

УДК 004.5

О.В. НИКИФОРОВ*, В.Г. ПУТЯТІН**

НЕЙРОМЕРЕЖЕВІ МОДЕЛІ УПРАВЛІННЯ ПРОЦЕСОМ ФУНКЦІОНУВАННЯ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

*Науковий центр Повітряних Сил Харківського національного університету Повітряних Сил імені Івана Кожедуба, м. Харків, Україна

**Інститут проблем реєстрації інформації НАН України, м. Київ, Україна

Анотація. *Нейромережеві моделі, які спочатку застосовувалися для моделювання процесів розпізнавання графічних зображень, на цей час знайшли широке використання і в галузі розпізнавання багатопараметричних об'єктів, а також регулювання параметрів роботи складних систем. Таке їх застосування є ефективним стосовно вирішення задач управління інформаційною безпекою. За допомогою нейромереж успішно вирішуються завдання класифікації загроз, вибору параметрів заходів захисту, регулювання режимів роботи інформаційних систем. У науковій літературі помічається велике різноманіття підходів та методів щодо створення штучних нейронних мереж. Це, на початковому етапі проєктування систем інформаційної безпеки, коли необхідно визначити основні параметри створюваного програмного забезпечення, представляє проблему для проєктувальника. Прорахунки, зроблені на даному етапі проєктування, можуть призвести до невдалого завершення всього проєкту. У статті виконаний короткий порівняльний аналіз штучних нейронних мереж, які відрізняються за методами налаштувань (навчання) мережі і за формою її структури. Охарактеризовано застосовність, переваги і недоліки таких методів налаштувань нейромереж, як метод зворотного розповсюдження помилки; генетичний алгоритм; ітеративний алгоритм Відроу-Хоффа з мінливим кроком; модифікований метод найменших квадратів; метод послідовного навчання. Показано відмінності структури для мереж, призначених для вирішення задач регулювання інформаційних процесів і розпізнавання багатопараметричних об'єктів. Для регулювання процесів структуру нейромережі розглянуто на прикладі нечіткої нейронної мережі ANFIS. Для задач класифікації приведена структура багатошарового перцептрона, в якому структура внутрішніх шарів відображає онтологічну мережу розглянутої предметної області. Представлені результати можуть бути використаними при обґрунтуванні виду нейронної мережі, що застосовується для розв'язання конкретної задачі в області інформаційної безпеки.*

Ключові слова: *штучна нейронна мережа, алгоритм налаштування, ваговий коефіцієнт, структура мережі, розпізнавання багатопараметричних об'єктів, налаштування процесу.*

Abstract. *Neural network models, which were initially used to model the processes of recognition of graphic images, are now widely used in the field of recognition of multi-parameter objects and in adjusting the parameters of complex systems. Such an application is effective in solving problems of information security management. With the help of neural networks, the tasks of classifying threats, choosing the parameters of protection measures, and adjusting the operating modes of information systems are successfully solved. In the scientific literature, there is a wide variety of approaches and methods for creating artificial neural networks. At the initial stage of designing information security systems, when it is necessary to determine the main parameters of the software being created, it poses a problem for the designer. Miscalculations made at this stage of the design can lead to the unsuccessful completion of the entire project. The article provides a brief comparative analysis of artificial neural networks that differ in the methods of setting (training) the network and in the form of its structure. The applicability, advantages, and disadvantages of such methods for tuning neural networks are characterized. These are the method of backpropagation of an error, genetic algorithm, iterative Widrow-Hoff algorithm with variable step, modified least squares method, and sequential learning method. The differences in the structure for networks designed to solve the problems of adjusting information processes and recognizing multi-parameter objects are shown. To regulate the processes, the structure of the neural network is studied on the example of*

the fuzzy neural network ANFIS. For the classification problems, the structure of a multilayer perceptron is given, in which the structure of the inner layers reflects the ontological network of the subject area under consideration. The presented results can be used to justify the type of neural network used for a specific task in the field of information security.

Keywords: artificial neural network; tuning algorithm; weight coefficient; network structure; recognition of multi-parameter objects; process setting.

DOI: 10.34121/1028-9763-2023-2-34-43

1. Вступ

Нейромережеві моделі використовуються як засіб виявлення та прогнозування загроз інформаційній безпеці в системах інформаційного захисту. Існують кілька методів побудови штучних нейронних мереж (ШНМ), що вирішують завдання такого роду. Для здійснення проектування систем інформаційної безпеки на основі ШНМ доцільно сформулювати стислу характеристику таких методів, що використовуються для їх створення.

