УДК 004.8

МЕТОДЫ РОЕВОГО ИНТЕЛЛЕКТА ДЛЯ СОВЕРШЕНСТВОВАНИЯ ОБУЧЕНИЯ НЕЙРОННОЙ СЕТИ

В.А. ЧАСТИКОВА, Д.С. ОСТАПОВ

Кубанский государственный технологический университет,
350072, Российская Федерация, г. Краснодар, ул. Московская, 2;
электронная почта:, chastikova_va@mail.ru

Алгоритм обратного распространения ошибки – один из наиболее популярных алгоритмов обучения нейронной сети с учителем – относится к методам градиентного поиска. Несмотря на относительно высокую точность настройки весовых коэффициентов нейронов, данный алгоритм имеет ряд недостатков: паралич нейронной сети и вероятность нахождения локальных минимумов ошибки. Целью данного исследования является разработка алгоритма, осуществляющего более точную настройку весовых коэффициентов нейронной сети по сравнению с алгоритмом обратного распространения ошибки. В статье предложен гибридный алгоритм, основывающийся на методах роевого интеллекта, обратного распространения ошибки и элементах жадного алгоритма. Благодаря совместному применению алгоритмов градиентного спуска и роевого интеллекта точность анализа данных нейронной сетью увеличилась. Приведены результаты исследований и сравнительного анализа функции ошибок при обучении и тестировании нейронной сети.
Ключевые слова: нейронная сеть, алгоритм обратного распространения ошибки, роевой интеллект, обучение нейронной сети.

Многослойный персептрон – одна из наиболее популярных архитектур искусственных нейронных сетей (НС), использующих метод обучения с учителем, которая в большинстве случаев настраивается с помощью метода обратного распространения ошибки (АОРО)[5,7].

Рассматриваемый алгоритм относится к методам градиентного поиска, имеющим проблему нахождения локальных экстремумов[8,10]; АОРО имеет итерационную составляющую и заключается в следующем.

Пусть $e_j(n)$ - ошибка нейрона j выходного слоя на итерации n, которая определяется:

$e_j(n) = d_j(n) - y(n)$, где d(n) – результат, который должен быть получен на итерации n в идеальном случае, y(n) – фактически полученный сигнал.

Среднеквадратическая ошибка нейронов выходного слоя[6,10]:

$E(n) = \frac{1}{2}\sum_{i=1}^{m} e_i^2(n)$, где m – число нейронов в рассматриваемом слое.

Каждый нейрон j формирует выходной сигнал:

$$v_j(n) = \sum_{i=1}^{m} w_{j_i}(n) y_i(n)$$

$$y_j(n) = f\left(v_j(n)\right)$$

Для того чтобы найти экстремум функции среднеквадратической ошибки, необходимо вычислить градиент ∇f [6]

$$\frac{\partial E(n)}{\partial w_{j_i}(n)} = \frac{\partial E(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \cdot \frac{\partial v_j(n)}{\partial w_{j_i}(n)}$$

Пусть $\delta_j(n) = \frac{\partial E(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)}$

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1$$

$$\frac{\partial y_j(n)}{\partial v_j(n)} = f'\left(v_j(n)\right)$$

$$\frac{\partial v_j(n)}{\partial w_{j_i}(n)} = y_i(n)$$

После расчета частных производных для нейронов выходного слоя (ВС):

$$\delta_j(n) = -e_i(n) f'\left(v_j(n)\right)$$

Таким образом, для нейронов ВС

$$\frac{\partial E(n)}{\partial w_{j_i}(n)} = -e_j(n) f'\left(v_j(n)\right) y_i(n)$$

Каждый из весов будет корректироваться на величину $\Delta w_{j_i} = -\eta \frac{\partial E(n)}{\partial w_{j_i}(n)}$, где η – параметр скорости обучения. Δw_{j_i} примет вид:

$$\Delta w_{j_i} = \eta \delta_j(n) y_i(n)$$

Итак, для нейронов выходного слоя локальный градиент $\delta_j(n) = -e_i(n) f'\left(v_j(n)\right)$

Однако, для нейронов внутреннего слоя ошибка на выходе неизвестна, поэтому $\delta_j(n)$ можно представить следующим образом:

$$\delta_j(n) - -\frac{\partial E(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} - -\frac{\partial E(n)}{\partial y_j(n)} f'\left(v_j(n)\right)$$

В связи с тем, что $E(n) = \frac{1}{2}\sum_{k=1}^{m} e_k^2(n)$, то $\frac{\partial E(n)}{\partial y_j(n)} = \sum_{k=1}^{m} e_k \frac{\partial e_k(n)}{\partial y_j(n)} = \sum_{k=1}^{m} e_k \frac{\partial e_k(n)}{\partial v_k(n)} \frac{\partial v_k(n)}{\partial y_j(n)}$, где k – выходной нейрон.

