

РАЗРАБОТКА АППАРАТНО-ПРОГРАММНОГО КОМПЛЕКСА ПО ОПРЕДЕЛЕНИЮ ПАТОЛОГИЙ СЕРДЕЧНО-СОСУДИСТОЙ СИСТЕМЫ НА ОСНОВЕ ВЕКТОРНОЙ ЭЛЕКТРОКАРДИОГРАММЫ

Разработчик - Колесников Дмитрий Андреевич

2023 г.

ПЕРЕЧЕНЬ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

В настоящей расчетно-пояснительной записке приняты следующие сокращения и обозначения.

АПК – аппаратно-программный комплекс

ЭКГ – электрокардиография

ВЭКГ – векторная электрокардиограмма

СППР – система поддержки принятия решений

ПО – программное обеспечение

ДВП – дискретное вейвлет-преобразование

ФНЧ – фильтр нижних частот

ФВЧ – фильтр верхних частот

СКО – среднееквадратическое отклонение

EF – ejection fraction

CLI – command-line interface

GUI – graphical user interface

ML – machine learning

DL – deep learning

SVM – support vector machines

KNN – k-nearest neighbors

CNN – convolutional neural network

MAE – mean absolute error

MSE – mean squared error

ОГЛАВЛЕНИЕ		с.
ВВЕДЕНИЕ		5
1 Медицинская часть		7
1.1 Общее строение кровеносной системы.....		7
1.2 Электрокардиография.....		8
1.3 Векторная электрокардиография.....		10
1.4 Преимущества ВЭКГ в определении дисфункции выброса желудочка..		13
2 Разработка АПК по вычислению ВЭКГ и его информативных признаков		16
2.1 Описание работы АПК.....		16
2.2 Предобработка электрокардиограммы.....		21
2.2.1 Считывание данных		21
2.2.2 Ресемплирование исходного сигнала		22
2.2.3 Детектирование PQRSТ комплекса		23
2.2.4 Фильтрация сигнала.....		26
2.3 Построение векторкардиограммы.....		27
2.3.1 Расчет и визуализация векторкардиограммы		27
2.3.2 Сглаживание полученного ВЭКГ		28
2.3.3 Сохранение ВЭКГ как масштабированные изображения		30
2.3.4 Сохранение ВЭКГ как облако точек		32
2.3.5 Результаты построения ВЭКГ испытуемых		32
2.4 Определение физиологических параметров.....		34
2.4.1 Выделение QRS и ST-T петель		34
2.4.2 Вычисление площадей петель.....		35
2.4.3 Определение угла QRS-T		36
2.5 Приложение с графическим интерфейсом.....		37
3 Создание обучающих данных для моделей машинного обучения.....		41
3.1 Исходный датасет ЭКГ.....		41
3.2 Генерация датасета информативных признаков ВЭКГ.....		41
3.3 Генерация датасета изображений на основе ВЭКГ.....		42
3.4 Генерация датасета облаков точек ВЭКГ.....		44
4 Машинное обучение на основе информативных признаков.....		46

4.1	Применение машинного обучения.....	46
4.2	Разведочный анализ и переподготовка данных.....	47
4.3	Бинарная классификация для идентификации заболеваний.....	49
4.3.1	Выбор метрик качества для задачи классификации... ..	49
4.3.2	Дерево решений.....	50
4.3.3	Случайный лес.....	53
4.3.4	Логистическая регрессия.....	54
4.3.5	К-ближайших соседей	55
4.3.6	Метод опорных векторов.....	57
4.3.7	Градиентный бустинг.....	58
4.3.8	Сравнение ML алгоритмов классификации	59
4.4	Определение величины фракции левого желудочка.....	60
4.4.1	Выбор метрик качества для задачи регрессии	60
4.4.2	Линейная регрессия.....	61
4.4.3	Градиентный бустинг.....	62
4.4.4	Сравнение ML алгоритмов регрессии.....	63
5	Машинное обучение на основе изображений проекций	64
5.1	Применение сверточных нейронных сетей.....	64
5.2	Преподготовка данных перед обучением.....	65
5.3	Архитектуры сетей.....	66
5.4	Обучение сетей для бинарной классификации.....	68
5.5	Обучение сетей для регрессии.....	73
5.6	Интерпретация работы сетей.....	75
6	Машинное обучение на основе облаков точек.....	78
6.1	Архитектура PointNet.....	78
6.2	Переподготовка данных.....	79
6.3	Обучение моделей.....	82
	ЗАКЛЮЧЕНИЕ	86
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	87
	ПРИЛОЖЕНИЕ А. Фрагмент исходного кода	92

ВВЕДЕНИЕ

Сердечно-сосудистые заболевания (болезни системы кровообращения) являются основной причиной смерти как в России, так и во всём мире [1].

Россия занимает лидирующее положение по смертности от сердечно-сосудистых заболеваний, среди которых наибольшую частоту имеет ишемическая болезнь сердца [2]. В 2018 году в России умерло 1 489 941 человек, из них 856 127 человек умерли из-за болезней системы кровообращения [3], что составляет 57,46% от общего числа смертей.

Своевременная диагностика патологий сердечного выброса играет важную роль в обеспечении эффективного лечения и повышении шансов на полное восстановление здоровья пациента. Разработка системы поддержки принятия решений в этой области актуальна, так как применение искусственного интеллекта может значительно ускорить диагностический процесс [4]. Алгоритмы машинного обучения способны анализировать большие объемы данных и выявлять патологии быстрее, чем это может сделать врач. Это критически важно в ситуациях, где требуется оперативное принятие решения.

Наиболее распространенным методом диагностики патологий сердца является регистрация электрокардиограммы – методика исследования электрических полей, образующихся при работе сердца. Результаты многоканального ЭКГ представляют собой проекции исходного вектора сердца на различные линии или оси, предоставляя информацию о работе сердечной мышцы в разных направлениях.

Векторная кардиография (ВЭКГ) представляет собой более мощный метод исследования. Она регистрирует изменения суммарного вектора электродвижущих сил сердца, что позволяет более точно анализировать работу сердца и выявлять патологии [5]. Наиболее ценной особенностью ВЭКГ является тот факт, что для определения проблем, связанных с проводимостью сердца и другими аномалиями, в большинстве случаев достаточно анализировать только один период кардиоцикла.

Однако, несмотря на потенциал ВЭКГ, многие алгоритмы исследования сердца на основе машинного обучения до сих пор ориентированы на классические сигналы ЭКГ [6]. Разработка и внедрение алгоритмов машинного обучения, способных анализировать ВЭКГ и делать выводы о состоянии сердца на основе этой информации, представляет собой перспективное направление.

Помимо этого, актуальным также является разработка аппаратно-программного комплекса, способного получать векторное представление электрокардиограмм на основе исходных многоканальных ЭКГ и осуществлять автоматическое вычисление информативных физиологических характеристик получаемых векторных петель, а так же оказывать поддержку принятия врачебных решений, основываясь на ВЭКГ конкретного заданного периода кардиоцикла.

Переход от классического ЭКГ к векторной ЭКГ (ВЭКГ) можно рассматривать как математический метод уменьшения размерности данных. Это является преобразованием исходных восьми синхронных сигналов ЭКГ, полученных из разных электродов, в один трехмерный вектор или в проекционные 2D изображения, отражающие электрическую активность сердца [7]. Этот переход сокращает объем данных и создает абстрактное представление кардиоцикла, что облегчает последующий анализ. Именно поэтому ВЭКГ представляет собой многообещающую технику, которая до сих пор не была широко использована в машинном обучении в области медицинской диагностики.

Именно поэтому, целью данной выпускной квалификационной работы является разработка технологии определения патологий сердечно-сосудистой системы на основе векторной электрокардиограммы, а также создание программного обеспечения для визуализации ВЭКГ на основе исходных ЭКГ сигналов.

1 Медицинская часть

1.1 Общее строение кровеносной системы

Сердечно-сосудистая система — система органов, которая обеспечивает циркуляцию крови в организме. Благодаря ей кислород и питательные вещества доставляются к органам и тканям тела, а углекислый газ, другие продукты метаболизма и отходы жизнедеятельности выводятся из организма.

Кровь представляет собой внутреннюю среду организма, диагностируя которую можно судить о патологических состояниях всех внутренних органов человека [8].

Кровеносная система человека является замкнутой. В ее состав входит сердце — мышечный орган, заставляющий кровь двигаться, ритмически, направляя её в кровеносные сосуды — полые трубки различного диаметра, по которым происходит циркуляция крови.

Сердце, как центральный орган кровообращения, представляет собой мышечный орган человеческой анатомии с четырьмя полостями: двумя предсердиями (правым и левым) и двумя желудочками (также правым и левым). Двигаясь в циклическом ритме, сердечные мышцы обеспечивают последовательные систолические (сокращение) и диастолические (расслабление) фазы, что позволяет перекачивать кровь в систему артерий и вен организма.

Сокращение стенок камер сердца называется систолой, их расслабление — диастолой.

Сердце связывает два круга кровообращения: малый, который транспортирует кровь к легким, где обогащает кислородом, и большой, доставляющий кровь к органам (рисунок 1).

Предсердия функционируют как резервуары, принимая кровь из больших вен и направляя ее в желудочки. Желудочки, в свою очередь, сокращаются, перекачивая кровь в систему артерий, что обеспечивает циркуляцию крови по всему организму [10]. Более того, сердце регулирует кровяное давление, обеспечивает кислород и питательные вещества для тканей и органов, а также

участвует в эффективной фильтрации шлаковых продуктов обмена веществ из системы органов и тканей.



Рисунок 1 – Общая структура кровеносной системы

Стрелками показаны направления токов крови [9]

Таким образом, сердце не только фундаментально для обеспечения жизнедеятельности организма, но и служит важным индикатором общей функциональности организма, и его здоровье имеет высший приоритет для поддержания общего благополучия.

1.2 Электрокардиография

Электрокардиография (ЭКГ) является методом исследования сердечной активности, который один из наиболее распространенных и информативных в медицинской практике. ЭКГ записывает электрическую активность сердца в виде графической кривой, называемой электрокардиограммой.

Принцип работы ЭКГ основан на регистрации электрических изменений, которые возникают в результате сердечного цикла. Сердце работает как электрический генератор, и каждое его сокращение создает электрический импульс, который распространяется по всему сердцу [11]. Электроды, прикрепленные к коже на определенных точках тела, регистрируют эти электрические изменения. Самым распространенным в медицинской практике является 12-канальный ЭКГ, который использует 10 электродов для

регистрации электрической активности сердца на 12 различных направлениях на поверхности тела пациента. Эти 12 направлений называются отведениями, и они регистрируют электрические сигналы, происходящие в разных частях сердца (рисунок 2).

Отведения I, II, III представляют верхние конечности и записывают электрическую активность в вертикальной плоскости.

Отведения aVR, aVL, aVF также представляют верхние конечности и предоставляют информацию о вертикальной плоскости, но в отличие от предыдущих они расположены так, чтобы включать в себя разные углы обзора сердца. Значения этих проекций может быть математически вычислено из основных трех отведений.

Отведения V1, V2, V3, V4, V5, V6 расположены на грудной клетке и записывают электрическую активность в горизонтальной плоскости. Они обеспечивают подробную информацию о работе различных частей сердца.

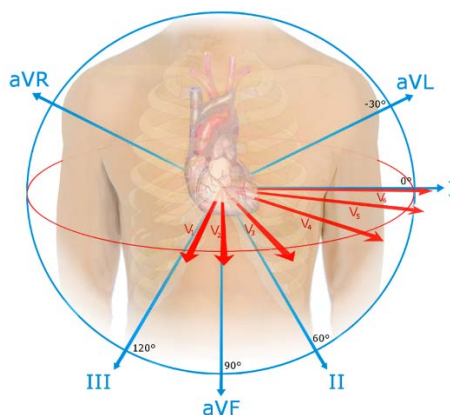


Рисунок 2 – Линии проецирования отведений [12]

ЭКГ является дешевым, неинвазивным и широкодоступным методом исследования. Проведение ЭКГ требует минимальной подготовки пациента и не предполагает использования лекарств или оборудования, которое требует специальной подготовки, именно поэтому такой вид исследований приобрел такую высокую популярность в медицине [13].

На самих сигналах ЭКГ выделяют несколько основных точек и сегментов, которые позволяют врачам оценить ритм, проводимость и возможные аномалии в работе сердца (рисунок 3) [11]. ЭКГ является

неотъемлемым инструментом для диагностики сердечных заболеваний, оценки эффективности лечения и мониторинга состояния пациентов.

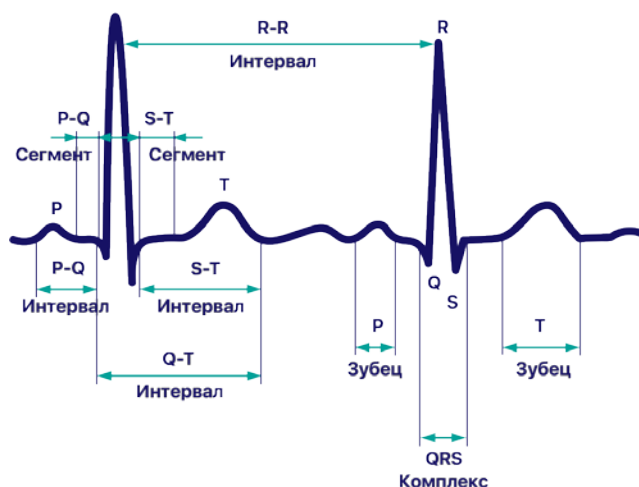


Рисунок 3 – Основные точки и сегменты ЭКГ [11]

1.3 Векторная электрокардиография

Векторная электрокардиография (Vectorcardiography, VCG, ВЭКГ) - это метод записи и анализа электрической активности сердца в трехмерном пространстве с использованием векторов [13].

Электрическая активность сердца возникает из-за специальных клеток, называемых кардиомиоцитами, которые генерируют электрические импульсы для контроля сокращения сердечной мышцы.

Суть ВЭКГ заключается в том, что каждый электрический импульс, который проходит через сердце во время его цикла, создает электрический вектор [14]. Этот вектор представляет собой направление и силу электрической активности в определенный момент времени. В то время как 12-канальная электрокардиограмма (ЭКГ) обеспечивает информацию о сердечной активности только в виде напряжений на разных точках тела, VCG более подробно описывает пространственное распределение электрической активности [15].

Пример внешнего вида проекций векторных петель на главные плоскости человека представлены на рисунке 4.

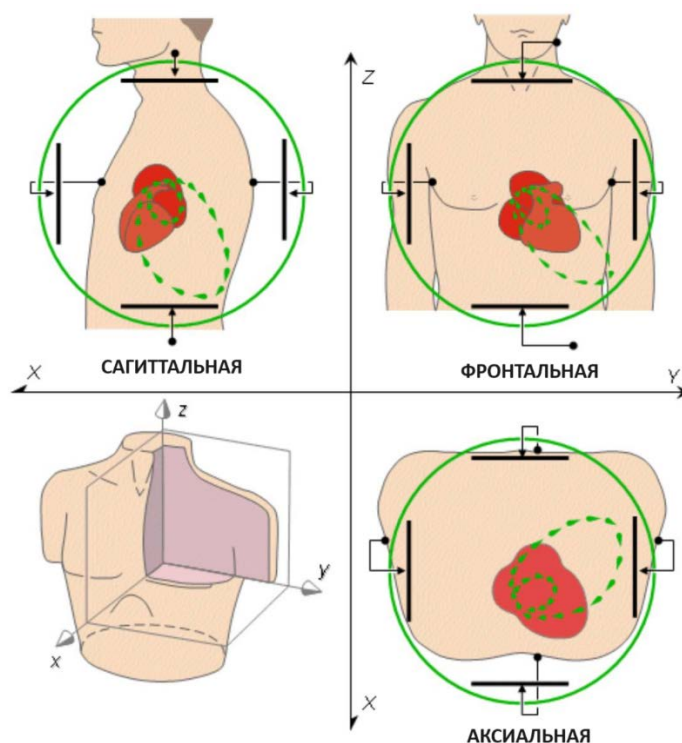


Рисунок 4 – Проекции ВЭКГ [14]

Для получения данных VCG используются специальные расположения электродов на торсе пациента. Эти отведения позволяют фиксировать электрические сигналы в трех измерениях, что обеспечивает более полное представление о сердечной активности. Одним из наиболее известных наборов отведений для регистрации ВЭКГ являются отведения Фрэнка XYZ.

Особенностью отведений Эрнеста Фрэнка, для векторной кардиографии (VCG) является их ортогональность. Это означает, что каждое отведение Фрэнка перпендикулярно (пересекается под углом 90 градусов) другим отведениям. Это обеспечивает независимость измерений и позволяет точно определить направление и силу электрической активности сердца в трехмерном пространстве [16].

Расположение электродов для отведений Фрэнка представлено на рисунке 5. Электроды обозначены как I, E, C, A, M, F, H.

Система отведений Эрнеста Фрэнка для векторной кардиографии (VCG) не стала широко распространенной в медицинской практике. Это произошло из-за сложности и неудобства в размещении множества электродов на пациенте, что затруднило применение в клинических условиях [17]. В

настоящее время, для мониторинга сердечной активности более популярна стандартная 12-канальная электрокардиограмма (ЭКГ), которая более удобна и доступна.

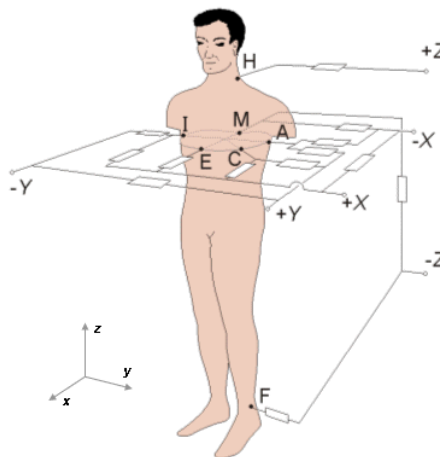


Рисунок 5 - Расположение электродов для отведений Фрэнка [16]

Современные достижения в компьютерной графике и беспроводных технологиях возродили интерес к сигналам векторной кардиограммы (VCG), которые используют меньше отведений, чем обычные 12-канальные электрокардиограммы (ЭКГ), для медицинских диагностических приложений. Тем не менее, большинство кардиологов привыкли к 12-канальной ЭКГ несмотря на то, что некоторые из отведений либо почти совпадают с другими, либо происходят от них, и, следовательно, содержат избыточную информацию.

Однако, повсеместный переход на ортогональные системы отведений не реализуем, так что с целью устранения этих ограничений были разработаны методы синтеза ВЭКГ из обычных 12-канальных электрокардиограмм (ЭКГ). Один из таких методов, предложенный Дауэром и его коллегами, основан на моделировании сердечно-телесной активности. Суть этого метода заключается в том, что каждое отведение ЭКГ выражается как линейная комбинация компонентов X, Y и Z ВЭКГ с коэффициентами, заданными моделью торса Фрэнка [18].

Для получения этих коэффициентов, Эденбрандт и Пальм провели обратное вычисление, вычислив "обратную" матрицу на основе данных Дауэра.

Эта матрица позволяет восстановить ВЭКГ из 12-канальной ЭКГ, что делает процесс быстрым и удобным, и может быть легко интегрировано в современные электрокардиографы, оснащенные микропроцессорами и быстрыми плоттерами либо имплементировано в дополнительные ПО, предоставляемые врачам [19].

Таким образом, использование ВЭКГ, полученной из обычных ЭКГ, становится более удобным и эффективным в клинической практике. Этот подход позволяет устранить проблемы, связанные с традиционными методами регистрации ВЭКГ, и сделать этот метод более доступным для медицинской диагностики и мониторинга сердечной активности.

1.4 Преимущества ВЭКГ в определении дисфункции выброса желудочка

Сердечная недостаточность - это заболевание, при котором сердце не способно эффективно перекачивать кровь по организму. Одной из форм сердечной недостаточности является дисфункция выброса желудочка

Сердечная недостаточность с сниженной фракцией выброса (HFrEF) представляет собой серьезное состояние, при котором левый желудочек сердца не может эффективно выталкивать кровь в организм [20]. Это состояние характеризуется снижением фракции выброса (LV ejection fraction, EF), которая обычно составляет менее 50% [21].

Фракция выброса (EF), также известная как фракция изгнания, является важным параметром, используемым в медицинской диагностике и оценке функции сердца. Она представляет собой числовую величину, которая выражает процентное соотношение между объемом крови, выбрасываемой из левого желудочка сердца при каждом сокращении (систоле), и общим объемом крови в левом желудочке.

Формула для расчета фракции выброса (EF) выглядит следующим образом:

$$EF = \frac{SV}{EDV} \cdot 100\% \quad (1)$$

где SV — ударный объем крови;

EDV — конечный диастолический объем (объем крови, находящейся в левом желудочке сердца в конце его диастолической фазы, то есть в момент максимальной релаксации и наполнения желудочка кровью перед следующим сокращением).

Снижение EF свидетельствует о том, что сердце имеет проблемы с сократительной функцией, что может быть вызвано различными факторами, такими как инфаркт миокарда, гипертония, клапанные болезни или другие сердечные нарушения. У пациентов с HFrEF снижается способность организма обеспечивать ткани и органы достаточным количеством кислорода и питательных веществ, что может привести к различным симптомам, таким как утомляемость, одышка, отеки и нарушения ритма сердца. Лечение HFrEF включает в себя медикаментозную терапию, контроль жидкостного баланса, изменение образа жизни и в некоторых случаях хирургическое вмешательство, такое как имплантация кардиостимуляторов или механических поддерживающих устройств [21].

Преимущества ВЭКГ (векторной ЭКГ) в определении дисфункции выброса желудочка заключаются в его способности предоставить более компактное представление данных и улучшить эффективность диагностики.

Трехмерное представление ВЭКГ позволяет более наглядно отобразить движение электрических векторов сердца во времени. Это особенно полезно при оценке сократительной функции желудочков, так как позволяет врачам и исследователям более четко видеть аномалии и изменения в траектории электрической активности сердца [15]. Такое трехмерное представление может быть полезным в диагностике дисфункции выброса желудочка и других сердечных нарушениях.

Кроме того, использование векторных петель как стартового представления данных может положительно сказаться на машинном обучении. Векторные петли представляют собой более компактное и информативное представление данных ЭКГ (переход из стандартных 12-канальных сигналов в трехканальное отображение на основе одного периода ЭКГ), что может улучшить эффективность алгоритмов машинного обучения при анализе сердечной активности. Однако для полноценного использования этого подхода требуется специализированный анализ и алгоритмы, способные извлекать значимую информацию из векторных петель. Разработка таких подходов и будет представлена в данной выпускной работе.

2 Разработка АПК по вычислению ВЭКГ и его информативных признаков

2.1 Описание работы АПК

Первым этапом разработки системы по определению патологий на основе данных векторной ЭКГ было создание программы по обработке исходных ЭКГ сигналов, записанных на многоканальных кардиографах, отображению их в виде трехмерного интерактивного облака точек и виде изображений проекций на главные плоскости, а также вычислению информативных физиологических параметров из полученных векторных петель.

Для работы данной программы необходимо иметь синхронно записанные сигналы с 3 основных и 6 грудных отведений. Усиленные отведения не участвуют в формировании ВЭКГ.

Самым удобным и популярным способом хранения данных является EDF формат, для работы с которым и был разработан данный АПК [22].

Разработанный скрипт выделяет информативные каналы и метаданные о приборе, на котором было произведена регистрация. На основе этих данных и производится обработка и итоговое представление результатов ВЭКГ вычисления.

Для более гибкой настройки имеется возможность включать и выключать фильтрацию артефактов дыхания, менять частоту дискретизации исходного сигнала, выбирать интересующие периоды кардиоцикла и находить основные точки PQRS на сигналах всех отведений. Так же добавлена возможность производить сохранение полученных проекций ВЭКГ на главные плоскости в виде черно-белых изображений для создания датасета для дальнейших исследований и ML разработок. Так же добавлена возможность включать сглаживание получаемых петель для более привычной демонстрации результатов. Код так же позволяет сохранять результат в виде csv файла облака точек и выводить любые интересующие логи производимых операций в виде графиков и изображений. Помимо вышеперечисленного, функция вычисляет

информативные параметры векторной кардиографии на основе выбранного врачом кардиоцикла.

