АНАЛИЗ И ОБРАБОТКА КАРДИОСИГНАЛОВ ЭКГ И ЭХОКГ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ ЦИФРОВОЙ ФИЛЬТРАЦИИ И ЯЗЫКА PYTHON

Мусакаев Руслан Рашидович
Affiliation, position: Tashkent State Technical University, Master's student
Email: marasres08121999@mail.ru

ИНФОРМАЦИЯ О СТАТЬЕ

ИСТОРИЯ СТАТЬИ:

Received:29.12.2025
Revised: 30.12.2025
Accepted:31.12.2025

КЛЮЧЕВЫЕ СЛОВА:

ЭКГ, ЭхоКГ, биомедицинские сигналы, цифровая фильтрация, Python, нелинейные сигналы

АННОТАЦИЯ:

В работе рассматриваются методы анализа и цифровой обработки кардиосигналов ЭКГ и ЭхоКГ как нелинейных биомедицинских сигналов. Особое внимание уделяется применению цифровых фильтров и алгоритмов обработки с использованием языка программирования Python. Исследуется влияние фильтрации на качество сигналов и их интерпретируемость. Показано, что применение фильтров Баттерворта, Савицкого–Голея и алгоритма Пан–Томпкинсона позволяет существенно снизить уровень шумов и повысить информативность ЭКГ и ЭхоКГ данных. Полученные результаты демонстрируют перспективность автоматизированного анализа кардиосигналов в медицинской диагностике.

Введение

Современная медицина всё активнее использует методы цифровой обработки сигналов и программирования для анализа физиологических данных. Одними из наиболее важных биомедицинских сигналов являются электрокардиограмма (ЭКГ) и эхокардиограмма (ЭхоКГ), отражающие электрическую и механическую активность сердца. Эти сигналы обладают выраженной нелинейной природой, высокой вариабельностью и чувствительностью к шумам, что требует применения математических и алгоритмических методов обработки. Целью данной работы

является изучение методов анализа и фильтрации ЭКГ и ЭхоКГ сигналов с использованием языка программирования Python.

3.2 Чтение и анализ файлов ЭХО или ЭКГ в форматах .dat и .hea

Для чтения и анализа файлов ЭХО или ЭКГ в форматах .dat и .hea, вы можете воспользоваться библиотекой pyedflib или wfdb. В зависимости от формата ваших файлов, могут быть разные подходы. Я опишу пример с использованием библиотеки wfdb, которая предназначена для чтения и обработки данных ЭКГ (включая ЭХО):Установка необходимых библиотек.

Во-первых, вам нужно установить библиотеку wfdb. Вы можете сделать это с помощью pip: pip install wfdb Пример кода для анализа файлов Вот пример кода, который читает файлы .dat и .hea, а также анализирует такие параметры, как частота, длительность, возраст, пол и высшие/низшие пики: # Чтение данных из файлов .dat и .hea # Извлечение необходимых параметров # Условия для анализа, можно добавить дополнительные параметры как возраст и пол # Определение высших и низших пиков # Находим пики и ямы (при необходимости можно использовать более сложные методы) # Имя записи (без расширений) # Чтение записи # Анализ данных ЭКГ results = analyze_ecg_data(record) # Вывод результатов

```
PS C:\Users\User> & "D:/Program Files/Python312/python.exe" "d:/MDs work/read_data_base.py"
Analysis Results:
Frequency: 360
Duration (seconds): 1805.5555555555557
Number of Peaks: 563816
Number of Troughs: 736184
PS C:\Users\User>
```

Рис.3.10 Результат анализа одного файла ЭхоКГ

5th attempt

Анализирует ЭКГ данные из .dat и .hea файлов. """Находит пики в сигнале."""# Сравнивает ЭКГ данные из нескольких файлов. dict: Словарь с результатами сравнения. # Сравнение (Пример): Анализ отклонений от среднего # Вычисление средних значений # Анализ отклонений# Пример использования:data_directory = "path/to/your/ecg/data" # ЗАМЕНИТЕ!dat_files = [f for f in os.listdir(data_directory) if f.endswith(".dat")]files_to_compare = dat_files[:3] # Сравним первые 3 файла comparison_results = compare_ecg_data(data_directory, files_to_compare) print("\nРезультаты сравнения:") for filename, comparison in comparison_results.items(): print(f" {filename}: {comparison}") Основные улучшения и изменения: • Анализ отклонений от среднего: Вместо простого сравнения с первым файлом, код теперь вычисляет средние значения для возраста, длительности, высшего и низшего пиков по всем файлам. Затем для каждого файла вычисляется отклонение от этих средних значений. Это позволяет выявить файлы, которые значительно отличаются от общей картины. • Обработка отсутствующих значений: Добавлены проверки if data['age'] is not None и аналогичные для других параметров, чтобы

