ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ МЕДИЦИНСКИЙ УНИВЕРСИТЕТ»
МИНИСТЕРСТВА ЗДРАВООХРАНЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Медико-профилактический факультет с отделением биологии
Кафедра медицинской физики с курсом информатики

Серегина Регина Робертовна

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ПРОГНОЗИРОВАНИЯ ПО ПРИЗНАКАМ БАЗЫ ДАННЫХ

Научный руководитель:
доктор физико-математических наук, доцент
заведующий кафедрой медицинской
физики с курсом информатики А.А.Кудрейко

Уфа - 2023

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ МЕДИЦИНСКИЙ УНИВЕРСИТЕТ»
МИНИСТЕРСТВА ЗДРАВООХРАНЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Медико-профилактический факультет с отделением биологии
Кафедра медицинской физики с курсом информатики

Серегина Регина Робертовна

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ПРОГНОЗИРОВАНИЯ ПО ПРИЗНАКАМ БАЗЫ ДАННЫХ

Научный руководитель:
доктор физико-математических наук, доцент
заведующий кафедрой медицинской
физики с курсом информатики А.А.Кудрейко

Уфа - 2023

Оглавление

ВВЕДЕНИЕ

Актуальность проблемы

На протяжении многих лет ученые пытаются разгадать тайну работы мозга и создать устройства, способные принимать решения. Идея создания машин, обладающих подобием человеческого интеллекта, была очень привлекательна. Первоначальные успехи отмечались разработкой компьютеров, играющих в простые игры и доказывающих теоремы. Впоследствии были спроектированы интеллектуальные вычислительные устройства по образу и подобию биологических систем, которые привели к созданию теории нейронных сетей, ставшей одним из самых мощных и полезных подходов к разработке искусственного интеллекта [20].

Сейчас мы живем в веке цифровых технологий [3]. В условиях цифровизации общества, процесс поиска новых знаний, получаемых путем анализа данных, просто необходим. Знания анализируются с различных точек зрения и обобщаются в полезную информацию. Важность извлечения знаний и информации сделало интеллектуальный анализ данных важным фактором, прямо или косвенно влияющим на жизнь человека в различных сферах, в том числе в медицине и биологии. Современные технологии, внедренные в медицинскую диагностику, оказали большое влияние на предварительную постановку диагноза. Подходы глубокого обучения для анализа медицинских баз данных дают возможность разрабатывать удобные инструменты для исследовательского анализа, которые могут быть применены к широкому спектру задач [26]. Такой подход позволяет решать проблемы, с которыми невозможно справиться вручную в силу большого объема информации. При применении к большим данным, машинное обучение иногда может обнаружить тонкие взаимосвязи, которые остаются незамеченными при ручном исследовании. Когда многие такие «слабы» отношения объединяются, они становятся сильными предикторами [4]**.** На данный момент разработано немало экспертных систем обработки данных, построенные на основе ряда закономерностей и правил. Заложенные в программах алгоритмы позволяют диагностировать заболевание по определенной симптоматике, предсказать дальнейшее течение и исход заболевания на основе выбранного метода лечения, а также позволяет рассмотреть причины возникновения патологий. Обнаруженные, технологиями анализа данных (Data Mining), скрытые закономерности позволяют разрабатывать новые методы диагностики. Следовательно разработка системных методов медицинской диагностики является актуальной задачей, которая относится к разделу классификации [56]. Примером может служить работа, в которой описывается

прогнозирование сахарного диабета. Её авторы рассматривают новый подход диагностики сахарного диабета, который основан на классификаторе нейронных сетей. Данный метод позволяет выявлять диабет на ранней стадии, что позволяет сохранить здоровье и жизни людей [60].

В данной выпускной квалификационной работе применяется аналитическая система Orange Data Mining. Данная программа является системой визуального программирования с открытым исходным кодом, который доступен для модификации и свободного распространения. Orange Data Mining был разработан Лабораторией биоинформатики Люблянского университета и институтом Йожефа Стефана (Bioinformatics Laboratory, Faculty of Computer and Information Science, University of Ljubljana, Slovenia). Программа предназначена для интеллектуального анализа данных, статистических исследований и визуализации данных. Orange Data Mining позволяет проводить эксперименты, выбирать систему рекомендаций и прогнозирования моделей. На основании изложенного можно выделить следующие функции Orange Data Mining:

1. Импорт и экспорт данных
2. Статистический анализ
3. Отчетность и аналитика
4. Визуализация данных
5. Интеллектуальный анализ данных
6. Коннекторы для источников данных.

Программа Orange Data Mining широко используется в биомедицине [53], биоинформатике, геномных исследованиях и обучении. В научных исследованиях программа используется как платформа тестирования новых алгоритмов машинного обучения и внедрения новых методов в генетике и биоинформатике. Программа так же нашла применение и в образовательном процессе. Её используют при обучении методам машинного обучения и интеллектуального анализа данных различного происхождения (биология, биомедицина [30], информатика). Для специалистов в области анализа данных, исследователей и ученых, данная аналитическая система является эффективным инструментом.

Благодаря адаптивному характеру машинного обучения, Orange Data Mining хорошо подходит для сценариев, в которых данные постоянно изменяются, свойство запросов и задач нестабильны или же написать код для решения фактически невозможно. Компоненты программы называются виджетами. Интерфейс программы Orange Data Mining предусмотрен для размещения виджетов и создания рабочего процесса анализа данных. Виджеты содержат базовые функциональные возможности, например чтение данных, отображение таблицы

данных и так далее. Каждый виджет представляет собой программный блок, который каким-либо образом обрабатывает поступившую на его вход информацию и передает её дальше, для обработки, визуализации или сохранения следующим виджетом.

В настоящей работе изложены новые методы применения алгоритмов и инструментов интеграции, слияния, предварительной обработки, отображения, анализа и интерпретации сложных биомедицинских данных с целью выявления проверяемых гипотез и построения реалистичных моделей прогнозирования. Используемые биомедицинские данные относятся к области биологических наук, которые включают в себя различные отрасли медицины, одной из которых является кардиология. Кардиология – область науки, занимающаяся изучением широкого спектра проблем, связанных как с нормальным функционированием, так и с патологией сердечно-сосудистой системы человека. С практической точки зрения современная кардиология решает вопросы заболеваний, которые на сегодняшний день занимают ведущее место в инвалидизации и смертности населения планеты. Именно поэтому данная работа основана на клинико-диагностических исследованиях пациентов с риском возникновения сердечной недостаточности.

Сердечная недостаточность – это синдром, при котором нарушена работа сердца. Иными словами, это патологическое состояние, при котором работа сердечно-сосудистой системы не обеспечивает потребностей организма в кислороде сначала при физической нагрузке, а потом и в покое. Проявляется одышкой, слабостью, сердцебиением и повышенной утомляемостью. Это означает, что указанные признаки можно распределить по тем или иным классам и произвести автоматический анализ данных. Имеющийся инструментарий программы Orange Data Mining позволил выявить корреляционные зависимости и определить группы риска пациентов с сердечной недостаточностью. Полученные корреляционные зависимости дают возможность сформулировать рекомендации пациентам. Схожее исследование выполнено в работе по прогнозированию диабета 2 типа, где авторы рассмотрели семь классификаторов глубокой нейронной сети. Результатом исследования взаимосвязи классификаторов и заболевания стало выявление диабета в раннем возрасте и определение необходимого лечения [55].

Целью выпускной квалификационной работы является выявление прогностических возможностей алгоритмов машинного обучения и визуализации данных у групп риска пациентов с сердечной недостаточностью.

Для достижения указанной цели, были поставлены следующие **задачи**:

1. Изучить теоретические основы интеллектуального анализа данных.

2. Рассмотреть существующие методы интеллектуального анализа и изучить имеющуюся базу данных пациентов с сердечной недостаточностью.

3. Оценить эффективность применения методов анализа данных.

Объектом исследования являются диагностические данные о пациентах с сердечной недостаточностью или ее отсутствием. Имеющиеся данные включают 11 диагностических признаков, которые могут оказать влияние на данное заболевание.

Полученные **результаты** работы могут быть использованы при принятии решений в клинической практике врача, определении и корректировке методов лечения заболевания, а также при обучении студентов на занятиях по цифровым технологиям и формировании их роли в практической деятельности врача.

ГЛАВА 1. ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

Стремительно растущий, широкодоступный объем данных делает наше время эпохой цифровых трансформаций, поэтому необходимы мощные и универсальные инструменты для автоматического извлечения ценной информации из огромных объемов данных и преобразования таких данных в организованные знания. Такая необходимость привела к появлению интеллектуального анализа данных [40].

1.1. Интеллектуальный анализ данных

Интеллектуальный анализ данных представляет собой автоматический поиск, в процессе которого выявляются закономерности и тенденции в больших наборах данных. В большинстве случаях, такие закономерности невозможно обнаружить при традиционном просмотре данных, в силу большого объема данных или слишком сложных взаимосвязей [6]. В последующем, мы перейдем к рассмотрению только медицинских данных.

Архивы медицинских данных содержат огромное количество сведений о различных случаях конкретных заболеваний, методах их диагностики. Нахождение закономерностей – одна из задач многих исследований медицинской тематики. Для решения таких задач все чаще применяют методы автоматического анализа данных. В свою очередь, необходимость обработки больших объемов данных в интеллектуальном анализе данных имеет ряд проблем:

1. Эффективность и масштабируемость алгоритмов. Для эффективного извлечения информации из огромного объема данных в базах данных, алгоритмы интеллектуального анализа данных должны быть эффективными и масштабируемыми.

2. Алгоритмы параллельного и распределенного анализа. Огромный размер баз данных, широкое распространение данных и сложность методов интеллектуального анализа данных, мотивируют разработку алгоритмов параллельного и распределенного анализа данных. Алгоритмы делят данные на разделы, которые после обрабатываются параллельно. В последствии результаты разделов объединяются.

3. Обработка сложных типов данных. База данных может содержать сложные объекты данных, объекты мультимедийных данных, временные данные и так далее. Одна система не может получить все эти данные.

4. Сбор информации из разнородных баз данных и глобальных информационных систем. Данные доступны в разных источниках данных в локальной или глобальной сети. Источники данных могут быть структурированными, полу структурированными или

неструктурированными, поэтому извлечение знаний из них создает проблемы для интеллектуального анализа данных [45].

Интеллектуальный анализ данных (Knowledge Discovery in Databases - сокр., KDD) включает в себя:

1.1.1. Автоматическое обнаружение шаблонов

Обнаружение шаблонов – это автоматическое распознавание закономерностей в данных. Метод применяется в статистическом анализе данных, обработке сигналов, анализе изображений, поиске информации, биоинформатике, сжатии данных и машинном обучении. Современные подходы к обнаружению шаблонов включают использование машинного обучения из-за доступности больших данных и большой вычислительной мощности. Во многих отношениях машинное обучение является основным средством, с помощью которого наука о данных проявляется в мире. Машинное обучение – это то, где вычислительные и алгоритмические навыки науки о данных сочетаются со статистическим мышлением науки о данных, и в результате получается набор подходов к выводу и исследованию данных, которые касаются не столько эффективной теории, сколько эффективных вычислений [61].

В машинном обучении обнаружение шаблонов — это присвоение метки заданному входному значению. Интеллектуальный анализ данных осуществляется путем построения моделей. Модель использует алгоритм обработки набора данных. Модель должна быстро и точно распознавать знакомые образы, узнавать и классифицировать незнакомые признаки данных, и автоматически быстро и легко распознавать закономерности [48].

Преимущества автоматического обнаружения шаблонов состоит в том, что распознавание шаблонов решает проблемы классификации и обнаружения поддельных биометрических данных, а также позволяет обнаружить определенные объекты под разными углами.

К недостаткам относится то, что автоматическое обнаружение шаблонов не может объяснить почему тот или иной объект распознается. Обнаружение синтаксических образов - медленный процесс и сложен в реализации, который требует повышенной точности больших наборов данных.

Модели интеллектуального анализа данных можно использовать для анализа данных, на которых они построены, но большинство типов моделей можно обобщить на новые данные.

1.1.2. Прогнозирование вероятных результатов

Многие формы интеллектуального анализа данных являются прогнозирующими. Прогнозы имеют связанную вероятность, то есть показывает, насколько вероятно, что этот прогноз верен. Вероятности предсказания показывают, насколько пользователь может быть

уверенным в этом прогнозе. Некоторые формы предсказательного анализа данных генерируют правила, которые являются условиями, подразумевающими данный результат. Алгоритмы прогнозирования являются примером контролируемого обучения [59].

Вероятное прогнозирование обобщает то, что известно о будущих событиях и является разновидностью вероятностной классификации. В машинном обучении вероятная классификация способна спрогнозировать результат распределения вероятностей по набору классов, учитывая признаки входных данных. Вероятностные классификаторы могут объединятся в ансамбли, то есть алгоритмы, которые включают в себя несколько алгоритмов машинного обучения [44].

Достоинствами ансамблевых методов является их простота в реализации, они не требуют большого количества функций и могут моделировать данные с использованием нелинейных функций.

Недостатком ансамблевых методов является то, что они не предсказывают вероятность того, будет ли результат равен единице или нулю. Метод выстраивает признаки от наиболее вероятного к наименее вероятному положительному результату, и такая оценка не всегда бывает полезной для различного рода организаций.

1.1.3. Создание действенной информации

Интеллектуальный анализ данных может извлекать полезную информацию из больших объемов данных. На текущий день многие компании генерируют огромные объемы данных. Организации нуждаются в решениях, которые помогут им справиться с объемами больших данных. Для таких компаний очень важно превратить данные в ценную информацию. Компании хоть и имеют доступ к большим объемам данных, но не могут их использовать максимально с пользой, так как данные требуют анализа и интерпретации.

Данные, которые легко использовать, называют – действующей информацией. Это означает что данные отображаются в надлежащем контексте, точно и в месте, где пользователи могут с ними ознакомиться. Подобные данные компании могут использовать для принятия более эффективных и обоснованных решений, разрабатывать эффективную стратегию, увеличивать доходы, сокращать расходы и повышать эффективность [41].

Изучаемые данные должны обладать следующими свойствами:

- Корректность. Данные должны быть правильными, а значит точными. Если входные данные будут неверны, то и выводы будут ошибочные.

- Доступность. Данные должны быть доступными, чтобы каждый пользователь, которому необходимы данные, мог получить доступ к данным в любом месте и в любое время.

- Актуальность. Для принятия более точных решений данные должны отражать реальность как можно ближе к реальному времени. Устаревшие данные неэффективны для решения проблем, поскольку ситуации, предполагаемые этими данными, не отражают ситуацию в текущее время.

- Понятность. Данные представленные визуально или статистически хорошо воспринимаются. Зачастую визуализацию данных, которая плохо подходит для определенного набора данных, сложно интерпретировать. А значит и выводы могут быть сделаны ошибочные.

1.1.4. Большие массивы данных и базы данных

Доступность правильных данных, в идеальном виде с надлежащим анализом помогают достичь необходимых целей и задач. Большие данные включают в себя совокупность процессов и инструментов для мгновенного использования и управления большими наборами данных. Обозначенная концепция помогает понять тенденции, предпочтения и закономерности в огромной базе данных, сгенерированных при взаимодействии людей с системами и между собой [57].

Сосредоточенность на больших данных позволяет компаниям использовать аналитику и определять ценных клиентов. В свою очередь, это помогает создавать компаниям новый опыт, услуги и продукты. Благодаря обработанным и сохраненным данным компании эффективно получают прибыль. Большие данные позволяют понять потребности аудитории и принять более взвешенные решения [58].

Сочетание больших данных и аналитики дает возможность определить основные причины сбоев, проблем и дефектов в реальном времени, своевременно быстро и точно выявить аномалии, пересчитать риски в краткие сроки, повысить способность моделей глубокого обучения точно классифицировать переменные и реагировать на них, обнаруживать мошеннические действия до того как они повлияют на организацию, а также улучшить результаты лечения пациентов, за счет быстрого преобразования данных медицинских изображений в полезные сведения [28, 37, 49].

В обозримом будущем большие данные станут главным инструментом для принятия решений, так как массивы больших данных обладают большим потенциалом.

1.1.5. Интеллектуальный анализ данных и статистика

Методы интеллектуального анализа данных подходят для больших наборов данных и их легче автоматизировать. Фактически, алгоритмы интеллектуального анализа данных часто требуют больших наборов данных для создания качественных моделей.

Большинство методов, используемых в интеллектуальном анализе данных, могут быть помещены в статистическую структуру. Однако методы интеллектуального анализа

данных отличаются от традиционных статистических методов.

Традиционные статистические методы, по большей части, требуют значительного взаимодействия с пользователем для проверки правильности модели. В итоге статистические методы трудно автоматизировать. Более того, статистические методы обычно плохо масштабируются для очень больших наборов данных. Статистические методы основаны на проверке гипотез или поиске зависимостей на основе меньших выборок из большей совокупности [32].

Предметом статистики являются прежде всего числовые данные. Наборы данных, используемые при интеллектуальном анализе, могут представлять собой сочетание текста, аудио и видео файлов, изображений и так далее. Целью интеллектуального анализа является поиск закономерностей в данных. В большинстве случаев, такие закономерности должны иметь отношение к предметной области. При интеллектуальном анализе данных не всегда понятно какую закономерность необходимо найти и это затрудняет классификацию информации. Слишком общее определение классов может привести к подгонке, а слишком конкретное определение может привести к пропуску закономерностей, которые должны быть идентифицированы. Применение статистического обучения в подобных случаях может быть использовано для определения вероятностных моделей. Появляется возможность управлять идентификацией ошибок измерения и статистической значимости различных точек данных. Статистический анализ можно использовать для определения точек данных, на которые влияет первопричина, и тех, которые обусловлены чистой случайностью.

Статистика позволяет использовать прогностическую аналитику и разрабатывать различные классификации, которые могут повлиять на результат. Эффективный анализ невозможен без статистики. Использование передовых статистических методов, применяемых в процессе интеллектуального анализа данных, помогает компаниям и предприятиям увеличивать свои доходы, оптимизировать эффективность, снизить затраты, а также повысить удовлетворенность клиентов [33].

1.2. Этапы интеллектуального анализа данных

Интеллектуальный анализ данных не обнаруживает информацию автоматически без руководства. Шаблоны, которые обнаруживаются в ходе интеллектуального анализа данных, будут сильно различаться в зависимости от того, как будет сформулирована проблема. Для получения значимых результатов необходимо правильно формулировать постановку задачи. Алгоритмы интеллектуального анализа данных чувствительны к конкретным характеристикам данных, таких как выбросы, нерелевантные столбцы, кодированные данных и данные, которые добавляются или исключаются [11].

Этапы интеллектуального анализа данных отображены на *Рис. 1*.

Рис. 1. Этапы интеллектуального анализа данных.

Постановка задачи

Машинное обучение включает в себя построение математических моделей, помогающих понимать данные. «Обучение» вступает в игру, когда мы даем моделям настраиваемые параметры, которые могут быть адаптированы к наблюдаемым данным, таким образом, программа может считаться «обучающейся» на основе данных. С момента адаптации указанных моделей к ранее просмотренным данным, их можно будет использовать для прогнозирования и понимания аспектов вновь наблюдаемых данных. Следовательно, понимание постановки задач в машинном обучении имеет важное значение для эффективного использования этих инструментов.

В процессе данного этапа интеллектуальный анализ начинается с понимания того, что будет определять успех в конце процесса. Далее, на предварительном плане, разработанном для достижения целей, данная информация преобразовывается в проблему интеллектуального анализа данных [5].

Сбор и подготовка данных

Данные – ключевой компонент. При этом речь идет не о любых данных, а о правильных [2]. Подготовка данных часто занимает много времени и есть большой риск

возникновения ошибок. При сборе неверных данных, выходящих за пределы диапазона и отсутствующих результатов, выводы также могут быть ошибочными и беспорядочными. Соответственно, успех интеллектуального анализа данных сильно зависит от качества подготовки данных.

После того как цель проекта четка определена, начинается следующий этап - сбор данных. На данном этапе критически обдумываются ограничения на данные, хранение, безопасность и сбор, а также оценивается, как эти ограничения повлияют на процесс интеллектуального анализа данных. Данный этап включает в себя поиск доступных источников данных, критерии защищенности данных при хранении, сбор информации и понимание того, как будет выглядеть окончательный результат или анализ. После определения источников данных, происходит объединение данных и их очистка. Данные могут быть представлены в виде текстовых файлов, электронных таблиц и во многих других форматах, а также храниться на в различных базах и на разных серверах. Довольно часто такие данные нуждаются в предварительной обработке.

Параллельно со сбором и очисткой данных, происходит их изучение. В процессе которого становится ясно насколько подходит подготовленный набор данных к исследуемой предметной области [21].

В задачах медицинской диагностики в роли объектов выступают пациенты. Признаки характеризуют результаты обследований, состояние пациента, методы диагностики. Примеры бинарных типов данных: пол, наличие симптомов. В номинальном типе значения используются в роли меток, при котором могут быть измерены номера телефонов, паспортов, ФИО людей и так далее. Обозначенные измерения используются с целью отличия объектов друг от друга.

Построение и оценка модели.

Модель интеллектуального анализа данных получает данные из структуры интеллектуального анализа, затем данные обрабатываются алгоритмом. Структура и модель являются отельными объектами. В структуре интеллектуального анализа данных хранится информация, определяющая источник данных, а модель хранит информацию, полученную в результате статистической обработки данных.

Модель пуста до момента, пока данные, предоставленные структурой не будут обработаны и проанализированы. После обработки, модель данных будет содержать метаданные, результаты и привязки к структуре интеллектуального анализа данных. Модели интеллектуального анализа данных бывают двух типов: прогнозирующие и описательные.

Прогнозирующая модель анализа данных предсказывает значения данных, используя известные результаты, полученные из различных наборов данных. Основная цель прогнозирующих моделей состоит в предсказании будущего на основе прошлых данных, но не всегда на основе статистического моделирования. К прогностической модели относятся классификация, регрессия, прогнозирование и анализ временных рядов.

Описательная модель различает шаблоны и отношения данных. Описательная модель не пытается обобщить статистическую совокупность или случайный процесс, как это делает прогностическая модель. Данная модель фокусируется на обобщении преобразованных данных в полезную информацию для отчетности и мониторинга, в виде таблиц, графиков и статистических показателей. К описательной модели относятся кластеризация, правила ассоциации и последовательности [24].

На этапе построения модели создается структура данных. Далее для структуры создается одна или несколько моделей. Модель включает указание на алгоритм анализа данных и его параметры, а также анализируемые данные. При определении модели используются различные фильтры, которые позволяют выборочно использовать структуры данных для создания модели.

На данном этапе также проводится оценка модели, которая показывает качество работы созданной модели. Если было создано несколько моделей, то на этом этапе делается выбор в пользу той модели, что покажет наилучший результат.

Для задач предсказательного типа, качество прогноза выдаваемого моделью можно оценить на проверочном наборе данных, при котором известно значение прогнозируемого параметра [16].

Развертывание знаний.

Для реализации поставленной задачи необходимы концепция и внедрение. И если предыдущие этапы описывали концепцию задачи, то на данном этапе реализуются способы внедрения построенной модели [50]. Наиболее эффективные модели развертываются в производственной среде, также возможна интеграция средств интеллектуального анализа данных и пользовательских приложений. Конечный пользователь будет получать результаты работы модели.

Развертывание знаний – это использование интеллектуального анализа данных в целевой среде. На этапе развертывания из данных можно извлечь ценную и полезную информацию.

1.3. Методы интеллектуального анализа данных. Функционал программы Orange Data Mining.

Эра больших данных ускорила использование интеллектуального анализа данных. Методы интеллектуального анализа данных, с их мощью и автоматизацией, способны справляться с огромными объемами данных и извлекать ценность [35]. Ниже мы рассмотрим семь основных методов интеллектуального анализа данных.

1.3.1. Ассоциация

Ассоциация — это метод интеллектуального анализа данных, который обнаруживает вероятность часто встречающихся шаблонов в базе данных. Зависимости между часто встречающимися шаблонами выражаются в виде правил ассоциации. Правило ассоциации состоит из двух частей: антецедент (если) и консеквент (тогда).

Антецедент – это элемент, находящийся в данных, а консеквент – это элемент, который находится в сочетании с антецедентом. Правила ассоциации создаются путем поиска в данных частых шаблонов «если - то» и использования критериев «поддержки» и «уверенности» для определения наиболее важных отношений. Поддержка – это показатель того, как часто данное правило появляется в анализируемой базе данных. Уверенность показывает сколько раз данное правило оказывается истинным. Правило может демонстрировать сильную корреляцию в наборе данных, потому что оно появляется очень часто, но при применении может встречаться гораздо реже. Это будет случай высокой поддержки, но низкой уверенности. Правило может не особо выделяться в наборе данных, но дальнейший анализ может показать, что оно встречается очень часто. Это будет случай высокой поддержки, но низкой уверенности. Существует еще третий параметр – подъем, который представляет собой отношение уверенности к поддержке. Если значение подъема является отрицательным, то существует отрицательная корреляция между точками данных. Если значение положительное, то есть положительная корреляция, а если отношение равно 1, то корреляции нет.

Существует три различных типа алгоритмов, которые можно использовать для создания ассоциированных правил при интеллектуальном анализе данных:

1. Априорный алгоритм – алгоритм, который идентифицирует часто встречающиеся отдельные шаблоны в базе данных, а после расширяет их до более крупных шаблонов, следя за тем, чтобы шаблоны появлялись в базе данных достаточно часто.

2. Алгоритм Eclat (осколка) – означает кластеризацию классов эквивалентности и обход решетки снизу вверх. Алгоритм является более эффективной версией априорного алгоритма, т.к. априорный алгоритм работает по горизонтали, имитирую поиск в ширину

графа, алгоритм Eclat работает по вертикали и в глубину графа. Вертикальный подход алгоритма Eclat делает его более быстрым, чем априорный алгоритм.

