

НЕЧЕТКИЙ MLP-ПОДХОД ДЛЯ РАСПОЗНАВАНИЯ НЕЛИНЕЙНЫХ СИСТЕМ© 2019 г. **А. Р. МАРАХИМОВ, К. К. ХУДАЙБЕРГЕНОВ**

Аннотация. В рассмотрении задач принятия решения распознавание нелинейных систем играет огромную роль. Распознавание нелинейных систем с помощью многослойного персептрона (MLP), обученного по алгоритму обратного распространения, становится значительно более сложным с увеличением количества входных данных, слоев, узлов и количества итераций в процессе вычисления. В этой работе мы предприняли попытку использования нечеткого MLP и его обучающего алгоритма для распознавания нелинейных систем. Предложили подход нечеткого MLP и его обучающего алгоритма, который позволяет ускорить процесс обучения, превышающего скорость такового в случае классического MLP. Результаты показывают значительное упрощение при поиске оптимальных параметров для нейронной нечеткой модели в сравнении с классическим MLP. Также было проведено сравнение показателей работы обучения классического MLP и предложенной нечеткой MLP-модели. Нами были проанализированы временная и пространственная сложности алгоритма. Также мы выяснили, что серьезно сократилось количество моментов, а показатели работы выросли в сравнении с классическим MLP.

ОГЛАВЛЕНИЕ

1. Введение	44
2. Архитектура MLP	45
3. Нечеткие MLP для распознавания нелинейных систем	46
4. Детали эксперимента и результаты	48
5. Заключение	49
Список литературы	51

1. ВВЕДЕНИЕ

Распознавание нелинейных закономерностей, т. е. построение их моделей по результатам исследований, является одной из важных задач в технике, экономике, медицине и прочих прикладных областях [5]. В работах [2, 13] был описан метод распознавания нелинейных закономерностей с использованием нечеткой базы знаний. Распознавание нелинейных закономерностей с использованием нечеткой базы знаний — это конструкция нечеткой базы знаний, которая, грубо говоря, представляет собой отношение между входными и выходными данными, используя лингвистические правила «ЕСЛИ—ТО» [3]. Эти правила были выработаны экспертами (также могут быть получены в результате выделения нечеткого знания из экспериментальных данных). Затем параметрическое распознавание переходит в изучение закономерностей путем нахождения таких весов лингвистических правил и таких функций принадлежности от нечетких параметров, что колебания результатов моделирования обучающих образов сводятся к минимуму.

В последние годы был достигнут существенный прогресс в области искусственных нейронных сетей (ANN). Упреждающие многослойные персептроны (MLP) наиболее широко изучены в работах [5, 6]. Комбинированная техника нечеткой базы знаний и нейронных сетей, названная нечеткими нейросетями, применяется для моделей в реальных задачах, например, при постановке медицинского диагноза [11]. В частности, данная техника была применена для задач распознавания образов. Новая модель была разработана А. П. Ротштейном и С. Д. Штовбой [13] для распознавания нелинейных закономерностей с нечеткими обучающими образами. Это управляемая нейросеть, названная нечеткой нейросетью и имеющая один скрытый слой в своей архитектуре. Кроме того, концепция нечеткого MLP была представлена в работах [7, 9, 10], в которых был

успешно применен метод для распознавания многомерных закономерностей. Эта же работа сосредотачивает свое внимание на развитии нечеткой MLP-модели для распознавания многомерных закономерностей с различной архитектурой. Работа этой модели сравнивается с работой классической MLP-модели, разработанной для тех же задач. Следует отметить, что и классическая, и нечеткая MLP-нейросети построены на основе обучения по алгоритму обратного распространения. Проблемы, освещенные в этой работе, также относятся и к другим нечетким нейросетевым моделям. Далее опишем общее содержание нашей работы: сводка предварительных сведений содержится в разделе 2, описывающем концепцию классического MLP; раздел 3 знакомит читателя с ранее предложенной нечеткой MLP-моделью и соответствующим алгоритмом; в разделе 4 представлены детальные вычислительные результаты; выводы содержатся в разделе 5. В самом конце работы находится список литературы, относящейся к данной работе.

2. АРХИТЕКТУРА MLP

Архитектура многослойного персептрона состоит из нескольких слоев нейронов [8, 16]. Входные данные подаются на первый слой (входной слой-0), после чего эти входные данные передаются на входы слоя-1. В слое-0 не проводится никаких вычислений, и его можно считать сенсорным слоем. Последний слой — выходной — выводит обработанные данные. Количество скрытых слоев, находящихся между входным и выходным слоями, может быть увеличено или, наоборот, уменьшено в зависимости от задачи, для которой предназначается модель. Если строить сеть с большим количеством слоев, то после ее обучения модель будет построена чрезвычайно точно. Однако такие модели считаются переобученными и непригодными для использования. Мы же рассмотрим сеть с 3 слоями: входным слоем, скрытым слоем и выходным слоем (рис. 1).