Код использует `setuptools` для упаковки Python-проекта в виде распределения, которое можно установить с помощью `pip`. Когда проект устанавливается, его файлы и зависимости копируются в стандартные места Python-пакетов, и Python интерпретатор может найти и использовать эти библиотеки при необходимости [23].

После установки проекта с помощью `setuptools`, команды, определенные в `entry_points`, становятся доступными из командной строки. В данном случае, после установки проекта, появится новая команда `get_VECG`, которую можно вызывать из командной строки для выполнения функции `main` из модуля `main.py`. Это позволяет запускать скрипт с определенными аргументами и параметрами прямо из командной строки без необходимости указания полного пути к скрипту.

Для более удобной работы был внедрен удобный парсинг аргументов при вызове функции, что позволяет гибко настраивать желаемый функционал [24].

В таблице 1 представлен полный список параметров с пояснениями, которые можно передать на вход программы ее перед запуском в CLI.

В программе предусмотрены проверки на введение неверных величин. Например, если будет введен номер периода, которого не существует или `n_term_finish` будет менее `n_term_start`, то программа прекратит работу, сообщив пользователю об ошибке. Если указать значение `n_term_start` и `n_term_finish` равным одному и тому же значению, то программа воспримет это как в случае с выбором для анализа одного конкретного периода кардиоцикла.

Для установки программы необходимо установить python 3 версии. Далее требуется клонировать к себе репозиторий и ввести в терминале команду для загрузки библиотек. Все инструкции по работе с кодом и по принципу установки расширения представлены подробно в описании проекта на GitHub. Так же имеется видео лекция, презентующая возможности данной программы.

Ссылка на проект - https://github.com/Koldim2001/vector_ECG

Таблица 1 – Описание аргументов для запуска программы

Параметр	Описание	Значение по умолчанию
--data_edf	Путь к файлу ЭКГ формата edf	-
--n_term_start	Номер интересующего для анализа периода кардиоцикла. Если необходимо построить результат за диапазон периодов, то данный параметр задает стартовое значение номера периода для записи в ЭКГ	3
--n_term_finish	Параметр задается исключительно при необходимости построить диапазон периодов. Значение является номером периода, до которого будет вестись запись ВЭКГ (включительно)	None
--filt	Включение / выключение цифровой фильтрации исходных сигналов с помощью фильтра Баттерворта 1 порядка (ФВЧ).	False
--f_sreza	Задание частоты среза ФВЧ фильтра. Используется исключительно при выборе режима --filt=True.	0.7 Гц
--f_sampling	Задание частоты дискретизации. Будет проведено ресемплирование исходного сигнала с использованием линейной интерполяции.	1500 Гц
--plot_3d	Включение / выключение режима для интерактивного отображения 3D графика ВЭКГ.	True

--show_detected_pqrst	Включение/выключение режима для построения ключевых точек PQRSТ для сигнала ЭКГ, полученных с помощью дискретных вейвлет преобразований.	False
--show_ecg	Включение / выключение режима для построения графиков всех отведений и обнаруженных QRS пиков, относительно которых ведется подсчет номеров n_term_start и n_term_finish.	False
--qrs_loop_area	Включение / выключение режима для расчета площади QRS петли по всем проекциям. Работает при отображении лишь одного периода ЭКГ.	True
--t_loop_area	Включение / выключение режима для расчета площади ST-T петли по всем проекциям. Работает при отображении лишь одного периода ЭКГ.	False
--show_log_loop_area	Включение / выключение режима для отображения отдельных петель. Доступен при включенной опции расчета площади какой-либо из петель: QRS_loop_area или T_loop_area.	False
--count_qrst_angle	Включение / выключение режима для вычисления пространственного угла QRST, а также проекции угла на фронтальную плоскость. Работает при отображении лишь одного периода ЭКГ.	True

--show_log_qrst_angle	Включение / выключение режима для трехмерного отображения угла QRST на ВЭКГ. Работает при выборе режима count_qrst_angle=True.	False
--save_images	Включение / выключение режима для сохранения графиков ВЭКГ трех плоскостей в качестве png изображений. Сохранение производится в папку saved_vECG, создающуюся в корне репозитория. Работает при отображении лишь одного периода кардиоцикла.	False
--cancel_showing	Включение / выключение режима для вывода любых графиков. Позволяет выключить отображение графических результатов для возможности использовать get_VECG в цикле по файлам ЭКГ.	False => отображение графиков включено
--mean_filter	Включение / выключение фильтра среднего для ST-T петли чтобы сгладить петли ВЭКГ	True
--help	Покажет существующие варианты парсинга аргументов в CLI	-

Таким образом, у врача есть возможность после установки расширения к себе на компьютер вызывать в командной строке функцию по обработке edf файлов. При этом требуется лишь сообщить путь к данному файлу и по желанию значения параметров. По умолчанию без передачи параметров будет использоваться предустановленный режим демонстрации результатов без ФВЧ фильтрации и с использованием режима по сглаживанию петель.

2.2 Предобработка электрокардиограммы

2.2.1 Считывание данных

Данные на вход приходят в формате EDF и после обработки мы получаем таблицу csv в которой все данные по каналам лежат в отдельных столбцах (рисунок 6).

	ECG I	ECG II	ECG III	ECG AVR	ECG AVL	ECG AVF	ECG V1	ECG V2	ECG V3	ECG V4	ECG V5	ECG V6
0	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
1	0.00001	0.00001	0.00000	-0.00001	0.00000	0.00001	-0.00001	0.00001	-0.00001	-0.00001	-0.00001	-0.00001
2	0.00002	0.00001	-0.00001	-0.00001	0.00001	0.00000	0.00000	0.00004	0.00000	0.00000	0.00000	-0.00001
3	0.00000	-0.00002	-0.00002	0.00001	0.00001	-0.00002	0.00000	0.00006	0.00000	0.00001	0.00000	0.00000
4	0.00000	-0.00002	-0.00002	0.00001	0.00001	-0.00002	0.00000	0.00007	-0.00001	-0.00001	0.00000	0.00000
...
7495	-0.00022	0.00034	0.00056	-0.00006	-0.00039	0.00045	-0.00054	0.00019	-0.00011	0.00012	-0.00003	-0.00001
7496	-0.00023	0.00034	0.00057	-0.00005	-0.00040	0.00045	-0.00054	0.00019	-0.00010	0.00012	-0.00003	0.00000
7497	-0.00023	0.00034	0.00057	-0.00005	-0.00040	0.00045	-0.00055	0.00019	-0.00011	0.00012	-0.00003	0.00000
7498	-0.00022	0.00036	0.00058	-0.00007	-0.00040	0.00047	-0.00056	0.00017	-0.00012	0.00010	-0.00004	-0.00001
7499	-0.00023	0.00036	0.00059	-0.00006	-0.00041	0.00047	-0.00056	0.00017	-0.00013	0.00010	-0.00005	-0.00001

Рисунок 6 – Отображение полученных многоканальных данных.

Формат EDF (European Data Format) представляет собой стандартный формат для хранения временных серий данных, включая ЭКГ (электрокардиограмма) и другие биомедицинские сигналы. Основные характеристики формата EDF [22]:

- мультиканальность: позволяет хранить данные с нескольких каналов, собранных одновременно с различных сенсоров,
- одновременное сэмплирование: все каналы имеют одинаковую частоту дискретизации (sampling rate),
- метаданные: хранит информацию о каждом канале, такую как его имя, единицы измерения и другие параметры,
- простота и эффективность: предназначен для хранения больших объемов данных и обеспечивает эффективное использование памяти.

Как можно увидеть из метаданных на рисунке 7, прибор МИОКАРД-12 имеет встроенный ФНЧ фильтр для предотвращения высокочастотных помех и аллайзинга. Встроенного ФВЧ не имеется. На данном приборе были

зафиксированы большинство сигналов в датасете, который был предоставлен для проведения исследований.

Measurement date	May 24, 2023 17:38:57 GMT
Experimenter	Unknown
Participant	Unknown
Digitized points	Not available
Good channels	12 EEG
Bad channels	None
EOG channels	Not available
ECG channels	Not available
Sampling frequency	250.00 Hz
Highpass	0.00 Hz
Lowpass	125.00 Hz

Рисунок 7 – Метаданные от МИОКАРД-12

Основными требованиями для разрабатываемого ПО к входным данным являются:

- формат EDF,
- исходная частота дискретизации сигналов более 200 Гц,
- наличие встроенного в прибор фильтра нижних частот (ФНЧ),
- наличие сигналов с отведений I, II, V1, V2, V3, V4, V5, V6.

Программа способна обработать как 9-канальный, так и 12-канальный сигнал при условии вышеописанных требований.

2.2.2 Ресемплирование исходного сигнала

Для того, чтобы строить отображение QRS области на ВЭКГ исходных 250-500 Гц дискретизации прибора слишком мало (рисунок 8, 9).

По этой причине, для создания более плавной итоговой картины с более частыми точками была реализована операция ресемплирования сигнала с помощью линейной интерполяции исходного сигнала. Значение частоты новой дискретизации равна по умолчанию 1500 Гц, но может быть задана как параметр на выход `get_VECG` скрипта (`--f_sampling`).

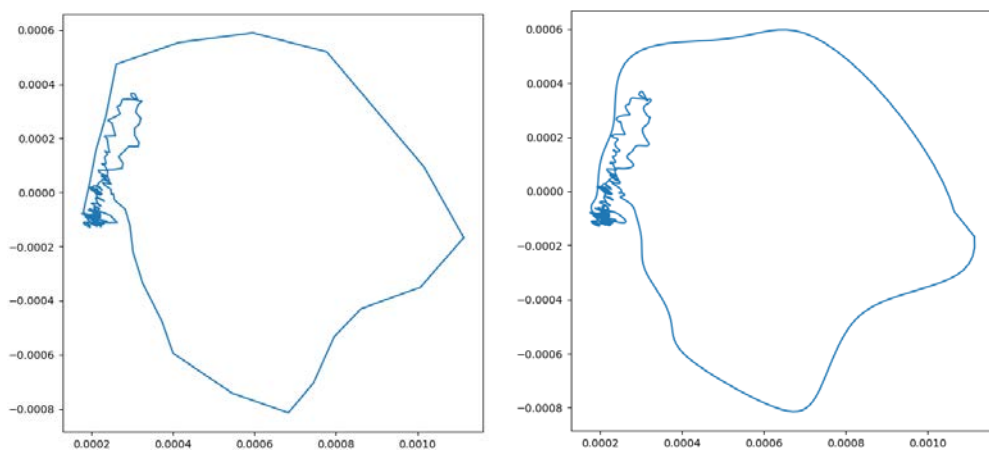


Рисунок 8 – Визуализация фронтальной проекции ВЭКГ до и после ресемплирования

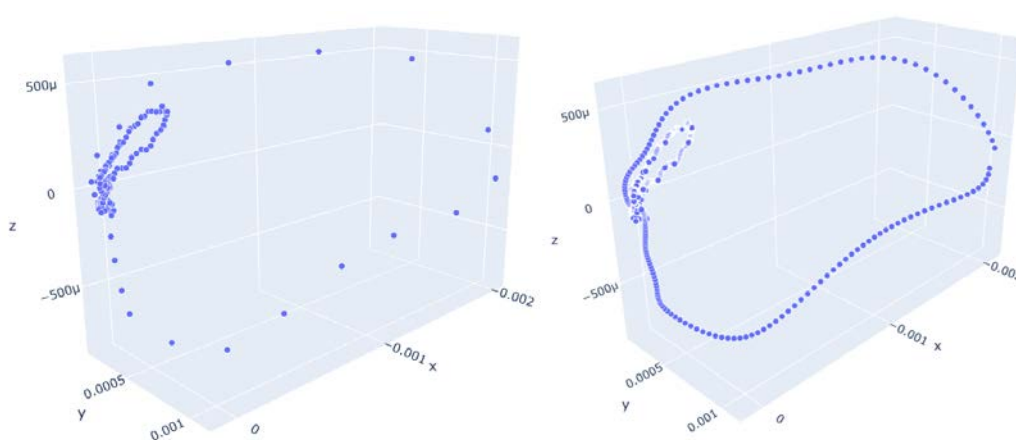


Рисунок 9 – Визуализация трехмерной ВЭКГ до и после ресемплирования

2.2.3 Детектирование PQRSТ комплекса

Для того, чтобы определить конкретный заданный период кардиоцикла или выделить интересующую область в несколько кардиоциклов, необходимо определять точки начала и конца каждого интервала. Для этого был реализован подход и рассмотрением сигнала от начала R зубца текущего до R зубца следующего цикла. Так что один цикл начинается и заканчивается на области R пика.

Передо мной стояла задача по разработке системы по детектированию R зубцов. Одним из исследованных мной подходов по его определению был самостоятельно обученный алгоритм градиентного бустинга, но он давал не очень высокую точность определения.

По этой причине стояла задача улучшения этого подхода с использованием более классических подходов в обработке сигналов. Был определен самый распространенный подход в детектировании всех основных комплексов PQRS, основанный на дискретном вейвлет преобразовании [25], [26].

Данный алгоритм поиска точек работает следующим образом:

- 1) Применение ДВП: ЭКГ сигнал разлагается на различные частоты с помощью ДВП.
- 2) Анализ высокочастотных коэффициентов: Высокочастотные коэффициенты ДВП анализируются для обнаружения быстрых изменений, соответствующих точкам R и S.
- 3) Анализ низкочастотных коэффициентов: Низкочастотные коэффициенты ДВП анализируются для обнаружения пиков, соответствующих точкам P и T.
- 4) Определение местоположения точек: Алгоритм определяет местоположение точек PQRS на основе обнаруженных пиков и их характеристик. Происходит усреднение областей найденных точек если близко имеются одинаковые предсказания.

Данный готовый алгоритм был применен для нахождения R зубцов на первом отведении, что позволило из-за параллельности записи каналов определить соответствующие пики и на остальных отведениях (рисунок 10).

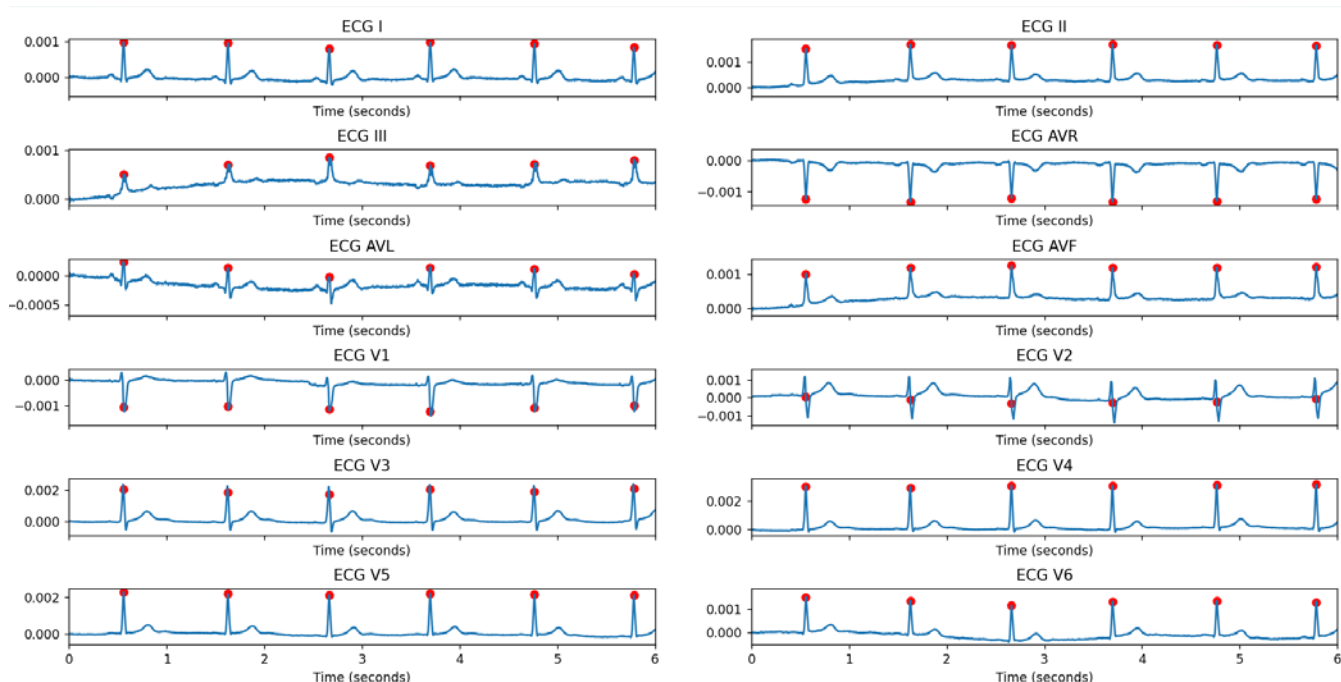


Рисунок 10 – Сигналы всех отведений с детектированными R зубцами

Таким образом, была получена возможность выбирать любой интересующий кардиоцикл для финальной визуализации ВЭКГ.

Помимо этого, был имплементирован алгоритм по поиску точек P, Q, S, T, который будет использоваться в вычислениях угла QRS и площадей петель. Пример нахождения этих точек можно увидеть на рисунке 11.

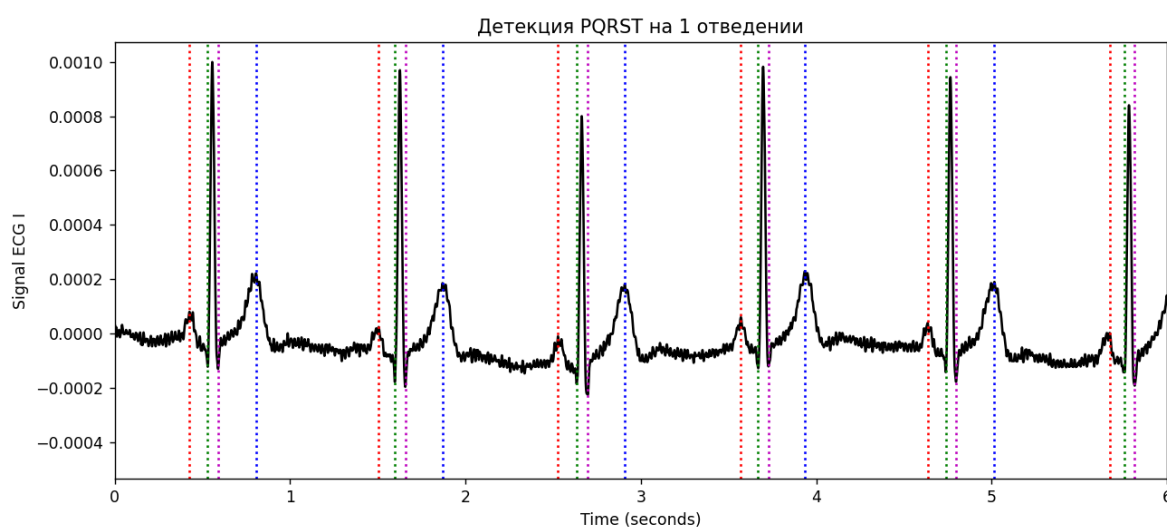


Рисунок 11 – Детекция точек P, Q, S, T

В программе реализован принцип повторной попытки детектировать точки в случае неуспешной попытки на первом отведении. Во время второй попытки используется для анализа второе отведение ЭКГ. При повторном

отсутствии результатов программа выдает уведомление о невозможности провести детектирование и прерывает свое выполнение.

2.2.4 Фильтрация сигнала

Еще одним из ключевых факторов, искажающим вид кардиограммы, является наличие артефактов дыхания. Они представляют собой колебания низких частот, что видно на рисунке 12.

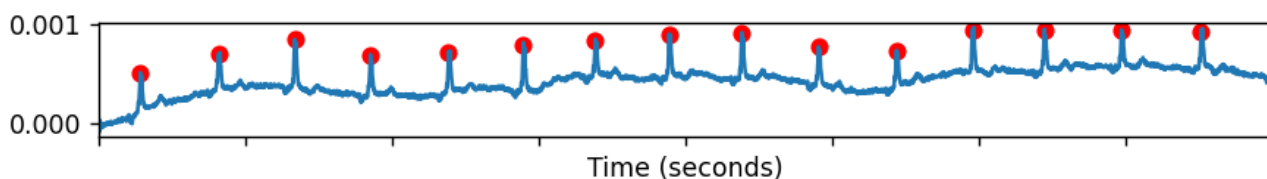


Рисунок 12 – Иллюстрация артефактов дыхания на кардиограмме

Для обработки данных артефактов имеется возможность применить фильтр высоких частот (ФВЧ) Баттерворта первого порядка (рисунок 13).

В процессе исследований был определен наилучший диапазон частот среза, который является одним из параметров при парсинге аргументов (--f_sreza) в финальном скрипте.

Наилучший диапазон для частоты среза составляет 0.5-0.8 Гц

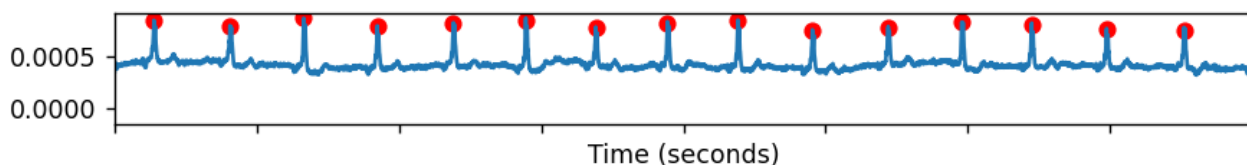


Рисунок 13 – Результат после цифровой ФВЧ фильтрации

При использовании фильтров более высокого порядка, происходят сильные искажения исходного ЭКГ, что делает дальнейшие результаты построения ВЭКГ ошибочными.

Положительный аспект фильтрации можно наблюдать при задании большого интервала по кардиоциклам для 3D построения ВЭКГ (рисунок 14, 15).

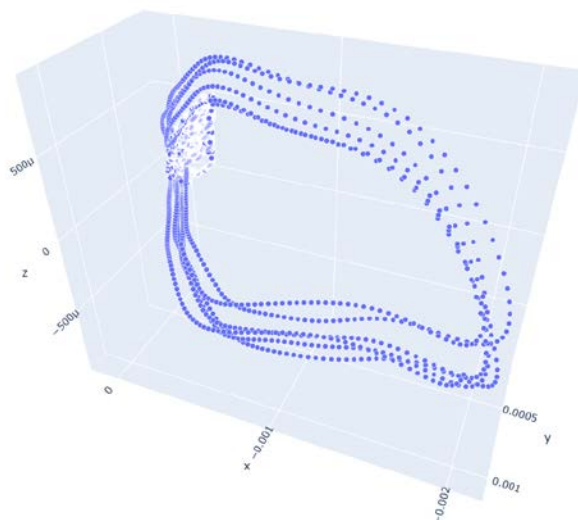


Рисунок 14 – Построение 6 последовательных периодов ВЭКГ без применения ФВЧ

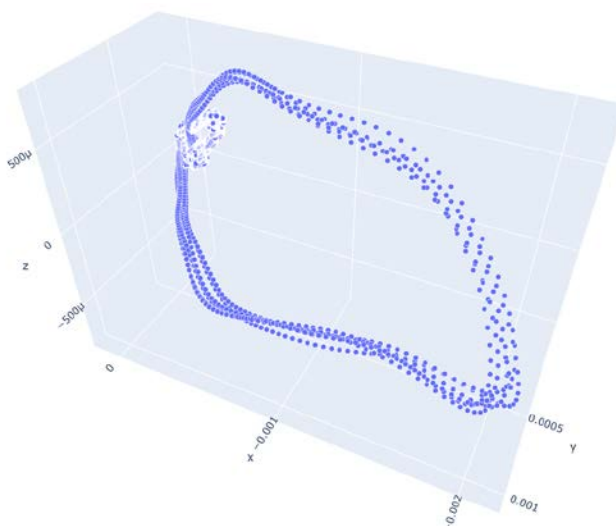


Рисунок 15 – Построение 6 последовательных периодов ВЭКГ с применения ФВЧ

Хорошо заметно, что при использовании фильтрации все циклы практически полностью идентичны и повторяют один и тот же контур. Тогда как в случае без нее, циклы расходятся и не повторяют друг друга.

