

Построение гибридных нейронных сетей с использованием элементов нечеткой логики

Н. В. Гридина^{1,II}, И. А. Евдокимов^I, В. И. Солодовников^I

^I Центр информационных технологий в проектировании Российской академии наук (ЦИТП РАН), г. Одинцово, Россия

^{II} Первый Московский государственный медицинский университет им. И. М. Сеченова Министерства здравоохранения Российской Федерации России, г. Москва, Россия

Аннотация. Рассмотрены вопросы использования элементов нечеткой логики при построении и обучении гибридных нейронных сетей для работы в условиях неопределенности. Приведен вариант реализации нечетких отношений и алгоритм, осуществляющий процедуру нечеткого композиционного вывода, на основе нейросетевого подхода. Представлена нейросетевая модель нечеткой системы в роли универсального аппроксиматора, а также случай, при котором нечеткость проявляется на этапе обучения.

Ключевые слова: гибридная нейронная сеть; нечеткая логика; нечеткие композиционные нейронные сети.

DOI 10.14357/20718594190209

Введение

Основные трудности, связанные с построением нечетких отношений и организацией нечеткого вывода, зачастую связаны с определением функций принадлежности условной части правил, а также с выбором решающего правила из совокупности кандидатов. Для решения подобных проблем возможно объединение средств нечеткого логического вывода и нейросетевого подхода в рамках гибридных систем. В их основу заложена идея использования в качестве механизма вывода аппарата нечеткой логики и способность нейронных сетей к обучению, что выражается в подстройке параметров, соответствующих функциям принадлежности с применением алгоритмов обучения, таких как алгоритм обратного распространения ошибки. В связи с этим, гибридные нейронные

сети не только используют априорную информацию, предоставленную экспертом, но и способны приобретать новые знания, при этом оставаясь логически прозрачными.

1. Элементы нечетких вычислений в рамках нейросетевого подхода

Нечеткая логика – это разновидность непрерывной логики, в которой логические формулы могут принимать истинностные значения между 0 и 1. В ее основу положено понятие нечеткого множества. Пусть U полное множество, охватывающее всю проблемную область. Нечеткое множество F множества U определяется через функцию принадлежности $\mu_F(u)$ (где u элемент множества U). Эта функция отображает элементы u множества U на множество чисел в отрезке $[0,1]$, которые указывают степень принадлежности каждого элемента нечеткому

*Работа выполняется в рамках темы № 0071-2019-0001 «Развитие теории и методов прикладной математики, нейросетевых технологий и систем управления процессами в задачах CAD-систем, анализа визуальных данных, защиты информации и прогнозирования»

✉ Солодовников Владимир Игоревич. E-mail: info2@ditc.ras.ru

множеству F . Нечеткие системы могут иметь функции принадлежности произвольной структуры, однако, обычно на практике используют треугольную и трапецевидную, которые с большой точностью могут быть аппроксимированы функцией гауссовского типа:

$$\mu_F(u) = \exp \left[- \left(\frac{u - c}{\sigma} \right)^2 \cdot b \right]$$

где c, σ, b - определяют значения центра, ширины и формы колоколообразной функции.

Для описания лингвистических переменных типа «очень большой» или «очень маленький» обычно используются линейные функции с насыщением, которые имеют нелинейный всюду дифференцируемый аналог в виде сигмоидной функции:

$$\mu_F(u) = \frac{1}{1 + \exp[-\alpha \cdot u - \beta]}$$

где α, β - отвечают за сдвиг и крутизну.

Гауссовская и сигмоидная функции являются основными и наиболее часто используемыми функциями активации в нейронных сетях.

Базовая модель нечеткого нейрона во многом схожа с моделью искусственного нейрона, основанного на применении функции активации к результату обработки взвешенного входа. Основным отличием является возможность обработки нечетких входных значений и применения специальных нечетких операций агрегирования [1].

К первому типу можно отнести нечеткий нейрон H_1 (Рис. 1) с четкими входными и выходными сигналами. Данный тип имеет n четких входов $x_1, \dots, x_n$. Весовые коэффициенты такого нейрона представляют собой нечеткие множества $A_i, i = 1 \div n$, т.е. операции взвешивания входных сигналов в классическом нейроне заменяются функциями принадлежности и происходит их объединение (агрегирование) в узле H_1 с выдачей сигнала $y = \mu_{n_1}(x_1, \dots, x_n)$, принадлежащего интервалу $[0, 1]$, который может рассматриваться как уровень правдоподобия.

