

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління  
(повна назва)

Кафедра електронних обчислювальних машин  
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

Рівень вищої освіти другий (магістерський)

Методи коригування нелінійності функції  
перетворення засобів вимірювань за допомогою  
штучних нейронних мереж  
(тема)

Виконав:

здобувач 2 року навчання,  
групи СПМ-23-4

Сергій АВАКІН  
(власне ім'я, прізвище)

Спеціальність 123 «Комп'ютерна інженерія»  
(код і повна назва спеціальності)

Тип програми освітньо-наукова  
(освітньо-професійна або освітньо-наукова)

Освітня програма Системне програмування  
(повна назва освітньої програми)

Керівник: доц. Олег ЗАПОРОЖЕЦЬ  
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ЕОМ

Андрій КОВАЛЕНКО  
(підпис) (власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ комп'ютерної інженерії та управління \_\_\_\_\_

Кафедра \_\_\_\_\_ електронних обчислювальних машин \_\_\_\_\_

Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_

Спеціальність \_\_\_\_\_ 123 «Комп'ютерна інженерія» \_\_\_\_\_  
(код і повна назва)

Тип програми \_\_\_\_\_ освітньо-наукова \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)

Освітня програма \_\_\_\_\_ Системне програмування \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві \_\_\_\_\_ Авакіну Сергію Миколайовичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи Методи коригування нелінійності функції перетворення засобів вимірювань за допомогою штучних нейронних мереж

затверджена наказом по університету від “ 21 ” квітня 2025 р. № 296 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 16 червня 2025 р.

3. Вхідні дані до роботи 1) вид нелінійної функції перетворення: синусоїдальна, степенева, гіперболічна, дробово-лінійна I та II, показникова I та II, логарифмічна I та II; 2) архітектура нейромережі: багатошаровий персептрон, радіально-базисна

4. Перелік питань, що потрібно опрацювати у роботі \_\_\_\_\_

1) загальна характеристика та властивості нейронних мереж;

2) багатошарові персептрони;

3) мережі радіальних базисних функцій;

4) побудова функцій перетворення засобів вимірювання;

5) коригування нелінійності функції перетворення засобів вимірювання за допомогою штучних нейронних мереж.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) \_\_\_\_\_

Слайд-презентація – 18 слайдів \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1 )

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Огляд методів апроксимації нелінійних функцій та коригування нелінійності функції перетворення засобів вимірювань	22.04.25–25.04.25	
2	Аналіз архітектур штучних нейронних мереж та методів їх навчання	26.04.25–30.04.25	
3	Розробка нейромережевої системи коригування нелінійності функції перетворення засобів вимірювань	01.05.25–15.05.25	
4	Проведення експериментальних досліджень	16.05.25–25.05.25	
5	Оформлення матеріалів кваліфікаційної роботи	26.05.25–04.06.25	
6	Подання кваліфікаційної роботи керівникові та її попередній захист	05.06.25–10.06.25	
7	Подання кваліфікаційної роботи на рецензування	11.06.25–13.06.25	

Дата видачі завдання 21 квітня 2025 р.

Здобувач \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_  
(підпис)

доц. Олег ЗАПОРОЖЕЦЬ \_\_\_\_\_  
(посада, власне ім'я, прізвище)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 88 с., 27 рис., 3 табл., 2 дод., 28 джерел.

ФУНКЦІЯ ПЕРЕТВОРЕННЯ, КОРИГУВАННЯ, ЛІНЕАРИЗАЦІЯ, ШТУЧНА НЕЙРОННА МЕРЕЖА, МЕРЕЖА РАДІАЛЬНИХ БАЗИСНИХ ФУНКЦІЙ, БАГАТОШАРОВИЙ ПЕРСЕПТРОН, МЕТОД НАВЧАННЯ.

Метою кваліфікаційної роботи є розробка та дослідження властивостей адаптивної системи коригування нелінійності функції перетворення засобів вимірювань на базі штучної нейронної мережі.

У ході виконання кваліфікаційної роботи розглянуто загальну характеристику та властивості штучних нейронних мереж, їх класифікацію та області застосування, архітектуру та методи навчання багатошарового персеプトрона та радіально-базисної нейромережі.

Розглянуто методи апроксимації нелінійних функцій перетворення засобів вимірювань. Запропоновано узагальнену структуру адаптивної системи коригування нелінійності функції перетворення засобу вимірювання за допомогою штучних нейронних мереж, а саме багатошарового персепторона та радіально-базисної нейромережі.

Для вивчення властивостей запропонованих нейромережових систем коригування функції перетворення засобів вимірювань застосовувалось комп'ютерне моделювання в середовищі MATLAB з використанням бібліотеки Neural Network Toolbox. У процесі моделювання досліджувався вплив виду нелінійності на якість коригування функції перетворення.

## ABSTRACT

Master's thesis: 88 pages, 27 figures, 3 tables, 2 appendices, 28 sources.

TRANSFORMATION FUNCTION, CORRECTION, LINEARIZATION, ARTIFICIAL NEURAL NETWORK, RADIAL BASIS FUNCTION NETWORK, MULTILAYER PERCEPTRON, LEARNING METHOD.

The major goal of this thesis is to develop and study the properties of an adaptive system for correcting the nonlinearity of the conversion function of measuring instruments based on an artificial neural network.

During the qualification work, the general characteristics and properties of artificial neural networks, their classification and areas of application, architecture and training methods of a multilayer perceptron and a radial basis neural network were considered.

Methods for approximating nonlinear transformation functions of measuring instruments are considered. A generalized structure of an adaptive system for correcting nonlinearity of the transformation function of a measuring instrument using artificial neural networks, namely a multilayer perceptron and a radial basis neural network, is proposed.

