МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра динамического моделирования и биомедицинской инженерии

Выделение последовательности интервалов между сердечными сокращениями методами машинного обучения

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

Студент 2 курса 2281 группы

направления 12.04.04 «Биотехнические системы и технологии»

институт физики

Наумов Артем Александрович

Научный руководитель:
Зав. кафедрой динамического моделирования и биомедицинской инженерии, д.ф.-м.н., доцент — подпись, дата 09.06.2023 — А.С. Караваев

Зав. кафедрой динамического моделирования и биомедицинской инженерии, д.ф.-м.н., доцент — подпись, дата 09.06.2023 — А.С. Караваев

Саратов 2023

Введение

Работа посвящена сопоставлению методов машинного обучения для выделения последовательности интервалов между сердечными сокращениями и разработки программного обеспечения для осуществления этой операции.

По данным всемирной организации здравоохранения и института им. Хопкинса сердечно-сосудистые заболевания лидируют по смертности (порядка 19 миллионов человек) и по инвалидизации во всех странах мира, и считается, что основными способами борьбы с ними являются профилактика, скрининг и ранняя диагностика. Они должны быть основаны на современных инструментальных методах и методах обработки и анализа сигналов, в частности анализ сердечного ритма считается весьма информативным сигналом, который широко используется для построения диагностических, терапевтических методов, при этом существует проблема, связанная с выделением последовательности интервалов между сердечными сокращениями. Есть мнение, что в общем случае эта задача не решается в силу большой нестационарности и сложности формы сигнала, но можно ожидать, что современные технологии машинного обучения могут содействовать в мониторинговых системах более точному выделению, поэтому целью работы являлась

Целый ряд исследований свидетельствует в пользу того, что одним из наиболее эффективных способов преодоления этой проблемы является профилактика и ранняя скрининг диагностика.

Структура работы. Выпускная квалификационная работа состоит из введения, четырех глав, заключения и списка литературы.

Глава 1 «Сердце как элемент системы кровообращения. Представления о строении и функции сердца» содержит основную информацию о строении сердца.

Глава 2 «2 Ритмическое возбуждение сердца. Автоматия и проводимость сердца» содержит основную информацию о некоторых дисфункциях сердца.

Глава 3 «ЭКГ и ее связь с электрофизиологическими процессами в сердце» содержит основную информацию об ЭКГ.

Глава 4 «Машинное обучение» содержит основной ход работы, метод разметки данных, визуализацию данных, а также рассмотренные в ходе работы методы машинного обучения.

В заключении сформулированы основные результаты и выводы.

Основное часть

Перспективным направлением реализации таких подходов является анализ ВСР, который содержит важную информацию о состоянии сердечно-сосудистой системы и элементов вегетативной регуляции организма. При этом выделение сигнала RR — интервалов, особенно в мобильных скрининг системах, является нетривиальной и в общем случае нерешенной задачей(physionet.org). Это обусловлено наличием шумов, артефактов и искажений в экспериментальных записях ЭКГ, изменением характерной формы ЭКГ у пациентов, страдающих различными патологиями в кровообращении. Поэтому целью моей дипломной работы является реализация в виде прикладной программы и тестирование на имеющемся наборе данных метода выделения последовательности RR - интервалов из сигнала ЭКГ методами машинного обучения. Для достижения поставленной цели были решены следующие задачи:

1. Обзор методов выделения RR - интервалов и выбор наиболее.
2. Подготовка наборов экспериментальных данных и их разметка.
3. Реализация методов обучения, анализ наборов данных.
4. Статистический анализ результатов методов, сопоставление рассмотренных подходов.

На данном слайде видим форма электрокардиосигнала, на нем выделяют P – волну, QRS – комплекс (с R – пиком), T – волну, которые соответствуют разным фазам сокращения сердца (работа предсердия, работа желудочков, восстановление). Чаще всего для детекции последовательности интервалов между сердечными сокращениями используют последовательность между R - R интервалами, так как эти пики, как правило, более выражены.

Универсального метода нет для выделения интервалов нет. В последние годы стремительно развивается машинное обучение и его подходы, что является перспективным направлением(physionet.org) в решении данной проблемы.