Мета статті. Головне завдання роботи – це дати загальну постановку характерних задач налаштування штучних нейронних мереж визначеної структури, придатних для розв’язання задач класифікації багатопараметричних об’єктів – основи методів управління процесами захисту інформації.

Нейронною мережею називається динамічна система, що складається з сукупності пов’язаних між собою за типом вузлів направлено графа елементарних процесорів, які називаються формальними нейронами, і здатна генерувати вихідну інформацію у відповідь на вхідний вплив [1].

Штучна нейронна мережа – це мережа простих елементів, званих нейронами, які отримують вхід, змінюють свій внутрішній стан (збудження) відповідно до цього входу, і виробляють вихід, залежний від входу та збудження. Мережа утворюється з’єднанням виходів певних нейронів із входами інших нейронів з утворенням орієнтованого зваженого графа. Ваги, як і функції, що обчислюють збудження, можуть змінюватися процесом навчання, який керується правилом навчання.

Штучні нейронні мережі – математичні моделі, а також їх програмні або апаратні реалізації, побудовані за принципом організації й функціонування біологічних нейронних мереж – мереж нервових кліток живого організму. Це поняття виникло при вивченні процесів, що протікають у мозку, і при спробі змодельовати ці процеси. Першою такою спробою були нейронні мережі Мак-Каллока й Піттса [2]. Згодом, після розробки алгоритмів навчання, одержувані моделі стали використовувати у практичних цілях: у завданнях прогнозування, для розпізнавання образів, у завданнях керування та ін.

Актуальність досліджень ШНМ підтверджується різноманіттям їх можливих застосувань. Теорія штучних нейронних мереж є алгоритмічним базисом розвитку нейрокомп’ютерів подібно до того, як булева алгебра більше 50 років була алгоритмічним базисом однопроцесорних і багатопроцесорних ЕОМ.

2. Математична постановка задач настроювання нейронних мереж

Формальний нейрон (рис. 1) має групу синапсів – односпрямованих вхідних зв’язків, з’єднаних із виходами інших нейронів, а також має аксон – вихідний зв’язок даного нейрона, за яким сигнал (збудження або гальмування) надходить на синапс наступних нейронів. Кожен вхід множиться на відповідну вагу, аналогічну синаптичній силі, і всі добутки підсумовуються, визначаючи рівень активації нейрона. Множину вхідних сигналів позначено вектором $\mathbf{X} = \{x_i\}$. Кожна вага w_i відповідає «силі» одного синаптичного зв’язку.

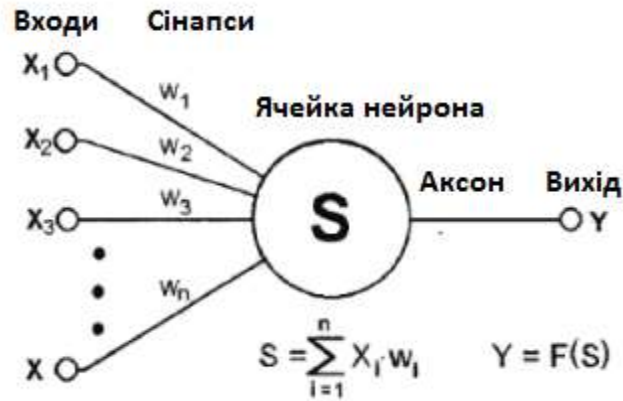


Рисунок 1 – Формальний нейрон

Для створення штучної нейронної мережі використовується штучний нейрон (рис. 2), який вперше був запропонований Мак-Каллоком і Піттсом у 1943 році [2].

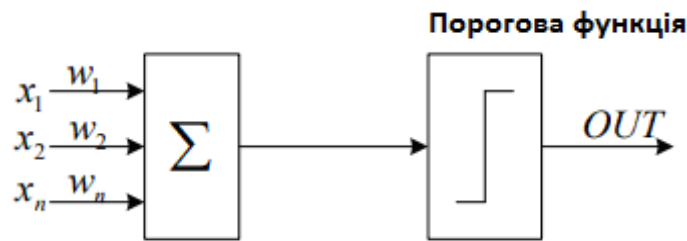


Рисунок 2 – Штучний нейрон Мак-Каллока і Піттса

Порогова активаційна функція нейрона Мак-Каллока і Піттса має вид

$$F(S) = \begin{cases} a, & \text{якщо } S < 0, \\ b, & \text{якщо } S \geq 0. \end{cases}$$

Для $a = -1$, $b = 1$ реалізується симетрична порогова функція, для $a = 0$, $b = 1$ – зміщена порогова функція.