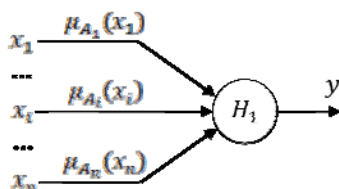


Рис. 1. Нечеткий нейрон первого типа

Операцию агрегирования взвешенных сигналов обозначим как $\otimes$, для реализации которой могут быть использованы нечеткие операторы t -нормы и t -конормы. Треугольной нормой называется двуместная действительная функция $T: [0, 1] \times [0, 1] \rightarrow [0, 1]$, удовлетворяющая условиям ограниченности, монотонности, коммутативности и ассоциативности. В качестве t -норм могут выступать: $\min(\mu_A, \mu_B)$; $\mu_A * \mu_B$; $\max(0, \mu_A + \mu_B - 1)$. Треугольной конормой также является двуместная действительная функция $S: [0, 1] \times [0, 1] \rightarrow [0, 1]$, удовлетворяющая аналогичным условиям. Например: $\max(\mu_A, \mu_B)$; $\mu_A + \mu_B - \mu_A * \mu_B$; $\min(1, \mu_A + \mu_B)$.

В зависимости от типа используемой операции можно получить различные схемы нечетких нейронов. Математическое описание нейрона можно представить в виде:

$$\mu_{n_1}(x_1, \dots, x_n) = \mu_{A_1}(x_1) \otimes \dots \otimes \mu_{A_n}(x_n)$$

Нечеткий нейрон «И» можно представить в виде композиции $\min$ - $\max$:

$$y = \min(w_1 \vee x_1, \dots, w_n \vee x_n).$$

В свою очередь нечеткий нейрон «ИЛИ» реализует композицию $\max$ - $\min$:

$$y = \max(w_1 \wedge x_1, \dots, w_n \wedge x_n).$$

Второй тип нечетких нейронов H_2 подобен нейрону первого типа, но все входы и выходы нейрона являются нечеткими множествами (Рис. 2).

Сигнал на каждом нечетком входе X_i преобразуется путем операции взвешивания, которая осуществляет модификацию входного термина в измененное нечеткое множество $X'_i = A_i * X_i$, $i = 1 \div n$, с использованием некоторого оператора $*$, где A_i - есть i -ый нечеткий вес. Операция взвешивания здесь не является функцией принадлежности, как в нейроне первого типа, вместо нее проводится операция модификации каждого нечеткого входа. Модифицированные входы объединяются оператором агрегации, выходом которого является нечеткий терм. Оператор агрегации может быть выполнен также на основе t -нормальной, либо t -

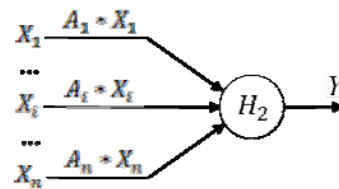


Рис. 2. Нечеткий нейрон второго типа

конормальной функции. Математическое описание нейрона второго типа имеет вид:

$$X'_i = A_i * X_i, i = 1 \div n, Y = X'_1 \otimes \dots \otimes X'_n$$

где Y – нечеткое множество, представляющее выход нейрона.

Нечеткий нейрон третьего типа H_3 (Рис. 3) с n нечеткими входами и одним нечетким выходом задается отношением в виде нечеткого правила IF – THEN:

$$\text{IF } X_{1i} \text{ and } X_{2i} \text{ and } \dots \text{ and } X_{ni} \text{ THEN } Y_i$$

где X_{ji} и Y_i – вход и выход i -ого нейрона, описываемого i -ым правилом.

Такой нейрон может быть представлен нечетким отношением вида:

$$R_i = I(T(X_{1i}, X_{2i}, \dots, X_{ni}), Y_i)$$

В нем I – функция импликации, T – это t -норма, реализующая операцию нечеткой конъюнкции.

$$Y_i = T(X_{1i}, X_{2i}, \dots, X_{ni}) \circ R_i$$

На основе рассмотренных типов нечетких нейронов можно конструировать нечеткие НС различной архитектуры, которые в отличие от обычных нейросетей обладают расширенными возможностями представления и обработки данных.