Следующие формулы описывают структуру архитектуры многослойного персептрона, изображенного на рис. 1:

$$net_j = \sum_{i=1}^n x_i w_{ji}^{(1)} + b_j, \tag{2.1}$$

где $w_{ji}^{(1)}$ — веса входного слоя, b_j — смещения входного слоя,

$$z_j = f(net_j), \tag{2.2}$$

$$net^* = \sum_{j=1}^m z_j w_j^{(2)} + b_0, \tag{2.3}$$

где $w_{ji}^{(2)}$ — веса выходного слоя, b_0 — смещения выходного слоя,

$$y = f(net^*) \tag{2.4}$$

$$\Delta w_j^{(2)} = \eta E f(net^*) y, \quad \Delta b_0 = \eta E f(net^*) \tag{2.5}$$

$$\Delta w_{ji}^{(1)} = \eta E f'(net_j) x^{(i)} f'(net^*) w_j^{(2)}, \quad \Delta b_i = \eta E f'(net_j) f'(net^*) w_j^{(2)} \tag{2.6}$$

где $i = \overline{1, n}$, $j = \overline{1, m}$, $f(\cdot)$ — одна из функций активации. Для многослойного персептрона мы используем функцию активации сигмоидального типа:

$$f(x) = \frac{1}{1 + \exp^{-x}}.$$

Псевдо-код для обучения MLP:

Инициализация

веса $w(1), w(2)$

смещения b, b_0 ,

скорость обучения

момент $t = 0$

Задание среднеквадратичной ошибки $MSE=0$ и критерия сходимости

do

for размера итерационного образа

// прямое распространение

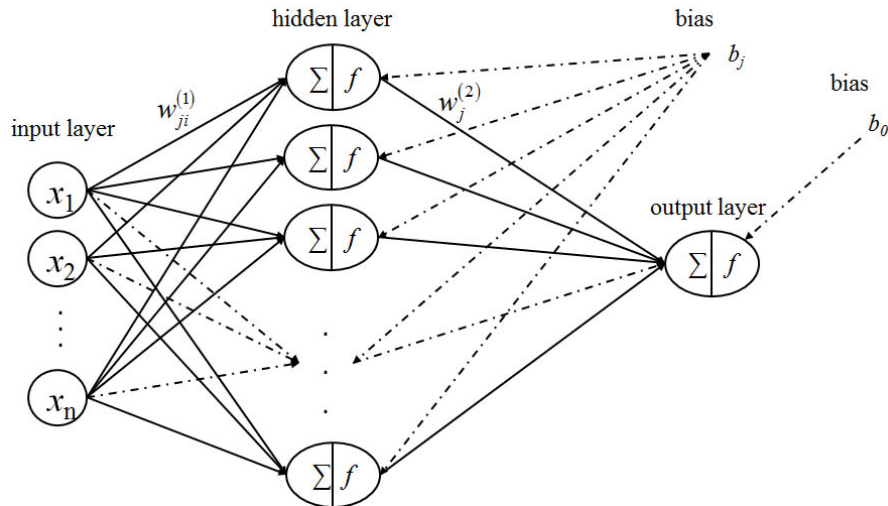


Рис. 1. Архитектура многослойного персептрона

$x \leftarrow$ произвольно выбирается обучающий образ
 for каждого скрытого нейрона
 Вычисление net_j (формула (2.1))
 Вычисление активации z_j (формула (2.2))
 end for
 Вычисление net^* (формула (2.3))
 Вычисление активации net^* (формула (2.4))
 Расчет ошибки, $ERROR = Target - y$;
 // обратное распространение ошибки для обновления весов и смещений
 Вычисление $\Delta w_j^{(2)}$ и обновление выходных весов (формула (2.5))
 Вычисление Δb_0 и обновление выходного смещения (формула (2.5))
 для каждого скрытого нейрона
 Вычисление $\Delta w_{ji}^{(1)}$ и обновление входных весов (формула (2.6))
 Вычисление Δb_i и обновление входных смещений (формула (2.6))
 end for
 end for
 while (критерий окончания)
 где $Target$ — это соответствующая цель.

3. Нечеткие MLP для распознавания нелинейных систем

Рассматриваемая нами архитектура нечеткого MLP довольно схожа с архитектурой классического MLP, и состоит из так называемых нечетких (*fuzzy*) принадлежностей наподобие фаззификации, дефаззификации и многослойного персептрона (рис. 3). В этой модели мы используем колоколообразную функцию принадлежности (рис. 2). Следует отметить, что мы рассматриваем функцию принадлежности именно такого типа в силу ее простоты. Однако для архитектуры нечеткого MLP можно применять и другие типы функций принадлежности.