3. Алгоритм роста FP (частого роста паттернов) – алгоритм повторяющихся шаблонов. Алгоритм работает в два этапа: построение FP – дерева и извлечение часто используемых шаблонов.

Метод ассоциации часто используют для анализа медицинских данных. Правила ассоциации могут быть полезны при медицинской диагностике. Используя анализ правил ассоциации, можно определить вероятность возникновения той или иной болезни при возникновении различных симптомов. В дальнейшем можно расширить симптоматическую базу, добавив новые симптомы и определить взаимосвязь между новыми признаками и соответствующими заболеваниями [27].

1.3.2. Классификация

Классификация – это процесс распределения изучаемых процессов по каким-либо существенным признакам. Классификация опирается на априорные ссылочные структуры, позволяющие разбить пространство всех возможных точек данных на множество классов, которые обычно не пересекаются [13]. Если регрессия отвечает на вопрос «сколько», то классификация – на вопрос «какого вида». Зависимая величина не числовая, а категориальная. В алгоритм поступает набор входных данных и соответствующие им выходные классы данных. Используя обучающий набор данных, алгоритм выводит модель или классификатор. Производная модель может быть деревом решений, логистической регрессией или нейронной сетью. Самая простая форма классификации – бинарный классификатор, порождающий на выходе одну из двух меток. Результатом классификации может быть также число с плавающей точкой от 0.0 до 1.0, означающее степень уверенности. В данном случае задается пороговая величина [18].

Классификаторы можно разделить на два основных типа:

1. Дискриминативный – это простой классификатор, который определяет только один класс для каждой строки данных. Он моделирует, полагаясь на наблюдаемые данные. Сильно зависит от качества данных, а не от распределений. Примером данного типа классификатора является логистическая регрессия, которую используют для зачисления в университеты на основе оценок учащихся и результатов тестов.

2. Генеративный – классификатор моделирует распределение отдельных классов и учится на модели, генерирующая данные посредством оценок и предположений. Алгоритм генеративной классификации используется для прогнозирования невидимых данных. Наиболее известным генеративным алгоритмом является – наивный байесовский

классификатор, который используют для обнаружения спама по предыдущим данным [8]. Примером может служить статья по фильтрации спама с помощью наивного байесовского метода. В данной статье авторы рассматривают и анализируют пять различных версий вышеуказанного классификатора для фильтрации спамы в различных наборах данных. По итогу исследования были выявлены две наиболее эффективные версии классификатора [47].

1.3.3. Кластеризация

Кластеризация – это процесс автоматического разбиения элементов некоторого множества на группы в зависимости от их схожести. Это неконтролируемый алгоритм, работающий с немаркированными данными. Группа точек данных будет объединятся вместе, чтобы сформировать кластер, в котором объекты будут принадлежать к одной и той же группе. Кластеризацию применяют к любому виду объектов при условии, что можно будет различить похожие и непохожие элементы [46]. Существует множество применений анализа кластеризации данных, таких как обработка изображений, анализ данных или распознавание образов. Примером являются изображения, сгруппированные на основе их цветов, форм на изображениях или того и другого. Люди могут группировать эти объекты, потому что понимают сходство и различие данных объектов между собой. Компьютеры, к сожалению, не обладают такой интуицией, поэтому кластеризация чего-либо с помощью алгоритмов начинается с представления объектов таким образом, чтобы они могли быть прочитаны компьютерами.

Кластерный анализ требует интерпретируемости данных. Результат кластеризации должен быть удобным и понятным. Основная цель кластеризации в анализе данных – убедиться, что случайные данные хранятся в группах на основе их характерного сходства.

Обычно данные перепутаны и не структурированы. Данные нельзя быстро проанализировать, и именно поэтому кластеризация информации так важна при интеллектуальном анализе данных. Группировка придает определенную структуру данным, организовав их в группы похожих объектов данных. Последующий анализ подобных данных становится намного проще и минимизирует возможность ошибки.

Существует несколько различных способов кластерного анализа, основанных на различных моделях. К каждой модели применяются отдельные алгоритмы, различающие ее свойства и результаты. Модели отличаются своей организацией и типом отношений между ними.

Типы кластеризации:

1. Централизованная – каждый кластер представлен одним средним вектором, и значения объектов сравниваются с этими средними значениями.

2. Распределенная – кластер построен с использованием статистических распределений.

3. Связанная – связанность в моделях основана на функции расстояния между элементами.

4. Групповая – алгоритмы имеют только групповую информацию.

5. Графическая – организация кластера и отношения между элементами определяются графиком.

6. Плотность – элементы кластера сгруппированы по областям, где наблюдения расположены наиболее плотно и похожи между собой.

Методы кластеризации данных хорошо подходят для анализа данных. Наборы данных, обработанные по кластерному методу, обеспечивают хорошие результаты по многим различным типам данных. Одним из значимых анализов по кластеру является обработка изображений, которая позволяет обнаруживать различные типы шаблонов в данных изображения. Данный анализ очень эффективен в биологических исследованиях [15].

1.3.4. Регрессия

Метод регрессии – это метод интеллектуального анализа данных, используемый для прогнозирования диапазона числовых значений с учетом определенного набора данных. Метод регрессии часто путают с методом классификации, так как оба метода применяются в прогнозном анализе. Только регрессия используется для прогнозирования числового или непрерывного анализа, а классификация распределяет данные по дискретным категориям.

Типы регрессии:

1. Полиномиальная регрессия (Polynomial regression)– это алгоритм машинного обучения, который используется для обучения линейной модели на нелинейных данных. Функция полиноминальной регрессии описывается выражением:

$$y = \sum_{j=0}^{k} b_j x^j$$

где, b_i – коэффициент полинома;

x^j - независимая переменная;

y^j - зависимая переменная;

Полиноминальная регрессия применяется в математической статистике при моделировании трендовых составляющих временных рядов и является методом прогнозирования.

2. Линейная регрессия (Linear Regression)– это метод регрессии, который использует линейную зависимость для построения связи между целевой переменной и одной или несколькими независимыми переменными. Уравнение линейной регрессии представлено данным уравнением:

$$Y_j = a + bx_j + e_j$$

где, a – свободный член (пересечение) линии оценки

b – коэффициент регрессии линии оценки

e - ошибка

X – независимая переменная

Y – зависимая переменная.

Целью регрессионного анализа является разработка статистической модели, позволяющей предсказывать значения зависимой переменной.

3. Хребет (Ridge) – это метод регрессии для анализа данных мультиколлинеарной регрессии. Возникновение линейной корреляции между двумя независимыми переменными известно как мультиколлинеарность.

Когда оценки методом наименьших квадратов получаются наименее смещенными и со значительной дисперсией, возникает гребенчатая регрессия, результаты которой сильно отличаются от реального значения. С другой стороны, гребенчатая регрессия уменьшает количество ошибок, добавляя степень смещения к оценочному значению регрессии.

4. Логистическая регрессия (Logistic regression) Данный метод используется, когда зависимая переменная является бинарной, таких как 0 и 1, «истина» и «ложь», «успех» или «неудача» (*Рис. 2*).

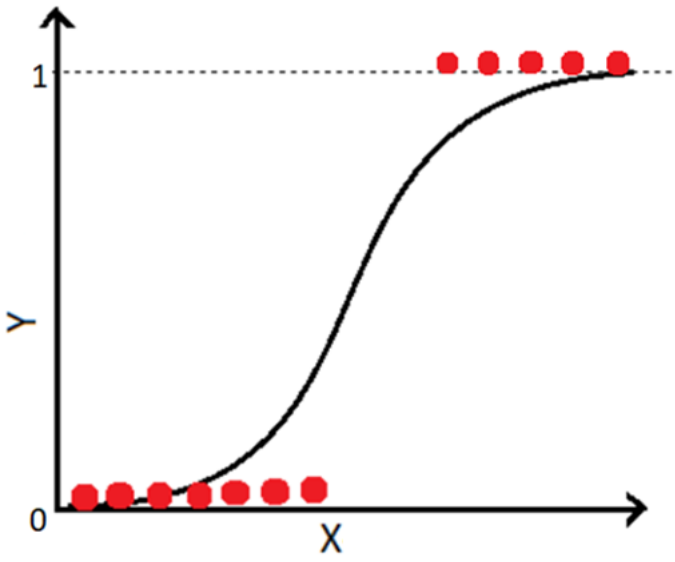


Рис. 2. Логистическая регрессия при решении задач классификации.

Все модели логистической регрессии могут быть записаны формулой:

$$f(y) = \frac{1}{1 + x^{-y}}$$

f(y) – вероятность того, что произойдет событие;

x – основание натуральных логарифмов;

y – стандартное уравнение регрессии.

Благодаря логистической регрессии можно оценивать вероятность того, что событие наступит для конкретного случая.

5. Лассо – регрессия (Lasso) – это метод регрессии, использующий сокращение. Все точки данных сводятся к центральной «средней» точке. Метод лассо – регрессии хорошо подходит для моделей с мультиколлинеарностью.

6. Эластичная сеть (ElasticNet) – это тип регуляризованной линейной регрессии, содержащий штрафные функции L1 и L2. Эластичная сеть — это своего рода линейная регрессия, интегрирующая штрафы за регуляризацию в градиентный спуск во время обучения. Например, геномный отбор с использованием регуляризованных моделей линейной регрессии [51].

7. Иерархическая регрессия (Hierarchical Regression) – это метод регрессии, который используют для проверки того, описывают ли представляющие интерес переменные статистически значимую дисперсию в зависимых переменных после принятия во внимание всех фактов. Иерархическую регрессию используют в медицинских исследованиях.

8. Дерево решений (Decision Tree) – это алгоритм машинного обучения, основанный на правиле: «если <условие>, то <результат>». Алгоритм сжимает большой набор данных во все меньшие и меньшие части, выстраивая при этом дерево решений. В связи с этим конечным результатом будет является дерево с листовыми узлами, содержащее набор узлов.

9. Случайный лес (Random Forest)– это контролируемая стратегия обучения метода регрессии, использующая подход ансамблевого обучения. С целью получения достоверного прогноза изученных данных, необходимо воспользоваться стратегией ансамблевого обучения, то есть объединить прогнозы нескольких способов машинного обучения.

1.3.5. Прогнозирование

Данный метод заключается в максимально точном прогнозирование будущего с учетом всей доступной информации, включая исторические данные и знания о любых будущих событиях, которые могут повлиять на прогнозы. Для хорошего прогноза важно учитывать следующие характеристики: точность, надежность, своевременность, простота в использовании и понимании и рентабельность.

Известно три основных вида прогнозирования:

1. Деревья решений

2. Регрессия (линейная, полиноминальная, логистическая)

3. Нейронные сети

Методы прогнозирования делятся на два типа: качественные и количественные. При отсутствии доступных данных или при несоответствии имеющихся данных прогнозам, необходимо использовать качественные методы прогнозирования. Это хорошо разработанные структурированные подходы к получению хороших прогнозов без использования исторических данных. Данный метод также известен как оценочный и обычно зависит от экспертного мнения. Соответственно качественный прогноз чаще всего необъективен. Качественное прогнозирование нематематический процесс, так как редко основаны на данных.

Количественные методы прогнозируют данные, анализируют предшествующие данные и опираются на статистические методы прогнозирования последующих событий. Количественный метод не прогнозирует результаты, основанные на мнениях или интуиции человека. При данном методе используются определенное количество данных и фактов, доступных в прошлом, проанализировав которые алгоритмы выдают соответствующие результаты. Существует широкий спектр методов количественного прогнозирования, часто разрабатываемых в конкретных дисциплинах для определенных целей. Многие задачи количественного прогнозирования используют анализ временных рядов, либо данные поперечного сечения [1].

1.3.6. Обнаружение выбросов

Выбросы – это точки данных, не принадлежащие определенной совокупности. Это аномальное наблюдение, лежащее далеко от других ценностей. Выброс – это наблюдение, которое расходится с хорошо структурированными данными (*Рис. 3*)

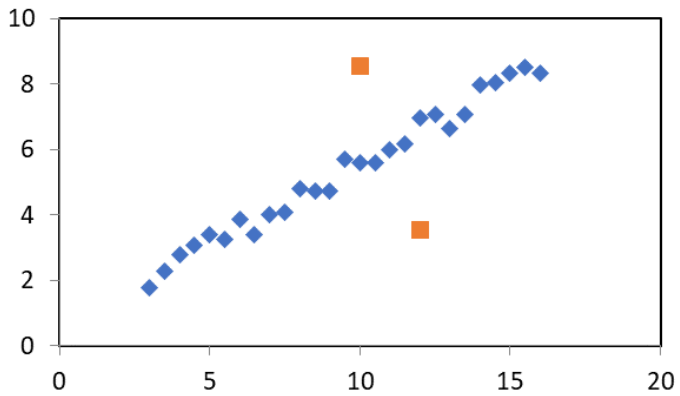


Рис. 3. Схематическое изображение выбросов данных.

Обнаружение выбросов одна из основных проблем интеллектуального анализа данных. Обнаружение аномалий в данных сердцебиения может помочь в прогнозировании сердечных заболеваний. На этапе подготовки наборов данных для моделей машинного обучения очень важно обнаружить все выбросы, чтобы избавиться или проанализировать их. Выделяют пять основных способов обнаружения выбросов:

1. Стандартное отклонение. Точка данных, превышающая стандартное отклонение будет является выбросом.

2. Блочные диаграммы. Представляют собой графическое изображение числовых данных через их квантили. Это очень простой и эффективный способ визуализации выбросов (*Рис. 4*).

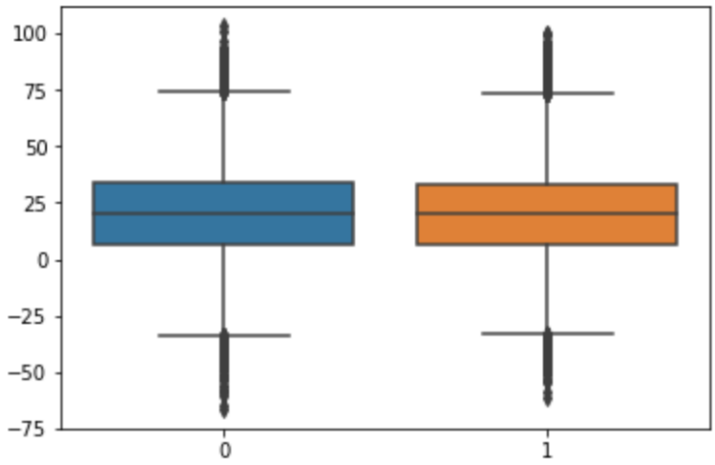


Рис. 4. Блочные диаграммы.

Любые точки данных, отображающиеся выше или ниже границ, считаются выбросами.

3. Кластеризация DBScan. Алгоритм кластеризации, в котором данные группируются. В силу этого алгоритм используется в качестве метода обнаружения аномалий на основе плотности с одномерными или многомерными данными.

4. Изолирующий лес. Алгоритм обучения без учителя, относящийся к семейству ансамблевых деревьев решений. Этот алгоритм работает с большими наборами данных и эффективен при обнаружении выбросов.

5. Надежный случайный вырезанный лес. Это неконтролируемый алгоритм для обнаружения выбросов. Работа которого заключается в связывании оценки выбросов. Низкие значения баллов указывают на то, что точка данных считается «нормальной». Высокие значения указывают на наличие аномалий в данных. Определения «низкий» и «высокий» зависят от приложения, но на практике предполагается, что значения величин, превышающие три стандартных отклонения от среднего балла, считаются аномальными [12].

1.3.7. Сопоставление шаблонов

Метод интеллектуального анализа данных помогающий идентифицировать аналогичные закономерности или тенденции в данных. Проблема поиска закономерностей в данных является фундаментальной и имеет долгую историю. Примером этого может служить открытие закономерностей в атомных спектрах, которое сыграло ключевую роль в развитии и проверке квантовой физике в начале двадцатого века. Область распознавания образов связана с автоматическим обнаружением закономерностей в данных с помощью компьютерных алгоритмов и с использованием этих закономерностей для выполнения таких действий, как классификация данных по различным категориям [42]. Распознавание образов предоставляет возможность дальнейшего совершенствования. Метод отслеживания шаблонов анализирует данные любого типа: звуки, изображения, тексты и другое.

ГЛАВА 2. ОБЗОР ЛИТЕРАТУРЫ. ПРИМЕНЕНИЕ ИНТЕЛЛЕКТУАЛЬНОГО АНЛИЗА ДАННЫХ

За прошедшее десятилетие обозначились способы применения глубокого обучения в машинном зрении, обработке естественного языка и аудиоданных, а также в приложениях подобных Deep Face, которые извлекают и идентифицируют человеческие лица из цифровых изображений [25]. Помимо этого, глубокое обучение произвело революцию в области биомедицинского анализа изображений. Традиционные подходы использовали алгоритмы для конкретных задач описания изображений с созданными вручную функциями, такими как морфология клеток, количество, интенсивность и текстура. Изучение признаков с помощью глубоких сверточных нейронных сетей является неявным, и обучение сети обычно фокусируется на конкретных задачах. Для обучения глубоких сети обычно требуется большое количество изображений, что ограничивает ее полезность [9].

2.1. Анализ изображений с помощью визуального программирования

Машинное обучение на изображениях не всегда требует обучения на тесно связанном наборе изображений. Глубокая сеть, предварительно обученная на достаточно большом количестве разнообразных изображений, может делать полезные выводы из широкого спектра новых наборов изображений. Идея основана на трансферном обучении, методе машинного обучения, сохраняющим знания, полученные от одной проблемы, в обученной модели и применяет их к другой проблеме, которая может быть совсем другой. Типичная глубокая сеть для анализа изображений содержит сверточные слои, определяющая структурные особенности изображений, за которыми следуют полноценно связные слои, объединяющие функции и находящие взаимодействия между ними[39]. Применительно к классификации узлы сети предпоследнего слоя содержат информацию о наиболее информативной комбинации признаков изображения, а последний слой включает по одному узлу для каждого класса. Для трансферного обучения можно перепрофилировать глубокую сеть, обученную на одном наборе изображений [19].

Orange Data Mining предлагает подход визуального программирования к анализу изображений, при котором пользователи могут сочетать встраивание изображений с помощью предварительно обученных глубоких моделей с кластеризацией и классификацией. Orange поддерживает выполнение основных функций интеллектуального анализа данных на изображениях в простой в использовании среде, где общие задачи анализа данных задуманы и выполнены в течение нескольких минут.

2.1.1. Анализ изображений с помощью визуального программирования

Набор инструментов Orange Data Mining поддерживает пользователей при сборке рабочих процессов анализа, состоящих из компонентов, в которые загружают изображения, встраивают их в векторное пространство и анализируют данные профили для вывода кластеров изображений или классификаций.

Анализ данных в Orange Data Mining реализован через рабочие процессы. Рабочий процесс состоит из виджетов – компонентов, которые обрабатывают, моделируют или визуализируют данные. Виджеты принимают данные и отражают или отправляют результаты в качестве выходных данных. Рабочие процессы анализа данных в Orange Data Mining определяются выбором виджетом и связей между ними.

На рис. 5 показан рабочий процесс, в который загружено множество изображений из выбранного каталога. Данный процесс представляет изображения с векторами признаков посредством встраивания, оценивает расстояние между векторами и между данными изображениями, а также вычисляет расстояние при кластеризации и визуальном отображении сходства изображений на графике многомерного масштабирования.

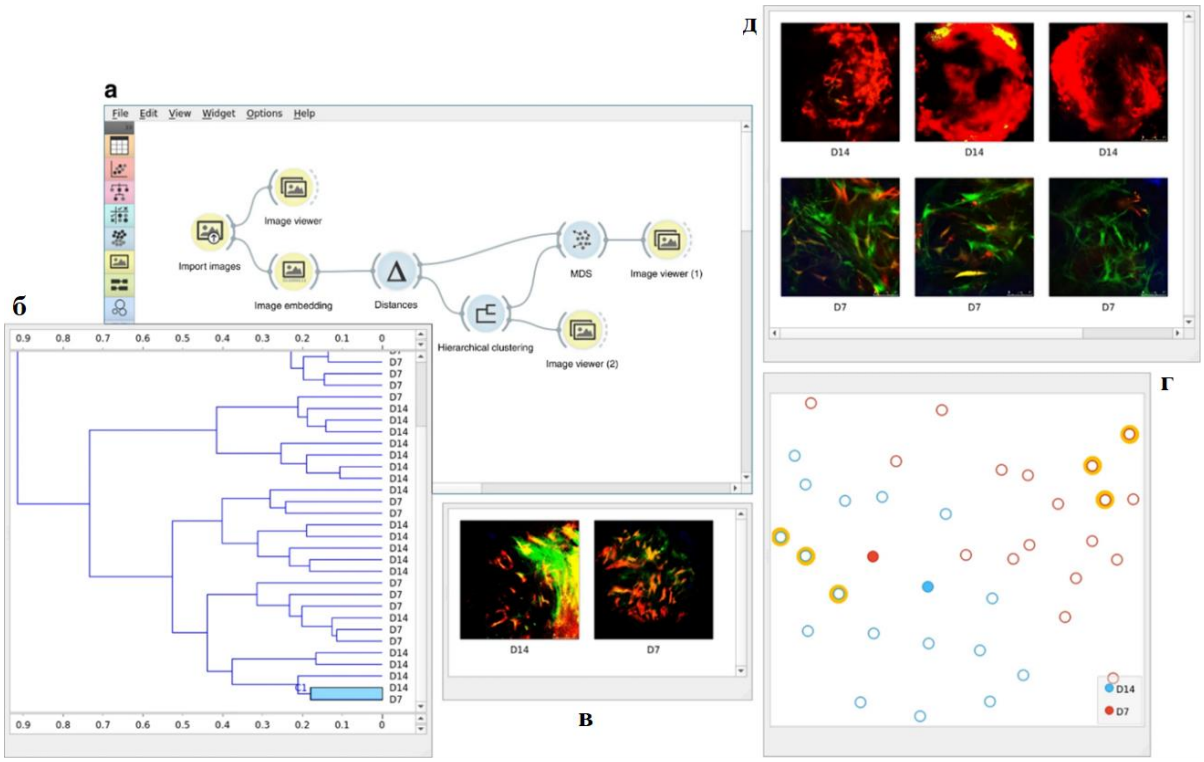


Рис. 5. Интеллектуальный анализ изображений.

Пользователи могут отслеживать выполнение каждого шага рабочего процесса Orange Data Mining и проверять каждый промежуточный результат. Например, просмотреть загруженные изображения, визуализировать результат иерархической кластеризации в дендрограмме и даже визуализировать выборку изображений из конкретной ветви дендрограммы или из секции точек на графике многомерного масштабирования.

Пользователи вместе с тем могут просматривать необработанные данные, поступающие от устройств для встраивания, или профили функций выбранных изображений в дендрограмме. Возможность проверки процесса и результатов на каждом этапе анализа помогает пользователям обрести уверенность в результатах и ознакомиться с процедурами анализа.

На *рис. 5а* показан неконтролируемый анализ изображений заживления костей. Рабочий процесс анализа данных начинается с импорта 37 изображений из локальной папки. Изображения можно просматривать в виджете «Средства просмотра изображений» и передать в модуль встраивания изображений (Image embeddings), настроенный на использование глубокого Web-поиска. Здесь вычисляются расстояния между встроенными изображениями, представленными в виде дендрограммы (*Рис. 5б*) с помощью виджета Hierarchical Clustering. Кластеры хорошо соответствуют времени, выраженном в днях после травмы (D7 и D14), но за некоторыми исключениями. Одним из таких исключений является ветвь двух изображений, выделенных на дендрограмме (*Рис. 5б*) и показанных в средстве просмотра изображений (*Рис. 5в*). Расстояния изображения также были переданы виджету «многомерного масштабирования» (MDS), который также демонстрирует разделение между образцами заживления кости в разное время, обозначенными разными цветами. Три репрезентативные точки MDS из D7 и D14 были выбраны вручную (*Рис.* 5*г*), и показаны в средстве просмотра изображений (*Рис. 5д*). Два изображения, выделенные на дендрограмме (*Рис. 5б*) также были переданы виджету MDS в качестве подмножества данных. В этой проекции данных они визуализируются как закрашенные точки (*Рис. 5г*), кажущимися близкими друг другу в силу их сходства. На рисунке показано как биолог может исследовать данные после кластеризации – сначала сосредоточив внимание на неправильно классифицированных образцах и просматривая изображения, а после выбирая некоторые из наиболее классифицированных изображений в качестве точки отсчета для дальнейшего исследования.

Виджеты в Orange Data Mining интерактивны, то есть они передают результаты при любом изменении параметров виджета или любом изменении выбора элементов при их графическом представлении. Примером может являться виджет иерархической кластеризации, который позволяет выбирать ветвь построенной дендрограммы. Поместив виджет Hierarchical Clustering в рабочий процесс, он выводит данные, соответствующие выбранной ветви, передающиеся в виждет визуализации изображения. Любое изменение выбора ветвей в дендрограмме распространяется по рабочему процессу и обновляет содержимое любого из нижестоящих виджетов [36].

2.1.2. Методика визуальной аналитики

Визуальная аналитика сочетает в себе интерактивную визуализацию и автоматизированный анализ данных, включая машинное обучение. Orange Data Mining охватывает все основные этапы фреймворков визуального анализа, включающим загрузку и преобразование данных, визуализацию данных с взаимодействием с пользователем, вывод моделей данных и визуализацию моделей. Orange Data Mining реализует компоненты для визуализации и обработки данных, а посредством визуального программирования помогает аналитикам данных комбинировать и связывать компоненты анализа данных и создавать рабочие процессы анализа данных. Типичный компонент рабочего процесса получает данные, обрабатывает их, визуализирует результат обработки и выводит результаты анализа или любой выбор пользователя для дальнейшей обработки нижестоящим компонентом рабочего процесса. Примером может быть виджет иерархической кластеризации, который получает попарные расстояния между элементами данных, строит соответствующую кластеризацию, визуализирует ее в виде дендрограммы и выводит элементы данных, связанных с выбранными пользователем ветвями дендрограммы. Выходные данные компонентов Orange Data Mining мгновенно обрабатываются при любом изменении входных данных, что приводит к созданию системы визуальной аналитики, при котором любое изменение в компоненте вызывает изменения в нижестоящих компонентах, обновляющим в последствии свои результаты и соответствующие визуализации данных и моделей.

