Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра комплексной информационной безопасности
электронно-вычислительных систем (КИБЭВС)

К ЗАЩИТЕ ДОПУСТИТЬ
Заведующий кафедрой КИБЭВС
доктор техн. наук, профессор
_____А.А. Шелупанов
«___» _____ 2020 г.

АЛГОРИТМЫ И ПРОГРАММНЫЕ СРЕДСТВА ПОСТРОЕНИЯ НЕЧЕТКИХ КЛАССИФИКАТОРОВ НА ОСНОВЕ МЕТАЭВРИСТИКИ «РАЗРЯД МОЛНИИ»

Дипломная работа по специальности
10.05.03 – «Информационная безопасность автоматизированных систем»
Пояснительная записка
ФБ ДР.503200.001 ПЗ

СОГЛАСОВАНО

Консультант по экономике
Ст. преподаватель каф. КИБЭВС
_____ С.В. Глухарева
«_____» _____ 2020 г.

Студент гр. 725
_____ Н.Е. Мельникова
«_____» _____ 2020 г.

Консультант по вопросам охраны труда
Доцент каф. КИБЭВС, канд. техн. наук
_____ Е.М. Давыдова
«_____» _____ 2020 г.

Руководитель
Проф. каф. КИБЭВС, доктор техн. наук
_____ И.А. Ходашинский
«_____» _____ 2020 г.

Томск 2020

Реферат

Дипломная работа, 87 с., 23 рис., 28 табл., 37 источников, 1 прилож.

НЕЧЕТКИЙ КЛАССИФИКАТОР, МЕТАЭВРИСТИКА, АЛГОРИТМ ОПТИМИЗАЦИИ, КЛАССИФИКАЦИЯ, АЛГОРИТМ РАЗРЯД МОЛНИИ, БИНАРИЗАЦИЯ, ОТБОР ПРИЗНАКОВ.

Объектом исследования в работе является процесс классификации данных.

Предметом исследования является алгоритм на основе метаэвристики «Разряд молнии».

Цель работы – усовершенствование классификации данных с помощью метаэвристического алгоритма «Разряд молнии».

Для достижения описанной цели проведен аналитический обзор текущего состояния исследований. В ходе работы разработаны непрерывный и бинарный алгоритмы на основе метаэвристики «Разряд молнии». Данные алгоритмы реализованы в виде программного средства. Предложено применение метаэвристики в задачах минимизации функций. Алгоритмы применены для оптимизации параметров классификатора и отбора его признаков. Проведены эксперименты, подтверждающие эффективность разработанных алгоритмов с использованием наборов реальных данных из репозитория KEEL, наборов данных KDD Cup 1999 Data, SVC 2004 и Malicious and Benign Websites.

Разработанные алгоритмы позволили уменьшить ошибку классификации и ее вычислительную сложность, увеличить интерпретируемость классификатора. Разработанные алгоритмы показали свою эффективность при построении классификаторов для анализа сетевых атак, определении подлинности рукописной подписи и могут быть применены при создании программных средств обеспечения информационной безопасности.

Разработка программных средств производилась с использованием языка программирования Python. В качестве среды разработки был использован PyCharm. Пояснительная записка выполнена в текстовом редакторе Microsoft Word.

Abstract

Graduation qualifying work, 87 pp., 23 figures, 28 tables, 37 sources, 1 applications.

FUZZY CLASSIFIER, METAURISTIC, OPTIMIZATION ALGORITHM, CLASSIFICATION, LIGHTNING SEARCH ALGORITHM, BINARIZATION, FEATURES SELECTION.

The object of research in the work is the data classification process.

The subject of the research is the algorithm based on the metaheuristic “Lightning search algorithm”.

The purpose of the work is to improve the classification of data using the metaheuristic “Lightning search algorithm”.

To achieve the described goal, an analytical review of the current state of research was carried out. During the work continuous and binary algorithms based on the lightning search algorithm were developed. These algorithms were implemented as software. The use of metaheuristics in the tasks of minimizing functions is proposed. Algorithms were used to optimize parameters of classifier and to select features. Experiments, that confirm the effectiveness of the developed algorithms, have been carried out using real data sets from the KEEL repository, KDD Cup 1999 Data, SVC 2004 and Malicious and Benign Websites datasets.

The developed algorithms have reduced the classification error and its computational complexity, increase the interpretability of the classifier. The developed algorithms have shown their effectiveness in constructing classifiers for the analysis of network attacks, determining the authenticity of handwritten signatures and can be used to create information security software.

Software development was carried out using the Python programming language. PyCharm was used as a development environment. Explanatory note wrote in a text editor Microsoft Word.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра комплексной информационной безопасности
электронно-вычислительных систем (КИБЭВС)

УТВЕРЖДАЮ
Зав. кафедрой КИБЭВС
_____ А.А. Шелупанов
«_____» _____2020 г.

ЗАДАНИЕ

на выпускную квалификационную работу
Студенту группы 725 факультет безопасности Мельниковой Наталии Евгеньевны

1 Тема работы: «Алгоритмы и программные средства построения нечетких классификаторов на основе метаэвристики «Разряд молнии». (Утверждена приказом по ВУЗу от «___» _____ 2020 г. № _____)

2 Срок сдачи студентом законченной работы: «___» _____ 2020 г.

3 Исходные данные: научно-техническая литература, описание нечеткого классификатора, язык программирования Python, наборы данных KEEL, набор данных KDD Cup 1999 Data, набор данных SVC2004, набор данных Malicious and Benign Websites.

4 Содержание расчётно-пояснительной записки (перечень подлежащих разработке вопросов):

– обзор актуальных литературных источников в области классификации, методов построения нечетких классификаторов, алгоритмов отбора признаков, алгоритмов оптимизации;

– разработка и программная реализация алгоритма «Разряд молнии» в непрерывном пространстве поиска;

– применение алгоритма «Разряд молнии» при минимизации функций;

– применение алгоритма «Разряд молнии» для оптимизации параметров нечётких классификаторов;

– бинаризация алгоритма «Разряд молнии» для решения задачи отбора информативных признаков при построении нечётких классификаторов;

– сравнение бинарного алгоритма «Разряд молнии» с аналогами при решении задачи отбора информативных признаков;

– проведение эксперимента и построение нечетких классификаторов на наборах данных KDD Cup 1999 Data, SVC2004, Malicious and Benign Websites;

– технико-экономическое обоснование проекта;

– вопросы безопасности жизнедеятельности.

5 Технические требования к отчёту по практике: оформление отчёта должно соответствовать требованиям ОС ТУСУР 01.2013.

Руководитель: профессор кафедры КИБЭВС, доктор техн. наук, И.А. Ходашинский

_____ _____

(дата) (подпись)

Консультант по экономике: старший преподаватель кафедры КИБЭВС, Глухарева Светлана Владимировна

_____ _____

(дата) (подпись)

Консультант по безопасности жизнедеятельности: канд. техн. наук, доцент кафедры КИБЭВС, Давыдова Елена Михайловна

_____ _____

(дата) (подпись)

Научный консультант: старший преподаватель кафедры КИБЭВС, Глухарева Светлана Владимировна

_____ _____

(дата) (подпись)

Задание принял к исполнению: студент группы 725, Мельникова Наталия Евгеньевна

_____ _____

(дата) (подпись)

Содержание

Компакт-диск:	В конверте
Пояснительная записка в формате PDF:	на обороте
ФБ ДР.503200.001 ПЗ	обложки

1 Введение

В последние годы отмечается возросший интерес к задачам классификации, как одной из важных задач анализа данных. Классификация применяется в различных сферах жизни. Нечеткие классификаторы являются одним из способов решения задач классификации.

Целью данной работы является усовершенствование классификации данных с помощью метаэвристического алгоритма «Разряд молнии» [1].

Для достижения цели необходимо решить следующие задачи:

- составление аналитического обзора;
- разработка и реализация алгоритма классификации;
- проведение эксперимента;
- анализ результатов.

Результаты работы могут быть использованы для классификации данных при создании программных средств обеспечения информационной безопасности.

Проведенный аналитический обзор литературы показал, что интерес к решению задачи нечеткой классификации не ослабевает. Несмотря на то, что авторам удалось получить сопоставимые или лучшие результаты по сравнению с аналогами, продолжается поиск новых алгоритмов и модификация существующих для улучшения классификации.

В данной работе будет предложено решение задачи отбор информативных признаков и задачи оптимизация параметров классификатора. Предложен способ построения нечеткого классификатора, основанного на новом метаэвристическом алгоритме «Разряд молнии».

Для решения задачи отбора признаков разработана бинаризация алгоритма «Разряд молнии». Применение алгоритмов направлено на уменьшение ошибки классификации и вычислительной сложности классификатора с сохранением высокой интерпретируемости полученных решений.

2 Обзор состояния исследования

2.1 Методы классификации

В последние годы отмечается возросший интерес к задачам классификации, как одной из важных задач анализа данных. Классификация применяется в различных сферах жизни: в науке, медицине, в экономике и банковском деле, в технических областях, в том числе бурно развивающихся информационных технологиях.

Решение задач классификации повлекло за собой развитие разнообразных методов классификации [2], таких как нейронные сети, машины опорных векторов, байесовские классификаторы, а также методы нечеткой классификации [3].

Нейронная сеть представляет собой набор нейронов, соединенных между собой. Нейрон обладает набором входов и одним выходом. В процессе функционирования нейрон осуществляет вычисление выходного значения на основе входных. Некоторые входы нейронов помечены как внешние входы нейронной сети, а некоторые выходы нейронов как внешние выходы. Нейронная сеть производит вычисление значения выходов на основе входов. В рамках классификации на вход подаются признаки объекта, на выходе выдается вектор принадлежности к классам. Недостатками нейронных сетей являются потребность в большом количестве циклов настройки внутренних параметров, сложность подготовки обучающей выборки, возможность возникновения тупиковых ситуаций при обучении, необходимость систем высокой сложности для некоторых видов задач, а также сложность интерпретации системы [4].

Метод опорных векторов заключается в переводе исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве. Недостатками данного метода являются неустойчивость по отношению к шуму в исходных данных, сложность в подборе параметров и структуры системы под конкретную предметную область [5].

Байесовский подход к классификации основан на теореме, утверждающей, что если плотности распределения каждого из классов известны, то искомый алгоритм можно выписать в явном аналитическом виде. Данный алгоритм обладает минимальной вероятностью ошибок, но на практике метод не позволяет без погрешности оценить плотность распределения классов по обучающей выборке. Другим недостатком является высокая вероятность переобучения для выборок малых размеров [6].

Отличительной особенностью метода нечеткой классификации является то, что

данная классификация не предполагает наличие жестких границ между соседними классами. Среди достоинств нечетких классификаторов можно выделить легкую интерпретируемость правил классификации, то есть их понятность пользователю.

2.2 Нечеткие классификаторы

Для построения нечеткого классификатора требуется решить ряд задач, таких как:

- отбор информативных признаков в обучающих данных;
- генерация базы нечетких правил;
- оптимизация параметров классификатора.

База нечетких правил является основой нечеткого классификатора. Она представляет собой набор ЕСЛИ-ТО правил с нечеткими антецедентами (ЕСЛИ-часть) и метками класса в консеквентах (ТО-часть). Антецедентные части правил разбивают входное пространство признаков на множество нечетких областей, а консеквенты задают выход классификатора, помечая эти области меткой класса [7]. Для генерации базы нечетких правил обычно используются алгоритмы кластеризации, которые позволяют получить начальное, «грубое» приближение нечеткого классификатора.

В данной работе будет предложено решение задачи отбора информативных признаков и задачи оптимизации параметров классификатора.

Процедура отбора значимых признаков позволяет уменьшить размерность задачи и избавиться от избыточных признаков, что увеличивает точность классификации и уменьшает вычислительную сложность.

Оптимизация параметров классификатора может значительно увеличить точность классификации за счёт «тонкой» настройки классификатора. Для оптимизации параметров антецедентов используются алгоритмы оптимизации, осуществляющие поиск в многомерном пространстве, где в роли координат многомерного пространства выступают параметры классификатора.

Построениям нечетких классификаторов были посвящены работы [8 - 17]. Авторам реализаций алгоритмов [8 - 17] удалось выявить характеристики систем, влияющие на вычислительную сложность. Исследования направлены на повышение компактности классификатора, то есть уменьшения числа правил, за счет чего увеличивается интерпретируемость. Несмотря на то, что авторам удалось получить сопоставимые или лучшие результаты по сравнению с аналогами, продолжается поиск новых алгоритмов и модификация существующих для улучшения классификации.

2.3 Методы оптимизации параметров

Методы оптимизации подразделяются на стохастические и детерминированные. Для поиска глобального оптимума стохастические методы используют элементы случайности. Детерминированные методы используют математические формулы, которые строго определяют процедуру поиска оптимума [18].

Среди стохастических методов выделяют метаэвристические методы оптимизации, которые основываются на понятии эвристики. Эвристика – это стохастическая процедура оптимизации, которая находит допустимое решение, достаточно близкое к оптимуму функции за приемлемое время [19]. Метаэвристика - это высокоуровневая проблемно-независимая алгоритмическая структура, которая предоставляет набор рекомендаций или стратегий для разработки эвристических алгоритмов оптимизации [20].

Метаэвристики представляют собой класс оптимизационных методов, позволяющий находить решения для широкого круга задач из различных приложений. Главное достоинство метаэвристических алгоритмов заключается в их способности находить решение без знания пространства поиска. Выполняя поиск, метаэвристика позволяет находить оптимальное или близкое к оптимальному решение. Упрощенно можно рассматривать метаэвристики как алгоритмы, реализующие направляемый случайный поиск возможных решений задачи пока не будет выполнено некое условие или достигнуто заданное число итераций.

Метаэристические алгоритмы оптимизации реализуют управление процессом поиска решения путем сохранения и анализа предыдущего опыта поиска. Элемент случайности в процессе поиска позволяет избегать попадания в локальные оптимумы. Абстрактный уровень описания алгоритмов дает возможность применения к широкому кругу задач. Метаэвристики широко используются при минимизации математических функций.

Метаэвристические алгоритмы можно разделить на три класса [8]: эволюционные, основанные на физических законах и роевом интеллекте. Эволюционные алгоритмы основаны на концепции эволюции в природе. Одним из самых популярных эволюционных алгоритмов является генетический алгоритм, который имитирует эволюцию в соответствии с теорией Дарвина.

Физические алгоритмы базируются на физических явлениях, таких как гравитационные и электромагнитные взаимодействия, например, алгоритм гравитационного поиска, алгоритм, использующий в основе теорию большого взрыва и другие. В этих алгоритмах агенты взаимодействуют и перемещаются в пространстве согласно законам

физики.

Алгоритмы роевого интеллекта подражают социальному поведению стад, роев и групп животных в природе. В этом типе алгоритмов поисковые агенты движутся и взаимодействуют подобно социальному поведению различных типов животных. Наиболее популярным алгоритмом разведки роя является алгоритм роящихся частиц (PSO), который основан на поведении стаи птиц. В работе [9] при построении классификатора для оптимизации использован алгоритм, основанный на поведении колонии муравьев. Авторы [10] предлагают вариант построения нечеткой системы типа Такаги-Сугено, где для оптимизации параметров системы был использован адаптированный к решению задачи метод под названием алгоритм «Стадо криля», основанный на перемещении антарктического криля в поисках пищи. В работе [11] изложен результат разработки нечеткого классификатора диабетической болезни, основанного на модифицированном алгоритме искусственной пчелиной колонии.

Метаэвристические алгоритмы находят свое применение не только при решении задач классификации, но также в любой задаче, которая может быть сведена к решению задачи оптимизации. Так, например, один из часто используемым методов - метод нечетких с-средних [21], в различных модификациях может быть использован для задач кластеризации, при построении искусственной нейронной сети, анализе изображений, структурном анализе и т.д. Поэтому обычно предлагаемые метаэвристические алгоритмы нуждаются в доработке и адаптации для решения конкретной задачи.

2.4 Методы отбора признаков

Часто наборы данных содержат большое количество признаков, при этом некоторые признаки могут быть неинформативными или зависеть от других признаков, так что их использование может снижать точность классификации, увеличивая при этом вычислительную сложность классификации. Для решения данной проблемы применяют отбор признаков.

Отбор признаков – это процедура, в ходе которой из исходного множества признаков выделяют подмножество, наиболее соответствующее решаемой задаче. В процессе отбора из набора признаков исключаются неинформативные признаки, тем самым уменьшается количество анализируемых данных и повышая эффективность классификации.

Для решения задачи отбора признаков применяют различные методы, числе которых фильтры, обертки, гибриды фильтров и оберток, а также встроенные (интегрированные)

методы [22-24].

Методы, относящиеся к фильтрам, анализируют данные и осуществляют отбор информативных признаков до применения алгоритма классификации. Преимуществом такого подхода является меньшая вычислительная сложность относительно других методов и простота применения.

Методы-обертки при отборе признаков используют информацию, полученную от методов классификации. Как следствие, методы, относящиеся к данной категории дают результаты лучше, чем методы-фильтры, так как могут находить более глубокие закономерности в данных. При этом возрастает вычислительная сложность метода и существует риск переобучения [24].

Встроенные (интегрированные) методы выполняют отбор признаков в процессе обучения и включают алгоритм отбора признаков в алгоритм построения классификатора.

2.5 Постановка задачи

Целью данной работы является усовершенствование классификации данных с помощью метаэвристического алгоритма «Разряд молнии». Для этого необходимо разработать алгоритмы и программные средства для отбора информативных признаков и оптимизации параметров с целью улучшения качества классификации для дальнейшего применения разработанных алгоритмов в задачах классификации сетевого трафика, аутентификации пользователя по рукописной подписи и определения вредоносных сайтов.

Так как нечеткий классификатор будет применяться при решении задач обеспечения информационной безопасности, то необходимо создать классификатор, обладающий низкой ошибкой классификации и низкой вычислительной сложностью. Вычислительная сложность классификатора зависит от количества используемых для классификации признаков.

Реализация данных задач будет выполнена путем построения нечётких классификаторов с оптимизацией параметров классификатора при помощи алгоритма, основанного на метаэвристике «Разряд молнии», и на наборах данных, отбор признаков которых проведен бинаризованным алгоритмом «Разряд молнии».

Таким образом, необходимо выполнить:

- обзор актуальных литературных источников в области классификации, методов построения нечетких классификаторов, алгоритмов отбора признаков, алгоритмов оптимизации;

- разработка и программная реализация алгоритма «Разряд молнии» в непрерывном

пространстве поиска;

- применить алгоритм «Разряд молнии» при минимизации тестовых математических функций;

- применить алгоритм «Разряд молнии» для оптимизации параметров нечётких классификаторов;

- бинаризовать алгоритм «Разряд молнии» для решения задачи отбора информативных признаков при построении нечётких классификаторов;

- сравнить бинарный алгоритм «Разряд молнии» с аналогами при решении задачи отбора информативных признаков;

- провести эксперимент и построить нечеткие классификаторы на наборах данных KDD Cup 1999 Data, SVC2004, Malicious and Benign Websites.

3 Описание разработанных алгоритмов

3.1 Теоретическое описание алгоритма

Метаэвристика под названием «Разряд молнии» («Lightening search algorithm») основана на природном явлении молнии и механизме распространения частицы-лидера и с использованием концепции быстрых частиц, известных как метательные снаряды. Эта метаэвристика была впервые предложена в работе [1] в 2015 году.

Молния - захватывающее и впечатляющее явление природы (рисунок 3.1). Она представляет собой электрический искровой разряд в атмосфере, обычно происходящий во время грозы, которая порождает вероятностный характер и изменчивые характеристики разрядов молнии. Для возникновения молнии необходимо, чтобы в относительно объёме облака образовалось электрическое поле с напряжённостью, достаточной для начала электрического разряда, а в значительной части облака существовало бы поле со средней напряжённостью, достаточной для поддержания начавшегося разряда.

Рисунок 3.1 – Разряд молнии

Процесс развития молнии состоит из нескольких стадий. На первой стадии в зоне, где электрическое поле достигает критического значения, начинается ударная ионизация, создаваемая вначале свободными зарядами, всегда имеющимися в небольшом количестве в воздухе, которые под действием электрического поля приобретают значительные скорости по направлению к земле и, сталкиваясь с молекулами, составляющими воздух, ионизуют их.

Запуск молнии происходит от высокоэнергетических частиц, вызывающих пробой на убегающих электронах под воздействием космического излучения или естественной

радиоактивности. Таким образом возникают электронные лавины, переходящие в нити электрических разрядов - стримеры, представляющие собой хорошо проводящие каналы, которые, сливаясь, дают начало яркому термоионизованному каналу с высокой проводимостью - ступенчатому лидеру молнии.

Движение лидера к земной поверхности происходит ступенями в несколько десятков метров со скоростью порядка 50 000 километров в секунду, после чего его движение приостанавливается на несколько десятков микросекунд, а свечение сильно ослабевает; затем в последующей стадии лидер снова продвигается на несколько десятков метров. Яркое свечение охватывает при этом все пройденные ступени; затем следуют снова остановка и ослабление свечения. Эти процессы повторяются при движении лидера до поверхности земли со средней скоростью 200 000 метров в секунду.

Предложенный алгоритм оптимизации основан на идее механизма распространения лидера шага. В нем рассматривается участие быстрых частиц, известных как метательные снаряды, в формировании бинарной древовидной структуры - траектории движения лидера шага, и одновременном формировании двух лидеров в точках разветвления вместо традиционного механизма лидера шага, который использует концепцию стримеров.