3.1. Колоколообразная функция принадлежности. Мы используем колоколообразную функцию принадлежности, как показано ниже. Главными преимуществами функции принадлежности данного типа являются простота и удобство в корректировке.

$$\mu(x; a, c) = \frac{1}{1 + \left(\frac{x-c}{a}\right)^2}$$

где a и c — заданные параметры: a — это коэффициент концентрационного расширения функции, а c — это координата максимума функции (рис. 2).

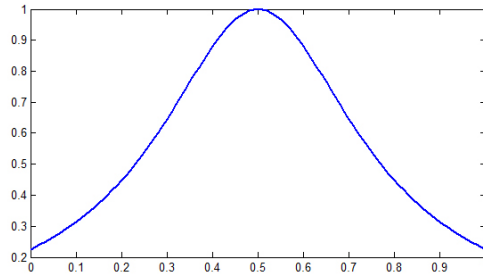


Рис. 2. График колоколообразной функции принадлежности

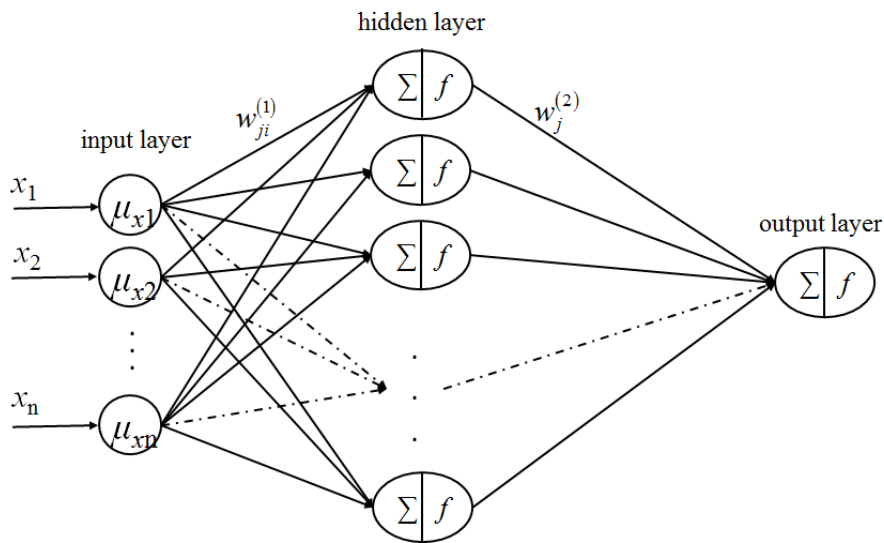


Рис. 3. Архитектура нечеткого MLP

3.2. Архитектура сети нечеткого MLP. Предлагаемая архитектура сходна с архитектурой классического многослойного персептрона; она показана на рис. 3. Процесс обучения нечеткой нейросети схож с процедурой обучения традиционных нейросетей, использующих правило «обратного распространения». Обучение этому алгоритму также состоит из двух этапов — прямого и обратного. Ввиду этого, здесь мы приводим только формулы, относящиеся к нечеткому MLP, в то время как остальные формулы содержатся в (2.1)–(2.6).

Степени принадлежности входных значений вычисляются по формуле (3.1):

$$\mu^j(x_i) = \frac{1}{1 + \left(\frac{x_i - c_i^j}{a_i^j}\right)^2}, \quad i = \overline{1, n}, \quad j = \overline{1, m}. \quad (3.1)$$

Таким образом, коэффициенты a_i и c_i обновляются по дельта-правилу (Δ -правило) [13]. На выходе ядро метода гравитационной дефаззификации состоит в нахождении четкого значения. Считается, что этот метод дефаззификации обеспечивает наибольшую точность и скорость регулировки нечеткой модели [15].

Псевдо-код для обучения нечеткого MLP:

Инициализация

веса $w(1), w(2)$

скорость обучения α

момент $t = 0$

Задание среднеквадратичной ошибки $MSE = 0$ и критерия сходимости ε

фаззификация (введение нечеткости) исходного образа обучения с помощью колоколообразной ФП

do

```

// прямое распространение
for размера итерационного образа ←
x ← произвольно выбирается обучающий образ
для каждого скрытого нейрона
Вычисление  $net_j$  (формула (2.1))
Вычисление активации  $z_j$  (формула (2.2))
end for
Вычисление  $net^*$  (формула (2.3))
Вычисление активации  $net^*$  (формула (2.4))
y = дефаззификация нечеткого умозаключения  $\mu_y(y)$ 
Расчет ошибки,  $ERROR = Target - y$ ;
// обратное распространение ошибки для обновления весов
Вычисление  $\Delta w_j^{(2)}$  и обновление выходных весов (формула (2.5))
для каждого скрытого нейрона
Вычисление  $\Delta w_{ji}^{(1)}$  и обновление входных весов (формула (2.6))
end for
для каждого из параметров функции принадлежности
Вычисление  $\Delta a_i$  и обновление коэффициента ФП
Вычисление  $\Delta c_i$  и обновление коэффициента ФП
end for
end for
while (критерий окончания)
где  $Target$  — это соответствующая цель.