Нейрон Мак-Каллока і Піттса використовується для ШНМ типу перцептрон. Для розпізнавання багатопараметричних об'єктів використовується штучний нейрон Мамдані з сигмоїдальною пороговою функцією

$$F(S) = \frac{1}{1 + e^{-\alpha S}}.$$

Для сигмоїдальної функції вихідне значення нейрона полягає у границях $[0; 1]$. За допомогою ШНМ із сигмоїдальними пороговими функціями нейронів зручно визначати значення нечітких лінгвістичних мінливих, якщо для термів (інтервалів значень) цих мінливих встановлені також сигмоїдальні функції належності.

Цінною властивістю такої функції є простий вираз її похідної:

$$F'(S) = \alpha F(S)(1 - F(S)).$$

Вагові коефіцієнти ШНМ визначаються у процесі навчання мережі. При навчанні використовуються такі алгоритми:

– з вчителем, включаючи:

- 1) алгоритм зворотного розповсюдження помилки (ЗРП);

- 2) генетичний алгоритм (ГА);
 - без вчителя (алгоритми адаптації або градієнтні алгоритми), включаючи:
 - 1) ітеративний алгоритм адаптації Уїдроу-Хоффа з мінливим кроком;
 - 2) модифікований метод найменших квадратів (ММНК);
 - 3) метод послідовного навчання.

Алгоритм зворотного розповсюдження помилки

Навчання за допомогою ЗРП передбачає такий порядок дій:

- обрати чергову навчаючу пару з навчаючої множини і подати на вхід мережі;
- обчислити вихід мережі;
- визначити різницю між виходом мережі і потрібним виходом (цільовим вектором навчаючої пари);
- скоригувати вагові коефіцієнти мережі так, щоб мінімізувати помилку;
- повторити такі кроки для кожного навчаючого вектора до тих пір, поки помилка на всій множині не досягне задовільного значення.

Коригування вагових коефіцієнтів мережі відбувається пропорційно помилці відк-лику. Для цього, спочатку, визначається помилка вихідного нейрона:

$$\delta_j = \Delta_j \cdot F'(S_j) = (t_j - S_j) \cdot [S_j (1 - S_j)],$$

де δ_j – помилка j -го нейрона, що належить вихідному шару ШНМ;

t_j – початкове значення вихідної функції в j -ому нейроні (термі);

S_j – визначене значення вихідної функції при встановлених вагових коефіцієнтах штучних нейронних мереж.

Потім, зворотним ходом, визначаються помилки для нейронів прихованих шарів та вхідного шару:

$$\delta_k = [S_k (1 - S_k)] \cdot \delta_j w_{k,j},$$

де δ_k – помилка k -го нейрона, що утворює підмножину нейронів, виходи яких є входами для j -го нейрона;

$w_{k,j}$ – ваговий коефіцієнт, встановлений для зв'язку k -го нейрона з j -им нейроном.

Значення коригувань для наявних вагових коефіцієнтів розраховуються як

$$\Delta w_{i,k} = \eta \cdot \delta_i S_k,$$

де η – коефіцієнт швидкості навчання ($\eta = 0,01 \div 1$).

Визначення необхідної кількості ітерацій (розміру навчальної вибірки) для налаштування ШНМ, щоб забезпечувалася помилка не більше заданої, здійснюється за допомогою такого співвідношення:

$$N \succ \frac{n}{\varepsilon},$$

де n – число зв'язків у ШНМ;

ε – задана помилка ШНМ.

Існують комерційні програмні продукти для налаштування вагових коефіцієнтів штучних нейронних мереж за допомогою зворотного розповсюдження помилки. Як приклад такого продукту слід навести програму Brein Maker [2].

Генетичні алгоритми

В основу генетичних алгоритмів покладено механізм природного відбору (еволюції) [3]. Ітерації процесу пошуку рішення здійснюються шляхом відтворення зі старої «популяції» альтернатив нової популяції з подальшим відбором перспективних альтернатив. Частоти вибору альтернатив залежать від їх властивостей (ступеня пристосованості особи).

