Раздел III. Машинное обучение и нейронные сети

УДК 004.89:004.85 DOI 10.18522/2311-3103-2020-3-133-146

Д.В. Балабанов, А.В. Ковтун, Ю.А. Кравченко

ДВУХЭТАПНЫЙ БУСТИНГ БИНАРНОЙ КЛАССИФИКАЦИИ НА ОСНОВЕ ПРИМЕНЕНИЯ БИОИНСПИРИРОВАННЫХ АЛГОРИТМОВ*

В процессе решения широкого круга прикладных задач возникает необходимость декомпозиции объектов. Как следствие, проблема классификации является актуальной проблемой в современных системах интеллектуального анализа данных. Бинарная классификация является одной из важнейших задач, и имеет целый ряд нерешенных проблем. Одной из таких проблем является эффективность автоматизированной классификации. В задачах автоматизированной классификации, актуально применение алгоритмического аппарата эволюционных вычислений. Таким образом целесообразно применение генетических и биоинспирированных алгоритмов, в задаче поиска оптимальных значений параметров классификатора. Для решения данной задачи предлагается применить алгоритм роя частиц(PSO). Данный алгоритм в контексте задачи поиска субоптимальных значений параметров классификатора способен обеспечить высокое качество классификации. Модификацией алгоритма является динамическое изменение значений координат, которые отвечают за тип функции ядра. Данная доработка позволяет значительно снизить затрачиваемое время разработки классификатора. Для повышения эффективности классификации целесообразно применять ансамбли алгоритмов. В работе приведена структура двухуровневого классификатора. На первом уровне данного классификатора, формируется ансамбль простых классификаторов которые формируют учебную выборку, которая, в дальнейшем используется алгоритмом роя частиц на втором этапе. Такой подход позволяет значительно уменьшить временные затраты, а также повысить качество получаемых решений. Алгоритм роя частиц(PSO), в контексте задачи поиска субоптимальных значений параметров классификатора способен обеспечить высокое качество классификации. Предложенный двухуровневый алгоритм был экспериментально протестирован. Произведено сравнение с аналогами, приведены сравнительные диаграммы. Описанные исследования показывают, что работа имеет высокую теоретическую значимость, а проведенные экспериментальные исследования доказывают высокую практическую значимость.

Классификация; бинарная классификация; биоинспирированные методы; метод опорных векторов; бустинг; алгоритм роя частиц.

D.V. Balabanov, A.V. Kovtun, Yu.A. Kravchenko

TWO-STAGE BOOSTING OF BINARY CLASSIFICATION BASED ON THE APPLICATION OF BIOINSPIRED ALGORITHMS

In the process of solving a wide range of applied problems, it becomes necessary to decompose objects. As a result, the classification problem is an urgent problem in modern data mining systems. Binary classification is one of the most important tasks, and has a number of unsolved problems. One such problem is the effectiveness of automated classification. In the tasks of automated classification, it is relevant to use the algorithmic apparatus of evolutionary computing. Thus, it is advisable to use genetic and bio-inspired algorithms in the task of finding the optimal

* Исследование выполнено при финансовой поддержке РФФИ в рамках научных проектов №№ 19-07-00099 и 18-07-00050.

values of the classifier parameters. To solve this problem, it is proposed to apply the particle swarm algorithm (PSO). This algorithm in the context of the task of finding suboptimal values of the parameters of the classifier is able to provide high quality classification. A modification of the algorithm is a dynamic change in the coordinate values that are responsible for the type of kernel function. This revision can significantly reduce the time spent developing the classifier. To increase the classification efficiency, it is advisable to use ensembles of algorithms. The paper presents the structure of a two-level classifier. At the first level of this classifier, an ensemble of simple classifiers is formed that form the training set, which is further used by the particle swarm algorithm in the second stage. This approach can significantly reduce time costs, as well as improve the quality of the resulting solutions. The particle swarm algorithm (PSO), in the context of the task of finding suboptimal values of the parameters of the classifier, is able to provide high quality classification. The proposed two-level algorithm has been experimentally tested. A comparison is made with analogues, comparative charts are given. The described studies show that the work is of high theoretical significance, and the conducted experimental studies prove high practical significance.

Classification; binary classification; bio-inspired methods; support vector method; boosting; particle swarm algorithm.

Введение. В современном мире тенденции информационных технологий, которые касаются сферы поддержки принятия решений, тесно связаны с такими понятиями как Big Data и Data Mining, а также с разработкой программных средств интеллектуального анализа данных. Такие программные средства способны обрабатывать сложноорганизованные многомерные структуры данных. Такие данные присутствуют в множестве областей производства, медицины, экономики, банковской деятельности и т.д.

В процессе решения широкого круга прикладных задач возникает необходимость декомпозиции объектов. Как следствие, проблема классификации является актуальной проблемой в современных системах интеллектуального анализа данных. Бинарная классификация является одной из важнейших задач, и имеет целый ряд нерешенных проблем. Одной из таких проблем является эффективность автоматизированной классификации.

Классификация является разделом машинного обучения. Задачу классификации можно описать следующим образом: существует множество объектов, которые разделены каким-то образом на классы. Множество объектов конечно, отношения объектов к классам также известны. Данное множество является обучающей выборкой. Объекты, не включенные в обучающую выборку являются неопознанными, т.е. у них не известна привязка к классу. Таким образом необходимо разработать такой алгоритм, который классифицирует произвольный объект из исходного множества. В частном случае классификации – бинарной классификации, присутствует только 2 класса, непересекающихся между собой.

В задачах автоматизированной классификации, актуально применение алгоритмического аппарата эволюционных вычислений. Таким образом целесообразно применение генетических и биоинспирированных алгоритмов, в задаче поиска оптимальных значений параметров классификатора. Для повышения эффективности классификации целесообразно применять ансамбли алгоритмов.