```

Временная и пространственная сложности одинаковы для предложенных алгоритмов. В модели для нечеткого MLP используется одна вспомогательная матрица, в которой хранятся нечеткие входные данные (образы). Следует отметить, что эта область памяти не влияет на скорость обучения описанных алгоритмов, а для современных компьютеров пространство памяти ($O(n \times m)$) и вовсе не является проблемой. В обеих моделях размер входных нейронов равен n ; размер скрытых нейронов равен m , а размер выходных нейронов равен 1. Вычисление требует прямого и обратного распространения. В процессе прямого расчета вычисление ведется с входного (слой-0) к скрытому слою (слой-1) и требует $O(n \times m)$ вычислений. Следующее прямое вычисление идет из скрытого слоя к выходному (слой-2) и требует $O(m)$ времени. Схожим образом идет и процесс обратного распространения — вычисления от выходного к скрытому слою и от скрытого ко входному слою требуют $O(m)$ и $O(m \times n)$ времени, соответственно. И, наконец, чтобы получить выходное значение, требуется лишь $O(1)$ времени для обоих алгоритмов. Из этого следует, что временная сложность алгоритма может быть вычислена по следующей форме:

$$T(n, m) = 2O(n \times m) + 2O(m) + O(1),$$

следовательно,

$$T(n, m) = O(n \times m).$$

4. ДЕТАЛИ ЭКСПЕРИМЕНТА И РЕЗУЛЬТАТЫ

Мы рассматриваем объект с двумя входными $x_1, x_2 \in [-2, 2]$ и одной выходной y закономерностями, приведенными ниже:

$$y = x_1 \times e^{(-x_1^2 - x_2^2)} \quad (4.1)$$

Необходимо провести обучение, основанное на вышеизложенных моделях, синтезировать нечеткую модель и отрегулировать ее в соответствии с нечетким обучающим образом. Пригодность нечеткой модели проверяется в следующем критерии:

$$MSE = \frac{1}{2} \sum_{i=1}^n (y_i^r - y_i^d)^2$$

где y^r — соответствующий выход обучающего образа, а y^d — это желаемый выход модели. Сравним результаты распознавания с четким и нечетким обучающими образами. Обучающие образы

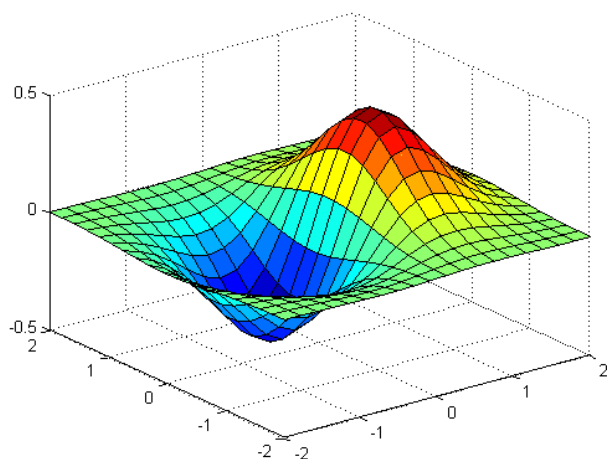


Рис. 4. Эталонная закономерность (4.1)