3.2 Алгоритм «Разряд молнии» в непрерывном пространстве поиска

3.2.1 Описание алгоритма

Алгоритм метаэвристики «Разряд молнии» может быть описан следующим образом:

Шаг 1. Сбросить счетчик итераций и счетчик времени канала.

Шаг 2. Сгенерировать расположение *n* частиц случайным образом.

Шаг 3. Рассчитать энергии частиц (значения фитнесс-функции, где под фитнесс-функцией понимается минимизируемая функция).

Шаг 4. Пока номер текущей итерации меньше максимального количества итераций выполнить для каждой частицы:

Шаг 4.1. Если счетчик времени канала больше максимального времени канала, перейти к шагу 4.2, иначе – к шагу 4.3.

Шаг 4.2. Удалить худшую частицу, перераспределив ее энергию к лучшей частице, сбросить счетчик времени канала. Перейти к шагу 4.3.

Шаг 4.3. Увеличить счетчик итераций и счетчик времени канала на 1. Рассчитать новую позицию частицы по формуле:

$$p'_i = p_i + normrand\ (\mu_i, \sigma_i),$$

где p'_i - новая позиция частицы;

p_i - текущая позиция частицы;

normrand - случайное число, сгенерированное из нормального распределения с параметрами μ_i и σ_i.

Рассчитать энергию в новой позиции.

Шаг 4.4. Если значение энергии в новой позиции лучше, чем в текущей, то перейти к шагу 4.5, иначе – к шагу 4.

Шаг 4.5. Если необходимо выполнить расщепление канала, то перейти к шагу 4.6, иначе – к шагу 4.7.

Шаг 4.6. Выполнить расщепление канала, найдя симметричную позицию частицы по формуле:

$$p^*_i = a + b - p_i,$$

где p^*_i - симметричная позиция частицы;

p_i - новая позиция частицы;

a и b - нижняя и верхняя границы поиска соответственно.

Рассчитать энергию в симметричной позиции. Если значение энергии в симметричной позиции лучше, чем в новой, заменить новую позицию частицы на симметричную.

Шаг 4.7. Заменить текущую позицию частицы значением новой позиции.

Шаг 5. Вернуть позицию с наилучшим значением энергии.

3.2.2 Пример работы алгоритма

Схематичное представление алгоритма приведено на рисунках 3.2 – 3.4. В начале работы алгоритма имеются 4 частицы.

В ходе первой итерации (рисунок 3.2) для каждой частицы выбирается новая позиция, траектория движения к которой показана пунктирной линией. В случае, если значение энергии в новой позиции лучше, частица выполняет переход (например, желтая и зеленая частицы), в противном случае частица сохраняет своё положение (например, синяя и красная частицы).

Во время второй итерации (рисунок 3.3) для каждой частицы опять выбирается новая позиция и выполняется переход в нее в случае лучшего значения энергий (синяя и красная частицы). Для зеленой частицы было выполнено расщепление канала.

На третьей итерации (рисунок 3.4) было выполнено уничтожение канала частицы с худшим значением энергии (синяя частица). Для других частиц был выполнен и поиск нового положения, траектория к которому показана пунктиром.

Рисунок 3.2 – Первая итерация алгоритма

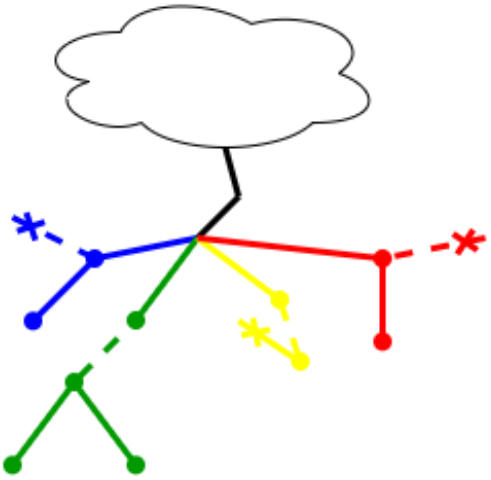

Рисунок 3.3 – Вторая итерация алгоритма

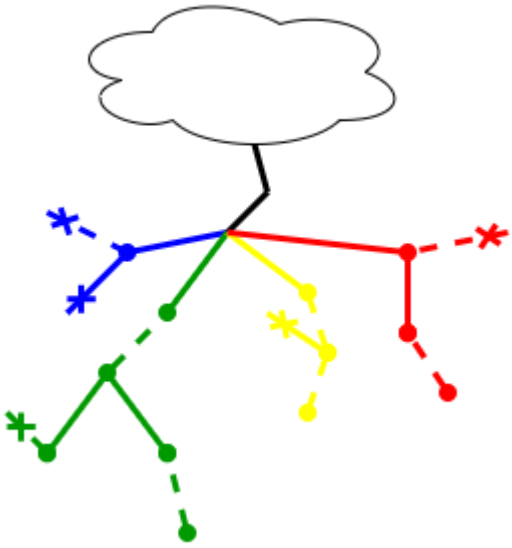

Рисунок 3.4 – Третья итерация алгоритма

3.2.3 Реализация алгоритма

Приведенный алгоритм был реализован на языке программирования Python.

Алгоритм обладает следующими параметрами:

n – количество частиц в популяции;

I – количество итераций;

max_channel_time – максимальное время канала, то есть количество проходов, через которое худший канал будет отброшен, а энергия перераспределена к лидеру;

fork_probability – вероятность расщепления канала;

μ – параметр нормального распределения (математическое ожидание);

σ – начальное значение параметра нормального распределения (среднеквадратическое отклонение).

Псевдокод реализации алгоритма приведен далее:

Вход: n,lb,ub,I,max_channel_time, fork_probability, μ, σ

Выход: p_{best}.

Popul := {p_1, p_2, …, p_n};

channel_time := 0

цикл пока (I > 0)

    цикл по i от 1 до n

        если (channel_time > max_channel_time) то

            Eliminate_whorst()

        Fork_best()

        channel_time := 0

    channel_time := channel_time + 1

    p_{new} = Popul[i] + $\mathrm{normrand}(\mu_i, \sigma_i)$

    если (F(p_{new}) < F(Popul[i])) то

        probability := rand(0, 1);

        если (probability > fork_probability) то

            Fork()

        иначе

            Popul[i] = p_{new}

    конец цикла

    p_{best} = Seach_min()

    I := I - 1;

конец цикла

вывод $p_{bestм}$

3.3 Бинарный алгоритм «Разряд молнии»

3.3.1 Описание алгоритма

Бинарный алгоритм «Разряд молнии» представляет собой модификацию алгоритма в непрерывном пространстве поиска.

Основная разница между бинарной и непрерывной версиями алгоритмов заключается в изменении позиций частиц. В бинарном варианте позиция каждой частицы представляет собой бинарный вектор, то есть вектор в котором каждый элемент представляет собой 0 или 1.

Инициализация начальных положений частиц производится аналогично непрерывному алгоритму. Далее, чтобы обновить положение каждой частицы применяется подходящая вероятностная функция, цель которой – выбрать между 0 и 1, поэтому вводится трансформационная функция *T*. Трансформационная функции отвечает за преобразование непрерывного пространства в дискретное пространство поиска при бинаризации метаэвристик. В то же время используется функция для обновления положений частиц, такая, чтобы уменьшить возможность изменения позиции частицы, если $|p_i|$ маленькое и увеличить возможность изменения позиции частицы при большом значении $|p_i|$.

На выходе алгоритм дает бинарный вектор, соответствующий отобранным признакам набора данных: 1 означает, что признак принимается значимым и учитывается в процессе классификации; 0 показывает, что признак отбрасывается.

В случае, если полученный на каком-либо шаге вектор полностью состоит из 0, в качестве позиции частицы на случайное место ставится 1, чтобы исключить ситуацию, когда ни один признак не был отобран.

Шаг с расщеплением на два симметричных канала был удален из исходного алгоритма, так в непрерывном пространстве выбор симметричного канала давал возможность улучшить ситуацию в плохих случаях, что не дает подобных преимуществ в пространстве поиска, ограниченном 0 и 1. Инверсия вектора также не дала улучшения точности.

В результате бинаризации метаэвристики «Разряд молнии» был составлен следующий алгоритм:

Шаг 1. Сбросить счетчик итераций и счетчик времени канала.

Шаг 2. Сгенерировать расположение n частиц случайным образом, расположение каждой из частиц – бинарный вектор.

Шаг 3. Рассчитать энергии частиц, как значения фитнесс-функции. В качестве фитнесс-функции берется минимизируемая функция. При решении задачи отбора признаков фитнесс-функция представляет собой ошибку классификации.

Шаг 4. Пока номер текущей итерации меньше максимального количества итераций выполнить для каждой частицы:

Шаг 4.1. Если счетчик времени канала больше максимального времени канала, перейти к шагу 4.2, иначе – к шагу 4.3.

Шаг 4.2. Удалить худшую частицу, перераспределив ее энергию к лучшей частице, сбросить счетчик времени канала. Перейти к шагу 4.3.

Шаг 4.3. Увеличить счетчик итераций и счетчик времени канала на 1. Рассчитать новую позицию частицы по формуле:

$$p_i' = \begin{cases} \overline{p_i}, & T(p_i + normrand\ (\mu_i, \sigma_i)) < rand; \\ p_i, & \text{в противном случае} \end{cases}$$

$$T(p_i) = |\ th\ (p_i)\ |,$$

где p'_{ij} – новая позиция частицы;

p_i – текущая позиция частицы;

T – трансформационная функция;

normrand – случайное число, сгенерированное из нормального распределения с параметрами μ_i и σ_i;

rand – случайное число из отрезка [0; 1].

Рассчитать энергию в новой позиции.

Шаг 4.4. Если значение энергии в новой позиции лучше, чем в текущей, то заменить текущую позицию частицы значением новой позиции.

Шаг 5. Вернуть позицию с наилучшим значением энергии.

3.3.2 Реализация алгоритма

Приведенный алгоритм был реализован на языке программирования Python.

Алгоритм обладает следующими параметрами:

n – количество частиц в популяции;

D – размерность задачи, соответствующая общему количеству признаков;

I - количество итераций;

max_channel_time – максимальное время канала, то есть количество проходов, через

которое худший канал будет отброшен, а энергия перераспределена к лидеру;

μ- параметр нормального распределения (математическое ожидание);

σ - начальное значение параметра нормального распределения (среднеквадратическое отклонение).

Псевдокод реализации алгоритма приведен далее:

Вход: n,D,I,max_channel_time, μ, σ

Выход: p_{best}.

Popul := {p_1, p_2, …, p_n};

channel_time := 0

цикл пока (I > 0)

цикл по i от 1 до n

если (channel_time > max_channel_time) то

Eliminate_whorst()

Fork_best()

channel_time := 0

channel_time := channel_time + 1

цикл по d от 1 до D

если T(Popul[i] [d] + normrand(μ_i, σ_i)) < rand([0, 1]) то

p_{new} [d] = 1

иначе

p_{new} [d] = 0

конец цикла

if_zero_vector := true

цикл по d от 1 до D

если (p_{new} [d] = 1) то

if_zero_vector := false

конец цикла

если (if_zero_vector) то

index := rand([1, D]

p_{new} [index] := 1

если (F(p_{new}) < F(Popul[i])) то

Popul[i] = p_{new}

конец цикла

p_{best} = Seach_min()

I := I - 1;

конец цикла

вывод p_{best}

3.4 «Жадный» алгоритм отбора признаков

«Жадный» алгоритм отбора признаков позволяет найти решение близкое к оптимальному. Алгоритм строится на идее, что если присутствие признака в наборе увеличивает точность классификации, то признак должен присутствовать в итоговом наборе отобранных признаков, так как признак считается значимым. Данное допущение не позволяет найти оптимальное решение, однако достоинством данного алгоритма является высокая скорость его работы.

На начальном этапе работы алгоритма создается бинарный вектор, заполненный нулями. Для каждой позиции в векторе поочередно 1 ставится на каждую позицию и рассчитывается значение фитнесс-функции для полученного вектора. Таким образом находится вектор, соответствующий лучшему значению фитнесс-функции на данном этапе. Аналогичным образом в найденном векторе ставится вторая единица. Если решения лучше, чем на предыдущем шаге, найдено не было, то алгоритм завершает работу, иначе действия повторяются до тех пор, пока решение не начнет ухудшаться или весь вектор не будет заполнен единицами.

На выходе алгоритм дает бинарный вектор, соответствующий отобранным в наборе данных признакам.

Алгоритм был реализован на языке программирования Python.

Алгоритм обладает параметром D – размерность задачи, соответствующая общему количеству признаков.

Псевдокод реализации алгоритма:

Вход: D

Выход: best_vector.

best_vector := {0, 0, …, 0};

best_error := 1

d := 0

цикл пока (d < D)

vector := best_vector

цикл по i от 1 до D

если (vector[i] ≠ 1) то

vector[i] := 1

error := Function(vector)

если (error < best_error) то

best_error := new_error

best_vector := new_ vector

конец цикла

d := d + 1;

конец цикла

вывод best_vector

3.5 Отбор признаков методом полного перебора

Метод полного перебора, как следует из названия, представляет собой полный перебор всех сочетаний 1 и 0 в бинарном векторе с расчётом значений фитнесс-функции для каждого сочетания. Данный алгоритм гарантированно позволяет найти оптимальное решение. Существенным недостатком алгоритма является большое количество вычислений фитнесс-функции, возрастающее с увеличением количества признаков в наборе данных. Количество вычислений фитнесс-функции определяется формулой:

$$2^n,$$

где n – количество признаков в наборе данных.

Как следствие, применение данного алгоритма требует больших вычислительных мощностей и долгого времени, что не позволяет его использовать на практике.

Параметр алгоритма D – размерность задачи, соответствующая общему количеству признаков.

Псевдокод реализации алгоритма:

Вход: D

Выход: best_vector.

best_vector := {0, 0, …, 0};

best_error := 1

Функция Change_vector

Вход: vector, index, value

Выход: best_vector, best_error.

Если (index < 0) то

возврат

vector1 := vector

vector1[index] = val

error = Function(vector1)

Если (error < best_error) то

best_error = error

best_vector = vector1

index := index – 1

Change_vector(vector, index, 1)

Change_vector(vector1, index, 1)

Change_vector(best_vector, D-1, 1)

Вывод best_vector

3.6 Алгоритм отбора признаков «Случайный поиск»

Алгоритм случайного поиска может выступать в качестве контрольного алгоритма и поэтому используется для сравнения с ним других алгоритмов.

Так как решается задача отбора признаков, то используется бинарная версия алгоритма случайного поиска. Каждое новое решение представляет собой бинарный вектор. На вход подается количество итераций.

На каждой итерации генерируется новое решение, представляющее собой бинарный вектор, состоящий из случайного числа нулей и единиц. Рассчитывается значение фитнесс-функции от сгенерированного вектора. Если полученное значение лучше, чем прошлое, новое значение запоминается. Чтобы исключить векторы, состоящие полностью из нулей, в алгоритм была добавлена соответствующая проверка, которая позволяет пропускать такие решения.

Алгоритм был реализован на языке программирования Python. Алгоритм обладает следующими параметрами:

N – количество частиц в популяции;

I – количество итераций;

D – размерность задачи, соответствующая общему количеству признаков.

Псевдокод реализации алгоритма:

```
Вход: N, I, D.
Выход: best_vector.
best_vector := {1, 1, …, 1};
best_error := Function(best_vector)
цикл пока (I > 0)
    цикл пока (N > 0)
        if_zero_vector := true
        цикл по i от 1 до D
            new_vector[i] := random([1, 0])
            если ( new_vector[i] = 1) то
                if_zero_vector := false
        если ( НЕ if_zero_vector) то
            если (error < best_error) то
                best_error = error
                best_vector = vector
        N := N - 1;
    конец цикла
    I := I - 1;
конец цикла
вывод best_vector
```

4 Минимизация функций

4.1 Описание эксперимента

Поскольку область применения метаэвристик намного шире, чем задачи классификации, было проведено сравнение разработанной реализации метаэвристического алгоритма «Разряд молнии» с другими метаэвристиками при минимизации тестовых математических функций.

Параметры алгоритмов, использовавшихся при проведении эксперимента, приведены в таблице 4.1.

Таблица 4.1 – Параметры алгоритмов

Алгоритм	Размер популяции	Количество итераций	Параметры
Разряд молнии	50	500	Максимальное время канала =5 Вероятность расщепления = 0,5 $\mu = 0$ σ = верхняя граница поиска /5
Случайный поиск	50	500	Нет
Искусственные пчелы	50	500	Количество пчел, отправляемых на лучшие участки = 5 Количество пчел, отправляемых на другие выбранные участки = 5 Количество лучших участков = 5 Количество выбранных участков = 5 Размер области для участка = 10
Рой летучих мышей	50	500	Уровень импульсной эмиссии = 0.9 Громкость звука = 0.5 Минимальная частота волны = 0 Максимальная частота волны = 0.02 Постоянная для изменения громкости звука = 0.9 Постоянная для изменения уровня импульсной эмиссии = 0.9

Продолжение таблицы 4.1

Алгоритм	Размер популяции	Количество итераций	Параметры
Стая ласточек	50	500	количество локальных лидеров = 3 количество бесцельных частиц = 6

Фитнесс-функции, для которых проводилось тестирование, приведены в таблице 4.2. Размерность пространства поиска была принята равной 5.

Таблица 4.2 – Параметры функций

Функция	Название	Формула	Пространство поиска	Минимум
F1	Sphere	$f(x)=\sum_{i=1}^{n} x_i^2$	$[-100, 100]^n$	0
F2	Schwefel 1.2	$f(x)=\sum_{i=1}^{n}(\sum_{j=1}^{i} x_j^2)$	$[-100, 100]^n$	0
F3	Schwefel 2.21	$f(x)=max_i\{\|x_i\|, 1 < i < n\}$	$[-100, 100]^n$	0
F4	Schwefel 2.22	$f(x)=\sum_{i=1}^{n}\|x_i\| + \prod_{i=1}^{n}\|x_i\|$	$[-10, 10]^n$	0
F5	Step	$f(x)=\sum_{i=1}^{n}(x_i + 0{,}5)^2$	$[-100, 100]^n$	0
F6	Sum of different power	$f(x)=\sum_{i=1}^{n}(\|x_i\|)^{i+1}$	$[-1, 1]^n$	0
F7	Функция Захарова	$f(x) = \sum_{i=1}^{n} x_i^2 + +(\sum_{i=1}^{n} 0{,}5ix_i)^2 + +(\sum_{i=1}^{n} 0{,}5ix_i)^4$	$[-5, 10]^n$	0
F8	Sum squares	$f(x) = \sum_{i=1}^{n} ix_i^2$	$[-5.12, 5.12]^n$	0
F9	Rastrigin	$f(x) = 10n + \sum_{i=1}^{n} [x_i^2 - -10\cos(2\pi x_i)]$	$[-5.12, 5.12]^n$	0
F10	Ackley	$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - -e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)}$	$[-32, 32]^n$	0
F11	Griewank	$f(x) = \sum_{i=1}^{n} \frac{x_i^2}{4000} - -\prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-600, 600]^n$	0

Продолжение таблицы 4.2

Функция	Название	Формула	Пространство поиска	Минимум
F12	Periodic	$f(x,y) = 1 + +(\sin x)^2(\sin y)^2 - -0.1(e^{-x^2y^2})$	$[-10, 10]^n$	0.9
F13	Styblinski-Tang	$f(x) = \frac{1}{2}\sum_{i=1}^{n}(x_i^4 - 16\,x_i^2 + +5x_i)$	$[-5, 5]^n$	-78,3323
F14	Matyas	$f(x,y) = 0{,}26(x^2 + y^2) - -0{,}48xy$	$[-10, 10]^n$	0
F15	Beale	$f(x,y) = (1{,}5 - x + xy)^2 + + (2{,}25 - x + xy^2)^2 + (1{,}5 - -x + xy)^2 + (2{,}625 - x + +xy^3)^2$	$[-4.5, 4.5]^n$	0

4.2 Результаты эксперимента

Для каждой функции было выполнено 10 прогонов, по результатам которых были найдены лучшее, худшее и среднее значения, а также среднее квадратическое отклонение (таблица 4.3).