2.3 Построение векторкардиограммы

2.3.1 Расчет и визуализация векторкардиограммы

ВЭКГ визуализируется в трех основных анатомических плоскостях человека (рисунок 16) и рассчитывается по формулам Дауэра (описаны в 1

главе), представленным ниже, которые позволяют сформировать массив проекций на основе первого и второго главных и шести грудных отведений ЭКГ [27]:

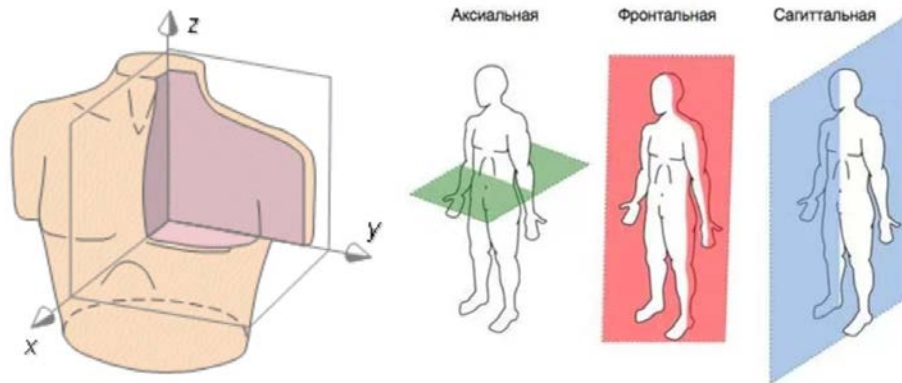


Рисунок 16 – Анатомические плоскости [14]

$$X = 0,172 \cdot V_1 + 0,074 \cdot V_2 - 0,122 \cdot V_3 - 0,231 \cdot V_4 - 0,239 \cdot V_5 - 0,194 \cdot V_6 - 0,156 \cdot D_I + 0,01 \cdot D_{II}, \quad (2)$$

$$Y = 0,057 \cdot V_1 - 0,019 \cdot V_2 - 0,106 \cdot V_3 - 0,022 \cdot V_4 + 0,041 \cdot V_5 + 0,048 \cdot V_6 - 0,227 \cdot D_I + 0,887 \cdot D_{II}, \quad (3)$$

$$Z = 0,229 \cdot V_1 + 0,310 \cdot V_2 + 0,246 \cdot V_3 + 0,063 \cdot V_4 - 0,055 \cdot V_5 - 0,108 \cdot V_6 - 0,022 \cdot D_I - 0,102 \cdot D_{II}, \quad (4)$$

где $V_1, V_2, V_3, V_4, V_5, V_6$ – значения сигналов с грудных отведений;
 D_I, D_{II} – значения сигналов с первых двух основных отведений.

2.3.2 Сглаживание полученного ВЭКГ

Была добавлена возможность визуализировать сглаженный вариант векторной ЭКГ. Результат данного сглаживания соответствует классическим представлениям из литературы о направлении диполя сердца в процессе кардиоцикла (Рисунок 17 [28]).

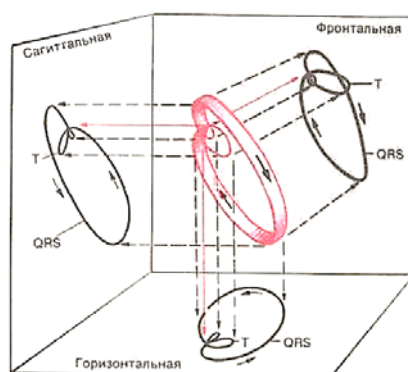
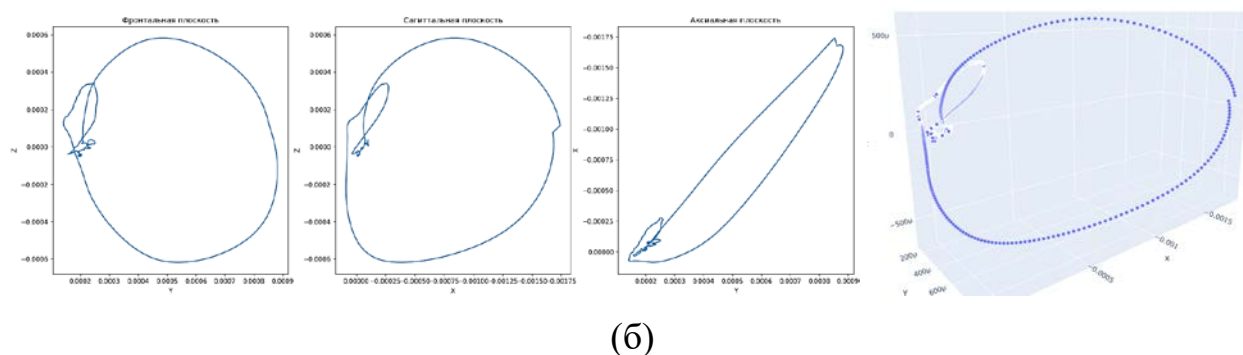
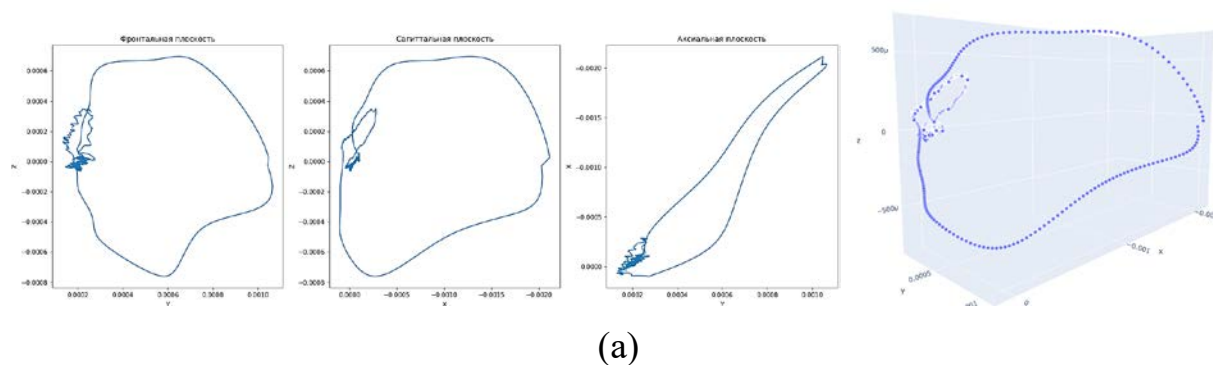


Рисунок 17 – Трехмерная векторная петля и ее проекции в трех плоскостях тела

Реализуется данное сглаживание с помощью фильтрации фильтром среднего массивов координат X, Y и Z у полученных сигналов.



а – до фильтрации, б – после фильтрации

Рисунок 18 – Реализация сглаживания петель

Значение окна фильтра равно целому числу, наиболее близкому к 2% от частоты дискретизации в герцах. Зависимость от частоты дискретизации обусловлена тем, что чем чаще берем точки при ресемплинге, тем больше нужно и окно чтобы убрать зигзаги на петлях.

Данная фильтрация хоть и сглаживает петлю ST-T, но значительно изменяет исходный внешний вид петли QRS, которая является информативным параметром при дальнейших исследованиях машинного обучения и получении информативных признаков. Результат применения фильтрации представлен на рисунке 18.

По этой причине данная фильтрация добавлена в код в виде опционального параметра, который будет отключен при генерации данных для обучения моделей.

2.3.3 Сохранение ВЭКГ как масштабированные изображения

В коде была добавлена возможность генерации датасета. Для каждого конкретного периода кардиоцикла и пациента можно создать 3 grayscale изображения, являющиеся проекциями ВЭКГ на основные плоскости. Это позволит в дальнейших исследованиях обучать нейронные сети и алгоритмы машинного обучения для классификации нормы и патологии, рассматривая данные ВЭКГ как изображения.

Создание множества фотографий на основе одного пациента с отличием по номеру цикла было добавлено для реализации естественной аугментации датасета без использования трансформаций.

Для того, чтобы форма ВЭКГ по всем проекциям соответствовала исходным масштабам и сами петли располагались соответственно реальным расположениям была реализована процедура масштабирования. Пример масштабирования представлен на рисунке 19.

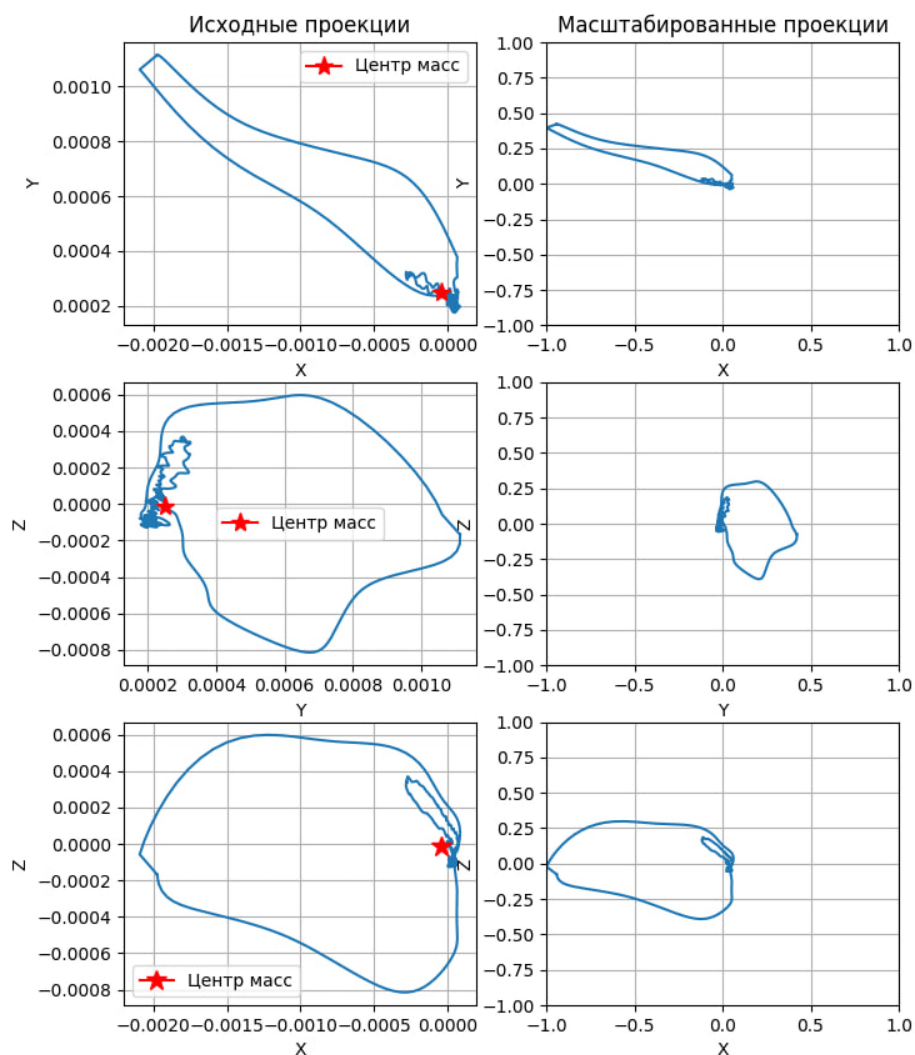
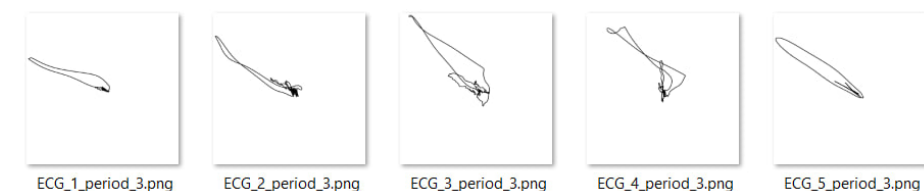


Рисунок 19 – Реализация масштабирования

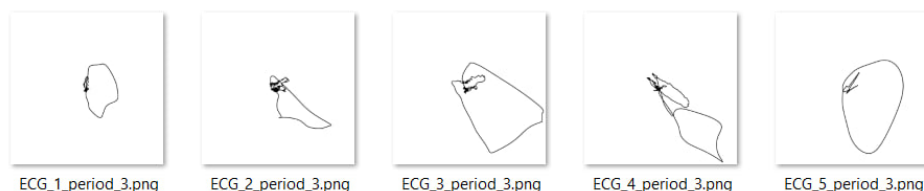
Первым делом вычисляется расположение центра масс ВЭКГ (то есть точка среднего значения по всем осям). Далее происходит вычитание координаты центра масс из всего массива точек ВЭКГ, что позволяет итоговый график перенести в координату нуля по трем осям.

После этого осуществляется определение максимальной по модулю координаты преобразованного векторного представления ЭКГ. После чего весь исходный массив трехмерных точек делится на это вычисленное значение. Таким образом мы приводим исходный ВЭКГ в диапазон от -1 до 1 включительно с сохранением реальных пропорций и относительных местоположений по всем проекциям.

Пример получаемых изображений проекций представлен на рисунке 20.



(а)



(б)



(в)

а – аксиальные XY проекции, б – фронтальные YZ проекции, в – сагиттальные XZ проекции

Рисунок 20 – Пример создания датасета изображений на основе пяти результатов ЭКГ после процедуры масштабирования (в примере представлен обработанный третий период кардиоцикла)

2.3.4 Сохранение ВЭКГ как облако точек

Для дальнейшего исследования возможности определения патологий на основе ВЭКГ было добавлена возможность сохранения ВЭКГ в виде облака точек в формате csv, где имеется 3 столбца (проекции на x, y, z).

Создание множества облаков на основе одного пациента с отличием по номеру цикла было добавлено для реализации естественной аугментации датасета без использования дополнительных трансформаций.

2.3.5 Результаты построения ВЭКГ испытуемых

В рамках преддипломной практики были проведены исследования на нескольких испытуемых, изъявивших желание.

Рассмотрим пример получения ВЭКГ с использованием исходных ЭКГ сигналов в формате edf, зарегистрированных на трех испытуемых с помощью прибора МИОКАРД-12 в НОМТЦ МГТУ им. Н.Э. Баумана.

На рисунке 21 представлены результаты в проекциях на главные плоскости. На рисунке 22 представлен вид интерактивного 3D режима визуализации результатов обработки векторного ЭКГ. Для более привычного представления проекций направление оси x на аксиальной и сагиттальной проекциях изменено.

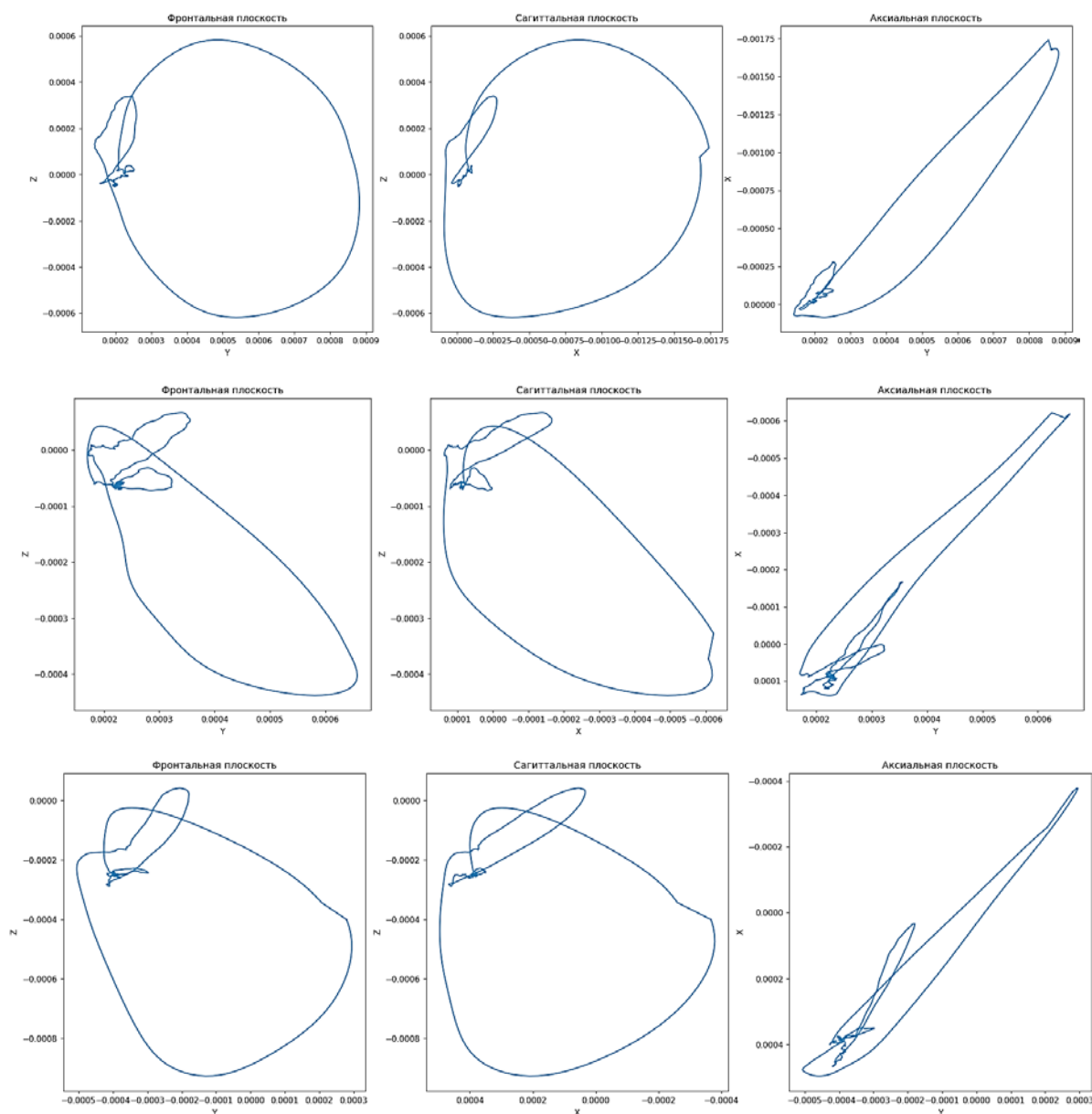


Рисунок 21 – Визуализация ВЭКГ в трех проекциях для трех испытуемых

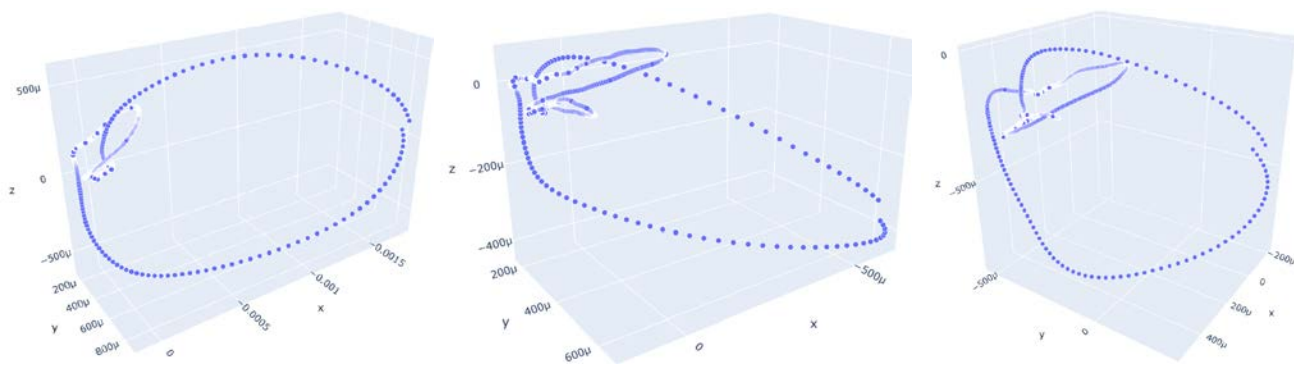


Рисунок 22 – 3D визуализация ВЭКГ для трех испытуемых

2.4 Определение физиологических параметров

2.4.1 Выделение QRS и ST-T петель

Для вычисления областей петель используются найденные точки пиков Q, S, а также точка окончания Т интервала.

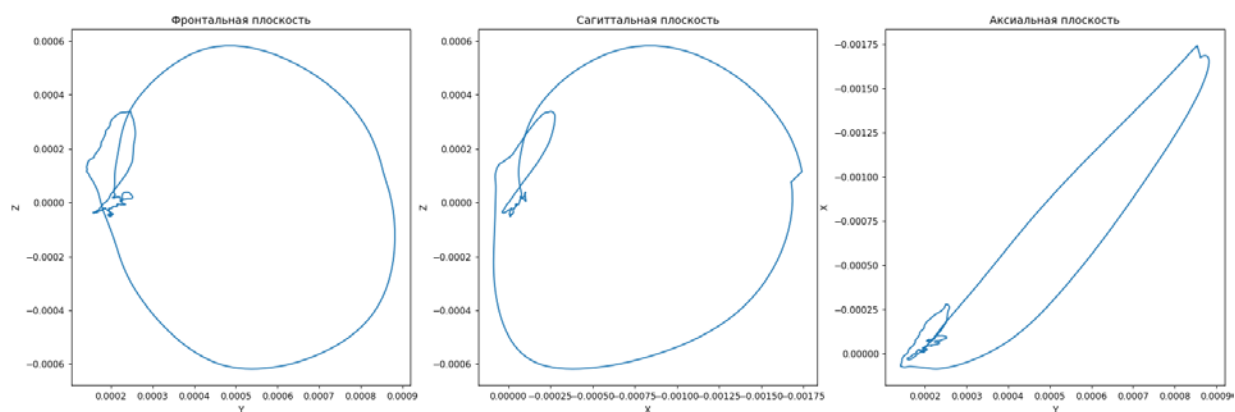
Так как все области строятся согласно введенному пользователем периоду кардиоцикла, поэтому все точки, интересующие нас для обработки петель, находятся как ближайшие к заданному по счету R зубцу.

Петля QRS определяется как набор точек ВЭКГ, лежащих между ближайшими Q и S пиками.

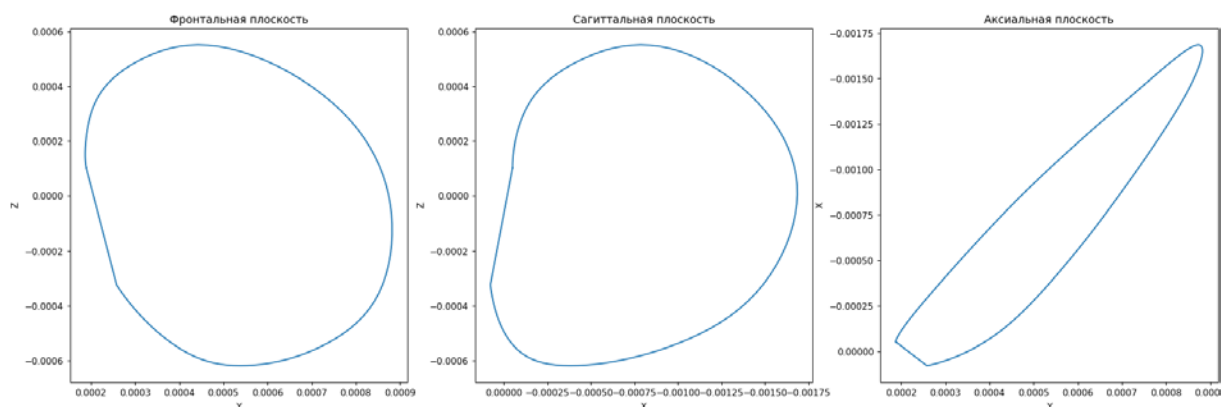
Петля ST-T определяется как область, начинающаяся на 25 миллисекунд позднее ближайшей точки S и заканчивающаяся в точки окончания ближайшего Т участка.