Для процесса встраивания, изображения представлены векторами признаков, полученными предварительно обученными глубокими сверточными сетями.

Рабочие процессы на *рис. 5* используют несколько стандартных процедур интеллектуального анализа данных, таких как вычисление попарных расстояний между элементами данных, иерархическая кластеризация и многомерное масштабирование для построения модели и перекрестной проверки. Программа Orange Data Mining встраивает библиотеки Python для машинного обучения и обработки данных, заключает их функциональные возможности в строительные блоки рабочих процессов, представляющие интерфейс, при котором пользователь может изменять параметры методов машинного обучения или просматривать результаты и соответствующие визуализации предполагаемых моделей.

2.2. Анализ данных по сердечным заболеваниям с использованием алгоритма кластеризации К-средних

Алгоритм k-средних – это наиболее простой метод кластеризации. Он разбивает множество элементов векторного пространства на заранее известное число кластеров k. Алгоритм стремится минимизировать среднеквадратичное отклонение на точках каждого кластера. Основная идея алгоритма заключается в том, что на каждой итерации рассчитывается центр масс для каждого кластера, полученного на предыдущем шаге. Затем векторы разбиваются вновь в соответствии с тем, какой из новых центров оказался ближе по выбранной метрике. Алгоритм останавливает анализ, когда на какой-либо из итераций не происходит изменения кластеров.

2.2.1. Кластеризация болезней сердца методом К-средних

Виджет точечной диаграммы показывает двумерное изображение точечной диаграммы на основании каждого постоянного и дискретного рассматриваемого атрибута. Рисунки представляют собой графики рассеяния после применения кластерного анализа k-средних болезней сердца с уровнями холестерина (*Рис. 6а*) и уровнями покоя (*Рис. 6б*).

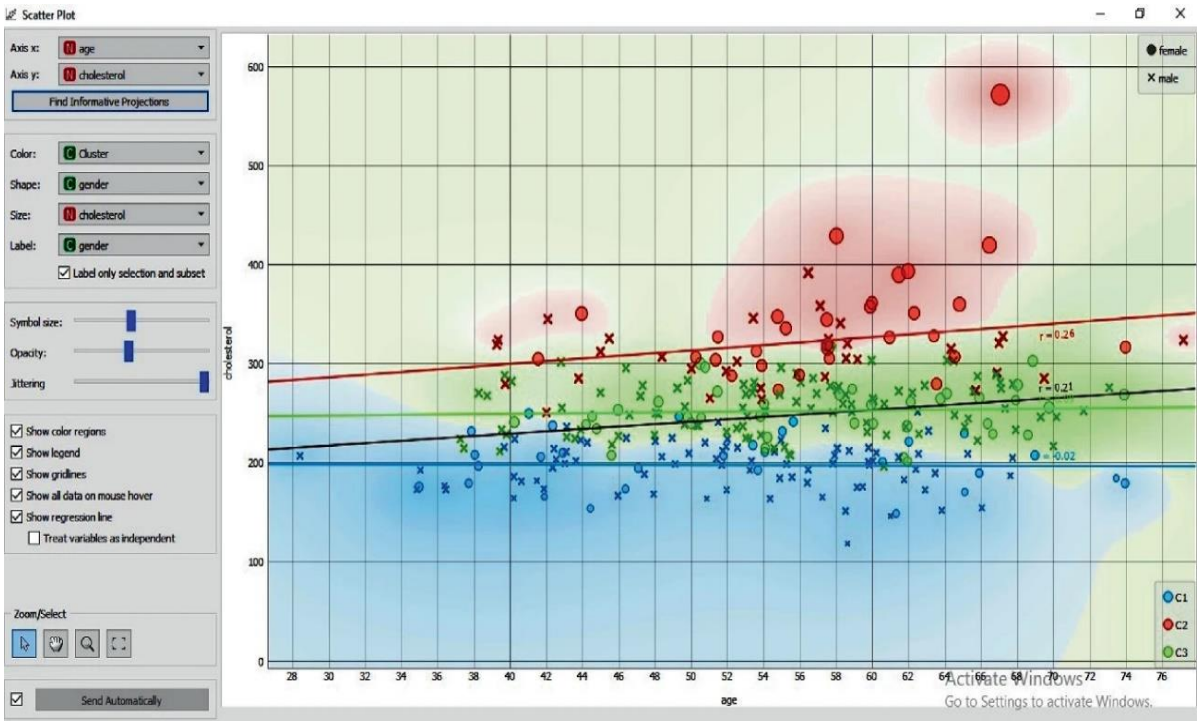


Рис. 6а. График рассеяния болезни сердца с уровнями холестерина.

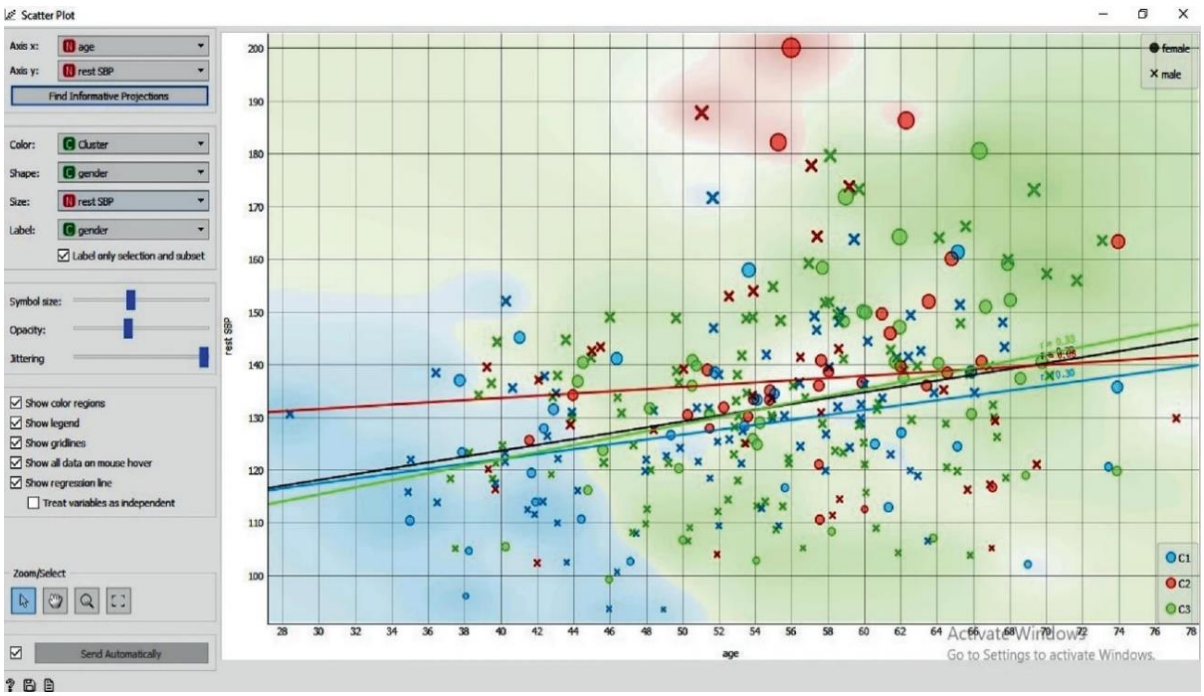


Рис. 6б. График рассеяния болезни сердца с уровнями покоя.

Данные представлены в виде набора точек комбинации фокусировки, каждая из которых имеет значение относительно X – свойство оси, выбирающее положение относительно оси уровня, наряду с этим стимул относительно атрибута оси Y определяет вертикальную ось. На диаграмме также видны различные свойства болезней сердца, холестерина и САД в покое.

Алгоритм кластеризации обозначает создание различных кластеров холестерина «С», таких как кластер «С1», «С2» и «С3». Кластеризация уровня «С1», очень низкий диапазон, например, холестерин 200, уровень «С2» от 200 до 239, высокий уровень холестерина, и диапазон «С3» - 240 или более высокого уровня.

На *рис. 7* показана диаграмма интеллектуального анализа данных Orange Data Mining в наборах данных о сердечных заболеваниях по систолическому артериальному давлению (САД) в покое. Анализ показывает алгоритм кластеризации k-средних по возрасту 28-78 лет и диапазону САД в покое 0-200. Такие кластеры как возраст «С1» от 28-40 лет показывают очень низкий диапазон САД в покое, возраст «С2» от 40-60 лет высокий диапазон САД в покое и возраст «С3» от 60-78 лет очень высокий диапазон САД в покое.

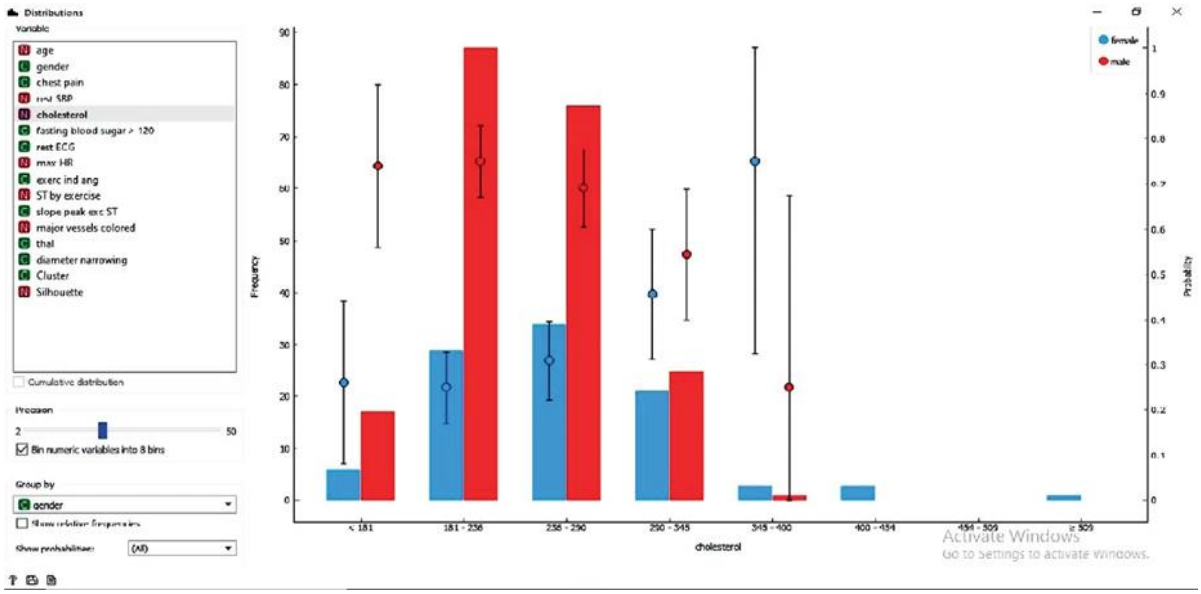


Рис. 7. Визуализация данных сердечных заболеваний по систолическому артериальному давлению.

Результат работы алгоритма кластеризации k-средних значительно упрощает анализ данных о сердечно-сосудистых заболеваниях. Выполнена оценка наборов данных о сердечных заболеваниях с использованием различных кластеров и цветов по отношению к холестерину и визуализации данных САД в покое [29].

ГЛАВА 3. ИССЛЕДОВАНИЕ ПРОГНОСТИЧЕСКИХ МОДЕЛЕЙ ПО ПРИЗНАКАМ БАЗЫ ДАННЫХ (НА ПРИМЕРЕ СЕРДЕЧНОЙ НЕДОСТАТОЧНОСТИ)

Прогностические алгоритмы не идеальны, и последний шаг в принятии решения все равно остается за человеком. Тем не менее аналитические платформы разумно сужают выбор сходных по симптоматике заболеваний, что значительно облегчает врачебному персоналу постановку диагноза. Следовательно, прогностические модели превосходно подходят для определения задач в медицине, в частности, при диагностике заболеваний по клинико-диагностическим признакам.

3.1. Orange Data Mining

Оптимизационная модель поддержки принятия врачебных решений (далее-модель) в данной работе реализована в программе Orange Data Mining (*Рис. 8*).

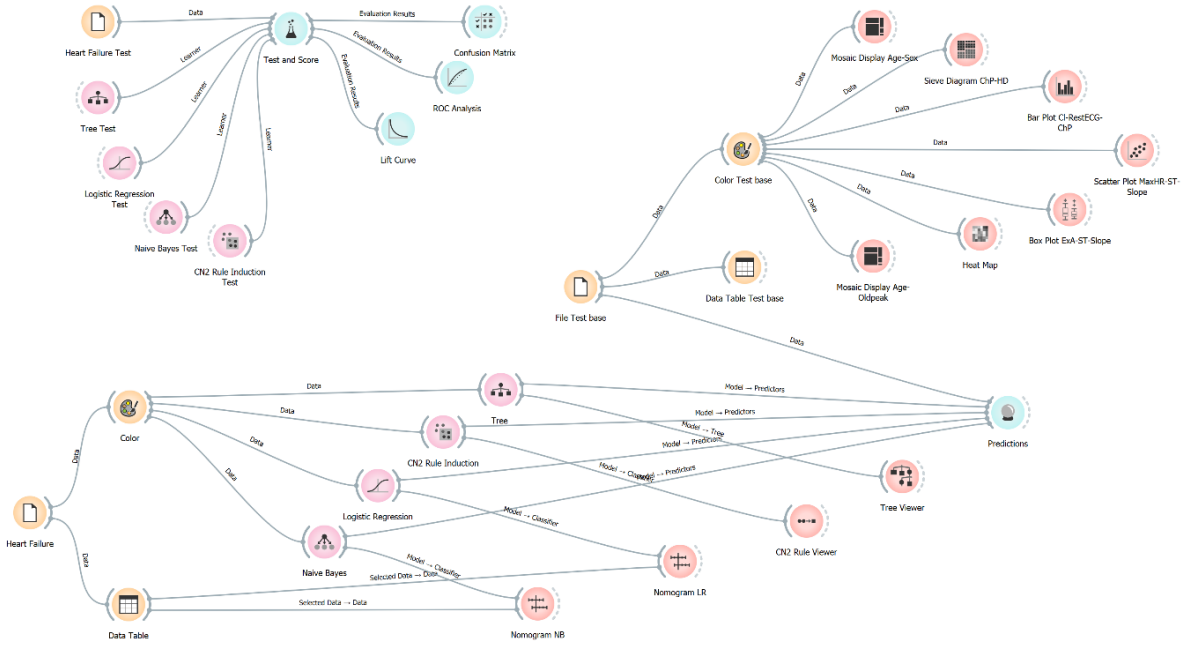


Рис. 8. Оптимизационная модель поддержки принятия врачебных решений.

В последующих параграфах мы подробно рассмотрим элементы этой модели.

3.2. База данных

Используемые в модели данные получены из открытого источника kaggle.com.

Kaggle – это платформа, на которой размещены открытые наборы данных. Наборы данных публикуются пользователями и различными организациями. Данные доступны для

исследования и создания моделей по машинному обучению. На платформе также проводятся открытые и закрытые конкурсы по созданию моделей исследования данных.

На данной платформе были загружены 4 набора данных с клинико-диагностическими исследованиями сердечной недостаточности (*Рис. 9*). Основной файл был создан путем объединения загруженных файлов. Все изучаемые данные уникальны и не содержат дубликатов в наборе.

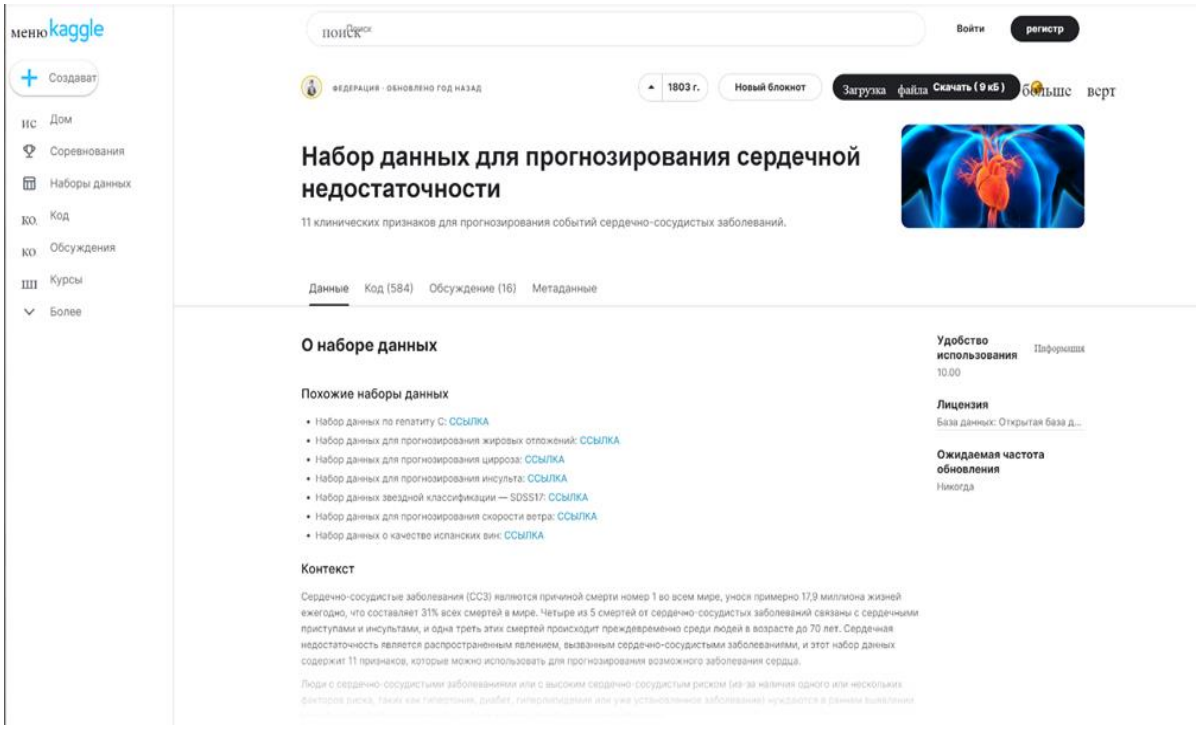


Рис. 9. Платформа kaggle.com

Набор данных содержит 2788 экземпляров данных, которые разделены на 11 признаков и диагноз. Признаки включают в себя:

1. Age - возраст пациента (лет)

2. Sex - пол пациента (F – мужской, M - женский)

3. ChestPainType – тип боли в груди (TA – типичная стенокардия, ATA – не типичная стенокардия, NAP – неангинозная боль, ASY – бессимптомная стенокардия)

4. RestingBP – артериальное давление в состоянии покоя (мм рт.ст.)

5. Cholesterol – холестерин (мм/л)

6. FastingBS – уровень сахара в крови натощак («1»: если уровень> 120 мг/дл, «0»: если уровень ≤ 120 мг/дл)

7. RestingECG – ЭКГ в покое («норма» - нормальное, «ST» - аномалия ST-T, LVH – вероятная или определённая гипертрофия левого желудочка по критериям Эстеса)

8. MaxHR – максимально достигнутая частота сердечных сокращений (числовое значение от 60 - 202)

9. ExerciseAngina – стенокардия, вызванная физической нагрузкой (Y – да, N - нет)

10. Oldpeak – старый пик ST (числовое значение, измеренное в депрессии)

11. ST_Slope – изменение смещения сегмента ST при ЭКГ (Down – нисходящий, Flat – плоский, Up - восходящий)

Диагноз:

12. HeartDisease – результат (1 - есть сердечная недостаточность, 0 - нет сердечной недостаточности).

Набор данных был разделен на 2 части. Первая часть содержит 2750 экземпляров данных и использовалась для обучения прогностических моделей, вторая часть содержит 38 экземпляров данных и использовалась для тестирования уже обученных алгоритмов.

3.3. Набор данных в Orange Data Mining, ее интерпретация и настройка параметров.

Работа в Orange Data Mining начинается с ввода данных. Исходный файл загружается в виджет «File» (*Рис. 10*).

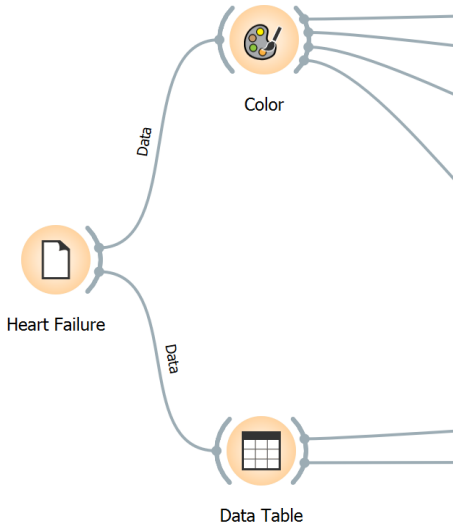


Рис. 10. Набор данных.

Данные в программе Orange Data Mining отображаются в виде атрибутов, которые выражены числовыми numeric или категориальными categorical значениями. Всем атрибутам данных, кроме HeartDisease присвоена роль feature (особенность), поскольку они являются признаками сердечной недостаточности. Атрибут HeartDisease имеет значение target (цель). Таким образом, искомый атрибут будет отображаться как результат работы алгоритмов.

Обработанные данные передаются в виджет Data Table, где отображаются в табличном виде. Источниками данных являются пациенты, а значения атрибутов – это результаты клинико-диагностических исследований (*Рис. 11*).

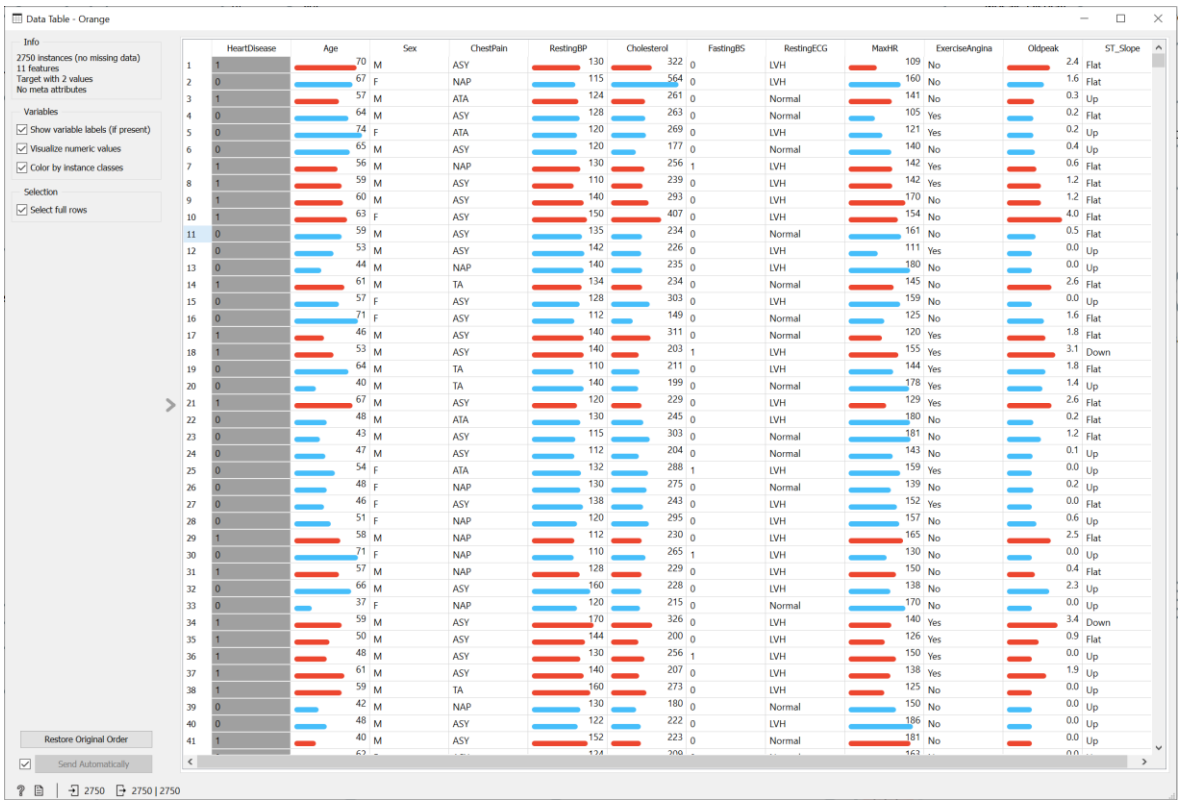


Рис. 11. Набор данных по сердечной недостаточности в виджете Data Table

В левой части, виджет отображает информацию о наборе данных: число экземпляров данных, число атрибутов, количество значений цели. В виджете можно визуализировать цветом значения атрибутов, относящиеся к разным классам. Можно выбрать часть экземпляров данных и отправить их на «выход» данных из виджета. С помощью кнопки Restore Original Order можно изменить или восстановить порядок экземпляров данных после сортировки на основе атрибутов. Виджет также дает возможность подготовить отчет.

Параллельно с визуализацией набора данных в виджете Data Table, данные из виджета File поступают в виджет Color. В данном виджете устанавливается цветовая легенда для атрибутов. В дальнейшем, при визуализации данных с помощью виджета «Color» можно будет просмотреть более четкую и понятную картину данных (*Рис. 12*).