На данном этапе развития, не существует универсальных алгоритмов и методов, которые будут способны решать задачи классификации и кластеризации. Существует множество различных принципов моделирования. В основе различных инструментов, лежат различные критерии, метрики, функции близости и т.д. Применение различных инструментов, даже по отношению к одному и тому же набору, может привести к всевозможным результатам. Таким образом, одним из вариантов получения качественного классификационного решения, является комбинирование алгоритмов [1–3].

Среди таким композиций или «ансамблей» алгоритмов, можно выделить бустинг и бэггинг. Данные принципы получили широкое, активное распространение благодаря высокому качеству получаемых решений. Идея бэггинга состоит в параллельной работе нескольких простых алгоритмов, где итоговое решение выбирается некоторым голосованием. Бустинг, в свою очередь, представляет собой последовательное применение нескольких алгоритмов, где каждый последующий алгоритм оперирует решениями предыдущего [3–7].

Цель работы – повышение эффективности алгоритмов, решающих задачу классификации и упорядочивания объектов, повышение адекватности принимаемых решений, основанных на инструментарии интеллектуальной обработки информации и биоинспирированном моделировании и применении бустинга биоинспирированных алгоритмов, позволяющих устранить недостатки существующих аналогов.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести аналитический обзор основных подходов к решению задачи бинарной классификации методом опорных векторов, изучить варианты применения генетических и биоинспирированных эвристик к поставленной задаче.

2. Описать задачу бинарной классификации, построить ее математическую модель на основе алгоритма опорных векторов.

3. Разработать модифицированный алгоритм роя частиц для дальнейшего применения его в ансамбле.

4. Привести структуру двухуровневого классификатора с применением бустинга. Данный ансамбль алгоритмов на первом этапе, при помощи нескольких классификаторов должен создать учебную выборку, которая в дальнейшем будет использоваться для работы алгоритма роя частиц. Такой ансамбль позволит получать повышенную точность классификации, несравнимую с точностью отдельных алгоритмов.

5. Произвести экспериментальные исследования, оценить эффективность биоинспирированного алгоритма, а также двухуровневого алгоритма.

Проведем аналитический обзор состояния исследований по выбранной тематике и дадим постановку задачи.

1. Аналитический обзор и постановка задачи классификации. Существует несколько алгоритмов для решения задачи классификации. В работе предлагается более подробно остановиться на алгоритме опорных векторов. Полученные с его помощью классификаторы, способны демонстрировать высокое качество решений, кроме этого ввиду своей относительной новизны, требуются дальнейшие исследования в области поиска субоптимальных значений его параметров. Параметры включают в себя как значения параметров ядра, так и значения параметра регуляции, а также используемой функции ядра. Изменение этих параметров непосредственно влияют на качество итоговой классификации.

Алгоритм опорных векторов (SVM классификатор)

Преимуществами алгоритма опорных векторов являются: способность работы с высоко-размерными входными данными, достаточная устойчивость к переобучению, высокая конкурентоспособность. Помимо этого, так как в основе алгоритма лежит решение задачи квадратичного программирования в выпуклой области, то такая задача имеет всего одно решение. SVM классификатор способен обучаться на сильно-коррелирующих выборках данных, размер которых составляют гигабайты.

Однако у любого алгоритма существуют недостатки. Недостатками SVM алгоритма являются: относительная нестабильность к шуму исходных данных, сложность выбора ядра в случае невозможности линейного разделения, поиск значения параметра регуляции (если классы линейно неразделимы)

Использование базового подхода к разработке SVM-классификатора на основе SVM-алгоритма применительно к сложноорганизованным многомерным данным больших объемов сопряжено с необходимостью решения задачи квадратичного программирования большой размерности [8], характеризующейся высокой сложностью выполнения расчетов и требующей существенных временных затрат. Для ослабления остроты проблемы высокой сложности выполнения расчетов и сокращения времени при разработке SVM-классификатора предлагаются различные оригинальные подходы [9–11].

Одним из вариантов является каскадный алгоритм опорных векторов [9]. Такой алгоритм обучается итеративно, на поднаборах исходного набора данных. Опорные векторы используются при формировании последующих учебных наборов данных. Такой алгоритм можно легко разделить на ансамбль алгоритмов, для ускорения работы.

Система Hadoop MapReduce является одним из лидеров в области Bid Data и работе с ними. В работе [10], представлены решения, направленные на использование классификаторов данной системы. В системе представлено огромное количество утилит и библиотек, для работы с SVM классификаторами.

Помимо прочих, интересным является подход уменьшения учебной выборки, для ускорения разработки итогового классификатора [11, 12]. В работе предлагается исключать неиспользуемые объекты, которые не имеют влияния на финальное решение.

По итогам рассмотрения, можно сделать вывод что проблеме разработки классификатора, методом опорных векторов, выделяется повышенное внимание, и ее решение является актуальной задачей.

Рассмотрим постановку задачи классификации.

Задача обучения по прецедентам включает в себя рассмотрение двух множеств Z и Y. Множество Z представляет собой множество объектов, а множество Y множество меток и допустимых решений. Также существует неизвестная целевая зависимость $f:Z \to Y$. Эти значения $f^*(z_i)$ будут получены только в конечном множестве объектов $Z^*=\{z_1,\ldots,z_s\}$ $(Z^* \subset Z)$ и равны $y_i^*(y_i^*= f^*(z_i))$.

Прецедентом называется кортеж «объект-ответ» $< z_i, y_i^* >$. Совокупность таких кортежей, где $i = (\overline{1,s})$ образуют учебный набор $U=\{<z_i, y_i^*>\}_{i=1}^{s}$. В этом учебном наборе каждый кортеж $< z_i, y_i^* >$ хранит в себе информацию $z_i \in Z^*$, а также метку класса $y_i^* \in Y$ принадлежности объекта z_i.

Таким образом имея учебный набор необходимо воссоздать некоторую зависимость f^*. Так как для решения такой задачи необходима программная реализация, такой процесс, создания неизвестной зависимости f^* называют обучением алгоритма. Итоговым результатом работы является решающая функция $A(z)$.

