

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ АЛГОРИТМОВ OVA, AVA И OVO В ЗАДАЧЕ МУЛЬТИКЛАССОВОЙ КЛАССИФИКАЦИИ

Вольф Д.А., Галин Р.Р.

Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия

runsolar@mail.ru, grt@ipu.ru

Аннотация. В статье приведен результат сравнительного анализа основных подходов к мультиклассовой классификации, осуществлён синтез их математических моделей на основе рассмотренных алгоритмов. Приведены их особенности, преимущества и недостатки, а также проведена оценка точности мультиклассовой классификации. Представлены результаты численных экспериментов.

Ключевые слова: мультиклассовая классификация, математическая модель, синтез, алгоритм, машинное обучение.

Введение

В современном мире, где данные становятся всё более разнообразными и объёмными, задача классификации объектов становится всё более актуальной. Одним из ключевых аспектов этой задачи – мультиклассовая классификация, которая позволяет классифицировать объекты по нескольким категориям одновременно. Такие подходы имеют свои преимущества и недостатки, и выбор конкретного подхода зависит условий задачи и доступных ресурсов. Однако, несмотря на их широкое применение, результаты анализа научной литературы показали отсутствие комплексного обзора этих подходов, который бы позволил исследователям лучше понять их особенности и выбрать наиболее подходящий для своих задач.

В настоящее время существует множество алгоритмов мультиклассовой классификации, которые используются в различных областях науки и техники. Однако, несмотря на их широкое применение, не существует качественных научных обзоров, посвящённых этим алгоритмам. Это может быть связано с несколькими причинами:

- Сложность исследуемой темы. Алгоритмы мультиклассовой классификации являются сложными и многогранными, что затрудняет их понимание и анализ. Для написания качественного научного обзора необходимо глубокое понимание этих алгоритмов и их особенностей.
- Отсутствие единого подхода. В настоящее время нет единого стандарта или методологии для проведения научных обзоров алгоритмов мультиклассовой классификации. Каждый автор может использовать свой подход к анализу и сравнению этих алгоритмов, что приводит к разнообразию результатов и мнений.
- Быстрое развитие предметной области исследования. Новые методы и реализующие их алгоритмы модернизируются, что делает задачу написания качественного обзора более сложной.
- Конкуренция и коммерческая тайна. Некоторые компании и организации могут быть не заинтересованы в сохранении своих разработок и методов в открытом доступе, что также может затруднить проведение научных исследований и проведении систематических анализов предметной области исследования.

Таким образом, отсутствие качественных научных обзоров на алгоритмы мультиклассовой классификации может быть обусловлено сложностью темы, отсутствием единого подхода, быстрым развитием предметной области исследования, недостатком ресурсов и конкуренцией. Тем не менее, авторами проведен качественный обзор алгоритмов мультиклассовой классификации на основе систематического анализа научной литературы. Были выявлены основные тенденции и проблемы в предметной области и предложены новые подходы и решения. Результаты проведенной работы позволили создать более полное и объективное представление о данной теме и способствовать её дальнейшему развитию.

В статье рассматриваются три основных алгоритма мультиклассовой классификации [1–4]:

- «Один против одного» (англ.: One-vs-One (OvO)),
- «Один против всех» (англ.: One-vs-All (OvA)),
- «Все против всех» (англ.: All-vs-All (AvA)).

Рассмотренные алгоритмы имеют свои особенности и преимущества, но их математическая формализация в современной литературе освещена недостаточно полно. Это затрудняет понимание принципов работы алгоритмов и их применение на практике.

Целью данной работы является проведение сравнительного анализа алгоритмов OvA, AvA и OvO и их математической формализации моделей. Для этого в статье рассматриваются основные принципы работы каждого подхода и соответствующие шаги алгоритмов с синтезом математических моделей, проводится анализ их преимуществ и недостатков, а также предоставлены результаты экспериментов, которые позволили оценить их точность на тестовом наборе данных.

1. Исследование модели персептрона в задачах бинарной классификации

Классификация — это подкатегория обучения с учителем, где целью является прогнозирование категориальных меток классов, к которым принадлежат новые образцы, на основе ретроспективных наблюдений. Мультиклассовая классификация — это задача, в которой необходимо отнести объект к одному из трёх или более классов. Это один из основных видов задач в машинном обучении.

