

*Л.А.Бойко, Навчально-науковий комплекс «Інститут прикладного системного аналізу», Національний технічний університет України «Київський політехнічний інститут»*

## **АРХІТЕКТУРА НЕЙРОННОЇ МЕРЕЖІ МОДЕЛІ NEFCLASS В ЗАДАЧАХ КЛАСИФІКАЦІЇ**

Бойко Л.А.

Архітектура нейронної мережі моделі NEFCLASS в задачах класифікації

У статті викладено основні відомості про нейронну мережу моделі NEFCLASS, її особливості при вирішенні класифікаційних задач. Розглянуто особливості навчання в системі NEFCLASS, зокрема представлено алгоритм навчання нечітких множин.

*Ключові слова:* алгоритм, нейронна мережа NEFCLASS, класифікація, модель.

Бойко Л.А.

Архитектура нейронной сети модели NEFCLASS в задачах классификации

В статье изложены основные сведения о нейронной сети модели NEFCLASS, ее особенности при решении классификационных задач. Рассмотрены особенности обучения в системе NEFCLASS, в частности представлен алгоритм обучения нечетких множеств.

*Ключевые слова:* алгоритм, нейронная сеть NEFCLASS, классификация, модель.

Приступаючи до розробки нейромережевого рішення, як правило, зустрічаєшся з проблемою вибору оптимальної архітектури нейронної мережі. Оскільки області застосування найбільш відомих парадигм перетинаються, то для розв'язання конкретної задачі можна використовувати зовсім різні типи нейронних мереж, і при цьому результати можуть виявитись однаковими. Чи буде та чи інша мережа краще й практичніше, залежить в більшості випадків від умов задачі.

Для розв'язання більш складних класифікаційних задач необхідно ускладнити мережу ввівши додаткові (приховані) шари нейронів, які

виконують проміжну попередню обробку вхідних даних, таким чином, щоб вихідний нейрон-класифікатор отримував на свої входи вже лінійно-роздільні множини. Такі структури носять назву багат шарові перцептрони.

Легко показати, що, в принципі, завжди можна обійтись одним прихованим шаром, який містить достатньо велику кількість нейронів. Дійсно, збільшення прихованого шару підвищує розмірність простору, в якому вихідний нейрон виконує класифікацію, що, відповідно, полегшує його задачу.

Перцептрони досить популярні в нейроінформатиці. І це обґрунтовано, в першу чергу, широким кругом доступних їм задач, в тому числі й задач класифікації, розпізнавання образів, фільтрації шумів, передбачення часових рядів тощо, причому застосування саме такої архітектури в ряді випадків досить оправдано, з точки зору ефективності розв'язання задачі. Розглянемо архітектуру нейронної мережі для задач класифікації на прикладі моделі NEFCLASS (NEuro Fuzzy CLASSifier).

**Загальні відомості.** Метою моделі NEFCLASS є отримання нечітких правил із множини даних, які можна розділити на різні класи. Нечіткі правила описують дані у формі:

$R$ : якщо  $x_1$  має функцію належності  $\mu_1$ ,  $x_2 - \mu_2$ , ... ,  $x_n - \mu_n$ , то зразок  $(x_1, x_2, \dots, x_n)$  належить класу  $i$ , де  $\mu_1, \dots, \mu_n$  – нечіткі множини.

Задача NEFCLASS полягає у тому, щоб визначити належність до класу  $C \subset R^n$  вхідного зразка  $x = (x_1, \dots, x_n) \in R^n$ . Мається на увазі, що перетин двох різних множин пустий. Розглянемо більш детально архітектуру моделі NEFCLASS.

Система NEFCLASS має 3-шарову послідовну архітектуру. Перший шар  $U_1$  містить вхідні нейрони, в яких представляються вхідні зразки. Активація  $a_x$  нейрону  $x \in U_1$  звичайно не змінює вхідне значення. Прихований шар  $U_2$  містить нечіткі правила, і третій шар  $U_3$  складається з вихідних нейронів кожного класу. Активація для нейронів правил  $R \in U_2$  і для нейронів вихідного

шару  $c \in U_3$  із зразком  $p$  обчислюється так:  $a_R^{(p)} = \min_{x \in U_1} \{W(x, R)(a_x^{(p)})\}$ ,

$a_c^{(p)} = \sum_{R \in U_2} W(c, R) \cdot a_R^{(p)}$ , або альтернативно  $a_c^{(p)} = \max_{R \in U_2} \{a_R^{(p)}\}$ , де  $W(x, R)$  – нечітка вага

з'єднання вхідного нейрону  $x$  з нейроном правила  $R$ , а  $W(R, c)$  – нечітка вага з'єднання нейрону правила  $R$  з нейроном вихідного шару  $c$ .

