ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В. ЛОМОНОСОВА»

ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ И ИНФОРМАТИКИ

БАКАЛАВРСКАЯ РАБОТА

«Модели классификации слабых акустических сигналов методами машинного обучения»

Выполнил студент
435 группы
Караблинов А. Г.

_____
подпись студента

Научные руководители:
к.т.н., доцент Грачев Е. А.

_____
подпись научного руководителя

к.ф.-м.н., Смирнов А.С.

_____
подпись научного руководителя

Допущена к защите
Зав. кафедрой _____
подпись зав. кафедрой

Москва
2018

Содержание

1 Введение

Развитие информационных технологий в последние десятилетия, связанное с резким скачком вычислительных мощностей, возникновением облачных сервисов хранения данных и высокоскоростными каналами передачи данных поспособствовало масштабному внедрению методов анализа данных, таких как «Машинное Обучение» («Machine Learning»), включая задачу классификации данных.

Целью применения машинного обучения является частичная или полная автоматизация решения сложных профессиональных задач в самых разных областях человеческой деятельности, включая медицину. Венозная тромбоэмболия является одной из самых распространенных причин смерти людей, при этом из них 67% предотвратимы при своевременном диагностировании болезни.

Моделирование состоит из двух этапов. На первом этапе при помощи метода решеточных уравнений Больцмана производится численное моделирование течения кровотока в сосуде. Наличие тромбов разных размеров или их отсутствие приводит к возникновению характерных механических колебаний плотности, представляющих собой акустические волны. Вторым этапом является использование алгоритмов машинного обучения для классификации акустических сигналов.

Разработанная модель классификации слабых акустических сигналов, возникающих внутри кровотока при наличии тромбов (или их отсутствии), позволит осуществлять в автоматическом режиме медицинский экспресс-скрининг нарушений венозной проходимости.

2 Описание задачи

В современном мире развитие технологий дошло до того уровня, при котором они начинают применяться во всех областях человеческой деятельности. Медицина является одной из самых важных и актуальных областей внедрения технологий, потому что многие болезни при своевременном диагностировании возможно вылечить. Данная работа посвящена классификации слабых акустических сигналов, возникающих при протекании кровотока области с тромбом. В результате наличия тромба возникают турбулентные потоки крови, которые можно зафиксировать различными методами, в том числе ультразвуковым допплеровским методом.

Ультразвуковые допплеровские методы являются эффективным средством неинвазивного исследования характеристик движения тканей в организме человека и широко применяются в кардиологии и сосудистой диагностике.

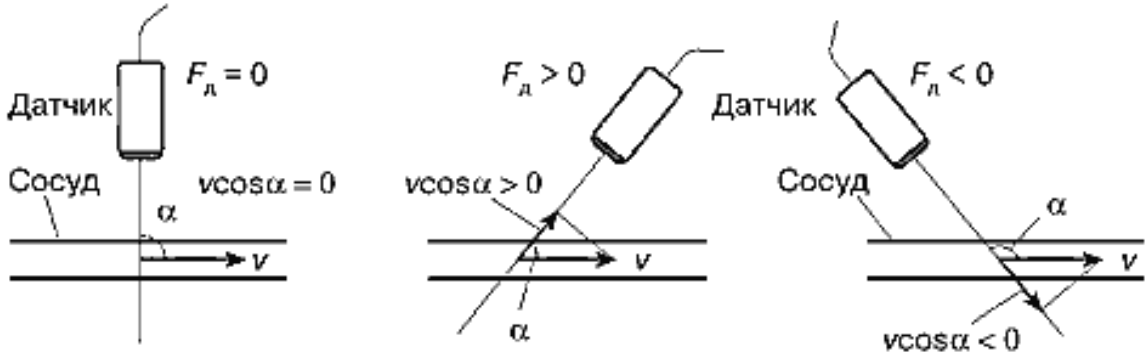

Рисунок 1: Схема расположения датчика в допплеровском методе.

Основой допплеровских методов является эффект Допплера, который состоит в том, что частота колебаний звуковых волн, излучаемых источником (передатчиком) звука, и частота этих же звуковых волн, принимаемых некоторым приемником звука, отличаются если приемник и передатчик движутся друг относительно друга (сближаются или удаляются). Тот же эффект наблюдается, если в приемник поступают сигналы источника звука после отражения движущимся отражателем. Этот последний случай имеет место при отражении ультразвуковых сигналов от движущихся биологических структур (например, клеточных элементов крови). Необходимо отметить, что при угле меньше 60 градусов происходит сильное искажение отраженного сигнала.

Целью работы является построение модели классификации слабых акустических сигналов, полученных из теоретической модели течения кровотока. Для сбора сигналов, была

использована модель на основе решеточных уравнений Больцмана (Lattice Boltzmann Method, LBM). Полученные сигналы обрабатывались, а далее применялись несколько алгоритмов классификации, которые описаны ниже. Но для применения алгоритмов классификации необходимо сначала поставить задачу классификации.

2.1 Постановка задачи

Пусть на множестве объектов X задана функция расстояния $\rho : X \times X \to [0, \infty)$. Существует целевая зависимость $y^* : X \to Y$, значения которой известны только на объектах обучающей выборки $X^l = (x_i, y_i)_{i=1}^l, \quad y_i = y^*(x_i)$. Множество классов Y конечно. Требуется построить алгоритм классификации $a : X \to Y$, аппроксимирующий целевую зависимость $y^*(x)$ на всем множестве X.

Соответственно, всю работу можно разделить на несколько задач:

1. Выбор параметров для теоретической модели, сбор сигналов и их обработка.
2. Построение модели классификации (в частности построения алгоритма a).

3 Обзор алгоритмов классификации

Задача классификации сигналов является одной из классических задач машинного обучения (machine learning, ML). В случае задачи классификации акустических сигналов, наиболее разработанными являются два подхода. Первый из них основан на «классическом ML», то есть сначала из звукового сигнала извлекается описывающий его некоторый набор признаков (средние частоты, дисперсии амплитуд, медианы, квартили, спектральная энтропия, MFCC и т.д.), после чего на подобных признаках обучаются классические классификаторы (например, ближайшие соседи или случайный лес). Подобный подход применяется в [1] для классификации речи по полу человека. Однако, наиболее популярным сегодня методом классификации звука является другой метод, основанный на искусственных нейронных сетях, в частности, сверточных нейронных сетях (convolutional neural networks, CNN). Их успехи обусловлены тем, что они не требуют извлечения какого-либо фиксированного набора признаков из сырых данных, а выучивают свои собственные представления данных в процессе оптимизации. Наиболее разработаны на данный момент сверточные архитектуры для классификации изображений, поэтому стандартным подходом является перевод одномерных аудиосигналов в двумерное time-frequency представление (спектрограммы, мел-спектрограммы) с помощью STFT (Short Time Fourier Transform), после чего обучение сверточного классификатора уже на этих двумерных входных данных.