Моей задачей являлось выделение последовательностей RR - интервалов.

Подготовка данных и их визуализация. Для обучения использовались данные 4 здоровых пациентов длительностью по 10 минут.

На слайде мы видим представление этих данных в табличном виде, и графическом.

	Name	ECG	Markup
0	patient_1	-75	0
1	patient_1	35	0
2	patient_1	125	0
3	patient_1	173	0
4	patient_1	194	0
...	...	...	...
450294	patient_1	-221	0
450295	patient_1	-219	0
450296	patient_1	-219	0
450297	patient_1	-250	0
450298	patient_1	-258	0

450299 rows × 3 columns

Рисунок 1 – Табличное представление данных

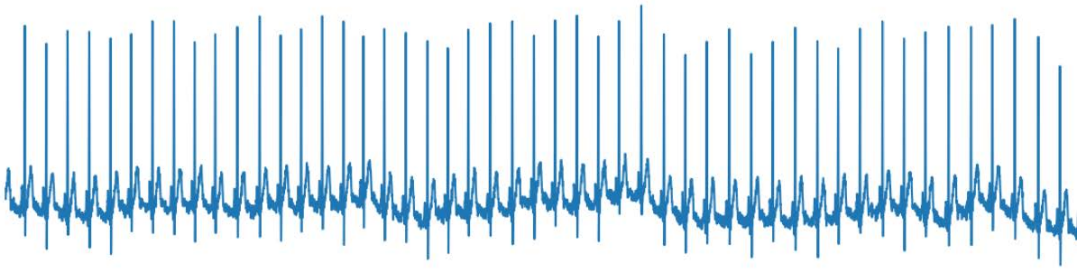

Рисунок 2 – Графическое представление данных

Далее мы добавляем в наше признаковое описание (в нашу таблицу) еще два признака: “Time” (время в мс), “Markup”.

На следующем слайде видим данные после масштабирования (на первой картинке мы видим 3 комплекса, соответственно, 3 R – пика, два RR – интервала).

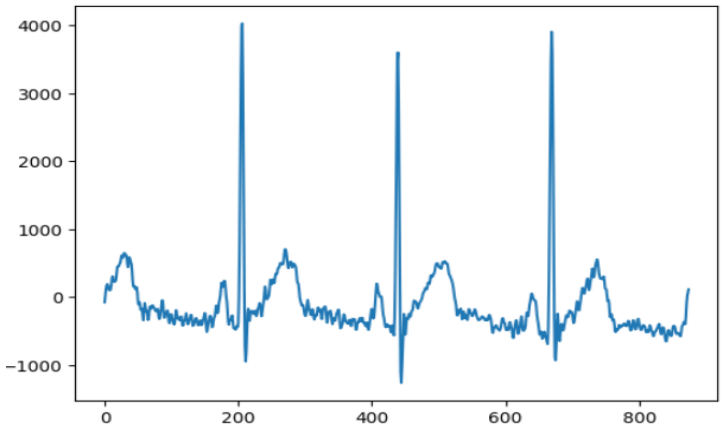

Рисунок 3 – Данные после увеличения масштаба

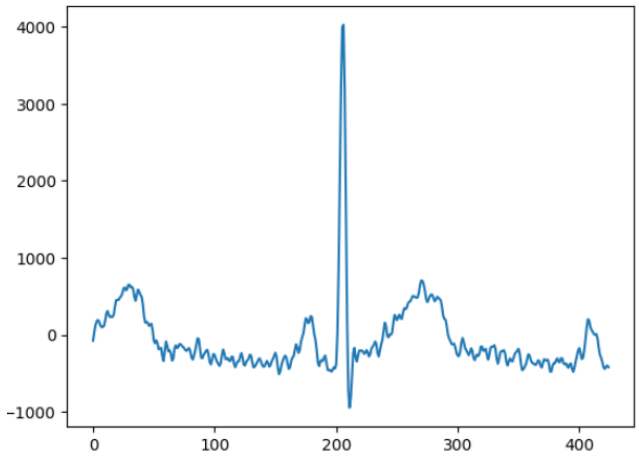

Рисунок 4 – Графическое представление данных

Это масштабирование нам помогает в дальнейшем при ручной разметке.