Для нейрона k выходного слоя

$$e_k(n) = d_k(n) - f_k(v_k(n))$$

$$\frac{\partial e_k(n)}{\partial v_k(n)} = -f'_k(v_k(n))$$

Так как

$$v_k(n) = \sum_{l=1}^{m} w_{k_l}(n) y_l(n)$$, то

$$\frac{\partial v_k(n)}{\partial y_j(n)} = w_{k_j}(n)$$

Таким образом,

$$\frac{\partial E(n)}{\partial y_j(n)} = -\sum_{k=1}^{m} e_k f'_k(v_k(n)) w_{k_j}(n) = -\sum_{k=1}^{m} \delta_k(n) w_{k_j}(n)$$

Локальный градиент скрытого нейрона j равен:

$$\delta_j(n) = f'_j(v_j(n)) \sum_{k=1}^{m} \delta_k(n) w_{k_j}(n)$$

Если в качестве функции активации нейрона используется логическая функция

$$f_j(v_j(n)) = \frac{1}{1 + e^{-\alpha v_j(n)}}$$, где $\alpha > 0$, то

$$f'_j(v_j(n)) = \frac{\alpha e^{-\alpha v_j(n)}}{\left(1 + e^{-\alpha v_j(n)}\right)^2}$$

Так как $y_j(n) = f_j(v_j(n))$, то после преобразований $f'_j(v_j(n))$ примет вид:

$$f'_j(v_j(n)) = \alpha y_j(n)(1 - y_j(n))$$

Локальный градиент для произвольного нейрона j можно вычислить по формуле:

$$\delta_j(n) = \alpha y_j(n)\left(1 - y_j(n)\right) \sum_{k=1}^{m} \delta_k(n) w_{k_j}(n)$$

Вторая производная функции $f_j(v_j(n))$:

$$\frac{\partial^2 f_j(v_j(n))}{\partial y_j(n)} = \frac{\partial f'_j(v_j(n))}{\partial y_j(n)} = \alpha - 2y_j(n)$$

$\frac{\partial^2 f_j(v_j(n))}{\partial y_j(n)}$ не является строго положительной функцией на всей области определения, из чего следует, что функция $f'_j(v_j(n))$ не является монотонной. В связи с этим направление ∇f нередко отклоняется от направления к точке, являющейся глобальным экстремумом, что подтверждает наличие проблемы нахождения локального оптимума алгоритмом обратного распространения ошибки.

В отличие от градиентных методов оптимизации, алгоритмы роевого интеллекта (РИ), лишены недостатка, заключающегося в нахождении локального экстремума[2-4], поэтому для совершенствования процесса обучения нейронной сети предлагается использовать гибридный метод АОРО и РИ.

Структура гибридного алгоритма

На каждой итерации на нейронную сеть подаётся обучающий пример, и выполняется метод обратного распространения ошибки; после него при необходимости можно настроить веса нейронов любым методом РИ, который способен выполнять поиск глобального оптимума функции. Для того чтобы АОРО осуществлял только поиск более эффективных решений (для реализации выходов из локальных минимумов), а не пытался их искать самостоятельно, работа данного алгоритма должна выполняться при условии, что нейронная сеть дала правильный ответ и ошибка нейронов невелика. Обучение НС с использованием только методов роевого интеллекта выполнять нецелесообразно в связи с тем, что они обладают достаточной точностью, и коэффициенты нейронной сети будут подстраиваться наилучшим образом для примеров, которые были поданы последними, слабо учитывая обучающую выборку, поданную на НС несколькими итерациями ранее. По этой же причине в процессе работы гибридного алгоритма не рекомендуется использовать слишком большое число частиц в рое и значительное число итераций для подстройки каждого элемента входных данных.

Подобно обучению машины Больцмана[8,9], вероятность использования РИ с каждой итерацией должна уменьшаться ~exp(-E/KT), где E – текущий номер итерации, T – количество примеров в базе знаний, K – настраиваемый коэффициент.