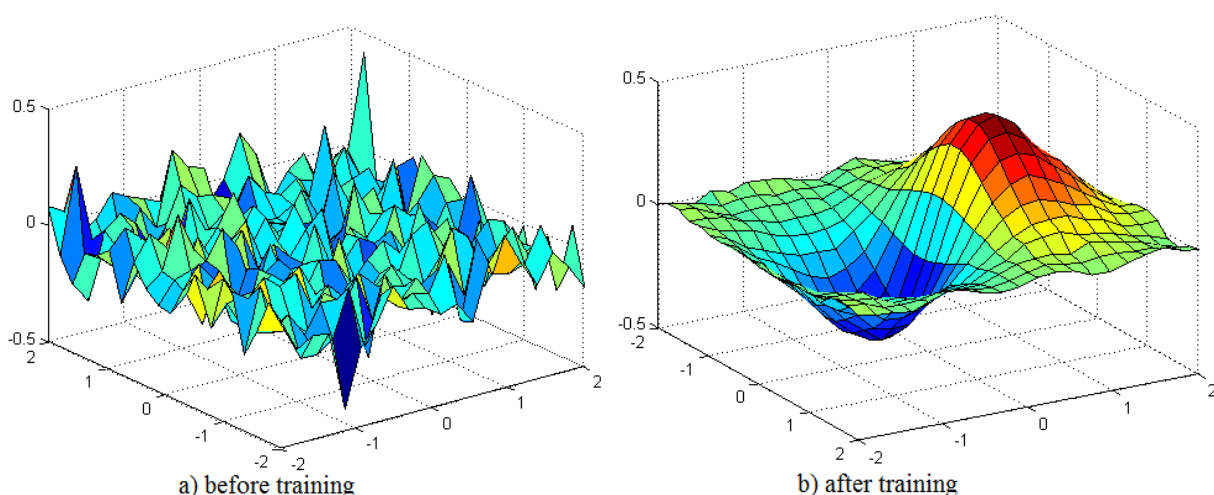


Рис. 5. Геометрические изображения объекта (4.1)

показаны формулой (4.1) и на рис. 4. Прежде чем обучать предложенную модель, ее веса должны быть произвольно определены. В связи с этим предварительно приведена исходная форма модели (рис. 5а). После обучения нейросети получаем модель, геометрическое изображение характеристик которой в сравнении с ее эталоном приведено на рис. 5б. Графики функций принадлежности для нечетких членов входных переменных до и после корректировки показаны на рис. 7, а значения их параметров a и c приведены в табл. 2 и табл. 3, соответственно.

Испытание четкой (рис. 6а) и нечеткой (рис. 6б) моделей указывает на приемлемое распознавание нелинейной закономерности (4.1). В табл. 1 показаны данные о сходимости для различных значений скорости обучения α . Из рис. 6б видно, что среднеквадратичная ошибка, полученная с помощью алгоритма для нечеткого MLP, достигает соответствующего минимума после 600–700 моментов.

Исходя из результатов эксперимента, можно заключить, что предложенная нечеткая модель обладает лучшей сходимостью, чем четкая модель.

5. ЗАКЛЮЧЕНИЕ

Был разработан нечеткий MLP для распознавания нелинейных систем. Входные данные фаззифицируются с помощью колоколообразной функции принадлежности, а затем подаются в нечеткую MLP модель. После дефаззификации выход предложенного нечеткого MLP дает четкое значение.

ТАБЛИЦА 1. Показатели работы моделей

α	Минимальная MSE		Время (сек)	
	MLP	Fuzzy-MLP	MLP	Fuzzy-MLP
0,30	1,0161	0,0234	26,71	24,10
0,40	1,0145	0,0207	25,74	22,42
0,45	1,0754	0,0159	25,97	21,53
0,50	1,2273	0,0268	23,68	20,04
0,55	1,3205	0,0272	23,55	20,19

ТАБЛИЦА 2. Параметры функций принадлежности для членов от переменной x_1

Terms		A_1	A_2	A_3	A_4	A_5	A_6
до обучения	a	0,3	0,3	0,3	0,3	0,3	0,3
	c	-1,9600	-1,2000	-0,4000	0,4000	1,2000	2,0000
после обучения	a	0,4203	0,2297	0,4620	0,4519	0,2225	0,4247
	c	-1,9947	-1,1667	-0,4678	0,4538	1,2048	1,9823

ТАБЛИЦА 3. Параметры функций принадлежности для членов от переменной x_2

Terms		B_1	B_2	B_3	B_4	B_5	B_6
до обучения	a	0,3	0,3	0,3	0,3	0,3	0,3
	c	-1,9600	-1,2000	-0,4000	0,4000	1,2000	2,0000
после обучения	a	0,2281	0,4192	0,1911	0,1907	0,4293	0,2350
	c	-1,9941	-0,6719	-0,0760	0,0801	0,6851	2,0225