Техника трансформации в бинарную классификацию — один из методов решения задачи мультиклассовой классификации заключается в том, что исходная задача преобразуется в несколько задач бинарной классификации.

Для этого каждый класс из исходной задачи становится отдельной задачей бинарной классификации, где положительный класс — это примеры из исходного класса, а отрицательный класс — примеры из всех остальных классов.

Например, если существует задача классификации изображений на три класса: «кошка», «цветок» и «овощ», то она может быть преобразована в три задачи бинарной классификации:

- «кошка» против «не кошка»;
- «цветок» против «не цветок»;
- «овощ» против «не овощ».

Каждая из этих задач решается отдельно, и затем результаты объединяются для получения окончательного результата мультиклассовой классификации. Однако, прежде чем переходить к рассмотрению мультиклассовой классификации, необходимо формализовать модель персептрона.

В статье [5] Уорреном Мак-Каллоком и Уолтером Питтсом была представлена первая концепция упрощенной клетки головного мозга (McCulloch-Pitts — MCP) с целью демонстрации, как работает биологический мозг и его математического моделирования искусственного интеллекта.

Согласно предложенной концепции, биологический нейрон может быть представлен как простой логический вентиль с двоичными выходами; множество сигналов достигают дендритов, которые встраиваются в тело клетки, и, если накопленный сигнал превышает определенный порог, тогда и только тогда генерируется выходной сигнал, который в дальнейшем передается аксоном.

Основные принципы модели:

1. Нейрон имеет несколько входов, каждый из которых может принимать двоичные значения: 0 или 1.
2. Каждый вход связан с весом, который определяет его вклад в выходное значение нейрона.
3. Выходное значение нейрона определяется как функция от входных значений и весов.

Формально такая модель может быть записана следующим образом:

$$z = w_1x_1 + w_1x_1 + \dots + w_mx_m = w^T x, \quad (1)$$

$$\phi(z) = \begin{cases} 1, & z \geq 0 \\ -1, & z < 0 \end{cases} \quad (2)$$

где:

z — общий вход;

x — вектор входных признаков;

w — вектор с весовыми коэффициентами;

$\phi(z)$ — единичная ступенчатая функция решения.

На вход (1) принимает вектор входных параметров $x = (x_1, \dots, x_n)$, где каждому компоненту вектора входных параметров x_i соответствует весовой коэффициент w_i , определяющий величину воздействия компонента x_i на персептрон.

Представленная модель является очень простой и не учитывает многие аспекты работы реальных нейронов, но она послужила основой для создания более сложных и реалистичных моделей. Становится очевидно, что основной задачей в (1) является нахождение таких w , чтобы выход максимально соответствовал.

В 1957 году в работе Ф. Розенблатта [6] было изложено правило обучения модели нейрона (порогового персептрона) (1), (2). Вместе с правилом порогового персептрона Розенблатт предложил

алгоритм, оптимально вычисляющий w , на которые, путем умножения x для принятия решения о том, активируется нейрон или нет. Таким образом, правило соответствует следующему:

1. Инициализация w нулями или небольшими случайными значениями.
2. Для каждого примера $x^{(i)}$:
 - а) вычислить (1);
 - б) обновить значения в w .

В модели выходными значениями являются метки классов, прогнозируемые с помощью (2), а обновление весовых коэффициентов в w может быть формально записано как:

$$w_j := w_j + \Delta w_j, \quad (3)$$

где Δw_j , вычисляется правилом обучения персептрона следующим образом:

$$\Delta w_j := \eta (y^{(i)} - \hat{y}^{(i)}) x_j^{(i)}, \quad (4)$$

где:

y – метка класса;

$\hat{y}^{(i)}$ – текущее выходное значение (прогнозируемая метка класса);

η – скорость подбора w .

Таким образом, сравнивая две рассмотренные модели можно заключить, что модель Маккаллока-Питтса представляет собой математическую модель искусственного нейрона, который имеет несколько входов и один выход. Функцией активации в модели Маккаллока-Питтса может выступать любая логическая функция, например, И, ИЛИ, НЕ. В персептроне используется пороговая функция активации, которая может быть сигмоидальной или ступенчатой. Персептрон Розенблатта обладает способностью к обучению, в то время как модель Маккаллока-Питтса не имеет такой возможности. Для обучения персептрона используется алгоритм коррекции ошибок, который позволяет настроить веса связей таким образом, чтобы нейронная сеть правильно классифицировала входные данные.