Мы сопоставляем R – пик со шкалой OX, на которой у нас отложено время в мс.

На следующем слайде мы видим наше признаковое описание пациента после полной разметки данных.

	Name	ECG	Markup	Time
0	patient_1	-75	0	1
1	patient_1	35	0	2
2	patient_1	125	0	3
3	patient_1	173	0	4
4	patient_1	194	0	5
...	...	...	...	...
450294	patient_1	-221	0	450295
450295	patient_1	-219	0	450296
450296	patient_1	-219	0	450297
450297	patient_1	-250	0	450298
450298	patient_1	-258	0	450299

450299 rows × 4 columns

Рисунок 5 – данные после разметки

После этого проводим то же самое с пациентами 2, 3 и 4. И объединяем их в единый набор данных.

Размер каждого первоначального набора данных составляет порядка 450000 значений, количество вручную размеченных R – пиков по каждому пациенту составляет порядка 1800 – 2000 значений.

Данные получились сильно дисбалансными с точки зрения двух классов (есть R – пик / нет R – пика). По графику ЭКГ после масштабирования (там, где один QRS – комплекс), можно примерно понять, что на 250 мс у нас приходится только R - пик.

Далее занимаемся обучением наших моделей. Для этого нам следует разбить наши данные на 3 выборки train/test/validation. На тренировочной мы обучим модель, на тестовой сверим результаты, третья же – та часть выборки, которую наш классификатор не видел в процессе обучения и проверки. Наличие этой выборки позволит нам снизить переобучение (подгонку модели под данные), а также построить графики для визуализации нашего результата.

Размеры выборок составляют третью часть от df. При обучении тестовая выборка также разбивается на несколько частей(фолдов), в нашем случае на 5, для применения метода кроссвалидации (перекрестной проверки). Метод заключается в том, что тестовая выборка разбивается на фолды (допустим,на 5 частей). На четырех из них (1,2,3,4) проводится обучение, на 5 части идет проверка. Потом берутся другие части, например, 1, 2, 3, 5, на четвертой проводится проверка, и так далее. Это также помогает снизить переобучение, с которым нам приходится постоянно бороться при обучении моделей.

Далее реализуем в коде метрики качества, которыми будем проверять наши модели. Accuracy (где TP и TP делится на общее количество объектов) здесь не подходит из-за дисбаланса классов (мы можем просто все 250 мс заполнить нулями, и точность в таком случае будет порядка 0.99, нам это никак не покажет работу нашей модели).

Метрики качества алгоритмов были выбраны следующие:

$$sensitivity = \frac{TP}{TP + FN}$$

$$specificity = \frac{TN}{TN + FP}$$

Чувствительность, специфичность, полнота, точность, F1-оценка, сбалансированная точность, positive predicted value, negative predictive value.

Дерево решений — эффективный инструмент интеллектуального анализа данных и предсказательной аналитики. Он помогает в решении задач по классификации и регрессии. Дерево решений представляет собой иерархическую древовидную структуру, состоящую из правила вида «Если …, то ...». За счет обучающего множества правила генерируются автоматически в процессе обучения.

Бинарное дерево – ациклический граф, в котором есть два типа вершин (вершина соединена с двумя дочерними вершинами или она не соединена ни с одной, тогда она остается листовой вершиной). Таким образом, имеются вершины двух типов: внутренние вершины и листовые вершины.

Так вот, в бинарном дереве определенная дополнительная информация связана с каждой внутренней вершиной и с каждой листовой. Во внутренней вершине у нас располагается некоторый предикат. Предикат способен, взяв объект, сказать, дальше мы его отправим в левую или правую ветвь дерева. В каждой листовой вершине мы будем записывать имя или метку одного из классов, то есть элемент множества y. Вот предикаты, которые будут находиться во внутренних вершинах дерева, мы будем брать из какого-то множества бинарных предикатов. Решающее дерево делит все пространство на непересекающиеся области.