Генетичні алгоритми застосовуються для:

- добору вагових коефіцієнтів зв'язків ШНМ при використанні методу зворотного розповсюдження помилки;
- добору правил навчання або параметрів, що керують навчанням штучних нейронних мереж;
- аналізу ШНМ із метою розуміння, що і чому це робить мережа.

При підборі вагових коефіцієнтів штучних нейронних мереж за допомогою генетичних алгоритмів знаходиться початкове рішення (первинне значення вектора ваги), яке досить близьке до оптимального. Існує комерційний програмний продукт Evolver [4], який реалізує початкову оптимізацію вагових коефіцієнтів за допомогою генетичних алгоритмів. Далі, при відомому початковому значенні вектора вагових коефіцієнтів відбувається їх поліпшення за допомогою методу зворотного розповсюдження помилки, наприклад, за допомогою програми Brein Maker [2].

При використанні генетичних алгоритмів необхідно:

- здійснити кодування інтервалів значень вагових коефіцієнтів ШНМ у вигляді «хромосом»;
- визначити функцію пристосованості «хромосоми» (критерій відповідності);
- розрахувати значення нейронів вихідного шару ШНМ при початкових значеннях «хромосом»;
- виконати випадкову селекцію «хромосом», якщо ймовірності їх вибору залежать від величин їх функцій пристосованості;
- провести нову популяцію «хромосом» шляхом схрещування відібраних пар;
- визначити функції пристосованості нової популяції «хромосом»;
- розрахувати новий вихід ШНМ;
- порівняти новий вихід зі старим виходом. Прийняти рішення про подальшу ітерацію генетичних алгоритмів.

Алгоритм адаптації Уідроу-Хоффа

Уідроу і Хофф модифікували перцептронний алгоритм Ф. Розенблатта, додатково ввівши симоїдальну функцію активації. Їхні моделі «Адлін» (з одним вихідним нейроном) і «Мадалін» (багато вихідних нейронів) набули широкого поширення. Вони довели, що мережа за певних умов сходиться до будь-якої функції, яку вона може уявити.

Процедура Уідроу-Хоффа розроблена для «чорної скриньки», в якій між входами та виходами існують лише прямі зв'язки. Процедура навчання Мадалін (ШНМ із кількома вихідними нейронами) у тому, що ваги внутрішніх зв'язків між нейронами підлаштовуються до того часу, доки встановиться необхідне співвідношення між вхідними і вихідними векторами. Процес складається із двох фаз, що чергуються. У першій фазі на входах задається вхідний вектор, а на виходах – необхідний вихідний вектор. Потім ваги всіх зв'язків, що з'єднують активні входи та виходи, збільшуються на малу величину δ . У другій фазі на вході формується той самий вхідний вектор, але вихід розраховується штучною нейронною мережею. При цьому має дотримуватися таке правило: вихід активізується лише тоді, коли сума ваги його зв'язків з активними входами позитивна. Після цього ваги всіх зв'язків, що з'єднують активні вхідні та вихідні елементи, зменшуються на величину δ . Якщо мережа виробила правильний вихідний вектор, то ці зменшення ваг точно компенсують їх збільшення, вироблені в першій фазі, оскільки в обох фазах активні одні і ті ж

пари вхід-вихід. Якщо мережа виробила вихідний вектор, який потрібен, то зміна ваг, розрахованих у першій фазі, зберігається. На практиці з метою навчання нейрона його охоплюють зворотним зв'язком через блок алгоритму адаптації.

У ланцюзі зворотного зв'язку порівнюється фактичний вихідний сигнал нейрона:

$$r = \sum_{i=1}^n w_i x_i,$$

де w_i , x_i – вагові коефіцієнти вхідних сигналів та самі вхідні сигнали нейрона, відповідно, з бажаним сигналом d . Алгоритм адаптації підлаштовує вагові коефіцієнти вектора вхідних сигналів так, щоб звести помилку $\varepsilon = d - r$ до мінімуму. Для цього вирішується задача знаходження безумовного екстремуму функції $\varepsilon^2(w_1, \dots, w_n)$:

$$\frac{d(\varepsilon^2)}{d\mathbf{W}} = 0,$$

де $\mathbf{W} = \{w_1, \dots, w_n\}$.