Замість застосування операцій досягнення максимуму та мінімуму можна користуватися іншими функціями  $t$ -норми і  $t$ -конорми відповідно.

База правил уявляє собою апроксимацію невідомої функції  $\varphi: R^n \rightarrow \{0,1\}^m$  і описує класифікаційну задачу, де  $\varphi(x) = (c_1, \dots, c_m)$  така, що  $c_i = 1$ ,  $c_j = 0$  ( $j \in \{1, \dots, m\}, j \neq i$ ) і  $x$  належить класу  $c_i$ .

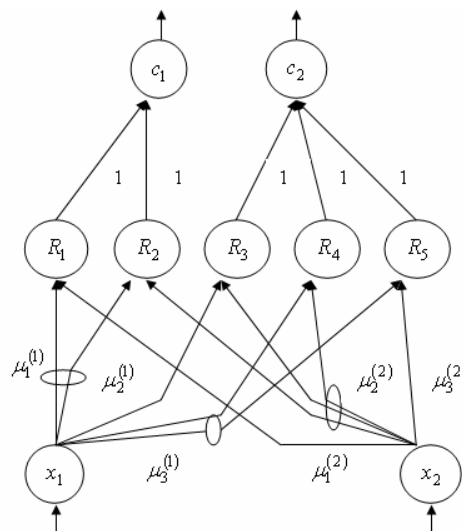


Рис. Система NEFCLASS у вигляді 3-шарової нейронної мережі з двома вхідними нейронами

Нечіткі множини і лінгвістичні правила представляють апроксимацію і визначають результат системи NEFCLASS. Вони отримуються із множини вибірок шляхом навчання. Обов'язково повинно виконуватися правило, що для кожного лінгвістичного значення може існувати тільки одне представлення нечіткої множини.

**Навчання в системі NEFCLASS.** Система NEFCLASS може бути побудована по частковим знанням про зразки. Користувач повинен визначити кількість початкових нечітких множин і задати значення  $k$  – максимальне

число вузлів правил, які можуть бути створені в прихованому шарі. Для навчання ми будемо використовувати визначену раніше трикутну функцію належності. Розглянемо безпосередньо алгоритм навчання.

Розглянемо систему NEFCLASS з  $n$  вхідними нейронами  $x_1, \dots, x_n$ ,  $k \leq k_{\max}$  нейронами правил і  $m$  вихідними нейронами  $c_1, \dots, c_m$ . Також задана навчаюча множина зразків  $L = \{(p_1, t_1), \dots, (p_s, t_s)\}$ , кожний з котрих складається із вхідного зразка  $p \in R^n$  і бажаного зразка  $t \in \{0, 1\}^m$ . Навчаючий алгоритм, мета якого створити  $k$  нейронів правил системи NEFCLASS, складається з наступних етапів:

1) Обираємо наступний зразок  $(p, t)$  із  $L$ .

2) Для кожного вхідного нейрону  $x_i \in U_i$  знаходимо таку функцію належності  $\mu_{ji}^{(p)}$ , що  $\mu_{ji}^{(p)} p(i) = \max_{j \in \{1, \dots, q_1\}} \{\mu_{ji}^{(p)} p(i)\}$

3) Якщо все ще вузлів правил залишилося  $< k_{\max}$  і не існує вузла правила  $R$  з  $W(x_1, R) = \mu_{j_1}^{(1)}$ , ...,  $W(x_n, R) = \mu_{j_n}^{(n)}$ , то створюємо такий вузол і поєднуємо його з вихідним вузлом  $c_1$ , якщо  $t_1 = 1$ .

4) Якщо ще залишилися необроблені зразки в  $L$  і  $k \leq k_{\max}$ , то йдемо на крок 1, а інакше стоп.

5) Визначаємо базу правил за однією з трьох наступних процедур:

а) “просте” навчання правил: залишаємо тільки перші  $k$  правил (зупиняємо створення правил, якщо було створено  $k = k_{\max}$  правил).