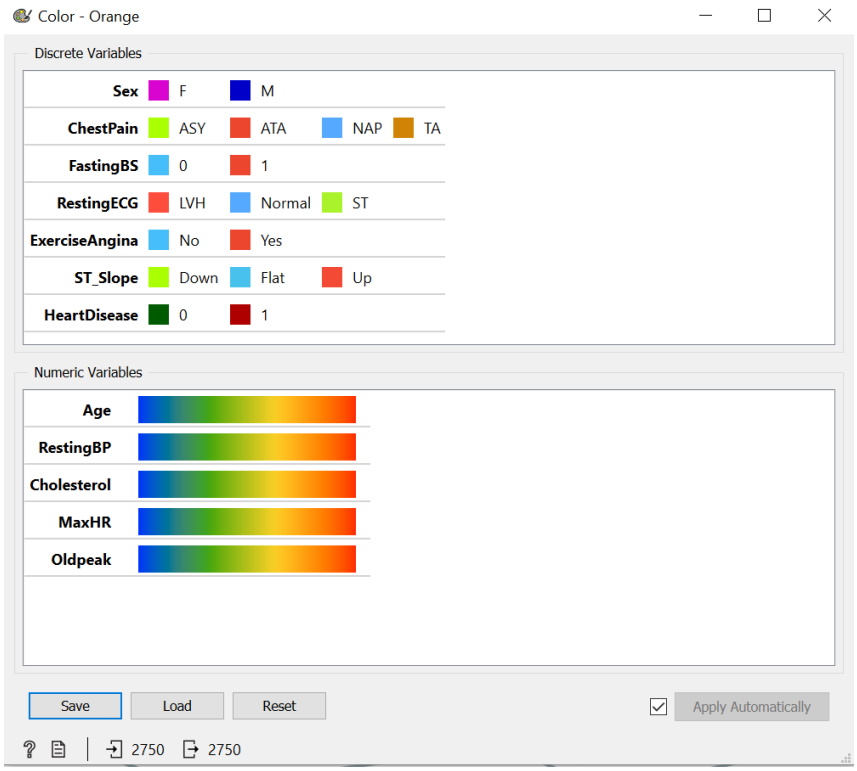


Рис. 12. Виджет Color.

3.4. Алгоритмы обработки и прогнозирования данных

Для того, чтобы построить модель с наиболее достоверными результатами прогноза сердечной недостаточности, рассмотрим четыре прогностические модели: Tree, Logistic Regression, Naive Bayes и CN2 Rule Viewer [31]. По методам классификации и обработки данных, вышеуказанные алгоритмы подходят для прогнозирования сердечной недостаточности на основе клинико-диагностических исследований (*Рис. 13*). Используемые алгоритмы относятся к качественным предсказательным моделям анализа данных.

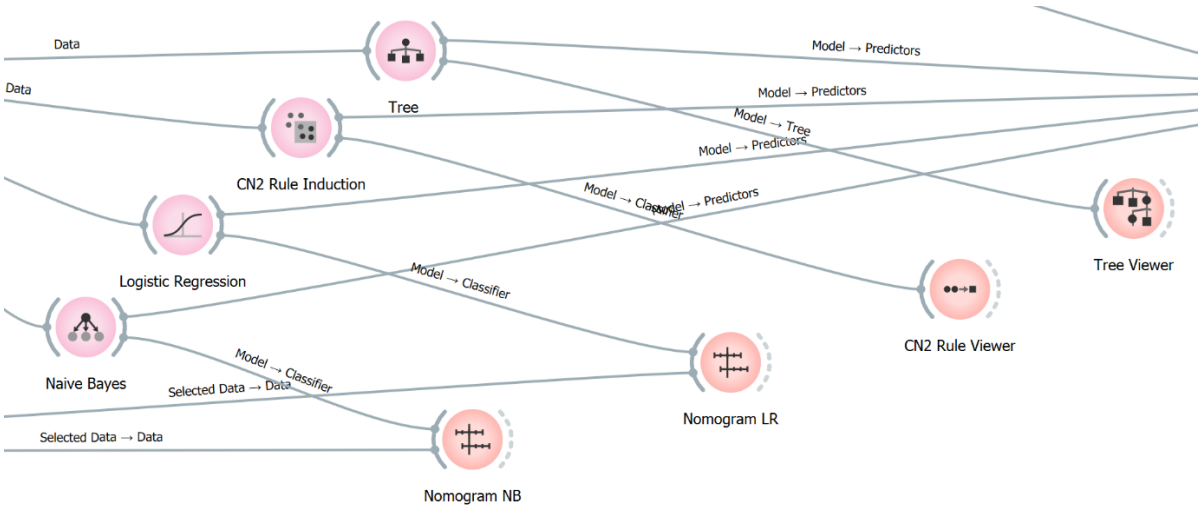


Рис. 13. Прогностические модели и их визуализация.

3.4.1. Дерево решений

Дерево решений (Tree) – это простейший и самый надежный метод классификации в интеллектуальном анализе данных. Это блок – схема, похожая на древовидную структуру, которая состоит из «узлов» и «веток». Основной корневой узел представляет собой всю выборку, которая в последующем делится на два и более однородных подузла. Узел, разделенный на подузлы, называется родительским узлом, тогда как подузлы являются дочерними элементами родительского узла. Если подузел разделяется на дополнительные подузлы, он называется узлом принятия решений. Узлы, которые не разделяются являются конечными узлами. Ветвью считается отдельная часть дерева. Разделение атрибутов данных происходит согласно правилу: «если-условие, то результат», по принципу максимального снижения энтропии результирующего подмножества относительно родительского. Дерево разбивает данные и сопоставляет их записям, не входящим в обучающий набор [22]. При выборе между равновероятностными классами, алгоритм «Tree» Orange Data Mining выбирает первый атрибут из имеющихся.

Для построения алгоритма «Tree» набор данных по сердечной недостаточности проходит через виджет Color. Далее набор данных поступает в виджет Tree и строится алгоритм.

Виджет Tree предоставляет возможность установить необходимые параметры, в том числе имя алгоритма. Для построения алгоритма Tree с набором клинико-диагностических данных сердечной недостаточности, было решено не использовать бинарное дерево, так как критерии некоторых признаков имеют более двух переменных. Количество минимальных экземпляров в узлах было выбрано – 10, для того чтобы алгоритм выстроил разбиение из необходимого количества обучающих примеров в каждой ветке. Данное количество предлагает наглядно увидеть разбивку, не разветвляясь на более мелкие неинформативные ветки. В строке разделения подмножеств, было установлено значение – 5. Данное ограничение дает возможность алгоритму не разбивать узлы с количеством экземпляров меньше заданного, что для нашего исследования является вполне разумным. В виджете был установлен лимит – 100, ограничивающий максимальную глубину дерева классификации до этого значения узлов. При достижении разделения 100%, алгоритм останавливает деление, этого достаточно, чтобы увидеть результат деления. В последствии конечный результат виден в одном из узлов. В других узлах видим вероятность предсказания результата.

Работа алгоритма Tree показана на *Рис. 14*, где видно, что алгоритм делит основной корневой узел на четыре ветки по категориям атрибута ChestPain, следовательно, все записи данных первоначально разделены на четыре основные ветви по типу боли в груди: типичная стенокардия, атипичная стенокардия, неангинозная боль и бессимптомная стенокардия.

Дальнейшее деление происходит по принципу: на каждом уровне выбирается переменная, позволяющая провести наилучшее разделение, то есть разделение с наибольшим информационным выигрышем [10].

Первая ветка – бессимптомная стенокардия. В данную ветку входят 845 из 2750 экземпляров данных, из которых 75% (634 записи) имеют сердечную недостаточность в анамнезе. Далее деление узла данных происходит по атрибутам: ST_Slope, Oldpeak, Age, Sex, MaxHR, RestingBP, Cholesterol, ExerciseAngina. В результате, можно отметить большой процент положительной диагностики сердечной недостаточности в данной ветви алгоритма. Это позволяет предположить, что у пациентов отсутствуют жалобы на общее самочувствие в следствии отсутствия боли в груди или других болезненных симптомов.

Вторая ветка – атипичная стенокардия. В данную ветку входят 469 из 2750 экземпляров данных, из которых 55, 2% (259 записи) не имеют сердечной недостаточности. Это свидетельствует о том, что у пациентов присутствует болевой синдром, который возникает вне зависимости от физической и эмоциональной нагрузки. В связи с этим, большинство пациентов проходят диагностические исследования, что позволяет предотвратить возникновение сердечной недостаточности.

Третья ветка – неангинозная боль. В данную ветку входят 719 из 2750 экземпляров данных, из которых 53,4% (384 записи) имеют сердечную недостаточность в анамнезе. Предположительно это связано с тем, что у пациентов присутствует некий болевой синдром, которой может возникнуть в любой части тела. В следствии с этим, больные не связывают боль, возникшую не груди с болезнью сердца. Следовательно, упускается время на предотвращение возникновения заболевания.

Четвертая ветка - типичная стенокардия. В данную ветку входят 717 из 2750 экземпляров данных, из которых 73,5% (527 записи) — это отсутствие сердечной недостаточности. При типичной стенокардии, болевой синдром возникает в области сердца, что заставляет пациентов отнестись более серьезно к своему состоянию здоровья и заблаговременно пройти диагностические исследования. В конечном итоге, это позволяет избежать возникновения заболевания.

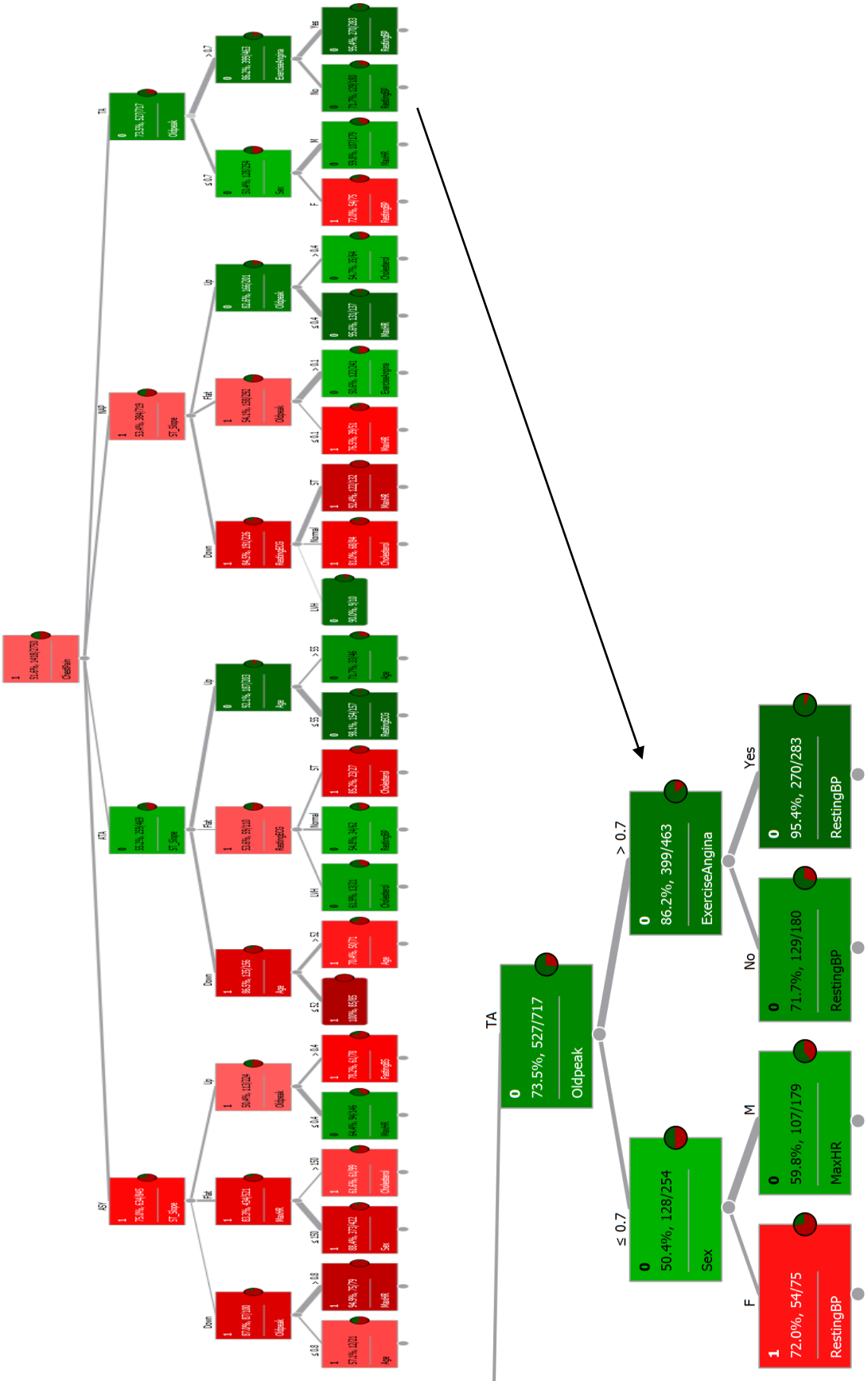


Рис. 14. Визуализация алгоритма дерево решений.

3.4.2. Логистическая регрессия

Логистическая регрессия (Logistic Regression) – хорошо известный вид классификации в линейном моделировании. Алгоритм логистической регрессии относится к типу регрессионного анализа. Регрессионный анализ – это тип метода прогнозного моделирования, использующийся для поиска взаимосвязи между зависимой переменной, в нашем случае диагнозе, и рядом независимых переменных, которые в нашей работе выражены показателями клинико-диагностических исследований [39].

Логистическая регрессия применима как для бинарной классификации, так и для многоклассовой – в форме мультиноминальной логистической регрессии. Бинарная логистическая регрессия используется для прогнозирования бинарного результата на основе набора независимых переменных позволяющая моделировать вероятность определенного события или класса. Она помогает понять влияние нескольких независимых переменных на одну переменную результата. По мимо бинарной, существует полиноминальная и порядковая логистическая регрессия. Полиноминальная логистическая регрессия используется, когда есть одна категориальная зависимая переменная с двумя или более возможными результатами. Порядковая логистическая регрессия используется, когда зависимая переменная упорядочена, то есть у зависимой переменной имеется значимый порядок двух или более категорий или уровней [14].

Недостатком логистической регрессии является то, что она не может предсказать непрерывный результат. Это связано с тем, что логистическая регрессия работает только тогда, когда зависимая переменная является двоичной. Кроме того, предполагается, что в данных нет пропущенных значений. К недостаткам логистической регрессии еще можно отнести неточность, когда это связано с небольшим количеством фактических наблюдений.

Набор данных из виджета Color поступает в виджет Logistic Regression, где выстраивается алгоритм классификации логистической регрессии.

Логистическая регрессия использует для прогнозирования вероятности возникновения какого-либо события сравнивая его с логистической кривой. События прогнозируются по значениям множества признаков. Логистическая регрессия учитывает абсолютный вклад каждого атрибута и выдает ответ в виде вероятности бинарного события. В данной работе результатом будет наличие или отсутствие сердечной недостаточности, которые будут соответствовать либо 1, либо 0. Алгоритм логистической регрессии анализирует, как признаки пациента коррелируют с результатами. Но прогностическая модель не может повлиять на определение признаков [38].

Регуляризацию можно поставить либо «L1» либо «L2». По первому типу регуляризации алгоритм отбирает из ряда признаков наиболее важные, которые сильнее

всего влияют на результат. По второму типу алгоритм устанавливает запрет на непропорционально большие весовые коэффициенты, тем самым предотвращая переобучаемость модели. В данной работе тип регуляризации установлен «L1».

Визуализацию алгоритма Logistic Regression отображает виджет Nomogram. В номограмму наряду с этим поступают данные из виджета Data Table. Номограмма показывает рейтинг и вклад каждого атрибута на результат анализа в каждом отдельном экземпляре данных. Виджет Data Table дает возможность выбрать необходимую запись, по которому номограмма покажет анализ и результат прогнозирования логистической регрессии. Номограмма ранжирует атрибуты по значимости: от наиболее к наименее важному.

На *рис. 15* показаны номограммы прогнозирования сердечной недостаточности и ее отсутствия у мужчины 70 лет, с бессимптомной стенокардией, артериальным давлением 130 мм рт.ст. в состоянии покоя, уровнем холестерина - 322 мм/л., уровнем сахара ниже 120 мг/дл., гипертрофией левого желудочка при ЭКГ в состоянии покоя, максимальной частотой сердечных сокращений равной – 109, отсутствием стенокардии при физических нагрузках, ишемической болезнью сердца в прошлом в анамнезе и изменением сегмента ST на ЭКГ в плоскость.

В ходе рассмотрения номограммы (*Рис. 15а*), на которой целевой класс имеет значение 1, мы видим, что наиболее значимым показателем при прогнозировании сердечной недостаточности алгоритм логистической регрессии посчитал тип боли в груди, далее изменение сегмента ST при ЭКГ, уровень холестерина и далее по наименее значимым показателям. Показатели стенокардии и ее отсутствия при физических нагрузках и артериальное давление, логистическая регрессия посчитала наименее значимыми. Таким образом, при подсечёте суммы баллов диагностических признаков, получаем итоговую цифру – 215,41 балла, что соответствует 75% вероятности возникновения сердечной недостаточности.

Номограмма (*Рис. 15б*) с целевым классом – 0 показывает вероятность отсутствия сердечной недостаточности. Показатели диагностических исследований ранжированы также как в предыдущей номограмме. Сумма баллов показателей равна – 127,63 балла, что соответствует 25% вероятности отсутствия сердечной недостаточности.

По итогу сравнения двух номограмм следует, что логистическая регрессия, проанализировав диагностические показатели у данного пациента прогнозирует большую вероятность возникновения сердечной недостаточности. Таким образом, можно просмотреть результат прогнозирования по каждому из 2750 экземпляров данных. Алгоритм логистической регрессии рассматривает признаки пациента, коррелирующие с его

результатами. Тем не менее, данная прогностическая модель не может повлиять на определение признаков.

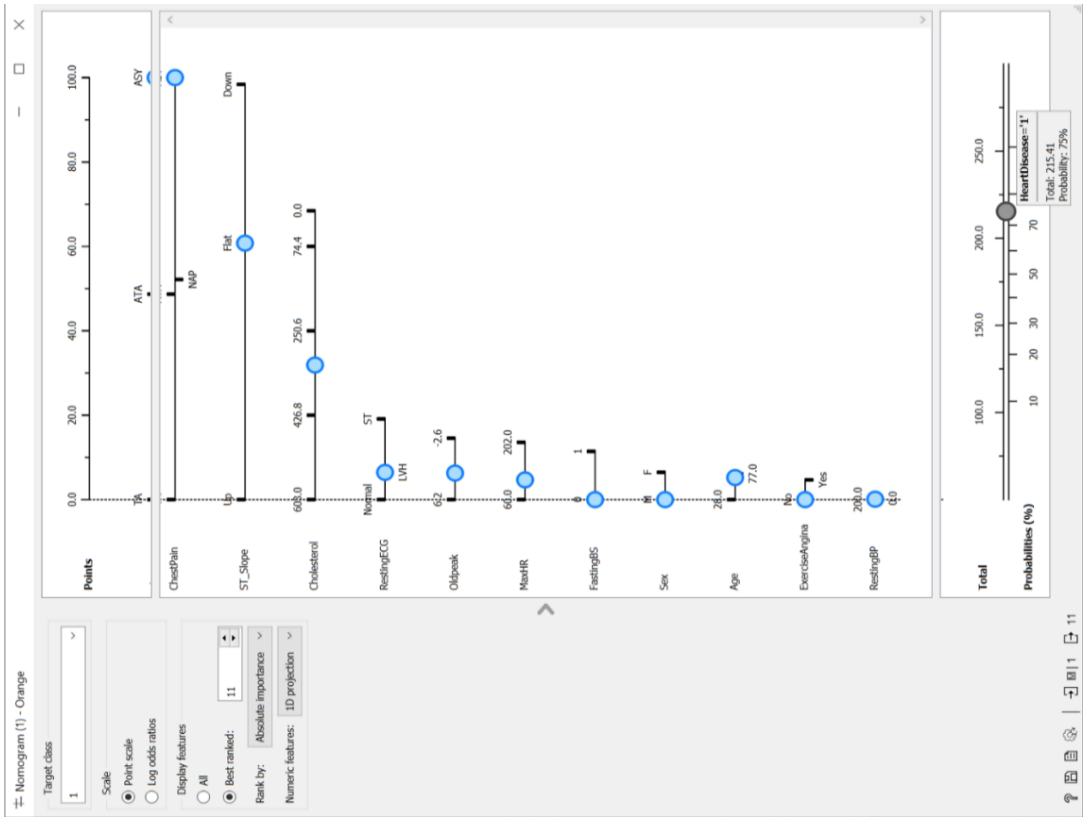


Рис. 15а. Номограмма, визуализирующая алгоритм логистической регрессии, на которой параметр Target class установлен 1

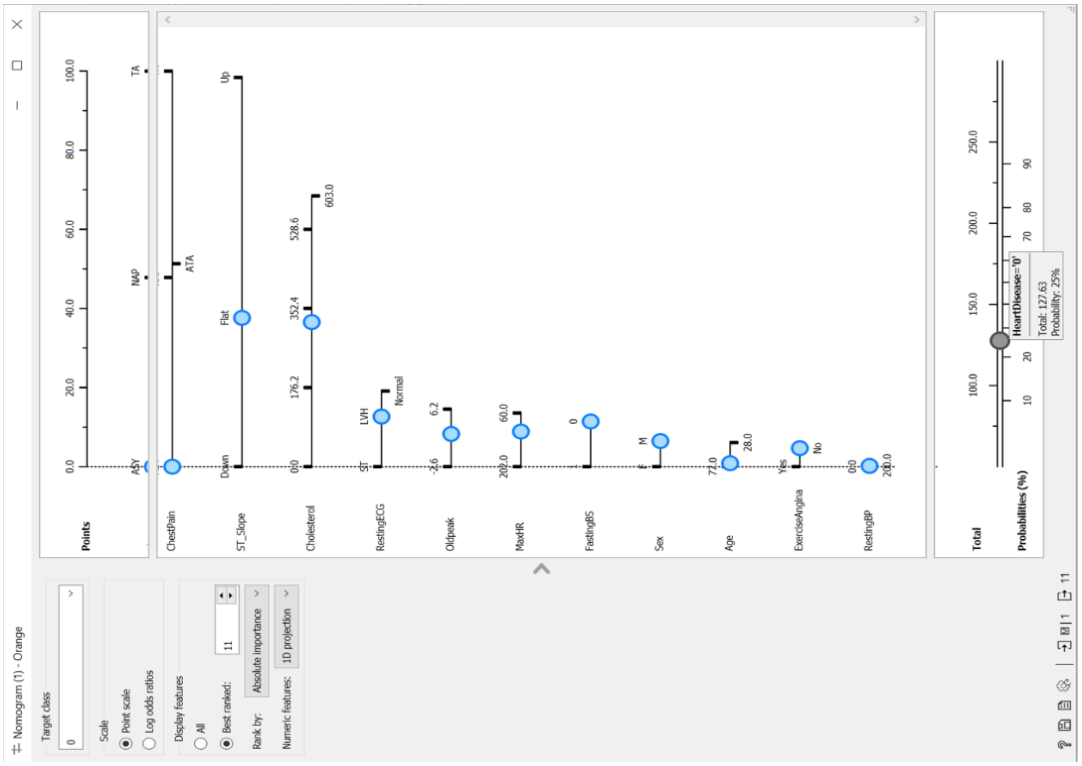


Рис. 15б. Номограмма, визуализирующая алгоритм логистической регрессии, на которой параметр Target class установлен 0

3.4.3. Наивный Байесовский алгоритм

Наивный Байесовский алгоритм (Naive Bayes) относится к вероятностным алгоритмам, основанным на применении теоремы Байеса, позволяющая определить вероятность события при условии, что произошло другое взаимозависимое с ним событие. Теорема Байеса названа в честь ее автора английского математика Томаса Байеса (1702-1761). Томас Байес был первым математиком, использующим условную вероятность для создания алгоритма. Алгоритм использует вероятность для вычисления пределов неизвестного параметра [43]:

$$P\ (y|x) = \frac{P(x|y)P(y)}{P(x)}$$

P (x) – априорная вероятность наступления события A (гипотеза)

P (e) – априорная вероятность наступления события B

P ($x|y$) – вероятность наступления события A, при наступлении события B.

P ($y|x$) – частота возникновения события B, при наступлении события A.

В результате формула Байеса дает возможность переоценить априорные представления о мире (P(y)) на основе частичной информации (данных), полученных в виде наблюдений (P(x|y)), в качестве вывода получая новое состояние этих представлений (P(y|x)) [17], то есть наивный Байесовский алгоритм исходит из предположений, что все атрибуты независимы, одинаково важны и в равной степени влияют на результат.

В современном мире наивный Байесовский алгоритм имеет множество применений, таких как фильтрация спама и классификация документов. Наивному Байесу требуется лишь небольшое количество обучающих данных для оценки требуемых параметров. Более того, наивный байесовский алгоритм значительно быстрее других сложных и продвинутых алгоритмов, прогнозирует результаты. С другой стороны, наивный байесовский алгоритм печально известен своей плохой оценкой, поскольку предполагает, что все признаки имеют одинаковую важность, что неверно в большинстве реальных сценариев [23].

Суть анализа данных, по сердечной недостаточности, на основе теоремы Байеса состоит в построении гипотезы, на основе имеющихся знаний в исследуемой области. На основе этих знаний, гипотезе приписывается нулевая вероятность. Затем анализируются данные, собранные при диагностическом исследовании и первоначальные выводы обновляются. Если новые диагностические данные подтверждают гипотезу, то вероятность возрастает, если не подтверждают – вероятность снижается.

Визуализацию алгоритма Naive Bayes отображает виджет Nomogram. Наряду с этим в номограмму наивного Байеса поступают данные из виджетов Color и Data Table. Номограмма наивного Байеса строится аналогично номограмме логистической регрессии. В

виджете Data Table выбирается необходимый экземпляр данных, по которому номограмма показывает анализ и результат прогнозирования алгоритма наивного Байеса, также показывается рейтинг и вклад каждого атрибута на результат анализа каждого экземпляра данных. При визуализации наивного Байеса номограмма строит график с положительными и отрицательными значениями, которые показывают наибольшее и наименьшее влияние атрибута на отсутствие и наличие сердечной недостаточности.