В качестве метода роевого интеллекта в работе использован гибридный алгоритм, основанный на методах роя частиц, пчелиной колонии и алгоритма поиска косяком рыб[1].

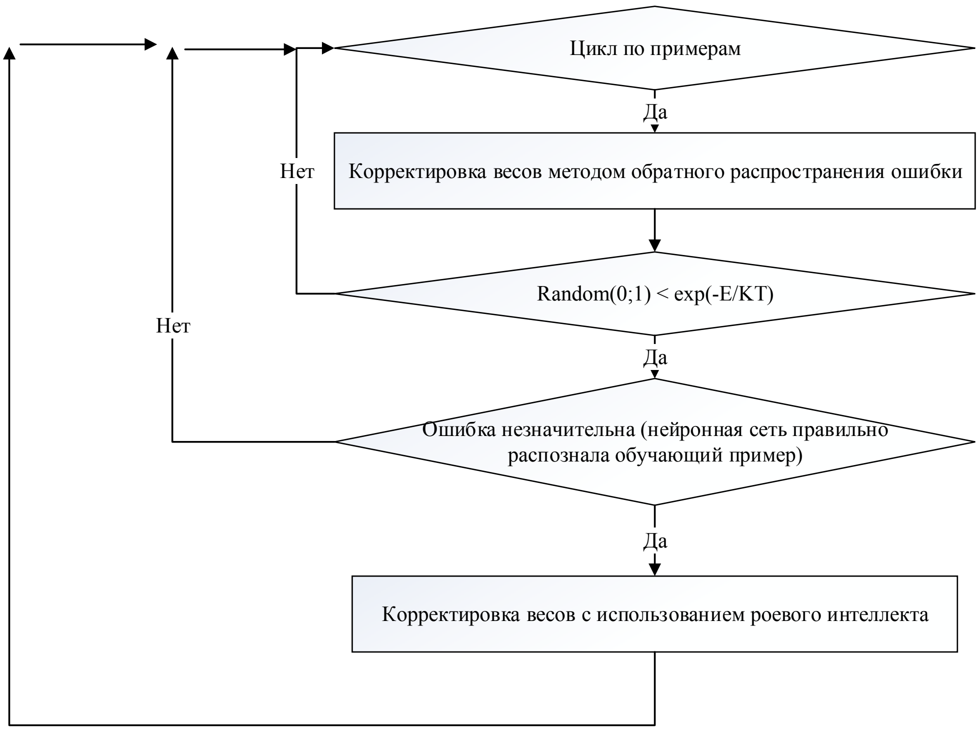

Рисунок 1 – Схема работы гибридного алгоритма

Среднюю ошибку нейронной сети можно представить следующим образом:

$\overline{E} = \frac{\sum_{i=1}^{n} |d_i - y_i|}{n}$, где y – сигнал, полученный на выходе нейрона i, d – сигнал, который должен выдать нейрон при отсутствии ошибки.

Целью алгоритма РИ является поиск минимума функции $\overline{E}$. Для этого на каждой итерации создаётся рой частиц; каждая частица является нейронной сетью, на которую подаётся пример. Одна из частиц должна иметь веса, полученные в результате работы АОРО; веса остальных частиц должны быть созданы случайным образом. Если минимальная ошибка среди частиц меньше её значения, полученной после работы АОРО, то нейронам присваиваются новые веса; в противном случае их значения остаются равными результатам, полученным после работы АОРО.

Результаты процесса обучения нейронной сети с использованием классического метода обратного распространения ошибки и гибридного алгоритма представлены на рисунке 2.

