННС для различных наборов данных.

Таблица 4.13 – Влияние числа эпох холостой работы ГА на результаты обучения нечетких нейронных сетей

Набор данных	Число эпох холостой работы генетического алгоритма			
	5	10	15	20
Точность классификации на обучающих данных, %				
Iris Data Set	<b>99,2</b>	99,1	99,2	99,2
Banknote Authentication Data Set	<b>99,6</b>	99,6	99,6	99,6
User Knowledge Modeling Data Set	<b>96,8</b>	96,8	96,8	96,7
Точность классификации на тестовых данных, %				
Iris Data Set	<b>99,4</b>	99,4	99,4	99,4
Banknote Authentication Data Set	<b>98,9</b>	98,9	98,9	98,9
User Knowledge Modeling Data Set	<b>95,6</b>	95,6	95,5	95,6
Точность классификации на валидационных данных, %				
Iris Data Set	<b>98,2</b>	98,2	98,1	98,2
Banknote Authentication Data Set	<b>99,2</b>	99,2	99,2	99,2
User Knowledge Modeling Data Set	<b>95,4</b>	95,3	95,4	95,4
Время обучения				
Iris Data Set	<b>00:07:09</b>	00:13:28	00:19:46	00:26:12
Banknote Authentication Data Set	<b>01:19:22</b>	02:31:17	03:05:31	04:20:08
User Knowledge Modeling Data Set	<b>02:33:25</b>	04:55:12	07:12:18	09:28:43

Видно, что число эпох холостой работы генетического алгоритма практически не оказывает влияния на достигаемую точность классификации данных из всех используемых наборов. При этом наблюдается линейное увеличение времени, затрачиваемого на обучение ННС. Так, например, для набора данных «Iris Data Set» при задании 5 эпох холостой работы ГА потребовалось чуть более 7 минут на обучение ННС, а при задании 20 эпох – более 26 минут. Для набора данных «Banknote Authentication Data Set» время обучения увеличилось, соответственно, примерно с 1 часа 19 минут до 4 часов 20 минут, а для набора данных «User Knowledge Modeling Data Set» – с 2 часов 33 минут до 9,5 часов. Следовательно, при проведении последующих исследований целесообразно оставить исходное число эпох холостой работы ГА.

В таблице 4.14 представлены зависимости влияния числа циклов холостого обучения ННС на результаты обучения для различных наборов данных.

Таблица 4.14 – Влияние числа циклов холостого обучения нечетких нейронных сетей на результаты обучения

Набор данных	Число циклов холостого обучения нечеткой нейронной сети			
	2	4	6	8
Точность классификации на обучающих данных, %				
Iris Data Set	<b>99,2</b>	99,2	99,2	99,2
Banknote Authentication Data Set	<b>99,6</b>	99,6	99,6	99,6
User Knowledge Modeling Data Set	<b>96,8</b>	96,8	96,8	96,8
Точность классификации на тестовых данных, %				
Iris Data Set	<b>99,4</b>	99,4	99,4	99,4
Banknote Authentication Data Set	<b>98,9</b>	98,9	98,9	98,9
User Knowledge Modeling Data Set	<b>95,6</b>	95,6	95,6	95,6
Точность классификации на валидационных данных, %				
Iris Data Set	<b>98,2</b>	98,2	98,2	98,2
Banknote Authentication Data Set	<b>99,2</b>	99,2	99,2	99,2
User Knowledge Modeling Data Set	<b>95,4</b>	95,4	95,4	95,4
Время обучения				
Iris Data Set	<b>00:07:09</b>	00:14:51	00:21:40	00:28:22
Banknote Authentication Data Set	<b>01:19:22</b>	02:58:04	04:17:32	06:03:47
User Knowledge Modeling Data Set	<b>02:33:25</b>	05:06:28	07:45:09	10:25:13

Видно, что число циклов холостого обучения ННС не влияет на точность классификации данных. При этом наблюдается линейное увеличение времени на обучение ННС. Так, для набора данных «Iris Data Set» при задании 2 циклов потребовалось чуть более 7 минут на обучение ННС, а при задании 8 циклов – более 28 минут. Для набора данных «Banknote Authentication Data Set» время обучения увеличилось с 1 часа 19 минут до 6 часов, а для набора данных «User Knowledge Modeling Data Set» – с 2,5 до 10,5 часов. Следовательно, целесообразно использовать исходное число циклов холостого обучения ННС.

Таким образом, в результате проведенных исследований удалось достигнуть высокой (не ниже 95%) точности классификации по всем наборам данных. При этом время обучения ННС составило от 7 минут до 2,5 часов в зависимости от набора данных, что является приемлемым для систем нейронечеткого моделирования.

#### **4.2.5 Рекомендации по выбору значений параметров инициализации и обучения нечетких нейронных сетей**

Результаты исследований позволили сформулировать следующие рекомендации по выбору значений параметров инициализации и обучения ННС:

- число градаций входных переменных: 5÷6;
- форма функции принадлежности: трапецеидальная;
- длина хромосомы в ГА: 10;
- число хромосом в ГА: 150;
- вероятность мутации генов дочерних хромосом: 5%;
- число эпох холостой работы ГА: 5;
- число циклов холостого обучения ННС: 2.

Следует отметить, что строго формально указанные рекомендации справедливы только для тех наборов данных, на которых обучались ННС. Однако, в силу общности найденных закономерностей можно утверждать, что результаты будут применимы и для других наборов данных. При этом всегда для каждого нового набора необходимо проводить дополнительные исследования для нахождения лучших значений параметров инициализации и обучения ННС.

В соответствии с указанными рекомендациями сформированы базы знаний для каждого из наборов данных. Рассмотрим эти базы знаний более подробно.

#### **4.2.6 Описание сформированных баз знаний при анализе наборов данных**

В результате обучения ННС на наборах данных «Iris Data Set», «Banknote Authentication Data Set» и «User Knowledge Modeling Data Set» сформированы соответствующие базы знаний. При обучении ННС использовались рекомендуемые значения параметров, описанные в предыдущем разделе.

В таблицах 4.15÷4.17 представлены фрагменты сформированных баз знаний.

Таблица 4.15 – Фрагмент базы знаний для набора данных «Iris Data Set»

№ п/п		X1		X2		X3		X4		Y	Вес правила
1	Если	1	И	2	И	1	И	1	То	1	0,08
2	Если	1	И	3	И	1	И	1	То	1	0,261
3	Если	1	И	3	И	1	И	2	То	1	0,096
4	Если	1	И	3	И	2	И	1	То	1	0,187
5	Если	1	И	3	И	2	И	2	То	1	0,092
6	Если	1	И	4	И	1	И	1	То	1	0,089
7	Если	1	И	4	И	1	И	2	То	1	0,063
8	Если	1	И	4	И	2	И	1	То	1	0,076
9	Если	2	И	1	И	3	И	3	То	2	0,067
10	Если	2	И	2	И	3	И	3	То	2	0,209
11	Если	2	И	2	И	4	И	4	То	3	0,064
12	Если	2	И	3	И	1	И	1	То	1	0,316
13	Если	2	И	3	И	1	И	2	То	1	0,141
14	Если	2	И	3	И	2	И	1	То	1	0,222
15	Если	2	И	3	И	2	И	2	То	1	0,132
16	Если	2	И	3	И	3	И	3	То	2	0,112
17	Если	2	И	3	И	4	И	3	То	2	0,061
18	Если	2	И	4	И	1	И	1	То	1	0,229
19	Если	2	И	4	И	1	И	2	То	1	0,164
20	Если	2	И	4	И	2	И	1	То	1	0,169
21	Если	2	И	4	И	2	И	2	То	1	0,144
22	Если	3	И	2	И	3	И	3	То	2	0,267
23	Если	3	И	2	И	3	И	4	То	2	0,095
24	Если	3	И	2	И	4	И	3	То	3	0,071
25	Если	3	И	2	И	4	И	3	То	2	0,182
26	Если	3	И	2	И	4	И	4	То	3	0,236
27	Если	3	И	2	И	4	И	4	То	2	0,095
28	Если	3	И	2	И	4	И	5	То	3	0,086
29	Если	3	И	2	И	5	И	4	То	3	0,065
30	Если	3	И	3	И	3	И	3	То	2	0,257
31	Если	3	И	3	И	3	И	4	То	2	0,125
32	Если	3	И	3	И	4	И	3	То	2	0,188
33	Если	3	И	3	И	4	И	4	То	3	0,193
34	Если	3	И	3	И	4	И	4	То	2	0,159
35	Если	3	И	3	И	4	И	5	То	3	0,093
36	Если	4	И	2	И	4	И	3	То	2	0,07
37	Если	4	И	2	И	4	И	4	То	3	0,159
38	Если	4	И	2	И	5	И	4	То	3	0,116
39	Если	4	И	3	И	3	И	3	То	2	0,061
40	Если	4	И	3	И	4	И	3	То	2	0,075
41	Если	4	И	3	И	4	И	4	То	3	0,206
42	Если	4	И	3	И	4	И	5	То	3	0,145
43	Если	4	И	3	И	5	И	4	То	3	0,134
44	Если	4	И	3	И	5	И	5	То	3	0,084

Таблица 4.16 – Фрагмент базы знаний для набора данных «Banknote Authentication Data Set»

№ п/п		X1		X2		X3		X4		Y	Вес правила
1	Если	2	И	2	И	4	И	4	То	1	0,04
2	Если	2	И	2	И	4	И	5	То	1	0,047
3	Если	2	И	2	И	5	И	4	То	1	0,039
4	Если	2	И	2	И	5	И	5	То	1	0,035
5	Если	2	И	3	И	3	И	5	То	1	0,039
6	Если	2	И	4	И	2	И	4	То	1	0,051
7	Если	2	И	4	И	2	И	5	То	1	0,051
8	Если	2	И	4	И	3	И	5	То	1	0,042
9	Если	3	И	2	И	3	И	5	То	1	0,051
10	Если	3	И	2	И	4	И	4	То	1	0,038
11	Если	3	И	2	И	4	И	5	То	1	0,082
12	Если	3	И	3	И	2	И	5	То	1	0,11
13	Если	3	И	3	И	3	И	5	То	1	0,12
14	Если	3	И	3	И	3	И	6	То	1	0,04
15	Если	3	И	3	И	4	И	5	То	1	0,042
16	Если	3	И	4	И	1	И	4	То	1	0,056
17	Если	3	И	4	И	2	И	3	То	1	0,043
18	Если	3	И	4	И	2	И	4	То	1	0,144
19	Если	3	И	4	И	2	И	5	То	1	0,135
20	Если	3	И	4	И	3	И	5	То	1	0,066
21	Если	4	И	3	И	2	И	5	То	1	0,052
22	Если	4	И	3	И	3	И	4	То	0	0,036
23	Если	4	И	3	И	3	И	5	То	0	0,063
24	Если	4	И	4	И	1	И	4	То	1	0,055
25	Если	4	И	4	И	2	И	4	То	1	0,05
26	Если	4	И	4	И	2	И	5	То	1	0,085
27	Если	4	И	4	И	2	И	5	То	0	0,068
28	Если	4	И	4	И	3	И	5	То	0	0,089
29	Если	4	И	5	И	2	И	3	То	0	0,04
30	Если	4	И	5	И	2	И	4	То	0	0,104
31	Если	4	И	5	И	2	И	5	То	0	0,076
32	Если	4	И	5	И	3	И	5	То	0	0,036
33	Если	5	И	3	И	2	И	5	То	0	0,046
34	Если	5	И	3	И	3	И	5	То	0	0,115
35	Если	5	И	3	И	3	И	6	То	0	0,046
36	Если	5	И	4	И	2	И	5	То	0	0,107
37	Если	5	И	4	И	2	И	6	То	0	0,051
38	Если	5	И	4	И	3	И	5	То	0	0,066
39	Если	5	И	4	И	3	И	6	То	0	0,044
40	Если	5	И	5	И	1	И	3	То	0	0,046
41	Если	5	И	5	И	1	И	4	То	0	0,05
42	Если	5	И	5	И	2	И	4	То	0	0,081
43	Если	5	И	5	И	2	И	5	То	0	0,082
44	Если	5	И	6	И	1	И	3	То	0	0,042

Таблица 4.17 – Фрагмент базы знаний для набора данных «User Knowledge Modeling Data Set»

№ п/п		X1		X2		X3		X4		X5		Y	Вес правила
1	Если	1	И	1	И	2	И	2	И	1	То	1	0,049
2	Если	1	И	1	И	2	И	3	И	1	То	1	0,045
3	Если	1	И	3	И	4	И	2	И	4	То	2	0,045
4	Если	2	И	1	И	2	И	2	И	1	То	1	0,049
5	Если	2	И	1	И	2	И	3	И	1	То	1	0,079
6	Если	2	И	1	И	2	И	3	И	2	То	1	0,044
7	Если	2	И	1	И	3	И	3	И	1	То	1	0,096
8	Если	2	И	1	И	3	И	3	И	2	То	1	0,063
9	Если	2	И	2	И	1	И	3	И	1	То	1	0,06
10	Если	2	И	2	И	1	И	3	И	2	То	1	0,065
11	Если	2	И	2	И	2	И	3	И	1	То	1	0,103
12	Если	2	И	2	И	2	И	3	И	2	То	1	0,104
13	Если	2	И	2	И	2	И	3	И	2	То	3	0,079
14	Если	2	И	2	И	2	И	4	И	1	То	1	0,049
15	Если	2	И	2	И	3	И	2	И	3	То	4	0,052
16	Если	2	И	2	И	3	И	2	И	4	То	4	0,055
17	Если	2	И	2	И	3	И	3	И	1	То	1	0,065
18	Если	2	И	2	И	3	И	3	И	2	То	1	0,066
19	Если	2	И	2	И	3	И	4	И	1	То	1	0,045
20	Если	2	И	2	И	3	И	4	И	2	То	1	0,045
21	Если	2	И	2	И	4	И	2	И	4	То	2	0,044
22	Если	2	И	2	И	4	И	2	И	5	То	2	0,044
23	Если	2	И	3	И	1	И	3	И	1	То	1	0,057
24	Если	2	И	3	И	1	И	3	И	2	То	1	0,057
25	Если	2	И	3	И	2	И	3	И	1	То	1	0,08
26	Если	2	И	3	И	2	И	3	И	2	То	1	0,078
27	Если	2	И	3	И	2	И	3	И	2	То	3	0,049
28	Если	2	И	3	И	3	И	2	И	4	То	2	0,049
29	Если	2	И	3	И	3	И	2	И	5	То	2	0,056
30	Если	2	И	3	И	3	И	3	И	2	То	3	0,062
31	Если	2	И	3	И	3	И	3	И	4	То	2	0,046
32	Если	2	И	3	И	4	И	2	И	4	То	2	0,059
33	Если	2	И	3	И	4	И	2	И	5	То	2	0,052
34	Если	2	И	3	И	4	И	3	И	2	То	3	0,046
35	Если	2	И	4	И	3	И	2	И	5	То	2	0,045
36	Если	3	И	2	И	2	И	3	И	1	То	1	0,103
37	Если	3	И	2	И	2	И	3	И	2	То	1	0,094
38	Если	3	И	2	И	3	И	2	И	3	То	4	0,048
39	Если	3	И	2	И	3	И	2	И	4	То	4	0,048
40	Если	3	И	2	И	4	И	2	И	3	То	4	0,056
41	Если	3	И	2	И	4	И	2	И	4	То	4	0,048
42	Если	3	И	3	И	3	И	3	И	2	То	3	0,054
43	Если	3	И	3	И	4	И	2	И	3	То	4	0,054
44	Если	3	И	3	И	4	И	3	И	2	То	3	0,047

Входными переменными в правилах базы знаний для набора данных «Iris Data Set» являются следующие [82]:

- 1) X1 – «длина чашелистика»;
- 2) X2 – «ширина чашелистика»;
- 3) X3 – «длина лепестка»;
- 4) X4 – «ширина лепестка».

Каждая входная переменная может принимать одно из пяти нечетких значений: 1 – очень малое, 2 – малое, 3 – среднее, 4 – большое, 5 – очень большое.

Значениями выходной переменной Y являются следующие классы решений:

- 1 (класс «Iris-setosa»);
- 2 (класс «Iris-versicolor»);
- 3 (класс «Iris-virginica»).

Сформированная из набора данных «Iris Data Set» база знаний состоит из 182 нечетко-продукционных правил. Каждое нечеткое правило включает в себя четыре входные переменные с пятью градациями и одно заключение с тремя возможными значениями (классами решений). Кроме того, все правила базы знаний имеют соответствующий вес, используемый для их ранжирования в процессе принятия решений.

Входными переменными в правилах базы знаний для набора данных «Bank-note Authentication Data Set» являются следующие:

- 1) X1 – дисперсия вейвлет-преобразованного изображения;
- 2) X2 – асимметрия вейвлет-преобразованного изображения;
- 3) X3 – эксцесс вейвлет-преобразованного изображения;
- 4) X4 – энтропия изображения.

Переменные могут принимать одно из шести нечетких значений: 1 – очень малое, 2 – малое, 3 – среднее, 4 – выше среднего, 5 – большое, 6 – очень большое.

Значениями выходной переменной Y являются следующие классы решений:

- 0 (подлинная банкнота);
- 1 (поддельная банкнота).

Сформированная из набора данных «Banknote Authentication Data Set» база знаний состоит из 454 правил. Каждое нечеткое правило включает в себя четыре входные переменные с шестью градациями и одно заключение с двумя возможными значениями (классами решений).

Входными переменными в правилах базы знаний для набора данных «User Knowledge Modeling Data Set» являются следующие:

- 1) X1 – доля времени на изучение материала;
- 2) X2 – доля числа повторений основного учебного материала;
- 3) X3 – степень изучения дополнительного учебного материала;
- 4) X4 – результат проверки знаний по дополнительному материалу;
- 5) X5 – результат проверки знаний по основному учебному материалу.

Каждая входная переменная может принимать одно из следующих нечетких значений: 1 – очень малое, 2 – малое, 3 – среднее, 4 – большое, 5 – очень большое.

Значениями выходной переменной Y являются следующие классы решений:

- 1 – очень низкий уровень знаний (Very Low);
- 2 – низкий уровень знаний (Low);
- 3 – средний уровень знаний (Middle);
- 4 – высокий уровень знаний (High).

Сформированная из набора данных «User Knowledge Modeling Data Set» база знаний состоит из 760 нечетко-продукционных правил. Каждое правило включает в себя 5 входных переменных с пятью градациями и одно заключение с четырьмя возможными значениями (классами решений).