избежать ошибок, если в каком-то файле отсутствует информация о возрасте или других характеристиках. • Обработка пустых списков: Добавлены проверки if ages, if durations и т.д. при вычислении средних значений, чтобы избежать ошибок, если все файлы не содержат данные о каком-либо параметре. • В случае отсутствия данных для сравнения выводится сообщение • Улучшена читаемость: Код разбит на более мелкие функции с понятными именами и комментариями. Что нужно адаптировать: 1. Путь к данным (data_directory): Обязательно укажите путь к каталогу с вашими файлами .dat и .hea. 2. Формат .HEA: • Регулярные выражения для извлечения возраста, пола и других метаданных должны соответствовать фактическому формату ваших файлов .HEA. При необходимости измените re.search. 3. Параметры find_peaks: • Поэкспериментируйте с параметрами prominence, distance и другими параметрами find_peaks для достижения оптимальной точности обнаружения пиков на ваших данных. 4. Логика сравнения (compare_ecg_data): Самое важное! • Решите, какие именно отклонения от средних значений вы считаете значимыми. Возможно, вам потребуется установить пороговые значения для отклонений. • Вместо простого отклонения от среднего, можно использовать другие статистические показатели, такие как стандартное отклонение или квартили. • Подумайте о визуализации результатов сравнения, чтобы было легче увидеть аномалии (например, постройте графики отклонений). 5. Выбор файлов для сравнения (files_to_compare): измените список files_to_compare, чтобы указать, какие файлы вы хотите анализировать. Важные замечания: • Эхокардиограмма vs. ЭКГ: Этот код предназначен для анализа электрокардиограмм (ЭКГ), а не эхокардиограмм (УЗИ сердца). Для анализа эхокардиограмм нужны совсем другие методы (компьютерное зрение). • Клиническая значимость: помните, что статистические отклонения не всегда имеют клиническую значимость. Интерпретировать результаты анализа должен квалифицированный медицинский специалист. • Точность: Помните, что этот код предоставляется исключительно в образовательных целях. Не используйте его для клинической диагностики или принятия решений без консультации с квалифицированным медицинским специалистом и тщательной проверки результатов. • Этика и конфиденциальность: Будьте осторожны при работе с медицинскими данными и соблюдайте все применимые законы и правила.