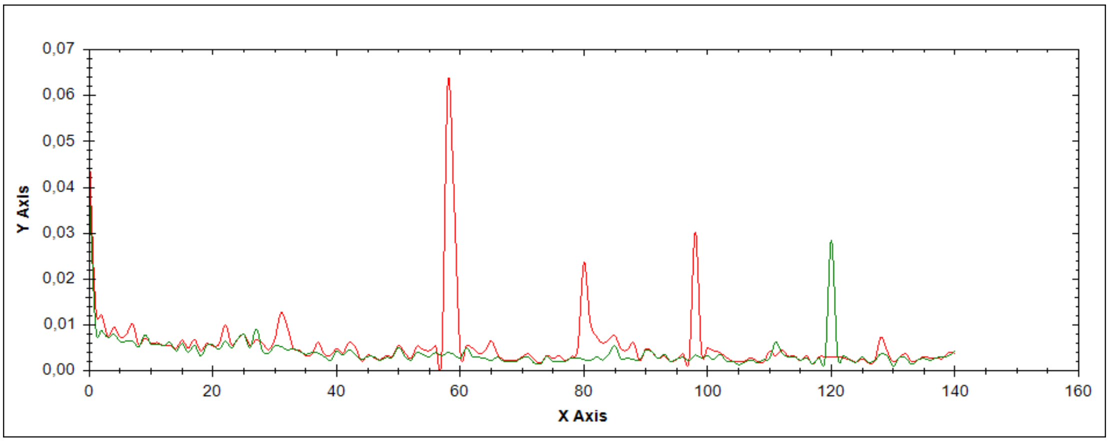

Рисунок 2 – Результаты процесса обучения НС

Красным цветом на рисунке 2 изображён график функции средней ошибки $\overline{E}$, полученной после работы классического АОРО; зелёным – значения $\overline{E}$, которые являются результатом выполнения гибридного алгоритма.

Пусть $D_i = E_i - E'_i$ – разность $\overline{E}$, полученных при разных способах обучения, где E_i и E'_I – ошибки, которые возникли после настройки весовых коэффициентов с использованием АОРО и гибридного алгоритмов соответственно.

График функции D представлен на рисунке 3.

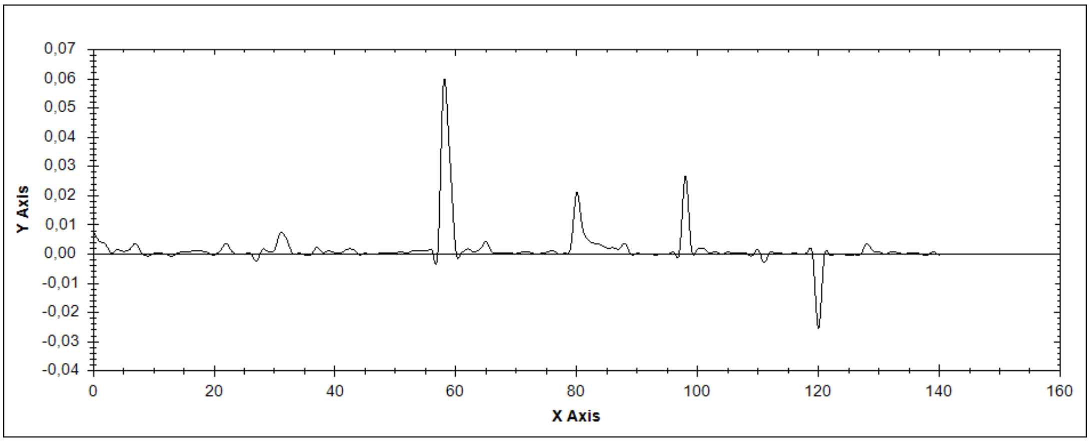

Рисунок 3 – График разности ошибок (функции D)

Как видно из рисунка 3, основная часть D=f(E,E') находится в верхней полуплоскости, из чего можно сделать вывод, что работа алгоритмов РИ положительно сказалась на процессе минимизации ошибки.

График функции ошибки, которая возникла в процессе тестирования работы нейронной сети, обученной с использованием АОРО и гибридного метода, представлен на рисунке 4.