Если отобразить лишь области, определенные выше и сделать их замкнутыми благодаря соединению последней точки с первой, то мы получим полностью замкнутые петли, которые уже можно будет использовать для вычисления площадей, а также значения угла QRST.

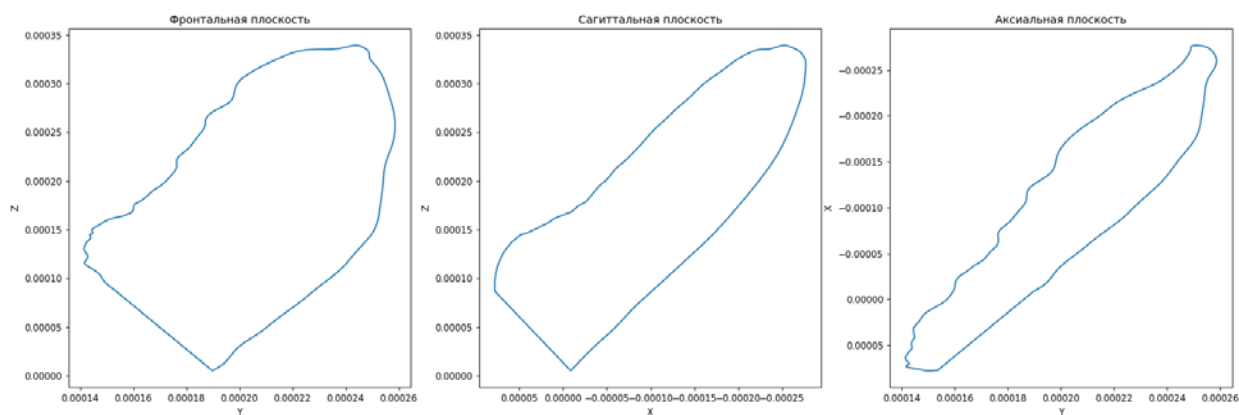
Пример определения петель представлен на рисунке 23.



(а)



(б)



(в)

а – полный ВЭКГ, б – петля QRS, в – петля ST-T

Рисунок 23 – Нахождение петель в исходной ВЭКГ

2.4.2 Вычисление площадей петель

Для вычисления площадей проекций петель был реализован подход с вычислением площади замкнутого полигона.

Каждый замкнутый контур можно разбить на большое число трапеций и посчитать финальную площадь петли как сумму площадей таких участков.

У петли ST-T иногда имеется большое число самопересечений в проекциях. Из-за этого площадь у такой фигуры зачастую интуитивно не определена, поэтому данный алгоритм может давать небольшие погрешности в финальном результате.

В случае с QRS петлей таких проблем не возникает, поэтому ее площадь практически всегда определяется достоверно.

2.4.3 Определение угла QRS-T

Угол QRS-T – это угол между вектором, представляющим среднюю электродвижущую силу QRS, и вектором, представляющим среднюю электродвижущую силу волны ST-T [29].

В норме этот угол во фронтальной плоскости равен менее 60° (обычно менее 30°). Выход угла QRST за указанные пределы означает нарушение взаимосвязей между свойствами возбуждения и восстановления.

Вычисление данного угла производится следующим образом:

На предыдущих шагах были определены значения облаков точек, соответствующих QRS и ST-T петлям. Поэтому, для нахождения средних электродвижущих силы требуется найти центр масс эти двух облаков (о есть среднее значение по всем осям).

После нахождения двух итоговых точек, соответствующих центрам масс, строится 2 вектора из начала координат и далее через скалярное произведение производится вычисление величины пространственного угла QRS-T.

Пример 3D построения этих векторов представлен на рисунке 24.

Синим представлен вектор, представляющий среднюю электродвижущую силу QRS. Красным показан вектор, представляющий среднюю электродвижущую силу волны ST-T.

обработчиком EDF файлов, поэтому взаимодействие было заменено с передачи аргументов напрямую на более интуитивный формат GUI. Внешний вид главного экрана приложения представлен на рисунке 26.

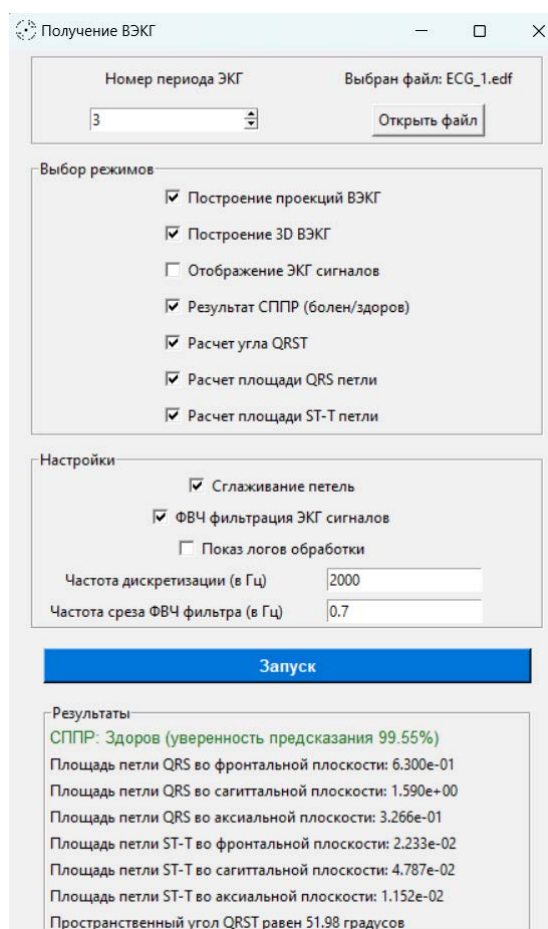


Рисунок 26 – Главный экран приложения

Была добавлена возможность интерактивного выбора файлов EDF с рабочего стола, включения и выключения имеющихся режимов отображения работы программы, а также возможность гибкой настройки режимов фильтрации и ресемплирования исходных ЭКГ сигналов.

Также были предусмотрены автоматические проверки наличия выбранного файла, существования заданного номера периода кардиоцикла и соответствие нужным типам данных для вводимых полей частот.

Программное обеспечение предоставляется врачам и разработчикам в виде файла установщика расширениям exe. После завершения процесса установки программу можно запустить непосредственно двойным нажатием мышки на иконку приложения.

Помимо запускаемого файла имеется файл конфигурации с расширением `json`, в котором можно изменить предустановленные параметры приложения и переключать режимы разработчика и врача. Отличием режима разработчика является наличие дополнительной опции по демонстрации графических логов обработки, таких как результаты детектирования опорных точек кардиограммы, пространственное расположение угла `QRST` и отдельное выделение петель `QRS` и `ST-T`.

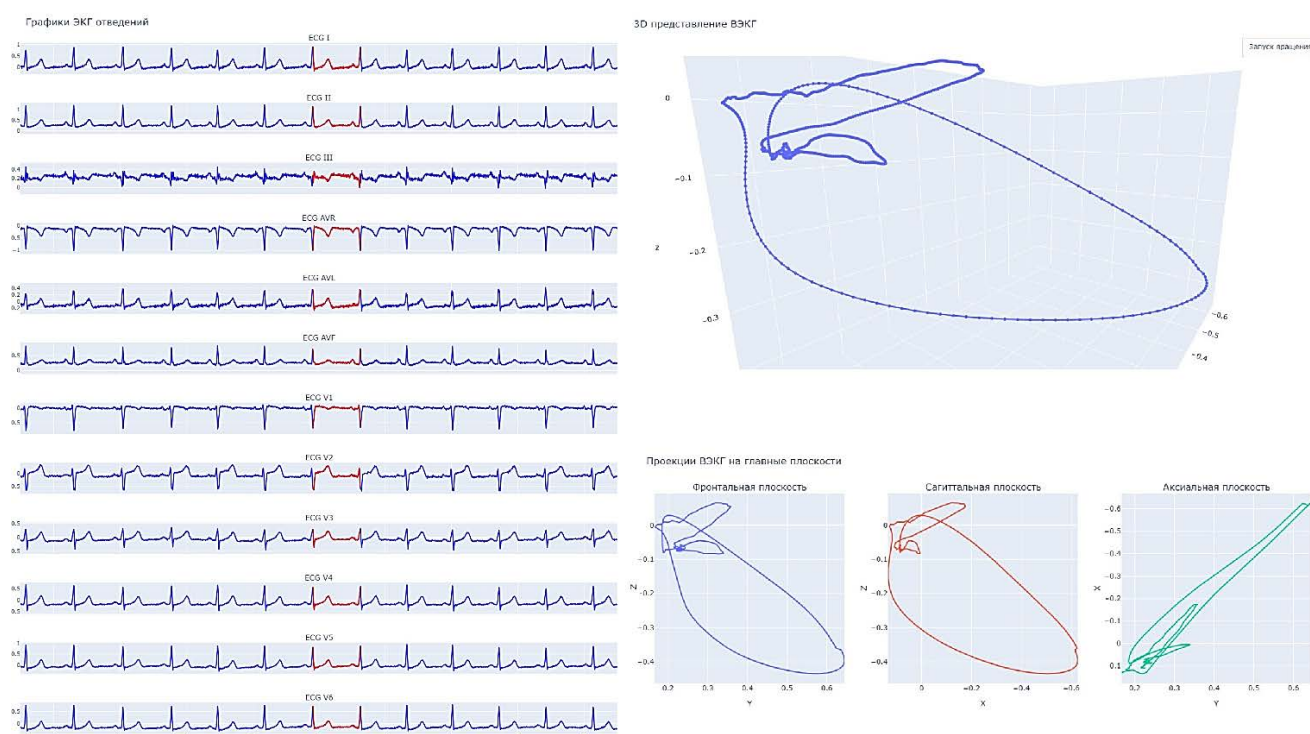


Рисунок 27 – Окна результатов обработки

Рядом с главным файлом программы имеется отдельная папка, в которой расположены веса обученной нейронной сети на базе PointNet, по поводу обучения которой можно прочесть в 6 главе, а также файл с архитектурой. Так как сеть на базе облаков точек показала самые низкие вероятности ошибок первого и второго родов, поэтому в проект был имплементирован режим системы поддержки принятия решений на основе данного подхода. При выборе такого режима по результатам обработки ВЭКГ конечному пользователю будет представлен результат предсказания бинарной метки класса и уверенность в нем на основе `softmax` функции к выходным значениям нейронов.

Итоговая программы может быть использована как интерактивный визуализатор сигналов отведений ЭКГ и трехмерного ВЭКГ на основе выбранного периода кардиоцикла, а также как СППР с возможностью вычисления информативных признаков на основе векторных петель.

Пример главных результирующих окон приложения представлен на рисунке 27. В данном примере был выбран седьмой период ЭКГ для построения векторных петель, а также включен режим сглаживания петель и ФВЧ фильтрации для удаления низкочастотных артефактов дыхания. Красным цветом выделяется область данного периода в окне визуализации отведений. Пример окон для просмотра логов обработки для режима разработчика представлен на рисунке 28.

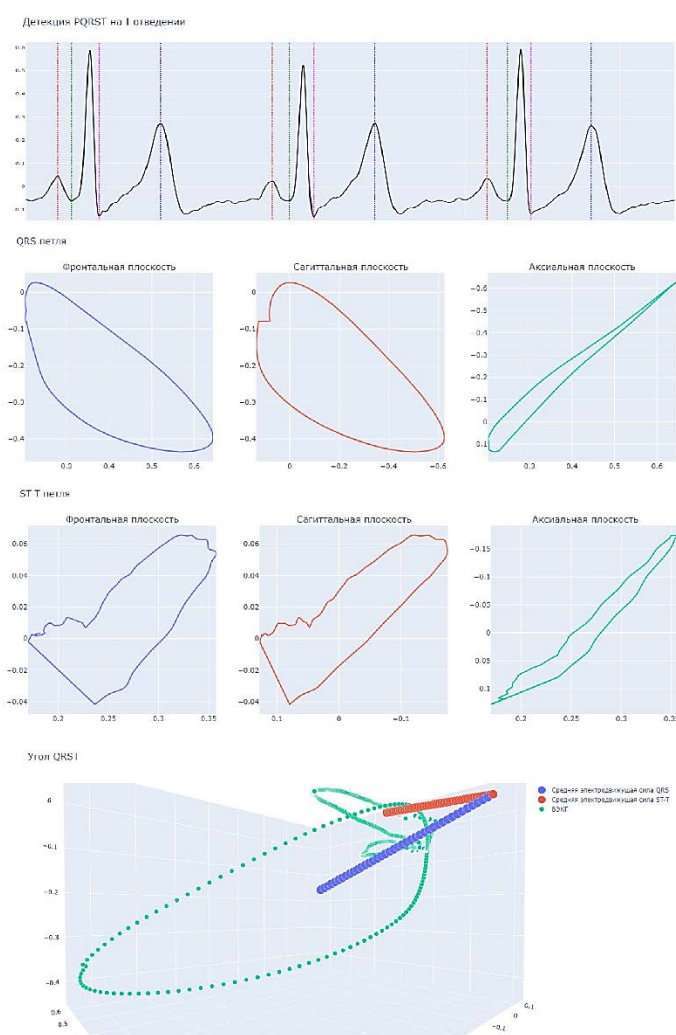


Рисунок 28 – Окна промежуточных графических результатов обработки

3 Создание обучающих данных для моделей машинного обучения

3.1 Исходный датасет ЭКГ

Для проведения исследований в области машинного обучения на основе представления данных в виде векторной кардиограммы был предоставлен набор размеченных сигналов ЭКГ.

Был представлен датасет, содержащий 342 файла формата edf и для каждого файла информация о параметре EF - фракция выброса левого желудочка, определенная по данным высокоточного ультразвукового аппарата. Условно считалось, что значение, которое 50 и более - показатель нормы. Менее 50 - признак сердечной недостаточности.

В данных имелась информация о поле каждого пациента, а также о его возрасте. Эти признаки тоже будут участвовать в моделях при разработке системы поддержки принятия решений на основе методов классического машинного обучения. Эти признаки будут добавлены к выделенным из ВЭКГ информативным параметрам.

3.2 Генерация датасета информативных признаков ВЭКГ

Для создания обучающих данных для моделей машинного обучения на основе информативных признаков был написан скрипт, реализующий проход в цикле по всем имеющимся файлам и генерации 7 основных признаков на основе каждого пациента: площади петли QRS в трех основных проекциях, площади ST-T петли в трех основных проекциях и значение пространственного угла QRS и проекции этого угла на фронтальную плоскость.

Так же для каждого пациента в список признаков добавляется значение пола и значение возраста. Так как признак пол является категориальным, по этой причине его представление переводится в one-hot представление [30].

Таким образом, для каждого пациента имеется 7 признаков от ВЭКГ и еще 3 отдельных признака: 2 численных признака на пол и 1 на возраст. Каждому пациенту соответствует численное значение фракция выброса левого желудочка и бинарный признак здоров (0) или болен (1).

Таким образом, по данной сформированной таблице будет проведено исследование классическими интерпретируемыми методами машинного обучения. Будет исследована задача бинарной классификации и регрессии на параметр EF.

3.3 Генерация датасета изображений на основе ВЭКГ

Для создания обучающих данных для моделей машинного обучения на основе изображений с трех основных отведений был написан скрипт, реализующий проход в цикле по всем имеющимся файлам и генерации для каждого пациента семи итоговых ВЭКГ. Функция при каждом проходе пациента запрашивала периоды с 3 по 9 включительно. Тем самым получалось порядка 21 итоговых изображений, так как по 3 изображения черно-белых создается для одного кардиоцикла (проекции на три главные плоскости).

Причина считывания, начинающегося лишь с 3 периода, связана с тем, что на более ранних периодах возможны искажения сигнала, связанные с фильтрацией ЭКГ перед генерацией результирующих векторных представлений. Такие искажения могут быть существенны для обучения моделей.

Далее был написан код по комбинированию этих трех одноканальных grayscale изображений в одно RGB изображение с помощью конкатенации каналов. В итоге получалось, что итоговое изображение в R канале хранит проекцию на сагиттальную плоскость, в G на аксиальную плоскость и B на фронтальную плоскость. Пример такой комбинации представлен на рисунке 29.

Таким образом, были получены для каждого пациента примерно по 7 изображений. Число изображений с одного пациента не равно строго семи так как у некоторых пациентов из-за низкой частоты сердцебиения физически отсутствовали некоторые номера запрашиваемых периодов и код пропускал такую обработку переходя к новому пациенту. Благодаря тому, что для каждого пациента генерировалось сразу много ВЭКГ, получилось значительно

нарастить датасет более чем в 5 раз используя естественную аугментацию данных.

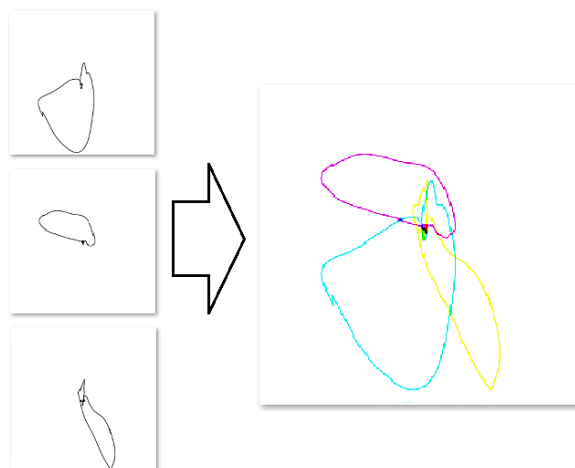


Рисунок 29 – Комбинирование одноканальных изображений проекций в одно трехканальное изображение

Из-за того, что повороты и масштабы значимы при анализе изображений проекций ВЭКГ, поэтому на тренировочном цикле не будут применяться никакие виды аугментаций помимо вышеописанных естественных для избегания сильного различия от исходных представлений.

Далее стояла задача по разделению данных на тренировочную и валидационную выборки. При этом важным аспектом являлось разделение таким образом, чтобы в разные классы не могли попасть изображения с одного и того же пациента. Был написан скрипт, комбинирующий все изображения с одного пациента вместе и определяющий случайным образом к какому классу стоит отнести все собранные фотографии при этом с учетом пропорции 75% на обучение и 25% на тестирование.

Результатом работы этой функции являлся готовый датасет, в котором имелось 2 папки: train и val. Внутри этих папок расположены изображения, отнесенные случайным образом благодаря функции. Так же в каждой из этих папок имелся csv файл, в котором находилась информация о значении фракции левого желудочка (EF) согласно исходной excel таблице с разметкой для каждой имеющейся фотографии в ней.

В итоговом получившемся датасете на трейне имелось 1711 и на валидации 558 изображений.

3.4 Генерация датасета облаков точек ВЭКГ

Для создания обучающих данных для моделей машинного обучения на основе данных в виде облаков точек был написан скрипт, реализующий проход в цикле по всем имеющимся файлам и генерации для каждого пациента семи итоговых ВЭКГ. Функция при каждом проходе пациента запрашивала периоды с 3 по 9 включительно. Тем самым получалось порядка 5 итоговых csv файлов с данными о координатах точек векторных петель.

По той же самой причине, что и со случаем по генерации датасета изображений, стартовый период был выбран равен 3. Остальные периоды до 9 включительно набирался для естественной аугментации по мере их наличия в исходных сигналах ЭКГ. Удалось произвести увеличение размера датасета более чем в 5 раз благодаря такому подходу.

При генерации облаков точек было выбрано фиксированное значение частоты дискретизации сигналов пациентов, равное 700 Гц. Так как у каждого человека значение периода кардиоцикла уникально, поэтому число точек в облаках не совпадает. Распределение числа точек близко к нормальному со средним значением 560 (рисунок 30).

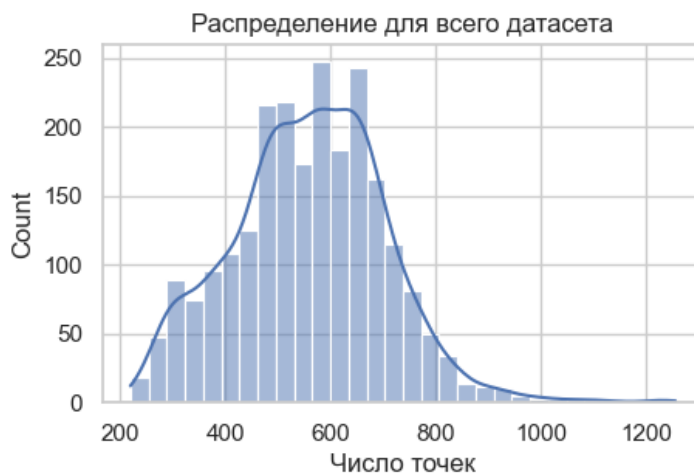


Рисунок 30 – Гистограмма распределения числа точек в облаках сформированного датасета

Далее стояла задача по разделению данных на тренировочную и валидационную выборки. При этом важным аспектом являлось разделение таким образом, чтобы в разные классы не могли попасть облака точек с одного и того же пациента. Был написан скрипт, комбинирующий все csv файлы с одного пациента вместе и определяющий случайным образом к какому классу стоит отнести все собранные файлы пациента с учетом пропорции 75% на обучение и 25% на тестирование.

Результатом работы этой функции являлся готовый датасет, в котором имелось 2 папки: train и val. Внутри этих папок расположены csv файлы с координатами облаков точек, отнесенные случайным образом благодаря функции. Так же в каждой из этих папок имелся ground_truth.csv файл, в котором находилась информация о значении фракции левого желудочка (EF) согласно исходной excel таблице с разметкой для каждого облака в ней.

В итоговом получившемся датасете на трейне имелось 1757 и на валидации 578 облаков.

4 Машинное обучение на основе информативных признаков

4.1 Применение машинного обучения

В век высоких технологий, где данные и информация играют центральную роль, медицинская сфера также подвергается влиянию передовых научных достижений. Однако в отличие от многих других областей, медицинская практика накладывает строгие требования к точности и надежности, ведь речь идет о здоровье и жизнях людей.

В этом контексте машинное обучение становится ключевой технологией, которая позволяет значительно улучшить качество медицинской помощи. Алгоритмы позволяют врачам увеличивать скорость постановки окончательного диагноза, а также увеличить точность, минимизируя человеческий фактор ошибки и усталости [31]. Они выступают в качестве системы поддержки принятия решений.

Зачастую, концепт постановки заболевания основывается на обработке собранного анамнеза пациента. То есть врач опирается на информативные признаки, представленные численными или категориальными данными, чтобы сделать вывод о том или ином заболевании [4].

Именно поэтому, для решения поставленной задачи по созданию технологии оценки величины фракции левого желудочка и определения диагноза по ВЭЖГ был рассмотрен принцип представления итоговых данных о пациенте в виде ключевых информативных параметров, полученных на основе анализа векторных петель, а так же общих знаниях о пациенте (возраст и пол).

Для построения моделей, прогнозирующих параметры на основе признаков, преимущественно выбирают методы на основе классических алгоритмов машинного обучения (ML) [32], [6]. Данные алгоритмы в отличии от полносвязных нейронных сетей являются наиболее предсказуемыми, исследованными, а главное наиболее интерпретируемыми.

В медицине интерпретируемость - это ключевой момент. Медицинские решения требуют понимания и доверия, а модели классического ML как раз предоставляют понятные правила и объяснимые результаты. Это существенно

важно при обсуждении диагнозов и лечебных решений между врачами и пациентами [31]. Это позволяет повысить степень доверия конечного потребителя – врача к разработанной СППР. Врач способен понять, что конкретно из представленных физиологических параметров сыграло главную роль, а что вообще не влияло при выставлении финального решения предлагаемой моделью.

В области глубокого машинного обучения интерпретируемость практически отсутствует. Имеется множество научных исследований в области расшифровки алгоритмов работы сетей, но в большинстве случаев для врачей и даже самих разработчиков эти алгоритмы представляются как черный ящик, выдающий предсказание согласно внутренним алгебраическим преобразованиям с весами, подобранными по принципу минимизации функции ошибки.