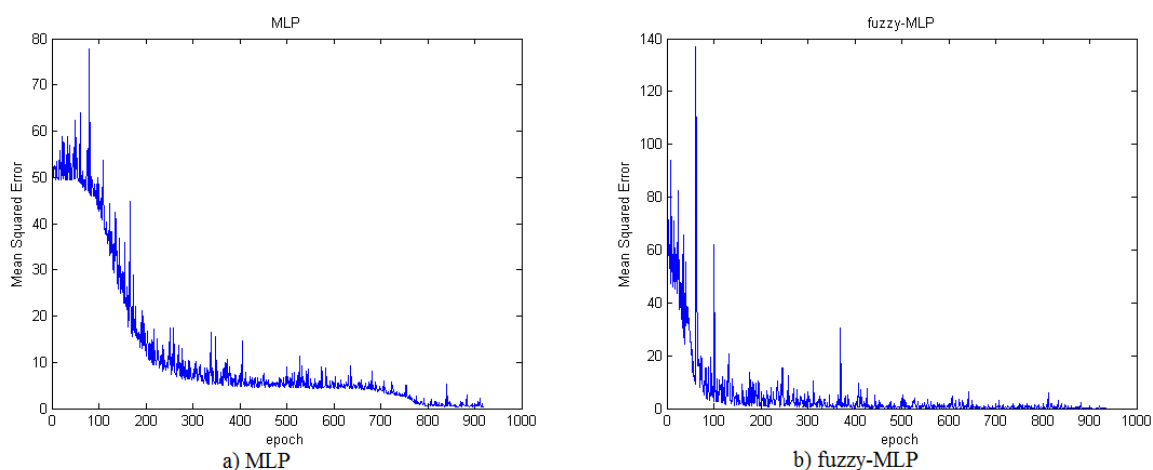


Рис. 6. Графики сходимости MLP (слева) и нечеткого MLP (справа)

Полученные результаты показывают, что предложенная модель сходится к своей минимальной среднеквадратичной ошибке (MSE) при 600–700 моментах и достигает коэффициента сходимости в 93–95%. Из результатов проведенных вычислительных экспериментов можно сделать вывод, что предложенная нечеткая модель значительно опережает классическую MLP модель в смысле скорости обучения и точности. Также вычислительные эксперименты демонстрируют, что нечеткость в опытных данных не является препятствием для распознавания нелинейных закономерностей.

Нечеткие обучающие образы и нечеткие нейромодели могут быть применены в системах управления, медицине и многих других областях, в которых опытные данные для распознавания исследуемых «входных—выходных» закономерностей построены на субъективной основе.