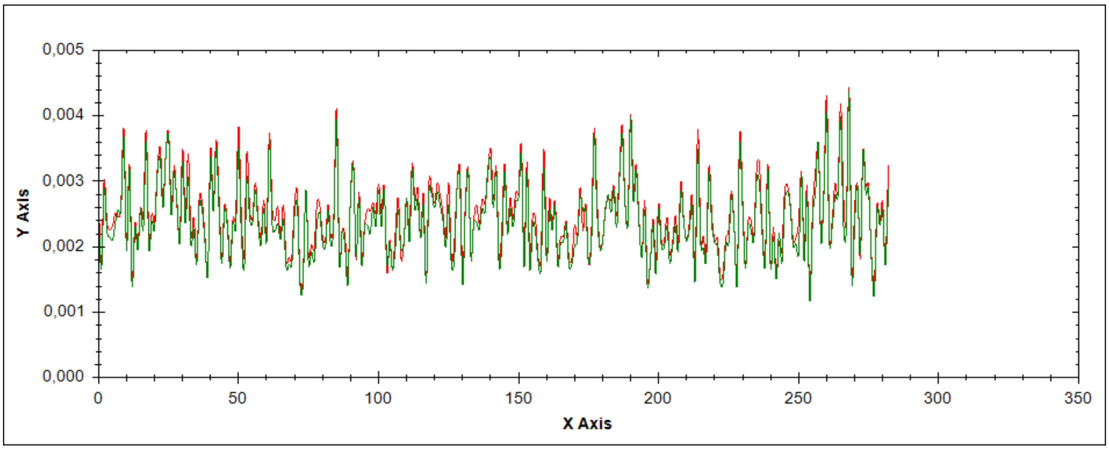

Рисунок 4 – График функции ошибки при тестировании

В связи с тем, что значения $\overline{E}$ очень малы, их графики находятся близко друг к другу, совместная декомпозиция значений $\overline{E}$ выглядит нечитабельно на рисунке 4; по этой причине целесообразно изобразить графически функцию разности ошибок при тестировании (рисунок 5).

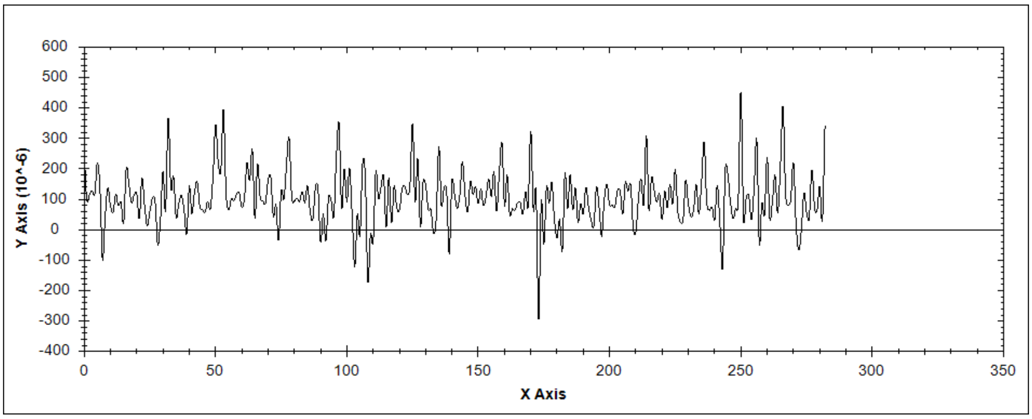

Рисунок 5 – График функции разности ошибок при тестировании

Как видно из рисунка 5, график функции разности ошибок при тестировании практически весь находится в верхней полуплоскости. Из этого можно сделать вывод, что применение методов РИ при обучении нейронной сети с использованием АОРО повышает качество настройки весовых коэффициентов нейронов.

Каждый ответ нейронной сети можно представить как x_A/μ_A, где x_A – полученный ответ, μ_A – вероятность того, что ответ правильный. Чем меньше ошибка, тем с большей точностью работает нейронная сеть, так как из двух результатов, полученных нейронной сетью x_A/μ_A и x_A/μ'_A, наилучшим будет тот, у которого функция принадлежности принимает наибольшее значение. С другой стороны, если рассматривать исключительно чёткие значения нейронной сети, средняя ошибка уменьшится на 2,5%.

ЛИТЕРАТУРА

1. Частикова В.А. Гибридный оптимизационный алгоритм грифов на основе механизмов роевого интеллекта / Частикова В.А., Остапов Д.С. // Научный журнал КубГАУ [Электронный ресурс]. - Краснодар: КубГАУ, 2014. - № 100. - Режим доступа: www.ej.kubagro.ru/2014/06/96/

2. Малыхина М.П., Частикова В.А., Власов К.А. Исследование эффективности работы модифицированного генетического алгоритма в задачах комбинаторики // Современные проблемы науки и образования. 2013. № 3. - URL: www.science-education.ru/109-9254.

3. Частиков А.П., Тотухов К.Е. Теоретические основы интеллектуальной симуляции промышленных роботов. Монография / Saarbrucken, 2013. - 111 с.

4. Карпенко А.П. Популяционные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов // Информационные технологии. 2012. № 7, с. 1-32.