Еще одним важным достоинством классического ML является необходимость в меньшем объеме данных для обучения [33]. Это может быть критически важно в случаях редких заболеваний или ограниченного доступа к большим медицинским наборам данных. К тому же, скорость обучения и предсказания моделей машинного обучения оказывается выше, что сокращает время реакции и позволяет быстро принимать важные решения.

В проведенном мной исследовании, рассматривались различные виды классических алгоритмов по классификации для определения статуса пациента - болен или здоров, а также регрессии для определения численного признака фракции (EF) левого желудочка, измеряемого в процентах.

4.2 Разведочный анализ и переподготовка данных

Перед тем как начать ML исследования, важным этапом являлся разведочный анализ данных (Exploratory Data Analysis) и предобработка. Рассмотрим этапы, которые были проведены для подготовки данных к дальнейшим анализам и моделированию.

Первым этапом была проведена оценка корреляций всех признаков. Была обнаружена полная отрицательная корреляция между признаками one-hot преобразованного категориального бинарного признака пола. Данный результат являлся очевидным из-за того, что когда один признак равен единице, второй в то же время равен 0 [30].

Так же высокое значение корреляции между EF и классом болен-здоров объяснимо тем, что метка класса основана на величине фракции левого желудочка. Если значение менее 50, то считаем пациента больным, а иначе здоровым.

Важным для анализа является определение признаков, имеющих высокую корреляцию с опорным исследуемым признаком фракции EF. Было определено, имеется высокая отрицательная корреляция, равна -0,43 с пространственным углом QRST и чуть ниже связь с его проекцией. Так же видна небольшая корреляция с признаком площади сагиттальной проекции петли QRS и гендером пациентов.

Следующим этапом разведывательного анализа данных проведено статистическое исследование значимости угла QRST для пациентов с наличием и отсутствием патологий.

Сначала было оценено распределение признака EF. Для проверки нормальности данных использовался тест Шапиро-Уилка. Значение p-value теста оказалось меньше уровня значимости 0,05, поэтому гипотеза о нормальности данных опроверглась.

Так как тест на нормальность не был пройден, поэтому был выбран для исследования значимости признака угол QRST на заболевание непараметрический тест Манна-Уитни [34].

Результаты анализа с использованием данного теста показали значительные различия между группами больных и здоровых пациентов в зависимости от значения угла QRST. Получившееся значение p, равное 0 (или очень близкое к нему), указывает на статистическую значимость различий.

В дальнейших исследованиях по обучению классических ML моделей будет определено, что этот угол является во всех методах самым значимым из признаков в постановке итогового предсказания моделями.

После проведения разведывательного анализа данные были случайным образом разделены на обучающий и тестовый наборы в соотношении 75 на 25. При обучении моделей лучшие гиперпараметры будут определяться по результатам кросс-валидации, а финальная оценка качества будет осуществляться на отделенном тестовом датасете.

Для обеспечения стабильности и сходимости алгоритма обучения был использован `StandardScaler` [35]. Этот метод стандартизирует значения признаков, обеспечивая им среднее значение 0 и стандартное отклонение 1. Это важно, так как некоторые алгоритмы машинного обучения могут быть чувствительными к различиям в масштабе признаков [33]. Настройка алгоритма `StandardScaler` на тренировочных данных и его последующее применение на валидационных данных помогли обеспечить согласованный и надежный процесс обучения и оценки модели.

4.3 Бинарная классификация для идентификации заболеваний

4.3.1 Выбор метрик качества для задачи классификации

После описанной переподготовки было обнаружено, что количество примеров класса "болен" составляет 179, в то время как количество примеров класса "здоров" составляет всего 93. Такой дисбаланс классов мог оказать влияние на процесс обучения и оценку ML моделей [4]. Так что передо мной стоял вопрос выбора основных опорных метрик по оценке качества моделей машинного обучения, учитывающих этот факт.

В области разработок систем поддержки принятия врачебных решений степень значимости ошибок первого и второго рода не является идентичной [31]. Поставить ложно положительных диагноз менее чревато последствиями, нежели пропустить истинно больного и назвать его здоровым. Таким образом, параметр полноты (`recall`) на класс патологии кажется наиболее правильным в

оценке. Но на практике модель с высоким значением полноты зачастую может всех без разбора назначать больными. Так что более удачной метрикой при оценке является значение F1-score - среднее гармоническое полноты (recall) и специфичности (precision). Эта метрика обеспечивает баланс между способностью модели к правильной классификации пациентов с патологией и минимизации ложных срабатываний [36].

По этой причине одной из основных анализируемых метрик был выбран F1-score, рассчитанный для класса патологии.

В качестве дополнительной метрики для оценки моделей на валидационных данных был выбран взвешенный средний F1-Score. Эта метрика учитывает дисбаланс классов, предоставляя среднее значение F1-Score с весами, зависящими от размеров классов [36]. Таким образом, она позволяет более объективно оценить производительность модели на дисбалансированных данных, какими и являются они в исследуемом датасете.

4.3.2 Дерево решений

Дерево решений (Decision Tree) - это алгоритм машинного обучения, основанный на принципе иерархического разбиения данных на подгруппы с целью принятия решений. Он строит структуру дерева, в которой каждый узел представляет собой признак данных, а каждое разделение (ветвь) узла определяет пороговое значение этого признака для осуществления классификации [33].

К преимуществам данного алгоритма можно отнести его самую высокую степень интерпретируемости. Результат этого алгоритма схож со структурой if-else в программировании и идентичен мышлению людей при решении задач. Так же данный метод не требует проведения масштабирования признаков и устойчив к выбросам в данных.

Основными его недостатками являются его склонность к переобучению, особенно при большой глубине, что может привести к низкой способности к

обобщению на новые данные, а также чувствительность к небольшим изменениям в данных, что делает его нестабильным [33].

В ходе исследования и настройки классификатора на основе дерева решений были подобраны лучшие гиперпараметры. Мной исследовались следующие гиперпараметры: `max_depth` (максимальная глубина дерева), `min_samples_split` (минимальное количество образцов для разделения узла) и `min_samples_leaf` (минимальное количество образцов в листьях).

Для определения наилучших значений этих предзаданных параметров, а также для оценки обобщающей способности модели, была применена кросс-валидация. Кросс-валидация представляет собой метод, при котором данные разбиваются на несколько наборов (фолдов), а модель учится и оценивается на разных комбинациях этих фолдов [37]. Это позволяет получить оценку производительности модели и более объективно подобрать гиперпараметры, от которых в классическом машинного обучения сильно зависит качество финальной модели.

По результату обучения модели с определенными заранее гиперпараметрами была проведена его оценка качества классификации на тестовых данных. Пропорция классов на тесте совпадает с пропорцией, на которой производилось обучение. Результаты обучения, представленные в виде матрицы ошибок, можно увидеть на рисунке 31.

Нормализованная матрица ошибок полезна для более наглядного сравнения качества работы модели на разных классах, особенно в случаях, когда классы имеют неравные размеры или дисбаланс [33]. В нормализованной матрице ошибок сумма значений в каждой строке будет равна 1, и это позволяет легче интерпретировать долю правильных и ошибочных классификаций для каждого класса.

Итоговые результаты метрик качества модели занесены в таблицу 2.

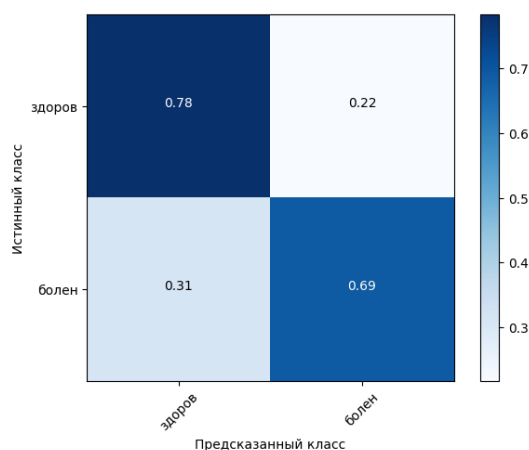


Рисунок 31 – Нормализованная матрица несоответствий метода дерева решений

В методе дерево решений важность признаков оценивается на основе того, как каждый признак используется при разделении данных в узлах дерева. Признаки, которые чаще используются для разделения считаются более важными для модели. Эта оценка важности признаков может быть выражена в виде процентов или относительных значений, позволяя определить, какие признаки вносят наибольший вклад в принятие решений моделью.

Представление значимости признаков можно увидеть на рисунке 32. Как видно на нем, самым важным признаком является площадь петли QRS в проекции на сагиттальную плоскость, а вторым по значимости является значение пространственного угла QRST. Признак пол не принимает участие в ветвлении дерева.

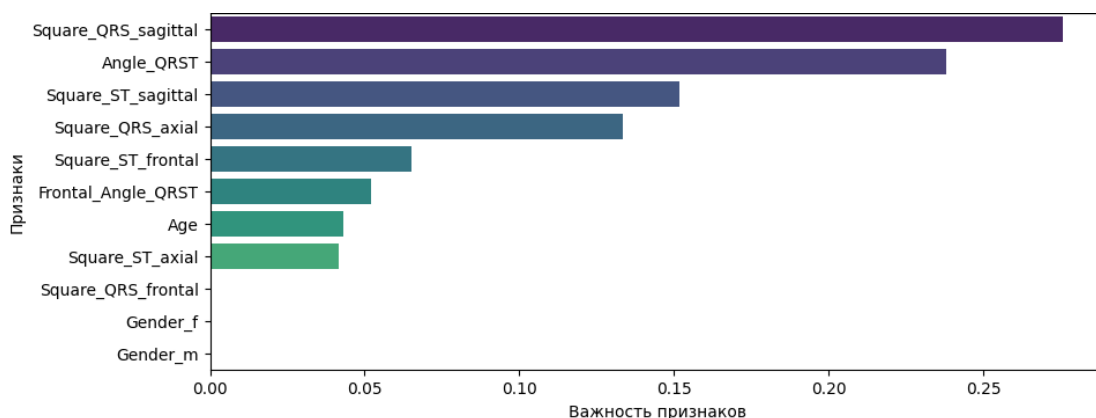


Рисунок 32 – Важность признаков в дереве решений

4.3.3 Случайный лес

Случайный лес (Random Forest) - это более мощный метод машинного обучения, чем классический метод дерева решений. Он основан на идее построения множества деревьев решений и объединения их результатов. В процессе обучения для каждого дерева создается подвыборка данных с возвращением (бутстрэп выборка), что способствует разнообразию данных и уменьшает переобучение [33].

Лучшие гиперпараметры модели подбирались с помощью кросс-валидации. Мной исследовались следующие параметры: `max_depth` (максимальная глубина дерева), `min_samples_split` (минимальное количество образцов для разделения узла), `min_samples_leaf` (минимальное количество образцов в листьях), а также число деревьев в ансамбле (`n_estimators`).

Данная модель показала значительно более высокие результаты на тестовом наборе данных и стала лидером среди всех исследованных мной моделей на информативных признаках. Результаты данной модели представлены на рисунке 33, а метрики указаны в таблице 2.

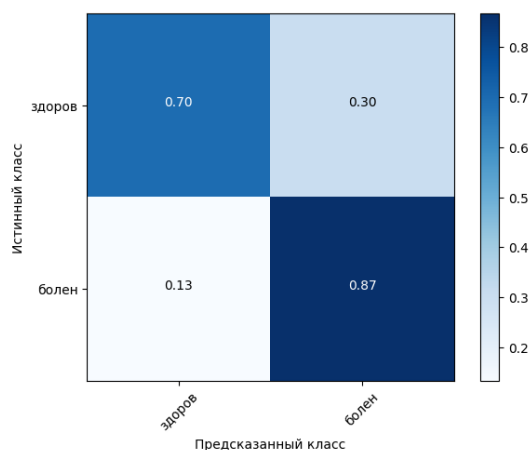


Рисунок 33 – Нормализованная матрица несоответствий метода случайный лес

Для данной модели наиболее значимыми параметрами для определения диагноза стали угол QRST, а также площадь петли ST в проекции на сагиттальную плоскость (рисунок 34).

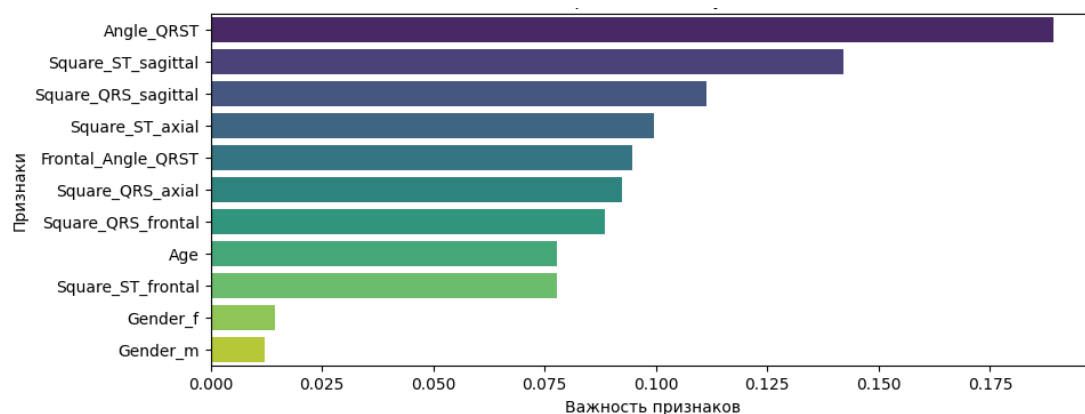


Рисунок 34 – Важность признаков в модели случайный лес

4.3.4 Логистическая регрессия

Логистическая регрессия - это статистический метод машинного обучения для решения задач классификации. Основное его предназначение - прогнозировать вероятность принадлежности объекта к одному из двух классов на основе линейной комбинации признаков. Логистическая регрессия использует логистическую функцию для преобразования линейной комбинации в вероятности и может быть интерпретирована как модель вероятности [33].

При выборе порога отсечения классов равным 0,5 были получены наилучшие результаты для данной модели. Относительно иных рассмотренных моделей логистическая регрессия не показала высоких результатов. Причиной этого может служить тот факт, что зависимость класса от признаков не является линейной.

Результаты данной модели представлены на рисунке 35, а метрики указаны в таблице 2. Оценка значимости признаков для этой модели осуществляется по оценке значения модуля полученных коэффициентов для линейной комбинации признаков (рисунок 36).

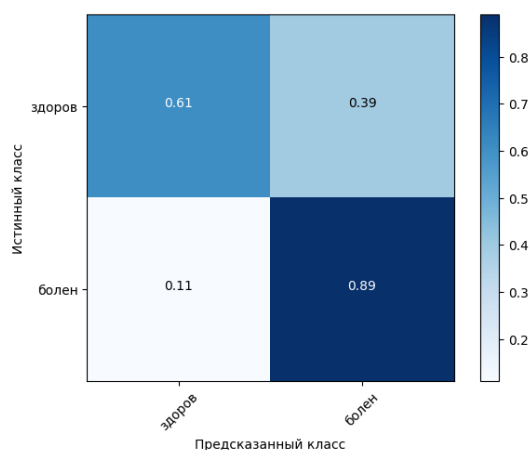


Рисунок 35 – Нормализованная матрица несоответствий метода логистическая регрессия

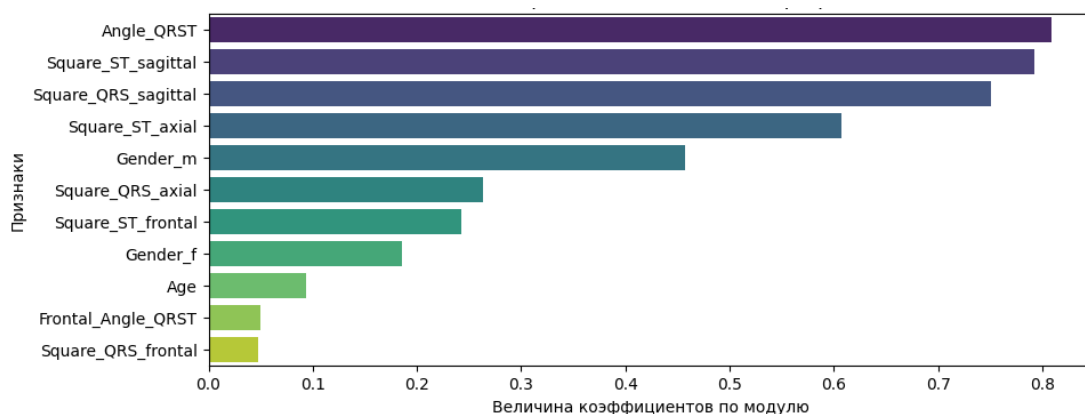


Рисунок 36 – Оценка значимости признаков для метода логистическая регрессия

4.3.5 К-ближайших соседей

Метод К-ближайших соседей (KNN) - это алгоритм машинного обучения, основанный на идее, что объекты, близкие в признаковом пространстве, имеют схожие характеристики. Преимущества метода KNN включают простоту реализации, простую интерпретируемость, способность к обработке неструктурированных данных и хорошие результаты на небольших наборах данных. Однако у него есть и недостатки, такие как высокая вычислительная сложность при большом объеме данных, так как алгоритм при инференсе проводит соответствие со всеми данными, которые встречались при обучении. То есть чем больше будет обучающих примеров, тем больше обработка

единичного запроса [37]. У метода также чувствительность к выбору параметра k (количество соседей) и необходимость нормализации признаков для улучшения производительности.

Именно этот параметр k и подбирался с использованием кросс-валидации с пятью фолдами. Лучшее значение оказалось равным 13.

Результаты данной модели представлены на рисунке 37, а метрики указаны в таблице 2. Модель показала равновероятные ошибки в обе стороны (первого и второго родов)

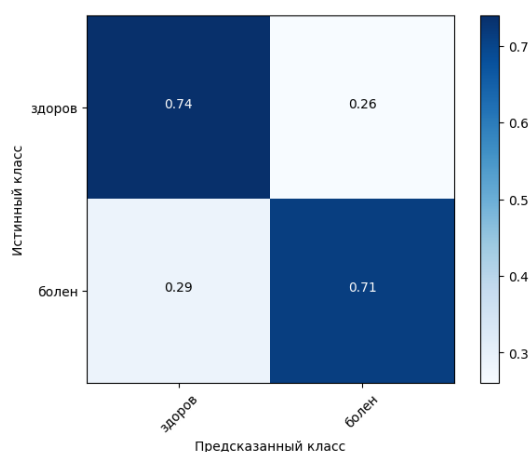


Рисунок 37 – Нормализованная матрица несоответствий метода KNN

Оценка значимости признаков для этой модели осуществляется с помощью Permutation Importance (значимость перестановок). Данная технология позволяет оценивать изменение качества модели при случайном изменении одного признака в тестовой выборке. На рисунке 38 можно увидеть, что самым значим признаком является пространственный угол QRST.

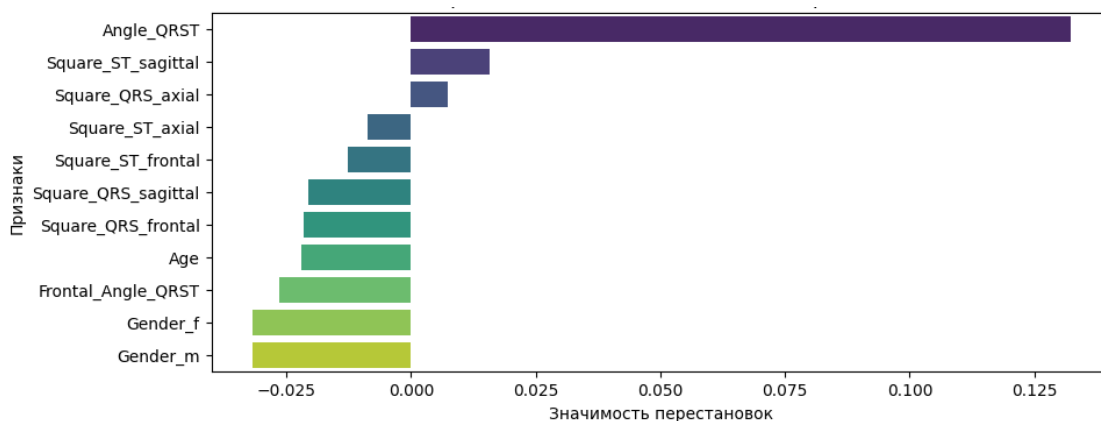


Рисунок 38 – Оценка значимости признаков для метода KNN

4.3.6 Метод опорных векторов

Метод опорных векторов (Support Vector Machines, SVM) – популярный алгоритм машинного обучения для задачи классификации и регрессии. Основная идея SVM заключается в поиске гиперплоскости, которая наилучшим образом разделяет данные разных классов. Гиперплоскость выбирается таким образом, чтобы максимизировать расстояние между ближайшими точками разных классов [37]. Значение полиномиальной функции, которая используется в методе опорных векторов с ядровой функцией выступает в виде гиперпараметра, который можно определить.

Благодаря кросс-валидации была определена степень, равная единице. То есть в качестве разделяющей поверхности выступает плоскость с полиномиальной степенью, равной единице.

Результаты обучения данной модели представлены на рисунке 39, а метрики указаны в таблице 2.

Метод SVM не предоставляет непосредственной оценки важности признаков. Это связано с тем, что данный алгоритм работает с гиперплоскостью, которая разделяет классы и не имеет внутреннего механизма для определения важности признаков.

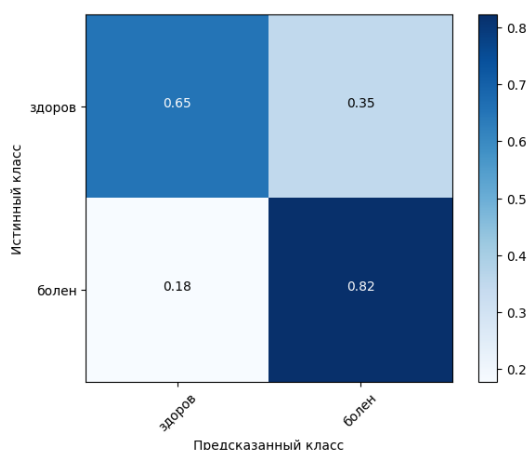


Рисунок 39 – Нормализованная матрица несоответствий метода SVM

4.3.7 Градиентный бустинг

Метод градиентного бустинга – один из самых популярных алгоритмов машинного обучения, который используется для задач классификации и регрессии. Его основной принцип действия заключается в построении ансамбля слабых моделей (пеньков) таким образом, чтобы каждая следующая модель корректировала ошибки предыдущей. Это достигается путем минимизации градиента функции потерь. Градиентный бустинг обеспечивает высокую точность и устойчивость к переобучению и является одним из наиболее эффективных алгоритмов машинного обучения [37].

Градиентный бустинг является самым распространенным алгоритмом в сфере разработок ML решений на информативных признаках и пользуется большой популярностью в сфере медицинских СППР.

Была использована технология кросс-валидации для подбора наилучших параметров скорости обучения (learning rate), значения числа последовательных алгоритмов, а также глубины единичного дерева. Лучшим значением глубины оказалось 3, а число деревьев равным 200.

При обучении моделей данного типа использовалась библиотека CatBoost [38]. Результаты обучения представлены на рисунке 40, а посчитанные метрики качества указаны в таблице 2. Модель имеет хорошие результаты для класса патология, но часто называет здоровых людей больными.

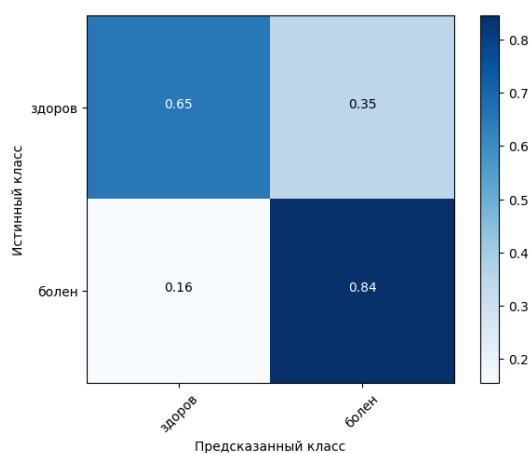


Рисунок 40 – Нормализованная матрица несоответствий метода градиентного бустинга

Для данного метода значимыми признаками выступают площади петель. Это является отличительной особенностью данного алгоритма. Во всех иных алгоритмах значение угла QRS занимало лидирующее положение в значимости, но для CatBoost лишь 4 место заняло (рисунок 41).