Процесс, создания зависимости f^*, используя учебный набор U, называют процессом построения решающей функции $A: Z \to Y$, что является *задачей обучения по прецедентам*. Данная решающая функция $A: Z \to Y$ приближает неизвестную целевую зависимость на объектах множества $Z^* \subset Z$ и на всем множестве Z [13].

Если множество Y конечно $Y=\{1,2,\ldots,c\}$, то такую задачу можно назвать *задачей классификации на c непересекающихся классов.* В таком случае можно сказать что множество Z разбито на подмножества $Z^1, Z^2, \ldots, Z^c$, где $Z^j = \{z \in Z | f(z) = j\}$ при $j \in \{1,2,\ldots,c\}$: $Z = \cup_{j=1}^{c}(z^j)$, а решающую функцию $A: Z \to Y$ называют классифицирующей функцией (правилом классификации) [14].

2. Задача бинарной классификации на основе алгоритма опорных векторов (SVM – алгоритм). Под задачей бинарной классификации понимают рассмотрение каждого объекта множества объектов Z, которое подлежит классификации. Каждому объекту множества ставится в соответствие q-мерный вещественный вектор характеристик

$$z_i=(z_i^1,z_i^2,\ldots,z_i^q), \quad (1)$$

где z_i^1 – числовое значение l-й характеристики для i-го объекта $l = (\overline{1,q})$, нормированное значениями [0;1]; $Y=\{-1,+1\}$ – множество ответов (метки классов); $f^*: Z \rightarrow Y$ – целевая зависимость, данные значения известны только на объектах учебного набора $U=\{<z_i, y_i^*>\}^s_{i=1}$, $y_i^*= f^*(z_i)$- число (-1 или +1), характеризующее классовую принадлежность объекта $z_i \in Z^*$, $Z^* \subset Z$ $(i = (\overline{1,s}))$. Необходимо построить классифицирующую функцию $A: Z \rightarrow Y$, аппроксимирующую целевую зависимость на пространстве Z. Для этого необходимо построить разделяющую гиперплоскость [14–16].

Чтобы получить максимально эффективный классификатор, требуется многоэтапное обучение, а также применение различных тестовых выборок. В последствие выбирается наилучший вариант обучения и тестирования. При использовании ансамбля алгоритмов, процесс поиска лучших выборок можно ускорить. В последствии «обученный» классификатор применяется для классификации неизвестных объектов из множества Z.

Пусть из учебного наборы $U=\{<z_i, y_i^*>\}^s i=1$ случайно выбраны S кортежей $(S<s)$,а также создана обучающая выборка $Train =\{<z_i, y_i^*>| z_i \in Z^*, y_i^*= f^*(z_i)\}^s i=1$. Итогом процесса обучения классификатора является определение разделяющей гиперплоскости. Данная гиперплоскость задается уравнением [14, 15]:

$$w^*z+b=0, \quad (2)$$

где w – вектор, перпендикулярный к разделяющей плоскости, $b \in R$ – параметр, определяющий смещение гиперплоскости относительно начала координат (при $b=0$ гиперплоскость совпадает с осью начала координат); w^*z – скалярное произведение векторов w и z.

Следует обратить внимание, что определение параметров классификатора задано с точность до нормировки: в случае одновременного умножения векторов w и b на положительную константу, алгоритм $A(z)$ не будет изменен. Таким образом, данная константа выбирается исходя из условия $w^*z_1+b=y_1$, для всех близких к разделяющей гиперплоскости объектов $z_i \in Z^*$. Расстояние, в данном случае, от разделяющей гиперплоскости до пограничных объектов обоих классов будет равно 1.

Идеальный случай построения разделяющей гиперплоскости представлен на рис. 1. В пространстве D-2, если объект находится на положительной части относительно гиперплоскости, ему присваивается первый класс, во всех остальных случаях второй класс. Полоса, разделяющая классы задается условием $-1< w^*z_1+b<1$. Гиперплоскость лежит идеально посередине данной полосы. Ей параллельны две полосы с направляющим вектором w, которые служат ей границами. Опорными векторами называют векторы, которые располагаются на границе полосы, которая разделяет классы. Именно эти векторы несут информацию о разделении [14]. Чем шире данная полоса, тем лучше, и тем легче можно классифицировать объекты. В обучающей выборке не должно быть объектов внутри границ этой полосы.

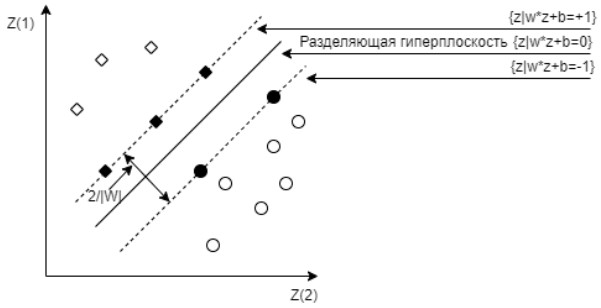

Рис. 1. Идеальный случай разделения классов гиперплоскостью

Для опорных векторов выполняется условие $w*z_1+b=y_1$. Остальные объекты располагаются на удалении, значит для всех $z_i \in Z*$ справедливы следующие неравенства [14, 15]:

$$\begin{cases} \boldsymbol{w} * \boldsymbol{z1} + \boldsymbol{b} \leq -\boldsymbol{1}, \text{если } \boldsymbol{y_i} = -\boldsymbol{1} \\ \boldsymbol{w} * \boldsymbol{z1} + \boldsymbol{b} \geq \boldsymbol{1}, \text{если } \boldsymbol{y_i} = +\boldsymbol{1} \end{cases} \textit{или} \;\; \boldsymbol{y_i}(\boldsymbol{w} * \boldsymbol{z1} + \boldsymbol{b}) \geq \boldsymbol{1} (\boldsymbol{i} = \overline{(\boldsymbol{1}, \boldsymbol{S})}). \quad (3)$$

Искомое правило классификации записывается в следующем виде:

$$A(z)=sign(w*z_1+b). \quad (4)$$

При допущении линейного разделения выборки, должны существовать такие w и b, при котором функционал числа ошибок будет равен нулю. При условии соблюдения ограничений:

$$\boldsymbol{Q}(\boldsymbol{w}, \boldsymbol{b}) = \frac{\boldsymbol{1}}{\boldsymbol{S}} \sum_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{S}} [\boldsymbol{y_i}(\boldsymbol{w} * \boldsymbol{z1} + \boldsymbol{b}) < 0]. \quad (5)$$

Однако, в данном случае присутствует несколько разделяющих плоскостей, которые реализуют тоже разбиение. Необходимо найти оптимальные значения w и b, которые позволяют разделяющей плоскости отстоять максимально удаленно от ближайших объектов обоих классов.