Далее я расскажу про некоторые наиболее популярные алгоритмы классификации. Если говорить о "классическом" машинном обучении, то это алгоритмы: метод ближайших соседей, решающие деревья и вариант с их ансамблированием - случайный лес. У каждого метода имеются свои достоинства и недостатки, но одним из основных параметром выбора подхода классификации, помимо вычислительных мощностей, является объем выборки, необходимый для того, чтобы не переобучиться. Нейронным сетям требуется гораздо больший объем обучающей выборки, по сравнению с классическими алгоритмами. Но при этом задачу распознавания образов в последние годы решают, чаще всего, при помощи нейронных сетей.

Алгоритмы применяются к заранее собранному набору данных, так называемому датасету, в котором каждый объект имеет некоторое признаковое описание. В задачах классификации эти данные должны быть заранее размечены, то есть каждый объект описывается набором признаков и соответственно ему присваивается своя метка класса. Весь набор данных заранее делится на обучающую подвыборку и на тестовую. Так же иногда имеется абсолютно независимый, контрольный, набор данных для окончательной валидации модели.

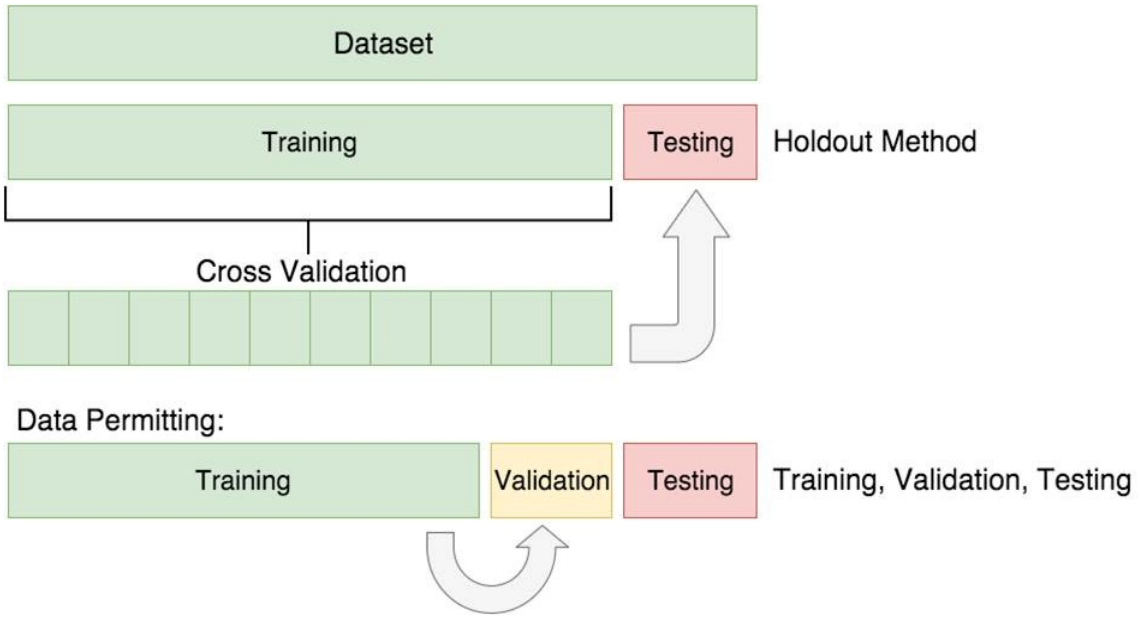

Рисунок 2: Деление набора данных на обучающую и тестовую выборку.

Задача алгоритма, заранее обучившись на обучающей выборке, предсказать класс объекта, основываясь только на его признаковом описании. А для сравнения алгоритмов и выявления наиболее подходящего под конкретные данные, используется тестовая выборка, на которой и проверяется качество работы алгоритма по заранее выбранной метрике. Обучение классификатора состоит в выборе общей формы классифицирующего правила со множеством различных параметров. Далее производится настройка параметров на "обучающем" наборе документов. Ниже приведены методы, используемые в задачах классификации акустических сигналов.

3.1 Метрические методы классификации

Во многих прикладных задачах измерять степень сходства объектов существенно проще, если формировать признаковые описания. Например, такие сложные объекты, как фотографии лиц, временные ряды, первичные структуры белков или акустические сигналы естественнее сравнивать непосредственно с друг с другом путём некоторого "наложения с выравниванием" или на основе выделенных признаков. Если мера сходства объектов введена достаточно удачно, то, как правило, оказывается, что схожим объектам очень часто соответствуют схожие ответы. В задачах классификации это означает, что классы образуют компактно локализованные подмножества.

Для формализации понятия "сходства" вводится функция расстояния в пространстве объектов X. Методы обучения, основанные на анализе сходства объектов, будем называть метрическими, даже если функция расстояния не удовлетворяет всем аксиомам метрики (в частности, аксиоме треугольника).

3.1.1 Метод ближайшего соседа и его дополнения

Для начала запишем постановку задачи классификации.

Пусть на множестве объектов X задана функция расстояния $\rho : X \times X \to [0, \infty)$. Существует целевая зависимость $y^* : X \to Y$, значения которой известны только на объектах обучающей выборки $X^l = (x_i, y_i)_{i=1}^l, \quad y_i = y^*(x_i)$. Множество классов Y конечно. Требуется построить алгоритм классификации $a : X \to Y$, аппроксимирующий целевую зависимость $y^*(x)$ на всем множестве X.