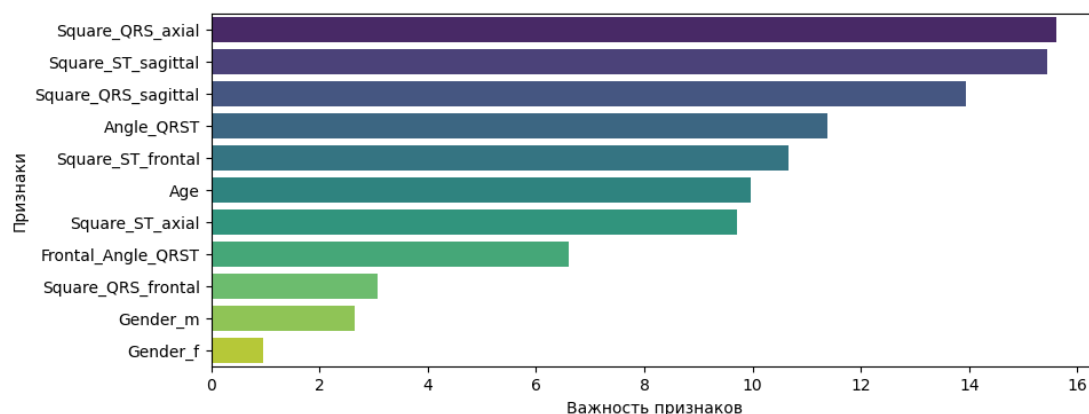


Рисунок 41 – Оценка значимости признаков для классификации градиентным бустингом

4.3.8 Сравнение ML алгоритмов классификации

Лучшие результаты на тестовом датасете показала модель случайного леса достигнув результатов точности 81%, а метрики F1 по классу патология 0,86. Взвешенное среднее для метрики F1 для данной модели равняется 0,81.

Так же лидирующие позиции заняли модели логистической регрессии и градиентного бустинга. Модель логистической регрессии показала наименьшую вероятность ошибки отнесения больного пациента к классу здоровых, равную 11%.

Худшие результаты показали модели KNN и дерево решений.

Результаты метрик на тестовом датасете для лучших моделей с подобранными гиперпараметрами представлены в таблице 2.

Таблица 2 – Метрики качества ML моделей

	Дерево решений	Случайный лес	Логистическая регрессия	KNN	SVM	Градиентный бустинг
accuracy	0,72	0,81	0,79	0,72	0,76	0,78
F1-score	0,77	0,86	0,85	0,77	0,82	0,84
Взвешенный F1-score	0,73	0,81	0,79	0,73	0,76	0,78
Precision	0,86	0,85	0,82	0,84	0,82	0,83
Взвешенный precision	0,76	0,81	0,79	0,75	0,76	0,78
Recall	0,69	0,87	0,89	0,71	0,82	0,84
Взвешенный recall	0,72	0,81	0,79	0,72	0,76	0,78

4.4 Определение величины фракции левого желудочка

4.4.1 Выбор метрик качества для задачи регрессии

Задача определения числовой величины с помощью алгоритмов машинного обучения называется задачей регрессии. Задачей является определить величину EF по имеющимся признакам для каждого пациента.

Опорными метриками для оценки качества прогнозирования EF будет являться средняя величина ошибки (MAE, Mean absolute error). Данная метрика показывает сколько в среднем ошибка в процентах у прогноза фракции [36].

Среднеквадратичная ошибка (MSE, Mean Squared Error) - это еще одна метрика, которая будет вычисляться для оценки качества модели предсказания. Она измеряет среднюю квадратичную разницу между фактическими и прогнозируемыми значениями. MSE имеет преимущество перед MAE в том,

что она штрафует модель за более крупные ошибки сильнее, что полезно в задачах, где большие ошибки более критичны. Однако эту метрику может быть сложнее интерпретировать, так как ее единицы измерения - квадраты единиц измерения целевой переменной [36].

4.4.2 Линейная регрессия

Линейная регрессия - это метод машинного обучения, используемый для прогнозирования числовых значений зависимой переменной на основе линейной комбинации независимых переменных. Он определяет наилучшие коэффициенты, минимизирующие разницу между фактическими и предсказанными значениями. Этот метод прост и интерпретируем, что делает его широко используемым [37].

Результаты обучения данной модели представлены в таблице 3. Гистограмма ошибок прогнозирования параметра фракции левого желудочка (EF) в процентах представлена на рисунке 42.

Оценка значимости признаков для этой модели осуществляется по оценке значения модуля полученных коэффициентов для линейной комбинации признаков (рисунок 43).

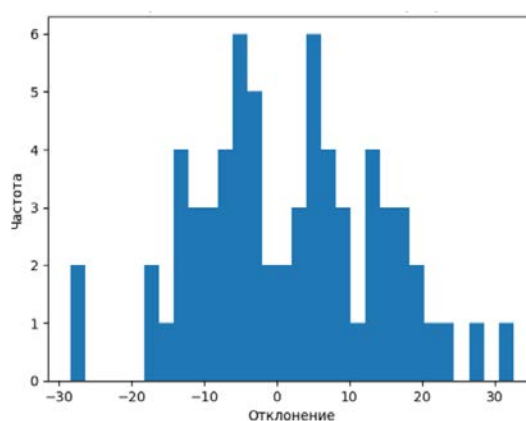


Рисунок 42 – Гистограмма отклонений предсказаний на тестовых данных линейной регрессией

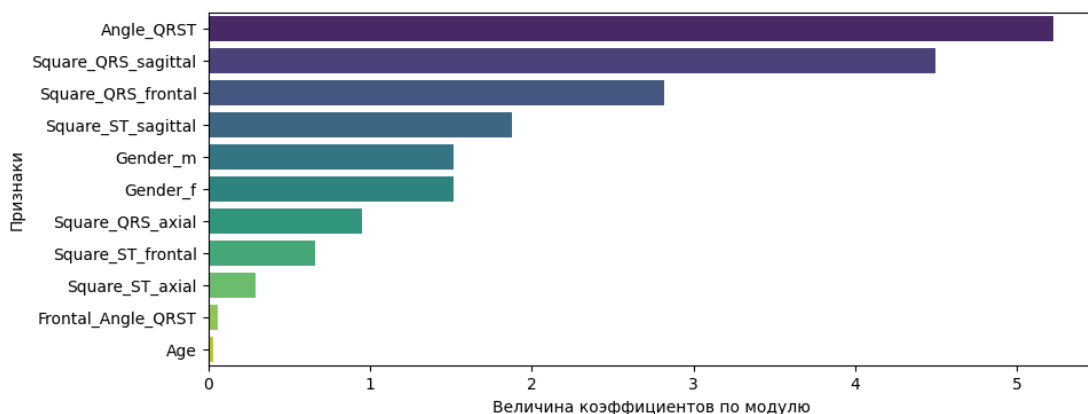


Рисунок 43 – Важность признаков для линейной регрессии

4.4.3 Градиентный бустинг

Градиентный бустинг помимо классификации способен решать и задачи регрессии. Для обучения моделей использовалась библиотека CatBooost.

Была использована технология кросс-валидации для подбора наилучших параметров скорости обучения (learning rate), значения числа последовательных алгоритмов, а также глубины единичного дерева. Лучшим значением глубины оказалось 6, а число деревьев равным 200.

Результаты обучения данной модели представлены в таблице 3. Гистограмма ошибок прогнозирования параметра фракции левого желудочка (EF) в процентах представлена на рисунке 44, а важность признаков для оценки на рисунке 45.

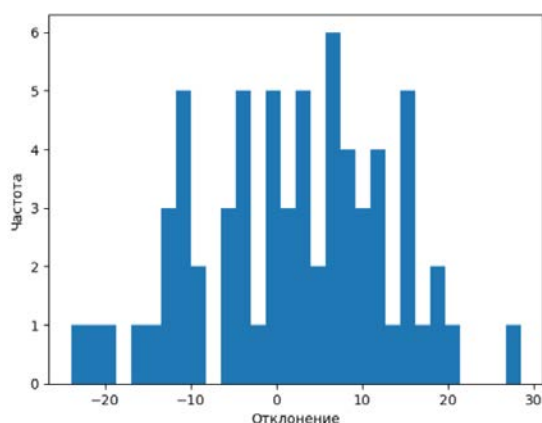


Рисунок 44 – Гистограмма отклонений предсказаний на тестовых данных градиентного бустинга

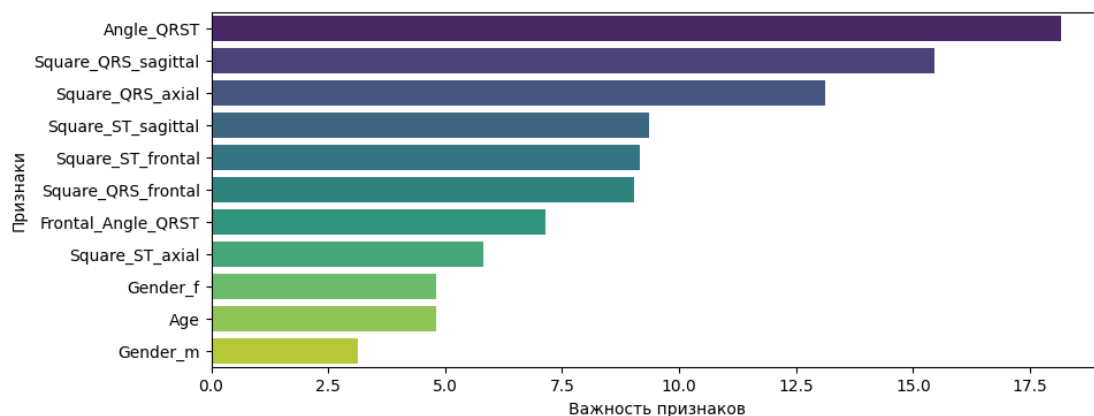


Рисунок 45 – Важность признаков для регрессии градиентным бустингом

4.4.4 Сравнение ML алгоритмов регрессии

Лучшие результаты на тестовом датасете показала модель градиентного бустинга, показав среднюю ошибку прогноза раной 9,2 %, в то время как линейная регрессия 10,4 % (таблица 3).

Таблица 3 – Метрики качества ML моделей регрессии

	Линейная регрессия	Градиентный бустинг
MAE	10,4	9,2
MSE	161,1	125,1

5 Машинное обучение на основе изображений проекций

5.1 Применение сверточных нейронных сетей

Нейронные сети являются мощным инструментом для анализа и обработки данных. Они состоят из нейронов, которые являются основными вычислительными единицами. Каждый нейрон принимает входные данные, обрабатывает их и передает результат следующему слою или на выход сети. Нейроны связаны между собой связями, которые имеют веса, определяющие вклад каждой связи в вычисления [39].

Принцип обучения нейронных сетей основан на минимизации функции ошибки, которая измеряет расхождение между предсказанными и фактическими значениями [40].

Для каждой конкретной задачи в области глубокого машинного обучения (DL) применяют определенные архитектуры сетей. Так, например, для задачи классификации и регрессии для входных данных в виде изображений применяют сверточные архитектуры.

Принцип работы сверточных нейронных сетей (Convolutional Neural Networks, CNN) основан на использовании сверточных слоев, которые позволяют автоматически извлекать признаки из изображений. Каждый сверточный слой содержит набор фильтров, которые проходят по изображению и выделяют определенные признаки, такие как границы, текстуры и формы. Затем результаты свертки объединяются в один массив признаков, который передается следующему слою [40].

Общая архитектура слоев в CNN:

- сверточные слои: применяют фильтры для извлечения локальных признаков,
- пулинг слои: уменьшают размерность данных и повышают инвариантность,
- слои активации: вводят нелинейность,
- полносвязные слои: используются для классификации и вывода предсказаний.

Сверточные нейронные сети являются одним из самых популярных типов нейронных сетей в области компьютерного зрения. Они широко используются в различных задачах, таких как распознавание объектов, сегментация изображений, классификация и детектирование.

Некоторые из наиболее известных архитектур сверточных нейронных сетей включают в себя VGG, ResNet, Inception и MobileNet. Эти архитектуры были разработаны для решения различных задач и имеют различные параметры и глубину.

В медицинской области сверточные нейронные сети также получили широкое применение. Они используются для анализа медицинских изображений, таких как рентгеновские снимки и МРТ-сканы, для диагностики различных заболеваний [41], [39].

Для решения задач в области обработки данных в виде изображений могут быть использованы как заранее обученные архитектуры сверточных нейронных сетей, так и специально разработанные модели, которые дообучаются на конкретных преподготовленных размеченных данных.

5.2 Преподготовка данных перед обучением

Для обучения сетей был подготовлен датасет, в котором для каждого изображения имелось соответствующее значение фракции левого желудочка и значение класса болен или здоров. Про то, как конкретно формировался данный датасет, было описано ранее. Согласно этим имеющимся данным, планировалось обучить нейронные сети на задачу бинарной классификации пациентов, а также регрессии для определения информативного параметра фракции изгнания.

Чтобы данные соответствовали размеру входного изображения, который используются в популярных архитектурах сверточных нейронных сетей, таких как ResNet, все имеющиеся изображения были приведены к размеру 224 на 224 пикселей. Кроме того, это может помочь ускорить процесс обучения модели,

так как меньший размер изображения означает меньше вычислительной нагрузки на модель.

Так же эксперименты показали, что для обучений сетей для классификации из-за сильной диспропорции классов модель не могла хорошо обучаться и начинала всех пациентов классифицировать как больных. Для решения данной проблемы было искусственно увеличено число здоровых примеров в 2 раза на тренировочной выборке. Это позволило в одном пакете при подаче на обучения иметь примерно одинаковую пропорцию больных и здоровых пациентов. Такой подход улучшил сходимость при обучении классификационных сетей. Для регрессии данный принцип не применялся, так как на тренировочных и валидационных данных имелись схожие распределения признака фракции выброса, имеющий высокую степень разброса.

5.3 Архитектуры сетей

В проведенном исследовании были использованы две основные архитектуры для анализа данных: собственная четырехслойная сверточная нейронная сеть (CNN) и классическая архитектура ResNet18 [42].

Выбор ResNet18 был обусловлен его широкой популярностью и высоким качеством в решении множества задач машинного обучения в медицинской области [43], [44]. В моем исследовании не рассматривались более глубокие вариации ResNet, так как было ограниченное количество доступных обучающих примеров без возможности дополнительной аугментации чтобы не потерять информативность, так что такое усложнение архитектуры могло бы привести к переобучению.

Архитектура ResNet (Residual Network) примечательна использованием "skip connections" или "residual connections". Эта концепция позволяет передавать входные данные через блоки нейронов и добавлять их к выходным данным блока. Это решает проблему затухания градиентов при обучении

глубоких нейронных сетей, так как градиенты могут "проходить мимо" нейронов с малыми весами. Пример данной архитектуры представлен на рисунке 46.

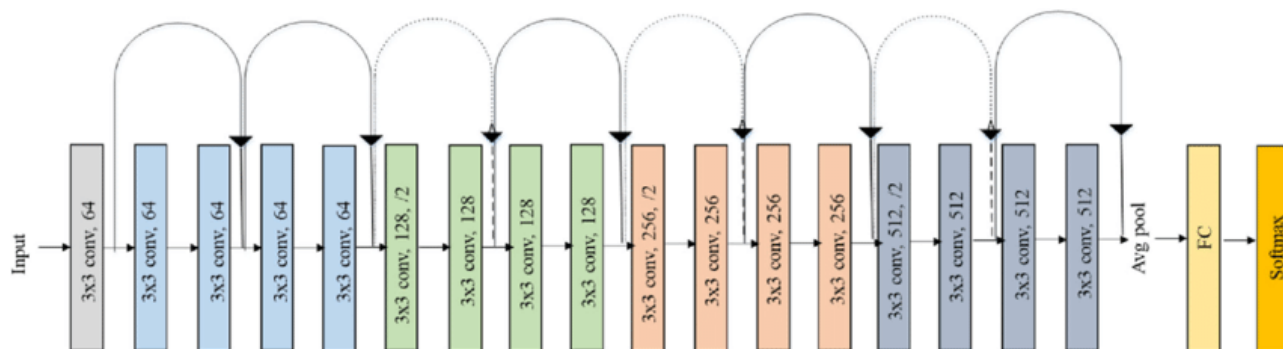


Рисунок 46 – Архитектура сети ResNet18 [42]

Skip connections создают альтернативный путь для градиентов, что упрощает обучение глубоких сетей и позволяет строить модели с сотнями слоев. Эта концепция сделала ResNet одной из наиболее успешных архитектур в задачах компьютерного зрения, где требуется извлечение сложных признаков из изображений.

Для оценки необходимости использования более сложных и глубоких архитектур в решении поставленной задачи была написана собственная неглубокая сеть SimpleNet, на которой тоже проводилось обучение для сравнения результатов инференса с классическим ResNet.

Данная нейронная сеть представляет собой простую сверточную нейронную архитектуру, состоящую из 4 сверточных слоев, 3 слоев пулинга и 3 полносвязных слоев. Архитектура данной сети представлена на рисунке 47.

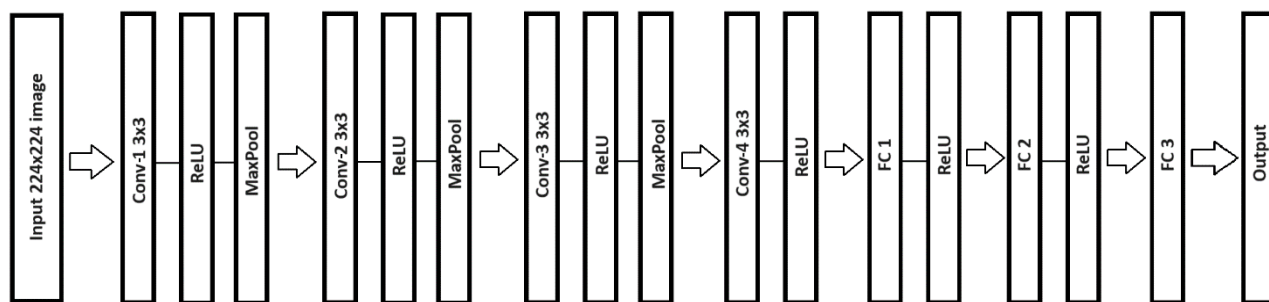


Рисунок 47 – Архитектура сети SimpleNet

Для решения задач были использованы два вида модификаций представленных сетей: отдельно для случая с классификацией и отдельно для регрессии.

В случае бинарной классификации использовались два выходных нейрона и функция потерь кросс-энтропии. Это позволяло модели предсказывать вероятность принадлежности к одному из двух классов.

Для задачи регрессии был использован один выходной нейрон в полносвязном участке сети и функция потерь MSE. Это позволяло модели предсказывать непрерывное числовое значение в зависимости от входных данных.

5.4 Обучение сетей для бинарной классификации

Как было сказано ранее, проводилось исследование двух архитектур нейросетей для данной задачи.

Для каждой из сетей подбиралось множество гиперпараметров обучения с помощью grid search принципа. Среди списка исследованных параметров были размеры пакета обучения (батча), величины стартовых скоростей обучений (learning rate) а также параметры шедулеров скорости обучения.

LR scheduler (от англ. learning rate scheduler) - это метод, который позволяет динамически изменять скорость обучения в процессе обучения модели. Это полезно, когда мы сталкиваемся с проблемой переобучения или недообучения модели.

Когда мы обучаем модель, мы устанавливаем начальную скорость обучения. Однако, по мере того, как модель улучшается, она может достичь локального минимума функции потерь, и дальнейшее уменьшение скорости обучения может помочь ей сойтись к глобальному минимуму.

В режиме со включенным шедулером проводился подбор параметров шага, то есть частоты выполнения операции снижения, а также значения гамма – величины коэффициента снижения при шаге.

Для удобного мониторинга экспериментов был поднят локальный сервер MLflow, в котором проводилось логирование всех параметров модели и оптимизаторов в процессе обучения, а также результатов обучения моделей в виде графиков и матриц ошибок.

Лучшая модель для сохранения определялась как та, что на валидации достигает наивысших значений метрики взвешенного F1-score. Таким образом, в процессе обучения сохранялась версия модели с той эпохи, где результаты были выше всего.

В результате для модели SimpleNet лучшие результаты были достигнуты на 10 эпохе обучения при оптимизаторе Адама с включенным шедулером с коэффициентом 0,5 и размером шага 5 эпох. В данном случае размер обучающего пакета был равен 32 изображениям. Тем самым одна эпоха обучения соответствовала 69 градиентным спускам. Стартовое значение learning rate равнялось 0,001. Графики изменения метрики взвешенного F1-score на валидационном и тренировочном датасетах в процессе обучения представлены на рисунке 48.

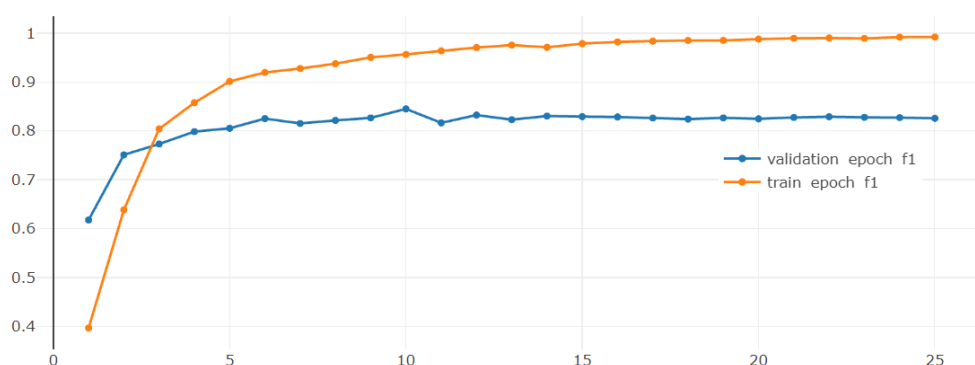


Рисунок 48 – Изменение метрики weighted F1-score при обучении SimpleNet с лучшими гиперпараметрами

Как видно на рисунке 49, значение функции потерь на валидационных данных, начиная с 10 эпохи, начало увеличиваться. Это говорит о том, что сеть начала переобучаться. В итоге лучшие результаты, которых смогла достичь сеть – это 84% точности и weighted F1-score равное 0,84. Значение остальных

метрик представлено в таблице 4, а нормированную матрицу ошибок можно увидеть на рисунке 50.