б) “краще” навчання правил: оброблюємо зразки в  $L$  і накопичуємо активації кожного нейрону правил для кожного класу зразків, які були розповсюджені. Якщо нейрон правила  $R$  показує більше накопичування активації для класу  $C_j$ , ніж для класу  $C_R$ , який був специфікований для наслідку правила, тоді змінюємо наслідок  $R$  на  $C_j$ , тобто поєднуємо  $R$  з нейроном виходу  $C_j$ . Продовжуємо обробку зразків в  $L$  далі і обчислюємо для кожного нейрону правил:

$$V_R = \sum_{p \in L} a_R^{(p)} \cdot e_p, \text{ де } e_p = \begin{cases} 1, & \text{якщо } p \text{ класифіковано вірно,} \\ -1, & \text{інакше.} \end{cases}$$

Залишаємо  $k$  нейронів правил з найвищими значеннями  $V_R$  і вилучаємо інші нейрони правил із системи NEFCLASS.

в) “кращій для кожного класу” алгоритм навчання: робимо так, як і в передостанньому випадку, але залишаємо для кожного класу  $C_j$  ті кращі  $\lfloor k/m \rfloor$  правил, наслідки яких представляють клас  $C_j$  (де  $\lfloor x \rfloor$  – ціла частина  $x$ ).

**Алгоритм навчання нечітких множин.** Алгоритм навчання з вчителем системи NEFCLASS повинен адаптувати його нечіткі множини пробігати циклічно через всю навчаючу множину  $L$ , продовжуючи нижче перераховані кроки, поки не виконається один з критеріїв останову.

Кроки:

1. Обираємо наступний зразок  $(p, t)$  із  $L$  і розповсюджуємо його через систему NEFCLASS і визначаємо вихідний вектор  $c$ .

2. Для кожного вихідного нейрону  $c_i$  визначаємо значення  $\delta_{c_i} = t_i - a_{c_i}$ .

3. Для кожного нейрону правил  $R$  з  $a_R > 0$ :

а) Визначаємо значення

$$\delta_R = a_R \cdot (1 - a_R) \cdot \sum_{c \in U_3} W(R, c) \delta_c;$$

б) Знаходимо таке  $x'$ , що

$$W(x', R)(a_{x'}) = \min_{x \in U_1} \{W(x, R)(a_x)\};$$

в) Для нечітких множин  $W(x', R)$  визначаємо  $\delta_a, \delta_b, \delta_c$ , використовуючи швидкість навчання  $\sigma > 0$ :

$$\delta_b = \sigma \cdot \delta_R \cdot (c - a) \cdot \text{sgn}(a_{x'} - b),$$

$$\delta_a = -\sigma \cdot \delta_R \cdot (c - a) + \delta_b,$$

$$\delta_c = \sigma \cdot \delta_R \cdot (c - a) + \delta_b,$$

і застосовуємо зміни до  $W(x', R)$ ;

г) Обчислюємо помилку правила

$$E = a_R \cdot (1 - a_R) \cdot \sum_{c \in U_3} (2 \cdot W(R, c) - 1) \delta_c .$$

В якості критерію останову можна взяти, наприклад, такі:

- 1) Помилка на протязі  $n$  ітерацій не зменшується.
- 2) Зупинити навчання по досягненню помилкою визначеного (бажано близького до нуля) значення.

**Висновки.** Незважаючи на великі можливості нейронних мереж, існує ряд недоліків, які все ж обмежують застосування нейромережових технологій. По-перше, нейронні мережі дозволяють знайти тільки субоптимальне рішення і відповідно вони неприйнятні для задач, в яких потрібна висока точність. Функціонуючи за принципом чорного ящика, вони також неприйнятні в випадку, коли необхідно пояснити причину прийняття рішення. Навчена нейромережа видає відповідь за доли секунд, однак відносно висока обчислювальна вартість процесу навчання як за часом, так і за об'ємом пам'яті також суттєво обмежує можливості їх використання. І все ж таки клас задач, для розв'язання яких ці обмеження не критичні, достатньо широкий.

### Література

1. **Зайченко Ю.П.** Конспект лекцій. Інтелектуальні системи. – [www.iasa.org.ua](http://www.iasa.org.ua)
2. **Круглов В.В., Борисов В.В.** Принятие решений на основе нечетких моделей : примеры использования. – М. : Горячая линия Телеком. – 2001.
3. **Осовский С.** Нейронные сети для обработки информации. – М. : Финансы и статистика, 2002. – 344 с. : ил.
4. **Вишневский Р.В., Грачева М.В.** Использование аппарата нечеткой математики в задаче оценки эффективности инвестиций. – <http://rvish.chat.ru/FuzInv2.htm>

Boiko L.A.

Architecture of neuron net of NEFCLASS model in classification tasks

The main information about the neuron net of NEFCLASS model, its peculiarities while solving classification tasks are given in the article. The features of education in the NEFCLASS system, in particular the algorithm of studying illegible number are considered.

*Keywords:* algorithm, neuron net of NEFCLASS, classification, model.