Характеристики БЗ представлены в таблице 4.18.

Таблица 4.18 – Характеристики баз знаний

Набор данных	Характеристики	Число правил в базе знаний	Обобщающая способность базы знаний и алгоритма вывода
Iris Data Set		182	98,2%
Banknote Authentication Data Set		454	99,2%
User Knowledge Modeling Data Set		760	95,4%



Мерой эффективности баз знаний (совместно с алгоритмом логического вывода) является обобщающая способность [57]. Наиболее высокая обобщающая способность получена на наборе данных «Banknote Authentication Data Set» и составляет 99,2%. Обобщающая способность на основе базы знаний, сформированной при обучении ННС на наборе данных «Iris Data Set», составляет 98,2%. Наименьшей обобщающей способностью в 95,4% обладает модель, соответствующая набору данных «User Knowledge Modeling Data Set». При этом сформированные базы знаний являются адекватными (их обобщающая способность совместно с алгоритмом вывода составляет не менее 95%).

Для окончательной оценки адекватности баз знаний необходимо сравнить точности классификации анализируемых данных различными методами.

#### 4.2.7 Сравнение точности классификации на основе сформированных баз знаний с точностью других методов классификации

С помощью программы Deductor [103; 152] произведено построение различных моделей классификации (см. рис.4.19).

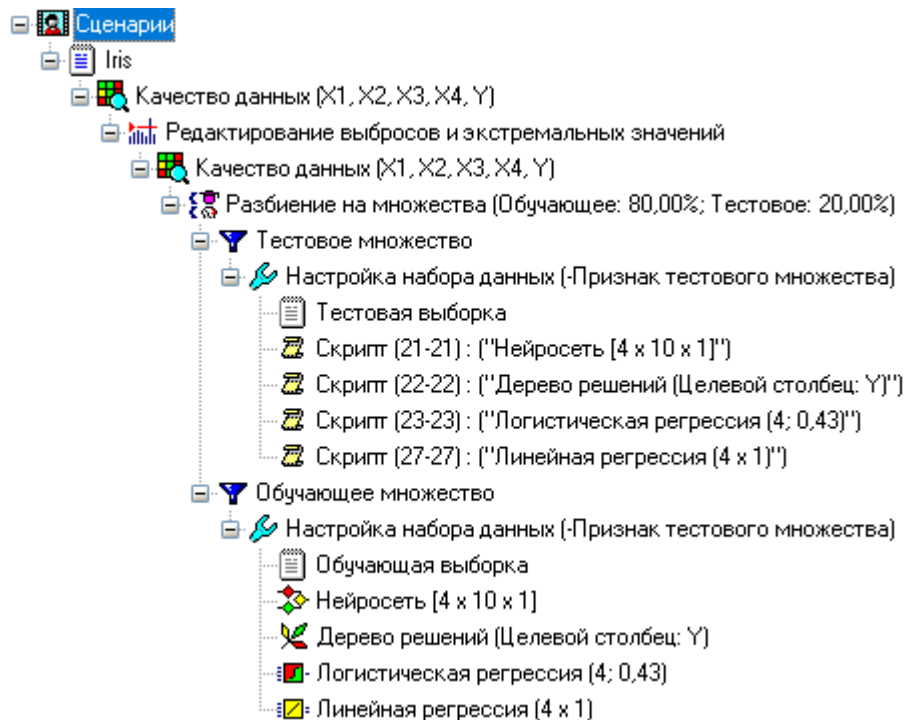


Рисунок 4.19 – Проект Deductor для построения моделей классификации

В представленном примере показано построение четырех моделей (нейронная сеть, дерево решений, логистическая регрессия, линейная регрессия) для набора данных «Iris Data Set». Аналогично построены модели для наборов данных «Banknote Authentication Data Set» и «User Knowledge Modeling Data Set».

В таблице 4.19 приведены результаты моделирования в Deductor, позволяющие сравнить точность классификации на основе указанных методов с точностью, достигнутой при использовании БЗ.

Таблица 4.19 – Точность классификации наборов данных различными методами

Метод классификации	Набор данных		
	Iris Data Set	Banknote Authentication Data Set	User Knowledge Modeling Data Set
Нейронная сеть	98,2%	98,9%	94,2%
Дерево решений	96,4%	98,5%	92,3%
Логистическая регрессия	–	98,2%	–
Линейная регрессия	94,5%	97,1%	86,5%
Нечеткая база знаний	98,2%	99,2%	95,4%

Для набора данных «Iris Data Set» точность классификации на основе БЗ совпадает с точностью нейронной сети, на 1,8% превышает точность дерева решений и на 3,7% – точность на основе линейной регрессии. Для набора данных «Banknote Authentication Data Set» точность БЗ на 0,3% превышает точность нейросетевой классификации, на 0,7% – дерева решений, на 1% – логистической регрессии и на 2,1% – линейной регрессии. Для набора данных «User Knowledge Modeling Data Set» точность БЗ на 1,2% превышает точность нейросетевой классификации, на 3,1% – дерева решений и на 8,9% – линейной регрессии. Следовательно, в среднем по всем наборам данных точность классификации на основе сформированных БЗ на 2,42% превышает точность других рассмотренных методов. Кроме того, применение нечетко-продукционных правил совместно с алгоритмом вывода улучшает интерпретируемость принимаемых решений.

Таким образом, формируемые на основе обучения ННС базы знаний могут быть эффективно использованы для оценки состояния объектов с точностью, не уступающей точности рассмотренных методов классификации [86; 87].

### **4.3 Апробация программного комплекса при формировании баз знаний для оценки состояния объектов в различных предметных областях**

#### **4.3.1 Формирование базы знаний для подбора геолого-технических мероприятий на нефтяном месторождении**

##### *Описание исходных данных*

Для обучения ННС и формирования базы знаний для подбора геолого-технических мероприятий (ввода скважин в эксплуатацию) использовались исходные данные, описываемые следующими входными переменными [91]:

1) общие сведения о каждой скважине:

- порядковый номер скважины;
- промысловый номер добывающей скважины;
- год начала отбора;
- длительность отбора (лет);
- была под закачкой или переведена под закачку;
- использованы средние значения параметров;
- тип зоны (нефтяная или водонефтяная);
- группа породы;
- тип скважины (не определен, вертикальная, горизонтальная);
- байт начала не нулевых дебитов нефти.

2) геолого-геофизические данные нефтяного пласта:

- проницаемость абсолютная;
- пористость;
- нефтенасыщенность начальная;
- нефтенасыщенная толщина начальная (м).

3) параметры на год начала отбора:

- площадь зоны дренирования;
- начальные балансовые запасы нефти;
- текущие балансовые запасы нефти;

- начальные подвижные запасы нефти;
- текущие подвижные запасы нефти;
- коэффициент охвата заводнением;
- расстояние до ближайшей нагнетательной скважины;
- минимальный дебит соседних скважин;
- средний дебит соседних скважин;
- максимальный дебит соседних скважин;
- минимальная обводненность соседних скважин;
- средняя обводненность соседних скважин;
- максимальная обводненность соседних скважин.

Исходные данные были представлены по 46 объектам нефтяного месторождения с карбонатными коллекторами. Объем данных составил 3398 записей [91].

В таблице 4.20 представлен фрагмент исходных данных для анализа.

Таблица 4.20 – Фрагмент исходных данных по скважинам

pronc	poris	nasich	tolsh	plosh	bal1	bal2	kohvat	rnagn	debsos	obvod	result
56,7	0,131	0,685	5,8	12	22	20,7	0,1375	7586	0,2	0,5	0
12	0,119	0,656	4,4	12	14,6	11,8	0,3531	17111	0,1	0,597	0
56,7	0,131	0,685	1,6	0,1	0	0	0,3543	0	0	0	0
56,7	0,131	0,685	6,4	12	24,3	23,8	0,0418	12390	0,4	0,794	0
56,7	0,131	0,685	8,4	4,4	5,8	5,8	0,005	5798	1,4	0,095	1
87	0,162	0,695	2,8	12	13,3	11,6	0,2483	12366	0	0	0
56,7	0,131	0,685	4,2	12	15,9	15	0,1424	11272	0	0	0
54,8	0,133	0,683	7,1	12	21	19,8	0,13	7855	0,5	0,955	0
30	0,148	0,49	1,4	12	4,3	4	0,2084	1005	0	0	0
10,4	0,108	0,644	5,2	12	11,3	10,5	0,3722	13787	0,3	0,773	0
50,7	0,138	0,714	9,9	12	34,4	31,1	0,0925	4270	0	0	0
56,7	0,131	0,685	4,6	7,3	5,3	4,9	0,1638	3190	0,9	0,837	0
45,3	0,12	0,681	14,9	12	32,3	31,4	0,0719	34181	1	0,265	0
2	0,097	0,796	16,1	12,4	34,8	29,4	0,4396	25489	0	0,065	0
3	0,116	0,749	16,9	12	60,4	59,9	0,0136	4034	1,1	0,709	0
13,2	0,121	0,687	3,7	12	14,8	14,3	0,0616	0	0	0	0
1	0,096	0,565	2,8	11,3	8,1	7,6	0,1563	0	0,3	0,755	0
6	0,127	0,66	19,7	0,4	1,4	1,4	0,0284	0	9,5	0,102	0
5,8	0,111	0,792	6,5	12	24,5	24,3	0,0097	0	1,4	0,215	1
4	0,123	0,6	5,8	12	24,3	23,7	0,0469	0	0	0	0
13,9	0,13	0,77	4,7	12	26,7	25,8	0,0529	0	0,9	0,794	1
13,9	0,094	0,74	5,2	13,1	28,5	26,1	0,14	0	0,2	0,778	0
1,3	0,106	0,623	5,8	12	16	14,9	0,243	0	0,2	0,858	0

В данной таблице в качестве входных переменных указаны следующие [91]:

- pronic – абсолютная проницаемость породы;
- poris – пористость породы;
- nasich – начальная нефтенасыщенность;
- tolsh – начальная нефтенасыщенная толщина;
- plosh – площадь зоны дренирования;
- bal1 – начальные балансовые запасы нефти;
- bal2 – текущие балансовые запасы нефти;
- kohvat – коэффициент охвата заводнением;
- magn – расстояние до ближайшей нагнетательной скважины;
- debsos – средний дебит соседних скважин;
- obvod – минимальная обводненность соседних скважин.

Единственная выходная переменная «result» характеризует описываемую скважину как эффективную в применении ( $result = 1$  при дебите скважины  $> 5$ ) или неэффективную ( $result = 0$  при дебите скважины  $\leq 5$ ).

### *Подготовка исходных данных к анализу*

Перед формированием базы знаний необходимо произвести подготовку исходных данных к анализу. Никакой метод машинного обучения не даст положительного результата, если данные будут плохого качества. Поэтому важным является оценка качества исходных данных и их подготовка к анализу [73; 91].

В наборе исходных данных могут встречаться значения, которые не укладываются в общую модель поведения объекта. Такими значениями могут быть выбросы и аномалии. При подготовке исходных данных к анализу необходимо выполнять поиск и корректировку таких значений, поскольку они являются одним из факторов, снижающих качество данных и достоверность результатов анализа [91].

В работе использован следующий алгоритм поиска и устранения выбросов и аномалий в данных для каждой количественной входной переменной:

1) определить среднее значение входной переменной:

$$x_{cp} = \frac{1}{n} \sum_{i=1}^n x_i,$$

где  $n$  – объем выборки (число значений входной переменной),  $x_i$  –  $i$ -е значение входной переменной;

2) определить среднеквадратическое отклонение от среднего значения:

$$S_{x_{cp}} = \sqrt{\frac{\sum_{i=1}^n (x_i - x_{cp})^2}{n(n-1)}};$$

3) найти выбросы и аномальные значения в данных на основе следующих критериев: порог отклонения  $S_{x_{cp}}$  от среднего 3 (для выбросов) и 5 (для аномалий);

4) обработать в исходных данных строки с выбросами и аномалиями.

Алгоритм поиска выбросов и аномалий позволил оценить качество исходных данных и выработать рекомендации по его улучшению. Всего в исходных данных объемом 3398 строк было найдено 917 строк с выбросами и 211 строк с аномалиями. Переменная «tip» оказалась непригодной для анализа. Единственная переменная «obvod» оказалась полностью пригодной к анализу. К остальным значениям переменных были применены методы повышения качества исходных данных [184]: удаления записей с аномальными значениями и заменой выбросов на медианное значение. После предобработки объем исходных данных сократился на 211 строк (с 3398 до 3187) в соответствии с количеством удаленных записей с аномальными значениями [91].

Следующим этапом подготовки данных к анализу явился анализ корреляций между входными и выходной переменными. При построении интеллектуальной модели целесообразно оценить зависимости в исходных данных для анализа, а именно выбрать систему входных признаков, значимо влияющих на выходной результат. Данная задача особенно актуальна при обучении ННС, способных эффективно обучаться в пространстве небольшой (до 10÷12 входных переменных) размерности. В результате корреляционного анализа данных произведен выбор входных переменных для обучения ННС [91].

В таблице 4.21 представлены значения коэффициентов линейной корреляции Пирсона, вычисленные для всех входных переменных по отношению к выходной.

Таблица 4.21 – Результаты корреляционного анализа данных

Входные переменные	Коэффициент корреляции с выходной переменной
gru	-0,03
pronis	-0,153
poris	-0,136
nasich	0,015
tolsh	0,02
plosh	0,028
bal1	0,01
bal2	0,007
podv1	0,017
podv2	0,017
kohvat	0,012
rnagn	0,013
debsos	0,19
obvod	-0,189

Из таблицы видно, что не все входные переменные имеют корреляционную зависимость с выходной. Так, например, такие переменные как «проницаемость», «пористость», «средний дебит соседних скважин» и «минимальная обводненность соседних скважин» имеют относительно высокую корреляцию с выходной переменной «результат». Это означает, что указанные переменные целесообразно использовать для обучения ННС. При этом порог корреляции, определяющий количество и состав входных переменных, подбирался экспериментально. После подготовки исходных данных и их анализа произведено формирование баз знаний с различным набором комбинаций входных переменных [91].

***Варианты комбинаций входных переменных в исходных данных  
при обучении нечеткой нейронной сети***

При обучении ННС и формировании базы знаний для подбора геолого-технических мероприятий (ввода скважин в эксплуатацию) на нефтяном месторождении для нахождения лучшего решения задачи использовались различные варианты комбинаций входных переменных в исходных данных. Проведены следующие исследования [91]:

- оценка влияния порога корреляции в исходных данных на число и состав входных переменных модели, на результаты обучения ННС и на время построения модели;

- оценка влияния порога эффективности скважины на результаты обучения ННС;

- оценка влияния вариантов расчета выходной переменной (первый ненулевой дебит скважины за последние 5 лет, средний дебит скважины за первые 2 года эксплуатации, средний дебит скважины за первые 3 года эксплуатации, максимальный дебит скважины за последние 5 лет) на результаты обучения ННС.

Рассмотрим результаты проведенных исследований.

***Влияние порога корреляции в исходных данных на  
результаты обучения нечеткой нейронной сети***

Анализ корреляционных зависимостей в исходных данных показал, что не все входные переменные в исходных данных одинаково значимо влияют на выходной результат. В частности, было показано, что такие входные переменные как «проницаемость», «пористость», «средний дебит соседних скважин» и «минимальная обводненность соседних скважин» имеют относительно высокую корреляцию с выходной переменной «результат». При этом очевидно, что выбор конкретного порога корреляции влияет на число и состав входных переменных модели, что в свою очередь влияет на результат обучения ННС и на время построения нейронечеткой модели.



В исследовании влияния порога корреляции в исходных данных на число и состав входных переменных модели, на результаты обучения ННС и на время построения нейронечеткой модели использовались следующие пороги значимости корреляции входных переменных с выходной: 0,05; 0,1 и 0,15. В соответствии с выбранными порогами корреляции произведен отбор входных переменных и обучение ННС.

В таблице 4.22 представлены результаты экспериментов.

Таблица 4.22 – Влияние порога корреляции в исходных данных на результаты нейронечеткого моделирования

Коэффициент корреляции	Число входных переменных	Состав входных переменных	Точность классификации на обучающей выборке, %	Точность классификации на тестовой выборке, %	Точность классификации на валидационной выборке, %	Время построения модели, мин
0,05	4	pronіc, poris, debsos, obvod	90,53	90,83	90,83	893
0,1	4	pronіc, poris, debsos, obvod	90,53	90,83	90,83	893
0,15	3	pronіc, debsos, obvod	90	90,34	90,25	140

Из таблицы видно, что при уменьшении порога корреляции и, соответственно, при увеличении числа входных переменных точность классификации данных (при обучении, тестировании и валидации модели) незначительно возрастает. Однако при этом возрастает время построения модели. Этот фактор необходимо учитывать при обучении ННС.

Следует отметить, что на этом этапе исследований в исходных данных для построения модели в качестве порога эффективности скважины использовалось значение, равное «5». То есть выходная переменная «result» характеризовала соответствующую скважину как эффективную в применении при ее дебите  $> 5$  и как неэффективную – при дебите  $\leq 5$ .

### ***Влияние порога эффективности скважины на результаты обучения нечеткой нейронной сети***

Рассмотрим, как влияет величина порога эффективности скважины на результаты обучения ННС. По экспертным оценкам скважину можно считать эф-

фективной, если ее дебит больше 5 тонн в сутки [144]. Этот критерий был взят за основу при подготовке исходных данных к анализу. Значение выходной переменной принималось равным «1» (скважина эффективная), если ее дебит удовлетворяет указанному условию. Соответственно, значение выходной переменной принималось равным «0» (скважина неэффективная), если ее дебит указанному условию не удовлетворяет. Поэтому все базовые эксперименты проводились, исходя из описанной методики расчета эффективности скважины.

Кроме того, проведены дополнительные эксперименты по оценке влияния порога эффективности скважины на результаты моделирования. При этом порог дебита менялся от 4 до 6 с шагом 1. В таблице 4.23 представлены полученные результаты экспериментальных исследований.

Таблица 4.23 – Влияние порога эффективности скважины на результаты нейронечеткого моделирования

Порог дебита	Точность классификации на обучающих данных, %	Точность классификации на тестовых данных, %	Точность классификации на валидационных данных, %
4	88,33	89,23	89,23
5	90,53	90,83	90,83
6	90,35	90,79	90,34

При уменьшении (до 4-х) или увеличении (до 6-ти) порога эффективности скважины точность классификации данных незначительно снижается. Следовательно, экспертная оценка эффективности скважины на уровне 5 тонн в сутки является адекватной и согласуется с результатами экспериментов [91]. При дальнейших исследованиях все выборки целесообразно формировать, исходя из данного порога. Следует отметить, что на этом этапе исследований для получения максимально точных результатов нейронечеткого моделирования использовалось 4 входных переменных: *pronis*, *poris*, *debsos* и *obvod*.