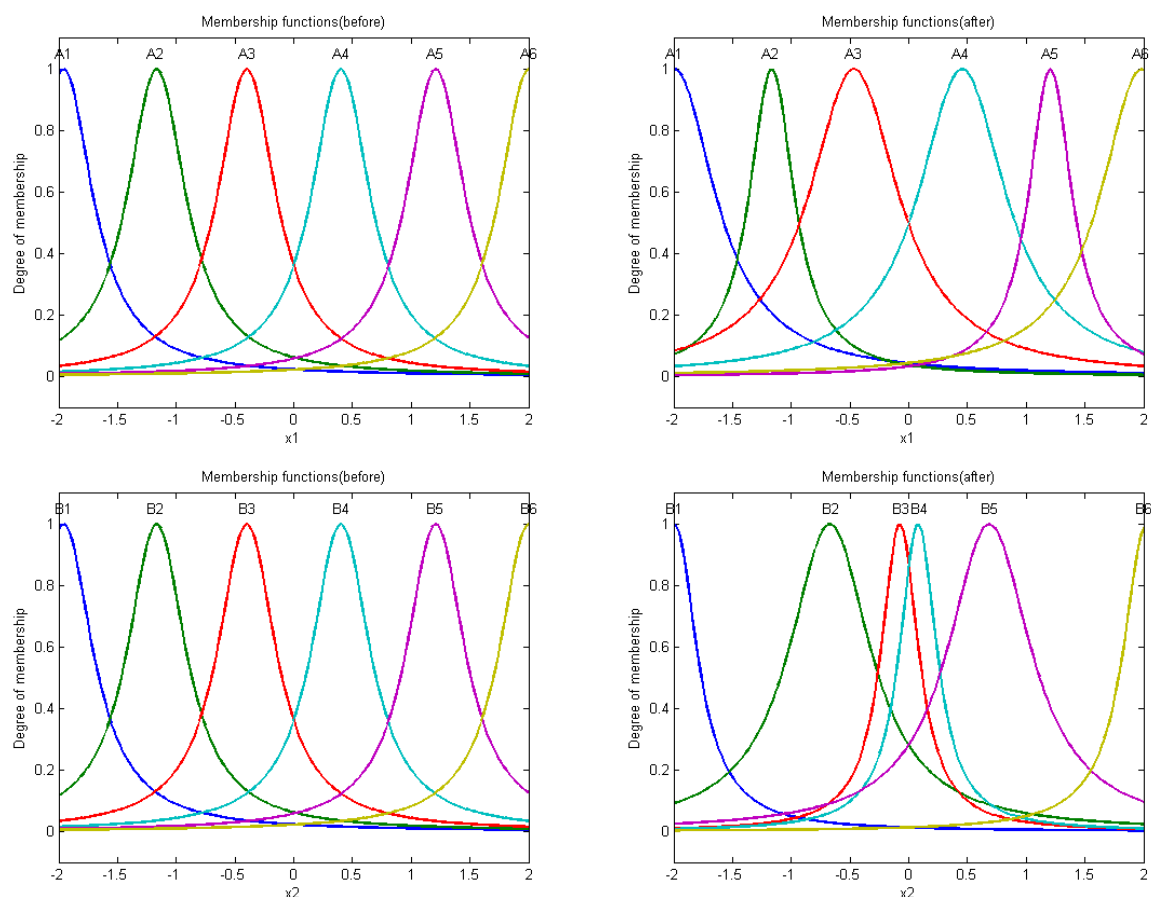


Рис. 7. Графики функций принадлежности

СПИСОК ЛИТЕРАТУРЫ

1. Борисов В. В., Круглов В. В., Федулов А. С. Нечеткие модели и сети. 2-е изд. — М.: «Горячая линия – Телеком», 2012.
2. Митюшкин Ю. И., Мокин Б. И., Ротштейн А. П. Soft Computing: идентификация закономерностей нечеткими базами знаний. — Вінниця: Універсум, 2002.
3. Пегат А. Нечеткое моделирование и управление. — М.: БИНОМ. Лаборатория знаний, 2013.
4. Штовба С. Д. Проектирование нечетких систем средствами MATLAB. — М.: «Горячая линия – Телеком», 2007.
5. Galushkin A. I. Neural networks theory. — Berlin—Heidelberg: Springer-Verlag, 2007.
6. Haykin S. Neural networks. A comprehensive foundation. 2nd ed. — New York: IEEE, 1999.
7. Jose K. M., Fabio M. A. Nonlinear system identification based on modified ANFIS// Proc. 2015 12th Int. Conf. on Informatics in Control, Automation and Robotics (ICINCO), Colmar, France, 21–23 July 2015. — Colmar, 2015. — С. 588–595.
8. Nikov A., Georgiev T. A fuzzy neural network and its matlab simulation// Proc. ITI99 21st Int. Conf. on Information Technology Interfaces, Pula, Croatia, June 15–18. — Pula, 1999. — С. 413–418.
9. Qing-Song M. Approximation ability of regular fuzzy neural networks to fuzzy-valued functions in MS convergence structure// Proc. 32nd Chinese Control Conf., Xian, China, 26–28 July 2013. — Xian, 2013. — INSPEC Acc. Num. 13862419.
10. Rakesh B. P., Satish K. Sh. Identification of nonlinear system using computational paradigms// Proc. Int. Conf. on Automatic Control and Artificial Intelligence, Xiamen, China, 3–5 March 2012. — Xiamen, 2012. — С. 1156–1159.
11. Rotshtein A. P. Design and tuning of fuzzy if-then rules for medical diagnosis// В сб.: «Fuzzy and neural-fuzzy systems in medical and biomedical engineering». — Boca-Raton: CRC Press, 1998. — С. 243–289.
12. Rotshtein A. P., Mityushkin Y. I. Extraction of fuzzy rules from experimental data using genetic algorithms// Cybernet. Systems Anal. — 2001. — № 3. — С. 45–53.

13. Rotshtein A. P., Shtovba S. D. Identification of non-linear dependencies of fuzzy knowledge bases with fuzzy learning inputs// Cybernet. Systems Anal. — 2006. — № 2. — С. 17–24.
14. Rumelhart D. E., Hinton G. E., Williams R. J. Learning internal representations by back-propagating errors// Nature. — 1986. — 323. — С. 533–536.
15. Zimmermann H. J. Fuzzy set theory and its applications. — Dordrecht–Boston: Kluwer, 1991.
16. Zongyuan Z., Shuxiang X., Byeong H. K., Mir M., Yunling L., Rainer W. Investigation and improvement of multi-layer perceptron neural networks for credit scoring// Expert Syst. Appl. — 2015. — 42, № 7. — С. 3508–3516.

А. П. Марахимов

Национальный университет Узбекистана им. М. Улугбека,
Узбекистан, 100174, г. Ташкент, ВУЗ городок, ул. Университетская, д. 4
E-mail: avaz.marakhimov@yandex.ru

К. К. Худайбергенов

Национальный университет Узбекистана им. М. Улугбека,
Узбекистан, 100174, г. Ташкент, ВУЗ городок, ул. Университетская, д. 4
E-mail: kabul85@mail.ru