Метрический классификатор. Для произвольного объекта $u \in X$ расположим элементы обучающей выборки $x_1, ...x_l$ в порядке возрастания расстояний до u:

$$\rho(u, x_u^{(1)}) \leq \rho(u, x_u^{(2)}) \leq ... \leq \rho(u, x_u^{(l)}), \tag{1}$$

где через $x_u^{(i)}$ обозначается i-й сосед объекта u. Соответственно, ответ на i-м соседе объекта u есть $y_u^{(i)} = y^*(x_u^{(i)})$. Таким образом, любой объект $u \in X$ порождает свою перенумерацию выборки. Соответственно метрический алгоритм классификации с обучающей выборкой X^l относит объект u к тому классу $y \subset Y$, для которого суммарный вес ближайших обучающих объектов $\Gamma_y(u, X^l)$ максимален:

$$a(u; X^l) = arg \max_{y \in Y} \Gamma_y(u, X^l); \quad \Gamma_y(u, X^l) = \sum_{i=1}^{l} [y_u^{(i)} = y] \omega(i, u); \tag{2}$$

где весовая функция $\omega(i, u)$ оценивает степень важности i-го соседа для классификации объекта u. Функция $\Gamma_y(u, X^l)$ называется оценкой близости объекта u к классу y.

Метрический классификатор определён с точностью до весовой функции $\omega(i, u)$. Обычно она выбирается неотрицательной, не возрастающей по i. Это соответствует гипотезе компактности, согласно которой чем ближе объекты u и $x_u^{(i)}$, тем выше шансы, что они принадлежат одному классу.

Обучающая выборка X^l играет роль параметра алгоритма a. Настройка сводится к запоминанию выборки, и, возможно, оптимизации каких-то параметров весовой функции. Выбирая весовую функцию $\omega(i, u)$, можно получать различные метрические классификаторы.

Алгоритм ближайшего соседа. Алгоритм ближайшего соседа (nearest neighbor, NN.) относит классифицируемый объект $u \in X^l$ к тому классу, которому принадлежит ближайший обучающий объект:

$$\omega(i,u) = [i = 1]; \quad a(u; X^l) = y_u^{(1)}. \tag{3}$$

Этот алгоритм является, по всей видимости, самым простым классификатором. Обучение NN сводится к запоминанию выборки X^l. Единственное достоинство этого алгоритма — простота реализации. Недостатков гораздо больше:

- Неустойчивость к погрешностям. Если среди обучающих объектов есть выброс — объект, находящийся в окружении объектов чужого класса, то не только он сам будет классифицирован неверно, но и те окружающие его объекты, для которых он окажется ближайшим.
- Отсутствие параметров, которые можно было бы настраивать по выборке. Алгоритм полностью зависит от того, насколько удачно выбрана метрика ρ.

В результате получаем низкое качество классификации.

Метод k ближайших соседей. Алгоритм k ближайших соседей (k nearest neighbors, kNN). Чтобы сгладить влияние выбросов, будем относить объект u к тому классу, элементов которого окажется больше среди k ближайших соседей $x_u^{(i)}, i = 1, ..., k$:

$$\omega(i,u) = [i \le k]; \quad a[u; X^l, k) = arg \max_{y \in Y} \sum_{i=1}^{k} [y_u^{(i)} = y]. \tag{4}$$

При $k = 1$ этот алгоритм совпадает с предыдущим, следовательно, неустойчив к шуму. При $k = l$, наоборот, он чрезмерно устойчив и вырождается в константу. Таким образом, крайние значения k нежелательны. На практике оптимальное значение параметра k определяют по критерию скользящего контроля с исключением объектов по одному (leave-one-out, LOO). Для каждого объекта $x_i \in X^l$ проверяется, правильно ли он классифицируется по своим k ближайшим соседям.

$$LOO(k, X^l) = \sum_{i=1}^{l} [a(x_i; X^l \; x_i, k \ne y_i] \to \min_k . \tag{5}$$

Преимущества алгоритма kNN:

- Простая реализация.
- Метод достаточно "грубый" , но отлично подходит для построения базового решения.

- Неплохая интерпретация.

Недостатки простейших метрических алгоритмов типа kNN:

- Приходится хранить обучающую выборку целиком. Это приводит к неэффективному расходу памяти и чрезмерному усложнению решающего правила. При наличии погрешностей (как в исходных данных, так и в модели сходства ρ) это может приводить к понижению точности классификации вблизи границы классов. Имеет смысл отбирать минимальное подмножество эталонных объектов, действительно необходимых для классификации.

- Поиск ближайшего соседа предполагает сравнение классифицируемого объекта со всеми объектами выборки за $O(l)$ операций. Для задач с большими выборками или высокой частотой запросов это может оказаться накладно. Проблема решается с помощью эффективных алгоритмов поиска ближайших соседей, требующих в среднем $O(\ln l)$ операций.

- В простейших случаях метрические алгоритмы имеют крайне бедный набор параметров, что исключает возможность настройки алгоритма по данным

3.2 Случайный лес (Random Forest)

Начать описание алгоритма случайного леса следует с дерева принятия решений, как с основного структурного элемента леса, ведь именно от того, каким образом построено каждое дерево, серьезно зависит качество работы и устойчивость всей финальной композиции.

3.2.1 Решающие деревья (Decision Trees)

Деревья решений одни из самых популярных методов машинного обучения. Идея этого метода состоит в построении решающего дерева на "обучающем" наборе данных. Рассмотрим базовый алгоритм построения решающего дерева. Начнем со всей обучающей выборки X и найдем наилучшее ее разбиение на две части $R_1(j,t) = \{x|x_j < t\}$ и $R_2(j,t) = \{x|x_j \geq t\}$ с точки зрения заранее заданного функционала качества $Q(X,j,t)$. Найдя наилучшие значения j и t, создадим корневую вершину дерева, поставив ей в соответствие предикат $[x_j < t]$. Объекты разобьются на две части - одни попадут в левое поддерево, другие в правое. Для каждой из этих подвыборок рекурсивно повторим процедуру, построив дочерние вершины для корневой, и так далее. В каждой вершине мы проверяем, не выполнилось ли некоторое условие останова - и если выполнилось, то прекращаем рекурсию и объявляем эту вершину листом. Когда дерево построено, каждому листу ставится в соответствие ответ. В случае с классификацией это может быть класс, к которому относится больше всего объектов в листе,

или вектор вероятностей (например, вероятность класса может быть равна доле его объектов в листе).