***Влияние вариантов расчета значений выходной переменной  
на результаты обучения нечеткой нейронной сети***

Необходимость изменения подхода к расчету дебита скважины (в отличие от экспертного подхода, по которому в качестве дебита берется первый ненулевой

дебит скважины за последние 5 лет) возникла вследствие неудовлетворительных результатов обучения ННС (точность классификации данных на уровне 90%). В этом разделе приводятся результаты экспериментов по оценке влияния вариантов расчета выходной переменной (первый ненулевой дебит скважины за последние 5 лет, средний дебит скважины за первые 2 года эксплуатации, средний дебит скважины за первые 3 года эксплуатации, максимальный дебит скважины за последние 5 лет) на результаты обучения ННС [91].

В таблице 4.24 представлены результаты проведенных экспериментов.

Таблица 4.24 – Влияние вариантов расчета значений выходной переменной на результаты нейронечеткого моделирования

Критерий расчета	Точность классификации на обучающих данных, %	Точность классификации на тестовых данных, %	Точность классификации на валидационных данных, %
первый ненулевой дебит скважины за последние 5 лет	90,53	90,83	90,83
средний дебит скважины за первые 2 года эксплуатации	93,32	93,92	93,85
средний дебит скважины за первые 3 года эксплуатации	97,54	95,7	95,68
максимальный дебит скважины за последние 5 лет	91,4	90	90,83

Из таблицы видно, что лучшие результаты классификации достигаются при расчете среднего дебита нефти по скважине за первые 3 года эксплуатации. Соответственно, этот критерий наиболее адекватно (с точки зрения достигаемых результатов классификации) характеризует реальный дебит нефти по скважине, и, как следствие, дальнейшую оценку ее эффективности [91].

### *Анализ результатов экспериментов*

Анализ проведенных экспериментов позволил выявить ряд особенностей и закономерностей в исходных данных, а также их влияния на результаты обучения ННС [91]. Во-первых, при уменьшении порога корреляции и, соответственно, при увеличении числа входных переменных в модели точность классификации данных незначительно возрастает. Однако при этом возрастает время построения модели. Следовательно, не имеет смысла включать в модель большое число входных

переменных. Для эффективной классификации данных достаточно использования четырех переменных.

Во-вторых, при уменьшении или увеличении порога эффективности скважины относительно начального значения точность классификации данных незначительно снижается. Следовательно, экспертная оценка эффективности скважины на уровне 5 тонн в сутки является адекватной и согласуется с результатами экспериментов. При дальнейших исследованиях все выборки целесообразно формировать, исходя из данного порога.

В-третьих, при подготовке данных к анализу необходимо исходить из того, что лучшие результаты классификации достигаются при расчете среднего дебита нефти по скважине за первые 3 года эксплуатации. Соответственно, этот критерий наиболее адекватно (с точки зрения достигаемых результатов классификации) характеризует дебит нефти по скважине, и, как следствие, оценку ее эффективности.

Таким образом, при формировании искомой базы знаний для подбора ГТМ (ввода скважин в эксплуатацию) на нефтяном месторождении реализованы следующие решения, отражающие оптимальный набор значений параметров [91]:

- 1) число входных переменных: 4;
- 2) порог эффективности скважины: 5;
- 3) расчет эффективного дебита нефти (значения выходной переменной): средний дебит нефти по скважине за первые 3 года эксплуатации.

Рассмотрим результаты формирования базы знаний при указанном наборе значений параметров.

### ***Формирование базы знаний при оптимальном наборе значений параметров***

В результате анализа исходных данных и построения нейронечеткой модели сформирована база знаний для определения вариантов ГТМ (ввода скважин в эксплуатацию) на нефтяном месторождении [91].

В таблице 4.25 представлен фрагмент сформированной БЗ.

Таблица 4.25 – Фрагмент базы знаний

pronc	poris	debsos	obvod	result	CF
1	1	1	1	0	0,02
1	1	1	2	0	0,02
1	1	1	3	0	0,01
1	1	1	4	0	0,01
1	1	2	1	0	0,06
1	1	2	1	1	0,08
1	1	2	2	0	0,04
1	1	2	2	1	0,01
1	1	2	3	0	0,03
1	1	2	4	0	0,03
1	1	3	1	0	0,07
1	1	3	1	1	0,08
1	1	3	2	0	0,09
1	1	3	3	0	0,05
1	1	3	3	1	0,08
1	1	3	4	0	0,02
1	1	3	5	0	0,08

База знаний состоит из 448 нечетких правил. Каждое правило включает в себя следующие входные переменные [91]:

- pronc – абсолютная проницаемость породы;
- poris – пористость породы;
- debsos – средний дебит соседних скважин;
- obvod – минимальная обводненность соседних скважин.

Значения входных переменных «1», «2», «3», «4», «5» означают, соответственно, «низкий», «ниже среднего», «средний», «выше среднего», «высокий».

Точность классификации данных при обучении ННС составила 97,54%, точность классификации при тестировании составила 95,7%, точность классификации при валидации модели – 95,68%. Достигнутая точность классификации данных характеризует высокую аппроксимирующую способность сформирован-

ной базы знаний, а, следовательно, ее адекватность и возможность практического использования [91].

Таким образом, полученные результаты точности классификации свидетельствуют о высокой обобщающей способности сформированной базы знаний (совместно с алгоритмом логического вывода) и возможности ее эффективного использования в составе СППР по подбору ГТМ (ввода скважин в эксплуатацию) на нефтяном месторождении [91].

В результате исследования разработаны и зарегистрированы следующие программы для ЭВМ:

1) «Программа построения и использования нечетких правил принятия решений по выбору проектных скважин для бурения на нефтяных месторождениях» [93];

2) «Программный комплекс поддержки принятия решений по формированию множества предпочтительных вариантов геолого-технических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяного месторождения» [79].

Первая программа использовалась на этапе подготовки данных к анализу, обучения ННС и формирования БЗ, используемой в дальнейшем для принятия решений по выбору проектных скважин для бурения на нефтяных месторождениях. Вторая программа представляет собой полноценную СППР, предназначенную для формирования множества предпочтительных вариантов геолого-технических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяного месторождения.

#### **4.3.2 Формирование базы знаний для оценки функционального состояния водителей автотранспортных средств**

##### *Актуальность решаемой задачи*

В настоящее время транспорт является неотъемлемой частью существования мировой инфраструктуры. Без автомобильных перевозок не представляется

ни один вид профессиональной деятельности. Поэтому вопрос транспортной безопасности касается каждого человека [21].

Защита интересов участников дорожного движения от незаконного вмешательства является важнейшей задачей. В настоящее время транспортная безопасность регулируется Федеральным законом «О транспортной безопасности» от 09.02.2007 № 16-ФЗ (ред. от 11.06.2021). Статья 11 данного закона посвящена информационному обеспечению в области транспортной безопасности [21; 122; 170].

Безопасность транспортного средства зависит от следующих факторов [21; 114; 124]:

- 1) безопасность транспортной единицы (в первую очередь зависит от производителя и от компании, занимающейся сервисным обслуживанием);
- 2) безопасность транспортных путей (зависит от вида транспорта, для каждого из которых имеются свои критерии оценки);
- 3) безопасная эксплуатация транспорта (по этому вопросу существует достаточно много руководящих документов и стандартов).

Основные влияющие факторы транспортной безопасности – это человеческий фактор, необходимость снижения рисков возникновения чрезвычайных ситуаций, рассмотрению которых последнее время посвящено много работ как в области профессиональной и психологической подготовки, так и в области создания автоматизированных систем управления транспортом [21; 154].

Одним из необходимых условий безопасной эксплуатации транспортного средства является необходимость нахождения водителя в нормальном функциональном состоянии (ФС) [21]. Эффективным методом оценки ФС человека является метод пупиллометрии, который основывается на его зрачковой реакции на световое импульсное воздействие [12; 125].

### ***Использование метода пупиллометрии для оценки функционального состояния водителей***

Как известно, реакция зрачков на изменения освещенности является безусловным рефлексом, который никак не контролируется со стороны головного

мозга [21; 183]. Поэтому метод пупиллометрии для оценки функционального состояния человека является объективным [129].

При изменении освещенности размеры зрачка меняются. Временной ряд данных, характеризующий изменение размера зрачка при изменении освещенности, называется пупиллограммой [123; 176]. Для ее получения человек располагается перед видеокамерой, после чего производится подача светового импульса и в течение 3-х секунд производится видеозапись зрачковой реакции. При этом на каждом кадре видеопоследовательности фиксируются нормированные размеры зрачков относительно размера радужной оболочки глаза [14]. Полученную таким образом пупиллограмму можно представить в виде следующего временного ряда [14; 21]:

$$P = \{(t_0, D_0), \dots, (t_i, D_i), \dots, (t_k, D_k)\},$$

где  $t_0$  – начальный момент времени,  $D_0$  – начальный нормированный диаметр зрачка,  $t_k$  – конечный момент времени,  $D_k$  – конечный нормированный диаметр зрачка, а  $t_i$  и  $D_i$  – соответственно момент времени и нормированный диаметр зрачка на  $i$ -м интервале временного ряда [21].

На рисунке 4.20 приведен пример графического изображения пупиллограммы.

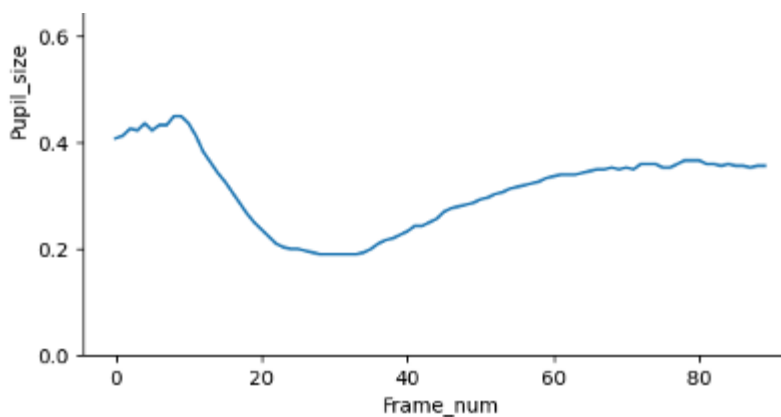


Рисунок 4.20 – Пример графического изображения пупиллограммы

По оси абсцисс указаны номера кадров видеопоследовательности, а по оси ординат – нормированные диаметры зрачков. В зависимости от поведения временного ряда, построенного для конкретного человека в конкретный момент времени, можно судить о его текущем функциональном состоянии.



### Описание и подготовка исходных данных к анализу

В работе в качестве исходных данных для анализа выступали временные ряды (пупиллограммы) в количестве 1200, используемые в [13] для оценки функционального состояния водителей автотранспортных средств в процедуре пред-рейсового медицинского осмотра. Половина исходных данных соответствовала нормальному функциональному состоянию человека, а оставшиеся 600 пупиллограмм – состоянию, характеризующемуся как «отклонение от нормы» [13].

Каждая пупиллограмма характеризовалась набором количественных параметров, представленных в таблице 4.26 [13].

Таблица 4.26 – Параметры пупиллограмм и формулы для расчета их значений

Параметр пупиллограммы	Формула для расчета значения параметра
Начальный нормированный диаметр зрачка	$D_0$
Минимальный нормированный диаметр зрачка	$D_{min} = \min(D_i)$ , где $i = 0..k$
Конечный нормированный диаметр зрачка	$D_k$
Время сужения зрачка	$t_s = t_i \mid D_i = D_{min}$
Время расширения зрачка	$t_r = t_k - t_s$
Амплитуда сужения	$A_s = D_0 - D_{min}$
Скорость сужения зрачка	$V_s = A_s / t_s$
Скорость расширения зрачка	$V_r = (D_k - D_{min}) / t_r$
Время с момента подачи светового импульса до начала сужения (латентное время реакции)	$t_l = \min(t_i) \mid D_i < D_0$
Диаметр половинного сужения	$D_{ps} = (D_0 + D_{min}) / 2$
Время половинного сужения	$t_{ps} = t_i < t_s \mid D_i = D_{ps}$
Время половинного расширения	$t_{pr} = t_i - t_s \mid D_i = D_{ps}$

Для выбора информативных параметров пупиллограмм с точки зрения их влияния на оценку функционального состояния человека произведен корреляционный анализ значений указанных параметров со значениями выходной переменной *Class* (0 – «норма», 1 – «отклонение») [103; 147; 152].

На рисунке 4.21 представлены результаты корреляционного анализа.













Поле	Class	/
$D_{min}$		0,999
$D_{ps}$		0,832
$A_s$		-0,596
$t_s$		-0,582
$t_r$		0,582
$t_{pr}$		0,424
$t_l$		-0,383
$D_0$		0,360
$V_r$		-0,256
$V_s$		-0,149
$t_{ps}$		0,085
$D_k$		0,065

Рисунок 4.21 – Результаты корреляционного анализа для выбора информативных параметров пупиллограмм

Задав порог значимости корреляционной зависимости на уровне 0,5, были выбраны следующие параметры, наиболее значимо влияющие на результат оценки функционального состояния человека:

- $D_{min}$  – минимальный нормированный диаметр зрачка;
- $D_{ps}$  – диаметр половинного сужения;
- $A_s$  – амплитуда сужения;
- $t_s$  – время сужения зрачка;
- $t_r$  – время расширения зрачка.

Следует отметить, что суммарное время сужения и расширения зрачка всегда равно 3 секунды (что соответствует условиям получения исходных пупиллограмм). Это означает, что параметры  $t_s$  и  $t_r$  являются линейно зависимыми между собой. Поэтому один из параметров ( $t_r$ ) был исключен из анализа. Следовательно, в итоговую выборку вошли 4 параметра:  $D_{min}$ ,  $D_{ps}$ ,  $A_s$  и  $t_s$ .

На рисунке 4.22 представлен фрагмент полученной выборки данных.

Dmin	Dps	As	ts	Class
0,0192	0,1527	0,2671	0,68	0
0,0073	0,2325	0,4504	0,52	0
0,0032	0,0471	0,0878	1,76	0
0,0036	0,1135	0,2198	0,88	0
0,0367	0,3511	0,6287	1,48	0
0,0095	0,2952	0,5715	0,2	0
0,0346	0,2731	0,4769	1,48	0
0,0109	0,3123	0,6027	1,4	0
0,0050	0,0144	0,0188	0,84	0
0,0187	0,2817	0,5260	2,88	0
0,3059	0,3774	0,1430	1,68	1
0,3157	0,3938	0,1562	2,16	1
0,3089	0,3861	0,1544	1,4	1
0,3087	0,3831	0,1489	1,28	1
0,3265	0,3776	0,1021	1,32	1
0,3053	0,3601	0,1097	1,4	1
0,3122	0,3701	0,1158	2,48	1
0,3174	0,3891	0,1434	1,12	1
0,3172	0,3773	0,1201	2,48	1
0,3129	0,3773	0,1289	1,88	1

Рисунок 4.22 – Фрагмент выборки данных для анализа

В представленном на рисунке фрагменте выборки данных первые 10 записей характеризуют функциональное состояние человека как «норма» (Class = 0), а оставшиеся 10 записей – как «отклонение» (Class = 1). На основе полученных данных обучена ННС и сформирована БЗ для оценки функционального состояния водителей [211].

### ***Формирование и исследование базы знаний для оценки функционального состояния водителей***

Для формирования БЗ использован разработанный программный комплекс [21]. Рассмотрим ее фрагмент и основные характеристики [20; 92].

В таблице 4.27 представлен фрагмент БЗ для оценки функционального состояния водителей.

Таблица 4.27 – Фрагмент базы знаний для оценки функционального состояния водителей

$D_{min}$	$D_{ps}$	$A_s$	$t_s$	$Class$	$CF$
3	3	1	2	1	0,519
1	2	2	1	0	0,326
3	3	2	2	1	0,281
1	1	1	2	0	0,14
1	3	3	2	0	0,138
3	3	1	1	1	0,122
3	3	2	1	1	0,122
1	1	2	1	0	0,099
1	2	2	2	0	0,095
1	2	3	1	0	0,095
1	1	2	2	0	0,092
3	3	1	3	1	0,087

База знаний состоит из 52 нечетких правил. Значения входных параметров «1», «2», «3» означают, соответственно, «низкий», «средний», «высокий».

Точность классификации данных при обучении ННС составила 98%, точность классификации при тестировании – 95,3%, точность классификации при валидации – 96,7%. Достигнутая точность классификации характеризует высокую аппроксимирующую способность сформированной базы знаний, а, следовательно, ее адекватность и возможность эффективного практического использования.

Следует отметить, что в работе [13] с помощью построения модели коллектива нейронных сетей была достигнута точность классификации данных на уровне 96,7%. При формировании базы знаний достигнута точность классификации 98%, что свидетельствует об эффективности разработанной технологии нейронечеткого моделирования. Следовательно, сформированную базу знаний можно эффективно использовать в составе системы поддержки принятия решений для оценки функционального состояния водителей автотранспортных средств [21].

### *Перспективы использования сформированной базы знаний*

Сформированная база знаний является адекватной с точки зрения решения поставленной задачи. Поэтому целесообразно ее практическое использование для поддержки принятия решений при оценке функционального состояния водителей [21].

На рисунке 4.23 представлена схема использования БЗ в составе СППР при прохождении водителем процедуры предрейсового медицинского осмотра [13].