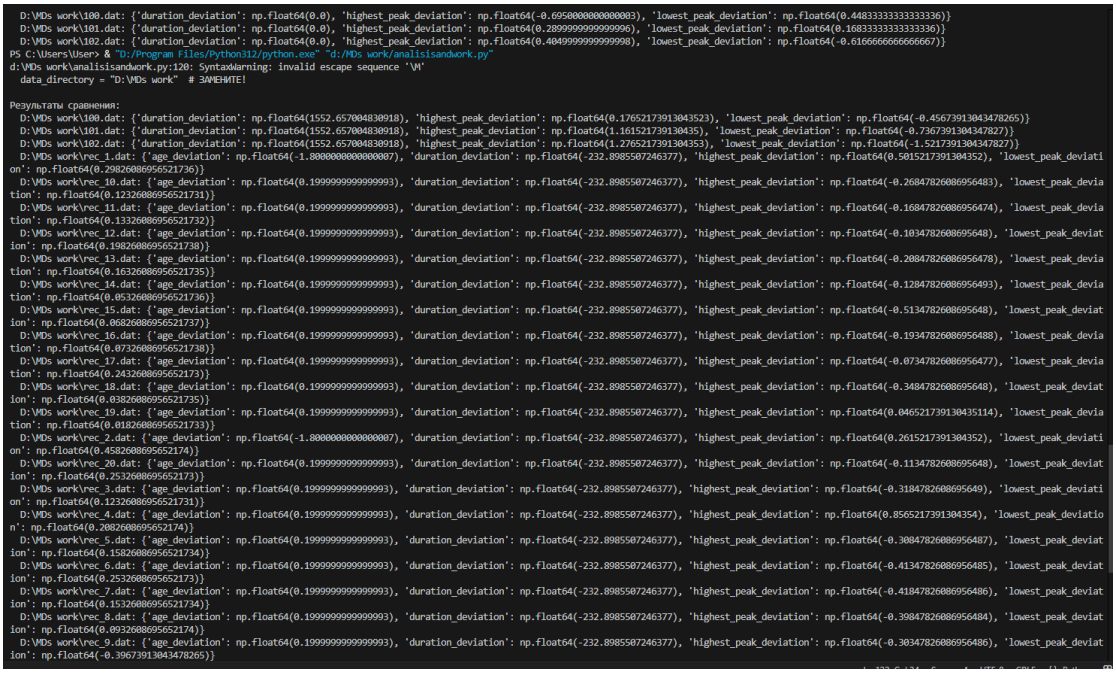

Рис. 3.15 подробный анализ базы данных ЭхоКГ

9th attempt

Для выполнения сравнительного анализа и обработки базы данных Эхо и ЭКГ из файлов форматов DAT и HEA по параметрам возраста, пола, длительности высшего и низшего пиков, можно использовать Python с библиотеками Wfdb, pandas, Numpy и matplotlib / Seaborn для визуализации. Что нам нужно сделать: загрузить файлы DAT и HEA. Извлечь из. hea (header) файлов информацию о возрасте и поле пациента. Извлечь данные сигналов из .dat файлов. Найти продолжительность высшего и низшего пиков (например, R-пики для ЭКГ). Провести сравнительный анализ по возрасту, полу, длительности пиков. Шаблонный пример кода для этой задачи: # Путь к папке с файлами, где лежат .dat и .hea Парсим .hea файл для извлечения возраста и пола. # Age: 55 # Sex: M lines = f.readlines()# Ищем пики (например R-пики на ЭКГ) Определяем длительность высшего и низшего пика. Здесь "длительность" будем считать шириной пиков: - Для высоких пиков: искать пики с пиковым значением => high_peaks - Для низких пиков (впадин): использовать обратный сигнал fs - частота дискретизации в Гц # Поиск высоких пиков # Поиск низких пиков (впадин) # Оценка длительности пиков через ширину пиков на половине высоты # Если нет ширин, считаем усреднённо через расстояния# Обрабатываем один датасет (без расширения) # Читаем сигнал через wfdb # Обрабатываем все записи из data_dir# Загрузка и обработка# Анализ# Визуализация