Функционалы качества. Как я уже указал выше, для построения дерева решений на каждом шаге необходимо выбирать признак и значение порога, по которому происходит оптимальное, по заданному критерию, разбиение. При решении прикладных задач часто используются следующие критерии:

- Чаще всего используется критерий **iGain** :

$$iGain(S) = H(S) - \sum_{v \in \{L,R\}} \frac{|S_v|}{|S|} H(S_v), \tag{6}$$

$$H(S) = -\sum_{c \in C} p_c \log_2(p_c), \tag{7}$$

  где C - множество классов рассматриваемой задачи, а p_c - вероятность класса c для множества S;

- Неопределенность Джинни:

$$G = 1 - \sum_k (p_k)^2 \tag{8}$$

- Ошибка классификации:

$$E = 1 - \max_k p_k \tag{9}$$

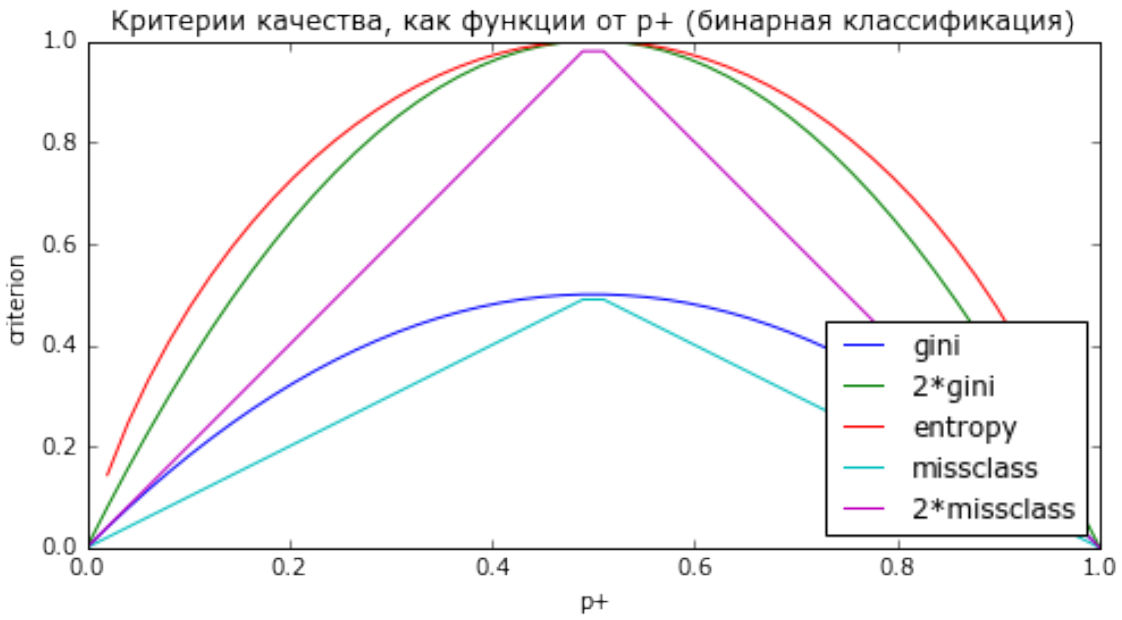

Рисунок 3: Критерий качества как функции от p_+ для задачи бинарной классификации.

Критерии останова. Можно придумать большое количество критериев останова. Перечислим некоторые ограничения и критерии:

- Ограничение максимальной глубины дерева.
- Ограничение минимального числа объектов в листе.
- Ограничение максимального количества листьев в дереве.
- Останов в случае, если все объекты в листе относятся к одному классу.
- Требование, что функционал качества при дроблении улучшался как минимум на s процентов.

С помощью грамотного выбора подобных критериев и их параметров можно существенно повлиять на качество дерева.

Однако в реальных задачах возникают объекты-выбросы, шумы и погрешности, которые своим появлением сильно портят качество классификации одним решающим деревом. Поэтому перед построением каждого нового дерева происходит сэмплирование с повторениями новой выборки $\{(x_i^k, y_i^k)\}_{i=1}^N$ из $\{(x_i, y_i)\}_{i=1}^N$, на которой происходит обучение дерева с номером

k. После построения всех деревьев, каждый тестовый объект z_i получает в качестве промежуточного ответа вектор меток, присвоенных ему каждым деревом, который преобразуется в финальную метку по методу простого голосования.

Преимущества и недостатки случайного леса.

Плюсы:

- Высокая точность предсказаний.
- Достаточно универсальный.
- Практически не чувствителен к выбросам в данных из-за случайного сэмплирования.
- Не требует тщательной настройки параметров.
- Способен эффективно обрабатывать данные с большим числом признаков и классов.
- Одинаково хорошо обрабатывает как непрерывные, так и дискретные признаки.
- Предполагает возможность сбалансировать вес каждого класса на всей выборке, либо на подвыборке каждого дерева.
- Возможность распараллеливания.

Помимо многих преимуществ алгоритм построения случайного леса имеет и ряд недостатков:

Минусы:

- Более сложная интерпретируемость по сравнению с одним деревом.
- Если данные содержат группы коррелированных признаков, имеющих схожую значимость для меток, то предпочтение отдается небольшим группам перед большими.
- Высокие требования к памяти.
- Медленно обрабатываются пропущенные значения.
- По мере приближения к листьям разбиения в узлах каждого дерева становятся все менее статистически обоснованными из-за малого числа рассматриваемых объектов.
- Качество полученной композиции сильно ухудшается в случае, когда на вход алгоритму подается мало размеченных данных, поскольку деревья не могут качественно выявить скрытые в данных закономерности.

4 Описание данных

4.1 Модель сбора сигналов

Для сбора сигналов использовалась модель течения жидкости на основе решеточных уравнений Больцмана. Метод решеточных уравнений Больцмана (Lattice Boltzmann Method, LBM) был разработан как новый инструмент для моделирования течений жидкости, теплопередачи и других сложных физических явлений. По сравнению с традиционными методами вычислительной гидродинамики метод решеточных уравнений Больцмана является методом моделирования на микро- и мезомасштабах, основанным на кинематике частиц. У него много преимуществ, таких как простое программирование, легкая реализация граничных условий и возможность распараллеливания. Метод решеточных уравнений Больцмана моделирует поток ньютоновской жидкости дискретным кинетическим уравнением Больцмана.

$$f_i(\vec{r} + \vec{v}dt, t + dt) = f_i(\vec{r}, t) + \Omega_i, \quad i = 1, 2, ..., 9, \tag{10}$$

где Ω - интеграл столкновений в представлении Батнагара-Гросса-Крука:

$$\Omega_i = \frac{1}{\tau}(f_i - f_i^{eq}), \tag{11}$$

где f - функция распределения плотности вероятности, i - направление движения псевдочастиц, τ - параметр релаксации.