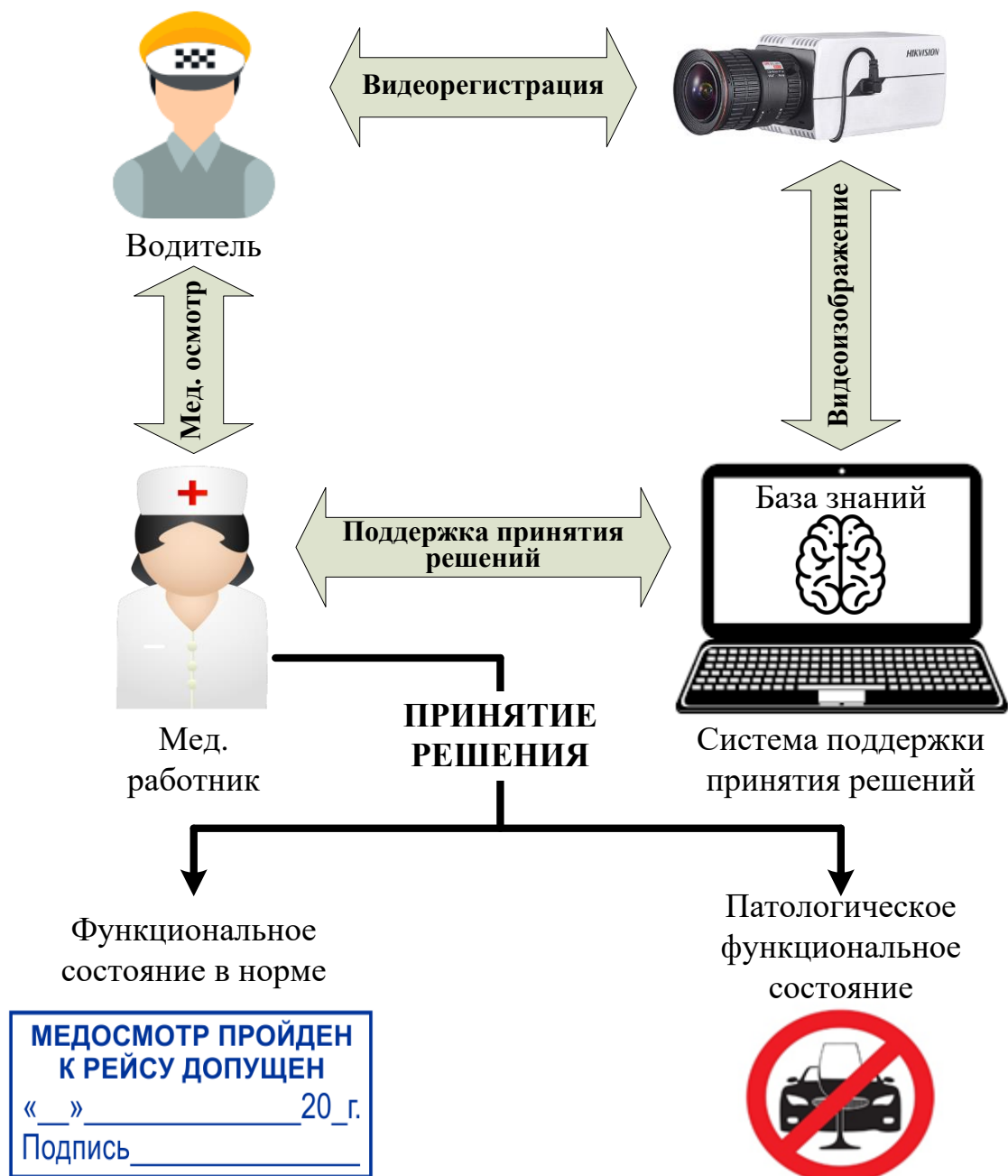


Рисунок 4.23 – Схема использования базы знаний в процедуре предрейсового медицинского осмотра водителей

Данные, полученные в результате видеорегистрации зрачковой реакции водителя, передаются в СППР. На основе нечетких правил базы знаний производится оценка ФС водителя, передаваемая медицинскому работнику. На основе этой информации, а также по результатам медицинского осмотра медицинский работник принимает окончательное решение о допуске водителя к управлению автотранспортом.

Как было показано в [13], использование такого подхода позволяет сократить материально-технические и временные затраты на прохождение предрейсового медицинского осмотра. Кроме того, использование СППР позволяет сделать эту процедуру более объективной с минимальным влиянием человеческого фактора.

Таким образом, в описанном исследовании решена актуальная задача оценки функционального состояния водителей автотранспортных средств на основе формирования нечеткой базы знаний. Полученная база знаний имеет высокую точность определения функционального состояния человека и позволяет получить интерпретируемый результат в виде решающего нечетко-продукционного правила [21]. Это повышает доверие со стороны пользователя интеллектуальной системы (медицинского работника) и помогает ему принять обоснованное решение.

### **4.3.3 Формирование базы знаний для определения фишинговых сайтов**

#### ***Актуальность определения фишинговых сайтов***

В настоящее время в сети интернет злоумышленники используют различные виды мошенничества, одним из которых является фишинг [118]. Существуют следующие методы фишинга [53]:

- голосовой фишинг – злоумышленник посредством телефонной связи пытается получить доступ к конфиденциальной информации клиентов банков с целью хищения их средств;

- SMS-фишинг – распространение сообщений, содержащих ссылки на фишинговые сайты;

- фишинг-копье – рассылки электронных писем конкретным пользователям;

- фишинг в поисковых системах – создание поддельных веб-страниц, проиндексированных в поисковой системе по определенным ключевым словам.

Фишинг получил большое распространение в интернет посредством использования фишинговых сайтов [120]. Такие сайты создаются для получения злоумышленником доступа к конфиденциальным данным пользователя. В большинстве методов фишинга используются замаскированные поддельные ссылки на фишинговые сайты, которые ведут на копию оригинального сайта организации. Злоумышленники чаще всего для этого используют интернет-адреса, в которых намеренно вносятся опечатки. Создание фишинговых сайтов не только приносит финансовые убытки, но и значительно снижают авторитет компании среди ее клиентов. Поэтому создание новых эффективных методов борьбы с фишинговыми сайтами является актуальным.

Одним из методов фишинга является использование URL-адресов, похожих на адреса оригинальных сайтов. При этом злоумышленникам не составляет труда заманить пользователей, заставив их переходить по мошенническим ссылкам.

Чтобы справиться с этой угрозой, наилучшей стратегией является препятствие для подключения к фишинговым сайтам путем определения фишинговых URL-адресов [25]. Большинство контрмер в этой области основаны на использовании баз данных с фишинговыми сайтами. Однако такой способ защиты является не эффективным. Для решения этой актуальной задачи предлагается подход, основанный на построении нейронечеткой модели [19; 99], а также на формировании и использовании нечеткой базы знаний [18].

### *Основные подходы к определению фишинговых сайтов*

Существует три базовых подхода к определению фишинговых сайтов [169]. Первый – ручная классификация, при которой пользователи самостоятельно определяют фишинговый сайт. Критерием для этого может быть использование

схожего URL-адреса (например, вместо «online.sberbank.ru» используется «online.sbrbank.ru»), отсутствие у сайта SSL-сертификата, грамматические или орфографические ошибки и т.д. Такая классификация не всегда эффективна, так как не все пользователи обладают достаточным количеством знаний, опыта и квалификации для определения фишинговых сайтов.

Второй подход заключается в том, чтобы использовать ключевые слова, которые есть на странице сайта, и проверять их в поисковой системе. Если сайт будет находиться в первых строках выдачи поисковой системы, то его можно считать легитимным, иначе – фишинговым. Данный подход может быть автоматизирован. Однако это требует больших вычислительных и временных затрат.

Третий подход использует алгоритмы машинного обучения [119]. Такие алгоритмы позволяют на основе анализа данных строить интеллектуальные модели классификации [16; 131; 148; 162; 180]: нейронные сети, нечеткие и нейронечеткие модели, базы знаний и др.

### *Получение и подготовка исходных данных к анализу*

В настоящее время наиболее актуально использование методов машинного обучения. При этом перспективным направлением исследований является разработка методов, сочетающих достоинства обучения по прецедентам (нейронные сети) с достоинствами дедуктивного обучения (нечеткие системы). Одной из реализаций этой концепции являются нейронечеткие модели [99; 140]. Актуальность их применения заключается в том, что такие модели формируют нечеткие БЗ для решения задач с интерпретацией полученного результата.

Автоматизация определения фишинговых сайтов в рамках нейронечеткого подхода потребовала решения следующих задач:

- сбора и подготовки данных для анализа;
- обучения ННС для формирования БЗ;
- проведения исследований для оценки эффективности этого подхода.

В работе для построения нейронечеткой модели использовались два набора данных, соответствующих фишинговым и легитимным сайтам. Данные формиро-



вались из различных источников. В качестве источника фишинговых сайтов использовались данные сервиса PhishTank [142; 224]. Объем данных составил 25000 уникальных URL-адресов. Для получения данных о легитимных сайтах использовались два источника: Alexa Internet и Common Crawl [163], а также написана утилита на языке Python. В результате собраны данные о 25000 легитимных сайтах. Таким образом, общее количество исходных данных составило 50000.

Из исходных данных экспертным путем выделено 10 входных признаков:

- 1) ранг сайта;
- 2) соответствие URL сайта его ip-адресу;
- 3) наличие регистрации у сайта;
- 4) время жизни сайта;
- 5) длина URL сайта;
- 6) наличие «@» в ссылке на сайт;
- 7) наличие редиректа в ссылке;
- 8) наличие символа «тире» в ссылке на сайт;
- 9) длина домена;
- 10) количество поддоменов в URL-адресе.

Выходной признак – это тип сайта: фишинговый (1) или легитимный (0).

Для оценки информативности входных признаков проведен корреляционный анализ [134], результаты которого представлены в таблице 4.28.

Таблица 4.28 – Коэффициенты корреляции входных признаков с выходным

№ п/п	Наименование входного признака	Коэффициент корреляции
1	ранг сайта (ranking)	0,517
2	соответствие URL сайта его ip-адресу (isIp)	0,013
3	наличие регистрации у сайта (valid)	-0,267
4	время жизни сайта (activeDuration)	-0,523
5	длина URL сайта (urlLen)	0,397
6	наличие «@» в ссылке на сайт (is@)	0,04
7	наличие редиректа в ссылке (isredirect)	0,073
8	наличие символа «тире» в ссылке на сайт (haveDash)	0,24
9	длина домена (domainLen)	0,232
10	количество поддоменов в URL-адресе (numOfSubdomain)	0,113

Корреляционный анализ позволил выбрать следующие информативные признаки (при пороге значимости 0,25):

- ранг сайта (ranking);
- наличие регистрации у сайта (valid);
- время жизни сайта (activeDuration);
- длина URL-адреса (urlLen).

Данные по остальным признакам были исключены. К оставшимся данным применены процедуры оценки их качества и очистки (исключения строк с выбросами и аномальными значениями). Результирующая выборка составила 34718 строк по 4 входным и 1 выходному признакам. Эти данные были использованы для обучения ННС и формирования БЗ.

### *Описание сформированной базы знаний*

В результате обучения ННС сформирована БЗ, фрагмент которой представлен в таблице 4.29.

Таблица 4.29 – Фрагмент сформированной базы знаний

ranking	valid	activeDuration	urlLen	type	CF
1	2	1	1	0	0,194
1	1	4	2	0	0,187
1	2	1	2	0	0,169
1	1	4	1	0	0,153
5	2	1	3	1	0,105
1	1	2	2	1	0,101
1	1	3	2	0	0,093
5	2	1	4	1	0,087
1	2	1	1	1	0,086
1	2	1	2	1	0,077
1	1	5	2	0	0,069
1	1	3	2	1	0,062

Общий объем базы знаний составил 68 нечетко-продукционных правил. Точность классификации данных при обучении ННС составила 93,1%, точность классификации при тестировании модели – 91,3%, точность классификации при валидации – 96,6%. Достигнутая точность классификации характеризует высокую аппроксимирующую способность сформированной базы знаний, а, следовательно, ее адекватность и возможность эффективного практического использования для определения фишинговых сайтов в сети Интернет [225].

Для оценки значимости полученных результатов произведено сравнение точности базы знаний с точностью нейросетевой модели [45], модели дерева решений [138], линейной и логистической регрессии [26]. Эти классификаторы были построены в среде аналитической платформы Deductor [103; 152].

В таблице 4.30 представлены полученные результаты.

Таблица 4.30 – Сравнение точности различных методов классификации

Метод классификации	Точность классификации, %
Многослойный персептрон	92,75
Дерево решений	92
Линейная регрессия	90
Логистическая регрессия	87
Нечеткая база знаний	96,6

Точность классификации на основе БЗ на 3,85% превышает точность нейросетевой классификации, на 4,6% – точность дерева решений, на 6,6% – точность на основе модели линейной регрессии и на 9,6% – на основе модели логистической регрессии. Таким образом, при использовании сформированной базы знаний достигается наибольшая точность классификации.

### *Перспективы использования сформированной базы знаний*

Сформированная база знаний может использоваться, как расширение в браузерах для определения фишинговых сайтов (см. рис. 4.24).

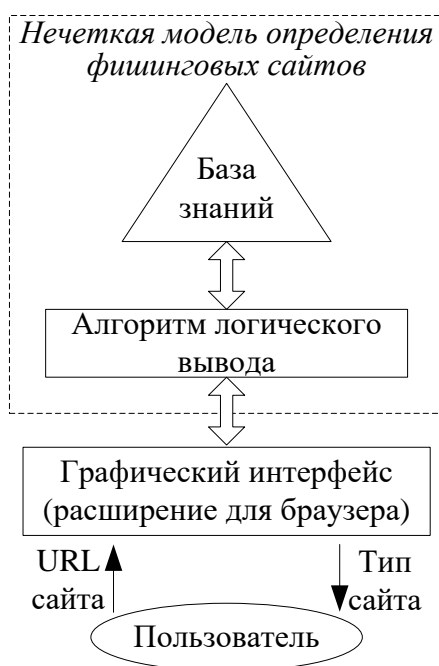


Рисунок 4.24 – Схема использования базы знаний для определения фишинговых сайтов

Как видно из рисунка, база знаний и алгоритм вывода представляют собой нечеткую модель определения фишинговых сайтов. Исходными данными для определения типа сайта является URL-адрес, введенный пользователем в строку браузера. При посещении фишингового сайта система должна предупредить и заблокировать доступ к его содержимому.

Таким образом, решена задача определения фишинговых сайтов. Полученные результаты показали эффективность предложенного подхода к решению поставленной задачи. Сформированная база знаний совместно с алгоритмом логического вывода показала высокую обобщающую способность. Это указывает на ее адекватность и возможность практического использования.

Следует отметить, что кроме описанных задач апробация разработанного математического, алгоритмического и программного обеспечения проводилась и в других предметных областях. В частности, на различных этапах диссертационного исследования решались задачи спам-классификации [80; 88; 101],

оценки вредоносности сетевого трафика в Android-приложениях [102], формирования баз знаний на примере анализа медицинских данных [96], анализа и прогнозирования временных рядов [95]. В каждой из задач показана эффективность предложенного подхода к формированию и использованию баз знаний. Однако в силу общности этих задач, описание их решения не вошло в содержание диссертации, а представлено в соответствующих публикациях.

#### 4.4 Выводы

На базе предложенных методов и алгоритмов построения нейронечеткой модели и формирования БЗ реализован программный комплекс, выполненный в среде Visual Studio 2019 на языке программирования C# и имеющий дружелюбный графический интерфейс. Комплекс состоит из модулей сэмплинга данных, построения нейронечеткой модели, формирования БЗ, оценки состояния объектов, а также модуля для выполнения исследований и экспериментов. Каждый модуль обладает соответствующими функциональными возможностями, в совокупности составляющими возможности всего программного комплекса. Он может быть использован не только для формирования баз знаний, но и выполнять роль СППР для оценки состояния объектов.

Для оценки эффективности реализованных в программном комплексе методов и алгоритмов потребовалось проведение следующих исследований:

- исследование влияния числа градаций и формы функций принадлежности на результаты инициализации и обучения ННС;
- исследование влияния значений параметров генетического алгоритма на результаты построения нейронечетких моделей и формирования БЗ;
- оценка точности классификации на основе формируемых баз знаний;
- сравнение точности различных методов классификации.

Для исследований выбраны 3 набора данных из репозитория машинного обучения UCI: «Iris Data Set», «Banknote Authentication Data Set» и «User Knowledge

Modeling Data Set». Выбранные наборы содержат реальные данные из различных предметных областей и соответствуют задаче оценки состояния объектов.

Проведенные исследования по оценке влияния числа градаций и формы ФП на результаты инициализации и обучения ННС показали целесообразность выбора не более 5÷6 градаций входных переменных с трапецеидальными ФП.

Исследования по оценке влияния значений параметров ГА на результаты построения нейронечетких моделей и формирования БЗ показали целесообразность выбора следующих параметров:

- длина хромосомы в ГА: 10;
- число хромосом в ГА: 150;
- вероятность мутации генов дочерних хромосом: 5%;
- число эпох холостой работы ГА: 5;
- число циклов холостого обучения ННС: 2.

В результате проведенных исследований достигнута высокая (не ниже 95%) точность классификации по всем наборам данных. При этом время обучения ННС составило от 7 минут до 2,5 часов в зависимости от набора данных, что является приемлемым для систем нейронечеткого моделирования. Для каждого набора данных сформированы соответствующие базы знаний. При обучении ННС выбирались рекомендуемые значения параметров.

В качестве меры эффективности баз знаний (совместно с алгоритмом вывода) использовалась обобщающая способность. Для набора данных «Banknote Authentication Data Set» она составила 99,2%, для «Iris Data Set» – 98,2%, для «User Knowledge Modeling Data Set» – 95,4%. При этом все базы знаний получились адекватными (их обобщающая способность совместно с алгоритмом логического вывода составляет не менее 95%).

Для окончательной оценки адекватности баз знаний произведено сравнение точности классификации наборов данных различными методами. Для этого в аналитической платформе Deductor построены следующие модели классификации для каждого из набора данных: нейронная сеть, дерево решений, логистическая и линейная регрессия. Исследование показало, что в среднем по всем наборам дан-

ных точность классификации на основе сформированных баз знаний на 2,42% превышает точность других рассмотренных методов. Кроме того, применение нечетко-продукционных правил совместно с алгоритмом логического вывода улучшает интерпретируемость принимаемых решений. Следовательно, формируемые базы знаний могут быть эффективно использованы в различных предметных областях, позволяя с высокой точностью производить интерпретируемую оценку состояния объектов.

Проведена апробация разработанного программного комплекса при формировании баз знаний для оценки состояния объектов в различных предметных областях: в нефтяной отрасли, в области интеллектуальных транспортных систем и в сфере информационной безопасности.

Сформирована база знаний для подбора ГТМ (ввода скважин в эксплуатацию) на нефтяном месторождении. Ее объем составил 448 нечетко-продукционных правил. Точность классификации обучающих данных составила 97,54%, тестовых данных – 95,7%, валидационных – 95,68%.

Сформирована база знаний для оценки функционального состояния водителей автотранспортных средств. Ее объем составил 52 нечетких правила. Точность классификации данных при обучении ННС составила 98%, при тестировании – 95,3%, при валидации – 96,7%.