DOI: 10.22363/2413-3639-2019-65-1-44-53

UDC 517.55

A Fuzzy MLP Approach for Identification of Nonlinear Systems

© 2019 A. R. Marakhimov, K. K. Khudaybergenov

Abstract. In case of decision making problems, identification of non-linear systems is an important issue. Identification of non-linear systems using a multilayer perceptron (MLP) trained with back propagation becomes much complex with an increase in number of input data, number of layers, number of nodes, and number of iterations in computation. In this paper, an attempt has been made to use fuzzy MLP and its learning algorithm for identification of non-linear system. The fuzzy MLP and its training algorithm which allows to accelerate a process of training, which exceeds in comparing with classical MLP is proposed. Results show a sharp reduction in search for optimal parameters of a neuro fuzzy model as compared to the classical MLP. A training performance comparison has been carried out between MLP and the proposed fuzzy-MLP model. The time and space complexities of the algorithms have been analyzed. It is observed, that number of epochs has sharply reduced and performance increased compared with classical MLP.

REFERENCES

1. V. V. Borisov, V. V. Kruglov, and A. S. Fedulov, *Nechetkie modeli i seti. 2-e izd* [Fuzzy Models and Nets. 2nd ed.], Goryachaya liniya – Telekom, Moscow, 2012 (in Russian).
2. Yu. I. Mityushkin, B. I. Mokin, and A. P. Rotshteyn, *Soft Computing: identifikatsiya zakonomernostey nechetkimi bazami znaniy* [Soft Computing: Identification of Regularities by Fuzzy Knowledge Bases], Un?versum, V?nnitsya, 2002 (in Russian).
3. A. Pegat, *Nechetkoe modelirovanie i upravlenie* [Fuzzy Modelling and Control], BINOM. Laboratoriya znaniy, Moscow, 2013 (in Russian).
4. S. D. Shtovba, *Proektirovanie nechetkikh sistem sredstvami MATLAB* [Design of Fussy Systems by Means of MATLAB], Goryachaya liniya – Telekom, Moscow, 2007 (in Russian).
5. A. I. Galushkin, *Neural networks theory*, Springer-Verlag, Berlin–Heidelberg, 2007.
6. S. Haykin, *Neural networks. A comprehensive foundation. 2nd ed*, IEEE, New York, 1999.
7. K. M. Jose and M. A. Fabio, “Nonlinear system identification based on modified ANFIS,” *Proc. 2015 12th Int. Conf. on Informatics in Control, Automation and Robotics (ICINCO), Colmar, France, 21–23 July 2015*, Colmar, 2015, pp. 588–595.
8. A. Nikov and T. Georgiev, “A fuzzy neural network and its matlab simulation,” *Proc. ITI99 21st Int. Conf. on Information Technology Interfaces, Pula, Croatia, June 15–18, Pula, 1999*, pp. 413–418.

9. M. Qing-Song, “Approximation ability of regular fuzzy neural networks to fuzzy-valued functions in MS convergence structure,” *Proc. 32nd Chinese Control Conf., Xian, China, 26–28 July 2013*, Xian, 2013, INSPEC Acc. Num. 13862419.
10. B. P. Rakesh and K. Sh. Satish, “Identification of nonlinear system using computational paradigms,” *Proc. Int. Conf. on Automatic Control and Artificial Intelligence, Xiamen, China, 3–5 March 2012*, Xiamen, 2012, pp. 1156–1159.
11. A. P. Rotshtein, “Design and tuning of fuzzy if-then rules for medical diagnosis,” In: *Fuzzy and neural-fuzzy systems in medical and biomedical engineering*, CRC Press, Boca-Raton, 1998, pp. 243–289.
12. A. P. Rotshtein and Y. I. Mityushkin, “Extraction of fuzzy rules from experimental data using genetic algorithms,” *Cybernet. Systems Anal.*, 2001, No. 3, 45–53.
13. A. P. Rotshtein and S. D. Shtovba, “Identification of non-linear dependencies of fuzzy knowledge bases with fuzzy learning inputs,” *Cybernet. Systems Anal.*, 2006, No. 2, 17–24.
14. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by back-propagating errors,” *Nature*, 1986, **323**, 533–536.
15. H. J. Zimmermann, *Fuzzy set theory and its applications*, Kluwer, Dordrecht–Boston, 1991.
16. Z. Zongyuan, X. Shuxiang, H. K. Byeong, M. Mir, L. Yunling, and W. Rainer, “Investigation and improvement of multi-layer perceptron neural networks for credit scoring,” *Expert Syst. Appl.*, 2015, **42**, No. 7, 3508–3516.

A. R. Marakhimov

National University of Uzbekistan named after M. Ulugbek, Tashkent, Uzbekistan

E-mail: avaz.marakhimov@yandex.ru

K. K. Khudaybergenov

National University of Uzbekistan named after M. Ulugbek, Tashkent, Uzbekistan

E-mail: kabul85@mail.ru