На Рис. 16 отображены номограммы прогнозирования сердечной недостаточности и ее отсутствия у 57 летней женщины с бессимптомной стенокардией, артериальным давлением 128 мм рт.ст. в состоянии покоя, уровнем холестерина - 303 мм/л., уровнем сахара ниже 120 мг/дл., гипертрофией левого желудочка при ЭКГ в состоянии покоя, максимальной частотой сердечных сокращений равной – 159, отсутствием стенокардии при физических нагрузках, отсутствием ишемической болезнью сердца в анамнезе и восходящим изменением сегмента ST при ЭКГ.

При рассмотрении номограммы (Рис. 16*а*), в которой целевому классу присвоено значение 1, очевидно, что наивный байесовский алгоритм проанализировав все показатели диагностических исследований, вывел суммарный балл равный отрицательному значению -59,94, который соответствует 34% вероятности возникновения сердечной недостаточности.

Номограмма (Рис. 16*б*) с целевым классом – 0 показывает вероятность отсутствия сердечной недостаточности. Сумма баллов показателей равна положительному значению 59,94 балла, что соответствует 66% вероятности отсутствия сердечной недостаточности.

В ходе сравнения двух номограмм видно, что наивный байесовский алгоритм, проанализировав диагностические показатели у данного пациента, прогнозирует большую вероятность отсутствия сердечной недостаточности.

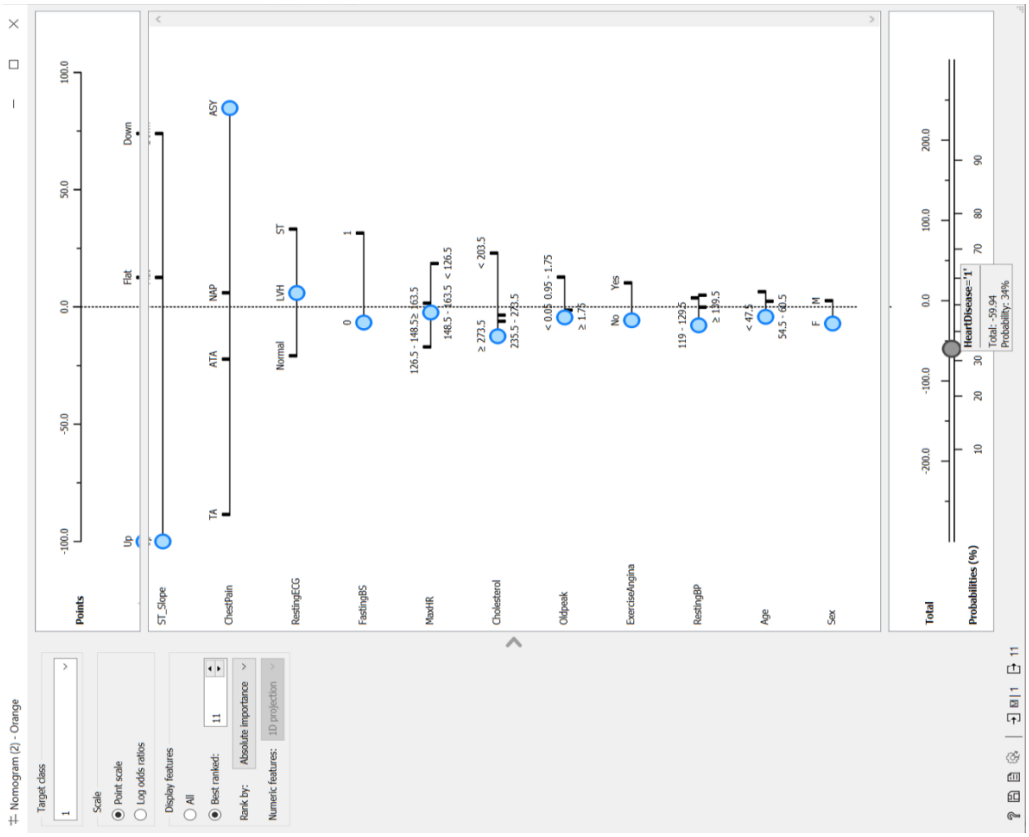


Рис. 16а. Визуализация прогноза сердечной недостаточности алгоритма Naive Bayes в номограмме

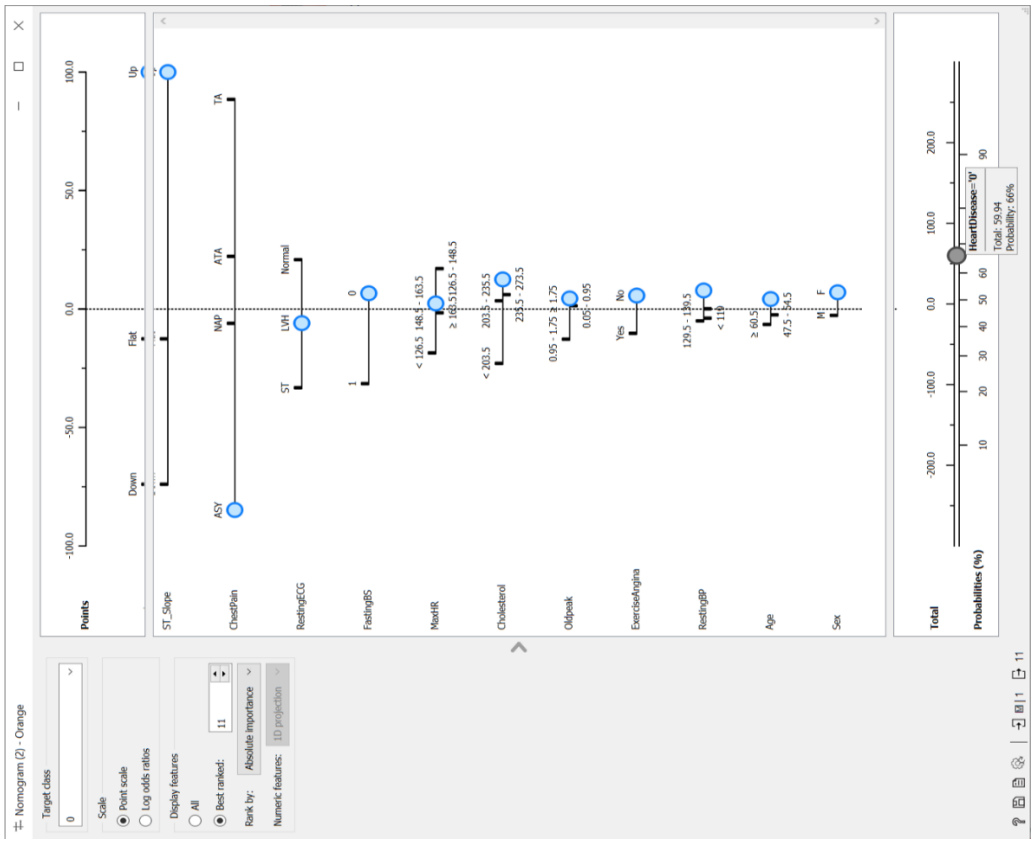


Рис. 16б. Визуализация прогноза отсутствия сердечной недостаточности алгоритма Naive Bayes в номограмме

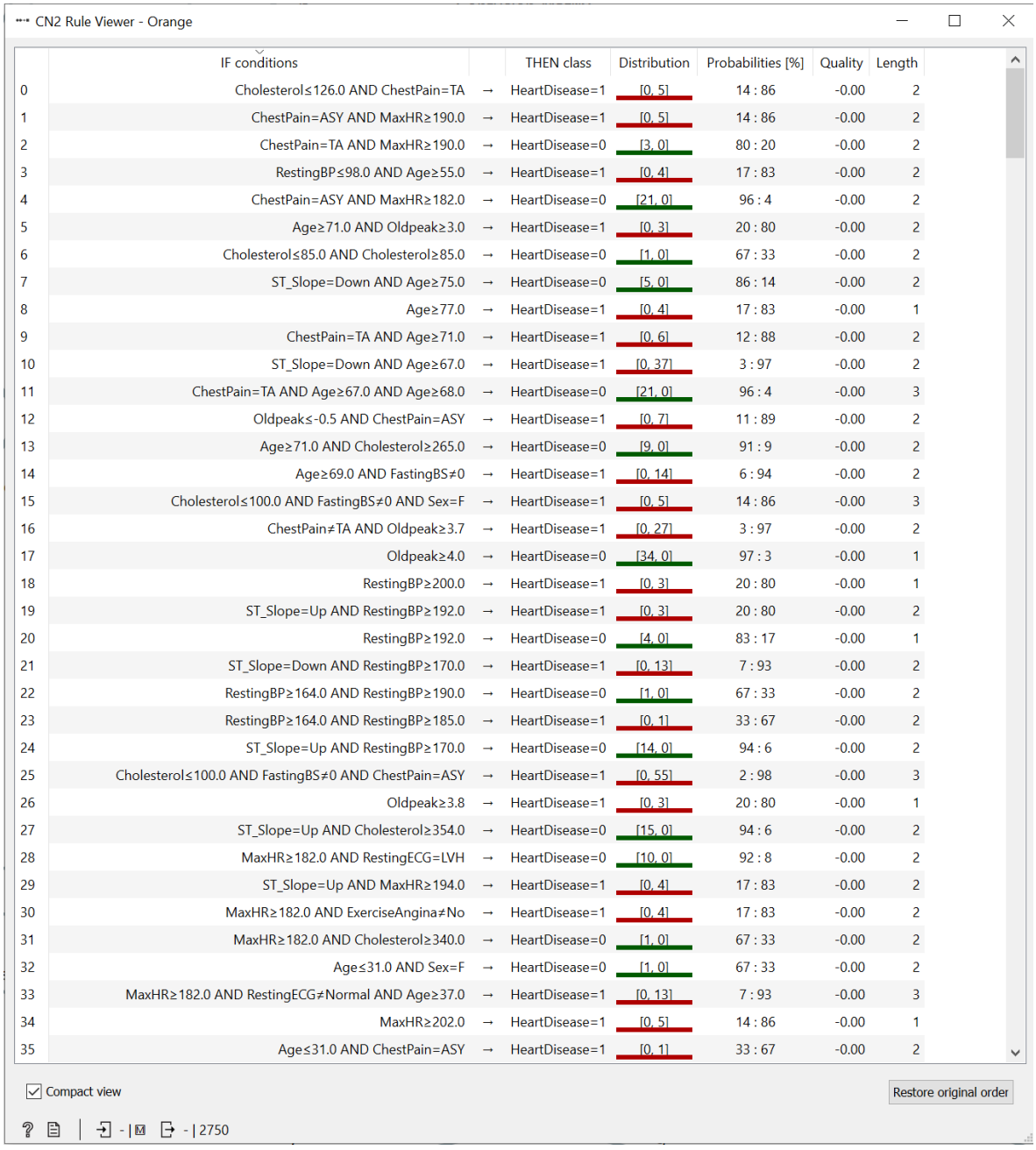
3.4.4. Индукция правил

Индукция правил (CN2 Rule induction) – это алгоритм, создающий упорядоченный набор правил, которые предсказывают класс при определенных условиях. Алгоритм анализирует все множество данных и находит правило на основе минимума энтропии, являющимся наилучшем среди всех возможных, добавляя его в набор правил. После того, как наилучшее правило найдено, алгоритм удаляет из обучающих данных, покрываемые им примеры и повторяет процедуру поиска наилучшего правила для оставшихся данных. Когда алгоритм не может выделить наилучшее правило для текущего набора данных или примеры в обучающимся наборе данных заканчиваются, он завершает работу. Алгоритм CN2 Rule induction объединяет в себе идеи алгоритмов AQ и ID3. Таким образом, алгоритм CN2 Rule induction создает набор правил подобный алгоритму AQ и может обрабатывать зашумленные данные как алгоритм ID3.

В настоящей работе обучающиеся данные поступают из виджета Color в виджет «CN2 Rule induction» в котором строится алгоритм обучения для индукции правил. В настройках алгоритма можно установить необходимое имя и выбрать упорядоченный или неупорядоченный порядок правил. В первом случае вызывается упорядоченный список правил, то есть в заголовке списка правил назначается класс – большинства. При втором варианте вызывается неупорядоченный набор правил, следовательно правила изучаются индивидуально для каждого класса. В данной работе был выбран упорядоченный набор правил. Также CN2 Rule Viewer дает возможность выбрать какой алгоритм покрытия будет использоваться при анализе: эксклюзивный или взвешенный. Эксклюзивный алгоритм покрытия, после охвата обучающего примера, удаляет его из дальнейшего рассмотрения. Взвешенный алгоритм покрытия, после покрытия примера обучения, уменьшает его вес, что в свою очередь уменьшает его влияние на дальнейшие повторение алгоритма. В нашей работе был выбран эксклюзивный алгоритм покрытия. Далее в настройках виджета выбирается «правило поиска». Устанавливается мера для оценки найденных гипотез: энтропия - мера неопределенности системы, точность Лапласса - мера оценки распределения случайной величины при приближенной заданной плотности вероятности или же WRAcc - взвешенная относительная точность. Выбирается «ширина луча», то есть отслеживается фиксированное количество лучей (альтернатив) при запоминании лучшего правила. В данной работе была выбрана энтропия в качестве меры оценки и ширину луча установили – 10. В настройках виджета необходимо выбрать «минимальное покрытие правил» и «максимальную длину правил». Минимальное покрытие правила охватывает минимально установленное необходимое количество примеров. Максимальная длина правила может

сочетать максимально установленное количество условий. В нашей работе было установлено значение 1, как минимальное покрытие и значение 10, как максимальная длина правила.

Визуализацию работы алгоритма CN2 Rule induction отображает виджет CN2 Rule Viewer (Рис. 17).



	IF conditions		THEN class	Distribution	Probabilities [%]	Quality	Length
0	Cholesterol≤126.0 AND ChestPain=TA	→	HeartDisease=1	[0, 5]	14 : 86	-0.00	2
1	ChestPain=ASY AND MaxHR≥190.0	→	HeartDisease=1	[0, 5]	14 : 86	-0.00	2
2	ChestPain=TA AND MaxHR≥190.0	→	HeartDisease=0	[3, 0]	80 : 20	-0.00	2
3	RestingBP≤98.0 AND Age≥55.0	→	HeartDisease=1	[0, 4]	17 : 83	-0.00	2
4	ChestPain=ASY AND MaxHR≥182.0	→	HeartDisease=0	[21, 0]	96 : 4	-0.00	2
5	Age≥71.0 AND Oldpeak≥3.0	→	HeartDisease=1	[0, 3]	20 : 80	-0.00	2
6	Cholesterol≤85.0 AND Cholesterol≥85.0	→	HeartDisease=0	[1, 0]	67 : 33	-0.00	2
7	ST_Slope=Down AND Age≥75.0	→	HeartDisease=0	[5, 0]	86 : 14	-0.00	2
8	Age≥77.0	→	HeartDisease=1	[0, 4]	17 : 83	-0.00	1
9	ChestPain=TA AND Age≥71.0	→	HeartDisease=1	[0, 6]	12 : 88	-0.00	2
10	ST_Slope=Down AND Age≥67.0	→	HeartDisease=1	[0, 37]	3 : 97	-0.00	2
11	ChestPain=TA AND Age≥67.0 AND Age≥68.0	→	HeartDisease=0	[21, 0]	96 : 4	-0.00	3
12	Oldpeak≤-0.5 AND ChestPain=ASY	→	HeartDisease=1	[0, 7]	11 : 89	-0.00	2
13	Age≥71.0 AND Cholesterol≥265.0	→	HeartDisease=0	[9, 0]	91 : 9	-0.00	2
14	Age≥69.0 AND FastingBS≠0	→	HeartDisease=1	[0, 14]	6 : 94	-0.00	2
15	Cholesterol≤100.0 AND FastingBS≠0 AND Sex=F	→	HeartDisease=1	[0, 5]	14 : 86	-0.00	3
16	ChestPain≠TA AND Oldpeak≥3.7	→	HeartDisease=1	[0, 27]	3 : 97	-0.00	2
17	Oldpeak≥4.0	→	HeartDisease=0	[34, 0]	97 : 3	-0.00	1
18	RestingBP≥200.0	→	HeartDisease=1	[0, 3]	20 : 80	-0.00	1
19	ST_Slope=Up AND RestingBP≥192.0	→	HeartDisease=1	[0, 3]	20 : 80	-0.00	2
20	RestingBP≥192.0	→	HeartDisease=0	[4, 0]	83 : 17	-0.00	1
21	ST_Slope=Down AND RestingBP≥170.0	→	HeartDisease=1	[0, 13]	7 : 93	-0.00	2
22	RestingBP≥164.0 AND RestingBP≥190.0	→	HeartDisease=0	[1, 0]	67 : 33	-0.00	2
23	RestingBP≥164.0 AND RestingBP≥185.0	→	HeartDisease=1	[0, 1]	33 : 67	-0.00	2
24	ST_Slope=Up AND RestingBP≥170.0	→	HeartDisease=0	[14, 0]	94 : 6	-0.00	2
25	Cholesterol≤100.0 AND FastingBS≠0 AND ChestPain=ASY	→	HeartDisease=1	[0, 55]	2 : 98	-0.00	3
26	Oldpeak≥3.8	→	HeartDisease=1	[0, 3]	20 : 80	-0.00	1
27	ST_Slope=Up AND Cholesterol≥354.0	→	HeartDisease=0	[15, 0]	94 : 6	-0.00	2
28	MaxHR≥182.0 AND RestingECG=LVH	→	HeartDisease=0	[10, 0]	92 : 8	-0.00	2
29	ST_Slope=Up AND MaxHR≥194.0	→	HeartDisease=1	[0, 4]	17 : 83	-0.00	2
30	MaxHR≥182.0 AND ExerciseAngina≠No	→	HeartDisease=1	[0, 4]	17 : 83	-0.00	2
31	MaxHR≥182.0 AND Cholesterol≥340.0	→	HeartDisease=0	[1, 0]	67 : 33	-0.00	2
32	Age≤31.0 AND Sex=F	→	HeartDisease=0	[1, 0]	67 : 33	-0.00	2
33	MaxHR≥182.0 AND RestingECG≠Normal AND Age≥37.0	→	HeartDisease=1	[0, 13]	7 : 93	-0.00	3
34	MaxHR≥202.0	→	HeartDisease=1	[0, 5]	14 : 86	-0.00	1
35	Age≤31.0 AND ChestPain=ASY	→	HeartDisease=1	[0, 1]	33 : 67	-0.00	2

Рис. 17. Визуализация алгоритма индукции правил

На *рис. 17* показан список правил, созданных алгоритмом CN2 Rule induction. Рассматривая работу алгоритма, на примере самого первого правила очевидно, что оно состоит из двух атрибутов: холестерин, который выше или равный 126 мм/л. и типичной стенокардией. По данному правилу была предсказана положительная вероятность сердечной недостаточности. Достоверность правила составляет 86%. По данному правилу было покрыто 5 экземпляров данных, которые в последующем были исключены из дальнейшего

анализа. Далее алгоритм создал следующее наилучшее правило, состоящее также из двух атрибутов: бессимптомной стенокардией и максимальной частотой сердечных сокращений меньше или равной 190, с положительным прогнозом сердечной недостаточности, достоверность которого 86%. По данному правилу было покрыто 5 экземпляров данных, которые в последующем аналогично, были исключены из дальнейшего анализа. Таким образом, алгоритм CN2 Rule induction создал 387 правил, по которым обработал весь набор обучающихся данных.

3.5. Результаты прогнозирования алгоритмов

Прогностические решения обученных алгоритмов отображаются в виджете Predictions. В данный виджет поступают уже обученные алгоритмы и данные из тестовой базы. Следовательно, виджет Predictions визуализирует прогнозы, для тестовой части данных, на основе обученных алгоритмов (Рис. 18).

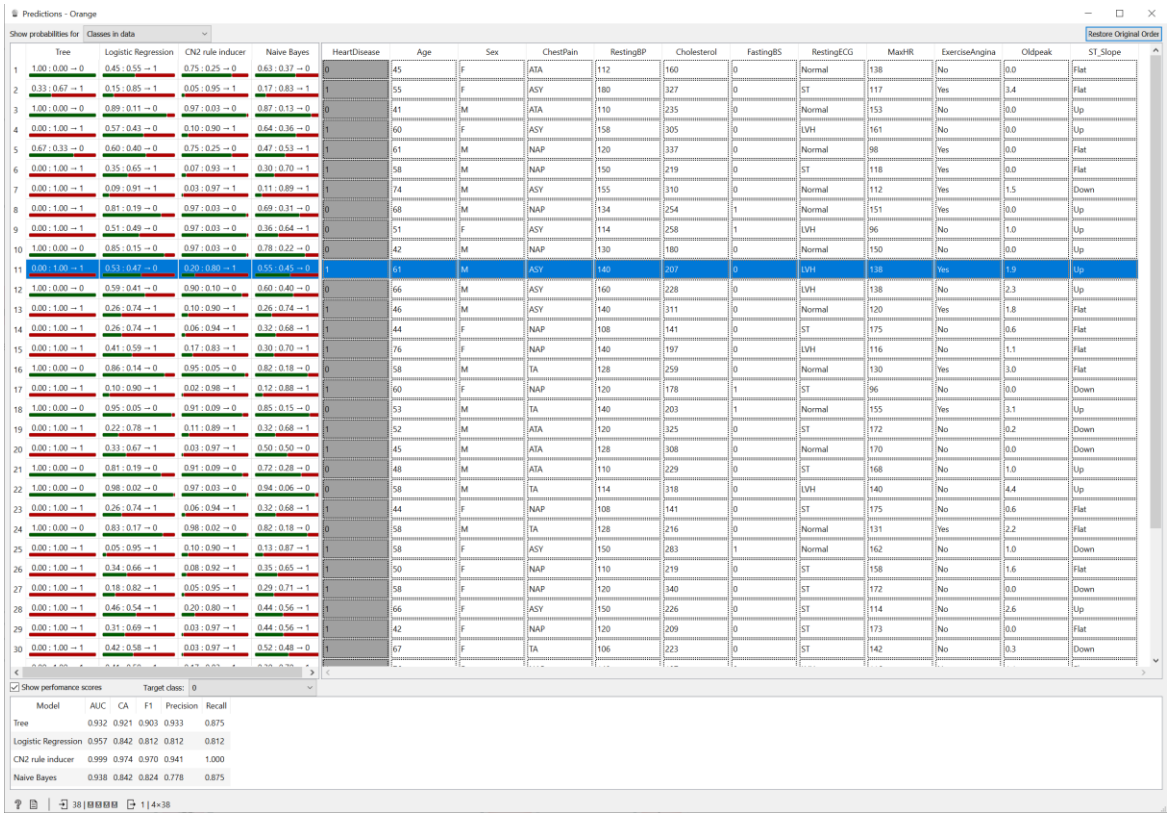


Рис. 18. Прогностические решения алгоритмов в виджете Predictions

Виджет показывает прогнозы алгоритмов, выраженные в процентном соотношении и достоверный диагноз, что дает возможность произвести сравнения между всеми задействованными алгоритмами и сопоставить точность прогноза истинному диагнозу.

При установке флажка около параметра Show performance scores, отображаются коэффициенты точности прогностических моделей на основе тестовой базы (*Рис. 19а*) и (*Рис. 19б*), что позволяет сравнить прогностические модели между собой. Параметр Target

class позволяет просмотреть коэффициенты точности в зависимости от интересующего параметра конечного результата.

Show perfomance scores Target class: 1

Model	AUC	CA	F1	Precision	Recall
Tree	0.932	0.921	0.933	0.913	0.955
Logistic Regression	0.957	0.842	0.864	0.864	0.864
CN2 rule inducer	0.999	0.974	0.977	1.000	0.955
Naive Bayes	0.938	0.842	0.857	0.900	0.818

Рис. 19а. Коэффициенты вероятности прогностических моделей при положительном прогнозе сердечной недостаточности.

Show perfomance scores Target class: 0

Model	AUC	CA	F1	Precision	Recall
Tree	0.932	0.921	0.903	0.933	0.875
Logistic Regression	0.957	0.842	0.812	0.812	0.812
CN2 rule inducer	0.999	0.974	0.970	0.941	1.000
Naive Bayes	0.938	0.842	0.824	0.778	0.875

Рис. 19б. Коэффициенты вероятности прогностических моделей при отрицательном прогнозе сердечной недостаточности.

Анализируя рассчитанные вероятности прогностических моделей по сердечной недостаточности и ее отсутствия на основе тестовой базы, очевидно, что самую высокую точность показывают алгоритмы Tree и CN2 Rule induction с показателями точности (CA) 0,921 и 0,974 соответственно. Алгоритмы Logistic Regression и Naive Bayes транслирует более низкие коэффициенты точности – 0,842 и 0,842 соответственно. Аналогичные показатели коэффициентов точности формируются у прогностических моделей и на отрицательный прогноз по сердечной недостаточности. Более подробно об коэффициентах точности и эффективности прогностических моделей написано ниже.

3.6. Тестирование алгоритмов принятия решений

После обучения прогностических моделей, мы переходим к этапу тестирования, на котором используем тестовые данные для проверки точности прогнозов модели [54]. Эффективность прогностических моделей можно посмотреть в виджете Test and Score. Данный виджет дает возможность получить представление о качестве и точности прогноза и определить в пользу какой модели лучше сделать выбор.

Для анализирования эффективности прогностических моделей, в виджет поступает полный набор собранных данных по сердечной недостаточности, состоящих из 2788 экземпляров данных. Наряду с этим в виджет заводим все прогностические модели, не имеющие связи с другими виджетами (Рис. 20).