Для вирішення цієї задачі існує досить велика кількість методів чисельного розв'язання екстремальних завдань. Наприклад, це методи покоординатного спуску, градієнтного спуску [5].

Перевагою алгоритму Уїдроу-Хоффа є відсутність обмеження на вигляд функції належності (терму) та її розташування у нормованому інтервалі, а недолік – недостатня швидкодія.

Модифікований метод найменших квадратів

Модифікований метод найменших квадратів (ММНК) [1] дозволяє підвищити швидкодію алгоритму Уїдроу-Хоффа за рахунок здійснення розрахунку синаптичних ваг нейронів за одну ітерацію за умови лінійних терм-множин фазифікатора лінгвістичної змінної.

Головна відмінність модифікованого методу найменших квадратів полягає в тому, що замість використання ітераційної процедури покоординатного або градієнтного спуску використовується процедура одноразового розв'язання алгебраїчної системи рівнянь:

$$\frac{\partial(\varepsilon_k^2)}{\partial w_{i,k}} = 2 \left(\sum_{j=1}^n w_{j,k} x_j - d_k \right) = 0, \quad i, k = 1, \dots, n,$$

де ε_k – помилка по виходу k -го нейрона;

n – кількість нейронів ШНМ;

d_k – встановлений вихід для k -го нейрона;

$w_{j,k}$ – вагові коефіцієнти зв'язків ШНМ.

Матриця вагових коефіцієнтів є квадратною, з розмірністю $n \times n$. Для відсутніх зв'язків відповідні синаптичні сигнали x_j набувають нульових значень.

Модифікований метод найменших квадратів застосовується лише для лінійних терм, піддіапазони яких рівні і рівномірно розташовані в нормованому інтервалі. Число терм терм-множини фазифікатора завжди повинно дорівнювати числу підінтервалів розбиття нормованого інтервалу. У разі, коли відбувається регулювання терм, ММНК не застосовується.

Алгоритм послідовного навчання

Метод адаптації з послідовним навчанням [6, 7] належить до методів послідовної ідентифікації лінійних систем у реальному часі, який заснований на принципі навчання з моделлю і в якому отримана модель імпульсної характеристики. Цей метод має збіжність трохи нижче, ніж ММНК. Його зручно застосовувати для процесів із параметрами, що повільно змінюються. Основна перевага методу послідовного навчання полягає у простоті алгоритму адаптації.

У методі послідовного навчання розглядається лінійна система з випадковим входом $x(t)$, виходом $S(t)$ і імпульсною характеристикою $g(t)$. Зв'язок між входом та виходом описується інтегралом згортки:

$$S(t) = \int_0^t g(\tau) \cdot x(t - \tau) d\tau,$$

де $x(t - \tau)$ – вимірювана величина.

У дискретній формі інтеграл згортки має вигляд

$$S_j = \sum_{i=1}^n g_i x_{j-i}. \quad (1)$$

Визначення імпульсної реакції g_i виконується шляхом ітераційних обчислень векторів $\phi_1^{(j)}, \dots, \phi_n^{(j)}$, $\forall j = n+1, n+2, \dots$, які мають, відповідно, наближатися до g_1, \dots, g_n у рівнянні (1), де j позначає номер ітерації. Тоді оцінка виходу моделі (нейрона) S_j^M при використанні $\phi_i^{(j)}$ дорівнює по аналогії з (1):

$$S_j^M = \sum_{i=1}^n \phi_i^{(j)} x_{j-i}, \quad (2)$$

де n – кількість входів (синаптичних зв'язків) нейрона.

Ввівши позначення

$$\begin{aligned} \mathbf{g} &= [g_1, \dots, g_i, \dots, g_n]^T, \\ \boldsymbol{\phi}_j &= [\phi_1^{(j)}, \dots, \phi_i^{(j)}, \dots, \phi_n^{(j)}]^T, \\ \mathbf{x}_j &= [x_{j-1}, \dots, x_{j-i}, \dots, x_{j-n}]^T, \end{aligned}$$

вирази (1) і (2) можуть бути перетворені на вигляд

$$\begin{aligned} S_j &= \mathbf{g}^T \mathbf{x}_j, \\ S_j^M &= \boldsymbol{\phi}_j^T \mathbf{x}_j. \end{aligned}$$

Помилка виходу моделі (нейрона) набуде вигляду

$$S_j - S_j^M = (\mathbf{g} - \boldsymbol{\phi}_j)^T \mathbf{x}_j = \mathbf{x}_j^T (\mathbf{g} - \boldsymbol{\phi}_j) = \mathbf{x}_j^T \Delta \boldsymbol{\phi}_j,$$

де $\Delta \boldsymbol{\phi}_j = \boldsymbol{\phi}_{j+1} - \boldsymbol{\phi}_j$ – корекція вектора $\boldsymbol{\phi}_j$ його наступної ідентифікації $\boldsymbol{\phi}_{j+1}$ з урахуванням помилки $(S_j - S_j^M)$.