Все вышеперечисленные достоинства этого метода позволили мне сделать выбор в пользу этого метода.

Выбранная модель течения жидкости была доработана с учетом требований поставленной задачи. На начальном этапе было проведено планирование эксперимента, оно позволило минимизировать общее число опытов и выбрать четкую стратегию варьирования параметров. В результате было определено количество и условие проведения опытов, которые были достаточны для решения задачи классификации с установленной точностью. Основные параметры модели:

- Изменяемые параметры:
  - Размер тромба: $0cm \leq \rho \leq 0.49cm$
  - Скорость потока: $8.4\frac{cm}{sec} \leq v_{phys} \leq 11\frac{cm}{sec}$
  - Вязкость: $0.019\frac{cm^2}{sec} \leq \nu \leq 0.027\frac{cm^2}{sec}$
  - Число Рейнольдса: $150 \leq Re \leq 250$

- Неизменяемые параметры
  - Размер сосуда $= 10cm \times 0,5cm$
  - Форма тромба (полусфера)
  - Положение точек записи

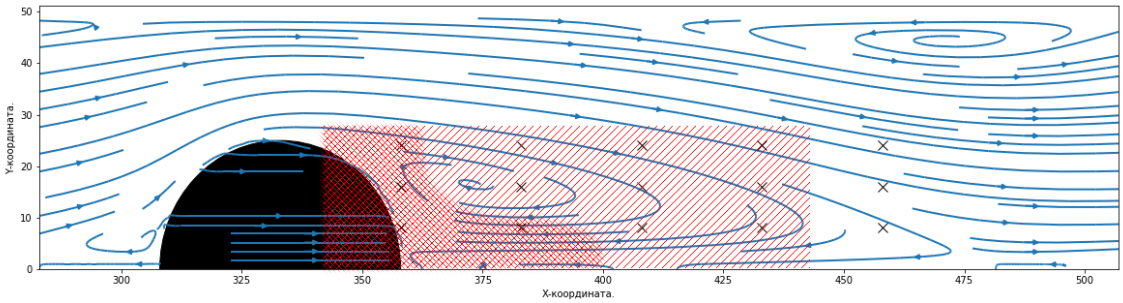

Рисунок 4: Сцена течения жидкости в сосуде с тромбом в 50 % высоты сосуда, получаемая из модели LBM (синие линие - линии тока, красным выделена область завихрения потока, крестами помечены точки записи сигнала)

В работе моделирование течения вязкой жидкости осуществляется в рамках метода решеточных уравнений Больцмана (LBM) с интегралом столкновений в многопараметрической форме с временным шагом вычислений $t = 0,0001$. В различных точках кровотока (количество точек 20) с определенной периодичностью, равной $t_c = 0,001$, собираются значения плотности крови. В результате получается временной ряд, который может быть использован для обучения алгоритма классификации.

Соответственно, для получения обучающей выборки достаточной, чтобы не "переобучиться" и максимально приблизиться к ситуациям в реальной врачебной практике, была доработана модель течения жидкости в сосуде. Выбраны средние размеры моделируемой вены, а также параметры (и пределы, в которых они будут изменяться) для сбора достаточной выборки. В результате было смоделировано 317 разных сцен, с записью в 20 точках, сразу после тромба. При моделировании каждому размеру тромба соответствовала своя метка класса: если тромб по высоте был меньше 30% высоты сосуда, то ему присваивалась метка класса 0, если же больше, то 1.

4.2 Полученные сигналы

В качестве выборки предлагается использовать данные из модели течения крови. Сигналы, смоделированные с помощью модели LBM, записывались в 20 точках, расположенных

сразу после тромба, с периодичностью 0.001 секунды, соответственно каждые 10 итераций моделирования. Всего было было получено 6260 сигналов. Данная модель дает качественную картину кровотока в венах нижних конечностей и позволяет получить данные о механических колебаниях плотности крови и скорости потока в различных точках. Вид сигнала представлен на рисунке 5:

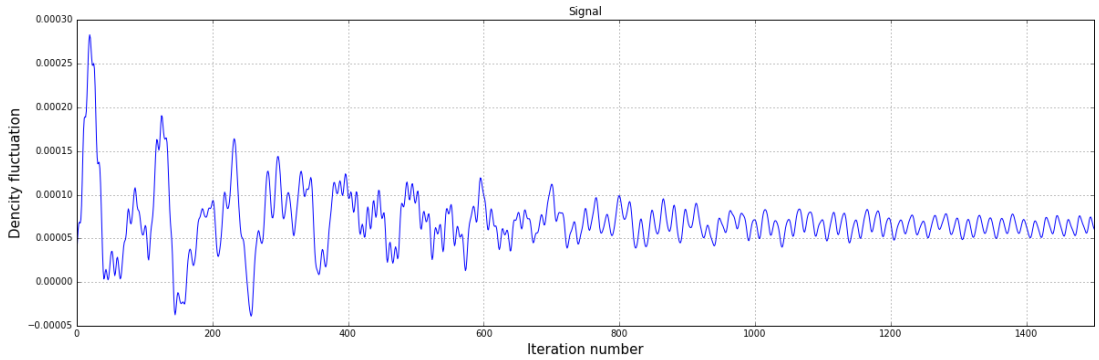

Рисунок 5: Зависимость плотности жидкости от времени в одной из точек.