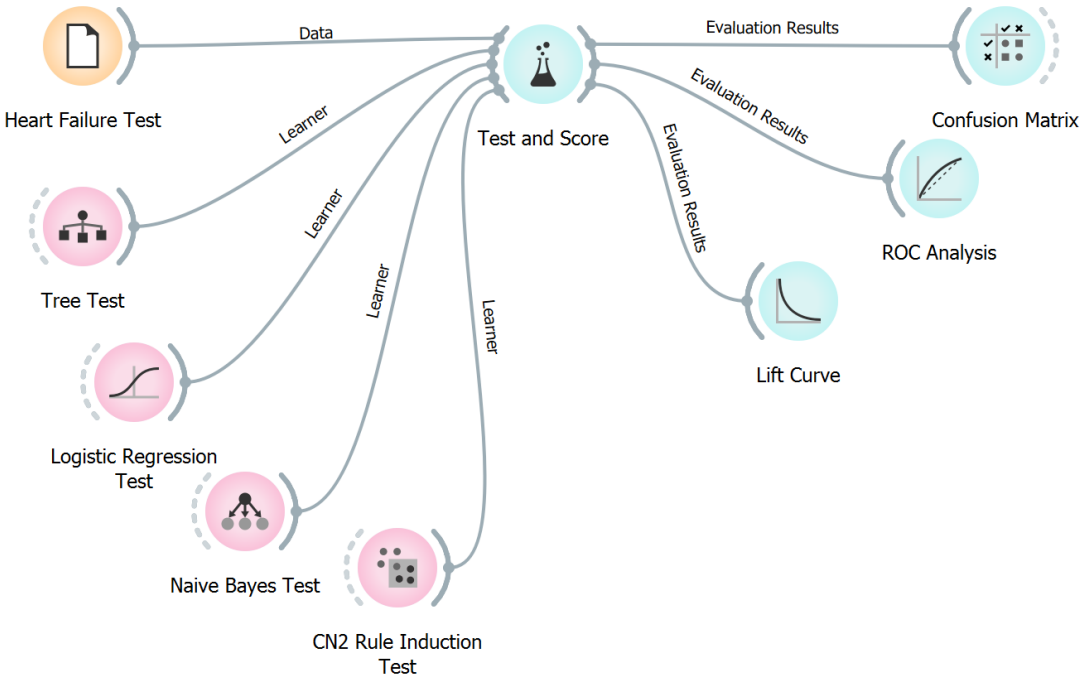


Рис. 20. Анализ эффективности прогностических моделей.

В наборе данных у атрибута HeartDisease устанавливаем роль target. Виджет тестирует алгоритмы обучения с использованием набора данных и выводит таблицу с показателями производительности алгоритмов, результатами оценки и сравнения алгоритмов между собой (*Рис. 21*). В таблице Evaluation Results отображены прогностические модели и показатели по следующим параметрам:

AUC – показывает площадь под кривой ROC. Чем ближе данные площади к «1», тем более точны результаты тестирования.

CA – показывает долю правильно классированных экземпляров данных.

F1 – показывает сбалансированное среднее значение точности и полноты. Чем ближе показатель к 1, тем более точные и полные результаты тестирования.

Precision – показывает точность, то есть долю истинно положительных результатов среди экземпляров данных, классифицированных как положительные.

Recall – отзыв, показывает долю истинно положительных результатов среди всех экземпляров данных.

Настройки таблицы позволяют просмотреть показатели в зависимости от интересующего типа. Можно увидеть общие показатели эффективности, или же показатели

в зависимости от конечного результата, в нашем случае положительного или отрицательного прогноза по сердечной недостаточности.

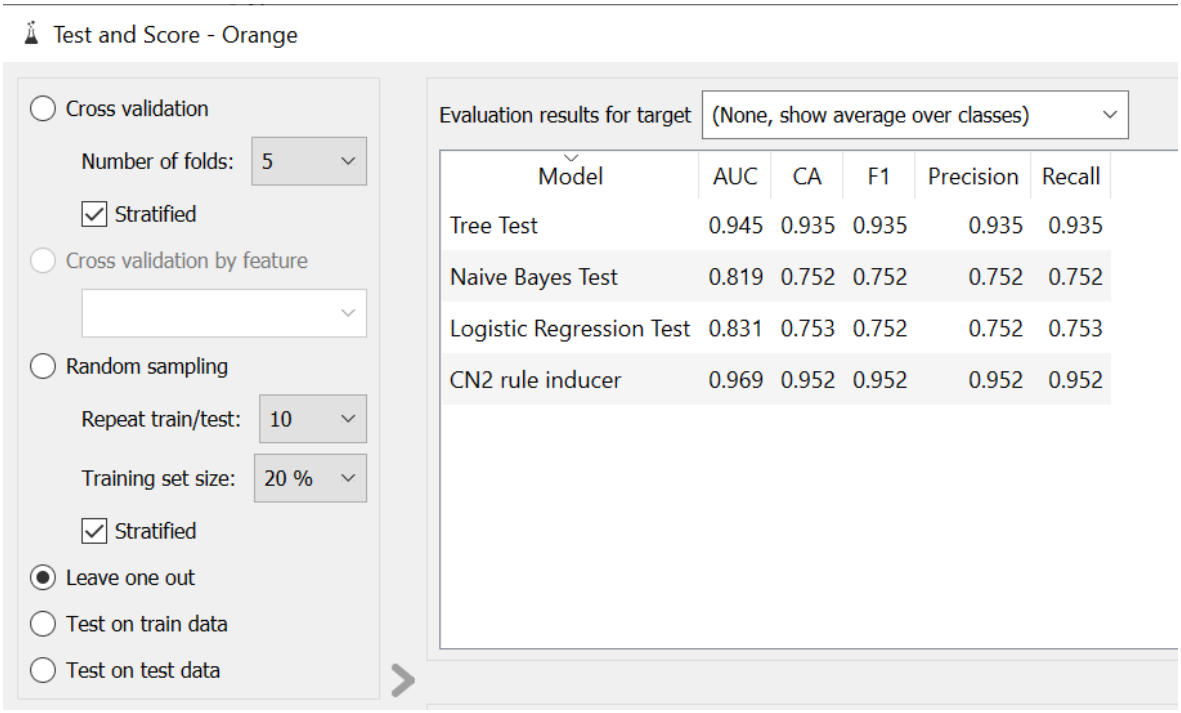


Рис. 21. Коэффициенты тестирования алгоритмов

Настройки виджета позволяют проверить эффективность прогностических моделей по различным методам выборки:

1. Cross validation.

При данном методе выборки происходит перекрестная проверка алгоритмов. Перекрестная проверка – это метод повторной выборки. Метод реализует разные части набора данных для тестирования и обучения модели. При данном методе выборки, виджет Test and Score формирует еще одну таблицу – Model Comparison by, которая выводит результаты попарного сравнения данных по выбранной оценки показателей. Число в таблице показывает разницу между алгоритмами в строках и столбцах. Более высокое число показывает более высокую разницу между процессами выборки алгоритмов.

2. Cross validation by feature - данный метод выборки выполняет перекрёстную проверку по признаку из мета-признаков.

3. Random sampling – данный метод выборки случайным образом разбивает данные, по заданным пропорциям, на обучающую и тестовую группу. Данная процедура повторяется определенное количество раз.

4. Leave one out – метод исключения одного. Выборка происходит также случайным образом, но удерживает по одному экземпляру данных за раз, создавая модель из оставшихся данных, которые в дальнейшем классифицирует. В нашей работе, для тестирования прогностических моделей использован данный метод выборки, так как метод очень надежен

и стабилен. Единственным недостатком данного метода выборки, является его медлительность.

5. Test on train data – данный метод использует весь набор экземпляров данных сначала для обучения, после для тестирования. Метод ненадежен и дает неверные результаты.

6. Test on test data – данный метод выборки позволяет использовать для тестирования наборы данных из другого источника.

Виджет Test and Score дает возможность выбрать класс для показателей эффективности прогностических моделей и установить баллы для попарного сравнения алгоритмов в таблице Model Comparison.

Рассматривая показатели в виджете Test and Score, очевидно, что алгоритмы Tree и CN2 Rule induction показывают самые высокие коэффициенты точности прогноза. Точность прогноза алгоритма CN2 Rule induction составляет 0,941 и это самый высокий показатель точности среди исследуемых прогностических моделей. Точность алгоритма Tree равна 0,923, что немного меньше, чем у правил индукции CN2, но все же достаточно высок в достоверности прогнозируемых результатов. Алгоритмы Logistic Regression и Naive Bayes показали наиболее низкие показатели точности, подразумевающие большой процент погрешности прогнозируемых результатов. Следовательно, для прогнозирования сердечной недостаточности по клинико-диагностическим признакам, данные прогностические модели наименее предпочтительны.

Визуализацию прогностических моделей отображают виджеты Confusion Matrix, ROC Analysis и Lift Curve.

Confusion Matrix

В виджете Confusion Matrix (Рис. 22) (матрица путаницы) показаны пропорции между прогнозируемой и фактической диагностикой сердечной недостаточности по каждому алгоритму. При выполнении прогнозов в матрице путаницы возникают четыре типа результатов:

1. Истинные положительные результаты – это когда предсказывается, что наблюдение принадлежит классу, и оно действительно принадлежит этому классу. Отображается в нижнем синем квадрате.

2. Истинные отрицательные результаты — это когда предсказывается, что наблюдение не принадлежит классу, и оно на самом деле не принадлежит классу. Отображается в верхнем синем квадрате.

3. Ложноположительные результаты– это когда предсказывается, что наблюдение относится к классу, хотя на самом деле не принадлежит к этому классу. Отображается в нижнем красном квадрате.

4. Ложноотрицательные результаты – возникают, когда предсказывается, что наблюдение не принадлежит к классу, хотя на самом деле оно принадлежит. Отображается в верхнем красном квадрате.

На рисунке, в сиреневых прямоугольниках отображено число правильно интерпретированных алгоритмом экземпляров данных. В розовых прямоугольниках показано число неверно интерпретированных экземпляров данных. Анализ матрицы путаницы можно посмотреть в количественном или процентном соотношении.

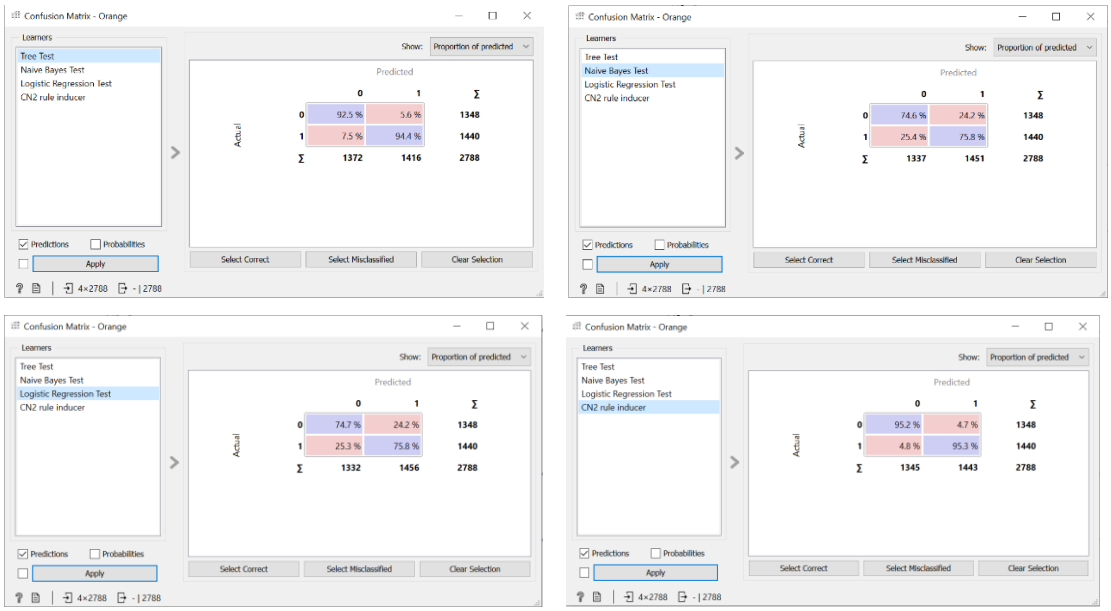


Рис. 22. Матрица путаницы прогностических моделей.

При изучении матрицы путаницы наглядно видно истинно положительные и отрицательные результаты прогностических моделей. В матрице путаницы также можно увидеть показатели ложноположительных и ложноотрицательных результатов прогностических моделей. У алгоритмов Logistic Regression и Naive Bayes данные показатели самые высокие. Следовательно данные алгоритмы не эффективны для прогнозирования подобного рода задач.

«ROC Analysis»

Результатом работы инструмента ROC-анализ являются графические результаты тестирования прогностических моделей (Рис. 23). Графики меняются в зависимости от того какой целевой класс был выбран. Чем ближе кривая к оси Y (TR Rate), и к верхней границе пространства ROC, тем точнее алгоритм. Ось Y (TR Rate) показывает соотношение

правильно интерпретированных экземпляров данных, а ось X (FP Rate) – соотношение ложно интерпретированных данных.

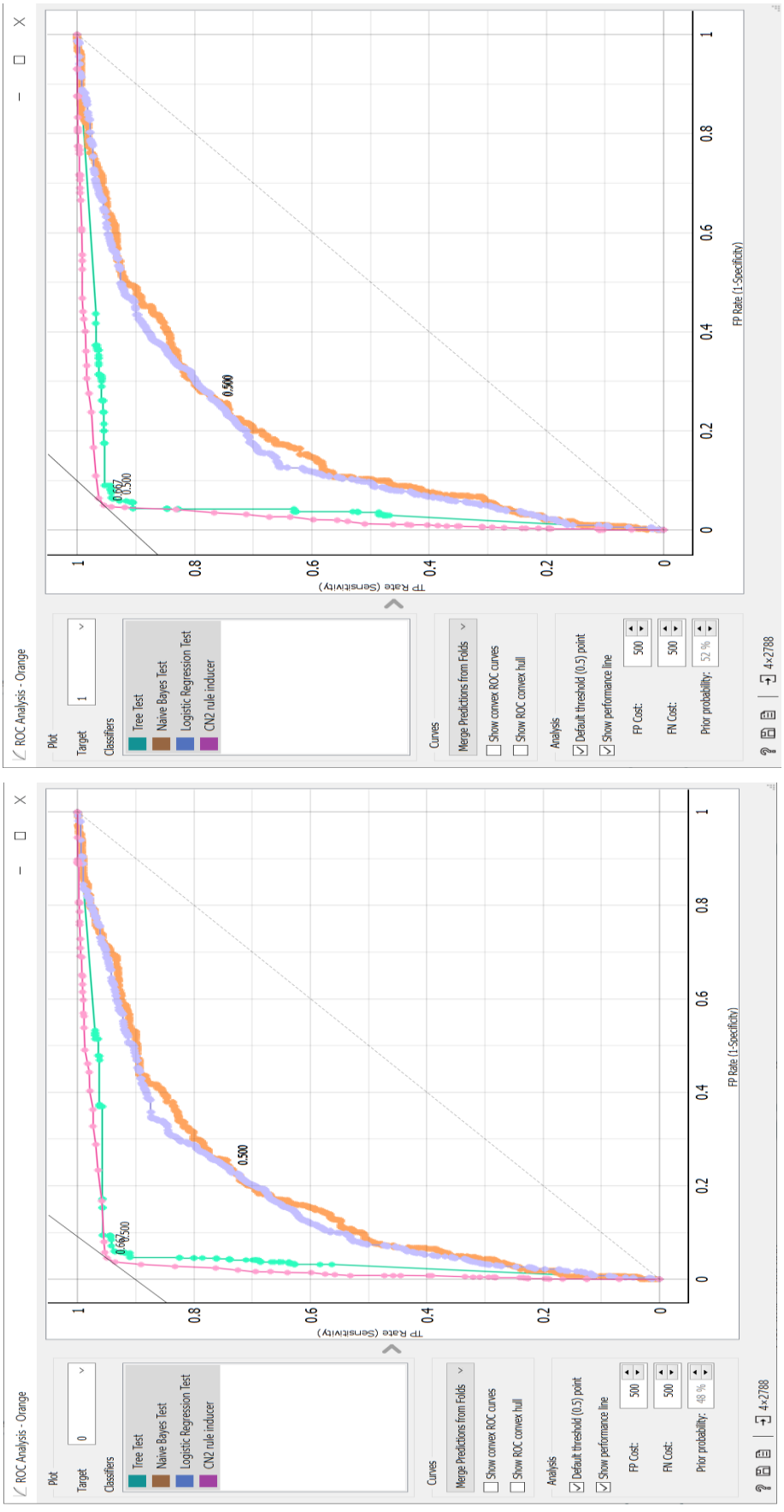


Рис. 23. Виджет ROC Analysis

На графике ROC-кривой, где искомое значение установлено 1, то есть график по положительной диагностике сердечной недостаточности, видно, что самые высокие показатели по оси Y (TR Rate) показывают алгоритмы CN2 rule inducer и Tree соответственно. На графике также обозначено, что у этих алгоритмов самые низкие показатели неверно интерпретированных данных, так как обе кривые достаточно близко расположены к верхней границе пространства ROC. Соответственно площадь под кривой ROC у данных алгоритмов больше, следовательно они более эффективны при прогнозировании. Алгоритмы Naïve Bayes и Logistic Regression показывают кривые с наиболее высокими ложными прогнозами. Площадь под кривой ROC у них меньше, таким образом, они менее эффективны.

На графике ROC-кривой, где искомое значение установлено 0, отображено аналогично построенные кривые. Кривые алгоритмов CN2 rule inducer и Tree также наиболее точно интерпретируют отсутствие сердечной недостаточности, как и ее наличие. Кривые алгоритмов Naïve Bayes и Logistic Regression показывают наименьшее соотношение правильно интерпретированных данных по отсутствию заболевания, как и по его наличию.

Выводы показанные ROC-анализом аналогично матрице путаницы, только отображены графически.

Lift Curve

Кривая производительности (Рис. 24) показывает кривые для анализа доли истинно положительных экземпляров данных (ось X) по отношению к количеству записей, которые классифицируются как положительные (ось Y). Кривая показывает измерение производительности выбранного алгоритма по сравнению со случайным алгоритмом

В ходе рассмотрения кривых производительности очевидно, что алгоритм CN2 rule inducer остается все также более эффективным по сравнению с другими прогностическими моделями. У данного алгоритма наибольшие показатели отношения между количеством случаев, которое алгоритм предсказал как положительные и теми случаями, которые действительно являлись истинными при диагностики сердечной недостаточности. При изучении кривой производительности алгоритма CN2 rule inducer аналогично самые высокие показатели отношения между количеством случаев, которые алгоритм предсказал как отрицательные и теми случаями, которые действительно показывали отсутствие сердечной недостаточности. Кривая производительности алгоритма Tree незначительно уступает индукции правил CN2. Единственным отличием кривой производительности алгоритма Tree от кривых других прогностических моделей является то, что кривая не доходит до оси Y. Это связано с тем, что данный алгоритм делает прогноз распределяя данные на узлы и подузлы,

делая анализ максимально простым и быстрым, данные в узлах собраны в более обширные группы.

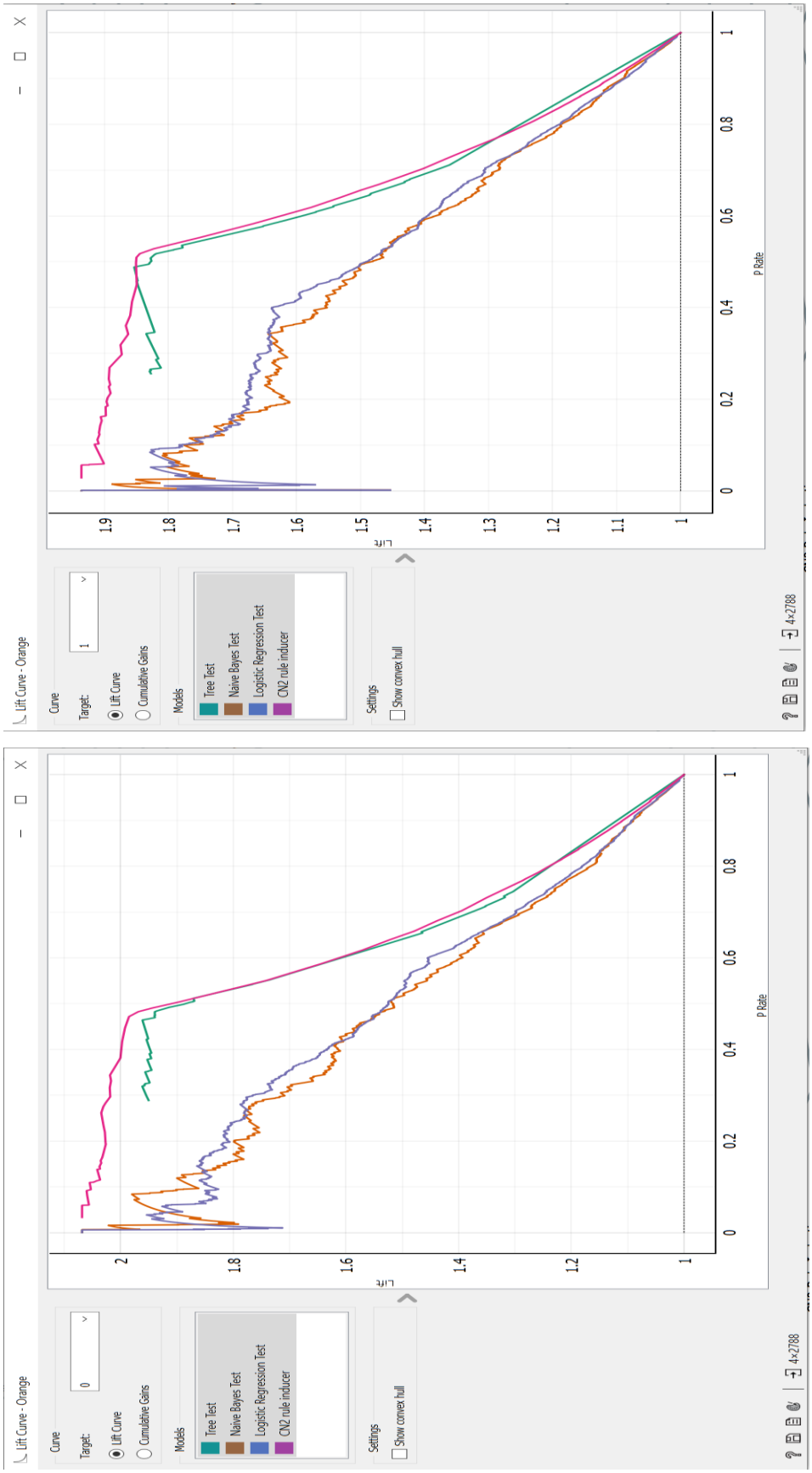


Рис. 24. Виджет Lift Curve

Кривые производительности алгоритмов Naïve Bayes и Logistic Regression уступают предыдущим алгоритмам по своим прогностическим показателям.

3.7. Визуализация прогностических данных

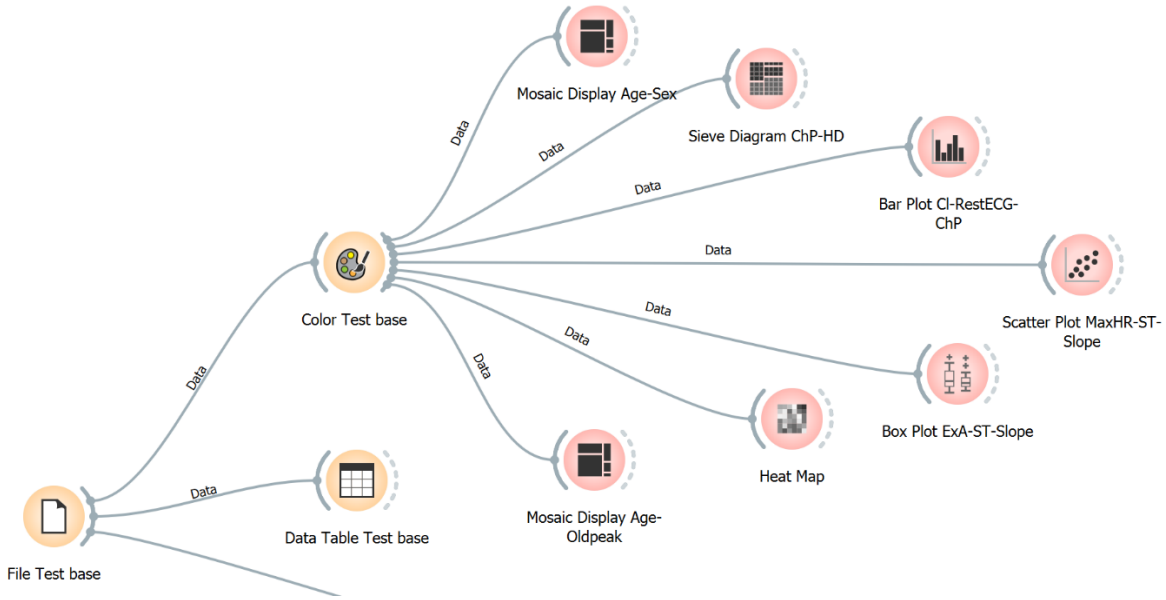


Рис. 25. Визуализация прогностических данных.

Выполнение анализа больших объемов данных - достаточно сложная задача. Для лучшего интерпретирования данных мы их визуализировали, так как физиологически восприятие визуальной информации является более понятной и удобной для человека. Этот подход дает представление о том, что делает структура и алгоритмы [34, 52]. Визуализированные схематически и графически данные улучшают восприятие и анализ данных. Для наглядного интерпретирования, в нашей работе представлена визуализация тестовых экземпляров данных по пациентам. (Рис. 25) [7].