В свою очередь, с помощью моделей порогового или адаптивного персептрона может быть составлен алгоритм, который будет определять принадлежность объекта к одному из классов. Однако стоит отметить, что эта модель будет работать верно только при условии линейной делимости классов, т.е. при существовании гиперплоскости, отделяющей одно множество от другого.

Далее в работе представлено подробное рассмотрение алгоритмов OvA, AvA и OvO, которые являются основными подходами к мультиклассовой классификации.

2. Обзор существующих алгоритмов мультиклассовой классификации

2.1. Алгоритм «Один против одного»

«Один против одного» (OvO) — алгоритм, который создаёт отдельный классификатор для каждой пары классов. Во время предсказания алгоритм выбирает классификатор, который лучше всего отличает текущий объект от всех остальных.

Идея самого метода заключается в том, чтобы разбить задачу мультиклассовой классификации на $K(K-1)/2$ задач бинарной классификации, где K — количество классов. Например, если у нас есть три класса, мы получим следующие задачи бинарной классификации:

1. Класс 1 против класса 2.
2. Класс 1 против класса 3.
3. Класс 2 против класса 3.

После разбивки обучается отдельный классификатор (например, персептрон или логистическая регрессия) для каждой задачи бинарной классификации. Далее осуществляется объединение результатов всех классификаторов следующим образом:

1. Если классификатор предсказывает, что объект принадлежит классу 1, считается, что он принадлежит данному классу.

2. Если ни один из классификаторов не предсказывает принадлежность объекта к какому-либо из классов с высокой уверенностью, используется метод голосования большинством или иными методами для определения окончательного результата.

В работах [7–11] приводится следующий принцип работы OvO алгоритма:

Шаг 1. Подготовка данных:

- 1.1. Сбор набора данных для задачи классификации.
- 1.2. Разделение данных на обучающую и тестовую выборки.

Шаг 2. Построение классификаторов:

- 2.1. Построение $K(K-1)/2$ бинарных классификаторов для каждого класса.
- 2.2. Обучение каждого классификатора на данных одного класса против данных другого класса.

Шаг 3. Классификация нового объекта:

- 3.1. Получение нового объекта для классификации.
- 3.2. Применение всех классификаторов к новому объекту.
- 3.3. Выбор класса, для которого значение соответствующего классификатора максимально.

Шаг 4. Оценка результата:

- 4.1. Сравнение предсказанного класса с истинным классом объекта.
- 4.2. Оценка точности классификации с помощью метрики, такой как точность, полнота или F-мера.

Шаг 5. Повторение:

- 5.1. Повторение шагов 3 и 4 для всех новых объектов.

Исходя из данного алгоритма, осуществляется этап синтеза математической модели классификатора. Для этого сформулируем следующее определение:

Определение. Пусть имеется K классов и набор данных X , содержащий объекты с метками классов $y \in \{1, \dots, K\}$. Для каждой пары классов (i, j) , где $i \neq j$ создается отдельный классификатор $f_{ij}(x)$, который обучен на данных этих двух классов. При классификации нового объекта x выбирается класс k , для которого значение $f_k(x)$ максимально.

Далее математическую модель данного классификатора для каждой (k_i, k_j) и $i \neq j$ можно записать следующим формальным образом:

$$f_{ij}: X \rightarrow [0, 1], \quad (5)$$

где:

X — пространство объектов, которые необходимо классифицировать.

Классификатор f_{ij} обучается на выборке, состоящей из объектов класса k_i и класса k_j . Новый объект x классифицируется следующим образом: для каждой пары классов (k_i, k_j) вычисляется предсказание классификатора $f_{ij}(x)$.