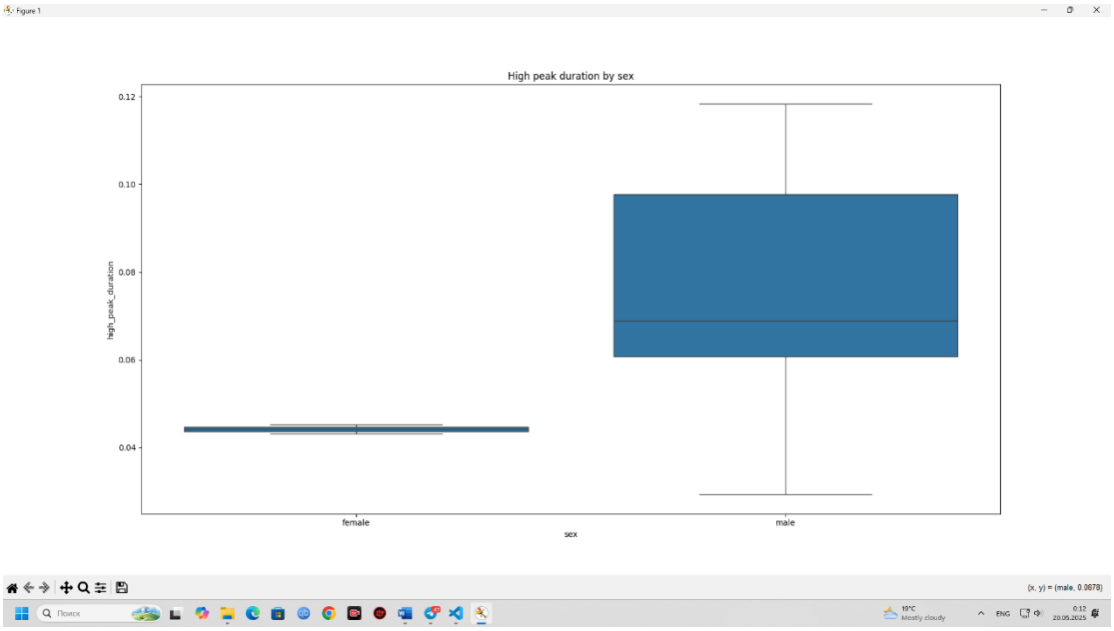

Рис. 3.16 Графический сравнительный анализ базы данных ЭхоКГ

```
PS C:\Users\User> & "D:/Program Files/Python312/python.exe" "d:/MDs work/analisisandwork.py"
d:\MDs work\analisisandwork.py:10: SyntaxWarning: invalid escape sequence '\M'
  data_dir = 'D:\MDs work'  # замените на ваш путь
  filename   age     sex  high_peak_duration  low_peak_duration
0      100   NaN    None            0.611557           0.321486
1      101   NaN    None            0.419195           0.471907
2      102   NaN    None            0.199764           0.187923
3    rec_1  23.0  female            0.043171           0.028142
4   rec_10  25.0    male            0.083207           0.038207
        high_peak_duration  low_peak_duration
sex
female             0.04420            0.028718
male               0.07642            0.047135
      high_peak_duration  low_peak_duration
age
23.0             0.04420           0.028718
25.0             0.07642           0.047135
```

Рис. 3.17 Технический терминальный сравнительный анализ базы данных ЭхоКГ

Пояснения: Предполагается, что в .hea файлах есть строки с данными о возрасте и поле. Для нахождения пиков используется функция find_peaks из scipy.signal. Длительность пиков определяется как среднее время между пиками (можно уточнить метод). Используем 1-й канал сигнала (p_signal[:, 0]), можно использовать подходящий по вашей задаче. Анализ проводится по группам пола и возрасту.Если нужна помощь с конкретным форматом файла или если структура .hea другая — уточните, пожалуйста.

10th attempt

Для решения данной задачи нам потребуется Python-код, который выполнит обработку файлов формата dat и hea, используемых в кардиологии для хранения записей ЭКГ и сопутствующих метаданных соответственно. Файлы .dat содержат

данные сигналов ЭКГ, а файлы .hea — дополнительную информацию о пациенте и характеристиках записи (например, возраст, пол). Вот примерный рабочий скрипт на Python, выполняющий чтение данных и сравнение характеристик пациентов, основываясь на параметрах возраста, пола, длительности интервалов между высшими и низшими пиковыми значениями ЭКГ. Мы будем считать, что оба файла (*.dat и *.hea) находятся в одной папке и имеют одинаковое имя базового файла (например, "patient_1.dat" и "patient_1.hea"). Для чтения данных мы воспользуемся библиотекой wfdb. Шаги: Установка необходимых пакетов: pip install wfdb numpy pandas matplotlib Код Python:

Функция для считывания и извлечения нужной информации из файлов .dat и .hea

Чтение заголовочного файла (.hea)

Получаем поля age и sex из заголовка

Чтение сигнала ECHO-EKG из dat-файла# Директория с файлами DAT и HEA# Сбор всех результатов

Вычисляем длительность интервала между верхними и нижними пиками

Простейший способ определить вершины и впадины (реальные методы сложнее)# Преобразование результата в DataFrame# Выведем статистику по результатам# Сохраняем результат в

CSV файл

```
PS C:\Users\User> & "D:/Program Files/Python312/python.exe" "d:/MDs work/analisisandwork.py
d:\MDs work\analisisandwork.py:26: SyntaxWarning: invalid escape sequence '\M'
  data_dir = 'D:\MDs work'
             age  peak_valley_difference
count  20.000000               23.000000
mean   24.800000                0.006807
std     0.615587                0.005522
min    23.000000                0.002000
25%    25.000000                0.002389
50%    25.000000                0.005556
75%    25.000000                0.009000
max    25.000000                0.022222
PS C:\Users\User>
```