2. Нейросетевая реализация нечетких отношений

Пусть X и Y – нечеткие множества в U и V соответственно, а R – нечеткое отношение в $U \times V$. Рассмотрим выражение для нечеткого отношения:

$$Y = X \circ R$$

где $\circ$ – операция композиции (например, $\max$ - $\min$ операция)

Пусть $U = \{x_1, \dots, x_n\}$ и $V = \{y_1, \dots, y_m\}$. Тогда можно записать $\max$ - $\min$ композицию:

$$\mu_Y(y_j) = \max_{x_i} (\min(\mu_X(x_i), \mu_R(x_i, y_j)))$$

Это выражение можно записать в матричном виде:

$$X \circ R = [\mu_X(x_1), \dots, \mu_X(x_n)] \circ \begin{bmatrix} \mu_R(x_1, y_1) & \dots & \mu_R(x_1, y_m) \\ \dots & \dots & \dots \\ \mu_R(x_n, y_1) & \dots & \mu_R(x_n, y_m) \end{bmatrix}$$

Такое соотношение соответствует следующей нейросетевой структуре [2] (Рис. 4).

Если использовать обычную операцию суммирования, то результирующий сигнал может быть слишком низким для активизации входа. Поэтому к каждому узлу добавляется порог

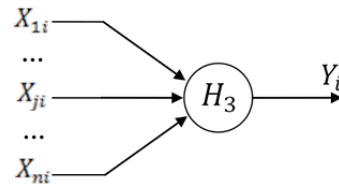


Рис. 3. Нечеткий нейрон третьего типа

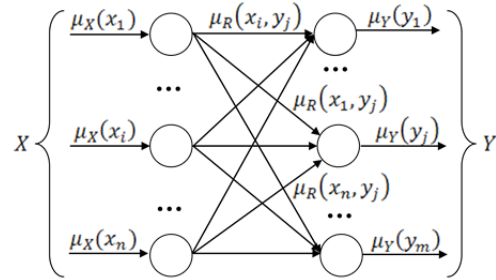


Рис. 4. Модель нейросетевой реализации нечетких отношений

(bias), обозначаемый $b_j \in [0,1]$. Тогда уравнение можно представить в виде:

$$\mu_Y(y) = \max \left(\max \left(\min \left(\mu_X(x_i), \mu_R(x_i, y_j) \right) \right), b(y) \right).$$

Добавление порога, в данном случае, аналогично использованию дополнительного входа равного 1 для искусственного нейрона сети. Такой вход в дальнейшем умножается на соответствующий ему весовой коэффициент и участвует в вычислении потенциала нейрона наравне с остальными значениями.

Основные трудности построения нечетких отношений связаны с настройкой большого числа функций принадлежности. Для устранения этих трудностей можно использовать подход, основанный на обучении НС с весовыми коэффициентами, соответствующими функции принадлежности.

3. Универсальный нечеткий аппроксиматор

Способность нечеткой системы выступать в роли универсального аппроксиматора вытекает из теоремы Ванга, которая может быть сформулирована следующим образом:

Для каждой вещественной непрерывной функции g , заданной на компакте U , и для произвольного $\varepsilon > 0$ существует нечеткая система, формирующая выходную функцию $f(x)$, для которой справедливо условие:

$$\sup_{x \in U} \|g(x) - f(x)\| \leq \varepsilon$$

Такая адаптивная нейро-нечеткая система вывода (Adaptive Network-based Fuzzy Inference System – ANFIS) состоит из набора нечетких правил «если-то» (if-then), использует гауссовские функции принадлежности, композиции в виде произведения, импликации в форме Ларсена и центроидный метод дефаззификации. При этом возможны модификации сети ANFIS в зависимости от применения алгоритмов нечеткого вывода: Сугено-Такаги, Сугено-Такаги-Канга (TSK) и Ванга-Менделя [3, 5].

Структура нечеткой нейронной сети для реализации модели Ванга-Менделя представлена на Рис. 5.

Первый слой выполняет фуззификацию входных переменных. В качестве фуззификатора используется обобщенная гауссовская функция:

$$\mu_{A_j}(x_j) = \exp \left[- \left(\frac{x_j - c_j^{(i)}}{\sigma_j^{(i)}} \right)^{2b_j^{(i)}} \right]$$

где $c_j^{(i)}, \sigma_j^{(i)}, b_j^{(i)}$ обозначают значения центра, ширины и формы j -ого компонента вектора x условной части для i -ого нечеткого правила вывода.