Выбирается класс $f_k(x)$, для которого предсказания классификаторов максимальны:

$$f(x) = \arg \max_{k \in K} \sum_{(k_i, k) \in K \times k} f_{ij}(x). \quad (6)$$

Если применяется вероятностный подход выбора, тогда модель можно записать следующим образом:

Пусть X — множество объектов, которые необходимо классифицировать, а также пусть $C = \{k_1, k_2, \dots, k_K\}$ — множество классов, где K — количество всевозможных классов. Если для каждого класса k_i построен бинарный классификатор f_{ij} , который принимает на вход объект $x \in X$ и возвращает вероятность того, что объект принадлежит классу k_i .

Тогда:

1. Во время прогнозирования для объекта x вычисляются вероятности $p_{i,j} = f_{ij}(x)$ для всех пар классов (k_i, k_j) , где $i \neq j$.

2. Выбирается класс $f_k(x)$, для которого вероятность $p^* = \max_{i,j} p_{i,j}$ максимальна.

Таким образом, математическая модель OvO представляет собой набор бинарных классификаторов, каждый из которых сравнивает один класс с другим классом. Во время прогнозирования выбирается класс, для которого принадлежность объекта максимальна близка (вероятна).

2.2. Алгоритм «Один против всех»

«Один против всех» (OvA) — алгоритм машинного обучения, который используется для задач классификации. Алгоритм основан на сравнении одного класса с остальными. Основная идея метода заключается в том, чтобы разбить задачу мультiclassовой классификации на несколько задач бинарной классификации. Для этого создается отдельный классификатор, который обучен отличать объекты одного класса от всех остальных. Во время предсказания алгоритм выбирает классификатор, соответствующий классу, к которому наиболее вероятно принадлежит объект. Затем результаты всех классификаторов объединяются, чтобы получить окончательный результат.

Принцип работы алгоритма OvA изложен в работах [12–15]. Его пошаговая инструкция приводится далее:

Шаг 1. Подготовка данных:

- 1.1. Сбор набора данных, содержащий объекты с метками классов.
- 1.2. Разделение данных на обучающую и тестовую выборки.

Шаг 2. Создание классификаторов:

- 2.1. Создание отдельного классификатора для каждого класса.
- 2.2. Обучение каждого классификатора на данных только этого класса.

Шаг 3. Классификация нового объекта:

- 3.1. Получение нового объекта для классификации.
- 3.2. Применение классификаторов к новому объекту.
- 3.3. Выбор класса, для которого значение соответствующего классификатора максимально.

Шаг 4. Оценка результата:

- 4.1. Сравнение предсказанного класса с истинным классом объекта.

Шаг 5. Повторение:

- 5.1. Повторение шагов 3 и 4 для всех новых объектов.

Придадим следующее формальное определение алгоритму.

Определение. Пусть имеется K классов и набор данных $\{(x_k, c_k)\}_{k=1}^K$, содержащий объекты с метками классов $c_k \in \{1, \dots, K\}$. Для каждого класса k создаётся классификатор $f_k(x)$, который обучен на данных только этого класса. При классификации нового объекта x , выбирается класс k , для которого значение $f_k(x)$ максимально.

Задается для каждого классификатора k найдены некоторые значения w_k . Пусть имеется пространство объектов $\{(x_k, c_k)\}_{k=1}^K$, где $c_k \in \{1, \dots, K\}$. И для каждого $j = 1, \dots, K$ решается подзадача бинарной классификации. Тогда:

$$\begin{cases} f_1(x_i, w_1) = 1 \rightarrow x_i \in c_1 \\ \dots \\ f_k(x_i, w_k) = 1 \rightarrow x_i \in c_k, \\ \dots \\ f_K(x_i, w_K) = 1 \rightarrow x_i \in c_K \end{cases} \quad (7)$$

$$f_k(x, w_k) = \begin{cases} 1, & c_k = j \\ -1, & c_k \neq j \end{cases} \quad (8)$$

Тогда решающая функция имеет следующий вид:

$$f(x) = \underset{k=1, \dots, K}{\operatorname{argmax}} f_k(x, w_k). \quad (9)$$

Таким образом, математическая модель OvA представляет собой набор K бинарных классификаторов, каждый из которых сравнивает один класс с остальными классами. Причем в каждом классификаторе положительные примеры — это все признаки, относящиеся к k -му классу, а отрицательные — все признаки, не относящиеся к данному классу.