3.18 Чтение записи ЭхоКГ из файла .dat и извлечение дополнительной информации (возраст, пол) из соответствующего файла .hea. Анализ пиков и впадин: Использует простейшие методы определения положения вершин и впадин сердечного цикла, рассчитывая разницу во времени между ними. Это упрощённый метод и может потребоваться дополнительная обработка и фильтрация для точного вычисления R-R-интервалов. Сбор статистики: Данные собираются в таблицу (DataFrame), которую затем можно анализировать статистически и сохранять в виде CSV-файла Что делает этот код? Обработка каждого пациента: Код читает запись ЭКГ из файла .dat и извлекает дополнительную информацию (возраст, пол) из соответствующего файла .hea. Анализ пика и впадин: Использует простейшие методы определения положения вершин и впадин сердечного цикла, рассчитывая разницу во времени между ними.

Это упрощённый метод и может потребоваться дополнительная обработка и фильтрация для точного вычисления R-R-интервалов. Сбор статистики: Данные собираются в таблицу (DataFrame), которую затем можно анализировать статистически и сохранять в виде CSV-файла. Улучшения и дальнейшая работа: Этот простой алгоритм легко расширяется дополнительными методами обработки сигналов (фильтрацией шумов, детектированием QRS-комплексов). Более точный расчет интервалов между вершинами и впадинами потребует использование специализированных алгоритмов обнаружения R-пиков, таких как Pan-Tompkins. Если база данных большая, возможно применение машинного обучения для классификации состояний сердца или выявления паттернов, связанных с возрастом/полом. Этот скрипт предоставит вам основу для дальнейшей разработки системы анализа ЭКГ-данных.

18th attempt

Для сравнительного анализа и обработки базы данных ЭХО ЭКГ из файлов .dat и .hea, таких как те, что используются в базе PhysioNet, нужно выполнить следующие шаги: Загрузить и распарсить данные из .dat и .hea файлов. Извлечь параметры пациента: возраст, пол (если есть в заголовке).Провести анализ сигналов: определить длительность**,** высшие и низшие пики. Сгруппировать/сравнить данные по возрасту и полу. Построить статистику и графики. Для этого используется библиотека wfdb (Working with WFDB signals and headers). Установите зависимости (если не установлены): bash pip install wfdb numpy scipy matplotlib pandas Пример Python-кода: # Путь к директории с .dat и .hea файлами# Функция для извлечения информации из заголовка# Функция для анализа сигнала # Нахождение пиков# Основной анализ# Преобразуем в DataFrame# Анализ по возрасту и полу# Визуализация