Якщо припустити, що

$$\Delta \Phi_j = (S_j - S_j^M) \frac{\mathbf{x}_j}{\mathbf{x}_j^T \mathbf{x}_j},$$

то рекурентне співвідношення для обчислення оцінок вагових коефіцієнтів набуде вигляду

$$\Phi_{j+1} = \Phi_j + (S_j - S_j^M) \frac{\mathbf{x}_j}{\mathbf{x}_j^T \mathbf{x}_j}, \quad S_1^M = 0.$$

З коефіцієнтом корекції помилки рекурентне співвідношення запишеться як

$$\Phi_{j+1} = \Phi_j + C(S_j - S_j^M) \frac{\mathbf{x}_j}{\mathbf{x}_j^T \mathbf{x}_j}, \quad 0 < C < 2.$$

Метод адаптації з послідовним навчанням має високу збіжність (достатньо двох, трьох ітерацій). При цьому він може використовуватися як при лінійних, так і при нелінійних термах нейронів.

Структура штучних нейронних мереж

При формуванні структури ШНМ можна виділити два підходи. Перший підхід – це ШНМ, які призначені для вирішення завдань регулювання процесів. Вони, як правило, мають у вихідному шарі один терм із сигмоїдальною пороговою функцією, що є регулюючим сигналом. Другий підхід – це ШНМ класифікації. У вихідному шарі міститься кілька термів зі ступінчастими пороговими функціями, які містять категоріальні значення та мережі, призначені для вирішення задач класифікації.

Як приклад ШНМ регулювання може бути представленою розповсюджена комерційна програмна платформа ANFIS (Adaptive Network-based Fuzzy Inference System), яка реалізує нечітку нейронну продукційну мережу [8, 9]. ANFIS входить до складу пакета Fuzzy Logic Toolbox середовища Matlab-Simulink. Дана мережа дозволяє реалізувати нечіткий регулятор динамічних об'єктів із попереднім навчанням і подальшим набуттям властивостей адаптації. При цьому можливі модифікації мережі ANFIS в залежності від застосування алгоритмів нечіткого виведення: Сугено-Такаґи, Сугено-Такаґи-Канґа (TSK) і Ванґа-Менделя.

Структура такої ШНМ (рис. 3) містить п'ять шарів, де:

- шар № 1 виконує функцію фазифікації вхідних чітких мінливих. Кількість нейронів у шарі відповідає потужності терм-множин вхідних величин;
- шари № 2, № 3 і № 4 служать для виведення заключень нечітких правил і обчислення нормованих оцінок застосовності цих правил. Кількість елементів у даних шарах дорівнює кількості правил, що використовуються;
- шар № 5 складається з єдиного елемента, де відбувається дефазифікація результату. Виведення чіткого або лінгвістичного значення регулюючого впливу.

Налаштування мережі відбувається за допомогою комбінації ММНК і методу ЗРП. Алгоритм ЗРП налаштовує параметри антецедентів (передумов), тобто, функцій належності фазифікатора. ММНК оцінює коефіцієнти заключень правил, так як вони лінійно зв'язані з виходом мережі. Кожна ітерація процедури налаштування виконується у два етапи. На першому етапі на входи подається навчаюча вибірка та за нев'язкою між бажаною і дійсною поведінкою мережі за допомогою ММНК знаходяться оптимальні параметри вузлів четвертого шару. На другому етапі остаточна нев'язка передається з виходу мережі на входи і методом ЗРП модифікуються параметри вузлів першого шару. При цьому знайдені на попередньому етапі коефіцієнти заключення правил не змінюються. Ітераційна

процедура налаштування продовжується, поки нев'язка перебільшує раніше встановлене значення.