Основным недостатком алгоритма является то, что обучение каждой модели осуществляется независимо от остальных. Из чего следует ухудшение точности конечного результата классификации.

2.3. Алгоритм «Все против всех»

«Все против всех» (AvA) — алгоритм, который создаёт один классификатор для всех классов. Во время предсказания алгоритм сравнивает текущий объект со всеми остальными и выбирает класс с наибольшей вероятностью. AvA учитывает корреляцию между классами, но более сложный в реализации, чем два предыдущих рассмотренных алгоритма и может привести к переобучению.

Принцип работы алгоритма изложен в работах [16–18] и представлен ниже:

Шаг 1. Подготовка данных:

- 1.1. Сбор набора данных, содержащий объекты с метками классов.
- 1.2. Разделение данных на обучающую и тестовую выборки.

Шаг 2. Создание классификаторов:

- 2.1. Для каждой пары классов создается отдельный классификатор.
- 2.2. Обучение каждого классификатора на данных одного класса против данных другого класса.

Шаг 3. Классификация нового объекта:

- 3.1. Получение нового объекта для классификации.
- 3.2. Применение всех классификаторов к новому объекту.
- 3.3. Выбор класса, для которого значение соответствующего классификатора максимально.

Шаг 4. Оценка результата:

- 4.1. Сравнение предсказанного класса с истинным классом объекта.
- 4.2. Оценка точности классификации с помощью метрики, такой как точность, полнота или F-мера.

Шаг 5. Повторение:

5.1. Повторение шагов 3 и 4 для всех новых объектов.

Придадим следующее формальное определение алгоритму.

Определение. Пусть имеется K классов и набор данных X , содержащий объекты с метками классов $c_k \in \{1, \dots, K\}$. Для каждой пары классов (i, j) создаётся отдельный классификатор $f_{ij}(x)$, который обучен на данных этих двух классах. При классификации нового объекта x выбирает класс k , для которого значение $f_k(x)$ максимально.

Тогда математическую модель классификатора можно записать следующим образом:

1. Пусть для пары классов (i, j) найдены все $f_{ij}(x)$ и выполняется разделение:

$$\begin{matrix} 0 & f_{12}(x, w_{12}) & \dots & f_{1K}(x, w_{1K}) \\ f_{21}(x, w_{21}) & 0 & \dots & f_{2K}(x, w_{2K}) \\ \dots & \dots & \dots & \dots \\ f_{K1}(x, w_{K1}) & f_{K2}(x, w_{K2}) & \dots & 0 \end{matrix}, \quad (10)$$

$$f_{ij}(x, w_{ij}) = \begin{cases} 1, & x \in c_i \\ -1, & x \in c_j' \end{cases} \quad (11)$$

при этом f_{ij} задается следующим образом:

$$f_{ij}(x, w_{ij}) = -f_{ji}(x, w_{ji}), i, j = 1, \dots, K, \quad (12)$$

2. Тогда решающая функция

$$f(x) = \underset{k=1, \dots, K}{\operatorname{argmax}} \sum_{\substack{i=1 \\ i \neq j}}^K [f_{ij}(x) = k]. \quad (13)$$

3. Эксперимент

Сравним точность алгоритмов OvA, AvA и OvO в задаче мультиклассовой классификации на примере набора данных с несколькими классами.

Гипотеза: предполагается, что алгоритм OvA имеет более низкую точность классификации по сравнению с алгоритмами AvA и OvO. Алгоритмы AvA и OvO будут иметь примерно одинаковую точность классификации.

Для проведения эксперимента выбран набор данных по двум признакам с тремя классами ирисов из базы Iris DB [19–21]. Для оценки точности методов мультиклассовой классификации использовались следующие метрики [22–24]:

1. Точность (Accuracy) — доля правильно классифицированных объектов среди всех объектов в наборе данных.
2. Полнота (Recall) — доля объектов из определённого класса, которые были правильно классифицированы как принадлежащие этому классу.
3. F1-мера (F1 score) — гармоническое среднее между точностью и полнотой.