3. Алгоритм роя частиц в задаче разработки SVM-классификатора. В случае использования PSO-алгоритма при разработке SVM классификатора частицам роя могут быть сопоставлены векторы, описывающие их позиции в пространстве поиска и закодированные параметрами функции ядра и параметром регуляризации: (x_i^1, x_i^2, C_i), где i – номер частицы $(i=1,m)$; x_i^1, x_i^2 – параметры функции ядра i-й частицы (при этом параметр x_i^1 полагается равным параметрам функций ядра d, γ, σ или k_2 (в зависимости от того, какому типу функции ядра соответствует частица роя); параметр x_i^2 полагается равным параметру функций ядра k_1, если частица роя соответствует сигмоидному типу функции ядра, в противном случае значение этого параметра считается равным нулю.

Тогда традиционный подход к применению PSO-алгоритма при разработке SVM-классификатора заключается в многократном применении PSO-алгоритма при фиксированном типе функции ядра (в соответствии с целью выбора субоптимальных значений параметров функции ядра и значения параметра регуляризации [17–19]. Такой подход предполагает выполнение следующей последовательности шагов.

Шаг 1. Определить значения параметров PSO-алгоритма: число частиц в рое m, масштабирующий коэффициент для скорости K, личный и глобальный коэффициенты ускорения $\varphi\hat{}$ и $\varphi\sim$, максимальное число итераций PSO-алгоритма N_{max}.

Определить типы T функций ядра, участвующие в поиске ($T = 1$ – полиномиальная однородная, $T = 2$ – полиномиальная неоднородная, $T = 3$ – радиальная базисная, $T = 4$ – радиальная базисная функция Гаусса, $T = 5$ – сигмоидная функция ядра) и границы изменения значений параметров функции ядра и значения параметра регуляризации C для выбранных типов функций ядра T: x^{1T}_{min}, x^{1T}_{max}, x^{2T}_{min}, x^{2T}_{max}, C^{T}_{min}, C^{T}_{max} ($x^{2T}_{min} = 0$ и $x^{2T}_{max}=0$ для $T =1,4$).

Шаг 2. Для каждого выбранного на шаге 1 типа функции ядра T сгенерировать начальное положение i -й частицы $(i =1,m)$ с помощью случайного вектора (x_i^1, x_i^2, Ci), где $x_i^1 \in [x^{1T}_{min}, x^{1T}_{max}]$, $x_i^2 \in [x^{2T}_{min}, x^{2T}_{max}]$ ($x_i^2 = 0$ при $T =1,4$), $Ci \in [C^{T}_{min}, C^{T}_{max}]$; инициализировать случайный вектор скорости $v_i(v_i^1, v_i^2, v_i^3)$ i -й частицы ($i =1,m$) ($v_i^2 = 0$ при $T =1,4$). Принять начальное положение i -й частицы $(i=1,m)$ за лучшее ее известное положение $(\hat{x}^1_i, \hat{x}^2_i, \hat{C}_i)$ и определить лучшую частицу среди всех частиц $(\tilde{x}^1, \tilde{x}^2, \tilde{C})$ для рассматриваемого типа функции ядра T. После чего N_{max} раз для каждой i -й частицы $(i =1,m)$ выполнить:

- коррекцию вектора скорости $v_i(v_i^1, v_i^2, v_i^3)$ i-й частицы и ее положения (x_i^1, x_i^2, Ci) по формулам:

$$v_i^j = \begin{cases} x * (v_i^j + \hat{\varphi} * \hat{r} * (\hat{x}_i^j - x_i^j) + \tilde{\varphi} * \tilde{r}(\tilde{x}^j - x_i^j), j = 1, 2, \\ x * (v_i^j + \hat{\varphi} * \hat{r} * (\hat{C}_i - c_i) + \tilde{\varphi} * \tilde{r}(\hat{C} - C_i), j = 3 \end{cases} \quad (6)$$

$$x_i^j = x_i^j + v_i^j \text{ для } j = 1, 2, \quad (7)$$

$$C_i = C_i + v_i^3, \quad (8)$$

где $\hat{r}$ и $\tilde{r}$ – случайные числа в интервале (0, 1), χ – коэффициент сжатия, – расчет точности SVM-классификатора с параметрами (x_i^1, x_i^2, Ci) для рассматриваемого типа функции ядра T с целью поиска оптимальной комбинации $(\tilde{x}^1, \tilde{x}^2, \tilde{C})$, обеспечивающей высокое качество классификации.

В результате для каждого типа T функции ядра, участвующего в поиске, будет определена частица с оптимальной комбинацией значений параметров $(\tilde{x}^1, \tilde{x}^2, \tilde{C})$, обеспечивающая высокое качество классификации при использовании соответствующего типа функции ядра T.

Шаг 3. Выбрать из полученных для каждого типа функции ядра T, включенного в поиск, те значения параметров $(\tilde{x}^1, \tilde{x}^2, \tilde{C})$ SVM-классификатора и соответствующий тип функции ядра T, при которых качество классификации оказалось максимальным (наилучшим).

Лучший тип и лучшие значения соответствующих ему значений параметров определяются по результатам сравнительного анализа лучших частиц, полученные при реализации PSO-алгоритма с фиксированным типом функции ядра.