Алгоритм решающих деревьев является вероятностным алгоритмом, то есть он относит объект к тому или иному классу по вероятности, которую он высчитывает на основе данных обучающей выборки. Изначально граница вероятности отнесения объекта к тому или иному классу равна 0.5. Заметил на частотной диаграмме, что основной класс лежит в границе до 0.1. После изменения исходной границы на 0.1 оценки работы данного алгоритма стали существенно выше.

По результатам имеем следующие оценки. Specificity = 0.99, sensitivity = 0.47.

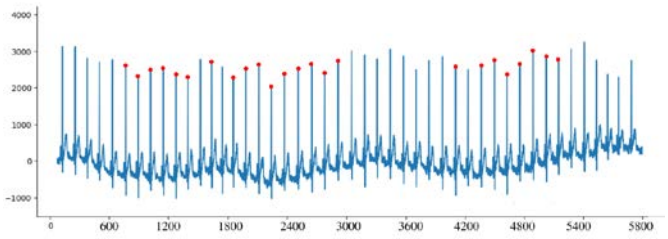

Рисунок 6 – предсказанные значения R – пиков

Метод случайного леса (random forest)— алгоритм машинного обучения, предложенный Лео Брейманом и Адель Катлер, заключающийся в использовании ансамбля решающих деревьев. Алгоритм применяется для задач классификации, регрессии и кластеризации. Основная идея заключается в использовании большого ансамбля решающих деревьев, каждое из которых само по себе даёт очень невысокое качество классификации, но за счёт их большого количества результат получается хорошим. Суть в разбиении выборки на несколько частей (поднаборов, сабсетов), и прогнозировании вероятности у каждого дерева решений по своей части набора данных. В дальнейшем полученные вероятности усредняются.

По результатам имеем следующие оценки. Specificity = 0.99, sensitivity = 0.25.

Скорее всего такой слабый результат является исходом того, что набор данных очень невелик, и при разбиении его на еще более маленькие сабсеты (а разбиение происходит еще и на кроссвалидацию), алгоритм не может вычленить необходимую закономерность. На большей выборке данных алгоритм должен себя проявить куда лучше, и лучше, чем алгоритм решающих деревьев.

Наивный байесовский классификатор — простой вероятностный классификатор, основанный на применении теоремы Байеса со строгими (наивными) предположениями о независимости. В зависимости от точной природы вероятностной модели, наивные байесовские классификаторы могут обучаться очень эффективно. Во многих практических приложениях для оценки параметров для наивных байесовых моделей используют метод максимального правдоподобия; другими словами, можно работать с наивной байесовской моделью, не веря в байесовскую вероятность и не используя байесовские методы. Несмотря на наивный вид и, несомненно, очень упрощенные условия, наивные байесовские классификаторы часто работают намного лучше нейронных сетей во многих сложных жизненных ситуациях.

Достоинством наивного байесовского классификатора является малое количество данных, необходимых для обучения, оценки параметров и классификации.

По результатам имеем следующие оценки. Specificity = 0.98, sensitivity = 0.65.

Логистическая регрессия или логит-модель (logit model) - статистическая модель, используемая для прогнозирования вероятности возникновения некоторого события путём его сравнения с логистической кривой. Эта регрессия выдаёт ответ в виде вероятности бинарного события (1 или 0).

По результатам имеем следующие оценки. Specificity = 0.98, sensitivity = 0.14.

Метод опорных векторов (SVM, support vector machine)— набор схожих алгоритмов обучения с учителем, использующихся для задач классификации и регрессионного анализа. Принадлежит семейству линейных классификаторов. Особым свойством метода опорных векторов является непрерывное уменьшение эмпирической ошибки классификации и увеличение зазора, поэтому метод также известен как метод классификатора с максимальным зазором. Основная идея метода — перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с наибольшим зазором в этом пространстве. Две параллельных гиперплоскости строятся по обеим сторонам гиперплоскости, разделяющей классы. Разделяющей гиперплоскостью будет гиперплоскость, создающая наибольшее расстояние до двух параллельных гиперплоскостей. Алгоритм основан на допущении, что чем больше разница или расстояние между этими параллельными гиперплоскостями, тем меньше будет средняя ошибка классификатора. К недостаткам данного метода относятся можно отнести то, что опорными объектами могут стать выбросы. По результатам имеем следующие оценки. Specificity = 0.99, sensitivity = 0.23.