Для оценки точности алгоритмов выполнены следующие шаги:

1. Данные разделены на обучающую и тестовую выборки. Обучающая выборка использовалась для обучения моделей, а тестовая выборка — для оценки их точности.
2. Обучение моделей на обучающей выборке. Для каждого алгоритма обучена модель на данных соответствующего типа.
3. Произведено тестирование моделей на тестовой выборке. Произведена оценка точности каждой модели на тестовых данных с помощью выбранных метрик.

Результаты тестирования показаны в таблице 1.

Таблица 1. Результаты тестирования точности алгоритмов OvO, OvA и AvA

Метод	Точность	Полнота	F1-мера
One-vs-All	0.79	0.78	0.75
One-vs-One	0.93	0.91	0.91
All-vs-All	0.89	0.89	0.89

Из результатов тестирования точности работы алгоритмов OvO, OvA и AvA можно выделить следующие особенности:

1. One-versus-One (OvO):
 - 1.1. Создаётся классификатор для каждой пары классов.

1.2. Может потребовать больше времени на обучение, но может обеспечить более высокую точность классификации за счёт использования большего количества информации о различиях между классами.

2. One-versus-All (OvA):

2.1. Для каждого класса создаётся отдельный классификатор.

2.2. Является простым и эффективным способом мультиклассовой классификации, который может быть использован в различных задачах машинного обучения. Однако он может привести к переобучению, если количество классов слишком велико.

3. All-versus-All (AvA):

3.1. Создаются все возможные пары классификаторов, и каждый классификатор обучается на данных одного класса против данных другого класса.

3.2. Может обеспечить более высокую точность классификации по сравнению с OvA, но требует больше вычислительных ресурсов.

В целом, метод OvO является хорошим выбором для задач бинарной классификации с небольшим количеством классов и достаточным количеством данных. Метод OvA подходит для мультиклассовой классификации, но может иметь неравномерность моделей и сложность интерпретации результатов. Метод AvA подходит для задач с большим количеством классов, но требует больше вычислительных ресурсов и может переобучаться на данных. Выбор метода также зависит от конкретных требований к точности и скорости работы модели. Например, если требуется высокая точность, то можно использовать метод AvA или OvA. Если же важна скорость работы, то лучше выбрать метод OvO.

4. Заключение

В данной статье были рассмотрены три основных подхода к мультиклассовой классификации: OvA, AvA и OvO. Эти подходы имеют свои преимущества и недостатки, и выбор конкретного подхода зависит от конкретной задачи и доступных ресурсов.

Результаты экспериментов показали, что каждый из подходов имеет свои особенности и может быть более или менее применим в зависимости от конкретных условий задачи. Алгоритм OvA является простым и эффективным способом мультиклассовой классификации, который может быть использован в различных задачах машинного обучения. Однако он может привести к переобучению, если количество классов слишком велико. Алгоритмы AvA и OvO могут обеспечить более высокую точность классификации за счёт использования большего количества информации о различиях между классами, но требуют больше вычислительных ресурсов.

Таким образом, выбор конкретного алгоритма для решения задачи мультиклассовой классификации зависит от требований к точности и вычислительным ресурсам. Для задач с небольшим количеством классов и ограниченными ресурсами может быть выбран алгоритм OvA. Для задач, требующих высокой точности классификации и наличия достаточных вычислительных ресурсов, могут быть выбраны алгоритмы AvA или OvO.

Дальнейшие исследования могут включать в себя разработку новых методов мультиклассовой классификации, основанных на комбинации рассмотренных подходов, а также исследование влияния различных параметров на эффективность этих методов.

Литература

1. *Kini G.R. and Thrampoulidis C.* Phase Transitions for One-Vs-One and One-Vs-All Linear Separability in Multiclass Gaussian Mixtures // ICASSP 2021 – 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 2021, P. 4020-4024.
2. *Daengduang S. and Vateekul P.* Enhancing accuracy of multi-label classification by applying one-vs-one support vector machine // 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), Khon Kaen, Thailand, 2016, P. 1-6.
3. *Joutsijoki H.* An Application of One-vs-One Method in Automated Taxa Identification of Macroinvertebrates // 2013 Fourth Global Congress on Intelligent Systems, Hong Kong, China, 2013, P. 125-130.
4. *Franchi G. et al.* One Versus All for Deep Neural Network for Uncertainty (OVNNI) Quantification // IEEE Access, vol. 10, 2022. P. 7300-7312.
5. *McCulloch W.S., Pitts W.* A logical calculus of the ideas immanent in nervous activity // The bulletin of mathematical biophysics. – 1943. – T. 5. – P. 115-133.
6. *Rosenblatt F.* The perceptron, a perceiving and recognizing automaton Project Para. – Cornell Aeronautical Laboratory, 1957.
7. *Liu J.Y., Jia B.B.* Combining one-vs-one decomposition and instance-based learning for multi-class classification // IEEE Access. – 2020. – T. 8. – P. 197499-197507.