Таблица 4.3 - Сравнение метаэвристики «Разряд молнии» с аналогами

Функция	Значение	Разряд молнии	Случайный поиск	Искусственные пчелы	Рой летучих мышей	Стая ласточек
F1	Лучшее	2.80226E-152	414.97908	2.40648E-152	0.00439	3.87224E-09
	Худшее	3.10967E-151	1447.71920	2.58479E-151	0.01556	0.51553
	Среднее	1.04621E-151	849.03020	1.20434E-151	0.00759	0.08471
	Ср.кв.от.	9.26247E-152	320.55118	8.23026E-152	0.00305	0.15674
F2	Лучшее	2.3949E-08	236.81302	1.99340E-07	0.00323	0.12137
	Худшее	167.50559	968.59807	0.00067	80.49435	490.18030
	Среднее	26.45534	570.82497	0.00012	15.47449	114.77438
	Ср.кв.от.	50.81198	247.58035	0.00023	28.98015	147.68620

Продолжение таблицы 4.3

Функция	Значение	Разряд молнии	Случайный поиск	Искусственные пчелы	Рой летучих мышей	Стая ласточек
F3	Лучшее	1.54085E-76	10.46838	0.00086	0.03617	0.08092
	Худшее	0.00042	24.86166	0.00575	10.57753	7.72190
	Среднее	4.23815E-05	17.51543	0.00233	1.99271	2.49458
	Ср.кв.от.	0.000127	4.54490	0.00160	3.50174	2.02855
F4	Лучшее	5.57294E-46	4.9943	0.00013	0.12700	0.00046
	Худшее	1.68648	8.40989	0.00406	0.21340	0.25411
	Среднее	0.27344	6.51689	0.00157	0.16896	0.06039
	Ср.кв.от.	0.56521	1.28111	0.00125	0.028952	0.08009
F5	Лучшее	0	503.0	0	9.0	0
	Худшее	0	2498.0	0	310.0	0
	Среднее	0	1449.0	0	68.5	0
	Ср.кв.от.	0	589.93423	0	84.79887	0
F6	Лучшее	1.0456E-07	0.00486	5.06558E-10	0.00016	1.6066E-10
	Худшее	1.01743E-05	0.031275	1.94748E-07	0.00069	2.11104E-05
	Среднее	3.42129E-06	0.015131	5.70579E-08	0.00040	3.054373E-06
	Ср.кв.от.	3.39298E-06	0.00971	6.02921E-08	0.00015	6.21478E-06
F7	Лучшее	3.69493E-130	2.72358	4.51058E-10	0.00668	0.00061
	Худшее	0.05329	28.25584	1.01051E-06	0.02649	1.70048
	Среднее	0.01012	13.03581	2.96055E-07	0.01465	0.33901
	Ср.кв.от.	0.02028	9.26288	3.50398E-07	0.00732	0.51368
F8	Лучшее	1.14682E-153	3.31066	1.17494E-10	0.01051	2.19811E-08
	Худшее	9.17702E-39	16.23975	5.74230E-08	0.05114	0.22923
	Среднее	9.17702E-40	10.47277	9.17719E-09	0.03338	0.03783
	Ср.кв.от.	2.75311E-39	3.70218	1.67194E-08	0.01431	0.07274
F9	Лучшее	1.98991	15.67849	2.98487	3.62861	1.19165
	Худшее	11.93947	37.40361	12.93445	28.22722	13.92939
	Среднее	8.05915	28.81368	7.16369	13.53770	5.29543
	Ср.кв.от.	2.96657	6.85864	3.52613	8.01839	3.33564

Продолжение таблицы 4.3

Функция	Значение	Разряд молнии	Случайный поиск	Искусственные пчелы	Рой летучих мышей	Стая ласточек
F10	Лучшее	0.0	10.37370	4.01033E-05	1.70239	0.00062
	Худшее	3.55271E-15	16.71616	0.00117	9.18484	2.87380
	Среднее	2.48689E-15	13.79228	0.00052	4.42415	1.02772
	Ср.кв.от.	1.62805E-15	1.75899	0.00040	2.51956	1.20347
F11	Лучшее	0.04926	4.82901	0.14287	0.47445	0.04947
	Худшее	0.66732	18.83099	0.60622	2.11248	0.21191
	Среднее	0.28133	10.67509	0.31285	1.07886	0.13369
	Ср.кв.от.	0.19445	3.68493	0.13196	0.53214	0.05232
F12	Лучшее	0.9	0.90177	0.9	0.90001	0.9
	Худшее	1.0	0.96284	0.9	1.00011	1.0
	Среднее	0.94	0.92772	0.9	0.98004	0.91
	Ср.кв.от.	0.04898	0.02057	0	0.04001	0.02999
F13	Лучшее	-78.33233	-78.32925	-78.33233	-78.33211	-78.33233
	Худшее	-78.33233	-78.19234	-78.33233	-78.32805	-78.33233
	Среднее	-78.33233	-78.26974	-78.33233	-78.33035	-78.33233
	Ср.кв.от.	1.27105E-14	0.04026	8.98773E-15	0.001408	1.42108E-14
F14	Лучшее	1.00096E-157	0.00022	2.21724E-199	2.02115E-07	8.96916E-25
	Худшее	1.79880E-156	0.00212	6.795402E-26	3.06057E-05	0.01085
	Среднее	7.79889E-157	0.00091	6.79540E-27	7.39812E-06	0.00109
	Ср.кв.от.	6.12105E-157	0.00054	2.03862E-26	8.90959E-06	0.00325
F15	Лучшее	0	0.00026	0	5.14561E-06	5.06334E-19
	Худшее	0	0.01240	0	0.76320	0.00011
	Среднее	0	0.00525	0	0.15279	1.12474E-05
	Ср.кв.от.	0	0.00351	0	0.30508	3.23144E-05

Значения, полученные при помощи алгоритма «Разряд молнии» и при помощи других метаэвристик, значимо не отличаются. Сравнимые значения свидетельствуют о работоспособности алгоритма и возможности его применения для минимизации функций. На некоторых функциях разработанный алгоритм дал результат лучше аналогов.

Проведен статистический анализ полученных результатов минимизации функций. Критерий Фридмана позволяет проверить гипотезу H_0 о равенстве средних с минимальными требованиями к выборочным данным: предполагается, что ошибки наблюдений независимы и имеют непрерывное распределение. При проверке гипотезы H_0 средние значения найденных минимумов функций заменяются их рангами. Сравнение проведено в одних шкалах оптимумов, то есть для функций F12 и F13 их значения были нормированы. Уровень значимости был выбран равным 0,05.

Значение статистики получено при помощи пакета статистического анализа SPSS Statistics и приведено на рисунке 4.1 вместе со средними рангами. Гипотеза H_0 отвергается, то есть средние на разных уровнях отличаются значимо. Иными словами, алгоритм минимизации влияет на результат. Поскольку оценивались средние значения полученных минимумов функций, то меньшее значение ранга соответствует лучшему алгоритму минимизации. Алгоритм «Разряд молнии» занимает второе место в ранжированном ряду.

Критерий Фридмана

Ряды

	Средний ранг
Разряд_молнии	2,09
Случайный_поиск	5,00
Искусственные_пчёлы	1,73
Рой_летучих_мышей	3,36
Стая_ласточек	2,82

Статистические критерии[a]

N	11
Хи-квадрат	29,630
ст.св.	4
Асимптотическая значимость	,000

a. Критерий Фридмана

Рисунок 4.1 – Критерий Фридмана для средних значений найденных минимумов функций

Графики сходимости функций представлены на рисунках 4.2 - 4.9.

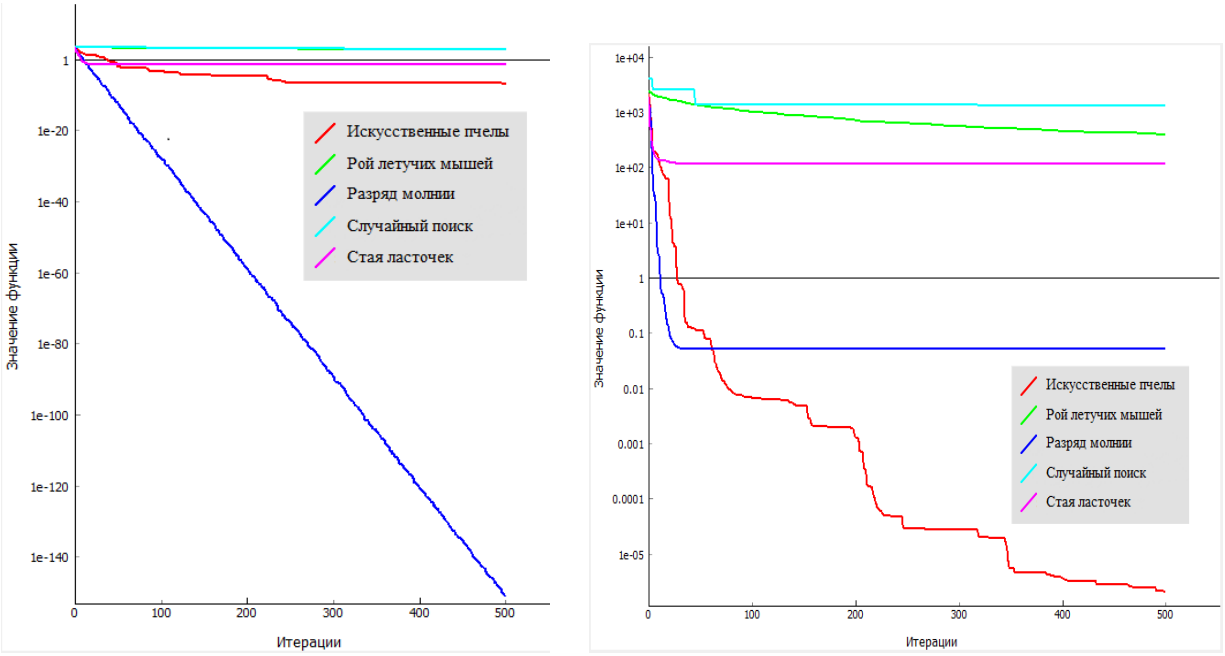

Рисунок 4.2 – Графики сходимости функций F1 (слева) и F2 (справа)

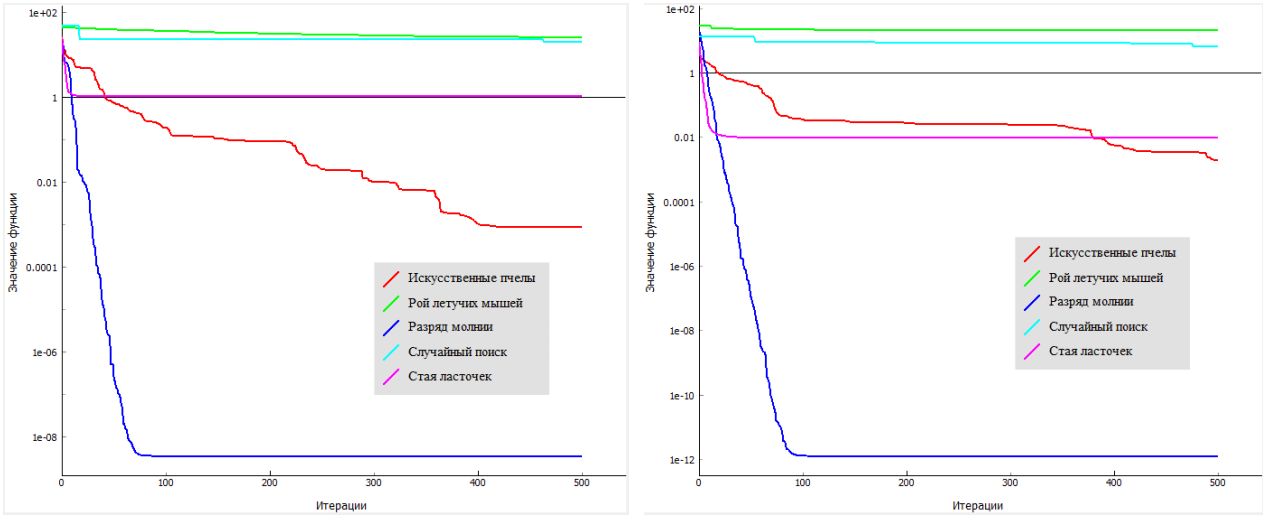

Рисунок 4.3 – Графики сходимости функций F3 (слева) и F4 (справа)

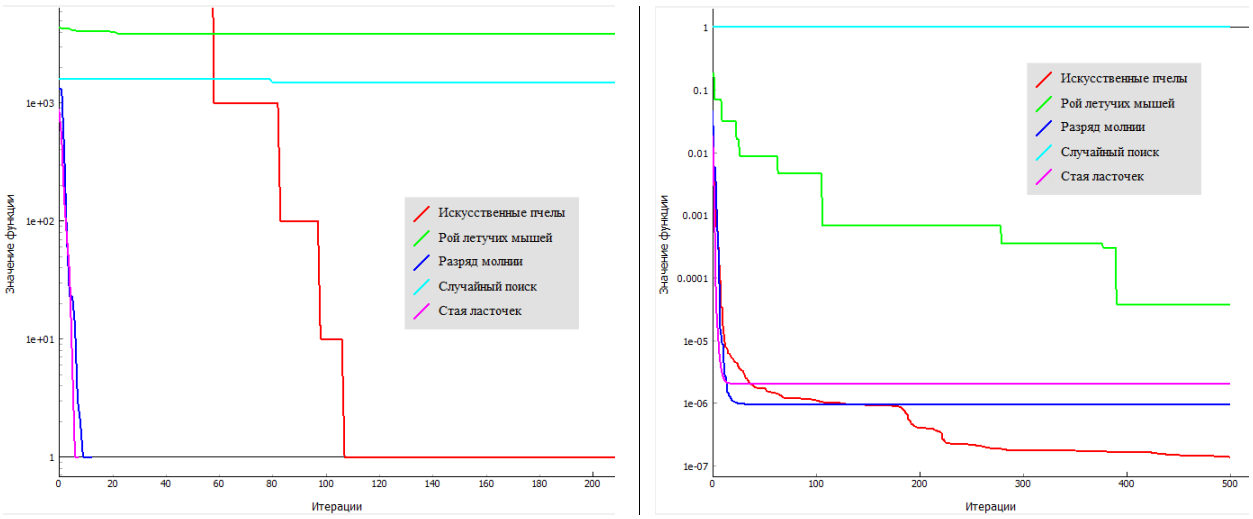

Рисунок 4.4 – Графики сходимости функций F5 (слева) и F6 (справа)

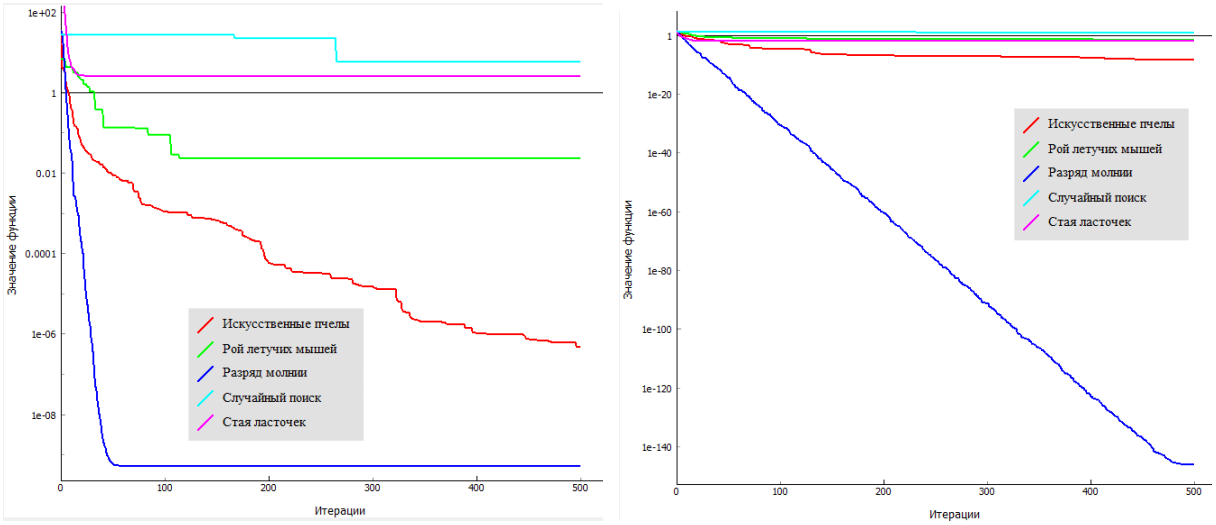

Рисунок 4.5 – Графики сходимости функций F7 (слева) и F8 (справа)

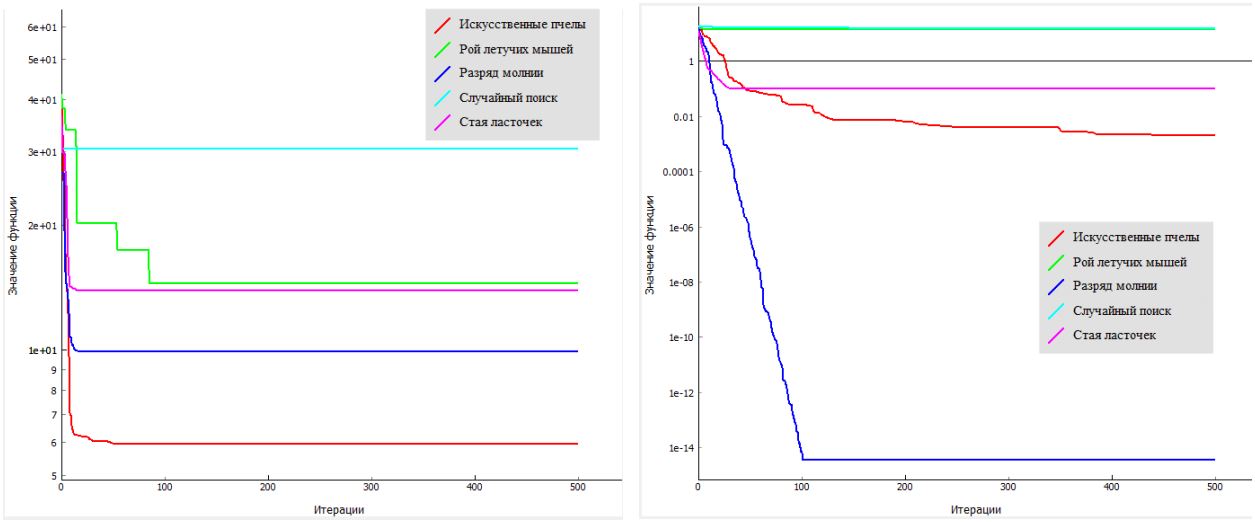

Рисунок 4.6 – Графики сходимости функций F9 (слева) и F10 (справа)

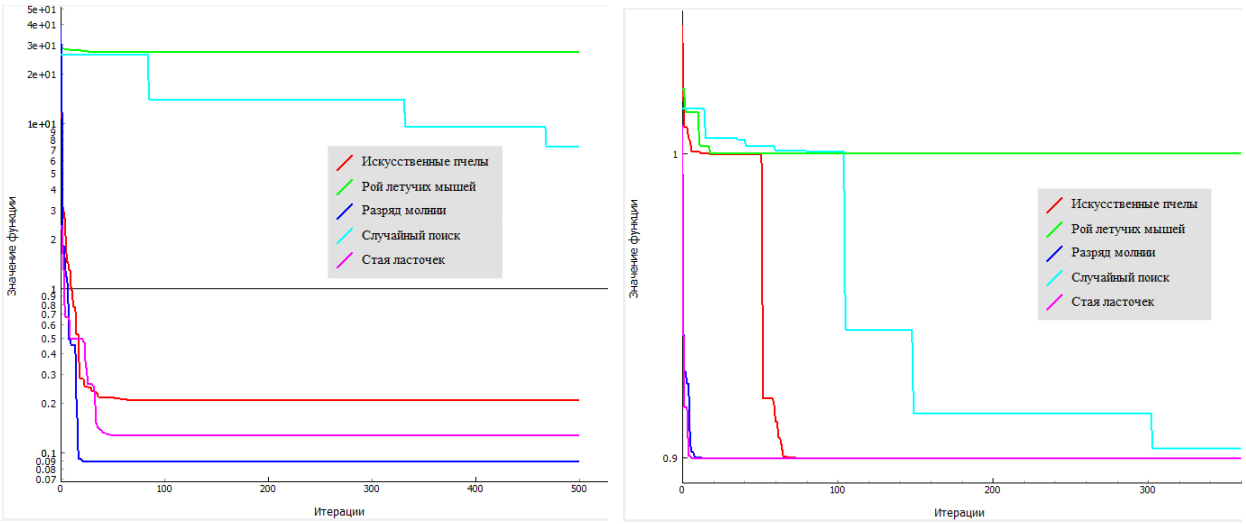

Рисунок 4.7 – Графики сходимости функций F11 (слева) и F12 (справа)

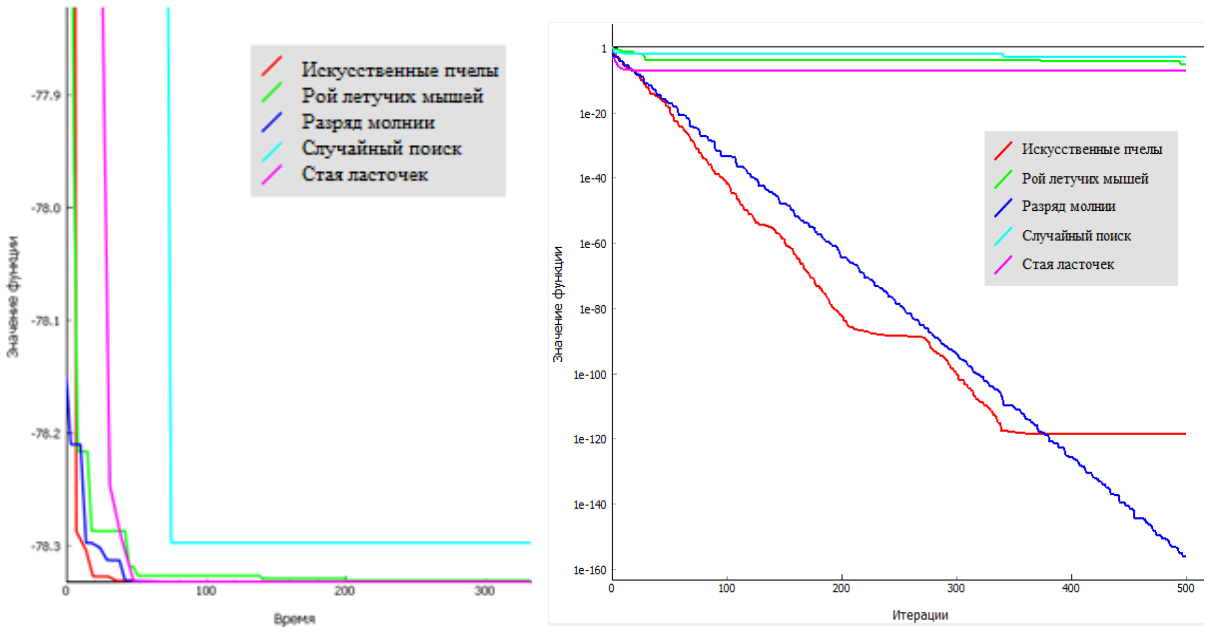

Рисунок 4.8 – Графики сходимости функций F13 (слева) и F14 (справа)

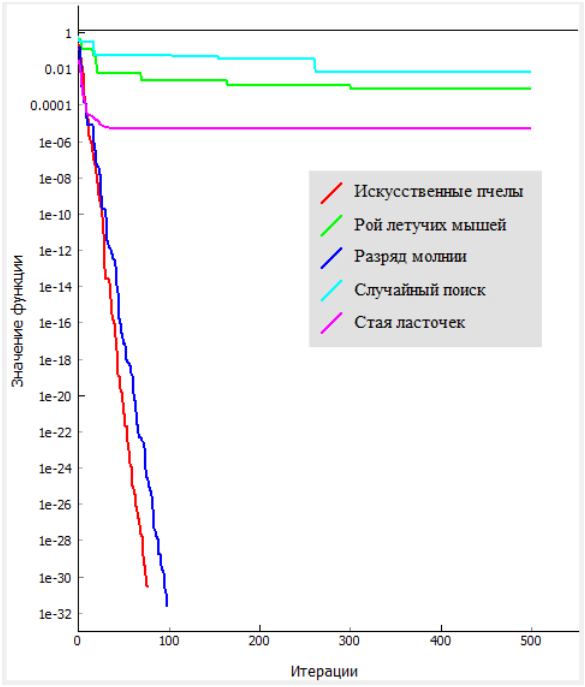

Рисунок 4.9 – График сходимости функции F15

Графики сходимости функций показывают быструю сходимость алгоритма, значительно лучшую, чем случайный поиск, и сравнимую с другими алгоритмами.