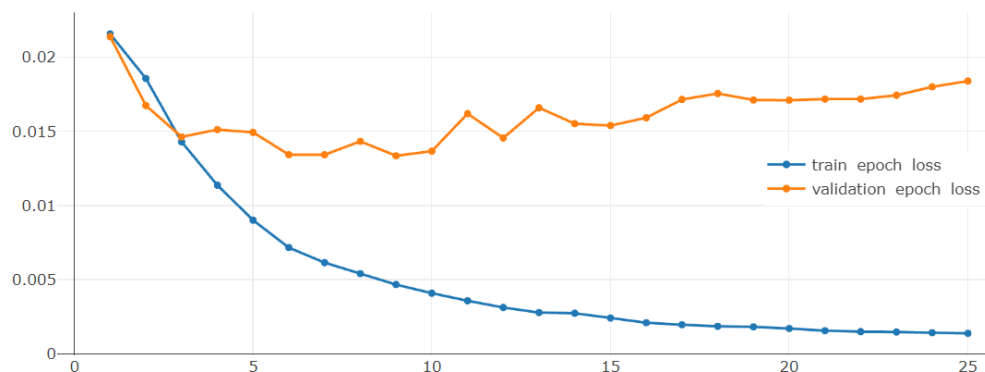


Рисунок 49 – Изменение функции потерь при обучении SimpleNet с лучшими гиперпараметрами

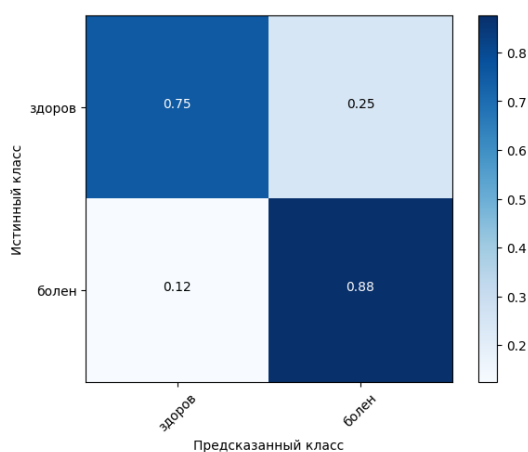


Рисунок 50 – Нормализованная матрица несоответствий SimpleNet

Для модели с архитектурой ResNet18 помимо вышеперечисленных параметров так же рассматривался вариант с обучением при инициализации случайных стартовых весов, а также вариант с дообучением сети (fine tuning). В результате проведенных исследований было установлено, что использование предобученных стартовых весов способно значительно улучшить качество модели при обучении.

Лучший результат обучения для модели ResNet18 был получен при дообучении со стартовым learning rate, равным 0,001, без снижения данной величины в процессе обучения (то есть без шедулера). В процессе обучения

использовался размера пакета (батча) равный 64 изображениям. То есть на одну эпоху обучения приходилось 34 градиентных спусков.

Графики изменения метрики взвешенного F1-score на валидационном и тренировочном датасетах в процессе обучения представлены на рисунке 51.

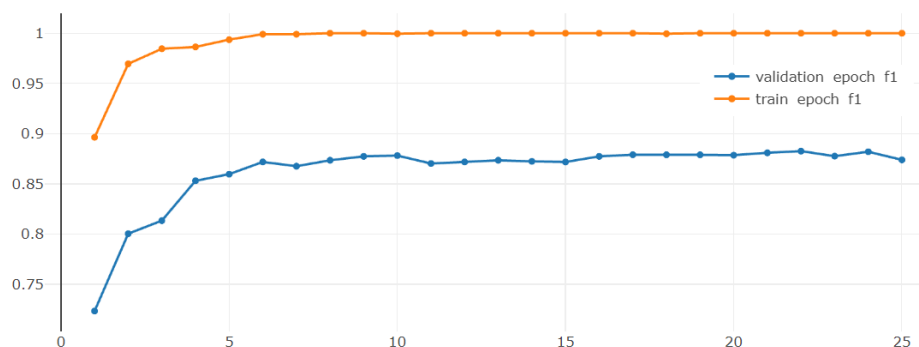


Рисунок 51 – Изменение метрики weighted F1-score при обучении ResNet18 с лучшими гиперпараметрами

Как видно на рисунке 52, значение функции потерь на валидационных данных, начиная с 10 эпохи, начало незначительно увеличиваться. Это говорит о том, что сеть начала переобучаться. В итоге лучшие результаты, которых смогла достичь сеть – это 89% точности и weighted F1-score равное 0,889. Значение остальных метрик представлено в таблице 4, а нормированную матрицу ошибок можно увидеть на рисунке 53.

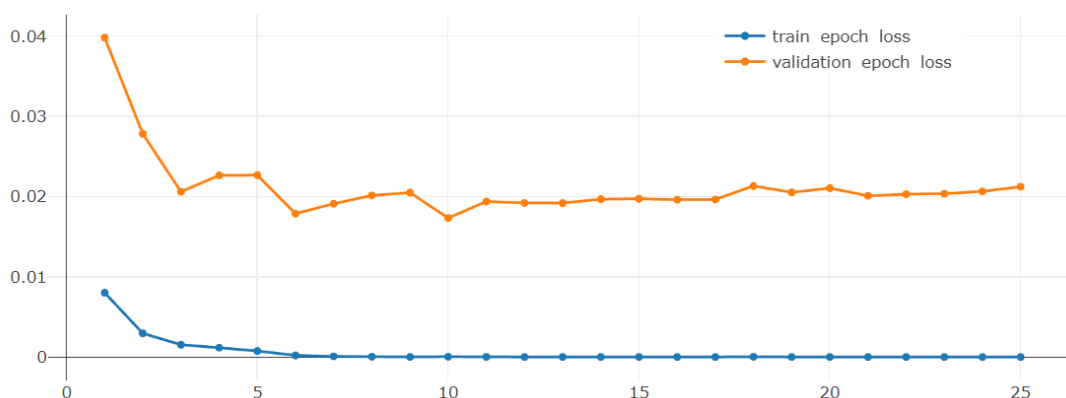


Рисунок 52 – Изменение функции потерь при обучении ResNet18 с лучшими гиперпараметрами

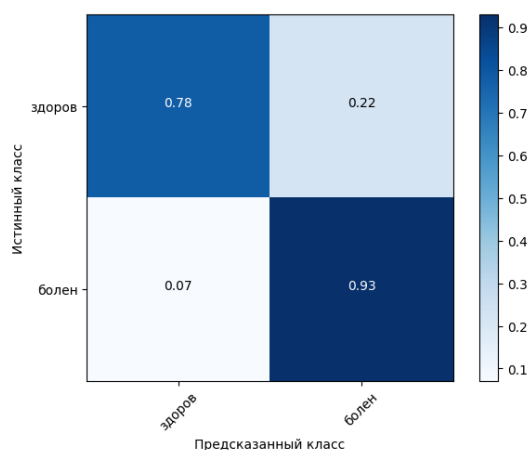


Рисунок 53 – Нормализованная матрица несоответствий ResNet18

Таким образом, в задаче классификации изображений с использованием сверточных нейронных сетей лучшей архитектурой стала ResNet18. При этом использование предобученных стартовых весов позволило добиться на 3% больше точности диагностирования заболеваний сердечно-сосудистой системы.

Таблица 4 – Метрики качества DL моделей на изображениях проекций

	SimpleNet	ResNet18	Fine tuned ResNet18
accuracy	0,84	0,86	0,89
F1-score	0,89	0,91	0,92
Взвешенный F1-score	0,84	0,86	0,89
Precision	0,91	0,90	0,92
Взвешенный precision	0,85	0,86	0,89
Recall	0,88	0,91	0,93
Взвешенный recall	0,84	0,86	0,89

Результаты сетей с лучшими гиперпараметрами каждого типа представлены в таблице 5.

5.5 Обучение сетей для регрессии

Аналогично сетям по классификации были проведены исследования по подбору наилучших гиперпараметров обучения моделей архитектуры SimpleNet и ResNet18. В качестве функции потерь использовалась MSE.

Во время процесса обучения лучшая модель определялась как та, которая достигла наименьшего MSE на валидации к какой-либо из эпох. Для анализа результатов проведенных экспериментов использовался MLflow Tracking API.

Лучшие результаты для архитектуры SimpleNet были достигнуты с включенным шедулером с коэффициентом 0,5 и размером шага 10 эпох при обучении с размером батча равным 32 изображениям. Данная модель достигла средней ошибки на тесте по определению параметра фракции выброса равной 8,6 %. Гистограмма распределения разниц между истинными и пересказанными значениями исследуемой величины представлена на рисунке 54.

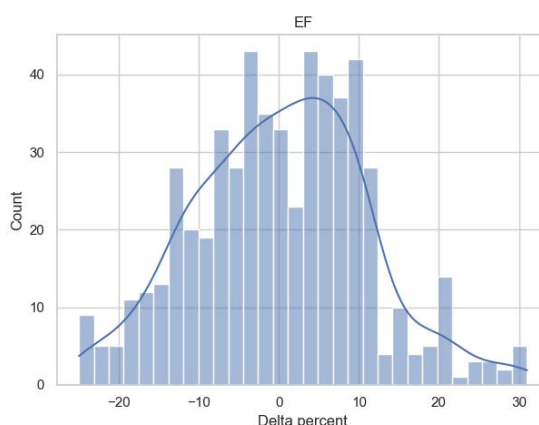


Рисунок 54 – Гистограмма отклонений предсказаний на тестовых данных сети SimpleNet с лучшими гиперпараметрами

Как и в случае с задачей классификации использование технологии fine tuning для обучения сетей с архитектурой ResNet показало лучшее качество. Для моделей с таким типом обучения лучшими гиперпараметрами являются:

использование 64 изображений в тренировочном батче, включенный шедулер с шагом в 15 эпох и коэффициентом 0,5. Такая модель смогла достичь средней ошибки на тесте по определению параметра фракции выброса равной 7,3 %. Гистограмма распределения разниц между истинными и пересказанными значениями исследуемой величины представлена на рисунке 55.

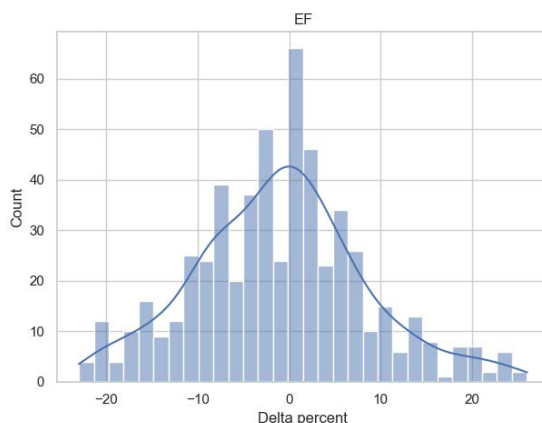


Рисунок 55 – Гистограмма отклонений предсказаний на тестовых данных сети ResNet18 с лучшими гиперпараметрами

Графики изменения метрик MSE и MAE на тестовом и валидационном датасетах в процессе обучения такой модели приставлены на рисунке 56 в логарифмической шкале для большей наглядности.

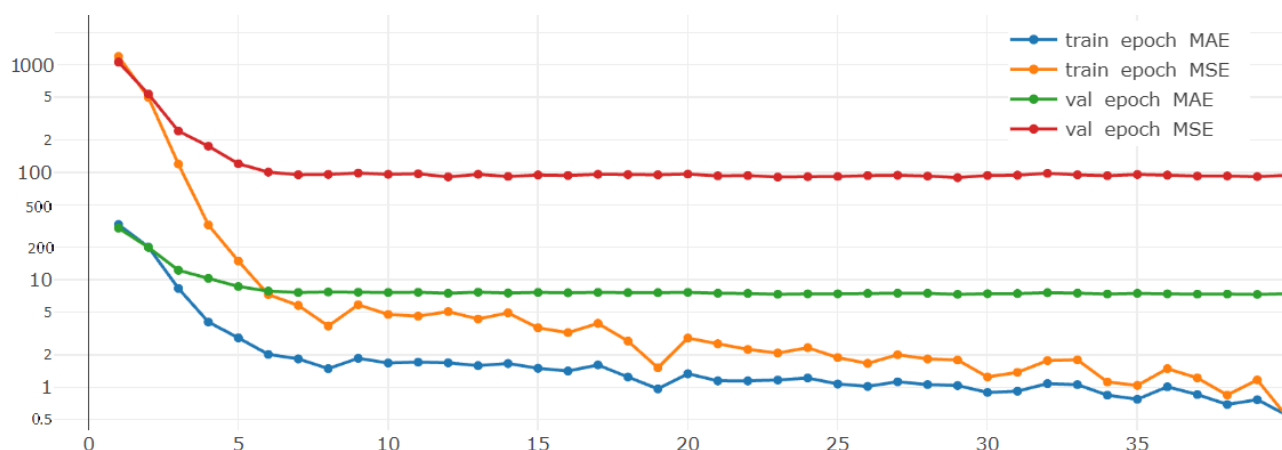


Рисунок 56 – Изменение метрик при обучении ResNet18 с лучшими гиперпараметрами

Результаты метрик по лучшим моделям каждого типа представлены в таблице 5.

Таблица 5 – Метрики качества DL моделей регрессии

	SimpleNet	ResNet18	Fine tuned ResNet18
MAE	0,84	0,86	0,89
MSE	0,89	0,91	0,92

5.6 Интерпретация работы сетей

Интерпретация работы сверточных нейронных сетей в медицинской практике имеет высокую важность, так как врачи и медицинский персонал должны четко понимать, какие признаки и области изображений привели к конкретным диагностическим выводам. Это помогает устанавливать доверие медицинского сообщества к автоматизированным системам анализа медицинских изображений.

В контексте моей исследовательской работы, я применил метод интерпретации, известный как Grad-CAM (Gradient-weighted Class Activation Mapping) [45].

Этот алгоритм вычисляет и визуализирует, какие области изображения считаются наиболее важными для определенного класса. Он основан на вычислении градиентов, относящихся к выбранному классу, и усреднении их по активациям в выбранном сверточном слое сети. Полученная карта важности активаций визуализируется на изображении, что помогает понять, на какие области сеть обращала внимание при классификации [45].

С помощью Grad-CAM я проанализировал признаки, которые обученная сеть-классификатор извлекает из последних и предпоследних сверточных слоев сети, и визуализировал их.

На рисунке 57 представлен пример того, на что обращает сеть внимание при определении данной векторных петель как класса нормы. На рисунке 58 показана карта внимания сети при верной классификации как патология.

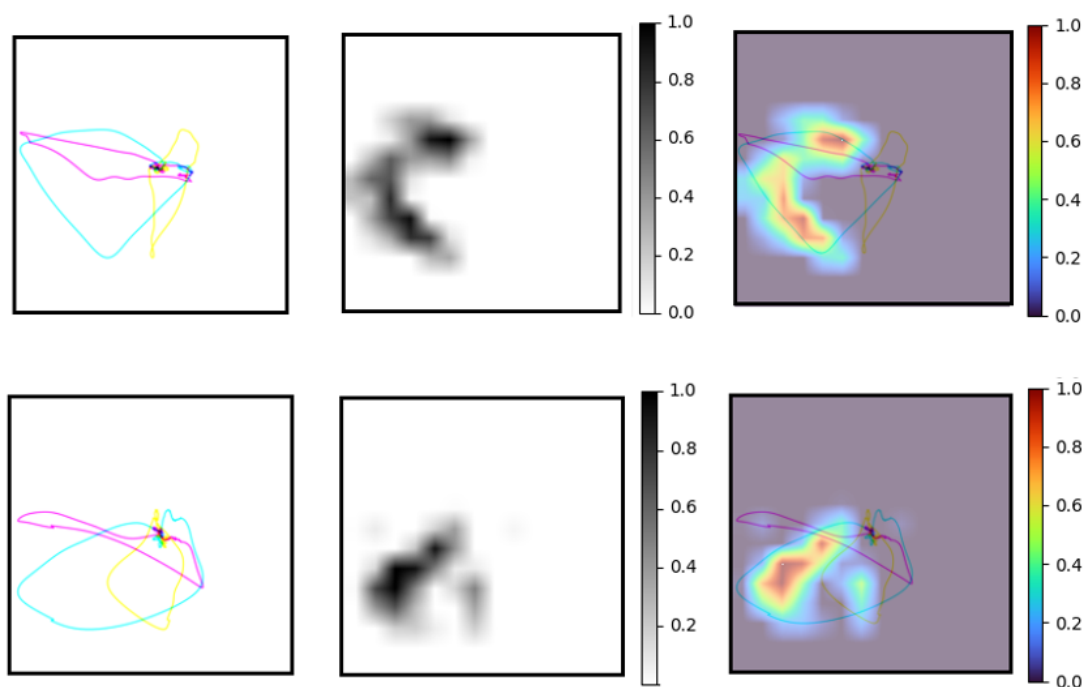


Рисунок 57 – Выделение значимы областей для классификации изображения проекций ВЭКГ как норма с помощью Grad-CAM

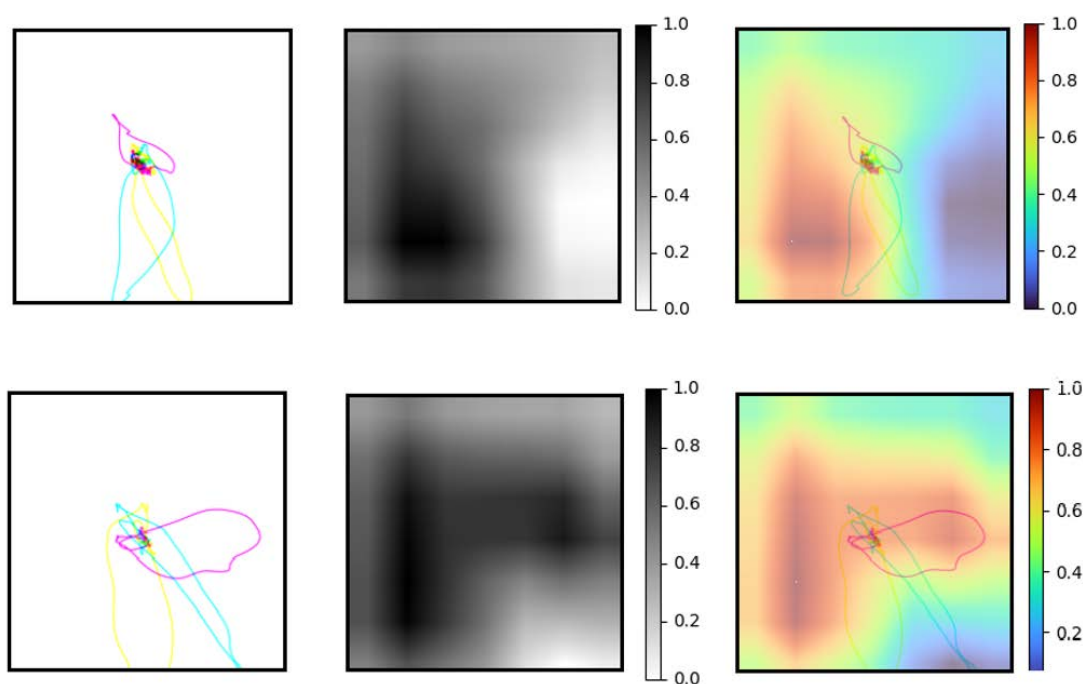


Рисунок 58 – Выделение значимы областей для классификации изображения проекций ВЭКГ как патология с помощью Grad-CAM

Наиболее наглядные области интереса для обнаружения нормы получается обнаружить при рассмотрении результатов с предпоследнего сверточного слоя. Можно обратить внимание, что сеть преимущественно

производит оценку по местоположению петли QRS и не обращает внимание на расположение ST-T областей. Так, для того чтобы определить изображение как проекции в норме, сеть смотрела на область, где преимущественно находится сагиттальная проекция петли.

При анализе патологий использовались результаты выходного последнего сверточного слоя, на котором находятся наиболее обобщенные данные об областях интереса. Можно заметить, что сеть преимущественно смотрит на левую область изображения (то есть рассматривает область классического расположения петель в норме). Можно обратить внимание на то, что все патологические петли QRS преимущественно направлены ниже или правее, чем в варианте нормы. По этой причине сеть и считает область классического расположения нормы петель информативной областью для оценки, попала ли туда текущая имеющаяся петля, а область расположения ST-T проекций опять находится не в самой высокой значимости для Grad-CAM.

6 Машинное обучение на основе облаков точек

6.1 Архитектура PointNet

Для обработки данных, представленных в виде набора трехмерных точек используют специализированные методы. Нейронные сети, которые работают с таким типом данных отличаются от обычных сетей тем, что обрабатывают трехмерные входы, а также должны быть инвариантны к перестановке элементов (порядок точек не должен иметь значения), что требует специфических архитектур и методов предобработки данных [46].

Самой популярной архитектурой, в области классификации облаков точек является PointNet [47]. Она позволяет эффективно обрабатывать неупорядоченные облака точек благодаря встроенным обучаемым преобразованиям данных, имеющимся в архитектуре сети (рисунок 59). За это отвечает элемент под названием T-Net (Transform Network), главная цель которого - автоматически определить матрицу преобразования для обработки входных данных.

Преобразование T-Net включает сверточные и полносвязные слои, операции активации ReLU и нормализации данных. Важно отметить, что T-Net представляет собой, по сути, отдельную нейронную сеть, обучаемую параллельно с основной классификационной архитектурой на базе одноканальных сверток.

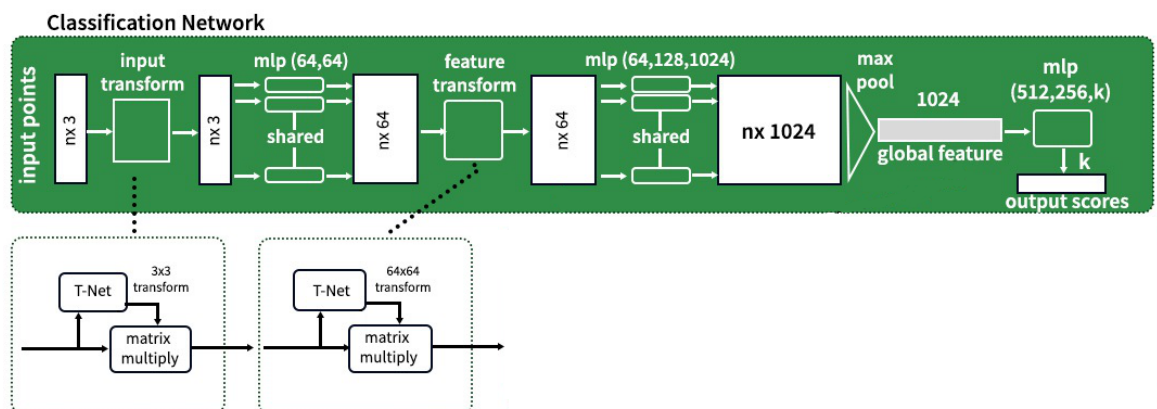


Рисунок 59 – Архитектура PointNet [47]

Итоговое строение PointNet – комбинация классификационного и отдельного блока трансформаций, способного определять, как необходимо

поворачивать, масштабировать и выравнивать входные точки, чтобы модель могла более эффективно извлекать признаки и выполнять классификацию [48].

Для обучения такой сети используют составную функцию потерь, чтобы параллельно обучать классификационную архитектуру, а также встроенную архитектуру трансформаций T-Net [47].

Первой компонентой является NLLLoss (Negative Log Likelihood Loss). Это стандартная функция потерь для задач классификации с несколькими классами. Она вычисляет отрицательный логарифм вероятности правильного класса для каждого образца и усредняет результаты по всем образцам в пакете.

Второй компонентой является LossReg (Дополнительная регуляризационная потеря). Эта часть функции потерь представляет собой норму разности между идентичной матрицей и произведением аффинных трансформаций. Она используется для принудительного ортогонализации и единичности матриц трансформации. Это помогает сохранить структурные свойства данных в процессе обучения.

Сумма этих двух компонентов образует итоговую функцию потерь. При этом значимость потерь не равнозначна, поэтому перед LossReg используется нормируемый коэффициент, равный 0,001, снижающий роль этой ошибки в финальной сумме.

6.2 Переподготовка данных

Для обработки данных ВЭКГ в виде облаков точек стояла задача в выборе метода нормирования данных.

Первым рассмотренным подходом был метод нормирования StandartScaller, который заключается в вычитании среднего значения и деления на среднеквадратическое отклонение [35]. Тем самым итоговые трехмерные данные приобретали среднее значение координат 0 и СКО, равный 1, по всем трем осям.

Вторым подходом был вид нормирования, реализующий вычитание центра масс облака точек и осуществляющий деление на максимально удаленную координату по всем осям. Благодаря этому, итоговое облако обрело среднее значение координат 0 и полностью лежало в сфере с радиусом, равным единице.

Результаты экспериментов показали, что нормирование на максимальную координату более эффективно в предобработке облака точек, ведь оно не искажает исходное распределение и лишь масштабирует ВЭКГ. Благодаря этому, при обучении с идентичными остальными гиперпараметрами, сеть показывала accuracy и weighted F1-score выше более чем на 0.02 пункта.