8. *Kudo M., Takahashi T., Yamana H.* Touch-Based Continuous Mobile Device Authentication Using One-vs-One Classification Approach // 2024 IEEE International Conference on Big Data and Smart Computing (BigComp). – IEEE, 2024. – P. 167-174.
9. *Varpa K. et al.* Applying one-vs-one and one-vs-all classifiers in k-nearest neighbour method and support vector machines to an otoneurological multi-class problem // User centred networked health care. – IOS Press, 2011. – P. 579-583.
10. *Pawara P. et al.* One-vs-One classification for deep neural networks // Pattern Recognition. – 2020. – Vol. 108. – P. 107528.
11. *Kang S., Cho S., Kang P.* Constructing a multi-class classifier using one-against-one approach with different binary classifiers // Neurocomputing. – 2015. – Vol. 149. – P. 677-682.
12. *Liu C.L.* One-vs-all training of prototype classifier for pattern classification and retrieval // 2010 20th International Conference on Pattern Recognition. – IEEE, 2010. – P. 3328-3331.
13. *Widodo S. et al.* The One-vs-Rest Method for a Multilabel Patent Classification Machine Learning Approach using a Regression Model // 2023 International Conference on Informatics, Multimedia, Cyber and Informations System (ICIMCIS). – IEEE, 2023. – P. 688-693.
14. *Vogiatzis A. et al.* A Novel One-vs-Rest Classification Framework for Mutually Supported Decisions by Independent Parallel Classifiers // 2021 IEEE International Conference on Imaging Systems and Techniques (IST). – IEEE, 2021. – P. 1-6.
15. *Tao W., Yongjia J., Xiangsheng R.* A novel two-level one-vs-rest classifier // 2019 2nd International Conference on Information Systems and Computer Aided Education (ICISCAE). – IEEE, 2019. – P. 645-648.
16. *Otto K.T.J. et al.* Machine Learning meets Kepler: Inverting Kepler's Equation for All vs All Conjunction Analysis // Machine Learning: Science and Technology. – 2024.
17. *Díaz-Domínguez D., Salmela L.* Computing all-vs-all MEMs in grammar-compressed text // International Symposium on String Processing and Information Retrieval. – Cham: Springer Nature Switzerland, 2023. – P. 157-170.
18. *Pimenta E., Gama J.* A study on error correcting output codes // 2005 portuguese conference on artificial intelligence. – IEEE, 2005. – P. 218-223.
19. *Thirunavukkarasu K. et al.* Classification of IRIS dataset using classification based KNN algorithm in supervised learning // 2018 4th International Conference on Computing Communication and Automation (ICCCA). – IEEE, 2018. – P. 1-4.
20. *Schlett T., Rathgeb C., Busch C.* Multi-spectral iris segmentation in visible wavelengths // 2018 International Conference on Biometrics (ICB). – IEEE, 2018. – P. 190-194.
21. *Khan M. T., Arora D., Shukla S.* Feature extraction through iris images using 1-D Gabor filter on different iris datasets // 2013 Sixth International Conference on Contemporary Computing (IC3). – IEEE, 2013. – P. 445-450.
22. *Yan B. C. et al.* Maximum f1-score training for end-to-end mispronunciation detection and diagnosis of L2 English speech // 2022 IEEE International Conference on Multimedia and Expo (ICME). – IEEE, 2022. – P. 1-5.
23. *Sepúlveda J., Velastin S.A.* F1 score assesment of Gaussian mixture background subtraction algorithms using the MuHAVi dataset. – 2015.
24. *Miaoli W.* Modified F score normalization method of multi-modal biometric recognition // 2015 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). – IEEE, 2015. – P. 15-19.