5 Оптимизация параметров классификатора

5.1 Описание эксперимента

Разработанная метаэвристика «Разряд молнии» была применена при построении нечеткого классификатора для оптимизации его параметров.

Эксперимент по оценке эффективности исследуемого метода построения нечетких классификаторов был проведен на известных наборах данных из репозитория KEEL [25]. Было использовано 14 наборов данных с различными количествами признаков, классов и экземпляров. Описание наборов данных приведено в таблице 5.1.

Таблица 5.1 – Результаты эксперимента при оптимизации параметров классификатора

Название	Признаки	Классы	Экземпляры
haberman	3	2	306
iris	4	3	150
newthyroid	5	3	215
glass	9	7	214
wisconsin	9	2	683
titanic	3	2	2201
magic	10	2	19020
wine	13	3	178
cleveland	13	5	297
heart	13	2	270
segment	19	7	2310
ring	20	2	7400
twonorm	20	2	7400
spambase	57	2	4597

В ходе эксперимента для различных наборов данных была вычислена ошибка классификатора без оптимизации и с применением различных метаэвристических алгоритмов, в том числе «Разряд молнии». В каждом алгоритме была использована популяция из 40 частиц и выполнено 300 итераций алгоритма.

Эксперимент проходил по схеме десятикратной кросс-валидации: полный набор данных делился на 10 частей, 9 из которых использовалась для обучения классификатора,

одна – для теста.

5.2 Результаты эксперимента

Результаты эксперимента для обучающих и тестовых выборок приведены в таблицах 5.2 и 5.3 соответственно, полученная ошибка классификации указана в процентах.

Таблица 5.2 – Результаты эксперимента при оптимизации параметров классификатора, обучающая выборка

Набор данных	До оптимизации	Стая птиц	Рой летучих мышей	Стая кошек	Гравитационный поиск	Серые волки	Роящиеся частицы	Стая ласточек	Разряд молнии	Кукушкин поиск	Искусственные пчелы
cleveland	55,33	34,78	41,59	36,84	34,32	33,06	39,81	35,54	37,14	37,84	39,49
glass	49,94	24,78	45,12	26,33	24,97	28,75	31,15	25,86	27,67	37,24	32,87
haberman	46,19	20,53	22,93	21,24	21,75	19,54	22,54	20,93	20,25	21,65	22,79
heart	32,67	12,8	19,74	17,43	14,72	11,69	17,94	12,81	14,86	14,22	17,15
iris	5,56	1,19	3,26	0,96	1,14	1,36	2,2	0,96	0,79	19,67	20,1
magic	43,05	18,99	23,73	16,61	17,5	16,55	20,07	19,61	17,1	2,27	1,07
newthyroid	4,19	0,41	3,74	0,31	0,36	0,67	1,21	0,31	0,36	2,27	1,07
ring	50,45	6,19	22,84	5,52	4,36	10,61	12,82	9,42	7,78	17,43	11,54
segment	19,74	6,92	19,74	12,76	6,87	19,47	10,76	6,98	13,88	19,74	12,13
spambase	60,58	10,7	23,35	60,58	60,58	36,21	60,58	17	59,92	15,4	20,23
wine	11,74	0,12	6,85	0,19	0,29	0,17	1,52	0,25	0,29	2,68	0,87

Продолжение таблицы 5.2

Набор данных	До оптимизации	Стая птиц	Рой летучих мышей	Стая кошек	Гравитационный поиск	Серые волки	Роящиеся частицы	Стая ласточек	Разряд молнии	Кукушкин поиск	Искусственные пчелы
titanic	32,3	21,84	22,42	25,65	22,19	22,12	23,21	21,81	21,68	21,83	22,17
twonorm	3,96	2,6	3,96	2,54	1,96	3,96	2,94	2,75	2,7	3,95	3,04
wisconsin	12,25	2,55	4,83	3,7	2,29	2,06	3,74	2,37	2,99	2,72	3,71

Таблица 5.3 – Результаты эксперимента при оптимизации параметров классификатора, тестовая выборка

Набор данных	До оптимизации	Стая птиц	Рой летучих мышей	Стая кошек	Гравитационный поиск	Серые волки	Роящиеся частицы	Стая ласточек	Разряд молнии	Кукушкин поиск	Искусственные пчелы
cleveland	57,17	43,45	43,64	47,4	45,98	44,28	47,44	45,76	44,41	42,17	46,44
glass	50,54	35,93	49,03	38,94	36,5	36,52	37,89	35,83	37,92	41,3	39,41
haberman	45,73	28,46	27,6	29,52	27,15	28,58	27,12	27,38	27,83	27,28	26,91
heart	32,96	21,98	23,58	29,75	26,17	19,14	25,93	20,49	21,23	19,38	23,95
iris	5,33	5,11	6,89	6,67	4,22	5,33	4,44	5,11	5,78	19,73	20,42
magic	43,12	19,48	23,79	16,93	17,81	17,1	20,49	20,02	17,64	4,46	3,55
segment	19,52	8,92	19,52	14,46	9	19,18	12,09	8,72	15,32	19,52	13,35

Продолжение таблицы 5.3

Набор данных	До оптимизации	Стая птиц	Рой летучих мышей	Стая кошек	Гравитационный поиск	Серые волки	Роящиеся частицы	Стая ласточек	Разряд молнии	Кукушкин поиск	Искусственные пчелы
newthyroid	4,59	3,08	4,13	4,3	4,32	4,31	5,1	3,09	4,79	4,46	3,55
ring	50,47	7,59	23,28	6,23	5,92	11,7	13,95	10,58	9,13	17,65	12,42
spambase	60,58	10,88	23,58	60,58	60,58	36,74	60,58	17,43	60,1	15,31	20,2
titanic	32,3	21,94	22,84	26,58	22,18	22,5	23,84	22,17	22,43	22,26	22,56
twonorm	3,91	3,42	3,91	3,42	2,9	3,91	3,64	3,34	3,53	3,93	3,43
wine	12,45	3,77	7,81	6,22	3,94	6,58	7,33	4,36	9,58	8,83	5,29
wisconsin	12	5,02	5,95	6,69	4,13	3,69	5,46	4,79	4,73	4,15	4,91

Проведенный эксперимент показал, что после применения оптимизации алгоритмом «Разряд молнии» в среднем по всем наборам данных после оптимизации ошибка классификации уменьшилась на 41,97%. Для обучающей выборки в среднем ошибка классификации уменьшилась на 55,63%, для тестовой - на 28,32%. Алгоритм дал результаты сравнимые с аналогами, на некоторых наборах данных – превосходящие аналоги.

Для сравнения алгоритма «Разряд молнии» с аналогами использовался статистический критерий Вилкоксона. Критерий Вилкоксона позволяет определить различимы ли результаты двух методов. Нулевая гипотеза H_0 в данном эксперименте: результаты разных алгоритмов оптимизации одинаковы. Альтернативная гипотеза H_1: различия между результатами алгоритмов статистически значимы. Нулевая гипотеза отвергается в пользу альтернативной, если рассчитанное значение статистики попадает в критическую область. Уровень значимости был выбран равным 0,05. В таблице 5.4 приведены результаты статистического критерия Вилкоксона для пар сравниваемых

алгоритмов на обучающих и тестовых выборках.

Таблица 5.4 – Результаты статистического критерия Вилкоксона

Алгоритм для сравнения с "Разрядом молнии"	Выборка	Наблюдаемое значение критерия Вилкоксона	Допустимая область критерия Вилкоксона	Нулевая гипотеза
Стая птиц	Обучение	215	[160; 246]	Принята
Рой летучих мышей	Обучение	177	[160; 246]	Принята
Стая кошек	Обучение	203	[160; 246]	Принята
Гравитационный поиск	Обучение	204	[160; 246]	Принята
Серые волки	Обучение	203	[160; 246]	Принята
Роящиеся частицы	Обучение	192	[160; 246]	Принята
Стая ласточек	Обучение	212	[160; 246]	Принята
Кукушкин поиск	Обучение	194	[160; 246]	Принята
Искусственные пчелы	Обучение	196	[160; 246]	Принята
Стая птиц	Тест	220	[160; 246]	Принята
Рой летучих мышей	Тест	194	[160; 246]	Принята
Стая кошек	Тест	200	[160; 246]	Принята
Гравитационный поиск	Тест	214	[160; 246]	Принята
Серые волки	Тест	207	[160; 246]	Принята
Роящиеся частицы	Тест	201	[160; 246]	Принята
Стая ласточек	Тест	218	[160; 246]	Принята
Кукушкин поиск	Тест	212	[160; 246]	Принята
Искусственные пчелы	Тест	210	[160; 246]	Принята

Результаты показали, что нулевая гипотеза была принята во всех случаях. Это означает, что при попарном сравнении разработанного алгоритма с другими метаэвристиками, алгоритм «Разряд молнии» дает статистически незначимую разницу в результате, то есть ошибки классификации сравнимы с ошибками классификации аналогов.

На основании полученных результатов, была оценена эффективность рассматриваемых алгоритмов при помощи статистического критерия Краскела-Уоллиса, который позволяет проверить гипотезу H_0 о равенстве средних с минимальными требованиями к выборочным

данным: предполагается, что ошибки наблюдений независимы и имеют непрерывные распределения. В качестве уровней фактора были взяты 10 алгоритмов оптимизации параметров классификатора. Выборочными данными выступили полученные ошибки классификации на тестовой выборке. Наблюдаемое значение статистики Краскела-Уоллиса составило 413,796. На уровне значимости $\alpha = 0,05$ критическая точка данной статистики $\chi^2_{(10-1)} = 16,919$. Так как наблюдаемое значение больше критического, то гипотеза H_0 отвергается в пользу альтернативной, то есть средние на разных уровнях отличаются значимо. Таким образом, алгоритм оказывает значимое статистическое влияние на ошибку классификации.

Проведено исследование графиков зависимостей ошибки классификации от количества вычислений фитнесс-функции, номера итерации и времени разработанного алгоритма. Графики для наборов данных iris и newthyroid приведены на рисунках 5.1 – 5.6. Графики показали более медленную сходимость алгоритма «Разряд молнии» на начальных итерациях и более резкое по сравнению с другими алгоритмами уменьшение ошибки в конце, свидетельствующее о быстрой сходимости алгоритма.

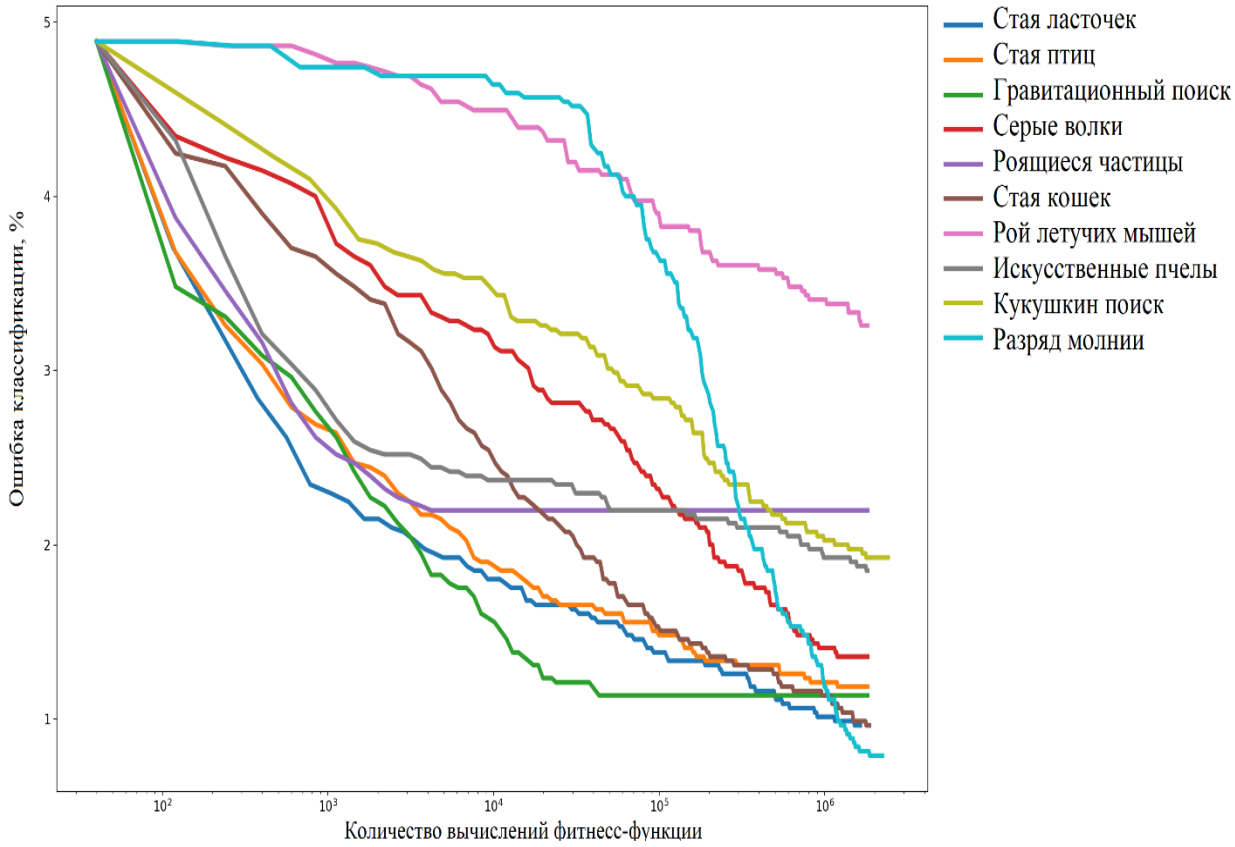

Рисунок 5.1 – График зависимости ошибки классификации от количества вычислений на наборе данных iris

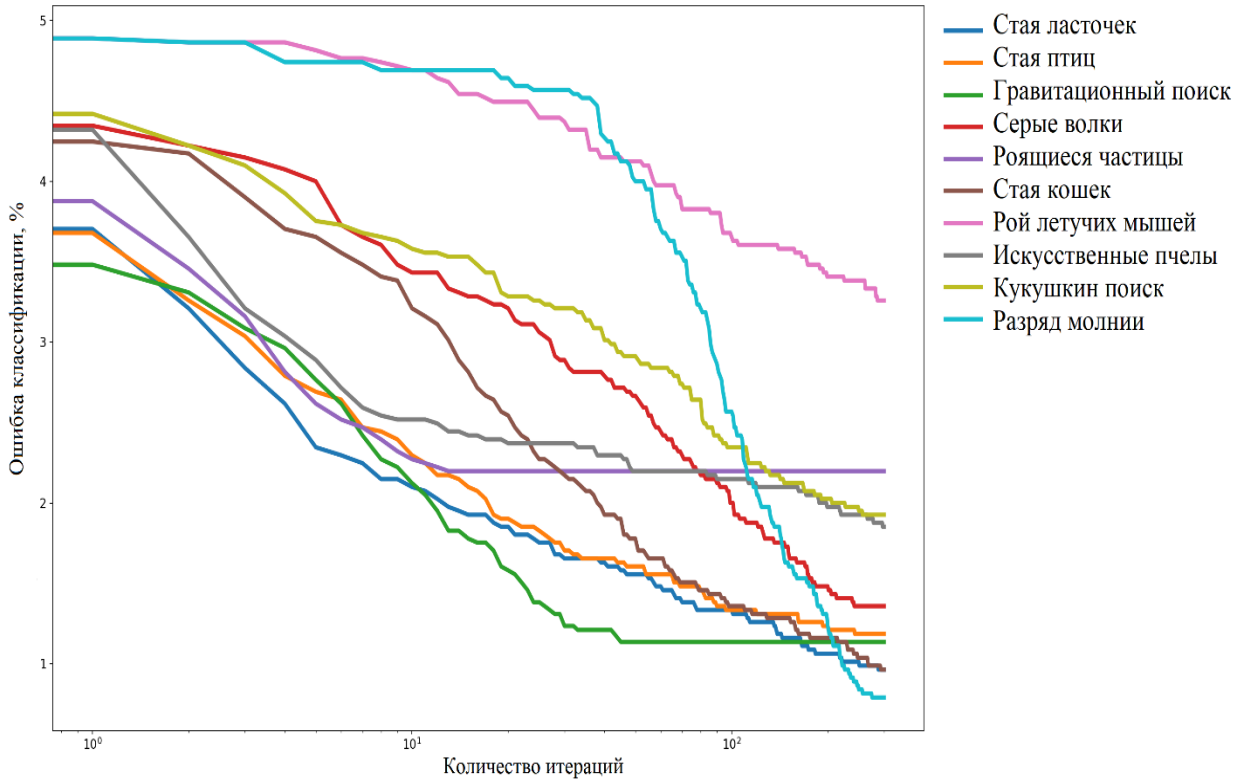

Рисунок 5.2 – График зависимости ошибки классификации от номера итерации на наборе данных iris

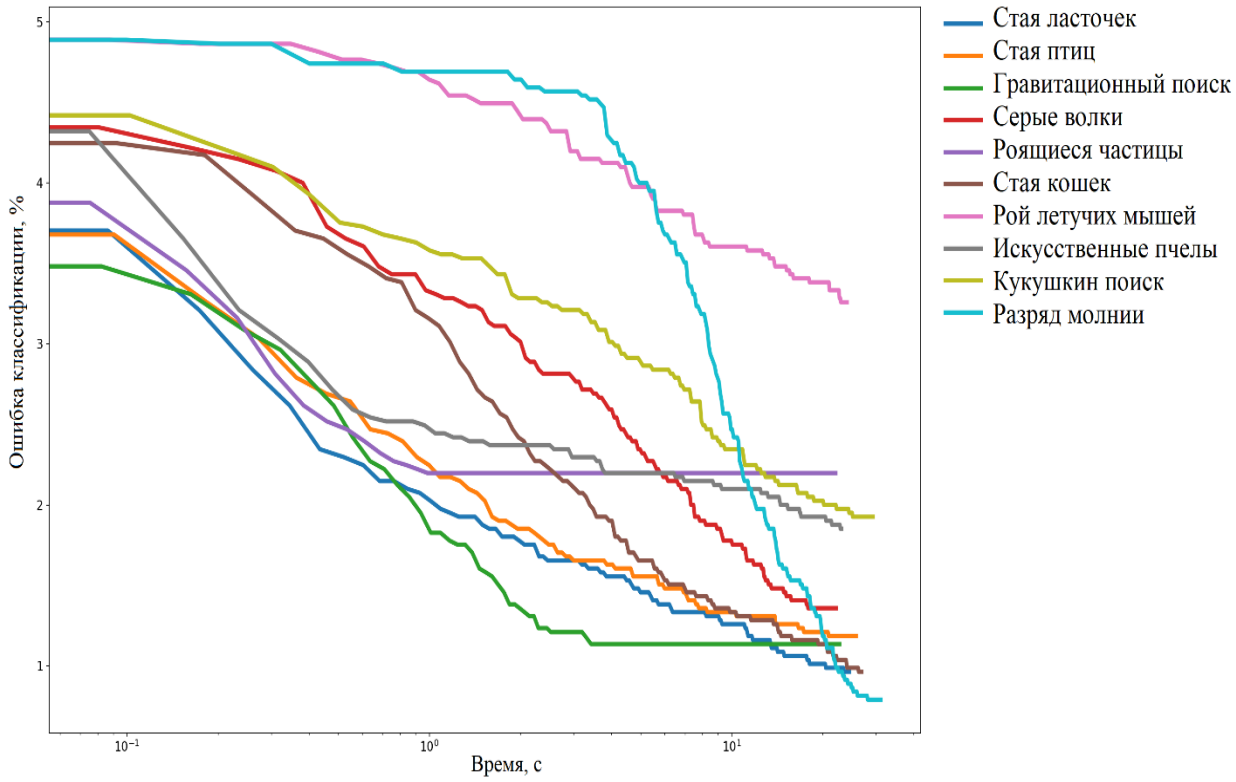

Рисунок 5.3 – График зависимости ошибки классификации от времени на наборе данных iris