Второй слой осуществляет перемножение истинностных значений предпосылок. Третий – содержит два линейных нейрона, где первый предназначен для расчета взвешенной суммы M правил вывода, а второй для генерации нормализующей величины. Они соответствуют числителю и знаменателю формулы для описания модели Ванга-Менделя. Единственный нейрон выходного слоя осуществляет нормализацию, формируя выходной сигнал $y(x)$, который определяется по формуле:

$$y(x) = \frac{\sum_{i=1}^M v_i \cdot \left[\prod_{j=1}^N \mu_{A_j}(x_j) \right]}{\sum_{i=1}^M \left[\prod_{j=1}^N \mu_{A_j}(x_j) \right]}$$

Первый и третий слои являются параметрическими. В первом слое настройке подлежат параметры функций фаззификации $c_j^{(i)}, \sigma_j^{(i)}, b_j^{(i)}$ для j -ой предпосылки условной части i -ого правила, а в третьем значения $v_1, v_2, \dots, v_M$, которые определяют центры функций принадлежности для каждого правила [4].

Обучение такой сети разделяется на несколько этапов, которые могут циклически повторяться друг за другом до достижения нужной точности

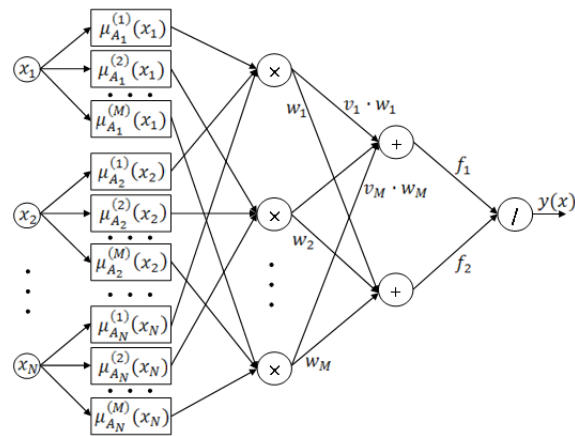


Рис. 5. Структура нечеткой нейронной сети для реализации модели Ванга-Менделя

аппроксимирующей способности сети. На первом этапе, с помощью решения системы линейных уравнений, вычисляются значения $v_1, v_2, \dots, v_M$, определяющие центры функций принадлежности. На втором осуществляется подстройка нелинейных параметров функций фаззификации $c_j^{(i)}, \sigma_j^{(i)}, b_j^{(i)}$, для чего применимы градиентные методы обучения.

4. Нейросетевая модель нечеткого композиционного вывода

Анализ процедуры классического нечеткого вывода позволяет выделить две главные проблемы, возникающие при организации вывода на основе нечеткой логики, которые более эффективно можно реализовать на основе нейросетевого подхода. Первая связана с определением функций принадлежности, используемых в условной части правил, а вторая – с выбором правила, определяющего решение, из совокупности правил-кандидатов. Рассмотрим модель, в которой нейросеть используется для организации вывода. Правила нечеткого вывода имеют следующий вид:

$$R_s : \text{IF } X = (x_1, \dots, x_n) \text{ is } A_s \\ \text{THEN } y_s = HC_s(x_1, \dots, x_n), \\ s = 1 \div r$$

где r – число правил вывода, A_s – представляет нечеткое множество условной части каждого правила.

Нейронная сеть HC_s обучается на входных данных из обучающей выборки. Выходы обученной нейросети рассматриваются как величины функций принадлежности нечетких множеств в условной части IF правил, так как

величина выходного сигнала определяет принадлежность данных к каждому правилу.

Предположим, что в рассматриваемой модели нечеткой системы, состоящей из нескольких нейросетей, имеется n входов $x_1, \dots, x_n$ и единственный выход y . Обозначим через HC_s - нейронную сеть, которая вычисляет выход y_s для каждого правила s . Тогда процедура нечеткого вывода реализуется следующим алгоритмом:

1. Формирование обучающей и тестовой выборок.

N_O , N_T - число примеров обучающей и тестовой выборок соответственно. Общее число примеров $N = N_O + N_T$.

2. Выделение в обучающей выборке кластеров схожих элементов.

Примеры обучающей выборки разделяются с учетом числа правил на r классов R_s , где $s = 1 \div r$. Каждая обучающая подвыборка для класса R_s определяется парами (x_i^s, y_i^s) , где $i = 1 \div N_O^s$, N_O^s - число примеров в обучающей подвыборке для класса R_s .