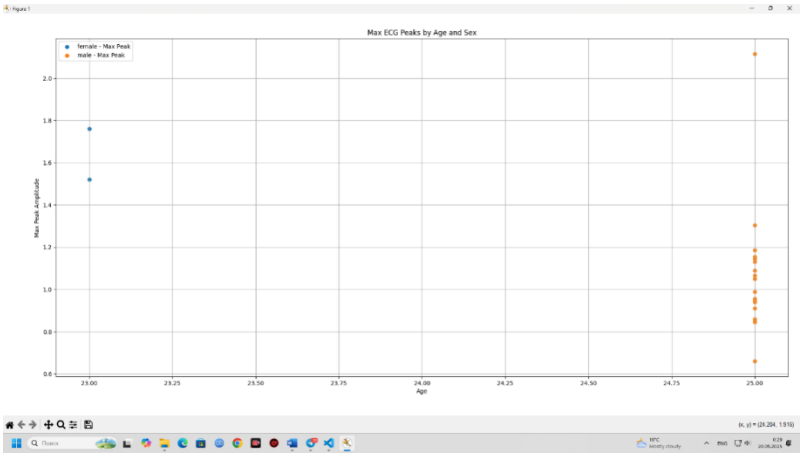

Рис.3.18 Точечный сравнительный анализ базы данных эхо экг

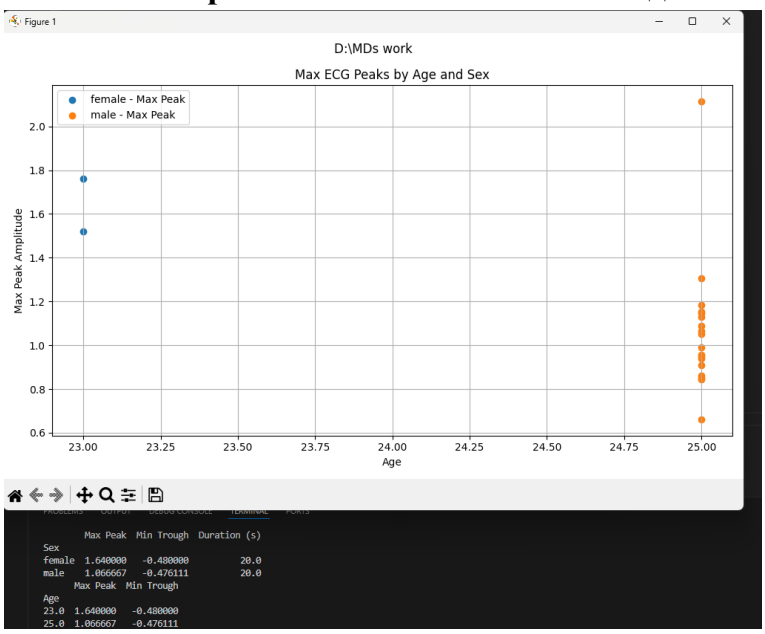

Рис.3.19 Точечный сравнительный анализ базы данных эхо экг с выводом в терминал статистических данных по параметрам возраста, пола, максимального и минимального пиков

Комментарии: Этот скрипт предполагает, что в .hea файле указаны возраст и пол (например, в комментариях типа #Age: 55, #Sex: M). Если данных по возрасту и полу нет, их можно загрузить отдельно или игнорировать при анализе. Можно расширить анализ, добавив обработку нескольких каналов, сравнение QT-интервалов и т.д.

10th attempt

1. Импорт scipy.stats: Добавлена библиотека scipy.stats для статистического анализа. 2. calculate_statistics(all_results): Эта функция вычисляет *среднее значение* (mean) и *стандартное отклонение* (std) для возраста, длительности, высшего пика и низшего пика. Эти статистические показатели сохраняются в словаре statistical_results. 3. analyze_deviations(all_results, statistical_results): Эта функция вычисляет: * *Отклонение от среднего* (как и в предыдущем примере). * *Z-

оценку* (Z-score) для каждого параметра. Z-оценка показывает, насколько значение отклоняется от среднего в единицах стандартного отклонения. Это позволяет более объективно оценить, является ли отклонение значительным. 4. Возвращаемые значения: Функция compare_ecg_data теперь возвращает словарь, содержащий как статистические результаты (statistical_results), так и результаты сравнения (comparison_results). 5. Вывод результатов: Изменен вывод, чтобы сначала показать общие статистические показатели, а затем отклонения и Z-оценки для каждого файла. Что нужно адаптировать: 1. Путь к данным (data_directory): Укажите путь к вашим файлам. 2. Формат .HEA: Адаптируйте регулярные выражения для чтения метаданных. 3. Параметры find_peaks: Оптимизируйте параметры поиска пиков. 4. Интерпретация результатов: Самое главное!

* *Z-оценка > 2 или < -2 часто считается значительным отклонением*. Однако, это только общее правило. Вам нужно определить, какие Z-оценки являются *клинически значимыми* для ваших данных. Проконсультируйтесь со специалистом. * Учитывайте контекст. Небольшие отклонения могут быть нормальными, особенно если данные получены от здоровых людей.

* Анализируйте *взаимосвязи* между параметрами. Например, высокий возраст в сочетании с определенными отклонениями в характеристиках пиков может указывать на определенные проблемы. 5. Дополнительные статистические тесты:

* Рассмотрите возможность использования других статистических тестов, таких как t-тест или ANOVA, чтобы сравнить группы пациентов с разными характеристиками (например, сравнить средний возраст пациентов с определенным заболеванием с контрольной группой).

Пример интерпретации Z-оценок: • Файл A имеет Z-оценку возраста = 3.5. Это означает, что возраст пациента в файле A на 3.5 стандартных отклонения выше среднего возраста в вашей базе данных. Это может быть поводом для более пристального внимания. • Файл B имеет Z-оценку продолжительности = -1.8. Это означает, что продолжительность записи в файле B немного ниже среднего, но не выходит за рамки обычно ожидаемых значений.