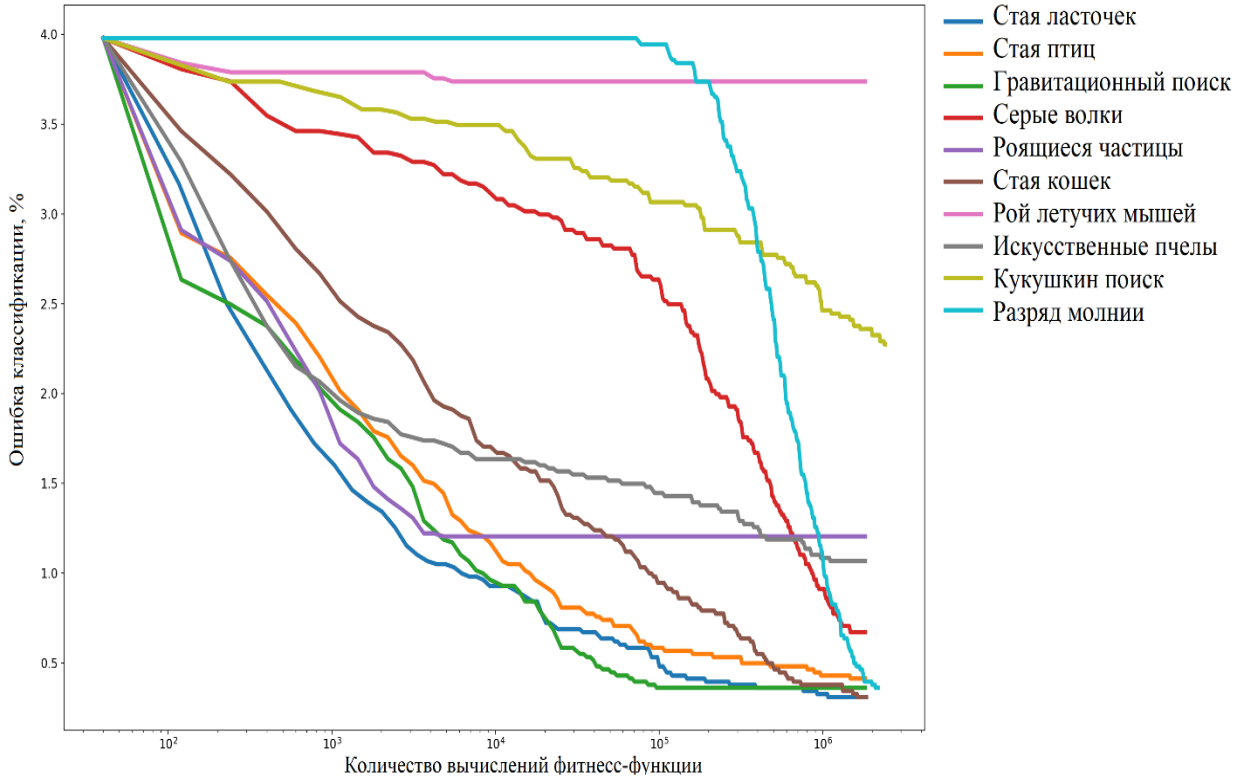

Рисунок 5.4 – График зависимости ошибки классификации от количества вычислений на наборе данных newthyroid

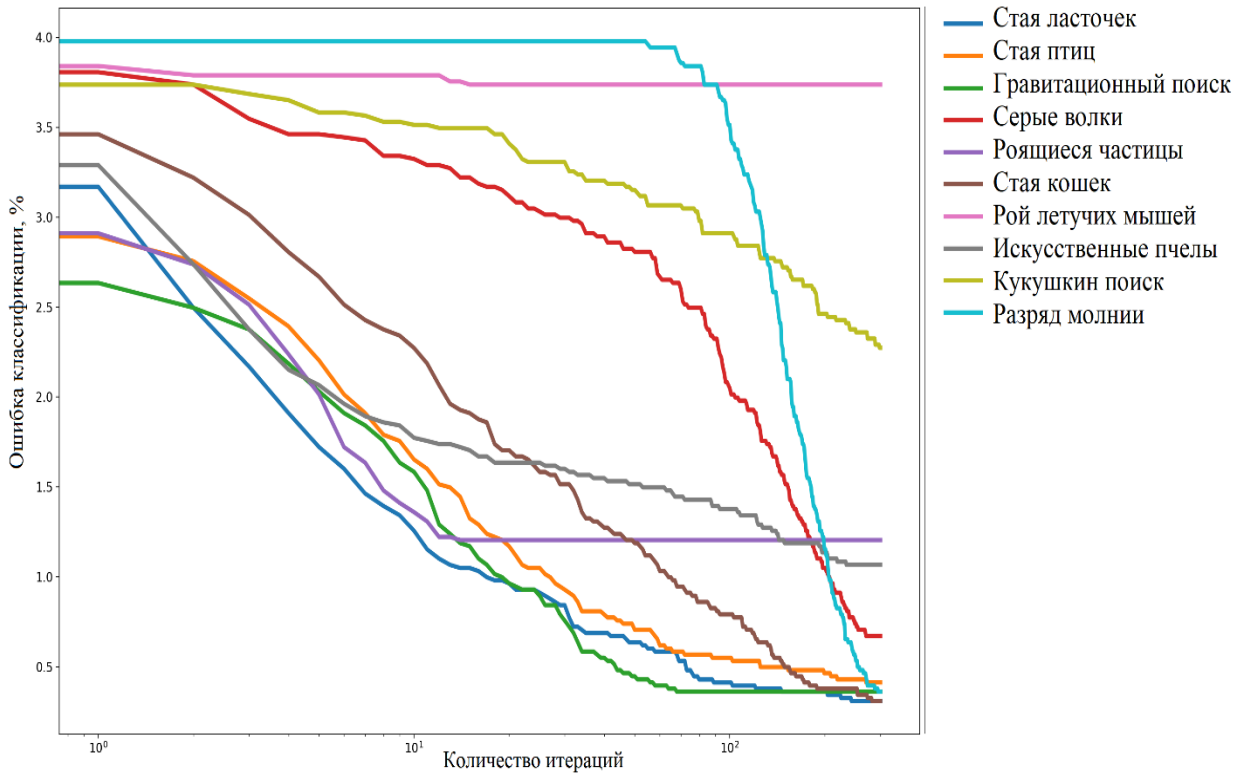

Рисунок 5.5 – График зависимости ошибки классификации от номера итерации на наборе данных newthyroid

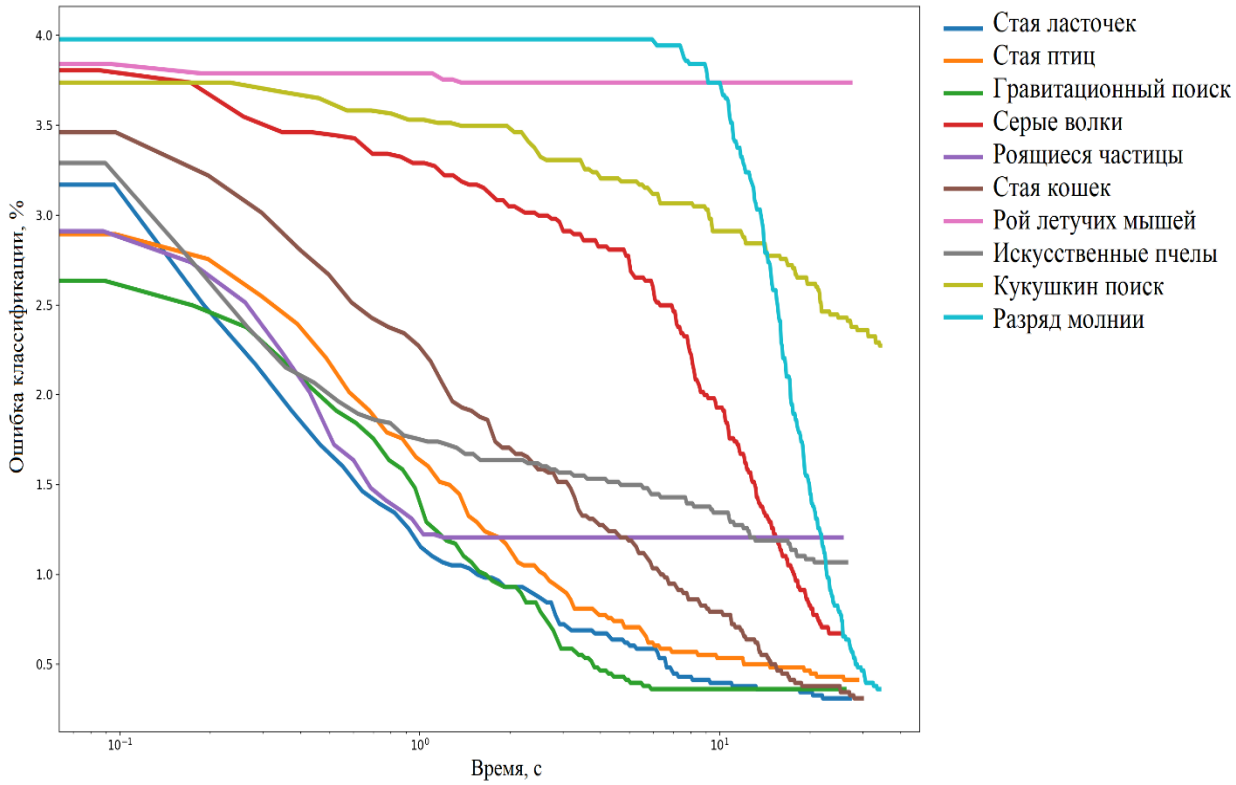

Рисунок 5.6 – График зависимости ошибки классификации от времени на наборе данных newthyroid

6 Отбор признаков классификатора

6.1 Описание эксперимента

При решении задачи отбора значимых признаков классификатора был использован бинарный вариант алгоритма «Разряд молнии».

Эксперимент проходил по схеме десятикратной кросс-валидации. Строился нечеткий классификатор, при этом были использованы все признаки набора данных. Далее запускался один из алгоритмов для отбора признаков и на каждой из 10 итераций кросс-валидации проводился отбор значимых признаков. Количество отобранных признаков запоминалось, в конце вычислялось среднее значение количества отобранных признаков и оценивалась ошибка классификации на выбранных значимых признаках. Оптимизация параметров классификатора при этом не производилась.

Эксперимент проводился на известных наборах данных из репозитория KEEL с различными количествами признаков. Использованные наборы данных и их характеристики приведены в таблице 6.1.

Таблица 6.1 – Наборы данных

Название	Признаки	Классы	Экземпляры
hayes-roth	4	3	160
mammographic	5	2	830
newthyroid	5	3	215
bupa	6	2	345
monk-2	6	2	432
appendicitis	7	2	106
ecoli	7	8	336
led7digit	7	10	500
glass	9	7	214
magic	10	2	19020
cleveland	13	5	297
heart	13	2	270
marketing	13	9	6876
australian	14	2	690

Продолжение таблицы 6.1

Название	Признаки	Классы	Экземпляры
letter	16	26	20000
vehicle	18	4	846
bands	19	2	365
hepatitis	19	2	80
segment	19	7	2310
ring	20	2	7400
twonorm	20	2	7400
thyroid	21	3	7200
wdbc	30	2	569
ionosphere	33	2	351
dermatology	34	6	358
texture	40	11	5500
spambase	57	2	4597
optdigits	64	10	5620
coil2000	85	2	9822
movement_libras	90	15	360

Вычисления производились на персональном компьютере со следующими характеристиками:

- процессор Intel Core i7-6400 CPU 3,40 GHz;
- установленная ОЗУ 32 Гб;
- ОС Windows 10;
- отсутствие подключения к сети Интернет.

6.2 Результаты эксперимента

Проведено сравнение результатов, полученных методом полного перебора, дающего оптимальный результат, «жадным» алгоритмом, который дает результат близкий к оптимальному, случайным поиском, который выступил в качестве контрольного алгоритма, и бинарным вариантом разработанного алгоритма «Разряд молнии». Отбор признаков методом полного перебора был осуществлен только для части наборов данных, так как на наборах с большим количеством признаков временные затраты превысили допустимые

значения. Полученные ошибки классификации на обучающих и тестовых выборках в процентах, количество отобранных признаков, а также время отбора в секундах приведены в таблице 6.2.

Таблица 6.2 – Результаты эксперимента

Набор данных	Параметр сравнения	Без отбора	Полный перебор	Жадный алгоритм	Случайный поиск	Разряд молнии
hayes-roth	Обучение	71,25	59,38	59,38	59,38	59,38
	Тест	71,01	59,38	59,38	59,38	59,38
	Признаки	4,00	1,00	1,00	1,00	1,00
	Время	-	0,11	0,40	12,61	29,83
mammographic	Обучение	32,13	31,04	31,16	31,04	31,04
	Тест	31,78	30,47	31,08	30,47	30,47
	Признаки	5,00	3,10	1,00	4,04	3,66
	Время	-	0,52	0,14	15,65	37,60
newthyroid	Обучение	4,29	3,51	3,62	3,51	3,51
	Тест	4,16	4,65	4,20	5,11	4,36
	Признаки	5,00	2,40	2,00	3,04	2,72
	Время	-	0,37	0,11	13,77	30,17
bupa	Обучение	51,11	40,61	41,03	40,61	40,61
	Тест	51,04	42,34	43,22	42,34	42,34
	Признаки	6,00	2,60	1,70	2,60	2,60
	Время	-	0,19	0,36	13,73	31,53
monk-2	Обучение	44,44	44,44	44,44	44,44	44,44
	Тест	44,40	44,40	44,40	44,40	44,40
	Признаки	6,00	1,00	1,00	6,00	2,76
	Время	-	0,52	0,14	14,35	30,37
appendicitis	Обучение	32,88	18,01	18,65	18,01	18,01
	Тест	33,97	23,42	21,61	22,22	22,42
	Признаки	7,00	2,40	1,90	2,72	2,70
	Время	-	0,26	0,39	13,81	34,22
ecoli	Обучение	54,50	50,93	55,45	50,93	50,93
	Тест	54,41	52,22	56,15	52,47	52,76
	Признаки	7,00	4,80	2,10	5,30	5,22
	Время	-	0,51	0,51	34,56	76,42
led7digit	Обучение	88,55	88,55	88,55	88,55	88,55
	Тест	92,32	92,32	92,32	92,32	92,32
	Признаки	7,00	1,00	1,00	7,00	2,86
	Время	-	0,51	0,49	38,60	83,50
glass	Обучение	50,05	41,02	44,53	41,02	41,09
	Тест	47,71	43,28	43,66	43,50	43,62
	Признаки	9,00	6,50	5,10	6,50	6,48
	Время	-	1,24	0,47	28,68	62,01

Продолжение таблицы 6.2

Набор данных	Параметр сравнения	Без отбора	Полный перебор	Жадный алгоритм	Случайный поиск	Разряд молнии
magic	Обучение	42,40	27,01	27,01	27,01	27,01
	Тест	42,07	27,03	27,03	27,03	27,03
	Признаки	10,00	1,50	1,50	1,50	1,50
	Время	-	83,26	7,65	729,25	1593,29
cleveland	Обучение	55,84	45,94	45,94	45,94	45,94
	Тест	57,69	45,93	45,93	46,31	46,31
	Признаки	13,00	1,00	1,00	3,78	3,06
	Время	-	20,05	0,74	32,11	66,42
heart	Обучение	32,80	30,53	30,78	30,53	30,53
	Тест	32,59	32,59	31,48	33,18	32,59
	Признаки	13,00	2,20	1,90	6,76	5,40
	Время	-	10,86	0,57	17,81	40,05
marketing	Обучение	90,31	81,75	81,75	81,75	81,75
	Тест	90,37	81,75	81,75	81,75	81,75
	Признаки	13,00	1,00	1,00	5,90	4,72
	Время	-	153,53	2,54	221,70	444,07
australian	Обучение	50,29	40,31	40,31	40,31	40,31
	Тест	50,58	40,29	40,29	40,50	40,38
	Признаки	14,00	2,00	2,00	5,90	5,14
	Время	-	27,42	0,73	22,29	49,61
letter	Обучение	61,58	59,74	59,97	59,95	61,11
	Тест	61,63	59,78	60,06	60,00	61,12
	Признаки	16,00	13,00	12,70	12,70	12,62
	Время	-	13882,72	33,72	2588,60	5547,79
vehicle	Обучение	70,02	-	57,17	56,29	55,88
	Тест	70,80	-	59,80	56,86	56,96
	Признаки	18,00	-	4,60	6,92	5,70
	Время	-	-	1,25	54,40	114,61
bands	Обучение	47,43	30,56	35,28	31,39	31,85
	Тест	47,04	34,77	38,65	36,15	35,39
	Признаки	19,00	7,40	1,60	8,40	7,28
	Время	-	1252,44	1,19	30,51	68,36
hepatitis	Обучение	72,78	-	15,13	13,88	13,88
	Тест	73,31	-	16,41	16,88	17,61
	Признаки	19,00	-	1,40	8,46	7,16
	Время	-	-	0,62	21,05	43,63
segment	Обучение	20,03	-	15,59	38,24	31,99
	Тест	20,26	-	16,28	38,36	31,92
	Признаки	19,00	-	7,20	3,56	3,88
	Время	-	-	2,28	142,42	273,95

Продолжение таблицы 6.2

Набор данных	Параметр сравнения	Без отбора	Полный перебор	Жадный алгоритм	Случайный поиск	Разряд молнии
ring	Обучение	50,45	-	34,50	70,62	64,33
	Тест	50,49	-	34,34	70,75	64,36
	Признаки	20,00	-	2,80	31,74	22,10
	Время	-	-	5,01	1179,26	2005,90
twonorm	Обучение	3,98	-	3,93	3,97	3,98
	Тест	3,95	-	4,05	3,97	3,95
	Признаки	20,00	-	18,80	19,84	20,00
	Время	-	-	3,46	141,47	309,29
thyroid	Обучение	92,79	-	3,12	3,14	3,15
	Тест	92,72	-	3,24	3,24	3,26
	Признаки	21,00	-	4,10	8,44	7,18
	Время	-	-	5,70	172,15	366,21
wdbc	Обучение	7,56	-	3,05	4,05	3,91
	Тест	7,03	-	4,57	4,75	4,82
	Признаки	30,00	-	6,10	12,02	9,86
	Время	-	-	2,51	43,61	93,23
ionosphere	Обучение	11,14	-	11,33	10,67	10,58
	Тест	10,79	-	12,89	13,50	13,81
	Признаки	33,00	-	3,40	22,58	18,40
	Время	-	-	2,90	39,34	83,03
dermatology	Обучение	24,39	-	27,37	12,68	15,26
	Тест	25,19	-	29,13	15,16	16,62
	Признаки	34,00	-	5,40	19,96	15,90
	Время	-	-	5,21	72,46	152,87
texture	Обучение	30,18	-	25,29	27,52	27,42
	Тест	30,18	-	25,64	27,60	27,41
	Признаки	40,00	-	5,30	16,82	13,72
	Время	-	-	78,62	734,06	1555,40
spambase	Обучение	39,42	-	20,20	23,02	23,30
	Тест	39,44	-	20,21	23,38	23,17
	Признаки	57,00	-	6,50	25,30	24,72
	Время	-	-	51,17	229,75	492,72
optdigits	Обучение	89,97	-	68,30	48,28	52,90
	Тест	89,93	-	69,19	48,85	52,83
	Признаки	64,00	-	17,80	44,26	33,66
	Время	-	-	274,73	388,30	1154,25
coil2000	Обучение	5,97	-	5,96	5,95	5,96
	Тест	5,97	-	5,99	6,01	5,97
	Признаки	85,00	-	1,00	50,80	79,52
	Время	-	-	396,43	761,86	1604,54

Продолжение таблицы 6.2

Набор данных	Параметр сравнения	Без отбора	Полный перебор	Жадный алгоритм	Случайный поиск	Разряд молнии
movement_libras	Обучение	52,28	-	48,31	14,56	21,37
	Тест	49,27	-	48,54	14,89	21,62
	Признаки	90,00	-	11,90	9,50	14,14
	Время	-	-	197,83	97,70	333,45

В сравнении с полным перебором в 81,25% при обучении и 62,5% при тестировании алгоритм «Разряд молнии» дал оптимальное решение. При этом заметно, что уже на 16 признаках алгоритм «Разряд молнии» на порядок обходит полный перебор по времени.

На 20 наборах данных для обучающих и 18 для тестовых выборок из 30 наборов данных алгоритм «Разряд молнии» дал результат лучше или такой же, как случайный поиск. Равенство в данном случае означает, что оба алгоритма нашли оптимальное решение.

На всех наборах данных алгоритм «Разряд молнии» дал результат лучше, чем «Жадный» алгоритм (или оба алгоритма нашли оптимальный результат).