3. Обучение $HC(\mu)$.

Для каждого входного вектора $X_i \in R_s$ определяется вектор функций принадлежности к правилу $M_i = (m_i^1, \dots, m_i^r)$ такой, что $m_i^s = 1$ и $m_i^k = 0$ для всех $k \neq s$. Нейронная сеть обучается на парах (X_i, M_i) . Функция принадлежности к части IF правила определяется как выходная величина m_i^s обученной сети, то есть $m_{A_s}(X_i) = m_i^s$, $i = 1 \div N$, $s = 1 \div r$. A_s - соответствует нечеткой части правила.

4. Обучение HC_s .

Обучающая выборка с входами $x_{i1}^s, \dots, x_{in}^s$ и выходной величиной y_i^s , $i = 1 \div N_s$ подается на вход и выход HC_s , которая является нейросетевой моделью части THEN в R_s . С помощью тестовой выборки вычисляется ошибка обобщения:

$$E_n^s = \sum_{i=1}^{N_s} (y_i - m_s(X_i) m_{A_s}(X_i))^2,$$

где $m_s(X_i)$ - наблюдаемый выход HC_s . Сеть считается обученной, если $E^s < \delta$, где δ - некая заранее заданная величина.

5. Принятие решения.

Для заданного входного вектора проводится вычисление выходной величины по аналогии с формулой дефаззификации, используемой в классическом композиционном выводе:

$$y_i^* = \frac{\sum_{s=1}^r m_{A_s}(X_i) m_s(X_i)}{\sum_{s=1}^r m_{A_s}(X_i)}$$

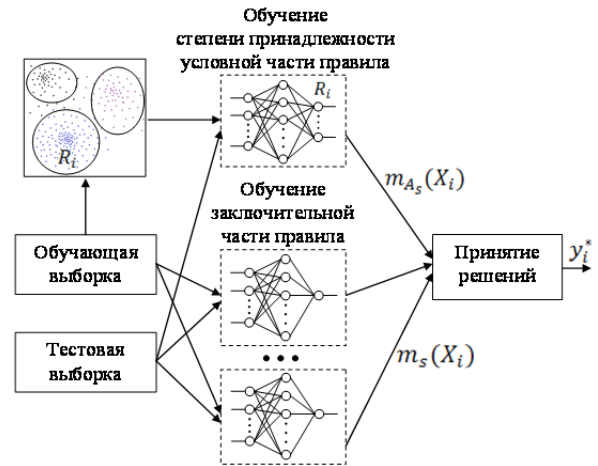


Рис. 6. Функциональная схема нейросетевого нечеткого композиционного вывода

Полученное таким образом значение является результатом нечеткого композиционного вывода. Функциональная схема описанного процесса представлена на Рис. 6.

Выделение в обучающей выборке кластеров схожих элементов может осуществляться с применением алгоритмов самоорганизации, таких как: С-means, пикового группирования, разностного группирования, нечеткой самоорганизации Густафсона-Кесселя [4].

5. Алгоритм обучения нечеткого персептрона

Возможно построение нечетких нейросетей, в которых нечеткость проявляется на этапе ее обучения. В этом случае модифицируется, либо стратегия самого обучения, либо отдельные параметры, заметно влияющие на качество обучения.

В алгоритме обучения каждый вектор обучающей выборки вносит определенный вклад в модификацию весовых коэффициентов. В случае, когда отдельные классы перекрываются, то примеры обучающей выборки, лежащие в зоне перекрытия, определяют неустойчивое поведение алгоритма обучения. Поэтому основная идея заключается в уменьшении влияния векторов, лежащих в зоне перекрытия, на изменение весовых коэффициентов.

Рассмотрим реализацию стандартного алгоритма обратного распространения ошибки в случае разбиения входного пространства на два пересекающихся класса.

Пусть имеется обучающая выборка, состоящая из векторов $\{X_1, \dots, X_p\}$, и функции принадлежности $\mu_j(X_k)$, $j = 1 \div 2$, $k = 1 \div p$, определяющие номера нечеткого разбиения пространства входов на два класса по отношению к этим векторам, показывающая степень принадлежности каждого из этих векторов к каждому из классов. Эти предположения могут быть представлены следующими соотношениями:

$$\sum_1^2 \mu_i(X_k) = 1; 0 < \sum_1^p \mu_i(X_k) < p;$$

$$\mu_i(X_k) \in [0,1].$$

Причем для векторов, чей класс является менее определенным, функция принадлежности находится в интервале до 0.5 (Рис. 7).