Сформирована база знаний для определения фишинговых сайтов. Объем базы знаний составил 68 правил. Точность классификации данных при обучении составила 93,1%, при тестировании – 91,3%, при валидации – 96,6%.

Все сформированные базы знаний обладают высокой аппроксимирующей способностью, свидетельствующей о возможности их эффективного практического использования для решения поставленных задач.

## ЗАКЛЮЧЕНИЕ

В диссертационной работе на основе выполненных исследований решена актуальная научная задача, имеющая значение с точки зрения развития и применения средств математического моделирования, численных методов и комплексов программ для формирования и использования в СППР нечетких баз знаний по оценке состояния объектов в различных предметных областях. Обоснована необходимость разработки нового математического и алгоритмического обеспечения по формированию и использованию баз знаний для оценки состояния объектов в условиях разнотипности и некомплектности входных данных. С учетом особенностей решаемой задачи сформулированы требования к виду нечетких правил, алгоритму логического вывода на правилах и к нечеткой нейронной сети. Решение научной задачи потребовало создания научно-обоснованной нейронечеткой модели, методов и алгоритмов, а также комплекса программ формирования и использования баз знаний для оценки состояния объектов.

Математическое и алгоритмическое обеспечение, разработанное в диссертации, состоит из следующих основных компонентов:

1) вид нечетко-продукционных правил для обработки разнотипных данных и алгоритм логического вывода на правилах, позволяющий обосновывать предлагаемые решения по оценке состояния объектов;

2) нейронечеткая модель, основанная на обучении ННС и позволяющая формировать базу знаний нечетко-продукционного типа для оценки состояния объектов;

3) методы и алгоритмы построения нейронечеткой модели, позволяющие реализовать этапы ее структурной и параметрической идентификации (настраивать значения параметров функций принадлежности и весов связей в ННС);

4) метод и алгоритм формирования базы знаний, реализующие построение оптимальных маршрутов прохождения входных образов по слоям ННС, генерацию нечетко-продукционных правил, соответствующих построенным маршрутам, и получение системы уникальных правил, составляющих искомую базу знаний.



На основе предложенных методов и алгоритмов реализован комплекс программ, позволяющий автоматизировать процессы формирования и использования баз знаний для оценки состояния объектов в условиях разнотипности и некомплектности входных данных для принятия решений. Проведенные исследования при анализе известных наборов данных позволили выбрать оптимальные значения параметров генетического алгоритма, обосновать целесообразность использования не более 5÷6 градаций входных переменных с трапецеидальными формами функций принадлежности и сформировать за приемлемое время (от 7 минут до 2,5 часов в зависимости от набора данных) адекватные базы знаний, обобщающая способность которых совместно с алгоритмом вывода составляет не менее 95%.

В среднем по всем наборам данных достигнуто повышение точности классификации на основе сформированных баз знаний на 2,42% по сравнению с точностью других известных методов (нейронной сети, дерева решений, линейной и логистической регрессии). Кроме того, улучшена интерпретируемость принимаемых решений за счет применения нечетко-продукционных правил предложенного вида совместно с разработанным алгоритмом логического вывода.

Сформированы базы знаний для подбора ГТМ (ввода скважин в эксплуатацию) на нефтяном месторождении, для оценки функционального состояния водителей автотранспортных средств и для определения фишинговых сайтов. Результаты апробации показали эффективность разработанного математического и алгоритмического обеспечения, а также пригодность программного комплекса к решению поставленных задач. Обобщающая способность сформированных баз знаний (совместно с алгоритмом вывода) составила соответственно 95,68%, 96,7% и 96,6%.

Таким образом, в диссертации решены все поставленные задачи, а ее цель, заключающаяся в повышении точности оценки состояния объектов и улучшении интерпретируемости принимаемых решений в условиях разнотипности и некомплектности входных данных, достигнута.

Перспективы диссертационного исследования связаны с разработкой и оценкой методов нечеткого гранулирования значений входных переменных в ННС, применением гладких форм функций принадлежности нечетких гранул, структурной оптимизацией ННС в процессе обучения, а также формированием и использованием баз знаний для оценки состояния объектов в других предметных областях.

## СПИСОК СОКРАЩЕНИЙ

- БЗ** – база знаний;
- ВВ** – валидационная выборка;
- ГА** – генетический алгоритм;
- ГТМ** – геолого-техническое мероприятие;
- ЛПР** – лицо, принимающее решения;
- ННС** – нечеткая нейронная сеть;
- ОВ** – обучающая выборка;
- СППР** – система поддержки принятия решений;
- ТВ** – тестовая выборка;
- ФП** – функция принадлежности;
- ФС** – функциональное состояние;
- CF** – Certainty Factor (коэффициент уверенности, вес правила).

## СПИСОК ЛИТЕРАТУРЫ

1. Абдуллаев, Н.Т. Использование нейронно-нечетких систем для настройки весов правил нечетких продукций при принятии решений в электромиографии / Н.Т. Абдуллаев, К.Ш. Исмаилова // Нейрокомпьютеры: разработка, применение. – 2013. – № 10. – С. 64-69.
2. Абдулхаков, А.Р. Математическое и программное обеспечение редукции нечетких правил в базах знаний интеллектуальных систем / А.Р. Абдулхаков, А.С. Катасёв: Монография. – Казань: ЦИТ, 2015. – 160 с.
3. Аверкин, А.Н. Когнитивные гибридные системы поддержки принятия решений и прогнозирования / А.Н. Аверкин, С.А. Ярушев, В.Ю. Павлов // Программные продукты и системы. – 2017. – № 4. – С. 632-642.
4. Аверкин, А.Н. Мягкие вычисления и измерения / А.Н. Аверкин, С.В. Прокочина // Интеллектуальные системы. – 1997. – № 2. – С. 93-114.
5. Ажмухамедов, И.М. Использование нечеткого когнитивного моделирования для оценки эффективности судебной системы / И.М. Ажмухамедов, Д.А. Мачуева, Г.Г. Галимзянова // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – 2016. – № 1. – С. 41-49.
6. Ажмухамедов, И.М. Методика оценки уровня безопасности информационных активов на основе нечетких продукционных правил / И.М. Ажмухамедов, О.М. Князева // Проблемы информационной безопасности. Компьютерные системы. – 2015. – № 1. – С. 7-16.
7. Аникин, И.В. Модели нечетких нейронных сетей / И.В. Аникин // Эволюционное моделирование / Под ред. В.А. Райхлина. Труды Казанского городского семинара «Методы моделирования». Вып. 2. – Казань: Наука, 2004. – С. 111-136.
8. Аникин, И.В. Распознавание динамической рукописной подписи на основе нечеткой логики / И.В. Аникин, Э.С. Анисимова // Вестник Казанского государственного энергетического университета. – 2016. – № 3 (31). – С. 48-64.

9. Апанович, В.С. О возможностях реализации генетических алгоритмов на языке программирования C# / В.С. Апанович // Актуальные направления научных исследований XXI века: теория и практика. – 2018. – Т. 6, № 6 (42). – С. 27-30.
10. Астахова, И.Ф. Алгоритм обучения нечеткой нейронной сети Ванга-Менделя для распознавания рукопечатных символов в работе почтовой службы / И.Ф. Астахова, В.А. Мищенко, А.В. Красноярров // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. – 2011. – № 2. – С. 144-148.
11. Ахатова, Ч.Ф. Алгоритм параметрической идентификации адаптивной нейронечеткой модели / Ч.Ф. Ахатова // Международная молодежная научная конференция «XXII Туполевские чтения (школа молодых ученых)». Материалы конференции. – Казань, 2015. – С. 44-48.
12. Ахметвалеев, А.М. К вопросу о бесконтактном определении физиологического состояния человека / А.М. Ахметвалеев, А.С. Катасёв, М.П. Шлеймович // Вестник НЦБЖД. – 2015. – № 1 (23). – С. 13-21.
13. Ахметвалеев, А.М. Нейросетевая модель и программный комплекс определения функционального состояния опьянения человека по зрачковой реакции на световое импульсное воздействие / А.М. Ахметвалеев: Дисс. на соиск. уч. степ. канд. техн. наук. – Казань, 2018. – 163 с.
14. Ахметвалеев, А.М. Нейросетевая модель определения функционального состояния опьянения человека в решении отдельных задач обеспечения транспортной безопасности / А.М. Ахметвалеев, А.С. Катасёв // Компьютерные исследования и моделирование. – 2018. – Т. 10, № 3. – С. 285-293.
15. Балашов, О.В. Сравнительная оценка параметров нечетких моделей в задаче аппроксимации функции / О.В. Балашов // Нейрокомпьютеры: разработка, применение. – 2009. – № 8. – С. 49-51.
16. Барабанов, В.Ф. Разработка программного обеспечения для классификации сигналов с использованием продукционной базы знаний / В.Ф. Барабанов,

- Н.И. Гребенникова, Д.Н. Донских // Вестник Воронежского государственного технического университета. – 2015. – Т. 11, № 3. – С. 45-48.
17. Бардамова, М.Б. Применение алгоритмов генерации данных для построения нечетких классификаторов / М.Б. Бардамова // Сборник избранных статей научной сессии ТУСУР. – 2021. – № 1-2. – С. 184-187.
  18. Баринов, А.И. Использование модели нечетких нейронных сетей для формирования базы знаний по определению фишинговых сайтов / А.И. Баринов, Д.В. Катасёва, А.С. Катасёв // Вестник технологического университета. – 2020. – Т. 23, № 10. – С. 64-67.
  19. Баринов, А.И. Распознавание фишинговых сайтов на основе модели коллектива нечетких нейронных сетей / А.И. Баринов, Д.В. Катасёва, А.С. Катасёв // Международный форум KAZAN DIGITAL WEEK - 2020. – Казань, 2020. – С. 200-208.
  20. Баринова, А.О. Формирование базы знаний для оценки функционального состояния водителей в интеллектуальных транспортных системах / А.О. Баринова, Д.В. Катасёва, А.С. Катасёв // Международный форум KAZAN DIGITAL WEEK - 2020. – Казань, 2020. – С. 208-215.
  21. Баринова, А.О. Формирование и использование базы знаний для оценки функционального состояния водителей автотранспортных средств / А.О. Баринова, Д.В. Катасёва, А.С. Катасёв // Вестник Технологического университета. – 2020. Т. 23, № 10. – С. 75-78.
  22. Батыршин, И.З. Нечеткие гибридные системы. Теория и практика / Под ред. Ярушкиной Н.Г. / И.З. Батыршин, А.О. Недосекин, А.А. Стецко, В.Б. Тарасов, А.В. Язенин, Н.Г. Ярушкина – М.: ФИЗМАТЛИТ, 2007. – 208 с.
  23. Батыршин, И.З. Методы представления и обработки нечеткой информации в интеллектуальных системах / И.З. Батыршин // Новости искусственного интеллекта. – 1996. – № 2. – С. 9-65.
  24. Батыршин, И.З. Основные операции нечеткой логики и их обобщения / И.З. Батыршин. – Казань: Отечество, 2001. – 100 с., ил.

25. Белицкий, В.Ю. Распространенные виды мошенничеств в сети интернет / В.Ю. Белицкий // Актуальные проблемы современности. – 2020. – № 2 (28). – С. 31-36.
26. Белько, И.В. Применение логистической регрессии при обработке экономических данных / И.В. Белько, Е.А. Криштапович // Экономика, моделирование, прогнозирование. – 2018. – № 12. – С. 177-180.
27. Берштейн, Л.С. Модели представления нечетких темпоральных знаний в базах временных рядов / Л.С. Берштейн, С.М. Ковалев, А.В. Муравский // Известия Российской академии наук. Теория и системы управления. – 2009. – № 4. – С. 130-141.
28. Бова, В.В. Применение генетических алгоритмов для оптимизации параметров нейросетевой модели в задачах извлечения знаний / В.В. Бова, А.Н. Дуккардт // Информатика, вычислительная техника и инженерное образование. – 2012. – № 3 (10). – С. 1-10.
29. Божко, А.Н. Рациональное упорядочение альтернатив в диалоге с ЛПР / А.Н. Божко // Наука и образование: научное издание МГТУ им. Н.Э. Баумана. – 2013. – № 2. – С. 311-322.
30. Большаков, А.А. Выбор модели представления знаний для интеллектуальной системы управления качеством обработки заготовок в производстве печатных плат / А.А. Большаков, Л.В. Малый // Прикаспийский журнал: управление и высокие технологии. – 2019. – № 2 (46). – С. 30-40.
31. Большаков, А.А. Разработка системы поддержки принятия решений для проектирования автостереоскопических дисплеев / А.А. Большаков, А.В. Ключиков // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – 2020. – № 4. – С. 38-48.
32. Будько, М.Б. Синтез полетного нейроконтроллера с помощью метода обучения на основе генетического алгоритма / М.Б. Будько, М.Ю. Будько, А.В. Гирник, В.А. Грозов // Вестник компьютерных и информационных технологий. – 2019. – № 4 (178). – С. 3-12.

33. Булдакова, Т.И. Анализ разнотипных данных в слабоструктурированной задаче бюджетного кредитования / Т.И. Булдакова, А.Ш. Джалолов // Математические методы в технике и технологиях - ММТТ. – 2018. – Т. 2. – С. 105-108.
34. Бухаров, Т.А. Обзор среды разработки программных приложений Visual Studio / Т.А. Бухаров, А.Р. Нафикова // Colloquium-journal. – 2019. – № 14-2 (38). – С. 101-104.
35. Васильев, В.И. Оценка рисков информационной безопасности с использованием нечетких продукционных когнитивных карт / В.И. Васильев, А.М. Вульфин, М.Б. Гузаиров // Информационные технологии. – 2018. – Т. 24, № 4. – С. 266-273.
36. Васильев, В.И. Экспертная система поддержки принятия решений при управлении рисками в процессе аудита информационной безопасности / В.И. Васильев, Т.З. Хисамутдинов, П.В. Матвеев, А.С. Красько // Информационная безопасность: матер. VII междунар. науч.-практ. конф. Таганрог: Изд-во ТРТУ, 2005. – С. 43-46.
37. Верещагина, С.С. Методы поддержки принятия решений при диагностировании промышленного электротехнического оборудования на основе нечеткой логики / С.С. Верещагина: Дисс. на соиск. уч. степ. канд. техн. наук. – Самара, 2021. – 163 с.
38. Вешнева, И.В. Архитектура интеллектуальной системы мониторинга и поддержки принятия решений на предприятиях пищевой промышленности / И.В. Вешнева, А.А. Большаков, А. Федорова // Проблемы управления и моделирования в сложных системах. Труды XXI Международной конференции. В 2-х томах. Под редакцией С.А. Никитова, Д.Е. Быкова, С.Ю. Боровика, Ю.Э. Плешивцевой. 2019. – С. 378-382.
39. Волович, М.Е. Методика расширения возможностей визуального OLAP-анализа / М.Е. Волович, И.С. Зизганов // Наука без границ. – 2018. – № 4 (21). – С. 85-93.
40. Воронецкий, Ю.О. Методы борьбы с переобучением искусственных нейронных сетей / Ю.О. Воронецкий, Н.А. Жданов // Научный аспект. – 2019. – Т. 13, № 2. – С. 1639-1647.

41. Вульфин, А.М. Алгоритмы обработки информации для диагностирования инженерной сети нефтедобывающего предприятия с интеллектуальной поддержкой принятия решений / А.М. Вульфин: Дисс. на соиск. уч. степ. канд. техн. наук. – Уфа, 2012. – 202 с.
42. Вульфин, А.М. Интеллектуальный анализ видеоданных в системе контроля соблюдения правил промышленной безопасности / А.М. Вульфин // Моделирование, оптимизация и информационные технологии. – 2020. – Т. 8, № 2 (29) – С. 1-16.
43. Гаврилова, Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский. – СПб.: Питер, 2001. – 384 с.: ил.
44. Гагарин, А.В. Интеллектуальный алгоритм оптимизации параметров ресурсоемких моделей / А.В. Гагарин // Информационные технологии. – 2008. – № 1. – С. 23-28.
45. Глазкова, А.В. Сравнение нейросетевых моделей для классификации текстовых фрагментов, содержащих биографическую информацию / А.В. Глазкова // Программные продукты и системы. – 2019. – № 2. – С. 263-267.
46. Глова, В.И. Методы многокритериального принятия решений в условиях неопределенности в задачах нефтедобычи / В.И. Глова, И.В. Аникин, М.Р. Шагиахметов. – Казань: Изд-во Казан. гос. техн. ун-та, 2004. – 31 с.
47. Глова, В.И. Система нечеткого моделирования для решения задач повышения нефтедобычи / В.И. Глова, И.В. Аникин, М.Р. Шагиахметов // Вестник Казанского государственного технического университета им. А.Н. Туполева. – 2001. – № 3. – С. 59-61.
48. Глова, В.И. Мягкие вычисления: учебное пособие / В.И. Глова, И.В. Аникин, А.С. Катасёв, М.А. Кривилёв, Р.И. Насыров. – Казань: Изд-во Казан. гос. техн. ун-та, 2010. – 206 с.
49. Глушенко, С.А. Обучение нейро-нечеткой сети с помощью генетического алгоритма / С.А. Глушенко, А.И. Долженко // Кибернетика и программирование. – 2017. – № 5. – С. 79-88.



50. Горелова, А.В. Алгоритм обратного распространения ошибки / А.В. Горелова, Т.В. Любимова // Наука и современность. – 2015. – № 38. – С. 151-156.
51. Григорьев, Ю.А. Обработка ненадежных знаний на основе нечеткой логики в системе продукции CLIPS / Ю.А. Григорьев // Информатика и системы управления. – 2002. – № 2 (4). – С. 74-81.
52. Гуськов, Г.Ю. Формирование базы знаний для поддержки процесса архитектурного проектирования программных систем / Г.Ю. Гуськов, А.М. Наместников, А.А. Романов, А.А. Филиппов // Онтология проектирования. – 2021. – Т. 11, № 2 (40). – С. 154-169.
53. Данько, О.С. Исследование техник фишинга и методов защиты от него / О.С. Данько, Т.А. Медведева // Молодой исследователь Дона. – 2021. – № 3 (30). – С. 60-66.
54. Дзенчарский, Н.Л. Поиск изображений с выделением особых точек на основе вейвлет-преобразования / Н.Л. Дзенчарский, М.В. Медведев, М.П. Шлеймович // Вестник Казанского государственного технического университета им. А.Н. Туполева. – 2011. – № 1. – С. 131-135.
55. Дмитриенко, В.Д. Решение проблемы дообучения классических нейронных сетей / В.Д. Дмитриенко, А.Ю. Заковоротный // Автоматизированные технологии и производства. – 2015. – № 4 (10). – С. 32-40.
56. Допира, Р.В. Методика и алгоритмы классификации воздушных объектов системой поддержки принятия решений / Р.В. Допира, А.Н. Потапов, В.Ю. Семенов, А.В. Гетманчук, М.В. Семин // Программные продукты и системы. – 2019. – № 1. – С. 115-123.
57. Дорогов, А.Ю. Измерение обобщающей способности нейронных сетей / А.Ю. Дорогов, В.С. Абатуров // Известия СПбГЭТУ ЛЭТИ. – 2013. – № 4. – С. 53-58.
58. Дубасова, А.А. Экспертные оценки как основа для принятия решения ЛПР / А.А. Дубасова, И.О. Бочков, М.Ю. Иванов, Н.А. Шевченко, М.Н. Петухов // Естественные и технические науки. – 2013. – № 2 (64). – С. 245-247.