Для сравнения алгоритма «Разряд молнии» с аналогами использовался статистический критерий Вилкоксона. Критерий Вилкоксона позволяет определить различимы ли результаты двух методов. Нулевая гипотеза H_0 в данном эксперименте: результаты разных алгоритмов отбора признаков идентичны. Альтернативная гипотеза H_1: различия между результатами алгоритмов статистически значимы. Нулевая гипотеза отвергается в пользу альтернативной, если рассчитанное значение статистики попадает в критическую область. Уровень значимости был выбран равным 0,05. В таблице 6.3 приведены результаты статистического критерия Вилкоксона для пар сравниваемых алгоритмов.

Из результатов видно, что нулевая гипотеза была принята при сравнении алгоритма «Разряд молнии» с методом случайного поиска и «Жадного» алгоритма и на обучающих, и тестовых выборках. Это показывает, что статистически алгоритмы дают сравнимый результат. При сравнении с полным перебором нулевая гипотеза была принята, из чего может быть сделан вывод, что алгоритм «Разряд молнии» дает результат близкий к оптимальному. При сравнении числа отобранных признаков в случаях сравнения с «Жадным» алгоритмом и Полным перебором нулевая гипотеза была отклонена. Это может быть объяснено тем, что разные количества и сочетания признаков могут давать близкий результат, а «Жадный» алгоритм отбирает минимум признаков. При сравнении числа отобранных признаков при сравнении со «Случайным поиском» нулевая гипотеза была

принята, что в действительности означает, что оба алгоритма отбирают примерно одинаковое количество признаков.

Таблица 6.3 – Результаты статистического критерия Вилкоксона

Пара алгоритмов	Данные	Наблюдаемое значение критерия Вилкоксона	Допустимая область критерия Вилкоксона	Нулевая гипотеза
«Разряд молнии» и «Случайный поиск»	Обучающая выборка	913	[782; 1048]	Принята
	Тестовая выборка	912	[782; 1048]	Принята
	Количество признаков	866	[782; 1048]	Принята
«Разряд молнии» и «Жадный» алгоритм	Обучающая выборка	909	[782; 1048]	Принята
	Тестовая выборка	914	[782; 1048]	Принята
	Количество признаков	211	[291; 461]	Отклонена
«Разряд молнии» и Полный перебор	Обучающая выборка	423	[291; 461]	Принята
	Тестовая выборка	430	[291; 461]	Принята
	Количество признаков	1130	[782; 1048]	Отклонена

7 Построение классификатора на наборе данных KDD Cup 1999 Data

Набор данных KDD Cup 1999 Data использовался для Третьего международного конкурса инструментов по сбору и интеллектуальному анализу данных, который проводился совместно с Пятой международной конференцией по сбору и интеллектуальному анализу данных KDD-99. Задача конкурса заключалась в создании инструмента для обнаружения вторжений в сеть, прогнозирующей модели, способной различать «плохие» соединения, называемые вторжениями или атаками, и «хорошие» нормальные соединения. Эта база данных содержит стандартный набор данных для аудита, который включает в себя широкий спектр вторжений, моделируемых в среде военной сети.

Данный набор данных содержит около 5000000 записей. Каждая запись характеризует сетевое соединение по 40 признакам и одной из 23 меток класса, среди которых различные сетевые атаки и нормальные соединения.

Набор данных необходимо было подготовить для обработки. Поскольку количество экземпляров классов не сбалансировано, то есть к некоторым классам принадлежит малое количество экземпляров, набор был преобразован. В первом варианте вместо 23 классов были использованы 2 метки классов: все соединения, относящиеся к сетевым атакам, были объединены в один класс, нормальные соединения – в другой. Во втором варианте 23 класса были объединены в 5 более крупных групп, а именно:

- нормальные соединения с меткой normal.

- DoS (Denial of Service) – атаки, направленные на отказ в обслуживании атакуемой системы. В данную группу вошли метки back, land, neptune, pod, smurf, teardrop;

- R2L (Remote to Local) – атаки, направленные на несанкционированное получение удаленного доступа. К данной группе относятся ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster;

- U2R (User to Root) – атаки, имеющие своей целью повышение привилегий пользователя до суперпользователя (сетевого администратора). Метки классов данной группы: buffer_overflow, loadmodule, perl, rootkit;

- Probing – атаки, направленные на сканирование сетевых портов с целью получения информации о системе и поиска уязвимостей. К данному типу атак были отнесены метки ipsweep, nmap, portsweep, satan.

Поскольку набор данных содержит большое количество данных, перед оптимизацией параметров классификатора проводилось разделение данных на указанное количество частей (подвыборок). В оптимизации параметров принимала участие только одна из

сформированных подвыборок. Подвыборка выбиралась случайным образом. При разделении в каждой части сохранялось распределение экземпляров по классам, как в исходных данных.

Полученные ошибки классификации при различном разделении перед оптимизацией параметров и перед отбором и оптимизацией приведены в таблицах 7.1 и 7.2.

Таблица 7.1 – Результаты ошибки классификации при различном разделении перед оптимизацией параметров

№	Классы	Разделение	Ошибка до оптимизации, обучение, %	Ошибка до оптимизации, тест, %	Ошибка после оптимизации, обучение, %	Ошибка после оптимизации, тест, %	Время, с
1	2	200	80,31	80,31	18,6	18,6	1287
2	2	150	80,31	80,31	14,96	14,96	1564
3	2	100	80,31	80,31	17,43	17,43	2106
4	2	50	80,31	80,31	19,31	19,3	4069
5	2	20	80,31	80,31	17,91	17,91	11050
6	2	12	80,31	80,31	2,33	2,34	18146
7	2	10	80,31	80,31	1,6	1,63	20061
8	2	5	80,31	80,31	1,63	1,61	44148
9	5	200	42,19	42,18	19,44	19,44	1369
10	5	150	42,19	42,18	17,67	17,67	1605
11	5	100	42,19	42,18	23,36	23,38	2240
12	5	50	42,19	42,18	17,35	17,38	4072
13	5	20	42,19	42,18	12,94	12,93	12047
14	5	12	42,19	42,18	5,7	5,71	25902
15	5	10	42,19	42,18	2,31	2,30	30309
16	5	5	42,19	42,18	2,38	2,39	69275
17	23	150	42,86	42,85	11,18	11,18	2384

Вычисления при оптимизации параметров производились на персональном компьютере со следующими характеристиками:

- процессор Intel Core i7-4510 CPU 3,40 GHz;

- установленная ОЗУ 8 Гб;

- ОС Windows 10;

- отсутствие подключения к сети Интернет.

Вычисления при отборе признаков и оптимизации параметров производились на персональном компьютере со следующими характеристиками:

- процессор Intel Core i7-6400 CPU 3,40 GHz;

- установленная ОЗУ 32 Гб;

- ОС Windows 10;

- отсутствие подключения к сети Интернет.

Таблица 7.2 – Результаты ошибки классификации при различном разделении перед отбором признаков и оптимизацией параметров

№	Классы	Разделение	Ошибка до оптимизации, обучение, %	Ошибка до оптимизации, тест, %	Ошибка после преобразований, обучение, %	Ошибка после преобразований, тест, %	Отбор признаков	Время, с
1	2	200	80,31	80,31	24,75	24,77	15,6	1330
2	2	150	80,31	80,31	14,24	14,26	14,8	1883
3	2	100	80,31	80,31	26,54	26,53	14	2739
4	2	50	80,31	80,31	18,18	18,19	15,1	5269
5	2	20	80,31	80,31	20,38	20,37	13,2	14865
6	2	10	80,31	80,31	1,7	1,7	13,4	31392
7	2	5	80,31	80,31	1,56	1,57	12,9	658891
8	5	200	42,19	42,18	32,49	32,5	9,6	1092
9	5	150	42,19	42,18	33	33	9,6	1597
10	5	100	42,19	42,18	28,15	28,15	10,1	3348
11	5	50	42,19	42,18	28,29	28,32	10,1	4985
12	5	20	42,19	42,18	16,03	16,02	10,2	14137
13	5	10	42,19	42,18	4,8	4,77	12,3	31322
14	5	5	42,19	42,18	3,29	3,31	12,2	68970

Проведенные эксперименты показали, что использование менее чем 1/10 части данных сильно увеличивает ошибку классификации. При этом обработка 1/5 части данных по сравнению с 1/10 увеличивает время в 2 раза и при этом не уменьшает ошибку классификации.

Для сравнения классификатора, построенного с использованием алгоритмов «Разряд молнии», с аналогичными системами был применен критерий «Akaike Information Criterion» (AIC) [26, 27].

Значение критерия AIC может быть вычислено по формуле:

$$AIC = \ln ER_T + \frac{2}{m}(1+c\ F_S),$$

где ER_T – ошибка классификатора;

F_S – количество признаков, используемых классификатором;

m – количество классифицируемых экземпляров классов;

c – коэффициент, позволяющий задать приоритет сложности системы над точностью: чем больше значение, тем выше приоритет.

В зависимости от точности и сложности системы критерий позволяет произвести выбор оптимального варианта классификатора из множества рассмотренных. Сложность системы в данном случае определяется количеством используемых признаков. Чем меньше значение критерия *AIC*, тем выше эффективность классификатора.

Аналогичные системы сетевых атак характеризуются ошибками первого и второго рода. За ошибку первого рода принимается отношение количества нормальных соединений, классифицированных, как сетевые атаки, к общему количеству нормальных соединений:

$$\mathrm{ER}_1 = \frac{\text{Количество некорректно опр.норм.соединений}}{\text{Общее количество норм.соединений}} * 100\%.$$

Ошибка второго рода - отношение количества сетевых атак, распознанных как нормальные соединения, к общему количеству сетевых атак:

$$\mathrm{ER}_2 = \frac{\text{Количество некорректно опр.атак}}{\text{Общее количество атак}} * 100\%.$$

Поскольку для систем-аналогов точность определяется значениями ошибок первого и второго рода, то критерий *AIC* может быть найден по формуле:

$$AIC = \ln(ER_1 + k\ ER_2) + \frac{2}{m}(1+c\ F_S),$$

где k > 1 – коэффициент, определяющий критичность ошибки второго рода относительно ошибки первого рода.

Сравнение проводилось с системами распознавания атак, сформированными на наборе данных KDD Cup 1999 Data, построенными следующими методами:

- метод гауссовых процессов GP-mt [27];

- метод опорных векторов SVM [27];

- метод нейронных сетей MLP;

- комбинации методов ассоциативных правил и нейронных сетей CPAR/MLP [28];

- методы на базе информационной теории PKID+Cons+FVQIT и EMD+Cons+FVQIT [29];

- метод на базе классификатора Байеса AODE [30];

- метод кластеризации AP [31].

Ошибки первого и второго рода для данных систем, а также количество используемых признаков приведены в таблице 7.3.

Таблица 7.3 – Значения критерия AIC для систем-аналогов распознавания сетевых атак

Классификатор	Ошибка первого рода ER_1, %	Ошибка второго рода ER_2, %	Признаки F_S	m	c	k	AIC
GP-mt	2,00	0,07	41	5000000	0,5	2	0,76
SVM	2,00	10	41	5000000	0,5	2	3,09
MLP	1,49	5,29	16	5000000	0,5	2	2,49
CPAR/MLP	1,58	4,86	16	5000000	0,5	2	2,42
PKID+Cons+FVQIT	7,27	0,48	6	5000000	0,5	2	2,11
EMD+Cons+FVQIT	5,50	1,54	7	5000000	0,5	2	2,15
AODE	0,10	0,46	-	5000000	0,5	2	0,02
AP	1,01	1,6	41	5000000	0,5	2	1,44

Для расчета критерия AIC для классификаторов, построенных с использованием алгоритмов «Разряд молнии» были выбраны лучшие из полученных значений точности из таблиц 7.1 и 7.2. Рассчитанные значения критерия AIC приведены в таблице 7.4.

Таким образом, найденные значения критерия AIC для классификаторов, построенных с использованием алгоритма «Разряд молнии» для оптимизации параметров классификатора и его бинаризованного варианта для отбора признаков, меньше, чем значения критерия AIC, рассчитанные для всех систем-аналогов за исключением системы

распознавания атак на основе метода AODE. На основании этого можно утверждать, что алгоритм «Разряд молнии» эффективен и может быть применен для построения систем распознавания сетевых вторжений.

Таблица 7.4 – Значения критерия AIC для классификаторов, построенных с использованием алгоритмов «Разряд молнии»

Классификатор	Ошибка классификатора ER_T, %	Признаки F_S	m	c	AIC
«Разряд молнии» для оптимизации параметров, 2 метки класса	1,61	41	1000000	0,5	0,48
«Разряд молнии» для оптимизации параметров, 23 метки класса	2,3	41	500000	0,5	0,83
«Разряд молнии» для оптимизации параметров и отбора признаков, 2 метки класса	1,57	12,9	1000000	0,5	0,45
«Разряд молнии» для оптимизации параметров и отбора признаков, 23 метки класса	3,31	12,2	1000000	0,5	1,20

8 Построение классификатора на наборе данных SVC 2004

8.1 Описание набора данных SVC 2004

Набор данных SVC 2004 предназначался для Первого международного конкурса проверки подписи и содержит данные, используемые для аутентификации по рукописной подписи. Задача определения подлинности рукописной подписи актуальна в связи с тенденцией перехода к цифровой экономике.

Набор состоит из двух отдельных задач для проверки подписи, каждая из которых основана на своей базе данных подписей. Данные о подписях для первой задачи содержат только информацию о координатах, то есть сигналы X и Y пера, данные о подписях для второй задачи также содержат дополнительную информацию, включая давление и углы наклона пера. Первая задача подходит для проверки подписи на небольших устройствах ввода на основе пера, а вторая - для цифровых планшетов.

Каждая база данных задачи содержит 40 файлов. В каждом файле содержится варианты нанесения отдельной подписи. Набор данных содержит 100 признаков, отражающих варианты нанесения подписи, и 2 класса (подпись является настоящей или поддельной).

В каждом файле подпись представлена как последовательность точек. В первой строке хранится одно целое число, которое является общим количеством точек в подписи. Каждая из следующих строк соответствует одной точке, характеризуемой функциями, перечисленными в следующем порядке (последние три функции отсутствуют в файлах сигнатур для первой задачи):

- X-coordinate - масштабированное положение курсора вдоль оси X;
- Y-coordinate - масштабированное положение курсора вдоль оси Y;
- Time stamp - системное время, в которое было опубликовано событие;
- Button status - текущее состояние пера (0 для поднятого пера и 1 для опущенного);
- Azimuth - вращение пера по оси Z по часовой стрелке;
- Altitude - угол наклона пера относительно положительной оси Z;
- Pressure - скорректированное значение нормального давления.

Сбор подписей производился при помощи цифрового планшета (планшет WACOM Intuos), в два этапа. Каждого автора подписи просили произвести 20 подлинных подписей в двух отдельных сессиях, еще 20 подписей-подделок были получены от других участников. Для соблюдения конфиденциальности участники не использовали свои настоящие подписи.

8.2 Описание эксперимента

Каждый файл с подписями был разбит на нарезки для кросс-валидации. Поскольку данных сравнительно немного, была использована 2-кратная кросс-валидация, т.е. данные в файле были разбиты на две одинаковые по количеству части, причем с одинаковым количеством оригинальных и поддельных подписей в каждой части. Классификатор строился на первой части, на второй проверялась его точность. Потом части менялись. Итоговая точность классификации была вычислена, как среднее значение точности по двум частям. В итоге для каждого файла была получена своя точность.

Ошибка классификации была получена до преобразований и после того, как были проведены отбор признаков и оптимизация параметров при помощи алгоритма «Разряд молнии». Данные были усреднены по 5 прогонам.

Эксперимент проводился на персональном компьютере со следующими параметрами:

- процессор Intel Core i7-4510 CPU 3,40 GHz;
- установленная ОЗУ 8 Гб;
- ОС Windows 10;
- отсутствие подключения к сети Интернет.

Ошибки классификации на обучающих и тестовых выборках, количество отобранных признаков и время, за которое был произведен отбор признаков и оптимизация параметров, полученные в результате эксперимента, приведены в таблицах 8.1 и 8.2.

Таблица 8.1 – Результаты эксперимента для Задачи 1

Подпись	Ошибка классификации, обучающая выборка, до преобразований, %	Ошибка классификации, тестовая выборка, до преобразований, %	Ошибка классификации, обучающая выборка, после преобразований, %	Ошибка классификации, тестовая выборка, после преобразований, %	Количество отобранных признаков	Время, с
Подпись 1	30,79	30,53	0	13,81	39,7	255
Подпись 2	48,82	43,42	0	18,42	36,8	253
Подпись 3	59,34	69,21	0	19,84	31,7	241
Подпись 4	54,08	35,92	0	18,71	36,5	246

Продолжение таблицы 8.1

Подпись	Ошибка классификации, обучающая выборка, до преобразований, %	Ошибка классификации, тестовая выборка, до преобразований, %	Ошибка классификации, обучающая выборка, после преобразований, %	Ошибка классификации, тестовая выборка, после преобразований, %	Количество отобранных признаков	Время, с
Подпись 5	48,82	36,05	0	29,76	37,4	248
Подпись 6	38,42	41,05	0	11,84	39,4	250
Подпись 7	46,18	35,79	0	23,68	35,4	245
Подпись 8	33,42	22,89	0	8,66	43,5	266
Подпись 9	33,16	38,16	0	13,73	33,0	243
Подпись 10	51,32	36,05	0	11,87	33,3	243
Подпись 11	35,79	28,16	0	20,95	36,7	251
Подпись 12	56,45	38,55	0	14,97	36,1	252
Подпись 13	48,68	48,68	0	20,95	39,6	257
Подпись 14	33,55	22,89	0	13,39	41,8	259
Подпись 15	61,84	43,55	0	17,95	38,0	257
Подпись 16	33,29	28,03	0	12,87	40,5	258
Подпись 17	41,18	40,79	0	12,42	38,6	256
Подпись 18	28,42	17,63	0	6,18	35,6	251
Подпись 19	43,42	43,68	0	23,53	36,4	252
Подпись 20	43,82	36,05	0	24,50	37,9	260
Подпись 21	35,92	36,05	0	9,79	38,7	255
Подпись 22	23,16	33,29	0	16,97	38,2	256
Подпись 23	36,05	36,05	0	13,50	41,3	255
Подпись 24	33,42	27,89	0	19,34	41,3	262
Подпись 25	35,92	23,03	0	7,16	41,5	254
Подпись 26	43,55	33,03	0	13,74	38,6	253
Подпись 27	48,82	33,03	0	15,89	35,4	244
Подпись 28	53,68	53,95	0	12,81	32,5	240

Продолжение таблицы 8.1

Подпись	Ошибка классификации, обучающая выборка, до преобразований, %	Ошибка классификации, тестовая выборка, до преобразований, %	Ошибка классификации, обучающая выборка, после преобразований, %	Ошибка классификации, тестовая выборка, после преобразований, %	Количество отобранных признаков	Время, с
Подпись 29	33,55	28,29	0	22,00	38,1	251
Подпись 30	33,29	27,63	0	14,97	34,3	243
Подпись 31	41,05	33,03	0	9,21	34,2	245
Подпись 32	43,29	33,16	0	15,34	36,3	245
Подпись 33	64,47	64,34	0	16,47	33,2	240
Подпись 34	36,05	36,18	0	14,89	42,0	254
Подпись 35	35,92	35,92	0	19,97	36,4	247
Подпись 36	41,18	46,05	0	15,45	35,2	244
Подпись 37	38,42	20,26	0	19,45	35,9	245
Подпись 38	28,29	23,55	0	9,84	38,6	250
Подпись 39	28,29	23,29	0	10,79	38,0	250
Подпись 40	64,21	56,32	0	17,87	35,4	247
Ср. знач.	41,73	36,04	0	15,84	37,33	251

Таблица 8.2 – Результаты эксперимента для Задачи 2

Подпись	Ошибка классификации, обучающая выборка, до преобразований, %	Ошибка классификации, тестовая выборка, до преобразований, %	Ошибка классификации, обучающая выборка, после преобразований, %	Ошибка классификации, тестовая выборка, после преобразований, %	Количество отобранных признаков	Время, с
Подпись 1	41,05	18,03	0	10,23	40,0	256

Продолжение таблицы 8.2

Подпись	Ошибка классификации, обучающая выборка, до преобразований, %	Ошибка классификации, тестовая выборка, до преобразований, %	Ошибка классификации, обучающая выборка, после преобразований, %	Ошибка классификации, тестовая выборка, после преобразований, %	Количество отобранных признаков	Время, с
Подпись 2	43,68	48,95	0	18,00	40,0	253
Подпись 3	51,32	36,05	0	13,47	37,1	248
Подпись 4	33,55	33,16	0	11,68	35,5	245
Подпись 5	53,95	40,92	0	28,05	37,6	248
Подпись 6	40,92	46,32	0	27,13	38,3	253
Подпись 7	38,42	33,55	0	13,84	37,0	249
Подпись 8	35,79	25,79	0	14,21	35,8	251
Подпись 9	43,55	35,79	0	25,53	38,0	249
Подпись 10	35,92	30,66	0	21,66	40,7	257
Подпись 11	38,55	25,92	0	8,63	37,5	254
Подпись 12	48,55	46,32	0	8,68	37,7	253
Подпись 13	30,53	35,79	0	11,79	38,9	255
Подпись 14	35,79	28,16	0	12,81	39,7	256
Подпись 15	33,29	28,42	0	18,34	37,9	256
Подпись 16	33,16	28,03	0	9,32	38,1	256
Подпись 17	33,42	25,66	0	14,81	37,8	254
Подпись 18	41,05	35,92	0	10,76	38,9	263
Подпись 19	38,55	25,79	0	12,26	37,7	260
Подпись 20	58,82	53,29	0	16,37	36,6	253
Подпись 21	45,92	35,92	0	11,16	31,3	246
Подпись 22	35,92	30,79	0	7,08	38,7	254
Подпись 23	43,68	41,05	0	31,42	38,8	250
Подпись 24	40,92	28,16	0	6,13	39,1	254
Подпись 25	38,68	28,29	0	11,31	39,2	253

Продолжение таблицы 8.2

Подпись	Ошибка классификации, обучающая выборка, до преобразований, %	Ошибка классификации, тестовая выборка, до преобразований, %	Ошибка классификации, обучающая выборка, после преобразований, %	Ошибка классификации, тестовая выборка, после преобразований, %	Количество отобранных признаков	Время, с
Подпись 26	35,92	38,68	0	14,89	42,6	256
Подпись 27	38,68	35,92	0	22,97	35,4	245
Подпись 28	33,42	23,29	0	8,68	36,4	246
Подпись 29	41,18	46,18	0	14,81	40,1	253
Подпись 30	43,68	40,79	0	11,68	33,4	242
Подпись 31	36,18	35,53	0	9,76	38,2	249
Подпись 32	49,08	30,53	0	17,94	36,0	244
Подпись 33	36,05	28,42	0	5,18	36,5	249
Подпись 34	43,55	35,66	0	16,31	35,3	245
Подпись 35	38,55	30,39	0	12,31	41,1	256
Подпись 36	33,16	43,55	0	17,42	36,5	247
Подпись 37	46,18	38,68	0	23,13	37,7	248
Подпись 38	38,29	51,32	0	15,71	38,6	252
Подпись 39	41,05	33,29	0	8,79	41,4	254
Подпись 40	35,92	30,66	0	19,95	37,5	248
Ср. знач.	40,15	34,74	0	14,86	37,87	252

В результате в обоих наборах данных после проведения отбора признаков и оптимизации параметров классификатора ошибка на обучающих выборках стала равна нулю, а на тестовых выборках уменьшилась более чем в 2 раза.