Степень влияния образца X_k на модификацию весовых коэффициентов можно определить как разность $|\mu_1(X_k) - \mu_2(X_k)|^m$, где m – константа, которая в случае неопределенности отнесения к одному из двух классов вектора X_k близка к нулю. Учитывая этот факт, можно модифицировать формулу, определяющую коррекцию весовых коэффициентов в алгоритме обратного распространения ошибки для четко-го персептрона:

$$w_j(t) = w_j(t-1) + h \cdot |\mu_1(X_k(t-1)) - \mu_2(X_k(t-1))|^m \cdot [d_k - y(t-1)] \cdot x_j^{(k)},$$

где h – скорость обучения, d_k и y – значения ожидаемого и полученного выходного сигнала при X_k входном векторе, j – номер компонента входного вектора $1 \leq j \leq n+1$, t – текущий такт обучения.

Из уравнения следует, что влияние неопределенности векторов на модификацию весовых коэффициентов можно уменьшить за счет изменения константы m . Действительно, подобно четкому персептрону, в нечетком персептроне можно найти разделяющую поверхность за конечное число итераций. При этом если:

$$|\mu_1(X_k(t)) - \mu_2(X_k(t))|^m = 1,$$

$$\mu_1(X_k), \mu_2(X_k) \in [0,1],$$

то нечеткий персептрон становится четким, и значения величин функций принадлежности также становятся четкими. При выборе стратегии обучения большое значение имеет выбор шага обучения. Поэтому для векторов значения функций принадлежности, которых близки к 0.5, можно установить значения $m > 1$ и на последующих итерациях постепенно его уменьшать.

При разработке алгоритма обучения нечеткого персептрона необходимо решить две задачи:

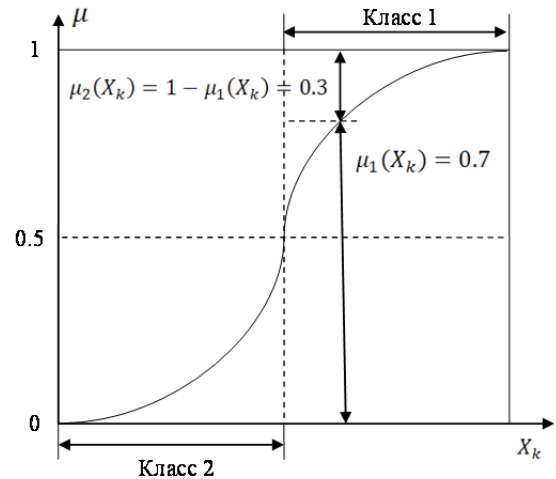


Рис. 7. Функция принадлежности для двух классов

- Определить способ вычисления функции принадлежности на каждом шаге итерации.
- Определить критерий останова.

При решении первой задачи можно использовать формулы разделения двух пересекающихся классов. Тогда для некоторого вектора X_k в классе 1 получаем:

$$\mu_1(X_k) = 0.5 + \frac{e^{f(dist_1 - dist_2)/dist} - e^{-f}}{2(e^f - e^{-f})},$$

$$\mu_2(X_k) = 1 - \mu_1(X_k),$$

где $dist_1$, $dist_2$, $dist$ – расстояние вектора X_k до среднего значения классов 1 и 2 и между средними значениями этих классов; f – положительная константа, подбираемая опытным путем ($f > 2$).

При решении второй задачи необходимо учесть тот факт, что, когда вектор не может быть однозначно отнесен к определенному классу, значение его функции принадлежности близко к 0.5, поэтому для останова процедуры обучения выбирается значение δ , показывающее близость функции принадлежности к величине 0.5. Тогда критерием останова будет выполнение следующего условия:

$$\mu_1(X_k) > 0.5 + \delta \text{ или } \mu_1(X_k) < 0.5 - \delta \text{ для } \forall k, k = 1 \div p.$$

Алгоритм обучения нечеткого персептрона состоит из следующих шагов:

1. Инициализация весовых коэффициентов
2. Вычисление степеней функции принадлежности на всех образцах выборки в каждом классе в соответствии с формулами для $\mu_1(X_k)$ и $\mu_2(X_k)$.
3. Подать на вход сети образцы обучающей выборки. Вычислить ошибку сети и величины, на

которые необходимо изменить весовые коэффициенты с учетом функции принадлежности для образцов выборки. Повторять пункт 3 до тех пор, пока не будет выполняться критерий останова.