59. Елисеева, Е.В. Математическая модель функциональной среды РИМ-системы на основе теории нечетких множеств и теории гранулирования / Е.В. Елисеева, Ю.С. Яковлев // Математические машины и системы. – 2009. – № 1. – С. 40-54.
60. Емалетдинова, Л.Ю. Моделирование диагностической деятельности врача на основе нечеткой нейронной сети / Л.Ю. Емалетдинова, Д.Ю. Стрункин // Искусственный интеллект и принятие решений. – 2010. – № 3. – С. 73-78.
61. Емалетдинова, Л.Ю. Нейронечеткая модель аппроксимации сложных объектов с дискретным выходом / Л.Ю. Емалетдинова, А.С. Катасёв, А.П. Кирпичников // Вестник Казанского технологического университета. – 2014. – Т. 17, № 1. – С. 295-299.
62. Емельянов, В.В. Динамические продукции и нечетко-темпоральные модели знаний / В.В. Емельянов, С.М. Ковалев, А.Е. Колоденкова // Мягкие измерения и вычисления. – 2018. – № 12 (13). – С. 51-56.
63. Еремеев, А.П. Методы и программные средства на основе нечетких таблиц решений для диспетчеризации лотов на производстве / А.П. Еремеев, О.В. Виноградов // Вестник Московского энергетического института. – 2009. – № 2. – С. 166-174.
64. Еремеев, А.П. Прототип диагностической системы поддержки принятия решений на основе интеграции байесовских сетей доверия и метода Демпстера-Шеффера / А.П. Еремеев, Р.Р. Хазиев, М.В. Зуева, И.В. Цапенко // Программные продукты и системы. – 2013. – № 1. – С. 3-6.
65. Жаравин, Д.Е. Использование генетических алгоритмов для обучения искусственной нейронной сети / Д.Е. Жаравин, Д.Ю. Козин, Д.Ю. Фомичев, С.Б. Федотовский // Вестник Вологодского государственного университета. Серия: Технические науки. – 2019. – № 2 (4). – С. 41-43.
66. Жбанова, Н.Ю. Структурная и параметрическая идентификация разностных нейронечетких переключаемых моделей и нечетких многоэтапных входных процессов / Н.Ю. Жбанова: Дисс. на соиск. уч. степ. канд. техн. наук. – Липецк, 2014. – 145 с.

67. Заболоцких, А.С. Влияние индивидуальных характеристик ЛПР на принятие решений / А.С. Заболоцких, И.А. Демененко // Интеграция наук. – 2017. – № 9 (13). – С. 36-37.
68. Заде, Л.А. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л.А. Заде. – М.: Мир, 1976. – 168 с.
69. Заяц, А.М. Построение нейронной сети классификации ирисов Фишера на базе JAVASCRIPT / А.М. Заяц, С.П. Хабаров // Известия Санкт-Петербургской лесотехнической академии. – 2019. – № 226. – С. 233-247.
70. Звездин, Е.Ю. Поэтапная оптимизация расстановки проектных скважин по неравномерной сетке с использованием программного модуля технико-экономической оценки запасов месторождений / Е.Ю. Звездин, М.И. Маннапов, А.В. Насыбуллин, Р.З. Саттаров, М.А. Шарифуллина, Р.Р. Хафизов // Нефтяное хозяйство. – 2019. – № 7. – С. 28-31.
71. Зинченко, Л.А. Бионические информационные системы и их практические применения / Л.А. Зинченко, В.М. Курейчик, В.Г. Редько. – М.: ФИЗМАТЛИТ, 2011. – 288 с.
72. Иванов, А.Д. Разработка приложения для анализа сетевого трафика и обнаружения сетевых атак / А.Д. Иванов, А.А. Кутищев, Е.Ю. Никитина // Вестник Пермского университета. Математика. Механика. Информатика. – 2021. – № 2 (53). – С. 57-64.
73. Игнатенко, А.М. Методы подготовки данных к анализу слабоструктурированных временных рядов / А.М. Игнатенко, И.Л. Макарова, А.С. Копырин // Программные системы и вычислительные методы. – 2019. – № 4. – С. 87-94.
74. Ильин, И.В. Оптимизация метода градиентного спуска с помощью нейронной сети / И.В. Ильин // Вестник современных исследований. – 2017. – № 4-1 (7). – С. 110-112.
75. Исмагилов, И.И. Перспективы развития методов прогнозирования на основе инструментария теории нечетких множеств и процедур групповой экспертизы / И.И. Исмагилов, Р.В. Бичурин // Современное состояние естественных и технических наук. – 2014. – № 16. – С. 32-35.

76. Камаев, В.А. Сравнение методов представления знаний на основе продукционной модели и семантико-фреймовой сети для поиска причин возникновения дефектов отливок / В.А. Камаев, Ю.Ф. Воронин // Кибернетика и программирование. – 2015. – № 1. – С. 1-15.
77. Карпова, А.Е. Kaggle - платформа для анализа данных / А.Е. Карпова // Вестник магистратуры. – 2018. – № 12-4 (87). – С. 48-49.
78. Катасёв, А.С. Свидетельство о государственной регистрации программы для ЭВМ № 2020660262. Программа для формирования баз знаний интеллектуальных систем поддержки принятия решений по оценке состояния объектов / А.С. Катасёв, Д.В. Катасёва. – М.: Роспатент, 2020.
79. Катасёв, А.С. Свидетельство о государственной регистрации программы для ЭВМ № 2020665887. Программный комплекс поддержки принятия решений по формированию множества предпочтительных вариантов геологических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяного месторождения / А.С. Катасёв, Д.В. Катасёва, И.В. Аникин, В.М. Трегубов, Л.Ю. Емалетдинова, Л.Р. Шайхразиева, Р.Г. Гирфанов, О.В. Денисов, Р.Г. Лазарева, Ф.М. Латифуллин, Р.З. Саттаров, Р.Р. Хафизов, А.В. Чирякин, М.А. Шарифуллина, А.В. Насыбуллин, Д.Р. Хаярова, Р.М. Шакирзянов. – М.: Роспатент, 2020.
80. Катасёв, А.С. Формирование базы знаний системы спам-фильтрации на основе нечеткой нейронной сети ANFIS / А.С. Катасёв, Д.В. Катасёва // Информационная безопасность и защита персональных данных. Проблемы и пути их решения: VII межрегиональная научно-практическая конференция. – Брянск, 2015. – С. 46-50.
81. Катасёв, А.С. Инвариантная нечетко-продукционная модель представления знаний в экспертных системах / А.С. Катасёв, Д.Р. Газимова // Вестник Казанского государственного технического университета им. А.Н. Туполева. – 2011. – № 1. – С. 142-148.
82. Катасёв, А.С. Математическое и программное обеспечение формирования баз знаний мягких экспертных систем диагностики состояния сложных объ-

- ектов: монография / А.С. Катасёв. – Казань: ГБУ «Республиканский центр мониторинга качества образования», 2013. – 200 с., ил.
83. Катасёв, А.С. Методы и алгоритмы формирования нечетких моделей оценки состояния объектов в условиях неопределенности / А.С. Катасёв // Вестник Технологического университета. – 2019. – Т. 22, № 3. – С. 138-147.
84. Катасёв, А.С. Методы формирования нечетких моделей оценки состояния объектов в условиях неопределенности / А.С. Катасёв: Дисс. на соиск. уч. степ. докт. техн. наук. – Казань, 2019. – 304 с.
85. Катасёв, А.С. Модели и методы формирования нечетких правил в интеллектуальных системах диагностики состояния сложных объектов / А.С. Катасёв: Дисс. на соиск. уч. степ. докт. техн. наук. – Казань, 2014. – 257 с.
86. Катасёв, А.С. Нейронечеткая модель формирования нечетких правил для оценки состояния объектов в условиях неопределенности / А.С. Катасёв // Компьютерные исследования и моделирование. – 2019. – Т. 11, № 3. – С. 477-492.
87. Катасёв, А.С. Нейронечеткая модель формирования правил классификации, как эффективный аппроксиматор объектов с дискретным выходом / А.С. Катасёв // Кибернетика и программирование. – 2018. – № 6. – С. 110-122.
88. Катасёв, А.С. Формирование нечетких правил фильтрации нежелательных электронных сообщений в инфокоммуникационных сетях / А.С. Катасёв, Д.В. Катасёва // Проблемы техники и технологий телекоммуникаций ПТиТТ-2014; Оптические технологии в телекоммуникациях ОТТ-2014. Материалы Международных научно-технических конференций. – Казань, 2014. – С. 320-322.
89. Катасёва, Д.В. Методика построения нейронечеткой модели формирования нечетких баз знаний для систем поддержки принятия решений / Д.В. Катасёва // XXV Туполевские чтения (школа молодых ученых): Международная молодежная научная конференция. – Казань, 2021. – С. 350-355.
90. Катасёва, Д.В. Модель нечеткого логического вывода для оценки состояния объектов / Д.В. Катасёва // Моделирование энергоинформационных процессов: X национальная научно-практическая конференция с международным участием. – Воронеж, 2021. – С. 101-105.

91. Катасёва, Д.В. Нейронечеткая модель и программный комплекс формирования баз знаний для оценки состояния объектов / Д.В. Катасёва // Прикаспийский журнал: управление и высокие технологии. – 2022. – № 1 (57). – С. 65-76.
92. Катасёва, Д.В. Нечетко-продукционная модель классификации и оценки состояния объектов в интеллектуальных системах поддержки принятия решений / Д.В. Катасёва // Международный форум KAZAN DIGITAL WEEK – 2021. – Казань, 2021. – С. 124-130.
93. Катасёва, Д.В. Свидетельство о государственной регистрации программы для ЭВМ № 2020666941. Программа построения и использования нечетких правил принятия решений по выбору проектных скважин для бурения на нефтяных месторождениях / Д.В. Катасёва, А.С. Катасёв. – М.: Роспатент, 2020.
94. Катасёва, Д.В. Свидетельство о государственной регистрации программы для ЭВМ № 2022612337. Нейронечеткая система формирования баз знаний для оценки состояния объектов / Д.В. Катасёва, А.И. Баринов. – М.: Роспатент, 2022.
95. Катасёва, Д.В. Технология анализа и прогнозирования временных рядов на основе нечеткой нейронной сети / Д.В. Катасёва // Аналитическая механика, устойчивость и управление. Труды XI Международной Четаевской конференции, посвященной 115-летию со дня рождения Н.Г. Четаева и памяти академика АН РТ Т.К. Сиразетдинова. – Казань, 2017. – С. 71-79.
96. Катасёва, Д.В. Формирование баз знаний интеллектуальных систем на примере нейронечеткого анализа медицинских данных / Д.В. Катасёва, А.О. Баринова // Вестник Технологического университета. – 2022. – Т. 25, № 2. – С. 67-70.
97. Катасёва, Д.В. Формирование и использование баз знаний для систем поддержки принятия решений по оценке состояния объектов / Д.В. Катасёва // Математические методы в технологиях и технике. – 2021. – № 8. – С. 97-100.
98. Катасёва, Д.В. Метод формирования нечетких баз знаний в системах поддержки принятия решений по оценке состояния объектов / Д.В. Катасёва // Проблемы вычислительной и прикладной математики. – 2022. – № 1(38). – С. 149-156.

99. Катасёва, Д.В. Нейронечеткая модель определения фишинговых сайтов / Д.В. Катасёва, А.И. Баринов // Вестник Технологического университета. – 2022. – Т. 25, № 1. – С. 69-72.
100. Катасёва, Д.В. Нечетко-продукционная модель оценки состояния объектов в системах поддержки принятия решений / Д.В. Катасёва // Вестник Технологического университета. – 2021. – Т. 24, № 12. – С. 105-108.
101. Катасёва, Д.В. Спам-классификация в инфокоммуникационных системах / Д.В. Катасёва, А.С. Катасёв // Информация и безопасность. – 2015. – Т. 18, вып. 3. – С. 380-383.
102. Катасёва, Д.В., Нейронечеткая модель оценки вредоносности сетевого трафика в Android-приложениях / Д.В. Катасёва, Д.Д. Лосева // Вестник Технологического университета. – 2022. – Т. 25, № 1. – С. 77-80.
103. Кацко, И.А. Практикум по анализу данных на компьютере: Учеб. пособие для вузов / И.А. Кацко, Н.Б. Паклин. – М.: Изд-во «КолосС», 2009. – 278 с.
104. Килин, Г.А. Структурно-параметрическая идентификация модели газотурбинной установки на основе генетического алгоритма / Г.А. Килин, К.А. Один, Б.В. Кавалеров // Фундаментальные исследования. – 2014. – № 11-7. – С. 1480-1484.
105. Клачек, П.М. Гибридные адаптивные интеллектуальные системы. Ч. 1: Теория и технология разработки: монография / П.М. Клачек, С.И. Корягин, А.В. Колесников, Е.С. Минкова. – Калининград: Изд-во БФУ им. И. Канта, 2011. – 374 с.
106. Климов, К.О. Особенности обработки больших данных в реальном времени в облаке AWS с использованием сервиса AWS KINESIS / К.О. Климов, Г.А. Пискун, Д.В. Лихачевский, В.Ф. Алексеев, В.В. Шаталова // Big Data and Advanced Analytics. – 2021. – № 7-1. – С. 322-327.
107. Ключева, И.А. Исследование аспектов применимости стратегий сэмплинга для решения проблемы несбалансированности структур данных / И.А. Ключева // Новые информационные технологии в научных исследованиях. материалы XXI Всероссийской научно-технической конференции студентов, молодых

- ученых и специалистов. Рязанский государственный радиотехнический университет. – 2016. – С. 198-199.
108. Ковалев, С.М. Построение базы знаний интеллектуальной системы контроля и предупреждения рискованных ситуаций для этапа проектирования сложных технических систем / С.М. Ковалев, А.Е. Колоденкова // *Онтология проектирования*. – 2017. – Т. 7, № 4 (26). – С. 398-409.
109. Колоденкова, А.Е. Интеллектуальная система поддержки принятия решений для диагностики и выбора схем лечения пациента / А.Е. Колоденкова, С.Г. Новокщенов // XIII Всероссийское совещание по проблемам управления ВСПУ-2019. Институт проблем управления им. В.А. Трапезникова РАН. – 2019. – С. 1879-1883.
110. Колядин, А.И. Вопросы выбора модели представления знаний в радиоэлектронной отрасли / А.И. Колядин // *Радиоэлектронная отрасль: проблемы и их решения*. – 2022. – № 5. – С. 36-40.
111. Комарцова, Л.Г. Нейросетевой метод извлечения знаний для мягкой экспертной системы / Л.Г. Комарцова // *Нейроинформатика*. – 2001. – Т. 1. – С. 124.
112. Комарцова, Л.Г. Эволюционные методы формирования нечетких баз правил / Л.Г. Комарцова // *Открытые семантические технологии проектирования интеллектуальных систем*. – 2011. – № 1. – С. 181-184.
113. Копкин, Е.В. Выбор дискретных диагностических признаков с учетом их ценности для распознавания технического состояния объекта / Е.В. Копкин, А.Н. Кравцов, О.Г. Лазутин // *Информация и космос*. – 2015. – № 2. – С. 111-117.
114. Корякин, В.М. О соотношении понятия «транспортная безопасность» со смежными понятиями в области безопасности на транспорте / В.М. Корякин // *Транспортное право и безопасность*. – 2017. – № 11 (23). – С. 26-32.
115. Кравченко, Т.К. Создание систем поддержки принятия решений: интеграция преимуществ отдельных подходов / Т.К. Кравченко, Н.Н. Середенко // *Искусственный интеллект и принятие решений*. – 2012. – № 1. – С. 39-47.
116. Кремлева, Э.Ш. Модели и методы интеллектуальной обработки данных для систем поддержки принятия решений (на примере систем экологической



- безопасности) / Э.Ш. Кремлева: Дисс. на соиск. уч. степ. канд. техн. наук. – Казань, 2021. – 180 с.
117. Кремлева, Э.Ш. Автоматическая генерация рекомендуемых систем на основе качественной интерпретации мониторинговой информации / Э.Ш. Кремлева, А.П. Снегуренко, С.В. Новикова, Н.Л. Валитова // Вестник Тверского государственного университета. Серия: Прикладная математика. – 2020. – № 3. – С. 50-67.
118. Крюкова, И.В. Фишинг как вид интернет-мошенничества / И.В. Крюкова, Э.Н. Алимамедов // Наукосфера. – 2021. – № 2-2. – С. 196-201.
119. Кубрин, Г.С. Применение методов машинного обучения для определения фишинговых сайтов / Г.С. Кубрин, Д.В. Иванов // Методы и технические средства обеспечения безопасности информации. – 2020. – № 29. – С. 23-24.
120. Кубрин, Г.С. Разработка классификаторов фишинговых сайтов на основе динамически формируемой выборки / Г.С. Кубрин, Д.В. Иванов // Проблемы информационной безопасности. Компьютерные системы. – 2021. – № 2 (46). – С. 74-81.
121. Кумаритов, А.М. Анализ малой выборки экспериментальных данных при управлении газоснабжением региона / А.М. Кумаритов, А.Э. Дзгоев, Р.Б. Шарипов // Известия высших учебных заведений. Проблемы энергетики. – 2018. – Т. 20, № 1-2. – С. 62-69.
122. Кунцман, М.В. Инвестиционная поддержка транспортной инфраструктуры для обеспечения транспортной безопасности в РФ / М.В. Кунцман, А.А. Султыгова, А.С. Швырева // Автомобиль. Дорога. Инфраструктура. – 2016. – № 4 (10). – С. 14-29.
123. Куприянов, А.С. Методы обработки и анализа пупиллограмм / А.С. Куприянов // Известия высших учебных заведений. Приборостроение. – 2009. – Т. 52, № 8. – С. 58-63.
124. Курчин, В.О. Безопасность транспортных средств, ее роль в решении проблем обеспечения безопасности дорожного движения / В.О. Курчин, В.Ф.