У исходного сигнала есть ряд существенных недостатков, поэтому над ним производится предобработка в несколько этапов. Сперва от сигнала отсекаются первая секунда измерений, поскольку в это время колебания устанавливаются, а время моделирования ограничено 3 секундами, так как моделирование более долгой сцены (10 секунд) показало неинформативность сигнала, начиная с 3 секунды, вследствие его однообразности и похожести на начало. Далее была вычтена единица из каждой точки сигнала, так как изначальные данные представляют собой колебания в районе единицы, а для использования некоторых функций и удобства работы с сигналом необходимы были колебания около нуля. Далее из полученных сигналов выделялись признаки - создавалось признаковое описание: минимум, максимум, среднее значение, преобладающие частоты, среднеквадратичное отклонение. Энергетический спектр сигнала представлен на рисунке 4:

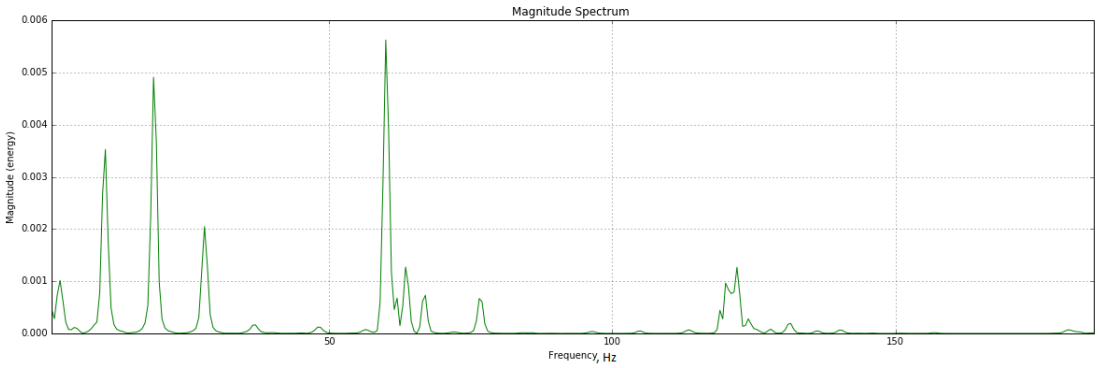

Рисунок 6: Энергетический спектр сигнала, вид которого на рисунке 4.

5 Метрики оценки качества алгоритма

При построении моделей классификации возникает проблема оценки качества модели и сравнения разных алгоритмов. Для этого существуют различные метрики оценки качества алгоритмов. А их анализ - неотъемлемая часть задач классификации.

Существует два подхода к оценке качества алгоритма. Первый - это некая "абсолютная" оценка качества, а второй - сравнение алгоритмов между собой. Вообще метрика выбирается под конкретную задачу, но в то же время есть некий стандартный набор, в котором имеются метрики, подходящие для выбранной задачи. В целом выбор метрики совсем не тривиальная задача, которую, зачастую, даже опытные ученые не всегда сходу могут решить.

Перед тем, как переходить непосредственно к описанию метрик необходимо ввести важную концепцию для их описания в терминах ошибок классификации - confusion matrix (матрица ошибок). В нашей задаче есть два класса, соответственно матрица ошибок для подобной задачи бинарной классификации будет выглядеть так:

		Предсказанный класс	
		True	False
Действительный класс	True	TP	FN
	False	FP	TN

В таблице содержится информация о том, сколько раз алгоритм принял верное и неверное решение по тому или иному сигналу. А именно:

- TP - истинно-положительное решение
- FP - ложно-положительное решение
- TN - истинно-отрицательное решение
- FN - ложно-отрицательное решение

FP, FN являются ошибками первого и второго рода соответственно.

На основе вышеперечисленных понятий производится подсчет метрик, которые позволяют не только оценить качество алгоритма, но и сравнить алгоритмы между собой. В данной работе были использованы следующие метрики, которые, как уже было сказано, являются распространенными и часто используемыми в задачах классификации:

Полная точность или **аккураность (accuracy)**:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \tag{12}$$

- Доля верно классифицируемых объектов по отношению ко всем ответам классификатора.

Эта метрика является самой интуитивно понятной, но при этом имеет ряд недостатков, самый весомый из которых - это бесполезность в задачах с неравными классами.

Точность (precision):

$$Precision\ = \frac{TP}{TP + FP}. \tag{13}$$

- Precision можно интерпретировать как долю объектов, названных классификатором положительными и при этом действительно являющимися положительными. Другими словами, количество положительных документов среди всех документов, которые классификатор считает положительными.

$$Precision\ = \frac{TN}{TN + FN}. \tag{14}$$

- Для негативного класса - доля объектов, действительно принадлежащих нулевому классу от общего числа примеров, классифицированных верно.

Полнота (recall):

$$Recall = \frac{TP}{TP + FN}. \tag{15}$$

- Recall показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел алгоритм.

$$Recall = \frac{TN}{FP + TN}. \tag{16}$$

- Для отрицательных примеров - доля правильно классифицированных негативных примеров от общего числа негативных примеров. Другими словами, это доля действительно негативных объектов из всех документов, распознанных как негативные.

Существует несколько различных способов объединить precision и recall в агрегированный критерий качества. **F-мера (F-measure)** (в общем случае F_β — среднее гармоническое precision и recall) :

$$F = (1 + \beta^2)\frac{Precision \times Recall}{(\beta^2 \cdot Precision) + Recall}. \tag{17}$$

- Мера, комбинирующая точность и полноту. В зависимости от коэффициента $0 \leq \beta \leq 1$ можно отдать предпочтение одной из метрик. При $\beta = 1$ она представляет собой сбалансированную F - меру с множителем 2. F-мера достигает максимума при полноте и точности, равными единице, и близка к нулю, если один из аргументов близок к нулю.

В нашей задаче, связанной с медициной, мы будем отдавать предпочтение recall, соответственно будем минимизировать количество ошибок второго рода, пытаясь сохранить precision.

6 Результаты классификации

Перед построением классифицирующей модели необходимо выделить обучающую и контрольную выборку из имеющихся у нас данных. В дальнейших результатах на обучающей выборке будет проведена кросс-валидация (КВ), а для проверки качества на независимой тестовой выборке алгоритм будет обучаться на обучающей выборке без КВ.

6.1 Метод ближайших соседей

Основным параметром для подбора в методе ближайших соседей является число соседей.