3.7.1. Визуализация Mosaic Display

Первоначально нами визуализировались соотношения данных возраста к полу пациентов (*Рис. 26*) в мозаичном дисплее, так как в основном заболеваниям сердца, а в частности, сердечной недостаточности подвержены люди пожилого возраста. Из представленного контингента у женщин в возрасте 60,5 лет и старше в 100% случаях имелись проблемы с сердечно-сосудистой системой (сердечная недостаточность). При анализе лиц мужского пола в этом же возрастном диапазоне в 50% случаях наблюдалась сердечная недостаточность и в 50% серьезных проблем с сердцем не было выявлено. В возрастном диапазоне от 56,5 до 60,5 лет, женщины в 4 случаях из 5 имели сердечную недостаточность, а мужчины в 4 случаях из 5 не имели. В возрастном диапазоне от 47 до 56,5

лет сердечная недостаточность была выявлена у женщин в 2 случаях из 4, а у мужчин только в 1 случае из 3. При возрасте до 47 лет, сердечная недостаточность имелась в 4 случаях из 5 у женщин и в 2 случаях из 4 у мужчин.

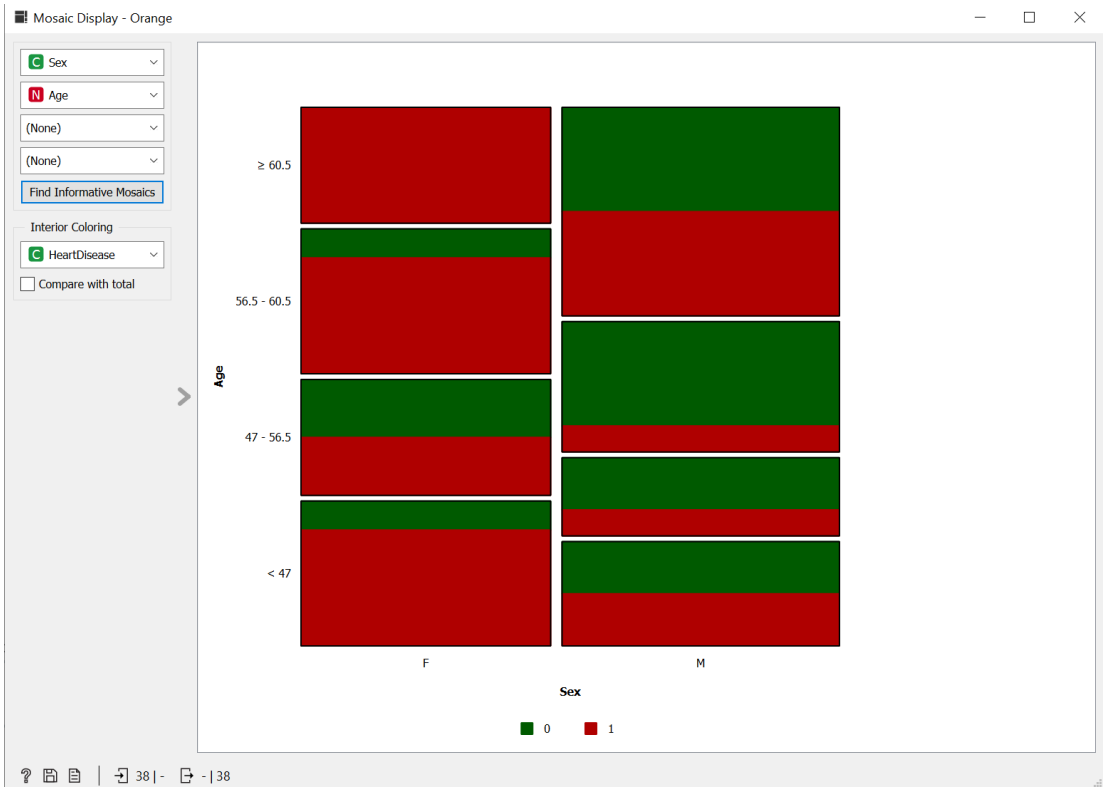


Рис. 26. Визуализация данных в мозаичном дисплее

Рассмотрев результаты визуализации, очевидно, что у женщин более чаще выявлялась сердечная недостаточность, чем мужчин. Вероятнее всего этому способствовали период беременности и родов, а также различные бытовые факторы.

3.7.2. Визуализация Sieve Diagram.

В основе ситовой диаграммы лежит визуализация частоты наблюдаемых обстоятельств (observed) и сравнение их с ожидаемыми частотами (expected). Площадь каждого прямоугольника пропорциональна ожидаемой частоте, а наблюдаемая частота показана количеством квадратов в каждом прямоугольнике. Разница между наблюдаемой и ожидаемой частотой отображается как плотность разноцветной штриховки.

Ситовая диаграмма (*Рис. 27*) в наших исследованиях отразила зависимость ожидаемой и наблюдаемой частоты типа боли в груди у пациентов с сердечной недостаточности и ее отсутствию. Из 38-ми наблюдаемых случаев, сердечная недостаточность не была выявлена у 16 пациентов. Типичная стенокардия (TA) была выявлена у 9 из 38 пациентов. По полученным данным ситовая диаграмма показала, что отсутствие сердечной недостаточности при типичной стенокардии ожидалось в 10%, а фактически наблюдалась у

21% случаях, то есть сердечная недостаточность встречалась у 8 человек, что свидетельствует о положительном отклонении. Соответственно штриховка наблюдаемой частоты внутри квадрата ожидаемой частоты - более частая.

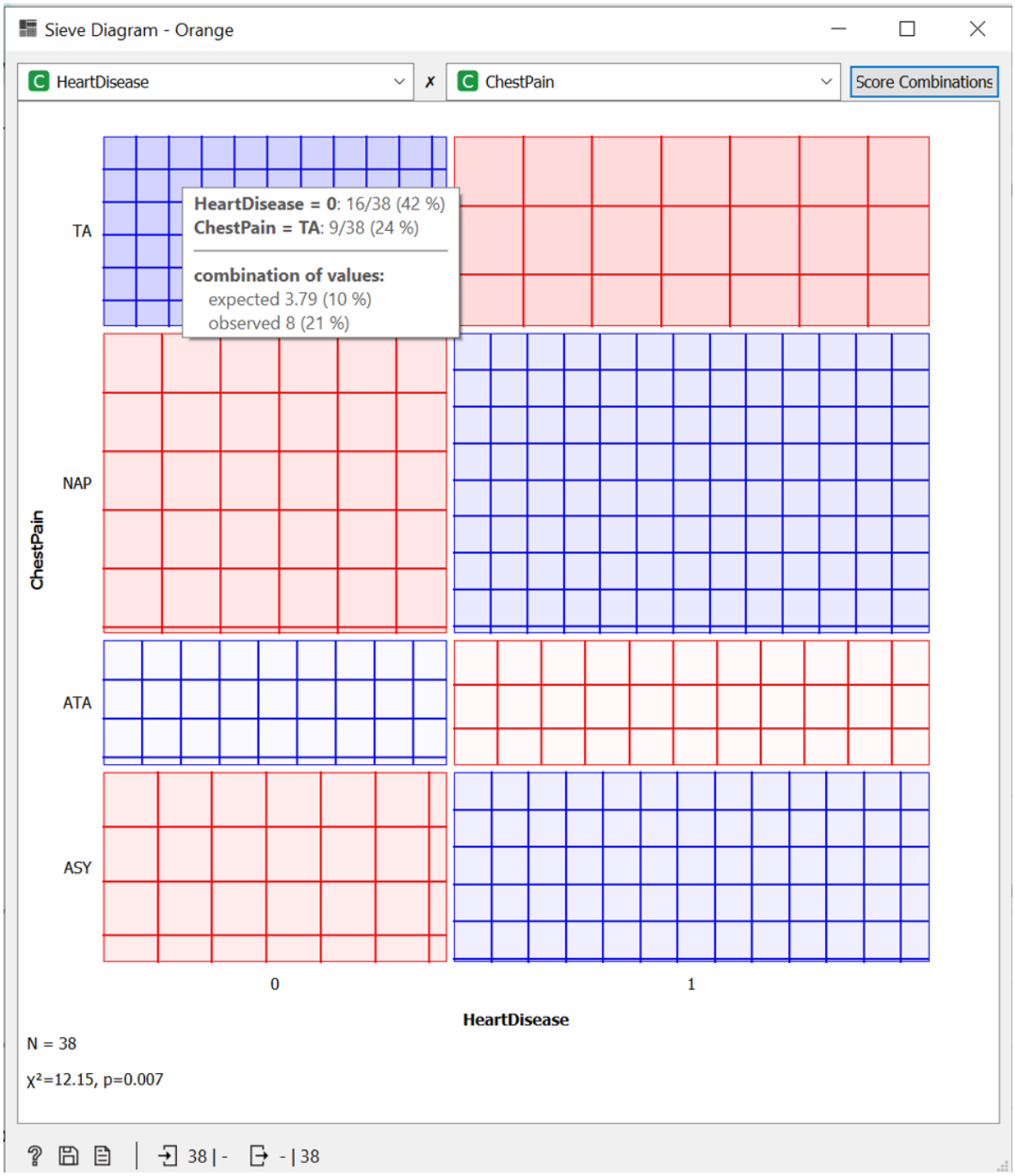


Рис. 27. Визуализация данных в Sieve Diagram.

При наличие сердечной недостаточности у 22 пациентов из 38 и типичной стенокардии в 9 случаях из 38, ситовая диаграмма показывает, что наличие сердечной недостаточности ожидалась в 14%, а фактически наблюдалась в 3% случаях, что также является положительным отклонением. Соответственно, штриховка наблюдаемой частоты сердечной недостаточности внутри квадрата ожидаемой частоты случаев была более редкая.

При неангинозной боли (NAP) в 14 случаях из 38 и отсутствии сердечной недостаточности в 16 случаях из 38, ситовая диаграмма показывает, что отсутствие заболевания ожидалось в 16% случаях, а фактически наблюдалось в 8%, что также является положительным отклонением. При неангинозной боли в 14 случаях из 38 и наличии сердечной недостаточности в 22 случаях из 38, ситовая диаграмма продемонстрировала, что наличие заболевания ожидалось в 21% случаев, но фактически наблюдалось в 29%, что может свидетельствовать об отрицательном отклонении. Штриховка наблюдаемой частоты сердечной недостаточности внутри квадрата ожидаемой частоты случаев была более частая, что указывает на значительное увеличение числа диагностированной сердечной недостаточности, чем ожидалось.

При атипичной стенокардии (ATA) в 6 случаях из 38 и отсутствии диагноза сердечной недостаточности в 16 случаях из 38, ситовая диаграмма показала, что отсутствие заболевания ожидалось в 7% случаях, а фактически наблюдалось в 8%, что являлось положительным отклонением. При атипичной стенокардии в 6 случаях из 38 и диагностировании сердечной недостаточности, ситовая диаграмма продемонстрировала, что наличие заболеваемости ожидалось в 9% случаях, а фактически наблюдалось в 8%, что является положительным отклонением.

При бессимптомном течении стенокардии (ASY) в 9 случаях из 38 и отсутствии диагноза сердечная недостаточность в 16 случаях из 38, ситовая диаграмма показала, что отсутствие заболевания ожидалось в 10% случаях, а фактически наблюдалось в 5%, что указывало на отрицательное отклонение. При бессимптомном течении стенокардии в 9 случаях из 38 и наличии диагноза сердечной недостаточности, ситовая диаграмма показывала, что присутствие заболеваемости ожидалось в 14% случаях, а фактически наблюдалось в 18%, что также является отрицательным отклонением.

Изучив ситовую диаграмму, очевидно, что при бессимптомной стенокардии и неангинозной боли, сердечную недостаточность при диагностике выявляют значительно чаще ожидаемых результатов. Это может быть связано с тем, что в первом случае боли нет и пациент не подозревает о развитии у него заболевания сердечно-сосудистой системы. Во втором, болезненные ощущения в области самого сердца пациент не чувствует, а болевые симптомы локализуются в другой части тела, например: в руке, спине или же в боку. Соответственно, не все пациенты могут связать данный болевой синдром с болезнью сердца (сердечной недостаточностью). Следовательно, сердечную недостаточность выявляют случайным образом при прохождении клинико-диагностических исследованиях.

3.7.3. Визуализация Bar Plot.

Гистограмма «Bar Plot» показывает числовые переменные и сравнивает их с категориальными переменными. В данной работе на гистограмме отражен уровень холестерина (Cholesterol), выраженный числовыми переменными типа боли в груди (ChestPain) и показателями ЭКГ в состоянии покоя (RestingECG), выраженных категориальными переменными (*Рис. 28*).

Рис. 28. Визуализация данных в Bar Plot.

На гистограмме видно, что если ЭКГ в состоянии покоя выявило аномальное положение интервала ST, то при всех видах боли в груди сердечная недостаточность присутствовала у большинства пациентов. Уровень холестерина при этом не всегда был высокий. Вместе с тем, эти изменения наблюдались и при низком уровне холестерина.

Если анализы ЭКГ в состоянии покоя показывали норму, то при всех видах боли в груди, сердечная недостаточность присутствовали только при повышенном холестерине.

Если ЭКГ в состоянии покоя выявило гипертрофию левого желудочка, то сердечная недостаточность, при бессимптомном течении стенокардии и неангинозной боли, наблюдалась при высоком уровне холестерина (выше 190 мм/дл). При гипертрофии левого желудочка и присутствии типичной и атипичной стенокардии, сердечная недостаточность не была выявлена, хотя уровень холестерина был выше нормы.

3.7.4. Визуализация Scatter Plot.

Точечная диаграмма демонстрирует двумерную точечную визуализацию экземпляров данных. Ось X показывает уровень максимальной частоты пульса, ось Y - изменение смещения сегмента ST на ЭКГ (*Рис. 29*). Если сегмент ST имеет восходящее положение (Up), то это может свидетельствовать о наличии гипоксии, а также о ранее перенесенном инфаркте миокарда ишемии в миокарде. Если сегмент ST находится на уровне изолинии (Flat), то это указывает об отсутствии серьезных изменений в сердце. Нисходящий сегмент ST (Down) может говорить об отдаленных последствиях перенесенного инфаркта миокарда или гипоксии сердца.

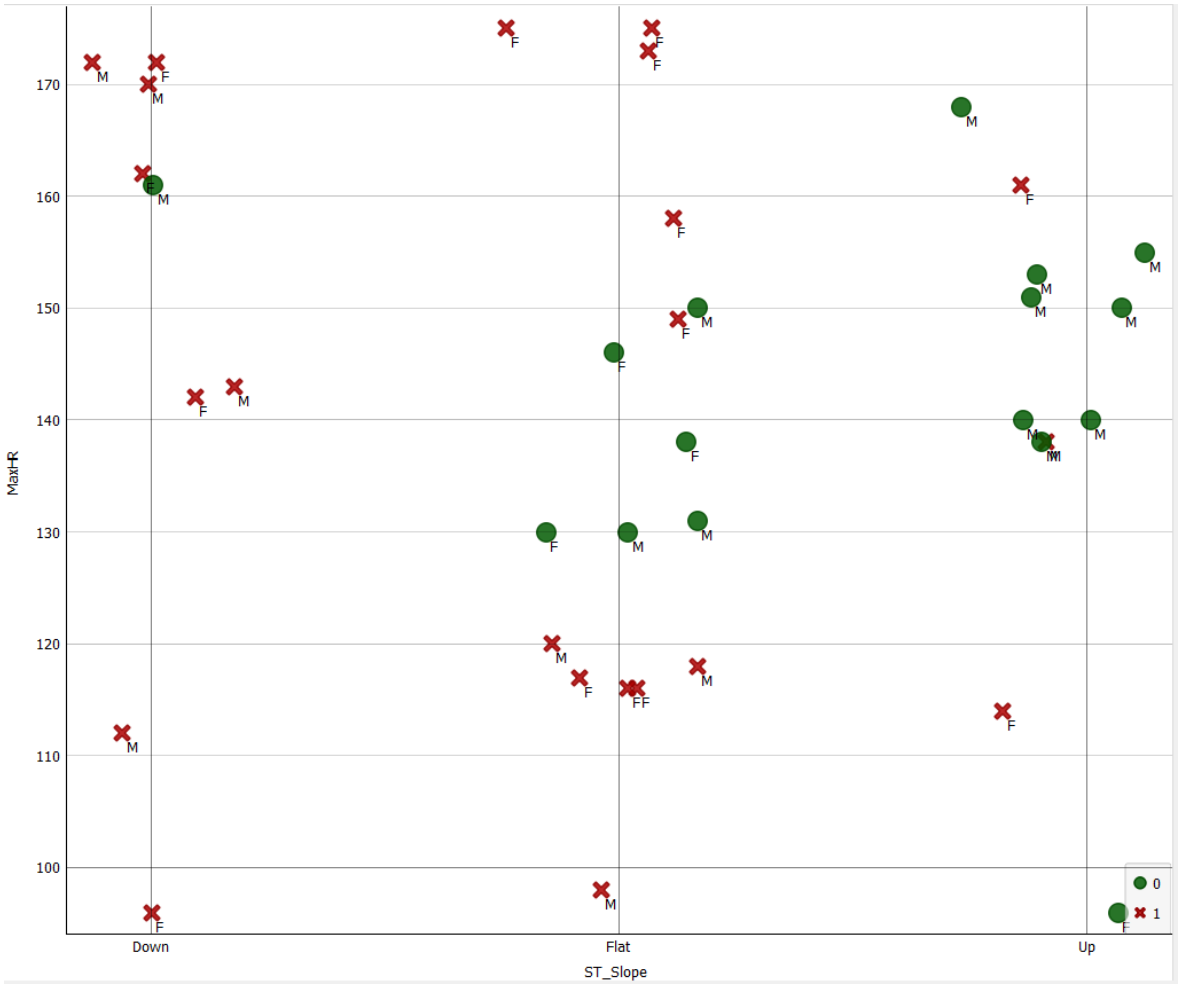


Рис. 29. Визуализация данных в Scatter Plot.

В ходе рассмотрения точечной диаграммы, очевидно, что сердечная недостаточность практически всегда наблюдалась у пациентов, перенесших ишемическую болезнь сердца. При этом частота пульса и уровень холестерина мог быть в норме.

Наличие у пациентов ишемической болезни сердца, о котором свидетельствует подъем сегмента ST, в будущем могла не приводить к сердечной недостаточности. Однако у таких пациентов в большинстве случаев присутствовала тахикардия, выше 135 единиц сердечных сокращений. Уровень холестерина изредка был высокий.

При смещении сегмента ST к изолинии, сердечная недостаточность наблюдалась у пациентов с максимальной частотой пульса до 120 и выше 145. Уровень холестерина колебался. При максимальной частоте пульса в диапазоне от 120 до 145, сердечная недостаточность приводила к развитию сердечной недостаточности у пациентов.

3.7.5. Визуализация Box Plot.

Box Plot отображает распределение значений исследуемых данных. На горизонтальной шкале визуализации показаны тестовые экземпляры данных, при этом по вертикали данные указывали о наличие или отсутствие стенокардии при физических нагрузках (ExerciseAngina). Изменения смещении сегмента ST на ЭКГ показаны разным цветовым сегментом (*Рис. 30*).

На данной визуализации можно увидеть, что у 26 из 38 пациентов отсутствовала стенокардия при физических нагрузках. Однако 8 из 26 пациентов в прошлом перенесли ишемическую болезнь сердца, 9 из 26 пациентов имели коронарную недостаточность и у 9 из 26 пациентов отмечалась тахикардия даже при минимальных физических нагрузках.

Стенокардия при физических нагрузках присутствовала у 12 из 38 пациентов, один из них в прошлом перенес инфаркт миокарда, 8 из 12 пациентов имели коронарную недостаточность, а у 3 из 12 было увеличенно число сердечных сокращений.

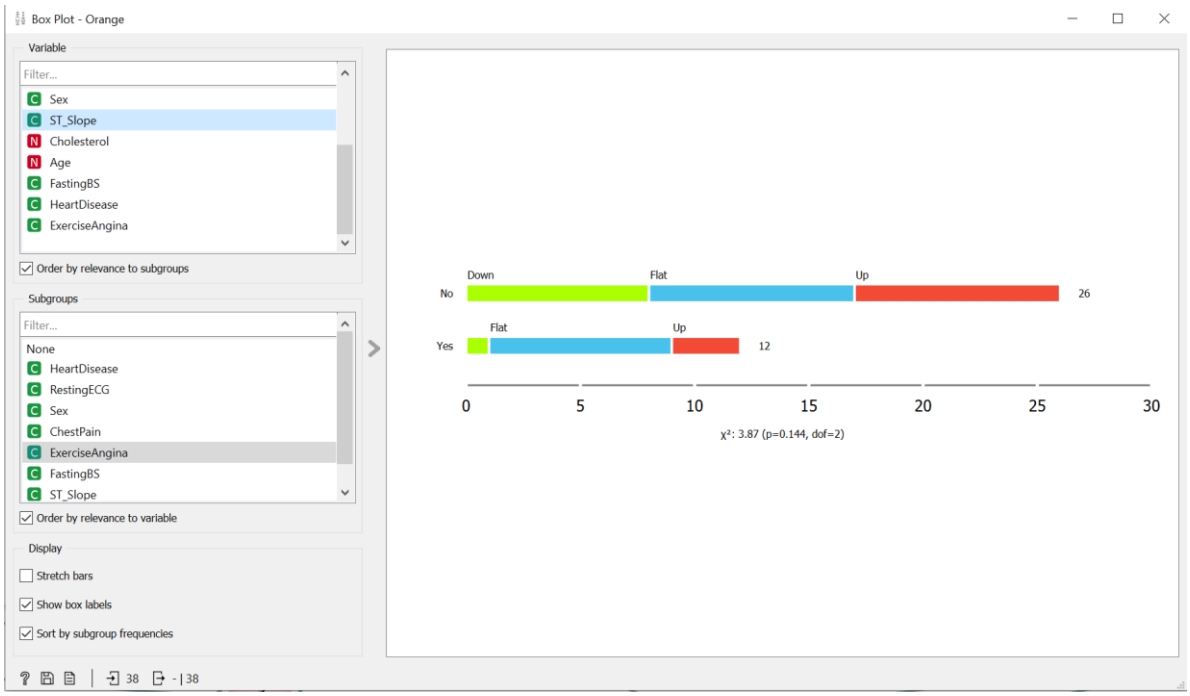


Рис. 30. Визуализация данных в Box Plot.

Изучив полученные данные визуализации, нельзя сказать, что наличие в анамнезе пациента перенесенного инфаркта миокарда, тахикардии и коронарной недостаточности может вызвать боли при любых физических нагрузках.

3.7.6. Визуализация Heat Map.

Тепловая карта – это графический метод визуализации значений атрибутов в двухсторонней матрице. Данный метод работает только с числовыми значениями. Данные числовые значения обычно представлены выбранной цветовой палитрой и демонстрируют разбросанный диапазон полученных результатов (слабые и более выраженные).

На *рис. 31* отображены пять числовых значений, связанных данных с наличием или отсутствием сердечной недостаточности. В правой части тепловой карты можно увидеть категориальные значения данных, в частности, указывающие на тип боли в груди.

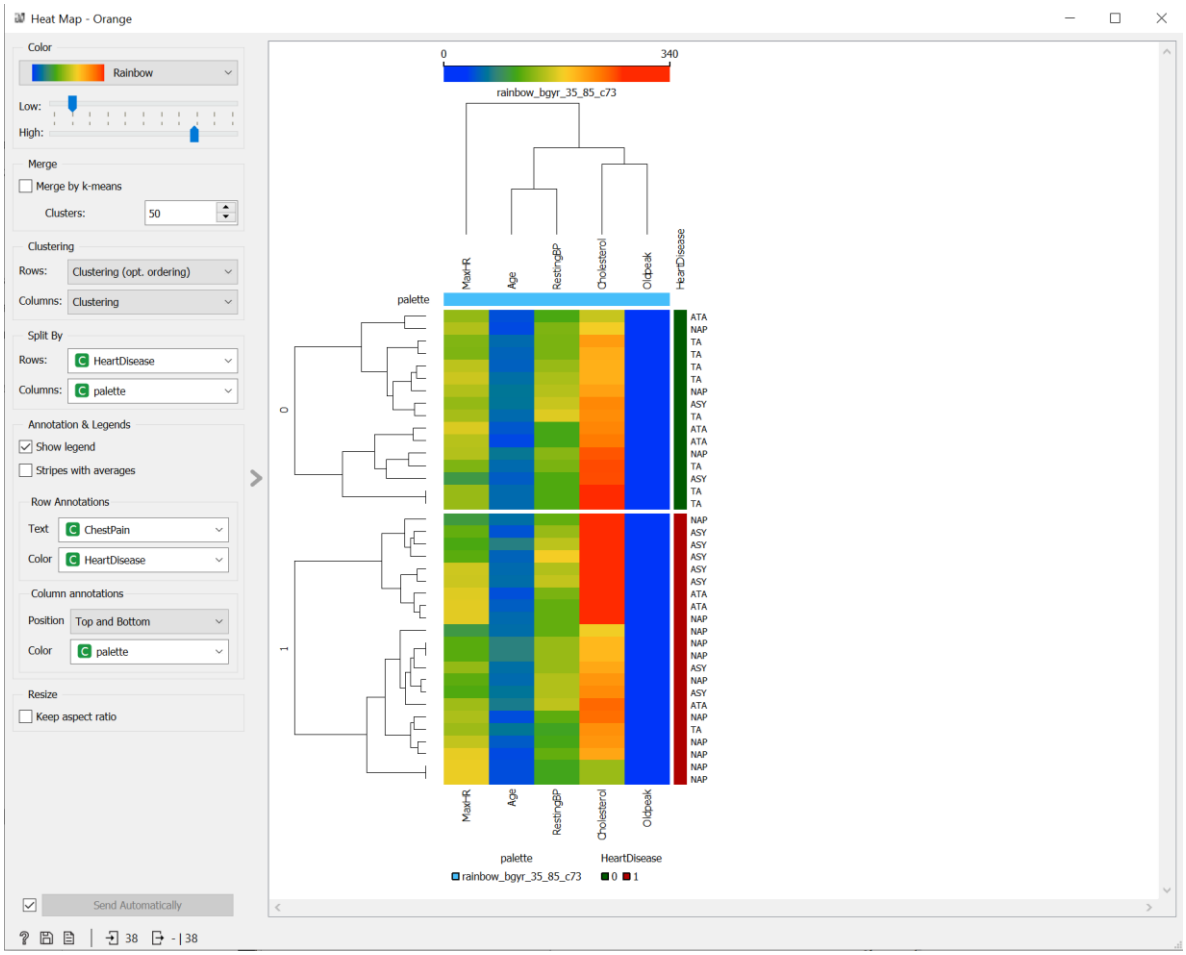


Рис. 31. Визуализация данных на тепловой карте.