4. Структура двухуровневого классификатора с применением бустинга. Самая главная проблема, которая возникает в результате применения вышеприведенного алгоритма роя частиц, это повышенное время выполнения, затрачиваемое при поиске оптимальных или около оптимальных параметров классификатора. В больших объемах, данных, в многомерной среде, поиск оптимальных параметров ядра, типа функции ядра занимает много времени [12, 20]. Одними из вариантов решения данной проблемы являются уменьшение количества частиц в рое, а также уменьшений числа итераций. Однако, такой подход неизбежно приводит к ухудшению качества финального классификатора, в виду уменьшения вариантов выбора.

Для решения проблемы повышенной затраты времени, был рассмотрен подход, уменьшающий группу объектов, которые учувствуют при генерации обучающей и тестовой выборок. Объекты, которые не имеют влияния на финальный результат классификации, в данном случае не рассматриваются. Таким образом при обучении классификатора учитываются только опорные векторы, параметр кото-

рых не равен 0. Так как именно опорные векторы служат хранилищем информации о классах, а также то, что с учетом их расположения строится гиперплоскость, считается целесообразным использовать данный подход.

Таким образом целесообразна структура двухуровневого ансамбля классификаторов, которая повысит качество классификации в системах, характеризующихся сложной организацией с многомерными данными. Помимо прочего, данный подход не ведет к существенному повышению затрачиваемого времени. На первом уровне предлагается использовать группу классификаторов, полученных в результате работы нескольких простых алгоритмов, например, генетических, которые можно модифицировать для повышения эффективности. Эта группа алгоритмов результатом своей работы являют обучающую выборку для алгоритма роя частиц на втором этапе. Биоинспирированный алгоритм использует сформированный набор опорных векторов, полученный на первом этапе. Ниже, на рис. 2, приведена двухуровневая структура классификации с использованием бустинга.

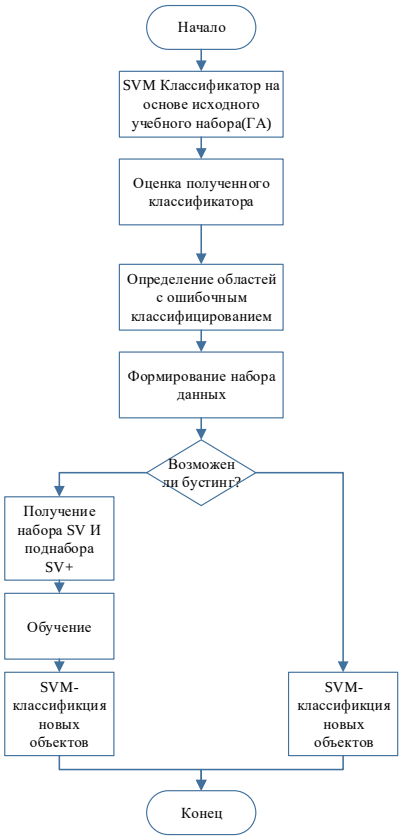

Рис. 2. Структура двухэтапного бустинга классификации объектов

1. Применение генетического алгоритма для создания t частных классификаторов. Классификаторы обучаются на учебном наборе. Для каждого из них используются свои обучающие выборки $TR_1, TR_2, \ldots, TR_t$. При этом для каждого из классификаторов задаются отдельно такие параметры как: типы и значения параметров функции ядра, а также параметры работы генетических алгоритмов, такие как вероятность мутации и т.д.

2. Результаты работы каждого частного классификатора, состоящие из набора опорных векторов $SV_1, SV_2, \ldots, SV_b$ объединяются для создания обобщенного набора SV из ℓ объектов ($\ell \leq s$, где s – размер учебного набора).

3. Создается поднабор SV^+, объекты которого(L) выделяются из общего набор опорных векторов ($L \leq \ell$). В данный поднабор относятся классификаторы, которые успешно справились с определением реальных классов опорных векторов. Создание такого набора позволяет отсеять наименее удачные классификаторы, для повышения эффективности обучения классификатора на втором этапе. Также создается поднабор SV^-, в который попадают наименее удачные классификаторы ($\ell \leq L$). Они потребуются для тестирования.

4. Итоговый классификатор создается с помощью бустинга биоинспирированного алгоритма, в данном случае PSO-алгоритма. Он использует набор удачных решений $_{SV}{}^+$ для обучения, а набор SV^- для тестирования. Определяются значения параметров итогового классификатора.

5. Применение найденных параметров для создания итогового классификатора.

6. Для объектов, не попавших в какой-либо поднабор, выполняется докласс-сификация.

7. Оценка результатов работы алгоритма. Выполняется оценка качества, оценка затраченного времени, удовлетворение критериев.

Таким образом в результате работы, получаем классификатор, обеспечивающих повышенную точность классификации. Точность работы двухуровневого классификатора выше точности отдельных частных SVM классификаторов, используемый независимо. Также использование ансамбля алгоритмов, позволит значительно сократить время обучения алгоритма PSO, и соответственно время классификации.

5. Экспериментальные исследования. Оценка эффективности разработанного алгоритма проводилась на реальных наборах данных. Были использованы наборы German, Australian, Firms – наборы для кредитной оценки, наборы WDBC и Hearts – данные диагностики в медицине, а также наборы Spam и МОТП12. Полученные результаты на заданных наборах сравнивались с результатами классификации, проводимой в пакете SPSS Modeler (интенсивно используется за рубежом), а также в статистическом пакете STATICTICA StatSoft (популярен в России).

Ниже представлены диаграмма сравнения точности разработанного классификатора с вышеописанными пакетами. На рис. 3 представлено сравнение точности с использованием радиальной базисной функции Гаусса. Параметры функции ядра были заданы по умолчанию. Параметры регуляции – по умолчанию.