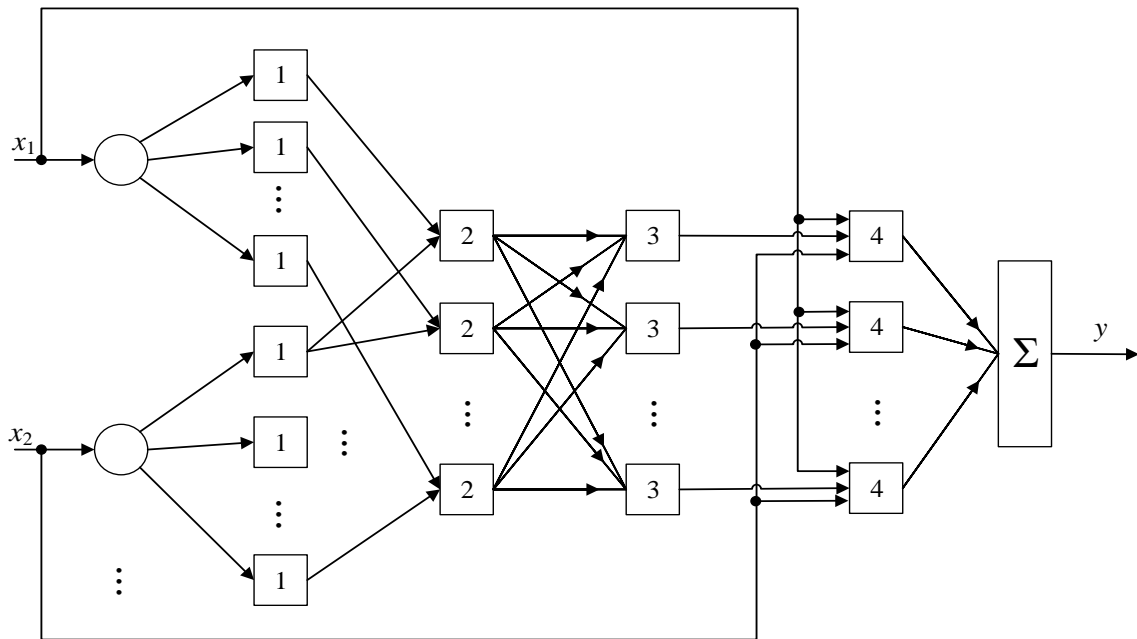


Рисунок 3 – Нечітка нейронна мережа ANFIS із застосуванням алгоритму Сугено-Такагі:
 1 – терми лінгвістичних мінливих; 2 – антецеденти; 3 – нормалізація сигналів;
 4 – заключення правил; 5 – рішення (складення вкладів усіх правил)

ШНМ ANFIS використовується при створенні автоматичних систем управління інформаційним трафіком із метою забезпечення інформаційної безпеки.

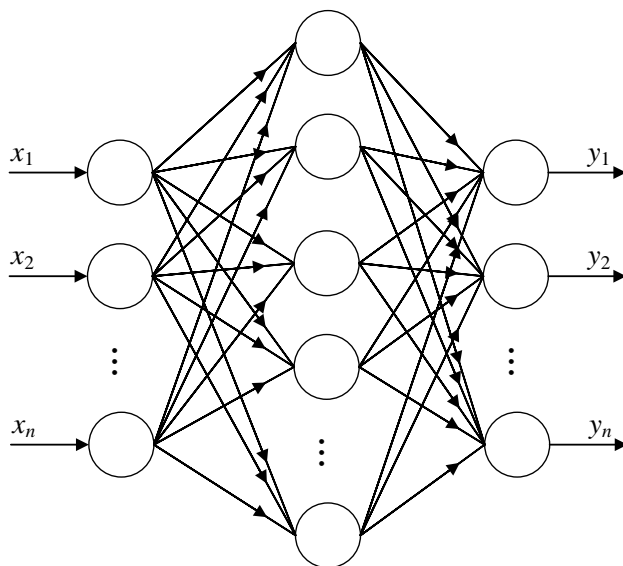


Рисунок 4 – Структура штучних нейронних мереж класифікації даних

ШНМ класифікації мають багатопшарову довільну структуру, яка визначається структурою онтологічної мережі області, що розглядається [10]. Найбільш часто для класифікації використовується архітектура ШНМ мережі прямого поширення, на входні нейрони яких подаються значення ознак об'єкта, що класифікується, а на виході формується мітка або числовий код класу. Зазвичай використовуються багатопшарові перцептрони. У таких мережах елементи вектора ознак надходять на входні нейрони та розподіляються на всі нейрони першого прихованого шару штучної нейронної мережі (рис. 4).