Для подачи данных на вход нейронной сети важным является то, что размер их должен быть фиксирован. Так как все облака имеют разное число точек, лежащее в диапазоне от 100 до 1000 штук, поэтому необходимо было каким-то образом семплировать одинаковое число образцов.

За фиксированное значение входного числа точек было взято 512 штук, так как это число является степенью числа 2 и ближе всего к среднему значению числа точек в облаках для всего сгенерированного датасета.

В случае, если точек в облаке меньше, чем 512, то происходит случайным образом дополнение датасета уже имеющимися точками, то есть дополнение дубликатами до 512. Если же точек больше, чем 512, то происходит случайное семплирование, то есть берутся случайным образом 512 точек из общего числа имеющихся точек в облаке ВЭКГ.

Исследование показало, что такое случайное семплирование приводило к снижению качества классификации. Было определена причина такого ухудшения. Так как ВЭКГ состоит из нескольких ключевых петель, длительность которых неодинаковая, поэтому для петли ST-T число точек значительно ниже, чем в QRS, так как пик QRS составляет очень мало время в кардиоцикле. По этой причине, при случайном семплировании с большей вероятностью будут выделяться точки ST-T или петли диастолы, нежели соответствующие QRS комплексу.

Эксперименты показали, что области QRS петли являются самыми значимыми для DL моделей, поэтому необходимо было придумать способ проводить неравномерное семплирование, которое с большей вероятностью будет брать точки QRS пика, нежели петли диастолы или ST-T интервала.

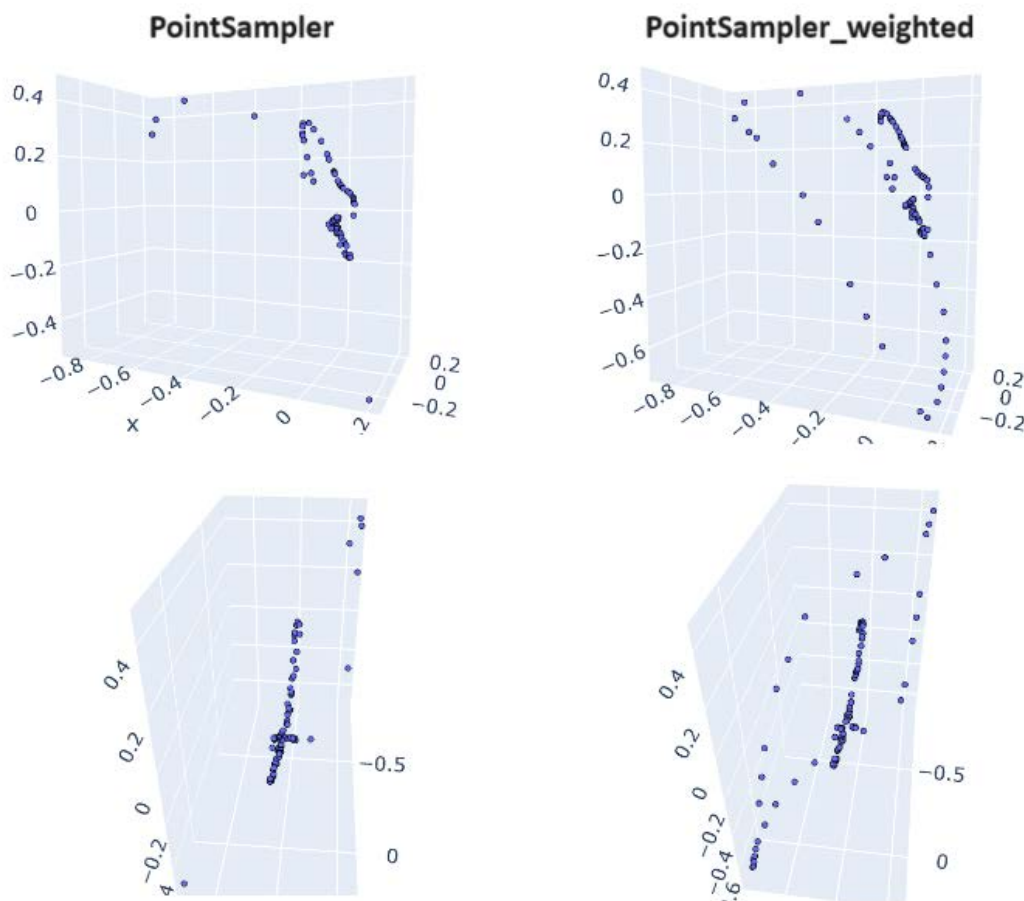


Рисунок 60 – Пример взвешенного и обычного семплирования 100 точек из исходного ВЭКГ

Так как после нормирования центр масс смещается в начало координат, поэтому начало координат находится в области петли диастолы, так как там самое большое число точек. Петля ST-T проходит чуть дальше от начала координат, а точки QRS области расположены на самом удаленном расстоянии от него. По этой причине, за основу веса значимости точки для семплирования было взято расстояние от центра масс ВЭКГ.

Тем сам, была разработана технология случайного взвешенного семплирования, которая с большей вероятностью берет точки наиболее удаленные, которые как раз соответствуют информативным петлям ВЭКГ. На

рисунке 60 показан пример семплирования 100 точек методом случайного и случайного взвешенного семплирования. Как можно увидеть, взвешенный случай даже при таком малом числе точек для вытягивания способен сохранить форму главной QRS петли.

В результате экспериментов было определено, что при схожих гиперпараметрах модель с классическим случайным семплированием 512 точек показывает ассурасу и weighted F1-score ниже более чем на 0.03 пункта по сравнению с взвешенным семплированием такого же числа точек.

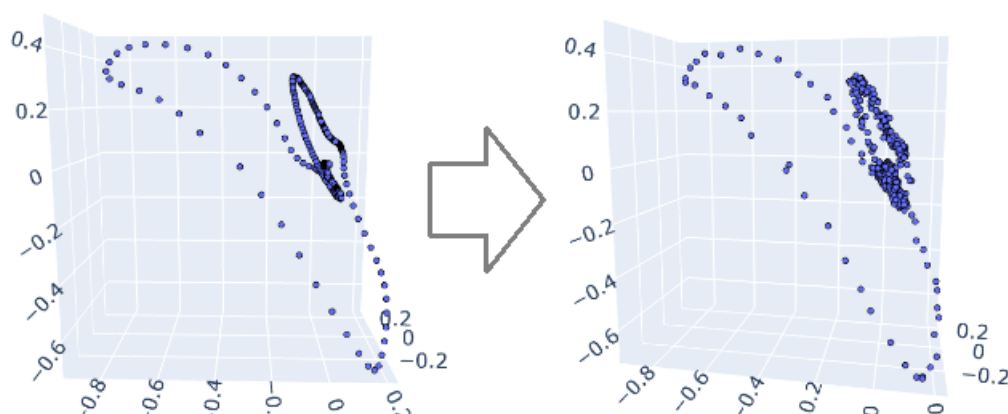


Рисунок 61 – Пример добавления шума на тренировочных данных

Помимо имеющихся природных аугментаций (генерация данных для одного пациента с нескольких периодов сердцебиения) была добавлена возможность случайным образом добавлять нормальный шум по всем осям для исходных точек с заданным значением СКО. Тем самым при обучении в данных можно вносить разнообразие и анализировать его вклад в качество обучения моделей PointNet архитектуры. Пример добавления нормального шума (с нормальным распределением) представлен на рисунке 61.

6.3 Обучение моделей

Как было сказано ранее, при обучении проводилось исследование подходов нормализации и семплирования и добавления шума как вида аугментации. Помимо этого, подбиралось множество гиперпараметров обучения с помощью grid search принципа. Среди списка исследованных

параметров были размеры пакета обучения (батча), величины стартовых скоростей обучений (learning rate) а также параметры шедулеров скорости обучения.

Для удобного мониторинга экспериментов был поднят локальный сервер MLflow, в котором проводилось логирование всех параметров модели и оптимизаторов в процессе обучения, а также результатов обучения моделей в виде графиков и матриц ошибок.

Лучшая модель для сохранения определяясь как та, что на валидации достигает наивысших значений метрики взвешенного F1-score. Таким образом, в процессе обучения сохранялась версия модели с той эпохи, где результаты были выше всего.

Было определено, что добавление люого аддитивного шума на обучающих данных негативно отражается на результатах обучения. Отсутствие шума повышает среднее accuracy и weighted F1-score более чем на 0.01 пункт.

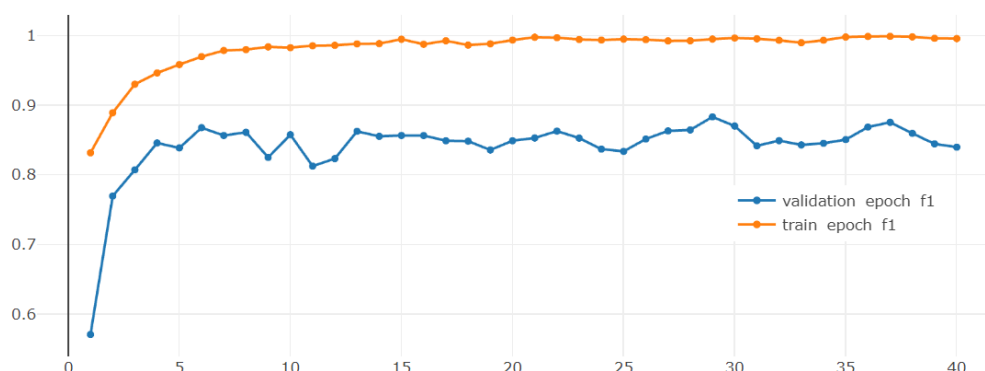


Рисунок 62 – Изменение метрики weighted F1-score при обучении PointNet с лучшими гиперпараметрами

В результате для модели PointNet лучшие результаты были достигнуты на 29 эпохе обучения при оптимизаторе Адама без шедулера. В данном случае размер обучающего пакета был равен 64 изображениям. Тем самым одна эпоха обучения соответствовала 27 градиентным спускам. Стартовое значение learning rate равнялось 0,00025. Для данного обучения использовалась нормализации деления на максимальное значение и полностью отсутствовал аддитивный шум. Графики изменения метрики взвешенного F1-score на

валидационном и тренировочном датасетах в процессе обучения представлены на рисунке 62.

Как видно на рисунке 56, значение функции потерь на валидационных данных, начиная с 25 эпохи, начало незначительно увеличиваться. Это говорит о том, что сеть начала переобучаться. В итоге лучшие результаты, которых смогла достичь сеть – это 89% точности и weighted F1-score равное 0,892.

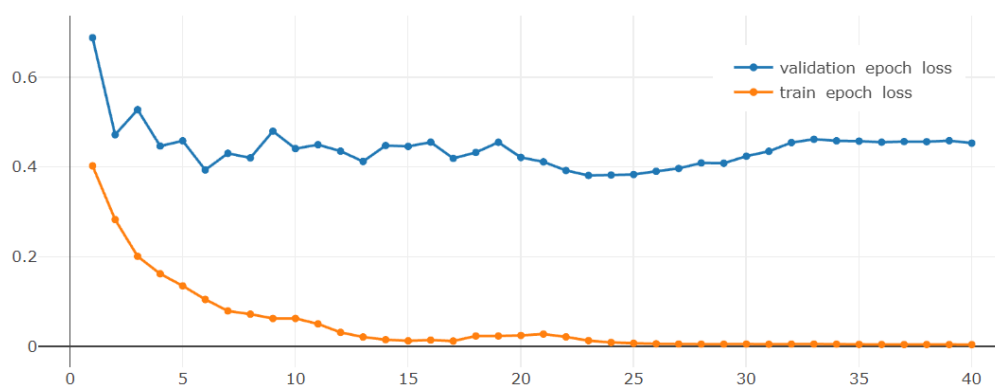


Рисунок 63 – Изменение функции потерь при обучении PointNet с лучшими гиперпараметрами

Значение остальных метрик представлено в таблице 6, а нормированную матрицу ошибок можно увидеть на рисунке 64.

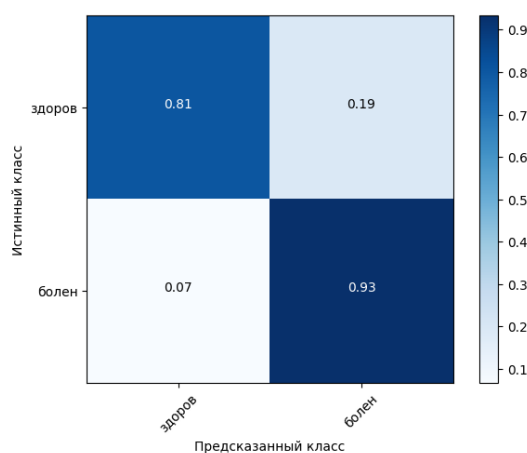


Рисунок 64 – Нормализованная матрица несоответствий PointNet

Таким образом, среди классификационных алгоритмов PointNet показал самые высокие значения качества на тестовых данных. Точность определения составила 89,3%, при этом вероятность назвать больного здоровым составляет лишь 7%.

Таблица 6 – Метрики качества PointNet модели

accuracy	0,89
F1-score	0,92
Взвешенный F1-score	0,89
Precision	0,92
Взвешенный precision	0,89
Recall	0,93
Взвешенный recall	0,89

В отличие от лучшей модели ResNet18 при обработке ВЭКГ как изображение проекций, модель PointNet имеет более высокое качество диагностирования нормы. Вероятность ошибки первого рода (False Positive) для модели PointNet составляет 19% против 22% для ResNet.

Если провести эксперимент и на вход обученной сети PointNet подать данные с семплированием в 50 точек вместо 512, на которых проводилось обучение, то в случае обычного случайного семплирования результат точности предсказания на валидации будет равен 82% и взвешенное F1-score равное 0,81, в то время как взвешенное семплирование дает результат лишь немного хуже чем в случае с 512 точками, равный 88% и 0,88 на взвешенном F1-score. Тем самым можно сделать вывод о том, что самым значимым признаком является наличие QRS петли. Именно по строению этой петли сеть делает итоговое решение о отнесении ВЭКГ к классу нормы или патологии.

ЗАКЛЮЧЕНИЕ

Таким образом, в данной работе представлен поэтапный процесс разработки аппаратно-программного комплекса по определению патологий сердечно-сосудистой системы на основе векторной электрокардиограммы.

Были проведены исследования по выбору подходов по обработке векторной электрокардиограммы для реализации системы поддержки принятия решений на его основе. Среди рассмотренных алгоритмов бинарной классификации архитектура на базе PointNet показала самые высокие значения качества на тестовых данных. Таким образом, представление векторных петель в виде отфильтрованного набора трехмерных точек является наиболее перспективным направлением в машинном обучении в сфере ВЭКГ анализа.

В процессе выполнения выпускной квалификационной работы были разработаны CLI и GUI версии приложения для построения ВЭКГ на основе многоканальных ЭКГ сигналов. В данное приложение были имплементированы СППР, интерактивная визуализация трехмерных векторных петель и электрокардиографических сигналов, а также проекционные изображения ВЭКГ на главные плоскости.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Школьников М. А. и др. Основные тенденции заболеваемости и смертности от сердечно-сосудистых заболеваний детей и подростков в Российской Федерации //Российский вестник перинатологии и педиатрии. – 2008. – Т. 53. – №. 4.
2. Харченко В. И. и др. Смертность от основных болезней системы кровообращения в России (Аналитический обзор официальных данных Госкомстата, Минздрава России, ВОЗ и экспертных оценок по проблеме) //Российский кардиологический журнал. – 2005. – №. 1.
3. Елизарова Н. Ю., Зайнуллин М. И. Сердечно-сосудистые заболевания и высокие технологии //ИННОВАЦИОННОЕ РАЗВИТИЕ НАУКИ И ТЕХНИКИ. – 2020. – С. 15-29.
4. Sidey-Gibbons J. A. M., Sidey-Gibbons C. J. Machine learning in medicine: a practical introduction //BMC medical research methodology. – 2019. – Т. 19. – С. 1-18.
5. Schmidt-Redemann B, Beckmann R, Schaupeter W, Schmidt-Redemann W, Vogt J. [Cardiomyopathy in Duchenne muscular dystrophy. Part 2: serum enzymes, vector-cardiography, and echo-cardiography in 143 patients (author's transl)] Medizinische Klinik. 1978 Nov;73(46):1621-1626.
6. Mincholé A. et al. Machine learning in the electrocardiogram //Journal of electrocardiology. – 2019. – Т. 57. – С. S61-S64.
7. Becking A. G. T., Burger H. C., Van Milaan J. B. A universal vector cardiograph //British Heart Journal. – 1950. – Т. 12. – №. 4. – С. 339.
8. Noordergraaf A. Circulatory system dynamics. – Elsevier, 2012. – Т.
9. Роев И. В., Йокоши Ч., Лютьен-Дреколл Э. Большой атлас по анатомии // М.: ВНЕШСИГМА. – 1997.
10. Сидоров Е. П. Анатомия и физиология человека //ЕП Сидоров.-М. – 1996.
11. Berkaya S. K. et al. A survey on ECG analysis //Biomedical Signal Processing and Control. – 2018. – Т. 43. – С. 216-235.

12. Mamun M. M. R. K. Significance of Features from Biomedical Signals in Heart Health Monitoring //BioMed. – 2022. – Т. 2. – №. 4. – С. 391-408.
13. Hasan M. A., Abbott D. A review of beat-to-beat vectorcardiographic (VCG) parameters for analyzing repolarization variability in ECG signals //Biomedical Engineering/Biomedizinische Technik. – 2016. – Т. 61. – №. 1. – С. 3-17.
14. Vondrak J., Penhaker M. Review of Processing Pathological Vectorcardiographic Records for the Detection of Heart Disease //Frontiers in Physiology. – 2022. – Т. 13. – С. 856590.
15. Vondrak J., Penhaker M. Review of Processing Pathological Vectorcardiographic Records for the Detection of Heart Disease //Frontiers in Physiology. – 2022. – Т. 13. – С. 856590.
16. Vectorcardiographic Lead Systems [Электронный ресурс]. URL: <https://www.bem.fi/book/16/16.htm> (дата обращения: 29.10.2023).
17. Kors J. A. et al. Reconstruction of the Frank vectorcardiogram from standard electrocardiographic leads: diagnostic comparison of different methods //European heart journal. – 1990. – Т. 11. – №. 12. – С. 1083-1092.
18. Dower G. E. A lead synthesizer for the Frank system to simulate the standard 12-lead electrocardiogram //Journal of electrocardiology. – 1968. – Т. 1. – №. 1. – С. 101-116.
19. Edenbrandt L., Pahlm O. Vectorcardiogram synthesized from a 12-lead ECG: superiority of the inverse Dower matrix //Journal of electrocardiology. – 1988. – Т. 21. – №. 4. – С. 361-367.
20. Душина А. Г., Лопина Е. А., Либис Р. А. Особенности хронической сердечной недостаточности в зависимости от фракции выброса левого желудочка //Российский кардиологический журнал. – 2019. – №. 2. – С. 7-11.
21. Simmonds S. J. et al. Cellular and molecular differences between HFpEF and HFrEF: a step ahead in an improved pathological understanding //Cells. – 2020. – Т. 9. – №. 1. – С. 242.

22. Kemp B., Olivan J. European data format 'plus'(EDF+), an EDF alike standard format for the exchange of physiological data //Clinical neurophysiology. – 2003. – T. 114. – №. 9. – C. 1755-1761.
23. Bucher R. Python for control purposes //Scuola Universitaria Professionale della Svizzera Italiana Dipartimento Tecnologie Innovative. – 2019. – T. 102.
24. Langtangen H. P. (ed.). Python scripting for computational science. – Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.
25. Yeh Y. C., Wang W. J. QRS complexes detection for ECG signal: The Difference Operation Method //Computer methods and programs in biomedicine. – 2008. – T. 91. – №. 3. – C. 245-254.
26. Sasikala P., Wahidabanu R. S. D. Robust r peak and qrs detection in electrocardiogram using wavelet transform //International Journal of Advanced Computer Science and Applications. – 2010. – T. 1. – №. 6.
27. Daniel, G., Lissa, G., Medina Redondo, D., Vásquez, L., and Zapata, D. Real-time 3D vectorcardiography: an application for didactic use. Journal of Physics Conference Series, 2007, vol. 90, no. 1. doi:10.1088/1742-6596/90/1/012013.
28. Шмидта Р., Тевса Г. Физиология человека. ТЗ. – 1996.
29. Дисперсия QT, угол QRST, желудочковый градиент. Варианты нормы зубцов ЭКГ – URL: <https://meduniver.com/Medical/cardiology/1458.html> (Дата обращения: 29.07.2023).
30. Seger C. An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing. – 2018.
31. Deo R. C. Machine learning in medicine //Circulation. – 2015. – T. 132. – №. 20. – C. 1920-1930.
32. Jambukia S. H., Dabhi V. K., Prajapati H. B. Classification of ECG signals using machine learning techniques: A survey //2015 International Conference on Advances in Computer Engineering and Applications. – IEEE, 2015. – C. 714-721.

33. Bonaccorso G. Machine learning algorithms. – Packt Publishing Ltd, 2017.
34. Nachar N. et al. The Mann-Whitney U: A test for assessing whether two independent samples come from the same distribution //Tutorials in quantitative Methods for Psychology. – 2008. – Т. 4. – №. 1. – С. 13-20.
35. Starovoitov V. V., Golub Y. I. Data normalization in machine learning //Informatics. – 2021. – Т. 18. – №. 3. – С. 83-96.
36. Naser M. Z., Alavi A. Insights into performance fitness and error metrics for machine learning //arXiv preprint arXiv:2006.00887. – 2020.
37. Berrar D. et al. Bishop C. M., Nasrabadi N. M. Pattern recognition and machine learning. – New York : springer, 2006. – Т. 4. – №. 4. – С. 738. Cross-Validation. – 2019.
38. Hancock J. T., Khoshgoftaar T. M. CatBoost for big data: an interdisciplinary review //Journal of big data. – 2020. – Т. 7. – №. 1. – С. 1-45.
39. Наркевич А. Н. и др. ИНТЕЛЛЕКТУАЛЬНЫЕ МЕТОДЫ АНАЛИЗА ДАННЫХ В БИОМЕДИЦИНСКИХ ИССЛЕДОВАНИЯХ: СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ //Экология человека. – 2021. – №. 5. – С. 53-64.
40. Бредихин А. И. Алгоритмы обучения сверточных нейронных сетей //Вестник Югорского государственного университета. – 2019. – №. 1 (52). – С. 41-54.
41. Яхшибоева Д. Э., Эрметов Э. Я., Яхшибоев Р. Э. Развитие цифровых технологии в медицине //Journal of new century innovations. – 2023. – Т. 20. – №. 1. – С. 100-107.
42. Targ S., Almeida D., Lyman K. Resnet in resnet: Generalizing residual architectures //arXiv preprint arXiv:1603.08029. – 2016.
43. Zhou Q. et al. Transfer learning of the ResNet-18 and DenseNet-121 model used to diagnose intracranial hemorrhage in CT scanning //Current Pharmaceutical Design. – 2022. – Т. 28. – №. 4. – С. 287-295.

44. Paul S., Agarwal S., Das R. Detection of COVID-19 Using ResNet on CT Scan Image //Proceedings of International Conference on Computational Intelligence, Data Science and Cloud Computing: IEM-ICDC 2020. – Springer Singapore, 2021. – C. 289-298.
45. Selvaraju R. R. et al. Grad-cam: Visual explanations from deep networks via gradient-based localization //Proceedings of the IEEE international conference on computer vision. – 2017. – C. 618-626.
46. Zhang Z., Dai Y., Sun J. Deep learning based point cloud registration: an overview //Virtual Reality & Intelligent Hardware. – 2020. – T. 2. – №. 3. – C. 222-246.
47. Qi C. R. et al. Pointnet: Deep learning on point sets for 3d classification and segmentation //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – C. 652-660.
48. Zhang B. et al. Explaining the pointnet: What has been learned inside the pointnet? //CVPR Workshops. – 2019. – C. 71-74.

ПРИЛОЖЕНИЕ А

Фрагмент исходного кода

Листов 15