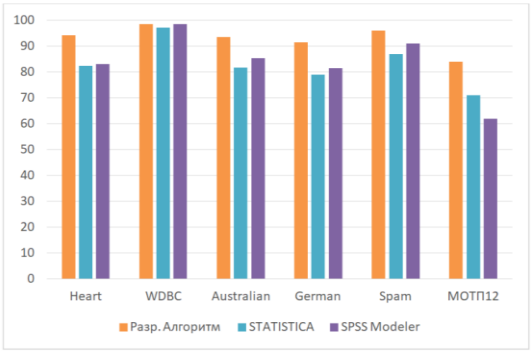

Рис. 3. Диаграмма точности классификаторов для радиальной базисной функции Гаусса

На рис. 4 представлено сравнение точности классификаторов для полиномиальной неоднородной функции ядра.

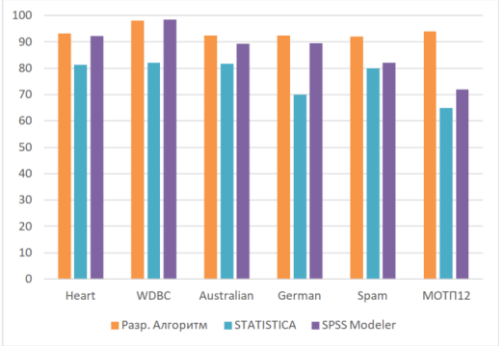

Рис. 4. Диаграмма точности классификаторов для полиномиальной неоднородной функции ядра

Таким образом можно сделать промежуточный вывод, что разработанный классификатор не уступает, а зачастую, и превосходит по точности, классификаторы, созданные в таких пакетах как SPSS Modeler и STATISTICA StatSoft. Однако, наряду с точностью алгоритма, возросло и время выполнения. Алгоритм долго обучается на неэффективном учебном наборе. Для решения этой проблемы предлагается использовать ансамбль алгоритмов.

Двухуровневый классификатор

По результатам анализа полученных данных, был сделан вывод, что по мере роста объема входных данных, увеличивается время поиска оптимальных значений параметров алгоритма роя частиц. Для решения данной проблемы предложен двухуровневый классификатор.

Эффективность данного классификатора можно увидеть на примере классификации набора МОТП12 (рис. 5). Данный набор считается сложноорганизованным и сложно-классифицируемым. Приведенные выше классификаторы справляются с классификацией за продолжительное время, несмотря на небольшой объем (400 эл. и 2 хар.) В процессе работы алгоритма, на первом уровне двухуровневого классификатора, обучаются 10 простых классификаторов, в которых используются различные значения параметров. Для каждого отдельного классификатора, затрачиваемое время не превышает 2 секунд. Эти частные классификаторы отобрали 224 опорных вектора, большинство из которых были классифицированы верно. Они в дальнейшем составили учебную выборку SV^+. Остальные объекты составили тестовую выборку SV^-.

Рис. 5. Диаграмма итоговой точности классификаторов

На основе учебной выборки, модифицированный алгоритм роя частиц находит субоптимальные значения в 1.5 раза быстрее, чем при использовании полного набора данных. Итоговый классификатор верно классифицировал все объекты тестовой выборки SV^-.

Точность решений частных классификаторов составила 80–92 %, а точность итогового классификатора, в среднем 96%. Таким образом использование двухуровневого классификатора повышает точность итоговой классификации, по сравнению с частными классификаторами. Помимо этого, сокращение учебного набора с 400 до 224 сократило время поиска более чем в 2 раза. Таким образом двухуровневый классификатор ускоряет процесс классификации, по сравнению с модифицированным PSO алгоритмом в 2 раза.

Таким образом использование двухуровневого классификатора повышает точность итоговой классификации, по сравнению с частными классификаторами. Помимо этого, сокращение учебного набора с 400 до 224 сократило время поиска более чем в 2 раза. Таким образом двухуровневый классификатор ускоряет процесс классификации, по сравнению с модифицированным PSO алгоритмом в 2 раза.

Заключение. Целью работы являлось повышение эффективности алгоритмов, решающих задачу классификации с применением биоинспирированных алгоритмов. Для достижения данной цели был проведен аналитический обзор существующих подходов к решению задачи бинарной классификации. В процессе анализа был выбран метод опорных векторов. Была изучена возможность применения генетических и биоинспирированных эвристик в задаче классификации. Была приведена постановка задачи классификации, а также задача бинарной классификации на основе алгоритма опорных векторов (SVM-алгоритма).

Для решения данной задачи был применен алгоритм роя частиц(PSO). Данный алгоритм в контексте задачи поиска субоптимальных значений параметров классификатора способен обеспечить высокое качество классификации. Модификацией алгоритма является динамическое изменение значений координат, которые отвечают за тип функции ядра. Данная доработка позволяет значительно снизить затрачиваемое время разработки классификатора.

Была приведена структура двухуровневого классификатора. На первом уровне данного классификатора, формируется ансамбль простых классификаторов которые формируют учебную выборку, которая, в дальнейшем используется PSO алгоритмом на втором этапе. Такой подход позволяет значительно уменьшить временные затраты, а также повысить качество получаемых решений.

По итогам экспериментальных исследований был сделан вывод что приведенный алгоритм роя частиц в задаче классификации превосходит по качеству классификации существующие решения, полученные в пакетах STATISTICA StatSoft и IBM SPSS Modeler.

Экспериментальные исследования двухуровневого классификатора показали, что он способен принимать решения высокой точности в многомерных, сложноорганизованных массивах данных. За счет использования ансамбля алгоритмов на первом уровне, была уменьшена и качественна улучшена учебная выборка для алгоритма роя частиц на втором уровне. Данный двухуровневый классификатор позволяет сократить затраты времени на поиск субоптимальных значений параметров классификатора, гарантируя высокую точность решений.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Воронцов, К.В.* Обзор современных исследований по проблеме качества обучения алгоритмов // Таврический вестник информатики и математики. – 2004. – № 1. – С. 5-24.
2. *Курейчик В.М., Курейчик В.В., Родзин С.И., Гладков Л.А.* Основы теории эволюционных вычислений. – Ростов-на-Дону: ЮФУ, 2010.