5. Малыхина М.П., Бегман Ю.В. Нейросетевая экспертная система на основе прецедентов для решения проблем обслуживания абонентов сотовой сети // Известия высших учебных заведений. Северо-Кавказский регион. Серия: Технические науки. 2009. № 3. - с. 6-9.

6. Хайкин С. Нейронные сети. Полный курс. – 2-е изд. – М.: Издательский дом «Вильямс», 2006. – 1104 с.

7. Яхъяева Г.Э.. Нечеткие множества и нейронные сети – М.: Бином, 2008 – 315 с.

8. Ярушкина Н.Г. Основы теории нечётких и гибридных систем. Учебное пособие. - М.: Финансы и статистика, 2004. – 320 с.

9. Галушкин А.И. Нейронные сети: основы теории. – М.:Горячая линия-Телеком, 2010, 496 с.

10. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы. - М.– Горячая линия-Телеком, 2006, 384 с.

REFERENCES

1. Chastikova V.A. Gibridnyj optimizacionnyj algoritm grifov na osnove mehanizmov roevogo intellekta / Chastikova V.A., Ostapov D.S. // Nauchnyj zhurnal KubGAU [Jelektronnyj resurs]. - Krasnodar: KubGAU, 2014. - № 100. - Rezhim dostupa: www.ej.kubagro.ru/2014/06/96/

2. Malykhina M.P., Chastikova V.A., Vlasov K.A. Issledovanie effektivnosti raboty modificirovannogo geneticheskogo algoritma v zadachah kombinatoriki // Sovremennye problemy nauki i obrazovaniya. - 2013. - № 3; URL: www.science-education.ru/109-9254.

3. Chastikov A.P., Totukhov K.E. Teoreticheskie osnovi intellektualnoi simulyacii promyshlennyh robotov. Monografiya. / Saarbrucken, 2013. - 111 s.

4. A.P. Karpenko. Populjacionnye algoritmy global'noj poiskovoj optimizacii. Obzor novyh i maloizvestnyh algoritmov // Informacionnye tehnologii. 2012. № 7, s. 1-32.

5. Malykhina M.P., Begman Yu.V. Neirosetevaya ekspertnaya sistema na osnove precedentov dlya resheniya problem obslujivaniya abonentov sotovoi seti // Izvestiya vysshih uchebnyh zavedenii. Severo-Kavkazskii region. Seriya: Tehnicheskie nauki. 2009. № 3. – s. 6-9.

6. Hajkin S. Nejronnye seti. Polnyj kurs. –2-e izd. – M.: Izdatel'skij dom «Vil'jams», 2006. – 1104 s

7. G.Je. Jahjaeva. Nechetkie mnozhestva i nejronnye seti – M.: Binom, 2008 – 315 s.

8. N.G. Jarushkina. Osnovy teorii nechjotkih i gibridnyh sistem. Uchebnoe posobie. - M.: Finansy i statistika, 2004. – 320 s.

9. A.I. Galushkin. Nejronnye seti: osnovy teorii – M.:Gorjachaja linija-Telekom, 2010, 496 s.

10. D. Rutkovskaja. Nejronnye seti, geneticheskie algoritmy i nechetkie sistemy. - M.– Gorjachaja linija-Telekom, 2006, 384 s.

SWARM INTELLIGENCE METHODS TO IMPROVE THE LEARNING NEURAL NETWORK

V.A. CHASTIKOVA, D.S. OSTAPOV

Kuban State Technological University,
2, Moskovskaya st., Krasnodar, Russian Federation, 350072;
e-mail: chastikova_va@mail.ru

Back propagation algorithm - one of the most popular algorithms, neural network training with the teacher - is the method of gradient search. Despite the relatively high setting accuracy of weight coefficients of neurons, this algorithm has a number of drawbacks: paralysis of the neural network and the likelihood of finding local minima of the error. The aim of this study is to develop an algorithm for more precise tuning of weight coefficients of the neural network in comparison with the backpropagation algorithm. This paper proposes a hybrid algorithm, which is based on the methods of swarm intelligence, backpropagation and elements of the greedy algorithm. Thanks to the combined use of gradient descent algorithms and swarm intelligence accuracy of the data analysis neural network has increased. The results of research and comparative analysis of the error function in the training and testing the neural network.

Keywords: neural network algorithm of back-propagation, swarm intelligence, neural network training.