Правильно підібравши конфігурацію та параметри штучної нейронної мережі, можна отримати доб-

рі результати класифікації навіть у тих випадках, коли класифікатори інших типів, що працюють лише у розмірності навчальних даних, не забезпечують прийнятних результатів. Недоліком є те, що конфігурація мережі, яка найкраще апроксимує функцію поділу класів

у просторі ознак, наперед невідома. Тому доводиться підбирати її експериментально або використовувати досвід аналогічних рішень.

Для обчислення значень вагових коефіцієнтів зв'язків штучних нейронних мереж, зазвичай, використовується алгоритм зворотного розповсюдження помилки.

3. Висновки

Таким чином, штучні нейронні мережі широко використовуються при розв'язанні задач управління інформаційною безпекою. При цьому застосовуються як ШНМ класифікації даних (розпізнавання багатопараметричних об'єктів), так і нечіткі продукційні нейронні мережі для регулювання інформаційних процесів.

Методи побудови ШНМ відрізняються за алгоритмами обчислення вагових коефіцієнтів мережі на етапі навчання та за структурою мережі.

Найбільш розповсюдженим алгоритмом налаштування штучних нейронних мереж є алгоритм зворотного розповсюдження помилки. Для прискорення обчислень застосовуються алгоритм адаптації Уїдроу-Хоффа та алгоритм модифікованого методу найменших квадратів. Однак обмеженням для застосування цих алгоритмів є умова лінійності моделі об'єкта (процесу) та відсутність регулювання терм фазифікованого простору.

Для нелінійних моделей застосовуються генетичний алгоритм та метод адаптації з послідовним навчанням. Недоліком цих методів є підвищена обчислювальна складність та отримання локального оптимуму замість глобального, що є характерним для нелінійних систем.

Структура ШНМ регулювання процесів залежить від логіки взаємодії правил виведення заключень щодо формування регулюючого впливу та має єдиний терм у вихідному шарі мережі. Структура ШНМ класифікації даних має структуру, яка апроксимує функцію поділу класів у просторі ознак. У початковий період дослідження така функція невідома. Тому доводиться структуру мережі підбирати експериментально або використовувати досвід аналогічних рішень.

СПИСОК ДЖЕРЕЛ

1. Руденко О.Г., Бодяньський Є.В. Штучні нейронні мережі: навч. посіб. Харків: ТОВ «Компанія СМІТ», 2006. 404 с.
2. Brain Maker Professional, Neural Network Simulation Software. User Guide and Reference Manual. California Scientific Software, 1993. 496 p.
3. Кононюк А.Ю. Нейронні мережі і генетичні алгоритми. К.: Корнійчук, 2008. 446 с.
4. Kryvyy R., Tkachenko S., Karkuljovskyy V. Analysis of Frameworks for Developing Genetic Algorithms. *MEMSTECH'2011: Proc. of the VII-th International Conference*. Lviv – Polyana, 2011. P. 209–210.
5. Дубровін В.І., Субботін С.О. Методи оптимізації та їх застосування в задачах навчання нейронних мереж: навч. посіб. Запоріжжя: ЗНТУ, 2003. 136 с.
6. Plett G.L. Adaptive inverse control of linear/nonlinear systems using dynamic neural networks. *IEEE Trans. Neural Networks*. 2003. Vol. 5, N 2. P. 360–376.
7. Субботін С.О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень. Запоріжжя: ЗНТУ, 2008. 341 с.
8. Ishibuchi H., Kwon K., Tanaka H. Implementation of fuzzy IF-THEN rules by fuzzy neural networks with fuzzy weights. *Proc. of EUFIT'93 Conference*. Aachen, Germany, 1993. P. 209–215.
9. Ishibuchi H., Fujioka R., Tanaka H. An architecture of neural networks for input vectors of fuzzy numbers. *Proc. IEEE Intern. Conf on Fuzzy Systems*. San Diego, USA, 1992. P. 1293–1300
10. Глибовець М.М., Олецкий О.В. Системи штучного інтелекту. К.: КМ Академія, 2002. 366 с.

Стаття надійшла до редакції 10.04.2023