3. *Карпенко А.П.* Популярные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов // Информационные технологии. – 2012. – № 7 (Приложение). – С. 1-32.
4. *Родзин С.И., Курейчик В.В.* Состояние, проблемы и перспективы развития биоэвристик // Программные системы и вычислительные методы. – 2016. – № 2. – С. 158-172.
5. *Курейчик В.М., Запорожец Д.Ю.* Роевой алгоритм в задачах оптимизации // Известия ЮФУ. Технические науки. – 2010. – № 7 (108). – С. 28-32.
6. *Гладков Л.А., Курейчик В.В., Курейчик В.М.* Генетические алгоритмы. – М.: Физматлит, 2010. – 368 с.
7. *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учеб. пособие. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. – 446 с.
8. *Clarke, B., Fokoue E., Zhang H.H*. Principles and Theory for Data Mining and Machine Learning. – Springer Science, LLC, 2009. – 781 p.
9. *Graf H.P., Cosatto E., Bottou L., Durdanovic I., Vapnik V.* Parallel Support Vector Machines: The Cascade SVM // Advances in Neural Information Processing Systems. – 2004. – Vol. 17. – P. 521-528.
10. *Priyadarshini A., Agarwal S.* A Map Reduce based Support Vector Machine for Big Data Classification // International Journal of Database Theory and Application. – 2015. – Vol. 8, No. 5. – P. 77-98.
11. *Demidova L., Sokolova Yu.* A Novel SVM-*k*NN Technique for Data Classification // 6-th Mediterranean Conference on Embedded Computing (MECO' 2017). – 2017. – P. 459-462.
12. *Zhang H., Berg A.C., Maire M., Malik J.* SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition, Proceedings // 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. – 2016. – Vol. 2. – P. 2126-2136.
13. *Воронцов К.В.* Комбинаторные оценки качества обучения по прецедентам // Докл. РАН. – 2004. – Т. 394, № 2. – С. 175-178.
14. *Дьяконов А.Г.* Анализ данных, обучение по прецедентам, логические игры, системы WEKA, RapidMiner и MatLab (Практикум на ЭВМ кафедры математических методов прогнозирования): учеб. пособие. – М.: Издательский отдел факультета ВМК МГУ им. М.В. Ломоносова, 2010. – 278 с.
15. *Lean Yu, Shouyang Wang, Kin Keung Lai, Ligang Zhou*. Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines. – Springer-Verlag Berlin Heidelberg, 2008. – 244 p.
16. *Вьюгин В.В.* Математические методы теории машинного обучения и прогнозирования: учеб. пособие. – М.: МФТИ, 2013. – 379 с.
17. *Демидова Л.А., Соколова Ю.С.* Аспекты применения алгоритма роя частиц в задаче разработки SVM-классификатора // Вестник Рязанского государственного радиотехнического университета. – 2015. – № 3 (53). – С. 84-92.
18. *Duggal P.S., Paul S., Tiwari P.* Analytics for the Quality of Fertility Data using Particle Swarm Optimization // International Journal of Bio-Science and Bio-Technology. – 2015. – Vol. 7, No. 1. – P. 39-50.
19. *Курейчик В.В., Курейчик В.М., Сороколетов П.В.* Анализ и обзор моделей эволюции // Известия Российской академии наук. Теория и системы управления. – 2007. – № 5. – С. 114-126.
20. *Chapelle O., Vapnik V., Bousquet O., Mukherjee S*. Choosing Multiple Parameters for Support Vector Machine // Machine Learning. – 2002. – Vol. 46. – P. 131-159.

REFERENCES

1. *Vorontsov, K.V.* Obzor sovremennykh issledovaniy po probleme kachestva obucheniya algoritmov [Review of modern research on the problem of the quality of learning algorithms], *Tavricheskiy vestnik informatiki i matematiki* [Tauride Bulletin of Informatics and mathematics], 2004, No. 1, pp. 5-24.
2. *Kureychik V.M., Kureychik V.V., Rodzin S.I., Gladkov L.A.* Osnovy teorii evolyutsionnykh vychisleniy [Fundamentals of the theory of evolutionary computing]. Rostov-on-Don: YuFU, 2010.
3. *Karpenko A.P.* Populyarnye algoritmy global'noy poiskovoy optimizatsii. Obzor novykh i maloizvestnykh algoritmov [Popular algorithms for global search engine optimization. Review of new and little-known algorithms], *Informatsionnye tekhnologii* [Information technologies], 2012, No. 7 (Appendix), pp. 1-32.