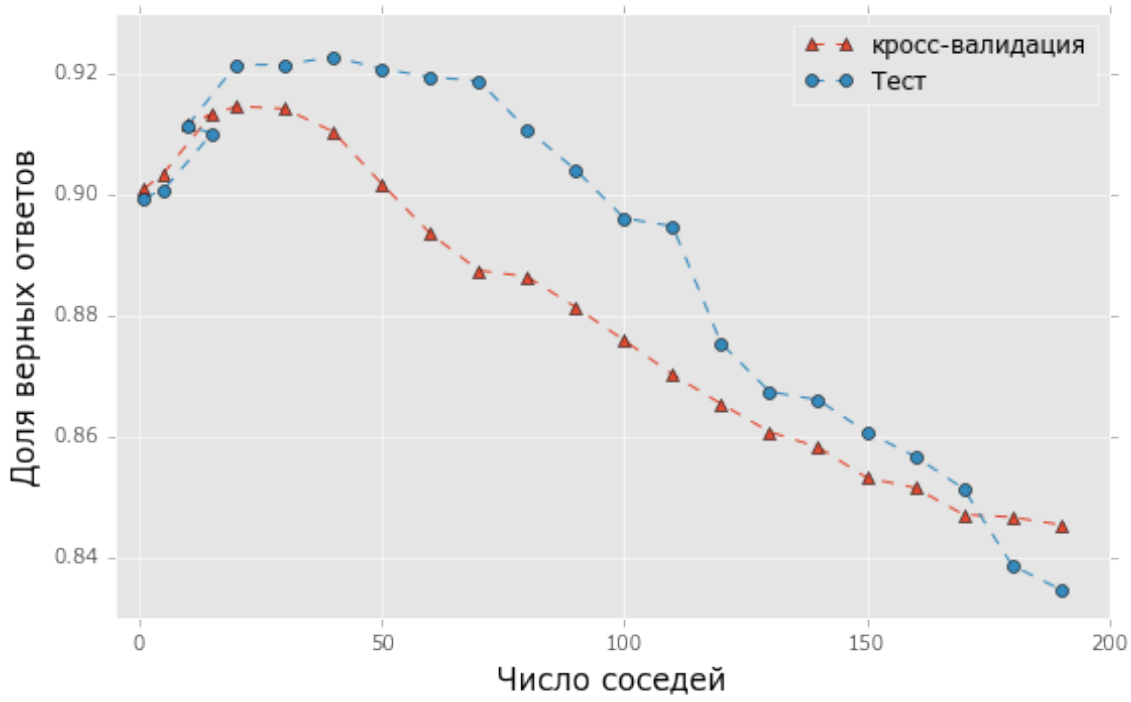

Рисунок 7: Доля верных ответов алгоритма ближайших соседей в зависимости от числа соседей.

Максимальная доля правильных ответов алгоритма достигается при количестве соседей $k = 45$.

Результаты классификации для алгоритма kNN с числом соседей 45:

- Accuracy = 0.922
- Precision = 0.949

- Recall = 0.937
- F-measure = 0.943

6.2 Случайный лес

В алгоритме классификации на основе случайного леса немного больше параметров для подбора, за счет которых можно немного поднять качество классификации.

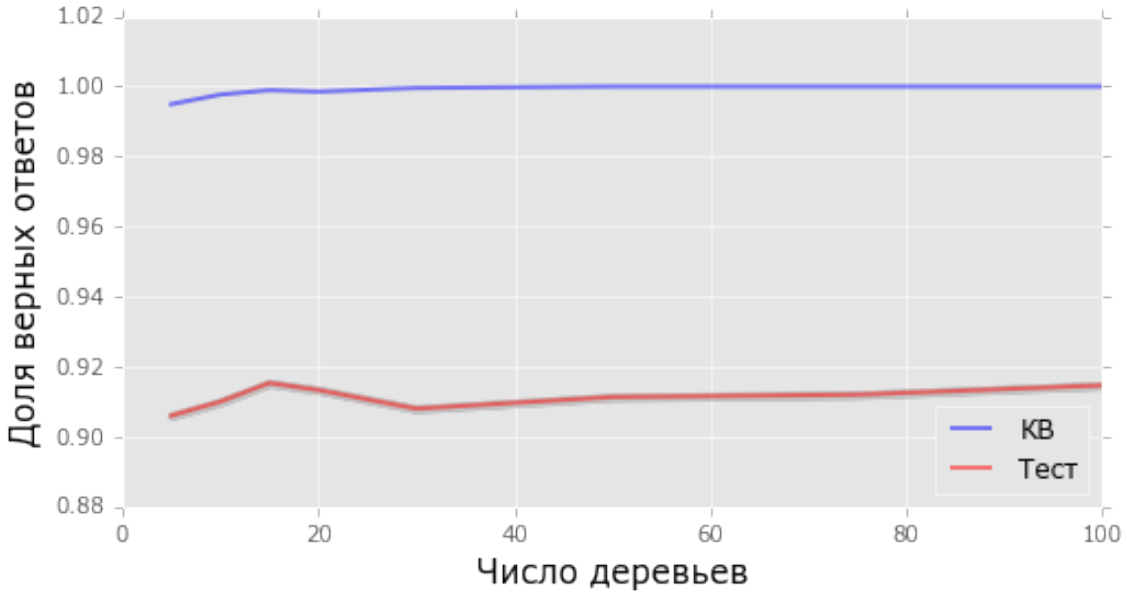

Рисунок 8: Доля верных ответов алгоритма на основе случайного леса в зависимости от числа деревьев.

Как видно, при достижении определенного числа деревьев наша доля верных ответов на тесте выходит на асимптоту, и мы можем сами решить, сколько деревьев оптимально для вашей задачи. На рисунке также видно, что на тренировочной выборке мы смогли достичь 100% точности, это говорит нам о переобучении нашей модели. Чтобы избежать переобучения, мы должны добавить параметры регуляризации в модель.

Начнем с подбора параметра максимальной глубины деревьев - max_depth. При этом фиксируем число деревьев 100.