Проведено сравнение классификатора, построенного с использованием алгоритма «Разряд молнии» с классификаторами на основе алгоритмов-аналогов. В качестве аналогов взяты следующие алгоритмы:

- «Мозговой штурм» с элементами дифференциальной эволюции для формирования базы нечетких правил классификатора (CBSODE);

- бинарный алгоритм «Мозговой штурм» с S-образной функцией преобразования (DBSOS);

- непрерывный алгоритмом «Мозговой штурм» с элементами дифференциальной эволюции (BSODE);

- FC – классификатор, построенный на основе алгоритма горной кластеризации с оптимизацией параметров методом «Кукушкин поиск» (FC).

Сравнение точности классификации алгоритмов приведено в таблице 8.3.

Таблица 8.3 – Сравнение алгоритма «Разряд молнии» с аналогами

Алгоритм классификации	Точность классификации на обучающей выборке, %	Точность классификации на тестовой выборке, %	Количество отобранных признаков
«Разряд молнии», 1 задача	100	84,16	37,33
«Разряд молнии», 2 задача	100	85,14	37,87
CBSODE	66,38	62,25	100
DBSOS	78,61	75,01	16,09
BSODE	89,80	85,65	100
FC	-	85,30	100

Из полученных результатов следует, что алгоритм «Разряд молнии» по точности классификации превосходит алгоритмы CBSODE и DBSOS и дает сравнимые результаты с алгоритмами BSODE и FC-классификатором. Таким образом, алгоритмы «Разряд молнии в непрерывном и бинарном пространствах, примененные для оптимизации параметров и отбора признаков, показали свою эффективность при построении классификатора для аутентификации пользователя по рукописной подписи.

9 Набор данных Malicious and Benign Websites

9.1 Описание набора данных Malicious and Benign Websites

Вредоносные веб-сайты являются серьезной проблемой, потому что затруднительно анализировать один за другим и заносить в черный список каждый URL-адрес. Набор данных Malicious and Benign Websites является результатом проекта [32], который состоял в составлении моделей для классификации и определения вредоносных и безопасных веб-сайтов по данным от сайта на уровне приложений и по характеристикам сетевого соединения. Были рассмотрены обычные и вредоносные URL-адреса из различных источников, все они были проверены. Для сбора данных была использована активная система обнаружения опасных серверов (ACOOC от англ. client honeypot) для получения сетевого трафика, а также некоторые другие инструменты для получения дополнительной информации, например, для определения страны, где находится сервер, использовался WHOIS.

Описание данных:

- URL - анонимный идентификатор проанализированного URL-адреса;

- URL_LENGTH - количество символов в URL;

- NUMBER_SPECIAL_CHARACTERS - количество специальных символов в URL, таких как «/», «%», «#», «&», «.», «=»;

- CHARSET - категориальный признак, обозначающий стандарт кодировки символов (также называемый набором символов);

- SERVER - категориальный признак, обозначающий операционную систему сервера, полученную из ответного пакета;

- CONTENT_LENGTH - размер содержимого HTTP-заголовка;

- WHOIS_COUNTRY - категориальный признак, обозначает страны, которые были получены в ответах серверов (для получения данного признака использовался API Whois);

- WHOIS_STATEPRO - категориальный признак, обозначает регионы, которые были получены в ответах серверов при помощи API Whois;

- WHOIS_REGDATE - Whois предоставляет дату регистрации сервера, поэтому эта переменная имеет значения даты;

- WHOIS_UPDATED_DATE - через Whois получена последняя дата обновления с анализируемого сервера;

- TCP_CONVERSATION_EXCHANGE - количество TCP-пакетов, которыми

обменивались сервер и honeypot-клиент;

- DIST_REMOTE_TCP_PORT - это количество обнаруженных портов, отличных от TCP;

- REMOTE_IPS - общее количество IP-адресов, подключенных к honeypot-клиенту;

- APP_BYTES - количество переданных байтов;

- SOURCE_APP_PACKETS - пакеты, отправленные из honeypot-клиента на сервер;

- REMOTE_APP_PACKETS - пакеты, полученные от сервера;

- APP_PACKETS - общее количество IP-пакетов, сгенерированных во время взаимодействия между honeypot-клиентом и сервером;

- DNS_QUERY_TIMES - количество DNS-пакетов, сгенерированных во время взаимодействия между honeypot-клиентом и сервером;

- TYPE – метка класса, категориальный признак, обозначающий тип анализируемой веб-страницы: 1 для вредоносных веб-сайтов и 0 для безопасных веб-сайтов.

В данном наборе некоторые признаки имеют нечисловой формат и представляют собой строки или даты. При этом оптимизация параметров классификатора основана на выполнении математических операций, поэтому нечисловые признаки необходимо было привести к числовому виду. Для категориальных признаков, имеющих строковый формат, каждому значению строки было поставлено в соответствие число, причем одинаковые строки были заменены одинаковыми числами. Признаки в формате даты и времени преобразованы к слепку времени, то есть количеству секунд с 00:00:00 часов 1 января 1970г.

9.2 Описание эксперимента

Эксперимент проводился по схеме десятикратной кросс-валидации. Сначала классификатор строился без отбора признаков и без оптимизации параметров, чтобы получить ошибку классификации до преобразований. Далее была найдена ошибку классификации после оптимизации параметров и после проведения отбора признаков совместно с оптимизацией параметров при помощи алгоритма «Разряд молнии».

Параметры алгоритма «Разряд молнии»:

- размер популяции 40 частиц;
- количество итераций 300;
- максимальное время канала 5;
- вероятность расщепления 0,5;
- $\mu = 0$;

- σ = 1.

Эксперимент проводился на персональном компьютере со следующими параметрами:

- процессор Intel Core i7-4510 CPU 3,40 GHz;
- установленная ОЗУ 8 Гб;
- ОС Windows 10;
- отсутствие подключения к сети Интернет.

9.3 Результаты эксперимента

В таблице 9.1 приведены результаты эксперимента: ошибка, усредненная по 5 прогонам, а также лучшее из полученных значений.

Таблица 9.1 – Результаты эксперимента

	Только оптимизация, ср. знач.	Только оптимизация, лучшее знач.	Оптимизация и отбор, ср. знач.	Оптимизация и отбор, лучшее знач.
Ошибка на обучающей выборке до преобразований, %	69,56	69,56	69,56	69,56
Ошибка на тестовой выборке до преобразований, %	69,2	69,2	69,2	69,2
Ошибка на обучающей выборке после преобразований, %	8,22	7,96	10,67	10,59
Ошибка на тестовой выборке после преобразований, %	8,72	8,26	10,97	10,23
Количество отобранных признаков	20	20	1,9	1,7
Время, с	359,76	357,08	555,81	551,64

Проведено сравнение классификатора, построенного с использованием алгоритма «Разряд молнии» для оптимизации параметров, с классификаторами на основе алгоритмов-аналогов. В качестве аналогов взяты классификаторы на основе следующих алгоритмов:

- бинарный алгоритм ласточек и непрерывный алгоритма ласточек (SSOD SSO);
- бинарный алгоритм ласточек и алгоритм, основанный на модели островов (SSOD

SSOI);

- алгоритм случайного леса (RF);
- стохастический градиентный спуск (SGD);
- алгоритм k-ближайших соседей (KNN);
- ансамблевый метод простого голосования (SVE).

Сравнение точности классификации алгоритмов приведено в таблице 9.2.

Таблица 9.2 – Сравнение алгоритма «Разряд молнии» при построении классификатора для набора данных Malicious and Benign Websites

Классификатор	Количество отобранных признаков	Точность классификации на обучающей выборке, %	Точность классификации на тестовой выборке, %
Разряд молнии	19	91,28	91,78
SSOD SSO	4	90,51	90,32
SSOD SSOI	4	90,78	90,67
RF	18	100	95,51
SGD	18	95,66	95,83
KNN	18	95,58	95,71
SVE	18	99,79	99,84

Алгоритм «Разряд молнии» показал точность классификации ниже, чем классификаторы на основе алгоритмов RF, SGD, KNN, SVE. При сравнении точности классификаторов, построенных алгоритмами SSOD SSO и SSOD SSOI, алгоритм «Разряд молнии» показал сравнимую точность. Следовательно, классификатор с оптимизацией параметров алгоритмом «Разряд молнии» может быть применен для распознания вредоносных веб-сайтов.

10 Вопросы охраны труда

10.1 Описание рабочего места

Производственная среда при выполнении работы представляет собой рабочее помещение, в котором находится рабочее место. Рабочее место представляет собой письменный стол, снабжённый персональным компьютером, и стулом. Освещение в рабочем помещении общее рассеянное и естественное боковое – освещение помещения через световой проём в наружной стене.

Для данной производственной среды выделены следующие вредные факторы:

- несоответствующий для физиологии человека микроклимат;
- повышенный уровень шума на рабочем месте;
- недостаток освещения на рабочем месте;
- низкая эргономичность рабочего места.

К психофизическим опасным и вредным производственным факторам относятся нервно-психические нагрузки:

- общее утомление (утомление плечевого пояса и рук, туловища и ног, усталость глаз);
- умственное напряжение;
- монотонность труда;
- эмоциональные перегрузки.

Безопасность при выполнении работы оценивалась путем определения соответствия вредных факторов установленным нормам, регламентированным санитарно-эпидемиологическими правилами и нормативами СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» [33].

10.2 Уровень шума в рабочем помещении

Повышенный уровень шума является вредным фактором, который может привести к нарушению слуха, сердечно-сосудистым заболеваниям, понижать тонус, иммунитет и т.д.

Согласно [33], для рабочего места при научной деятельности и программировании допустимый уровень шума для не должен превышать 50 дБА.

Источником шума на рабочем месте является персональный компьютер. Уровень шума системы охлаждения персонального компьютера составляет в 40 дБА при повышенной нагрузке (рисунок 10.1). Поскольку уровень шума меньше 50 дБА, то рабочее место безопасно и соответствует требованиям по уровню шума.

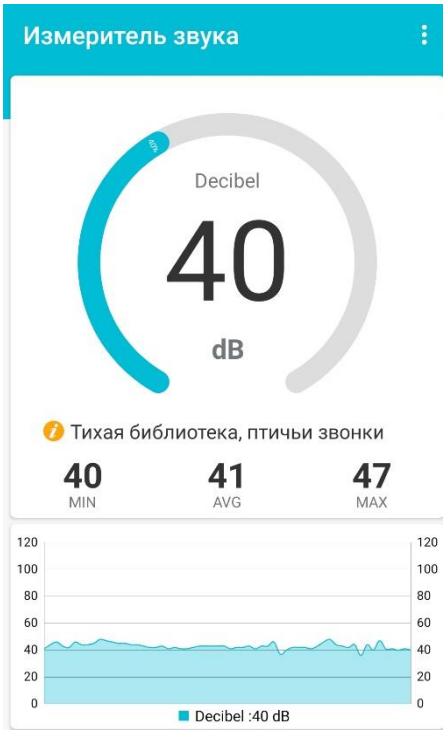

Рисунок 10.1 – Показания измерений шума персонального компьютера

10.3 Микроклимат в рабочем помещении

Микроклимат в помещении определяется температурой, влажностью и скоростью движения воздуха. Вредные факторы: пониженная или повышенная температура, высокая или низкая влажность, высокая скорость движения воздуха.

Работа с использованием ПЭВМ является основным видом деятельности и связана с нервно-эмоциональным напряжением, поэтому должны обеспечиваться оптимальные параметры микроклимата для категории работ 1а (сидячая работа, не требующая высокого физического напряжения) в соответствии с СанПин 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений» [34].

Для определения безопасности помещения проведено измерение показателей микроклимата (рисунок 10.2) и сравнение их с допустимыми значениями для теплого периода года и категории работ 1а. Показатели микроклимата приведены в таблице 10.1.

Рисунок 10.2 – Показания термометра

Таблица 10.1 – Показатели микроклимата

	Температура воздуха, °С	Скорость движения воздуха, м/с	Относительная влажность воздуха, %
Нормы	23-25	0,1	60-40
Показатели	24	менее 0,1	56

Все измеренные показатели находятся в допустимых пределах, поэтому температура, влажность и скорость движения воздуха соответствуют нормам, то есть условия на рабочем месте соответствуют оптимальным величинам показателей микроклимата.

10.4 Освещение в рабочем помещении

Рабочий стол размещен таким образом, что монитор компьютера был ориентирован боковой стороной к окну, естественный свет при этом падал слева. Согласно [33] освещенность на поверхности стола должна быть 300 - 500 лк. Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк.

Произведены измерения освещенности на поверхности рабочего стола. Измеренное значение освещенности составило 80 лк (рисунок 10.3). Данное значение освещенности не соответствует требованиям освещенности рабочего места, требуется дополнительное освещение, рекомендуется установить местный источник освещения.

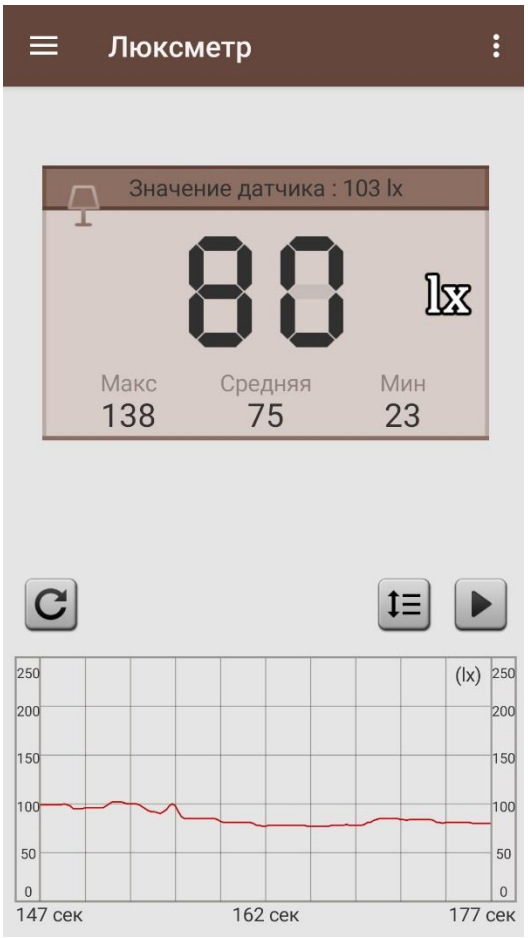

Рисунок 10.3 – Показания прибора люксметр

10.5 Эргономичность рабочего места

Недостаток эргономичности является вредным фактором. Требования к организации и оборудованию рабочих мест ПЭВМ для взрослых пользователей представлены в [33]. Значения эргономики рабочего места приведены в таблице 10.2.

В основном, значения эргономики рабочего места соответствуют нормам, поэтому рабочее место можно считать эргономичным. Рекомендуется заменить рабочий стул на более эргономичное рабочее кресло, которое должно быть подъемно-поворотным, регулируемым по высоте и углам наклона сиденья и спинки, а также соответствовать нормам, приведенным в [33].

Таблица 10.2 – Значения эргономики рабочего места

Показатель	Норма	Значение
1. Экран монитора от глаз пользователя	600 – 700 мм и >500 мм	600 мм
1. Высота рабочей поверхности стола	680 – 800 мм	770 мм
3. Ширина поверхности стола	800, 1000, 1200, 1400 мм	1000 мм
4. Глубина поверхности	800, 1000 мм	800 мм
5. Высота пространства для ног	> 600 мм	750 мм
6. Ширина пространства для ног	> 500 мм	800 мм
7. Глубина пространства для ног (на уровне колен)	> 450 мм	800 мм
8. Глубина пространства для ног (на уровне вытянутых ног)	> 650 мм	750 мм
9. Общая площадь рабочего места	> 6 $м^2$	12 $м^2$
10. Ширина сидения стула	>400 мм	460 мм
11. Глубина сидения стула	>400 мм	420 мм
12. Высота сидения стула	400-550 мм	430 мм
13. Высота спинки стула	300 ± 20 мм	500 мм
14. Ширина спинки стула	> 380 мм	310 мм

11 Технико-экономическое обоснование работы

11.1 Обоснование целесообразности работы

В последние годы отмечается возросший интерес к задачам классификации, как одной из важных задач анализа данных. Классификация применяется в различных сферах жизни: в науке, медицине, в экономике и банковском деле, в технических областях, в том числе бурно развивающихся информационных технологиях. Результаты работы могут быть использованы для классификации данных при создании программных средств обеспечения информационной безопасности. Разработанные алгоритмы позволяют повысить точность и эффективность классификации.

11.2 Организация и планирование работ

Выполнение работы осуществляли:

- исполнитель (студент, далее И);
- руководитель (далее Р).

Исполнитель выполнял поставленные задачи. Руководитель контролировал правильность выполнения поставленных задач, а также осуществлял контроль соблюдения сроков работы, проводил консультирование, задавал вектор развития работы, рецензировал выполненные этапы работы. На выполнение работы исполнителю отводится 36 рабочих дней по 6 часов в день, руководителю на руководство и консультирование – 24 часа.

В таблице 11.1 приведён график трудоёмкости выполнения работ.

Таблица 11.1 – Перечень работ и оценка их трудоёмкости

Наименование этапов и содержание работ	Исполнитель (должность)	Трудоемкость		Количество исполнителей, чел.	Стоимость одного часа работ, руб/ч	Общая стоимость работы, руб.	Продолжительность	Срок исполнения, дни
		Нормо-часы, н-ч	Процент от общей трудоемкости, %					
1 Постановка задачи	Р	2	2,50	1	572	1144,00	6	0,33
	И	4		1	66,67	266,67	6	0,67

Продолжение таблицы 11.1

Наименование этапов и содержание работ	Исполнитель (должность)	Трудоемкость		Количество исполнителей, чел.	Стоимость одного часа работ, руб/ч	Общая стоимость работы, руб.	Продолжительность	Срок исполнения, дни
		Нормо-часы, н-ч	Процент от общей трудоемкости, %					
2. Составление и утверждение задания	Р	2	2,50	1	572	1144,00	6	0,33
	И	4		1	66,67	266,67	6	0,67
3. Обзор литературы	Р	2	9,17	1	572	1144,00	6	0,33
	И	20		1	66,67	1333,33	6	3,33
4. Разработка и реализация алгоритма «Разряд молнии»	Р	2	15,83		572	1144,00	6	0,33
	И	36		1	66,67	2400,00	6	6,00
5. Применение алгоритма «Разряд молнии» при минимизации функций	Р	2	8,33		572	1144,00	6	0,33
	И	18		1	66,67	1200,00	6	3,00
6. Применение алгоритма «Разряд молнии» для оптимизации параметров нечётких классификаторов	Р	3	10,42		572	1716,00	6	0,50
	И	22		1	66,67	1466,67	6	3,67
7. Бинаризация алгоритма «Разряд молнии»	Р	3	14,58	1	572	1716,00	6	0,50
	И	32		1	66,67	2133,33	6	5,33
8. Сравнение бинарного алгоритма «Разряд молнии» с аналогами	Р	2	9,17	1	572	1144,00	6	0,33
	И	20		1	66,67	1333,33	6	3,33
9. Проведение эксперимента и построение нечетких классификаторов на наборах данных KDD Cup 1999 Data, SVC2004, Malicious and Benign Websites	Р	2	15,83	1	572	1144,00	6	0,33
	И	36		1	66,67	2400,00	6	6,00
10. Разработка технико-экономического обоснования	Р	2	4,17	1	572	1144,00	6	0,33
	И	8		1	66,67	533,33	6	1,33
11. Разработка документации по безопасности жизнедеятельности	И	8	3,33	1	66,67	533,33	6	1,33
12. Оформление пояснительной записки	Р	2	4,17	1	572	1144,00	6	0,33
	И	8		1	66,67	533,33	6	1,33
Всего: 12	Р	24	100,00			13728		4
	И	216				14400		36

В соответствии с таблицей 11.1 была составлена диаграмма Ганта, которая приведена в приложении А.