Рассматривая тепловую карту, можно отметить, что у пациентов с сердечной недостаточностью или ее отсутствием показатель Oldpeak имел область синего градиента. Это связано с тем, что все числовые показатели данного атрибута были достаточно низкими и варьировали от 0 до 4,4 единиц, о чем свидетельствовала синяя область показателя Oldpeak. Таким образом, в данном случае сложно интерпретировать данный атрибут только на тепловой карте. Ярко выраженным показателем на тепловой карте является холестерин, так как показатели холестерина имели самые высокие значения. По цветовой палитре значения холестерина попали в красную и оранжевую область. Записи данных, где уровень

холестерина показан желтым цветом означает, что это его низкий уровень. Наоборот, красный цвет указывал на высокий уровень холестерина. Итак, по тепловой карте видно, что у 50% пациентов, у которых диагностировали сердечную недостаточность, был высокий уровень холестерина в крови. Показатели максимальной частоты сердечных сокращений и артериального давления были расположены в желто-зеленой области. Рассматриваемые показатели находились в среднем диапазоне тепловой карты. Темно зеленые участки обозначают низкие показатели атрибутов, а желтые наиболее высокие значения. По тепловой карте видно, что большинство желтых участков проявились у тех пациентов, у которых диагностировалась сердечная недостаточность. Возраст пациентов попадал в синюю область, в которую включены пациенты и молодого и пожилого возраста. Ярко синие участки тепловой карты показывали более молодой возраст пациентов, тогда как темно синие участки обозначали пожилых пациентов.

Изучая в целом тепловую карту, очевидно, что пожилой возраст, показатели максимальной частоты сердечных сокращений и холестерина имели особенно ярко выраженный характер при положительной диагностике сердечной недостаточности. У таких пациентов чаще встречалась бессимптомная стенокардия или боль носила неангинозный характер. При отсутствии сердечной недостаточности у пациента мог отмечаться любой тип боли в груди, уровень холестерина изредка носил высокий характер, незначительна отличалась частота сердечных сокращений, артериальное давление было в пределах нормы и все полученные данные не зависели от возраста пациента.

3.7.7. Визуализация Mosaic Display

На мозаичном дисплее отражены соотношения возраста пациентов и присутствие у них ишемической болезни сердца (ИБС) в анамнезе, что позволило выявить у пациентов сердечную недостаточность и ее отсутствие (*Рис. 32*).

В первом столбце на ЭКГ показана депрессия в диапазоне меньше 0,1 сегмента ST, что указывало на отсутствие ИБС у пациента в анамнезе. У таких пациентов в возрасте меньше 47 лет, сердечная недостаточность диагностировалась только в 30%, в возрасте от 56,5 до 60,5 лет в 100%, в возрасте старше 60,5 лет в 50% случаях.

Второй столбик показывает на ЭКГ депрессию сегмента ST от 0,1 до 1.05, что может означать перенесённую в прошлом ИБС. У пациентов с подозрением на ИБС в возрасте меньше 47 лет в 100% случаях диагностировалась сердечная недостаточность, в возрасте от 47 до 56,5 лет в 30%, в возрасте от 56,5 до 60,5 лет в 100%, а в возрасте старше 60,5 лет диагностировалась в 50%.

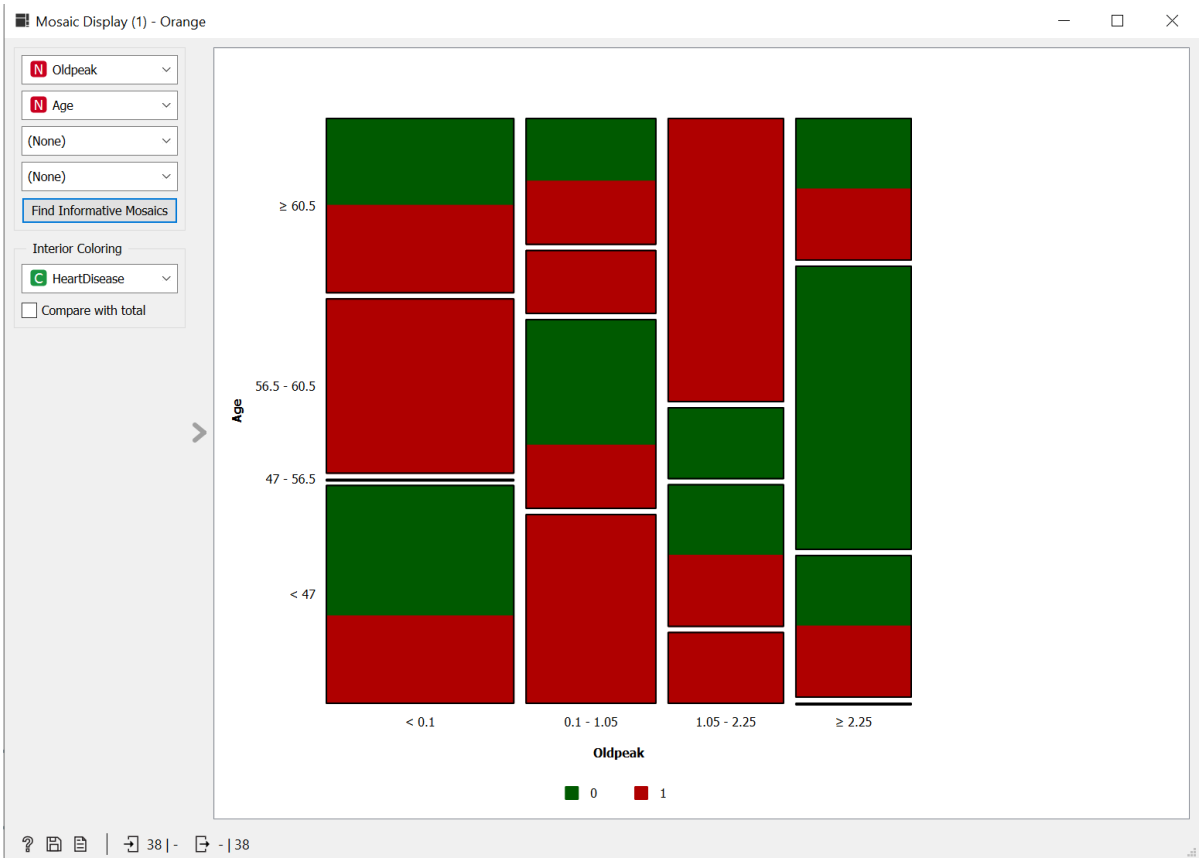


Рис. 32. Соотношение возраста пациента и наличие ишемической болезни сердца.

Третий столбик демонстрирует на ЭКГ депрессию сегмента ST в диапазоне от 1.05 до 2,25, что свидетельствовало о перенесенной ИБС в прошлом. У таких пациентов в возрасте младше 47 лет в 100% диагностировалась сердечная недостаточность, в возрасте от 47 до 56,5 лет в 50%, в возрасте от 56,5 до 60,5 лет сердечная недостаточность не была диагностирована, а в возрасте старше 60,5 лет в 100% случаях.

Четвертый столбик указывал на ЭКГ на депрессию сегмента ST в диапазоне от 2,25 и выше, что является признаком перенесенной ИБС пациентом в недавнем прошлом. У таких пациентов в возрасте от 47 до 56,5 лет сердечная недостаточность диагностировалась в 50%, в возрасте от 56,5 до 60,5 лет не была диагностирована, а в возрасте старше 60,5 лет сердечная недостаточность была выявлена в 50% случаях.

Таким образом, данный мозаичный дисплей демонстрирует результаты тестового набора данных, в который входит 38 пациентов, что не позволяет сделать вывод о зависимости, перенесенной ИБС в анамнезе и возраста пациента на наличие или отсутствие сердечной недостаточности.

Заключение

В выполненной работе были проанализированы и сопоставлены четыре прогностические модели интеллектуального анализа данных на примере сердечной недостаточности. Выполнив тестирование алгоритмов и оценку прогноза каждой модели, можно заключить, что алгоритм правил индукции является наиболее эффективным и обладает наименьшей погрешностью. Показатели коэффициента точности алгоритма правил индукций составили 95,2% при прогнозировании отсутствия сердечной недостаточности и 95,3% при диагностировании сердечной недостаточности. Вероятности ложноположительных и ложноотрицательных результатов составили 4,8% и 4,7% соответственно.

Показатели точности алгоритма дерева решений незначительно уступают по эффективности алгоритму правил индукции. Показатели коэффициента точности алгоритма составили 92,5% при прогнозировании отсутствия сердечной недостаточности и 94,4% при диагностировании сердечной недостаточности. Вероятности ложноположительных и ложноотрицательных результатов составили 7,5% и 5,6% соответственно. Погрешность результатов алгоритма дерева решений немного выше, чем у алгоритма правил индукций, но в сравнении с другими исследуемыми прогностическими моделями значительно меньше. Соответственно данный алгоритм также весьма эффективен для помощи принятия врачебных решений при диагностике сердечной недостаточности. С целью повышения достоверности прогнозируемых результатов, совместное использование выше приведенных прогностических моделей позволит улучшить достоверность прогнозируемых данных.

Прогностические модели логистическая регрессия и наивный Баейс показали невысокие показатели точности прогнозов. У логистической регрессии показатели коэффициента точности алгоритма составили 74,7% при прогнозировании отсутствия сердечной недостаточности и 75,8% при диагностировании сердечной недостаточности. Вероятности ложноположительных и ложноотрицательных результатов составили 25,3% и 24,2% соответственно. У алгоритма наивного Байеса показатели коэффициента точности составили 74,6% при прогнозировании отсутствия сердечной недостаточности и 75,8% при прогнозировании наличия сердечной недостаточности. Вероятности ложноположительных и ложноотрицательных результатов составили 25,4% и 24,2% соответственно. Тестирование и оценка данных алгоритмов показали коэффициенты точности, которые значительно ниже вышеизложенных исследуемых прогностических моделей. Прогнозирование сердечной недостаточности на примере тестового набора данных по средствам данным алгоритмов,

также показали высокую погрешность прогнозов. Следовательно данные прогностические модели наименее подходят для диагностики сердечной недостаточности. Мы связываем это с тем, что для логистической регрессии не хватает обучающей выборки, а алгоритм наивного Байеса предполагает независимость всех атрибутов данных и учитывает позитивное и негативное влияние каждого атрибута. Данное предположение алгоритма в анализе данных сердечной недостаточности выдает более высокие значения ошибочных прогнозов, так как совокупность нескольких клинико-диагностических данных увеличивает вероятность возникновения сердечной недостаточности в то время, как по отдельности эти признаки могут не приводить к возникновению сердечной недостаточности. Это может свидетельствовать об отдельных индивидуальных физиологических особенностях пациентов.

Корректное применение прогностических моделей позволяет медицинским организациям правильно распределять нагрузку на персонал, особенно в условиях ограниченных диагностических ресурсов. Прогностические модели превосходно подходят для определенных задач медицины, в частности при диагностике заболеваний по клинико-диагностическим признакам. Таким образом, прогностические модели позволяют увидеть даже незначительные закономерности при диагностике заболеваний, что позволяет снизить затраты на дополнительную диагностику и увеличить эффективность работы медицинского персонала.

Список литературы

1. Анализ данных / М.Ю. Архипова, В.П. Сиротин, В.С. Мхитарян [и др.]. - М.: Изд-во Юрайт, 2016. – 491 с.

2. Андерсон, К. Аналитическая культура. От сбора данных до бизнес-результатов / К. Андерсон. – СПб.: «Питер», 2015. – 392 с.

3. Благирев, А.П. Big data простым языком / А.П. Благирев. - М.: "Издательство АСТ", 2019. - 256 с.

4. Бринк, Х. Машинное обучение / Х. Бринк, Дж. Ричардс, М. Феверолф. - СПб.: "Питер", 2017. - 336 с.

5. Вьюгин, В.В. Математические основы теории машинного обучения и прогнозирования / В.В. Вьюгин. – М.: Издательство МЦНМО, 2013. – 304 с.

6. Гифт, Н. Прагматичный ИИ. Машинное обучение и облачные технологии / Н. Гифт. - СПб.: "Питер", 2019. – 306 с.

7. Грас, Д. Data Science. Наука о данных с нуля / Д. Грас. – СПб.: «БХВ-Петербург», 2021. – 416 с.

8. Джеймс, Г. Введение в статистическое обучение с примерами на языке R / Г. Джеймс, Д. Уиттон, Т. Хасти, Р. Тибширани. - М.: ДМК Пресс, 2017. – 456 с.

9. Джулли, А. Библиотека Keras – инструмент глубокого обучения / А. Джулли, С. Пал. - М.: ДМК Пресс, 2018. – 298 с.

10. Кук, Д. Машинное обучение с использованием библиотеки H2O / Д. Кук. - М.: ДМК Пресс, 2018. – 252 с.

11. Култыгин, О.П. Использование искусственного интеллекта – реальность и преспективы / О.П. Култыгин // Journal of applied informatics. – 2019. – Vol. 14. - DOI: org/10.24411/1993-8314-2019-10010.

12. Люгер, Д.Ф. Искусственный интеллект. Стратегии и методы решения сложных проблем / Д.Ф. Люгер. – М.: ИД «Вильямс», 2003. – 864 с.

13. Марманис, Х. Алгоритмы интеллектуального Интернета. Передовые методики сбора, анализа и обработки данных / Х. Марманис, Д. Бабенко. - СПб.; М.: «Символ», 2011. – 468 с.

14. Мерков, А.Б. Распознавание образов: Введение в методы статистического обучения / А.Б. Мерков. – М.: Едиториал УРСС, 2011. – 254 с.

15. Молниеносный анализ данных / Х. Керау, Э. Конвински, П. Венделл, М. Захария. - М.: ДМК Пресс, 2015. – 306 с.

16. Нархид, Н. Apache Kafka. Потоковая обработка и анализ данных / Н. Нархид, Г. Шапира, Т. Палино. - СПб.: «Питер», 2019. – 320 с.
17. Николенко, С. Глубокое обучение. Погружение в мир нейронных сетей / С. Николенко, А. Кадурин, Е. Архангельская. - СПб.: «Питер», 2018. – 479 с.
18. Паттерсон, Д. Глубокое обучение с точки зрения практики / Д. Паттерсон, А. Гибсон. - М.: ДМК Пресс, 2018. – 418 с.
19. Потапов, А.С. Распознавание образов и машинное восприятие / А.С. Потапов. – СПб.: Политехника, 2007. – 547 с.
20. Рашид, Т. Создаем нейронную сеть / Т. Рашид. - СПб.: ООО "Альфа-книга", 2017. - 272 с.
21. Сергеев, Н. Аналитика и Data Science / Н. Сергеев. – М.: Издательские решения, 2019. – 330 с.
22. Флах, П. Машинное обучение. Найка и искусство построения алгоритмов, которые извлекают знания из данных / П. Флах. - М.: ДМК Пресс, 2015. – 400 с.
23. Хайкин, С. Нейронные сети / С. Хайкин. – М.: Издательский дом «Вильямс», 2006. – 1104 с.
24. Чашкин, Ю.Р. Математическая статистика. Анализ и обработка данных / Ю.Р. Чашкин. – Ростов н/Д: «Феникс», 2010. – 236 с.
25. Шарден, Б. Крупномасштабное машинное обучение вместе с Python / Б. Шарден, Л. Массарон, А. Боскетти. - М.: ДМК Пресс, 2018. – 360 с.
26. Шолле, Ф. Глубокое обучение на Python / Ф. Шолле - СПб.: "Питер", 2018. - 400 с.
27. Ын, А. Теоритический минимум по Big Data / А. Ын, К. Су. - СПб.: «Питер», 2019. – 208 с.
28. An iterative model of the generalized Cauchy process for predicting the remaining useful life of lithium-ion batteries / G. Hong, W. Song, Y. Gao, A. Kudreyko [et al.] // Measurement. – 2022. – Vol. 187. – DOI: org/10.1016/j.measurement.2021.110269.
29. Analysis of Heart Disease Data Using K-Means Clustering Algorithm in Orange Tool / S. Kodati, K.P. Reddy, G. Ravi, N. Sreekanth // Intelligent Manufacturing and Energy Sustainability. - 2021. – Vol. 213. – DOI: org/link.springer.com/chapter/10.1007/978-981-33-4443-3_40.
30. Analyzing COVID-19 Dataset through Data Mining Tool “Orange” / U. Thange, V.K. Shukla, R. Punhani, W. Grobbelaar // IEEEXplore. – 2021. – DOI: org/10.1109/ICCAKM50778.2021.9357754.
31. Andriesse, D. Practical Binary Analysis / D. Andriesse. - William Pollock, 2019. – 460 p.
32. Bellemare, A. Building Event-Driven Microservices / A. Bellemare. - O’Reilly Media, Inc., 2020. – 324 p.

33. Bruce, P. Practical statistics for data scientists / P. Bruce, A. Bruce, P. Gedeck. - O'Reilly Media, Inc., 2020. – 363 p.
34. Canning, J. Data structures & Algorithms in Python / J. Canning, A. Broden, R. Lafore. - Addison-Wesley, 2017. – 1160 p.
35. Data Mining for Business analytics / G. Shmueli, P.C. Bruce, P. Gedeck, N.R. Patel. - John Wiley & Sons, Inc., 2020. – 607 p.
36. Democratized image analytics by visual programming through integration of deep models and small-scale machine learning / P. Godec, M. Pancur, N. Ilenic [et al.] // Nature Communications. – 2019. – Doi.org/nature.com/articles/s41467-019-12397-x.
37. Fractional Lévy stable motion: Finite difference iterative forecasting model / H. Liu, W. Song, M. Li, A. Kudreyko [et al.] // Chaos, Solitons & Fractals. – 2020. – Vol. 133. – DOI: org/10.1016/j.chaos.2020.109632.
38. Goodfellow, I. Deer learning / I. Goodfellow, Y. Bengio, A. Courville. - Massachusetts Institute of Technology, 2016. – 800 p.
39. Goswami, T. Statistical Modeling in Machine Learning / T. Goswami, G.R. Sinha. - Academic Press, 2022. – 398 p.
40. Han, J. Data Mining Concepts and Techniques / J. Han, M. Kamber, J. Pei. - Elsevier Publishers, 2012. – 740 p.
41. Huyen, C. Designing Machine Learning Systems / C. Huyen. - O'Reilly Media, Inc., 2022. – 389 p.
42. Jordan, M. Pattern recognition and machine learning / M. Jordan, J. Kleinberg, B. Scholkorf. - Springer Science+Business Media, LLC, 2006. – 758 p.
43. Li, R. Simulation with Python / R. Li, A. Nakano. – Apress, 2022. – 175 p.
44. Machine learning and data science / P. Agrawal, C. Gupta, A. Sharma [et al.]. - Scrivener Publishing, 2022. – 271 p.
45. Machine learning and wireless communications / Y.C. Eldar, A. Goldsmith, D. Gunduz, H.V. Poor. - Cambridge University Press, 2022. – 559 p.
46. Mahout in Action / S. Owen, R. Anil, T. Dunning, E. Friedman. - Manning Publications, 2012. – 387 p.
47. Metsis, V. Spam Filtering with Naive Bayes - Which Naive Bayes? / V. Metsis, I. Androutsopoulos, G. Paliouras // DBLP. – 2006. – DOI: org/researchgate.net/publication/221650814_Spam_Filtering_with_Naive_Bayes_-_Which_Naive_Bayes.
48. Muller, A.C. Machine Learning with Python / A.C. Muller, S. Guido. - O'Reilly Media, Inc., 2016. – 340 p.

49. Multifractional and long-range dependent characteristics for remaining useful life prediction of cracking gas compressor / W. Song, S. Duan, E. Zio, A. Kudreyko // Reliability Engineering & System Safety. – 2022. – Vol. 225. – DOI: org/10.1016/j.ress.2022.108630.

50. Neural network design / M.T. Hagan, H.B. Demuth, M.H. Beale, O.D. Jesus. - PWS Publishers, 2014. – 1012 p.

51. Ogutu, J.O. Genomic selection using regularized linear regression models: ridge regression, lasso, elastic net and their extensions / J.O. Ogutu, T. Schulz-Streeck, H.-P. Piepho // BMS Proceedings. – 2012. – Vol. 6. – DOI: org/bmcproc.biomedcentral.com/articles/10.1186/1753-6561-6-S2-S10.

52. Pajankar, A. Practical python data visualization / A. Pajankar. – Apress, 2021. – 168 p.

53. Peker, M. Use of Orange Data Mining Toolbox for Data Analysis in Clinical Decision Making: The Diagnosis of Diabetes Disease / M. Peker, O. Özkaraca, A. Şaşar // Expert System Techniques in Biomedical Science Practice. – 2018. – P. 143-167. - DOI: org/10.4018/978-1-5225-5149-2.ch007

54. Pierson, L. Data Science for Dummies / L. Pierson. - John Wiley & Sons, Inc., 2017. – 329 p.

55. Prognosis and Treatment Prediction of Type-2 Diabetes Using Deep Neural Network and Machine Learning Classifiers / M. Kowsher, M.Y. Turaba, T. Sajed, M.M.M. Rahman // IEEEXplore. – 2020. – DOI: org/10.1109/ICCIT48885.2019.9038574.

56. Raj, S. The Pythonic Way / S. Raj. – India: BPB Publications, 2022. – 1073 p.

57. Reis, J. Fundamentals of data engineering / J. Reis, M. Housley. - O'Reilly Media, Inc., 2022. – 446 p.

58. Remaining useful life prediction for Lithium-ion batteries using fractional Brownian motion and Fruit-fly Optimization Algorithm / H. Wang, W. Song, E. Zio, A. Kudreyko // Measurement. – 2020. – Vol. 161. – DOI: org/10.1016/j.measurement.2020.107904.

59. Rojas, R. Neural Networks. A Systematic Introduction / R. Rojas. – Springer, 1996. – 509 p.

60. Type 2: Diabetes mellitus prediction using Deep Neural Networks classifier / P.B.M. Kumar, R.S. Perumal, R.K. Nadesh, K. Arivuselvan // International Journal of Cognitive Computing in Engineering. – 2020. – Vol. 1. – DOI: org/10.1016/j.ijcce.2020.10.002.

61. VanderPlas, J. Python data science handbook / J. VanderPlas. - O'Reilly Media, Inc., 2016. – 548 p.

АНТИПЛАГИАТ
ОБНАРУЖЕНИЕ ЗАИМСТВОВАНИЙ

СПРАВКА

о результатах проверки текстового документа
на наличие заимствований

Башкирский государственный медицинский университет

ПРОВЕРКА ВЫПОЛНЕНА В СИСТЕМЕ АНТИПЛАГИАТ.ВУЗ

Автор работы: Серегина Р. Р.
Самоцитирование рассчитано для: Серегина Р. Р.
Название работы: ИССЛЕДОВАНИЕ АЛГОРИТМОВ ПРОГНОЗИРОВАНИЯ ПО ПРИЗНАКАМ БАЗЫ ДАННЫХ
Тип работы: Выпускная квалификационная работа
Подразделение: ФГБОУ ВО БГМУ Минздрава России

РЕЗУЛЬТАТЫ

■ ОТЧЕТ О ПРОВЕРКЕ КОРРЕКТИРОВАЛСЯ: НИЖЕ ПРЕДСТАВЛЕНЫ РЕЗУЛЬТАТЫ ПРОВЕРКИ ДО КОРРЕКТИРОВКИ

СОВПАДЕНИЯ	9.96%	СОВПАДЕНИЯ	9.96%
ОРИГИНАЛЬНОСТЬ	90.04%	ОРИГИНАЛЬНОСТЬ	90.04%
ЦИТИРОВАНИЯ	0%	ЦИТИРОВАНИЯ	0%
САМОЦИТИРОВАНИЯ	0%	САМОЦИТИРОВАНИЯ	0%

ДАТА ПОСЛЕДНЕЙ ПРОВЕРКИ: 13.06.2023

ДАТА И ВРЕМЯ КОРРЕКТИРОВКИ: 13.06.2023 14:54

Структура документа: Проверенные разделы: содержание с.1-2, основная часть с.3-67, библиография с.68-69

Модули поиска: ИПС Адилет; Модуль поиска "БГМУ"; Библиография; Сводная коллекция ЭБС; Интернет Плюс*; Сводная коллекция РГБ; Цитирование; Переводные заимствования (RuEn); Переводные заимствования по eLIBRARY.RU (EnRu); Переводные заимствования по Интернету (EnRu); Переводные заимствования издательства Wiley ; eLIBRARY.RU; СПС ГАРАНТ: аналитика; СПС ГАРАНТ: нормативно-правовая документация; Медицина; Диссертации НББ; Коллекция НБУ; Перефразирования по eLIBRARY.RU; Перефразирования по СПС ГАРАНТ: аналитика; Перефразирования по Интернету; Перефразирования по Интернету (EN); Перефразирования по коллекции издательства Wiley ; Патенты СССР, РФ, СНГ; СМИ России и СНГ; Шаблонные фразы; Кольцо вузов; Издательство Wiley; Переводные заимствования

Работу проверил: Кобзева Наталья Рудольфовна
ФИО проверяющего

Дата подписи: 13.06.2023

ФГБОУ ВО БГМУ Минздрава России
НАУЧНАЯ БИБЛИОТЕКА
Подпись проверяющего

Чтобы убедиться в подлинности справки, используйте QR-код, который содержит ссылку на отчет.

Ответ на вопрос, является ли обнаруженное заимствование корректным, система оставляет на усмотрение проверяющего. Предоставленная информация не подлежит использованию в коммерческих целях.