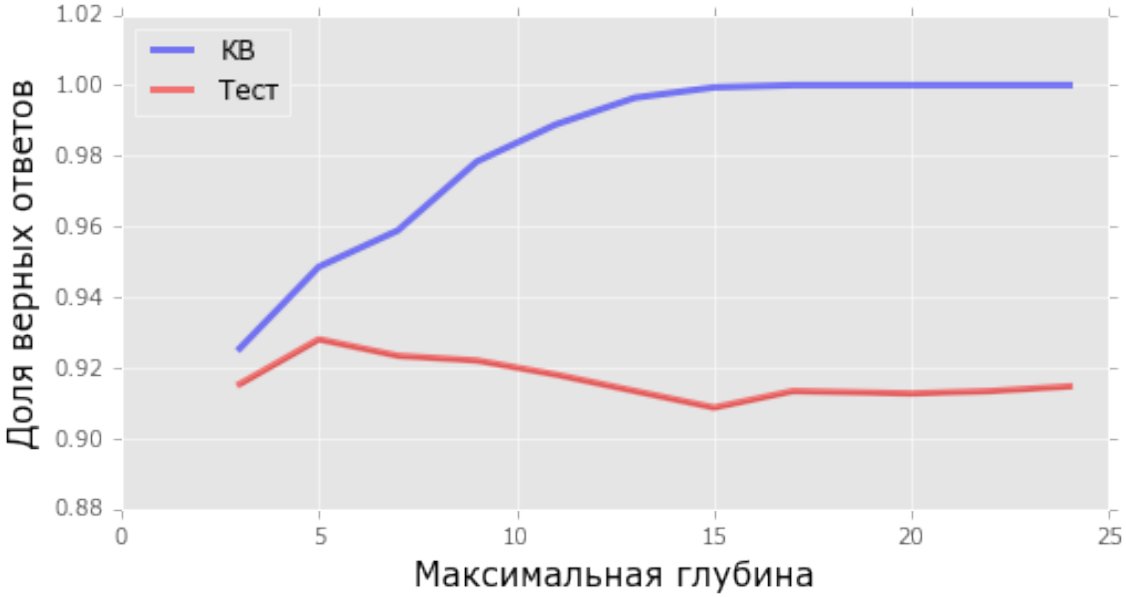

Рисунок 9: Доля верных ответов алгоритма на основе случайного леса в зависимости от максимальной глубины деревьев.

Параметр max_depth немного поднимает долю верных ответов, и мы уже не так сильно переобучаемся.

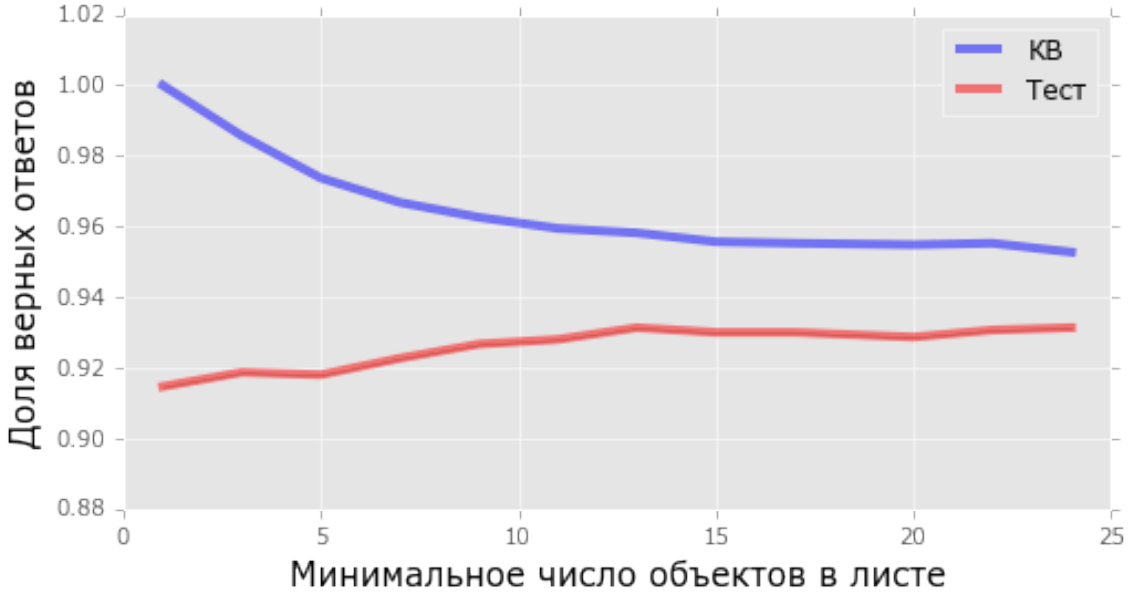

Рисунок 10: Доля верных ответов алгоритма на основе случайного леса в зависимости от минимального числа объектов в листьях.

В данном случае мы еще немного выигрываем в точности на валидации, и при этом сильно уменьшаем переобучение до 2% при увеличении точности до 93%.

Максимальная доля правильных ответов алгоритма достигается при числе деревьев $n\_estimators = 100$ и минимальным числом объектов в листьях $min\_samples\_leaf = 13$.

Результаты классификации для алгоритма Random Forest с числом деревьев 100 и минимальным числом объектов в листьях 13:

- Accuracy = 0.931
- Precision = 0.945
- Recall = 0.96
- F-measure = 0.952

7 Заключение

В настоящей дипломной работе:

- Проведено исследование проблематики возникновения тромба и настроена теоретическая модель течения жидкости для сбора слабых акустических сигналов.
- Предложены и исследованы модели классификации на основе методов машинного обучения, таких как метод ближайших соседей и метод, основанный на случайном лесе.
- Программно реализованы все поставленные эксперименты сбора данных и методы классификации полученных слабых акустических сигналов.
- Итоговый алгоритм показал приемлемый результат работы, следовательно данная работа доказывает свою актуальность и практическую применимость, и является серьезным шагом к реализации метода экспресс-скрининга больных сосудистыми заболеваниями.

8 Список литературы

1. Buyukyilmaz M., Cibikdiken A. O. Voice Gender Recognition Using Deep Learning. – 2016.
2. Никитин Ю. М. и др. Ультразвуковая допплеровская диагностика в клинике. – 2004
3. Кунцевич Г. И. Ультразвуковая допплерография сосудов дуги аорты и их ветвей: Методические рекомендации //Москва,-1996г. – 1996.
4. Воронцов К. В. Лекции по метрическим алгоритмам классификации //М.: МФТИ. – 2007.
5. Воронцов К. В. Лекции по алгоритмическим композициям //КВ Воронцов. – 2012.
6. Choi K. et al. A comparison on audio signal preprocessing methods for deep neural networks on music tagging //arXiv preprint arXiv:1709.01922. – 2017.
7. Hershey S. et al. CNN architectures for large-scale audio classification //Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. – IEEE, 2017. – С. 131-135.
8. Breiman L. Random forests //Machine learning. – 2001. – Т. 45. – №. 1. – С. 5-32.
9. Adam S. Y., Yousif A., Bashir M. B. Classification of Ischemic Stroke using Machine Learning Algorithms //International Journal of Computer Applications. – 2016. – Т. 149. – №. 10.
10. Журавлёв Ю. И. Об алгебраическом подходе к решению задач распознавания или классификации // Проблемы кибернетики. 1978
11. Biau G. et al. A weighted k-nearest neighbor density estimate for geometric inference // Electronic Journal of Statistics.— 2011.— Vol. 5.— Pp. 204–237.
12. Scikit-learn, http://scikit-learn.org