- Карев // Автомобильный транспорт Дальнего Востока. – 2018. – № 1. – С. 185-190.
125. Куцало, А.Л. Динамическая пупиллометрия как метод скрининг диагностики отравлений промышленными токсикантами / А.Л. Куцало, М.В. Цимбал, Д.С. Хомич, М.Г. Вареников, Н.В. Штейнберг // Медицина экстремальных ситуаций. – 2018. – Т. 20, № S3. – С. 487-493.
126. Лабинский, А.Ю. Аппроксимация функций многих переменных нечеткой нейронной сетью / А.Ю. Лабинский // Природные и техногенные риски (физико-математические и прикладные аспекты). – 2017. – № 2 (22). – С. 11-17.
127. Лабинский, А.Ю. Использование адаптивной гибридной нечеткой нейронной сети для аппроксимации функций // Проблемы управления рисками в техно-сфере. – 2020. – № 1 (53). – С. 89-95.
128. Липатов, А.А. Методы формального представления и обработки неопределенных количественных и качественных данных для выявления групп объектов / А.А. Липатов, В.Н. Ушаков // Мягкие измерения и вычисления. – 2018. – № 3 (4). – С. 16-24.
129. Липницкая, А.В. Сравнительный анализ прямой и содружественной реакций зрачков на действие света / А.В. Липницкая, В.А. Прохоцкая // Медицина и здравоохранение: материалы V международной научной конференции. – 2017. С. 19-22.
130. Локтев, Д.А. Оценка параметров состояния объектов по их образам в системах неразрушающего контроля / Д.А. Локтев // Приборы и системы. Управление, контроль, диагностика. – 2020. – № 10. – С. 14-21.
131. Ломовцева, Н.А. Использование модуля нейронных сетей в системе Statistica для решения задачи классификации / Н.А. Ломовцева, Ю.Е. Кувайскова // Вестник Ульяновского государственного технического университета. – 2021. – № 4 (96). – С. 59-61.
132. Малый, Л.В. Система интеллектуальной поддержки принятия решений поддержания качества производства печатных плат / Малый Л.В. // Математические методы в технике и технологиях - ММТТ. – 2019. – Т. 7. – С. 142-148.

133. Малых, В.Л. Системы поддержки принятия решений в медицине / В.Л. Малых // Программные системы: теория и приложения. – 2019. – Т. 10, № 2 (41). – С. 155-184.
134. Малютина, О.П. Особенности применения коэффициента линейной корреляции Пирсона в психологических, педагогических и акмеологических исследованиях (на примере обучения студентов гуманитарных специальностей) / О.П. Малютина // Вестник Воронежского государственного университета. Серия: Проблемы высшего образования. – 2013. – № 1. – С. 79-81.
135. Марахимов, А.Р. Повышение обобщающей способности нейронных сетей и селекция обучающих выборок / А.Р. Марахимов, К.К. Худайбергенов, Н.А. Игнатъев // Проблемы вычислительной и прикладной математики. – 2020. – № 2 (26). – С. 99-107.
136. Марухленко, А.Л. Организация системы сетевого мониторинга и оценки состояния информационной безопасности объекта / А.Л. Марухленко, К.Д. Селезнёв, М.О. Таныгин, Л.О. Марухленко // Известия Юго-Западного государственного университета. – 2019. – Т. 23, № 1. – С. 118-129.
137. Марушин, А.С. Исследование эффективности стандартного генетического алгоритма с модифицированными операторами скрещивания / А.С. Марушин // Актуальные проблемы авиации и космонавтики. – 2017. – Т. 2, № 13. – С. 47-48.
138. Мифтахова, А.А. Применение метода дерева решений для решения задач классификации и прогнозирования / А.А. Мифтахова // Инфокоммуникационные технологии. – 2016. – Т. 14, № 1. – С. 64-70.
139. Михайлова, А.Н. Использование метода колеса рулетки при решении задачи маршрутизации с помощью генетического алгоритма / А.Н. Михайлова // Вестник науки. – 2020. – Т. 1, № 5 (26). – С. 88-91.
140. Мохаммед, А.Г. Нейронечеткая модель краткосрочного прогнозирования потребления электроэнергии / А.Г. Мохаммед // Прикаспийский журнал: управление и высокие технологии. – 2013. – № 2 (22). – С. 047-056.

141. Мунерман, В.И. Параллельная обработка данных средствами Microsoft Azure / В.И. Мунерман, О.С. Слепенкова // Системы компьютерной математики и их приложения. – 2017. – № 18. – С. 102-104.
142. Найденов, А.С. Применение методов машинного обучения в задаче обнаружения атак типа подмена адреса ресурса в сети интернет / А.С. Найденов // Вестник Рязанского государственного радиотехнического университета. – 2017. – № 61. – С. 46-53.
143. Наместников, А.М. Нейрокомпьютинг как средство извлечения знаний / А.М. Наместников, А.Э. Рахматулин, Н.Г. Сапегин // Вестник Ульяновского государственного технического университета. – 1999. – № 2 (6). – С. 106-111.
144. Насыбуллин, А.В. Дебит многозабойной горизонтальной скважины в пяти и семи точечном элементе / А.В. Насыбуллин, В.Ф. Войкин // Нефтяная провинция. – 2015. – № 3 (3). – С. 65-75.
145. Насыбуллин, А.В. Оптимизация размещения проектных скважин с использованием проектного модуля для технико-экономической оценки запасов нефтяных месторождений / А.В. Насыбуллин, Д.А. Разживин, Ф.М. Латифуллин, Р.З. Сагтаров, С.В. Смирнов, Р.Р. Хафизов, М.А. Шарифуллина // Нефтяная провинция. – 2018. – № 4 (16). – С. 163-174.
146. Недосекин, А.О. Нечетко-множественная оценка стоимости объекта интеллектуальной собственности / А.О. Недосекин, З.И. Абдулаева, А.Н. Козловский // Аудит и финансовый анализ. – 2019. – № 4. – С. 83-86.
147. Новиков, С.П. Применение корреляционного анализа для исследования экспериментальных данных / С.П. Новиков, Е.Ю. Зайцева // Молодой исследователь Дона. – 2019. – № 4 (19). – С. 57-64.
148. Новикова, С.В. Кодирование мягких циклических данных при помощи псевдо-нечеткой меры / С.В. Новикова, Э.Ш. Кремлева, Н.Л. Валитова // Вестник Тверского государственного университета. Серия: Прикладная математика. – 2019. – № 3. – С. 90-101.
149. Новикова, С.В. Способ управления качеством урбоэкосистемы с помощью качественной интерпретации количественного ответа системы нечеткого вы-

- вода / С.В. Новикова, Ю.А. Тунакова, А.Р. Шагидуллин // Академический вестник ELPIT. – 2018. – Т. 3, № 1 (3). – С. 32-55.
150. Нурушев, Е.Т. Проблемы и решение задач авиаремонтного завода в условиях неопределенности состояния объектов ремонта / Е.Т. Нурушев, Р.М. Хисматуллин, Ж.А. Даев // Автоматизация. Современные технологии. – 2018. – Т. 72, № 1. – С. 7-10.
151. Паклин, Н.Б. Адаптивные модели нечеткого вывода для идентификации нелинейных зависимостей в сложных системах: Дисс. на соиск. уч. степ. к-та техн. наук / Н.Б. Паклин. – Ижевск, 2004. – 167 с.
152. Паклин, Н.Б. Бизнес-аналитика: от данных к знаниям: учебное пособие / Н.Б. Паклин, В.И. Орешков. – 2-е изд., испр. – СПб.: Питер, 2013. – 704 с.
153. Поляков, И.В. Построение оптимальных каскадов Виолы-Джонса при помощи «жадных» алгоритмов перебора управляющих параметров с промежуточным контролем по валидационной выборке / И.В. Поляков, Е.Г. Кузнецова, С.А. Усилин, Д.П. Николаев // Сенсорные системы. – 2016. – Т. 30, № 3. – С. 241-248.
154. Поначугин, А.В. Современные вопросы разработки и внедрения автоматизированной системы управления на городском пассажирском транспорте / А.В. Поначугин, В.А. Соколов // Проблемы машиностроения и автоматизации. – 2019. – № 3. – С. 39-47.
155. Пономарева, К.А. Построение обоснованных классификационных моделей при принятии решений и прогнозировании / К.А. Пономарева, А.А. Ступина, А.В. Федорова, Л.Н. Корпачева // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. – 2021. – № 1. – С. 109-114.
156. Потапов, В.И. Модели и задачи оценки надежности нейронной системы при обучении и переобучении нейронной сети после отказов / В.И. Потапов // Информационные технологии. – 2011. – № 11. – С. 59-64.

157. Почесуева, Ю.Ю. Решение плохо формализованных задач с помощью гибридных интеллектуальных систем / Ю.Ю. Почесуева, Н.Г. Левченко // Научные проблемы транспорта Сибири и Дальнего Востока. – 2011. – № 2. – С. 28-31.
158. Ротштейн, А.П. Интеллектуальные технологии идентификации: нечеткая логика, генетические алгоритмы, нейронные сети / А.П. Ротштейн. – Винница: УНИВЕРСУМ - Винница, 1999. – 320 с.
159. Сафьянников, Н.М. Интеллектуальная измерительная система для реализации способа оценки функционального состояния центральной нервной системы человека / Н.М. Сафьянников, П.Н. Буренев // Международная конференция по мягким вычислениям и измерениям. – 2017. – Т. 2. – С. 179-182.
160. Северин, С.Н. Количественная оценка противопожарного состояния объекта защиты / С.Н. Северин, М.А. Тищенко, В.В. Шаптала // Проблемы управления рисками в техносфере. – 2020. – № 3 (55). – С. 27-34.
161. Седов, В.А. Использование языка Function block diagram для реализации треугольных функций принадлежности / В.А. Седов, Н.А. Седова // Информационные технологии. Радиоэлектроника. Телекоммуникации. – 2016. – № 6-2. – С. 214-218.
162. Сергиенко, М.А. Об одном подходе к формированию базы знаний для классификации объектов / М.А. Сергиенко, И.В. Данилова // Международный научно-исследовательский журнал. – 2020. – № 7-1 (97). – С. 96-103.
163. Сикулер, Д.В. Ресурсы, предоставляющие данные для машинного обучения и проверки технологий искусственного интеллекта / Д.В. Сикулер // Информационные и математические технологии в науке и управлении. – 2021. – № 2 (22). – С. 39-52.
164. Симонова, Л.А. Методика оценки величины остаточного ресурса основных узлов ДВС на основе нечеткой логики / Л.А. Симонова, Д.Н. Демьянов, А.А. Капитонов // Научно-технический вестник Поволжья. – 2021. – № 12. – С. 136-139.
165. Симонова, Л.А. Методика формирования альтернативных технологических маршрутов с применением фреймовой модели представления знаний / Л.А.

- Симонова, Г.М. Капитонова // Научно-технический вестник Поволжья. – 2020. – № 4. – С. 48-51.
166. Симонова, Л.А. Разработка базы знаний для системы нечеткого логического вывода процесса прецизионной штамповки / Л.А. Симонова, К.Н. Гавариева // Научно-технический вестник Поволжья. – 2020. – № 1. – С. 62-64.
167. Синьков, Д.В. Кодирование категориальных данных для использования в машинном обучении / Д.В. Синьков, А.Д. Ваничкин // Молодой ученый. – 2020. – № 21 (311). – С. 70-72.
168. Сурков, М.А. Обзор и анализ методов обогащения данных / М.А. Сурков // Точная наука. – 2021. – № 112. – С. 12-15.
169. Сычева, А.В. Некоторые способы совершения «дистанционного» мошенничества / А.В. Сычева // Вестник Волгоградской академии МВД России. – 2020. – № 4 (55). – С. 167-173.
170. Титова, Н.К. Понятие и содержание терминов «транспортная безопасность» и «угроза транспортной безопасности»: теоретический аспект / Н.К. Титова // Транспортное право. – 2012. – № 3. – С. 30-33.
171. Трусов, П.В. Математическая модель дыхательной системы человека с учетом эволюции функциональных нарушений / П.В. Трусов, М.Ю. Цинкер // Математическое моделирование в естественных науках. – 2016. – Т. 1. – С. 384-387.
172. Ушаков, В.А. Фреймово-продукционные и логические модели представления знаний в процессах обнаружения и диагностирования предотказных состояний в технических объектах / В.А. Ушаков, В.С. Дрогайцев, Д.Е. Крупейников // Вестник Саратовского государственного технического университета. – 2012. – Т. 2, № 2 (66). – С. 177-184.
173. Фархутдинова, К.Р. Процедура формирования нечеткой базы знаний с весовыми коэффициентами принятия решений для рынка сервисных услуг нефтегазового бизнеса / К.Р. Фархутдинова // Евразийский союз ученых. – 2016. – № 3-1 (24). – С. 116-119.

174. Финн, В.К. Обнаружение эмпирических закономерностей в последовательностях баз фактов посредством ДСМ-рассуждений / В.К. Финн // Научно-техническая информация. Серия 2: Информационные процессы и системы. – 2015. – № 8. – С. 1-29.
175. Финн, В.К. Эвристика обнаружения эмпирических закономерностей и принципы интеллектуального анализа данных / В.К. Финн // Искусственный интеллект и принятие решений. – 2018. – № 3. – С. 3-19.
176. Фоменко, В.Н. Математические модели зрачковых реакций глаза человека (пупиллограмм) / В.Н. Фоменко, А.С. Куприянов // Известия Петербургского университета путей сообщения. – 2010. – № 4 (25). – С. 220-231.
177. Фуремс, Е.М. Приближенное решение обратной задачи об упаковке в контейнеры с учетом предпочтений лица, принимающего решения / Е.М. Фуремс // Искусственный интеллект и принятие решений. – 2018. – № 3. – С. 112-121.
178. Ходашинский, И.А. Биоинспирированные методы параметрической идентификации нечетких моделей / И.А. Ходашинский, П.А. Дудин, А.В. Лавыгина // Доклады Томского государственного университета систем управления и радиоэлектроники. – 2007. – С.81-92.
179. Ходашинский, И.А. Идентификация нечетких систем: методы и алгоритмы / И.А. Ходашинский // Проблемы управления. – 2009. – № 4. – С. 15-23.
180. Ходашинский, И.А. Отбор признаков и построение нечеткого классификатора на основе алгоритма прыгающих лягушек / И.А. Ходашинский, М.Б. Бардамова, В.С. Ковалев // Искусственный интеллект и принятие решений. – 2018. – № 1. – С. 76-84.
181. Ходашинский, И.А. Применение генетического алгоритма для обучения нечетких систем типа синглтон / И.А. Ходашинский, А.В. Лавыгина // Обзорные прикладной и промышленной математики. – 2007. – Т. 14, № 6. – С. 1143-1144.
182. Хубаев, Г.Н. Алгоритмы классификации лиц, принимающих решения, по уровню профессиональных знаний и творческим способностям / Г.Н. Хубаев // Наука и мир. – 2016. – № 5-2 (33). – С. 168-176.



183. Цимбал, М.В. О референтных нормах показателей пупиллометрии при использовании компьютерного комплекса для регистрации и оценки зрачковой реакции / М.В. Цимбал, А.Л. Куцало, Н.В. Штейнберг, Д.С. Хомич // Вестник Волгоградского государственного медицинского университета. – 2020. – № 1 (73). – С. 124-128.
184. Цымблер, М.Л. Очистка сенсорных данных в интеллектуальных системах управления отоплением зданий / М.Л. Цымблер, Я.А. Краева, Е.А. Латыпова, Е.В. Иванова, Д.А. Шнайдер, А.А. Басалаев // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. – 2021. – Т. 10, № 3. – С. 16-36.
185. Челноков, Н.А. Решение задач классификации изображений методами анализа данных / Н.А. Челноков, В.В. Становов // Актуальные проблемы авиации и космонавтики. – 2018. – Т. 2, № 4 (14). – С. 107-109.
186. Шаврук, Е.Ю. Интеллектуальные технологии и системы искусственного интеллекта для поддержки принятия решений / Е.Ю. Шаврук, Г.Р. Ванкович // Проблемы экономического роста и устойчивого развития территорий. Материалы IV международной научно-практической интернет-конференции. – 2019. – С. 243-244.
187. Шафроненко, А.Ю. Адаптивная кластеризация данных с пропущенными значениями / А.Ю. Шафроненко, В.В. Волкова, Е.В. Бодянский // Радиоэлектроника, информатика, управление. – 2011. – № 2 (25). – С. 115-119.
188. Шведов, А.С. Аппроксимация функций с помощью нейронных сетей и нечетких систем / А.С. Шведов // Проблемы управления. – 2018. – № 1. – С. 21-29.
189. Юсуфов, Н.А. Применение сводных таблиц MS Excel для анализа массивов данных / Н.А. Юсуфов // Проблемы развития АПК региона. – 2015. – Т. 24, № 4 (24). – С. 134-137.
190. Ярушкина, Н.Г. Нечеткие нейронные сети (часть 1) / Н.Г. Ярушкина // Новости искусственного интеллекта. – 2001. – № 2-3. – С. 47-52.
191. Ярушкина, Н.Г. Нечеткие нейронные сети (часть 2) / Н.Г. Ярушкина // Новости искусственного интеллекта. – 2001. – № 4. – С. 23-29.