Таблица 1 – Результаты

	Specificity	sensitivity
Решающие деревья	0.99	0.47
Рандомный лес	0.99	0.25
Наивный байесовский классификатор	0.98	0.65
Логистическая регрессия	0.98	0.14
Метод опорных векторов	0.99	0.23

Заключение

Практически все рассмотренные методы машинного обучения при решении задачи выделения последовательности интервалов между сердечными сокращениями показали довольно слабо из-за довольно небольшого набора данных. Сравнить данные методы можно по длительности обучения модели, а также их точности. Наиболее длительными по обучению моделей являются алгоритмы случайного леса, метод опорных векторов. «Случайный лес», не смотря на его слабый результат может себя показать куда лучше при увеличении объема выборки (впрочем, как и другие алгоритмы, но здесь разница будет куда существенна). Наиболее точным оказался наивный байесовский классификатор, этот результат единственный, который нас устроил на данный момент времени.

Решений данной задачи вообще существует огромное количество, но у каждого есть свои плюсы и минусы (например, для решения задачи с помощью производной возможно большое количество ложных срабатываний).

В ходе дальнейшей работы планируется увеличить выборку данных, разметить данные с точки зрения оставшихся интервалов, сегментов и QRS комплекса для дальнейшего обучения классификатора находить остальные интервалы, чтобы появилась возможность автоматической разметки ЭКГ.

Также в дальнейшем планируется дополнить датасет не только данными от здоровых людей, чтобы по добавленным новым признакам обучить модель выявлять различные дисфункции сердца (особенно полезно это будет для анализа длительных записей ЭКГ (сутки, недели)).

Список использованных источников

1. https://github.com/topics/johns-hopkins-university
2. https://www.mediasphera.ru/issues/zhurnal-nevrologii-i-psikhiatrii-im-s-s-korsakova/2013/8/downloads/ru/031997-72982013810
3. П.Брюс, Э.Брюс, Практическая статистика для специалистов Data Science, (дата обращения: 11.02.2023),
4. Р.М.Баевский, Анализ вариабельности сердечного ритма при использовании различных электрокардографических систем, (дата обращения: 11.03.2023),
5. Байесовский классификатор //ru.wikipedia.org URL: https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%B8%D0%B2%D0%BD%D1%8B%D0%B9_%D0%B1%D0%B0%D0%B9%D0%B5%D1%81%D0%BE%D0%B2%D1%81%D0%BA%D0%B8%D0%B9_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%82%D0%BE%D1%80 (дата обращения: 17.03.2023),
6. К.Элбон, Машинное обучение с использованием Python, с.107, (дата обращения: 11.01.2023),
7. Э.Траск, Глубокое обучение, с. 68, (дата обращения: 11.02.2023),
8. European Heart Journal, Heart rate variability, (дата обращения: 7.02.2023),
9. Хасти Тибширани Фридман, Основы статистического обучения, (дата обращения: 11.02.2023),
10. Д.Грас, Data Science, наука о данных с нуля, (дата обращения: 11.04.2023),
11. Э.Нильсен, Практический анализ временных рядов, с.102, (дата обращения: 11.02.2023),
12. Б.Рамсундар, П.Истман, П.Уолтерс, В.Панде, Глубокое обучение в биологии и медицине, (дата обращения: 11.02.2023),
13. Физиология сердца и его дисфункции, //openedu.ru URL:https://openedu.ru/course/spbu/PHYHEART/?session=spring_2021,
14. И.Трухан, болезни сердечно-сосудистой системы, (дата обращения: 12.05.2023),
15. Оливиа В.Эдейр, Секреты кардиологии, с.57, (дата обращения: 11.02.2023),

16. Сумароков А.В., Клиническая кардиология, с.78, (дата обращения: 11.02.2023),
17. Решающие деревья // ru.wikipedia.org URL https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_%D1%80%D0%B5%D1%88%D0%B5%D0%BD%D0%B8%D0%B9 (дата обращения: 17.03.2023).

09.06.2023 Наумов А.А.