В итоге данный алгоритм позволяет уменьшению влияния векторов, лежащих в зоне перекрытия классов, что способно положительно сказаться на процесс обучения гибридного нечеткого персептрона.

Заключение

Построение систем правил и организация вывода на основе нечеткой логики зачастую сталкивается с проблемой определения функций принадлежности условной части правил, а также с выбором решающего правила, из всей совокупности кандидатов. В свою очередь использование нейросетевого подхода сопряжено с проблемой интерпретации полученного результата и его предпосылок. Правила содержатся в весовых коэффициентах, функциях активации и связях между нейронами, но обычно их структура слишком сложна для восприятия и не представляется возможным отделить влияние

определенного признака на целевое значение. Интеграция нейросетевых технологий с принципами нечеткой логики в рамках гибридных систем позволяет использовать не только априорную информацию, предоставленную экспертом при построении систем правил, но и определяет способность приобретать новые знания, при этом оставаясь логически прозрачными.

Литература

1. Andrzej Piegat. Fuzzy modeling and control. Physica-Verlag HD, 2001. – pp.728.
2. Ходашинский И.А., Малютин Н.Д. Модели статических искусственных нейронов. Обзор, Доклады ТУСУРа, 2015, № 2 (36). С. 100-107.
3. Борисов В.В., Круглов В.В., Федюлов А.С. Нечеткие модели и сети. 2-е изд., стереотип. — М.: Горячая линия–Телеком, 2012. — 284 с.
4. Круглов В.В., Дли М.И., Голунов Р.Ю. Нечеткая логика и искусственные нейронные сети — М.: ФИЗМАТЛИТ, 2001. - 201 с.
5. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Горячая линия–Телеком, 2006. — 452 с.
6. Осовский С. Нейронные сети для обработки информации — М.: Финансы и статистика, 2004. – 344 с.

Construction of hybrid neural networks using fuzzy logic elements

N. V. Gridina^{I,II}, I. A. Evdokimov^I, V. I. Solodovnikov^I

^I Center of Information Technologies in Engineering RAS, Odintsovo, Moscow Regions, Russia

^{II} I.M. Sechenov First Moscow State Medical University, Moscow, Russia

Abstract. The issues of using fuzzy logic elements in the construction and training of hybrid neural networks for proceeding under conditions of uncertainty are considered. An embodiment of the fuzzy relations implementation and the fuzzy compositional output algorithm based on the neural network approach are presented. A neural network model of a fuzzy system is presented as a universal approximator, as well as the case while fuzziness appears at the learning stage.

Keywords: hybrid neural network; fuzzy logic; fuzzy compositional neural networks.

DOI 10.14357/20718594190209

References

1. Andrzej Piegat. Fuzzy modeling and control. Physica-Verlag HD, 2001. – pp.728
2. Khodashinskiy I.A., Malyutin N.D. 2015. Modeli staticheskikh iskusstvennykh neyronov [Models of static artificial neurons] // Obzor, Doklady TUSURa [Review, TUSUR reports], № 2 (36). pp. 100-107.
3. Borisov V.V., Kruglov V.V., Fedulov A.S. 2012. Nchetkiye modeli i seti. 2-ye izd., stereotip [Fuzzy models and networks. 2nd ed., Stereotype] — M.: Goryachaya liniya [Hotline]–Telekom. — 284 p.
4. Kruglov V.V., Dli M.I., Golunov R.YU. 2001. Nchetkaya logika i iskusstvennyye neyronnyye seti [Fuzzy logic and artificial neural networks] — M.: FIZMATLIT. - 201 p.
5. Rutkovskaya D., Pilin'skiy M., Rutkovskiy L. 2006. Neyronnyye seti, geneticheskiye algoritmy i nechetkiye sistemy [Neural networks, genetic algorithms and fuzzy systems]. M.: Goryachaya liniya [Hotline]–Telekom. — 452 p.
6. Osovskiy S. 2004. Neyronnyye seti dlya obrabotki informatsii [Neural networks for information processing] — M.: Finansy i statistika [Finance and Statistics]. – 344 p.