192. Ярушкина, Н.Г. Нечеткие нейронные сети в когнитивном моделировании и традиционных задачах искусственного интеллекта / Н.Г. Ярушкина // Научная сессия МИФИ-2005. VII всероссийская научно-техническая конференция «Нейроинформатика – 2005». Лекции по нейроинформатике. – 2005. – С. 166-212.
193. Atukorale, A.S. Boosting the hong network / A.S. Atukorale, T. Downs, P.N. Suganthan // Neurocomputing. – 2003. – Vol. 51. – P. 75-86.
194. Bache, K. UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>] / K. Bache, M. Lichman. – Irvine, CA: University of California, School of Information and Computer Science. 2013.
195. Banknote Authentication Data Set [Электронный ресурс]. – Режим доступа: <https://archive.ics.uci.edu/ml/datasets/banknote+authentication> (дата обращения: 31.03.2022).
196. Bismukhamedov, R.F. Generative transformer framework for network traffic generation and classification / R.F. Bismukhamedov, A.F. Nadeev // T-Comm. – 2020. – Vol. 14, No. 11. – P. 64-71.
197. Bolshakov, A.A. Decision support system for selecting designs of autostereoscopic displays / A.A. Bolshakov, A.V. Klyuchikov // Studies in systems, decision and control. – 2021. – No. 342. – P. 73-88.
198. Bolshakov, A.A., Decision support algorithm for parrying the threat of an accident / A.A. Bolshakov, A. Kulik, I. Sergushov, E. Scripal // Studies in Systems, Decision and Control. – 2020. – Vol. 260. – P. 237-247.
199. Bykovtsev, Y.A. Estimation of the accuracy of a control system with a fuzzy PID controller based on the approximation of the static characteristic of the controller / Y.A. Bykovtsev, V.M. Lokhin // Mechatronics, Automation, Control. – 2021. – Vol. 22, No. 12. – P. 619-624.
200. Carlsen, L. The Iris dataset revisited - a partial ordering study / L. Carlsen, R. Bruggemann // Informatica (Ljubljana). – 2020. – Vol. 44, No. 1. – P. 35-44.
201. Carpita, M. Exploring and modelling team performances of the Kaggle european soccer database / M. Carpita, E. Ciavolino, P. Pasca // Statistical Modelling. – 2019. – Vol. 19, No. 1. – P. 74-101.

202. Chalumuri, A. A hybrid classical-quantum approach for multi-class classification / A. Chalumuri, B.S. Manoj, R. Kune // *Quantum Information Processing*. – 2021. – Vol. 20. – No. 3. – P. 119-131.
203. Chupin, M.M. Neuro-fuzzy model in supply chain management for objects state assessing / M.M. Chupin, A.S. Katasev, A.M. Akhmetvaleev, D.V. Kataseva // *International Journal of Supply Chain Management*. – 2019. – Vol. 8, No.5. – P. 201-208.
204. Cordon, O. A novel framework to design fuzzy rule-based ensembles using diversity induction and evolutionary algorithms-based classifier selection and fusion / O. Cordon, K. Trawinski // *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. – 2013. – Vol. 7902 LNCS (PART 1). – P. 36-58.
205. Dzik, C.S. Real-time AWS resources monitoring and analytics / C.S. Dzik, I.I. Piletski // *Big Data and Advanced Analytics*. – 2021. – No. 7-1. – P. 25-30.
206. Eliseev, S.V. Improving the reliability and safety of vehicle operation as a task of evaluating and forming the dynamic states of technical objects / S.V. Eliseev, N.K. Kuznetsov, A.V. Eliseev, P.A. Lontsikh // *Journal of Advanced Research in Technical Science*. – 2020. – No. 20. – P. 27-33.
207. Gholamy, A. How to explain the efficiency of triangular and trapezoid membership functions in applications to design / A. Gholamy, O. Kosheleva, V. Kreinovich // *Ontology of Designing*. – 2019. – Vol. 9, No. 2 (32). – P. 253-260.
208. Hoffmann, F. Evolutionary Algorithms for fuzzy control system design / F. Hoffmann // *Proceedings of IEEE*. – 2001. – Vol. 20, No 5. – P. 1318-1333.
209. Ignatov, V.O. Method optimization of models parameters for the nonstationary traffic in telecommunication network / V.O. Ignatov // *Problems of Informatization and Management*. – 2008. – Vol. 1, No. 23. – P. 11-15.
210. Iris Data Set [Электронный ресурс]. – Режим доступа: <https://archive.ics.uci.edu/ml/datasets/iris> (дата обращения: 31.03.2022).
211. Ismagilov, I.I. Formation of a knowledge base to analyze the issue of transport and the environment / I.I. Ismagilov, A.A. Murtazin, D.V. Kataseva, A.S. Katasev,

- A.O. Barinova // *Caspian Journal of Environmental Sciences*. – 2020. – Vol. 18, No. 5. – P. 615-621.
212. Ismagilov, I.I. Fuzzy neural network model for rules generating of the objects state determining in uncertainty / I.I. Ismagilov, L.A. Molotov, A.S. Katasev, L.Y. Emaletdinova, D.V. Kataseva // *Helix*. – 2018. – Vol. 8, No. 6. – P. 4662-4667.
213. Ismagilov, I.I. Fuzzy regression analysis using trapezoidal fuzzy numbers / I.I. Ismagilov, G. Alsaied // *Industrial Engineering and Management Systems*. – 2020. – Vol. 19, No. 4. – P. 896-900.
214. Ivanov, A.I. Precision statistics: neuroet networking of chi-square test and shapiro-wilk test in the analysis of small selections of biometric data / A.I. Ivanov, S.E. Vjatchanin, E.A. Malygina, V.S. Lukin // *Reliability & Quality of Complex Systems*. – 2019. – No. 2 (26). – P. 27-34.
215. Kahraman, H.T. The development of intuitive knowledge classifier and the modeling of domain dependent data / H.T. Kahraman, S. Sagiroglu, I. Colak // *Knowledge-Based Systems*. – 2013. – Vol. 37. – P. 283-295.
216. Kolodenkova, A.E. Diagnostics of the technical condition of industrial equipment based on a system of hierarchical production rules / A.E. Kolodenkova, S.S. Vereshchagina, V.O. Tuvaeva, A.N. Guda // *Lecture Notes in Networks and Systems*. – 2022. – Vol. 330. – P. 180-187.
217. Kovalev, S. Fuzzy model based intelligent prediction of objective events / S. Kovalev, A. Sukhanov, V. Styskala // *Advances in Intelligent Systems and Computing*. – 2016. – Vol. 423. – P. 23-33.
218. Kultygin, O.P. Business intelligence as a decision support system tool / O.P. Kultygin, I.V. Lokhtina // *Journal of Applied Informatics*. – 2021. – Vol. 16, No. 1 (91). – P. 52-58.
219. Mamdani, E.H. Application of fuzzy logic to approximate reasoning using linguistic synthesis / E.H. Mamdani // *IEEE Transactions on Computers*. – 1977. – Vol. 26. – P. 1182-1191.

220. Mamdani, E.H. Experiment in linguistic synthesis with a fuzzy logic controller / E.H. Mamdani, S. Assilian // *International Journal of Human Computer Studies*. – 1999. – No 51(2). – P. 135-147.
221. Minnikhanov, R.N. Approaches for image processing in the decision support system of the center for automated recording of administrative offenses of the road traffic / R.N. Minnikhanov, I.V. Anikin, M.V. Dagaeva, T.I. Asliamov, T.E. Bolshakov // *Computer Research and Modeling*. – 2021. – Vol. 13, No. 2. – P. 405-415.
222. Mittal, M. Performance evaluation of threshold-based and k-means clustering algorithms using Iris dataset / M. Mittal, R.K. Sharma, V.P. Singh // *Recent Patents on Engineering*. – 2019. – Vol. 13, No. 2. – P. 131-135.
223. Nedosekin, A.O. The economic resilience evaluation using fuzzy sets and soft computing / A.O. Nedosekin, Z.I. Abdoulaeva, N.A. Karpenko, T.A. Nikitina // *Lecture Notes in Networks and Systems*. – 2022. – Vol. 307. – P. 105-112.
224. Orunsolu, A.A. A lightweight anti-phishing technique for mobile phone / A.A. Orunsolu, M.A. Alaran, A.A. Adebayo, S.O. Kareem, A. Oke // *Acta Informatica Pragensia*. – 2017. – Vol. 6, No. 2. – P. 114-123.
225. Panishev, O.Y. Creation of a fuzzy model for verification of malicious sites based on fuzzy neural networks / O.Y. Panishev, E.N. Ahmedshina, D.V. Kataseva, A.S. Katasev, A.M. Akhmetvaleev // *International Journal of Engineering Research and Technology*. – 2020. – Vol. 13, No. 12. – P. 4432-4438.
226. Panishev, O.Y. Neurofuzzy model of formation of knowledge bases for selection of geological and technical measures in oil fields / O.Y. Panishev, E.N. Ahmedshina, D.V. Kataseva, I.V. Anikin, A.S. Katasev, A.M. Akhmetvaleev, A.V. Nasybullin // *International Journal of Engineering Research and Technology*. – 2020. – Vol. 13, No. 11. – P. 3589-3595.
227. Pavlov, S.V. Model of a fuzzy dynamic decision support system / S.V. Pavlov, V.A. Dokuchaev, S.S. Mytenkov // *T-Comm*. – 2020. – Vol. 14, No. 9. – P. 43-47.
228. Sugeno, M. A note on derivatives of functions with respect to fuzzy measures / M. Sugeno // *Fuzzy Sets and Systems*. – 2013. – No 222. – P. 1-17.

229. Sugeno, M. On improvement of stability conditions for continuous Mamdani-like fuzzy systems / M. Sugeno, T. Taniguchi // *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*. – 2004. – No 34(1). – P. 120-131.
230. User Knowledge Modeling Data Set [Электронный ресурс]. – Режим доступа: <https://archive.ics.uci.edu/ml/datasets/User+Knowledge+Modeling> (дата обращения: 31.03.2022).
231. Vecherskaya, S.E. Selection of criteria for a decision support system for an art university / S.E. Vecherskaya // *Informatics and Education*. – 2021. – No. 3 (322). – P. 56-62.
232. Zadeh, L.A. Fuzzy Sets / L.A. Zadeh // *Information and Control*. – 1965. – Vol. 8. – P. 338-353.

Приложение 1 Свидетельства о регистрации  
программ для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020660262

Программа для формирования баз знаний  
интеллектуальных систем поддержки принятия решений  
по оценке состояния объектов

Правообладатель: *федеральное государственное бюджетное  
образовательное учреждение высшего образования "Казанский  
национальный исследовательский технический университет  
им. А.Н. Туполева - КАИ" (RU)*

Авторы: *Катасёв Алексей Сергеевич (RU),  
Катасёва Дина Владимировна (RU)*

Заявка № 2020619537


Дата поступления 25 августа 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 01 сентября 2020 г.



Руководитель Федеральной службы  
по интеллектуальной собственности

 Г.П. Ивлиев

## РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020665887

«Программный комплекс поддержки принятия решений по формированию множества предпочтительных вариантов геолого-технических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяного месторождения»

Правообладатель: *Государственное бюджетное образовательное учреждение высшего образования «Альметьевский государственный нефтяной институт» (RU)*

Авторы: *см. на обороте*

Заявка № 2020665114

Дата поступления 23 ноября 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 02 декабря 2020 г.

Руководитель Федеральной службы  
по интеллектуальной собственности

Г.П. Ивлиев





Авторы: *Катасёв Алексей Сергеевич (RU), Катасёва Дина Владимировна (RU), Аникин Игорь Вячеславович (RU), Трезубов Владимир Михайлович (RU), Емалетдинова Лилия Юперовна (RU), Шайхразиева Ляйсан Равилевна (RU), Гирфанов Руслан Габдульянович (RU), Денисов Олег Владимирович (RU), Лазарева Регина Геннадьевна (RU), Латифуллин Фарит Миннеахметович (RU), Саттаров Рамиль Зайтунович (RU), Хафизов Рафаэль Ринатович (RU), Чирикин Александр Владимирович (RU), Шарифуллина Мария Александровна (RU), Насыбуллин Арслан Валерьевич (RU), Хаярова Динара Рафаэлевна (RU), Шакирзянов Ринат Михайлович (RU)*

## РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2020666941

**Программа построения и использования нечетких правил  
принятия решений по выбору проектных скважин для  
бурения на нефтяных месторождениях**

Правообладатель: *федеральное государственное бюджетное  
образовательное учреждение высшего образования "Казанский  
национальный исследовательский технический университет  
им. А.Н. Туполева - КАИ" (RU)*

Авторы: *Катасёва Дина Владимировна (RU),  
Катасёв Алексей Сергеевич (RU)*

Заявка № 2020666508

Дата поступления 14 декабря 2020 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 18 декабря 2020 г.



*Руководитель Федеральной службы  
по интеллектуальной собственности*

*Г.П. Ивлиев*

## РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022612337

**Нейронечеткая система формирования баз знаний для  
оценки состояния объектов**

Правообладатель: *федеральное государственное бюджетное  
образовательное учреждение высшего образования  
"Казанский национальный исследовательский  
технический университет им. А.Н. Туполева - КАИ"  
(RU)*

Авторы: *Катасёва Дина Владимировна (RU), Баринов  
Андрей Игоревич (RU)*

Заявка № 2022611380

Дата поступления **04 февраля 2022 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **10 февраля 2022 г.**

*Руководитель Федеральной службы  
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ  
Сертификат 0x75699C000BAE3B78543B083DF819A6CD1  
Владелец: **Ивлиев Григорий Петрович**  
Действителен с 24.12.2021 по 24.12.2022

*Г.П. Ивлиев*

## Приложение 2 Акты о внедрении и использовании результатов диссертационного исследования

**УТВЕРЖДАЮ**

Директор

Центра обслуживания бизнеса

ПАО «Татнефть»



Р.Р. Павлов

2021 г.

**АКТ**

об опытно-промышленном внедрении  
результатов кандидатской диссертации

Катасёвой Дины Владимировны

**Мы, нижеподписавшиеся:**

Л.А. Камалова – зам. начальника управления инвестиций, к.э.н.;

Р.Г. Гирфанов – руководитель гр. МОБП ЦОБ;

О.В. Денисов – вед. бизнес-аналитик гр. МОБП ЦОБ, к.т.н.;

составили настоящий акт о том, что результаты диссертационной работы Катасёвой Дины Владимировны на соискание ученой степени кандидата технических наук, включающие:

- нейронечеткую модель формирования баз знаний по выбору проектных скважин для бурения на нефтяных месторождениях;
- методы инициализации параметров нейронечеткой модели;
- методы и алгоритмы настройки параметров нейронечеткой модели;
- алгоритм использования сформированных баз знаний для поддержки принятия решений по подбору геолого-технических мероприятий (ввода скважин в эксплуатацию) на нефтяных месторождениях;
- программный комплекс поддержки принятия решений по формированию множества предпочтительных вариантов геолого-технических мероприятий (ввода скважин в эксплуатацию) при разработке нефтяных месторождений

прошли апробацию в управлении инвестиций и Центре обслуживания бизнеса ПАО «Татнефть», внедрены в бизнес-процесс формирования оптимальной программы геолого-технических мероприятий по всем объектам разработки в условиях ресурсных ограничений с учетом вариативности налоговых моделей.

Практическое использование результатов работы позволило:

- произвести выбор оптимальной стратегии проведения геолого-технических мероприятий по объектам разработки ПАО «Татнефть» в условиях ресурсных и финансовых ограничений;

- произвести ранжирование объектов разработки с целью рекомендации перехода на налоговую модель на дополнительный доход от добычи углеводородного сырья;
- снизить нагрузку на геолого-технологический персонал по рациональному выбору проектных скважин для бурения на нефтяных месторождениях.

Считаем целесообразным рекомендовать разработанный программный комплекс к внедрению в крупных российских нефтегазодобывающих компаниях, эксплуатирующих множество объектов разработки с высокой наполненностью программы геолого-технических мероприятий (ПАО «Лукойл», ПАО «Сургутнефтегаз», ПАО «Газпром нефть»). Экономическая эффективность этого не вызывает сомнений.

Заместитель начальника управления инвестиций  
ПАО «Татнефть», к.э.н.

Л.А. Камалова

Руководитель группы методологического обеспечения бизнес-процессов  
Центра обслуживания бизнеса  
ПАО «Татнефть»

Р.Г. Гирфанов

Ведущий бизнес-аналитик  
группы методологического обеспечения бизнес-процессов  
Центра обслуживания бизнеса  
ПАО «Татнефть», к.т.н.

О.В. Денисов





Кроме того, результаты диссертационного исследования Катасёвой Д.В. используются на кафедре при выполнении обучающимися выпускных квалификационных, курсовых и научно-исследовательских работ.

Директор института компьютерных  
технологий и защиты информации,  
к.т.н., доцент



В.М. Трегубов

Заведующий кафедрой систем  
информационной безопасности,  
д.т.н., профессор



И.В. Аникин



**УТВЕРЖДАЮ**

Директор ГБУ «Безопасность  
дорожного движения»,  
член-корреспондент Академии наук  
Республики Татарстан,

Д.т.н., профессор



/ Р.Н. Минниханов /

«04» марта 2022 г.

**АКТ**

об использовании результатов диссертационной работы  
Катасёвой Дины Владимировны

Комиссия в составе:

председатель: Дагаева М.В., начальник Центра ИТС,

члены комиссии:

- Эминов Б.Ф., ведущий специалист сектора инновационного развития Центра ИТС, к.ф.-м.н., доцент,
- Сабитов А.А., главный специалист технического сектора Центра ИТС,
- Большаков Т.Е., главный специалист технического сектора Центра ИТС,
- Файзрахманов Э.М., главный специалист сектора международного взаимодействия Центра ИТС.

составила настоящий акт о том, что результаты кандидатской диссертации аспиранта кафедры систем информационной безопасности КНИТУ-КАИ Катасёвой Д.В., включающие вид нечетко-продукционных правил, алгоритм логического вывода на правилах, нейронечеткую модель, методы ее построения, а также реализующий их комплекс программ используются в научных исследованиях и разработках Центра интеллектуальных транспортных систем ГБУ «Безопасность дорожного движения» при формировании базы знаний для оценки функционального состояния водителей автотранспортных средств.



Полученные результаты нейронечеткого моделирования и сформированную базу знаний планируется использовать в разработке интеллектуальной системы поддержки принятия решений по оценке функционального состояния водителей при прохождении процедуры предрейсового медицинского осмотра.

Считаем целесообразным рекомендовать полученные в диссертации Катасёвой Д.В. результаты к использованию в других организациях для формирования баз знаний и поддержки принятия решений по оценке состояния объектов в различных предметных областях.

Председатель комиссии:

начальник Центра ИТС


 / Дагаева М.В. /

Члены комиссии:

ведущий специалист сектора


инновационного развития

Центра ИТС, к.ф.-м.н., доцент

 / Эминов Б.Ф. /


главный специалист

технического сектора Центра ИТС

 / Сабитов А.А. /

главный специалист


технического сектора Центра ИТС

 / Большаков Т.Е. /

главный специалист сектора

международного взаимодействия

Центра ИТС

 / Файзрахманов Э.М. /