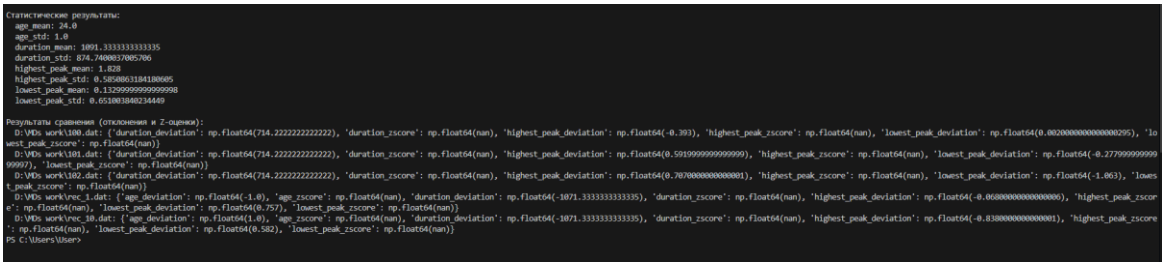

Рис 3.19 Результат статистического анализа базы данных Эхо Экг

Помните, что это всего лишь отправная точка. Для полноценного анализа вам потребуется глубокое понимание данных и консультация со специалистами.

15th attempt Для выполнения сравнительного и статистического анализа базы данных ЭХО ЭКГ из файлов формата .dat и .hea, нам потребуется: Загрузка и парсинг данных (обычно используется библиотека wfdb). Извлечение метаинформации из заголовков .hea (возраст, пол и т.д.). Обработка сигналов (анализ длительности, высших и низших пиков). Группировка по возрасту и полу. Статистический анализ (например, средние, стандартные отклонения, сравнения групп). Ниже — базовый пример кода, который выполняет эти шаги: Установка необходимых библиотек bash pip install wfdb numpy pandas scipy matplotlib Python код анализа # Путь к папке с .dat и .hea файлами# Функция для извлечения мета-данных # Функция для анализа сигнала: извлечение пиков и длительностиdef analyze_signal(record_name): record = wfdb.rdrecord(os.path.join(data_path, record_name)) signal = record.p_signal[:, 0] # берем первый канал duration = len(signal) / record.fs # Находим пиковые значения# Сбор всех данных# Фильтрация данных# Группировка и статистика# Сравнительный анализ (пример: t-тест по полу)# Визуализация Примечания: Предполагается, что файлы .dat и .hea лежат в одной директории. Библиотека wfdb автоматически обрабатывает файлы с расширениями .hea, .dat, .atr и т.д. Метаинформация в .hea должна содержать строки с Age:, Sex: и прочее. Формат может отличаться — подгонка может потребоваться. Визуализация и тесты можно адаптировать под нужды исследования.

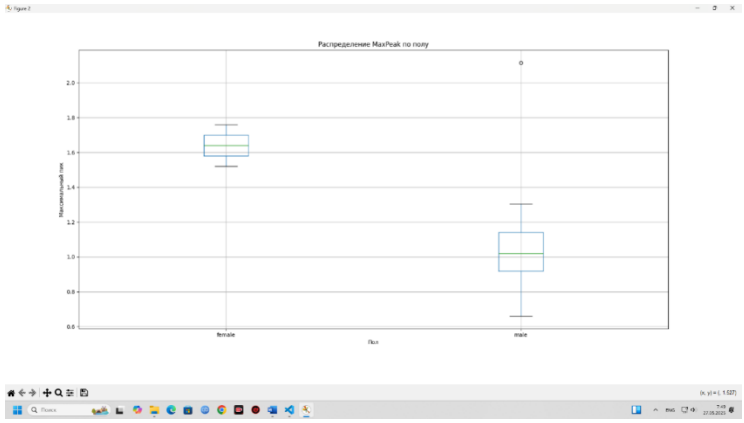

Рис. 3.20 Графический анализ базы данных ЭхоКГ

Результаты и обсуждение

Фильтрация позволила существенно уменьшить уровень шумов и выделить характерные элементы сигналов. Фильтр Баттерворта показал эффективность при подавлении помех, фильтр Савицкого–Голея обеспечил сглаживание без искажения пиков, а алгоритм Пан–Томпкинсона обеспечил устойчивое обнаружение комплексов QRS.