11.3 Смета затрат

Смета затрат включает:

- затраты на оборудование;
- затраты на оплату труда и страховые взносы;
- затраты на основные и вспомогательные материалы;
- затраты на электроэнергию;
- накладные расходы.

11.3.1 Затраты на оборудование

Для программной реализации алгоритмов и проведения экспериментов был использован персональный компьютер, стоимостью 60 000 рублей на момент покупки.

Стоимость компьютера менее 100 000 рублей, поэтому, согласно п. 1 ст. 256 НК РФ [35], оборудование не признается амортизируемым имуществом и его стоимость списывается полностью. Затраты на оборудование представлены в таблице 11.2.

Таблица 11.2 – Затраты на оборудование

Оборудование	Цена за единицу оборудования, р.	Кол-во, шт.	Стоимость (кол-во*цена)
Персональный компьютер	60 000	1	60 000
		Итого:	60 000

Итого на оборудование было потрачено 60 000 рублей.

11.3.2 Затраты на оплату труда и страховые взносы

Для расчета данной статьи рассчитан фонд оплаты труда по формуле:

$$ФОТ = О_{зп} + Д_{зп},$$

где $О_{зп}$ – основная заработная плата, которая рассчитывается по формуле;

$Д_{зп}$ – дополнительная заработная плата.

$$О_{зп} = ЗП_{пр} + Премия + РК,$$

где $ЗП_{пр}$ – прямая заработная плата в рублях;

РК – районный коэффициент, который для Томской области составляет 1,3.

Для расчета оплаты труда по экономической части работы необходимо

руководствоваться нормами почасовой оплаты труда, установленными в ТУСУР.

Исполнитель – студент, не имеющий высшего образования, может быть принят на работу техником. Ставка техника составляет 14400 руб. согласно приказу ректора ТУСУР № 759 от 11.09.2019 г. Почасовая ставка руководителя на основании приказа ректора ТУСУР № 1178 от 28.12.2019 составила 572 рубля для профессора, доктора наук.

Дополнительная заработная плата (ДЗП) составляет 15% от основной. К дополнительной заработной плате относятся выплаты за непроработанное время: оплата времени отпусков, перерывов в работе, установленных действующим законодательством для отдельных категорий работников, и т.п. Выплата премий и дополнительной заработной платы не предусмотрена, так как вся работа была выполнена в рабочие сроки, согласно графику.

Прямую заработную плату для руководителя можно вычислить по формуле:

$$ЗП_{пр} = Ставка \cdot n,$$

где Ставка – ставка почасовой оплаты;

n – количество часов, отведенных для работы.

По формуле рассчитан оклад руководителя:

$$ЗП_{пр(рук)} = 572 \frac{руб}{ч} \cdot 24\ ч = 13728\ руб$$

С 1 января 2017 г. страховые взносы взимаются на основании гл. 34 Налогового кодекса РФ (НК РФ) [35].

В 2020 г. общий размер взносов составляет 30,17 % от дохода сотрудника.

Тариф страхвзносов в ПФР — 22 %.

Тариф по взносам на ОМС — 5,1 %.

Тариф страховых взносов на ВНиМ — 2,9 %.

Страхование от несчастного случая — 0,17 %.

Таким образом, страховые взносы можно рассчитать, используя формулу:

$$СтраховыеВзносы = ФОТ \cdot 0{,}3017.$$

Рассчитана основная заработная плата, фонд оплаты труда и страховые взносы для руководителя. Премия и дополнительная заработная плата руководителю не выплачивались, поэтому при расчетах данные значения будут равны 0. Районный коэффициент для Томской области составляет 1,3 или 30% от прямой заработной платы:

$$О_{зп(рук)} = 13728 + 0 + 13728 \cdot 0{,}3 = 17846{,}40\ руб;$$

$$ФОТ_{(рук)} = 17846{,}40 + 0 = 17846{,}40\ руб;$$

$$СтраховыеВзносы_{(рук)} = 17846{,}40 \cdot 0{,}3017 = 5384{,}26\ руб;$$

Рассчитана основная заработная плата, фонд оплаты труда и страховые взносы для исполнителя за полное время работы (36 рабочих дней или 1,5 месяца). Премия и дополнительная заработная плата исполнителю не выплачивались, поэтому при расчетах данные значения будут равны 0. Районный коэффициент для Томской области составляет 1,3 или 30% от прямой заработной платы:

$$О_{зп(исп)} = (14400 + 0 + 14400 \cdot 0{,}3) \cdot 1{,}5 = 28080 \text{ руб};$$

$$ФОТ_{(исп)} = 28080 + 0 = 28080 \text{ руб};$$

$$СтраховыеВзносы_{(исп)} = 28080 \cdot 0{,}3017 = 8471{,}74 \text{ руб};$$

Все данные внесены в таблицу 11.3.

Таблица 11.3 – Затраты на оплату труда и страховые взносы

Участник	$ЗП_{пр}$, руб.	Премия, руб.	РК, руб	$О_{зп}$	$Д_{зп}$	ФОТ	Страховые взносы	Всего
1. Исполнитель	21600	0	6480	28080	0	28080	8471,74	36551,74
2. Руководитель	13728	0	4118,40	17846,40	0	17846,40	5384,26	23230,66
Итого:	35328	0	10598,4	45926,4	0	45926,4	13856	59782,40

Таким образом, суммарные расходы на оплату труда участникам работы составили 59782,40 рублей.

11.3.3 Затраты на основные и вспомогательные материалы

Данная статья расходов включает расходы по приобретению и доставке основных и вспомогательных материалов, необходимых для выполнения работы: материалы для изготовления образцов и макетов, и материалы, необходимые для оформления требуемой документации (затраты на бумагу, картридж для принтера, CD-диск и т.п.).

Расчёты затрат на основные и вспомогательные материалы представлены в таблице 11.4.

Таблица 11.4 – Затраты на основные и вспомогательные материалы

Наименование материала	Количество	Цена за единицу, руб.	Сумма, руб.
CD-диск	1	100	100
Конверт для диска	1	20	20
		Итого:	120

Затраты на основные и вспомогательные материалы составили 120 рублей.

11.3.4 Затраты на электроэнергию

Данная статья затрат включает в себя затраты по электроэнергии на технологические нужды. Основными источниками потребления электроэнергии являются работа компьютера, принтера и освещение.

Затраты на электроэнергию рассчитываются по формуле:

$$C_{эл} = W_y \cdot T_g \cdot S_{эл},$$

где W_y – установленная мощность (кВт);

T_g – время работы оборудования (час);

$S_{эл}$ – тариф на электроэнергию (руб./кВт·ч).

Согласно Приказу № 6-585 от 11.12.2019 «О тарифах на электрическую энергию для населения и приравненных к нему категорий потребителей Томской области на 2020 год» [36] тариф для населения, проживающего в городских населенных пунктах в домах, оборудованных стационарными электроплитами и (или) электроотопительными установками, и приравненных к ним (тариф указывается с учетом НДС) составляет 2,45 руб./кВт·ч.

Мощность персонального компьютера, использованного в работе, составляет 100 Вт./ч. Для освещения использовалась одна светодиодная лампа мощностью 11 Вт/ч. Мощность принтера – 10 Вт/ч.

Для персонального компьютера затраты на электроэнергию составили:

$$C_{эл(к)} = 0{,}1\,\frac{кВт}{ч} \cdot 216ч \cdot 2{,}45\,\frac{руб}{кВт} \cdot ч = 52{,}92\ руб;$$

Затраты на электроэнергию для лампы составили:

$$C_{эл(л)} = 0{,}011\,\frac{кВт}{ч} \cdot 216ч \cdot 2{,}45\,\frac{руб}{кВт} \cdot ч = 5{,}82\ руб;$$

Расчёты затрат также представлены в таблице 11.5.

Время работы компьютера и освещения рабочего места составляет 100% от времени, затраченного исполнителем на выполнение данной работы, то есть 216 часов.

Таблица 11.5 – Затраты на электроэнергию

Наименование оборудования	Количество, шт.	Потребляемая мощность, кВт	Время работы, ч	Тариф, руб за кВт/ч	Сумма затрат, р
Персональный компьютер	1	0,1	216	2,45	52,92
Светодиодная лампа	1	0,011	216	2,45	5,82
				Итого:	58,74

Суммарные затраты на электроэнергию составили 58,74 рублей.

11.3.5 Накладные расходы

Накладные расходы – расходы на управление, хозяйственное обслуживание при выполнении работы, в том числе расходы на транспорт и доступ в сеть Интернет при выполнении работы. Абонентская плата за месяц доступа в сеть Интернет от провайдера РосТелеком составляет 500 рублей. Затраты на накладные расходы представлены в таблице 11.6.

Таблица 11.6 – Накладные расходы

Наименование	Количество	Цена за 1 ед., руб	Сумма затрат, руб
Печать пояснительной записки на черно-белом лазерном принтере	100 шт	2,00	200,00
Брошюрование до 20 мм (до 165 л) вместе с расходными материалами	1 шт	50,00	50,00
Доступ в Интернет	1,5 месяца	500,00	750,00
		Итого:	1000,00

Суммарно накладные расходы составили 1000,00 рублей.

11.3.6 Сводная смета затрат

На основании всех проведенных расчётов составлена сводная смета затрат на выполнение работы. Сводная смета представлена в таблице 11.7.

Таблица 11.7 – Сводная смета затрат

Статья затрат	Сумма затрат, руб
Затраты на оборудование	60000,00
Затраты на оплату труда со страховыми взносами	59782,40
Затраты на основные и вспомогательные материалы	120,00
Затраты на электроэнергию	58,74
Накладные расходы	1000,00
Итого:	120961,14

На основе сводной сметы затрат была построена круговая диаграмма (рисунок 11.1), отражающая вклад каждой статьи расходов в суммарные затраты на выполнение работы.

Около 99% расходов ушло на две статьи расходов: затраты на оборудование (49,60%) и затраты на оплату труда исполнителю и руководителю (49,42%).

Таким образом, в процессе технико-экономического обоснования работы проведено планирование этапов реализации работы, проведён расчёт сметы затрат, в которую вошли

затраты на оборудование, оплату труда и страховые взносы, основные и вспомогательные материалы, затраты на электроэнергию и накладные расходы. Общая сумма всех затрат на выполнение работы составила 120961,14 рублей. Основными затратными статьями являются затраты на оборудование (49,60%) и затраты на оплату труда исполнителю и руководителю (49,42%). Остальные статьи расходов составляют в сумме около 1% от общей суммы затрат.

Рисунок 11.1 – Общие затраты на выполнение работы

12 Заключение

Целью работы являлось усовершенствование классификации данных с помощью метаэвристического алгоритма «Разряд молнии».

Для достижения описанной цели проведен аналитический обзор текущего состояния исследований, показавший актуальность поставленной задачи. В результате работы разработаны непрерывный и бинарный алгоритмы на основе метаэвристики «Разряд молнии». Данные алгоритмы реализованы в виде программных средств. Разработка производилась с использованием языка программирования Python.

Предложено применение метаэвристики «Разряд молнии» в задачах минимизации функций. Разработанные алгоритмы применены для оптимизации параметров классификатора и отбора его признаков. Проведены эксперименты, подтверждающие эффективность разработанных алгоритмов с использованием наборов реальных данных из репозитория KEEL. Построены классификаторы для наборов данных KDD Cup 1999 Data, SVC 2004 и Malicious and Benign Websites.

Разработанные алгоритмы позволили уменьшить ошибку классификации и ее вычислительную сложность, увеличить интерпретируемость классификатора. Разработанные алгоритмы показали свою эффективность при построении классификаторов для анализа сетевых атак, определения подлинности рукописной подписи, распознания вредоносных веб-сайтов и могут быть применены при создании программных средств обеспечения информационной безопасности.

Результаты работы были представлены на XXV Международной научно-технической конференции студентов, аспирантов и молодых учёных «Научная сессия ТУСУР – 2020» [37], которая состоялась в ТУСУРе 25 – 27 мая 2020 г.

Список использованных источников

1 Shareef, H. (2015). Lightning search algorithm. H. Shareef, A.A. Ibrahim, A.H. Mutlag. In: Applied Soft Computing Journal, Vol. 36, pp. 315–333.

2 Черезов Д.С. Обзор основных методов классификации и кластеризации данных / Д.С. Черезов, Н.А. Тюкачев // Вестник ВГУ, серия системный анализ и информационные технологии. - 2009. - Т. 1. - С. 25–29.

3 Mekh, M.A. (2017). Comparative Analysis of Differential Evolution Methods to Optimize Parameters of Fuzzy Classifiers. M.A. Mekh, I.A. Hodashinsky. In: Journal of Computer and Systems Sciences International, Vol. 56, Issue 4, pp. 616–626.

4 Лекции по искусственным нейронным сетям [Электронный ресурс] – Режим доступа: http://www.machinelearning.ru/wiki/images/c/cc/Voron-ML-NeuralNets.pdf (дата обращения: 22.06.2020).

5 Машина опорных векторов [Электронный ресурс] – Режим доступа: http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf (дата обращения: 22.06.2020).

6 Байесовский классификатор [Электронный ресурс] – Режим доступа: http://www.machinelearning.ru/wiki/index.php?title=Байесовский_классификатор (дата обращения: 22.06.2020).

7 Ходашинский И.А. Построение нечеткого классификатора на основе методов гармонического поиска / И.А. Ходашинский, М.А. Мех. // Программирование. - 2017. - № 1. – С. 54-65.

8 Bernal, E. (2020). Fuzzy galactic swarm optimization with dynamic adjustment of parameters based on fuzzy logic. E. Bernal1, O. Castillo, J. Soria, F. Valdez. In: SN Computer Science, Vol. 1, pp. 1–19.

9 Vieira, M.S. (2007). Ant Colony Optimization Applied to Feature Selection in Fuzzy Classifiers. S.M. Vieira, J.M.C. Sousa, T.A. Runkler. P. Melin et al. In: IFSA, pp. 778–788.

10 Ходашинский И.А. Алгоритмы «Стадо криля» и кусочно-линейной инициализации для построения систем типа Такаги - Сугено / И.А. Ходашинский, И.В. Филимоненко, К.С. Сарин. // Автометрия. - 2017. - Т. 53, № 4. – С. 84-94.

11 Beloufa, F. (2013). Design of fuzzy classifier for diabetes disease using Modified Artificial Bee Colony algorithm. F. Beloufa, M.A. Chikh. In: Elsevier Ireland, Computer methods and programs in biomedicine, Vol. 112, Issue 1, pp. 92–103.

12 Kim, M.H. (2005). Design of T–S Fuzzy Classifier via Linear Matrix Inequality Approach. M.H. Kim, Y.H. Joo, J.B. Park, H.J. Lee. In: Lecture Notes in Artificial Intelligence, Vol.

3613, pp. 406-415.

13 Oh, S.-K. (2012). Design of optimized cascade fuzzy controller based on differential evolution: Simulation studies and practical insights. S.-K. Oh, W.-D. Kima, W. Pedrycz. In: Engineering Applications of Artificial Intelligence, No. 25, pp. 520–532.

14 Elragal, H.M. (2010). Using Swarm Intelligence for Improving Accuracy of Fuzzy Classifiers. In: World Academy of Science, Engineering and Technology International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering, Vol. 4, No. 8, pp. 1135-1142.

15 Marinak, M. (2014). Fuzzy control optimized by a Multi-Objective Differential Evolution algorithm for vibration suppression of smart structures. M. Marinaki, Y. Marinakis, G.E. Stavroulakis. In: Civil-Comp Ltd and Elsevier Ltd., Computers and Structures, No. 147, pp. 126–137.

16 Eftekhari, M. (2008). Eliciting transparent fuzzy model using differential evolution. M. Eftekhari, S.D. Katebi, M. Karimi, A.H. Jahanmiri. In: Elsevier, Applied Soft Computing, No.8, pp. 466–476.

17 Parvaresh, A. (2012). Fault Detection and Diagnosis in HVAC System Based on Soft Computing Approach. A., A. Hasanzade, S. M. A. Mohammadi, A. Gharaveisi. In: International Journal of Soft Computing and Engineering (IJSCE), Vol. 2, No. 3, pp. 114-120.

18 Звонков, В. Б. Сравнительное исследование классических методов оптимизации и генетических алгоритмов / В. Б. Звонков, А. М. Попов // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. – 2013. – № 4 (50) – С. 23-27.

19 Матренин, П.В. Методы стохастической оптимизации: учеб. пособие / П.В. Матренин, М.Г. Гриф, В.Г. Секаев. – Новосибирск: Изд-во НГТУ, 2016. – 67 с.

20 Fred Glover and Kenneth Sörensen. (2015). Metaheuristics. In: Scholarpedia, Vol. 10, No.4.

21 Nayak, N. (2015). Fuzzy C-Means (FCM) Clustering. J. Nayak, B. Naik, H.S. Behera. L.C. Jain. In: Computational Intelligence in Data Mining, Smart Innovation, Systems and Technologies, Vol. 2, pp. 133-149.

22 Kohavi, R. (1997). Wrappers for feature subset selection. R. Kohavi, G. H. John. In: Artificial Intelligence, Vol. 97, pp. 273-324.

23 Ramirez-Gallego, S. (2018). An information theory-based feature selection framework for big data under Apache Spark. S. Ramirez-Gallego, H. Mourino-Talin, D. Martinez-Rego, V. Bolon-Canedo, J. M. Benitez, A. Alonso-Betanzos, F. Herrera. In: IEEE Transactions on Systems,

Man, and Cybernetics, Vol. 48, No. 9, pp. 1441–1453.

24 Bolon-Canedo, V. (2015). Feature Selection for High-Dimensional Data. V. Bolon-Canedo, N. Sanchez-Marono, A. Alonso-Betanzos. Springer International Publishing.

25 KEEL - Knowledge Extraction based on Evolutionary Learning [Электронный ресурс]. – Режим доступа: http://www.keel.es (дата обращения: 22.06.2020).

26 Yen, J. (1998). Application of Statistical Information Criteria for Optimal Fuzzy Model Construction. J. Yen, L. Wang. In: IEEE Trans. Fuzzy Systems, Vol. 6, No. 3, pp. 362-372.

27 Ходашинский, И.А. Построение компактных и точных нечетких моделей на основе статистических информационных критериев / И.А. Ходашинский // Информатика и системы управления. – 2014. – № 1(39). – С. 99-107.

28 Мех, М.А. Повышение точности вывода нечёткого классификатора алгоритмами дифференциальной эволюции и минного взрыва на наборе данных KDD / М.А. Мех, С.Р. Субханкулова // Научно-практическая конференция студентов «Наука и практика: проектная деятельность – от идеи до внедрения». – 2015.

29 Faraoun, K.M. (2006). Genetic programming approach for multi-category pattern classification applied to network intrusions detection. K. M. Faraoun, A. Boukelif. In: International Journal of Computational Intelligence and Applications, Vol. 6, No 1, pp. 77-99.

30 Sheikhan, M. (2009). Misuse Detection Using Hybrid of Association Rule Mining and Connectionist Modeling. M. Sheikhan, Z. Jadidi. In: World Applied Sciences Journal 7 (Special Issue of Computer & IT), No. 7, pp. 31-37.

31 Porto-D´ıaz, L. (2009). Combining Feature Selection and Local Modelling in the KDD Cup 99 Dataset. L. Porto-D´ıaz, D. Mart´ınez-Rego, A. Alonso-Betanzos, O. Fontenla-Romero. In: Artificial Neural Networks, Vol. 5768, pp. 824-833.

32 Urcuqui, C. (2017). Machine Learning Classifiers to Detect Malicious Websites. A. Navarro, J. Osorio, M. Garcıa. In: CEUR Workshop Proceedings, Vol. 1950, pp. 14-17.

33 Постановление Главного государственного санитарного врача РФ от 3 июня 2003 г. № 118 О введении в действие санитарно-эпидемиологических правил и нормативов СанПиН 2.2.2/2.4.1340-03 [Электронный ресурс] – Режим доступа: http://base.garant.ru/4179328/ (дата обращения: 22.06.2020).

34 Санитарные правила и нормы СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений» [Электронный ресурс] – Режим доступа: http://base.garant.ru/4173106/ (дата обращения: 22.06.2020).

35 Налоговый кодекс Российской Федерации (часть вторая) от 05.08.2000 N 117-ФЗ (ред. от 08.06.2020) [Электронный ресурс] – Режим доступа:

http://www.consultant.ru/document/cons_doc_LAW_28165/ (дата обращения: 22.06.2020).

36 Приказ № 6-585 от 11.12.2019 «О тарифах на электрическую энергию для населения и приравненных к нему категорий потребителей Томской области на 2020 год» [Электронный ресурс] – Режим доступа: https://ensb.tomsk.ru/upload/Приказ%20ДТР%20от%2011.12.2019%20№%206-585(1).pdf (дата обращения: 22.06.2020).

37 Мельникова, Н.Е. Построение нечеткого классификатора на основе метаэвристики «Разряд молнии» // Международная научно-техническая конференция студентов, аспирантов и молодых ученых «Научная сессия ТУСУР – 2020». - 2020.

Приложение А

(обязательное)

Диаграмма Ганта