4. *Rodzin S.I., Kureychik V.V.* Sostoyanie, problemy i perspektivy razvitiya bioevristik [State, problems and prospects for the development of bioheuristics], *Programmnye sistemy i vychislitel'nye metody* [Software systems and computational methods], 2016, No. 2, pp. 158-172.
5. *Kureychik V.M., Zaporozhets D.Yu.* Roevoy algoritm v zadachakh optimizatsii [Swarm algorithm in optimization problems], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2010, No. 7 (108), pp. 28-32.
6. *Gladkov L.A., Kureychik V.V., Kureychik V.M.* Geneticheskie algoritmy [Genetic algorithms]. Moscow: Fizmatlit, 2010, 368 p.
7. *Karpenko A.P.* Sovremennye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennye prirodoy: ucheb. posobie [Modern search optimization algorithms. Algorithms inspired by nature: tutorial]. Moscow: Izd-vo MGTU im. N.E. Baumana, 2014, 446 p.
8. *Clarke, B., Fokoue E., Zhang H.H*. Principles and Theory for Data Mining and Machine Learning. Springer Science, LLC, 2009, 781 p.
9. *Graf H.P., Cosatto E., Bottou L., Durdanovic I., Vapnik V.* Parallel Support Vector Machines: The Cascade SVM, *Advances in Neural Information Processing Systems*, 2004, Vol. 17, pp. 521-528.
10. *Priyadarshini A., Agarwal S.* A Map Reduce based Support Vector Machine for Big Data Classification, *International Journal of Database Theory and Application*, 2015, Vol. 8, No. 5, pp. 77-98.
11. *Demidova L., Sokolova Yu.* A Novel SVM-*k*NN Technique for Data Classification, *6-th Mediterranean Conference on Embedded Computing (MECO' 2017),* 2017, pp. 459-462.
12. *Zhang H., Berg A.C., Maire M., Malik J.* SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition, Proceedings, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, Vol. 2, pp. 2126-2136.
13. *Vorontsov K.V.* Kombinatornye otsenki kachestva obucheniya po pretsedentam [Combinatorial evaluations of the quality of training based on precedents], *Dokl. RAN* [Reports of the Russian Academy of Sciences], 2004, Vol. 394, No. 2, pp. 175-178.
14. *D'yakonov A.G.* Analiz dannykh, obuchenie po pretsedentam, logicheskie igry, sistemy WEKA, RapidMiner i MatLab (Praktikum na EVM kafedry matematicheskikh metodov prognozirovaniya): ucheb. posobie [Data analysis, case studies, logic games, WEKA, RapidMiner and MatLab systems (computer Workshop of the Department of mathematical forecasting methods): textbook]. Moscow: Izdatel'skiy otdel fakul'teta VMK MGU im. M.V. Lomonosova, 2010, 278 p.
15. *Lean Yu, Shouyang Wang, Kin Keung Lai, Ligang Zhou*. Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines. Springer-Verlag Berlin Heidelberg, 2008, 244 p.
16. *V'yugin V.V.* Matematicheskie metody teorii mashinnogo obucheniya i prognozirovaniya: ucheb. posobie [Mathematical methods of the theory of machine learning and forecasting: textbook]. Moscow: MFTI, 2013, 379 p.
17. *Demidova L.A., Sokolova Yu.S.* Aspekty primeneniya algoritma roya chastits v zadache razrabotki SVM-klassifikatora [Aspects of applying the particle swarm algorithm to the problem of developing an SVM classifier], *Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta* [Bulletin of the Ryazan state radio engineering University], 2015, No. 3 (53), pp. 84-92.
18. *Duggal P.S., Paul S., Tiwari P.* Analytics for the Quality of Fertility Data using Particle Swarm Optimization, *International Journal of Bio-Science and Bio-Technology*, 2015, Vol. 7, No. 1, pp. 39-50.
19. *Kureychik V.V., Kureychik V.M., Sorokoletov P.V.* Analiz i obzor modeley evolyutsii [Analysis and review of evolution models], *Izvestiya Rossiyskoy akademii nauk. Teoriya i sistemy upravleniya* [Proceedings of the Russian Academy of Sciences. Theory and control systems], 2007, No. 5, pp. 114-126.
20. *Chapelle O., Vapnik V., Bousquet O., Mukherjee S*. Choosing Multiple Parameters for Support Vector Machine, *Machine Learning*, 2002, Vol. 46, pp. 131-159.

Статью рекомендовал к опубликованию к.т.н. С.Г. Буланов.

Балабанов Дмитрий Валерьевич – Южный федеральный университет; e-mail: dbalabanov@sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371651; кафедра систем автоматизированного проектирования; аспирант.

Ковтун Антон Владиславович – e-mail: anton.kovtun93@gmail.com; кафедра систем автоматизированного проектирования; аспирант.

Кравченко Юрий Алексеевич – e-mail: yakravchenko@sfedu.ru; кафедра систем автоматизированного проектирования; доцент.

Balabanov Dmitry Valerievich – Southern Federal University; e-mail: dbalabanov@sfedu.ru; 44, Nekrasovskiy lane, Taganrog, 347928, Russia; phone: +78634371651; the department of computer aided design; graduate student.

Kovtun Anton Vladislavovich – e-mail: anton.kovtun93@gmail.com; the department of computer aided design; graduate student.

Kravchenko Yury Alekseevich – e-mail: yakravchenko@sfedu.ru; the department of computer aided design; associate professor.

УДК 004.032.26 DOI 10.18522/2311-3103-2020-3-146-156

В.В. Бахчевников, В.А. Деркачев, А.Н. Бакуменко

СПОСОБ ИСПОЛЬЗОВАНИЯ СРЕДСТВ БЫСТРОГО ПРОТОТИПИРОВАНИЯ ДЛЯ РЕАЛИЗАЦИИ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ НА ПЛИС

Исследования в области искусственного интеллекта ведутся с возрастающим интересом с каждым годом. Области применения искусственного интеллекта довольно обширны: автоматизация, анализ большого объема данных, технологии умного дома, машинное зрение и т.д. Технологии искусственного интеллекта базируются на использовании искусственных нейронных сетей, имеющие в своей основе принципы нервной системы животных. При этом актуальным вопросом является реализация искусственных нейронных сетей на различных программно-аппаратных платформах: программируемых логических интегральных схемах (ПЛИС) типа FPGA (Field Programmable Gate Array), на интегральных схемах специального назначения (Application-Specific Integrated Circuit, ASIC), GPU, CPU и т.д. ПЛИС наилучшим образом проявляют себя в маломощных мобильных системах. ASIC демонстрируют наибольшую производительность с недостатком: высокая цена разработки. Проблема быстрого прототипирования проектов, основанных на использовании искусственных нейронных сетей, для ПЛИС привычными методами (с помощью HDL-языков, HDL-кодеров, графического программирования) заключается в том, что либо такой проект сложен и длителен в отладке (HDL-языки), либо не оптимален получающийся код (HDL-кодеры), либо высока длительность разработки проекта и сложность реконфигурации нейронной сети (графическое программирование). Поэтому в рамках данной работы рассматривается эффективный метод проектирования полносвязных и сверточноых нейронных сетей для их реализации на ПЛИС использованием пакета Xilinx System Generator for DSP и Matlab/Simulink. Генерируемые таким образом искусственные нейросети легко реконфигурируемы и позволяют решать следующие задачи: распознавание изображений, оптимальная фильтрация (например, для задач подповерхностной радиолокации).

Искусственный интеллект; искусственные нейронные сети; реализация на ПЛИС; сверточная нейросеть; метод проектирования.