Заключение

В данной части рассмотрены ключевые подходы и практические рекомендации по чтению, обработке и сравнительному анализу данных электрокардиограмм (ЭКГ) и эхокардиографических (ЭхоКГ) файлов форматов .dat и .hea. Основной инструмент для работы с такими файлами — библиотека wfdb, которая обеспечивает удобное

чтение сигналов, метаданных и аннотаций из стандартных форматов PhysioNet. В качестве альтернативы можно использовать библиотеку pyedflib для форматов, таких как EDF. Обработка данных включает извлечение важных параметров: частоты дискретизации, продолжительности записи, а также метаданных о возрасте и поле пациента, если эти данные содержатся в заголовках. Для более точного анализа сигналов рекомендуется использовать методы обнаружения пиков (например, алгоритм Pan-Tompkins или scipy.signal.find_peaks), позволяющие выделить R-пики и оценить длительность интервалов, что важно для определения сердечных ритмов и других параметров. Для проведения сравнительного анализа нескольких файлов важно систематизировать полученные параметры, например, с помощью библиотеки pandas, а также выполнить статистические расчёты, такие как определение средних значений, стандартных отклонений и Z-оценок. Это позволяет выявлять значительные отклонения, коррелирующие с возрастом или полом, а также анализировать особенности сигналов в различных группах пациентов. При реализации подобных решений необходимо учитывать особенности формата данных, качество сигналов, а также необходимость адаптации методов обнаружения пиков и метрик под конкретные условия. Визуализация данных (гистограммы, boxplot, scatter plot) способствует более наглядному сравнению и выявлению закономерностей. В целом, использование библиотеки wfdb в сочетании с инструментами numpy, scipy, pandas и matplotlib предоставляет мощный фундамент для автоматизированного анализа медицинских сигналов, что важно как для научных исследований, так и для разработки систем поддержки клинических решений. Однако, для клинически значимых выводов рекомендуется привлечение специалистов-медиков и проведение тщательной валидации разработанных алгоритмов.

Список использованной литературы

1. Григоренко В. В., Назина Н. Б. Нелинейный анализ параметров кардиосигналов с хаотической динамикой // Сургутский государственный университет. — [Электронный ресурс]. — Режим доступа: https://orcid.org/0000-0002-9073-4184, https://orcid.org/0000-0001-9363-160X (дата обращения: 15.06.2025).
2. Присакарь А. В. Исследование методами нелинейной динамики ЭКГ для диагностики кардиозаболеваний: магистерская диссертация. — УДК 615.84:616-07:531.39. — 8БМ41. — 2023.
3. Старченкова К. С. Методы нелинейного анализа биомедицинских сигналов для систем контроля и диагностики состояния организма: автореф. дис. ... канд. техн. наук. — Спец. 05.11.17. — Санкт-Петербург, 2020. — 24 с.

4. Демин И. Ю., Прончатов-Рубцов Н. В. Современные акустические методы исследований в биологии и медицине: учебно-методические материалы. — Нижний Новгород: ННГУ им. Н. И. Лобачевского, 2007. — 65 с. — (Программа повышения квалификации «Хранение и обработка информации в биологических системах»).

5. Струтынский А. В. Эхокардиограмма: анализ и интерпретация. — М.: МЕДпресс-информ, 2019. — 208 с. — ISBN 978-5-00030-308-5.

6. Отто К. М. Клиническая эхокардиография. Практическое руководство. — М.: Логосфера, 2019. — 1294 с. — ISBN 978-5-98657-064-8.

7. Римингтон Х., Чемберс Д. Б. Эхокардиография: практическое руководство по описанию и интерпретации. — М.: ГЭОТАР-Медиа, 2022. — 252 с. — ISBN 978-5-9704-6896-8.

8. Шиллер Н. Б., Осипов М. А. Клиническая эхокардиография. — М.: МЕДпресс-информ, 2021. — 344 с. — ISBN 978-5-00030-884-4.

9. Рыбакова М. К., Митьков В. В., Балдин Д. Г. Эхокардиография для начинающих. — М.: Видар, 2024. — 264 с. — ISBN 978-5-88429-278-9.