To study the properties of the proposed neural network systems for adjusting the conversion function of measuring instruments, computer simulation was used in the MATLAB environment using the Neural Network Toolbox library. During the simulation, the influence of the type of nonlinearity on the quality of the adjustment of the transformation function was investigated.

## ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ .....	8
ВСТУП .....	9
1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА ВЛАСТИВОСТІ НЕЙРОННИХ МЕРЕЖ.....	11
1.1 Поняття про штучні нейронні мережі.....	11
1.2 Математична модель штучного нейрона.....	12
1.3 Класифікація нейронних мереж .....	15
1.3.1 Одношарова нейромережа .....	17
1.3.2 Багатошарова нейромережа .....	18
1.3.3 Рекурентна нейромережа .....	19
1.4 Области застосування нейронних мереж .....	21
2 БАГАТОШАРОВІ ПЕРСЕПТРОНИ .....	23
2.1 Структура та властивості багатошарового персептрона.....	23
2.2 Навчання багатошарового персептрона.....	25
2.3 Метод зворотного поширення помилки .....	29
2.4 Алгоритм Левенберга-Марквардта .....	32
3 МЕРЕЖІ РАДІАЛЬНИХ БАЗИСНИХ ФУНКЦІЙ .....	35
3.1 Архітектура радіально-базисної мережі .....	35
3.2 Навчання радіально-базисних нейромереж.....	37
4 ПОБУДОВА ФУНКЦІЙ ПЕРЕТВОРЕННЯ ЗАСОБІВ ВИМІРЮВАННЯ .....	40
4.1 Функція перетворення засобу вимірювання та підходи до її визначення .....	40
4.2 Побудова лінійної функції перетворення методом найменших квадратів.....	43
4.3 Побудова нелінійних залежностей методом найменших квадратів .....	46
4.4 Апроксимація функцій за допомогою штучних нейронних мереж.....	49

5 КОРИГУВАННЯ НЕЛІНІЙНОСТІ ФУНКЦІЇ ПЕРЕТВОРЕННЯ ЗАСОБІВ ВИМІРЮВАННЯ ЗА ДОПОМОГОЮ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ.....	55
5.1 Постановка задачі дослідження.....	55
5.2 Пристрій корекції на базі багат шарового персептрона.....	59
5.3 Пристрій корекції на базі радіально-базисної нейромережі.....	62
5.4 Результати експериментальних досліджень.....	64
ВИСНОВКИ.....	72
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	74
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	77
ДОДАТОК Б Текст програми моделювання .....	87

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

МНК – метод найменших квадратів

ШНМ – штучна нейронна мережа

DBSCAN – просторова кластеризація застосунків із шумом на основі щільності (англ., Density-Based Spatial Clustering Of Applications With Noise)

LMA – алгоритм Левенберга-Марквардта (англ., Levenberg-Marquardt Algorithm)

MLP – багатошаровий перцептрон (англ., MultiLayer Perceptron)

PCA – метод головних компонентів (англ., Principal Component Analysis)

RBFN – мережа радіальних базисних функцій (англ., Radial Basis Function Network)

## ВСТУП

Останні десятиліття характеризуються стрімким розвитком інформаційних технологій, які дозволили радикально і докорінно змінити не тільки традиційний уклад людського життя, а й цілі галузі економіки у більшості розвинених країн світу. Безумовним досягненням наукової думки стали успіхи в розробці штучного інтелекту. Дослідження в області штучних нейронних мереж, глибокого навчання, фаззі-систем дозволили реалізувати достатньо складні методи математичної обробки та аналізу значних масивів даних та розв'язувати цілий ряд задач класифікації, розпізнавання образів, оптимізації, ідентифікації, діагностики, управління, апроксимації складних функціональних залежностей тощо.

Галузь метрології та вимірювальної техніки теж не залишається осторонь і намагається активно застосовувати більш ефективні методи обробки інформації, які дозволять підвищити достовірність і точність результатів вимірювань. Роль таких методів ще більше зростає з ускладненням вимірювань і підвищенням вимог до їх точності.

Нові галузі прикладної математики, такі як теорія штучних нейронних мереж, нечітка логіка, робастна та непараметрична статистика, інтервальний аналіз, вейвлет-аналіз тощо, дають математичний апарат для розв'язання тих задач, для яких класичні методи обробки даних є непридатними або малоефективними. До створення нових методів обробки вимірювальної інформації спонукають потреби сучасної практики вимірювань, для якої характерним є ускладнення вимірювальних задач та засобів вимірювань.

Розширення можливостей інтелектуального опрацювання даних та результатів вимірювань тісно пов'язане з розвитком методології вимірювань, удосконаленням математичних методів, а також широким впровадженням засобів обчислювальної техніки у вимірювальні пристрої, включаючи мікропроцесори і мікроконтролери.

Досягнення в області сучасних методів обробки сигналів дозволяють підвищити точність вимірювань і реалізувати досить складні вимірювальні процедури. Одним із перспективних напрямів розвитку інтелектуальних технологій управління та обробки інформації є штучні нейронні мережі. Нейронні мережі використовуються для вирішення складних завдань, які вимагають аналітичних обчислень подібних до тих, які робить людський мозок. Сюди відносяться розпізнавання образів, класифікація (кластеризація), прогнозування, апроксимація, прийняття рішень та управління, стиснення даних та асоціативна пам'ять.

На сьогоднішній день існує досить багато досліджень нейронних мереж. У роботах Вассермана [1], Хайкіна [2], Рохаса [3], Бодянського та Руденко [4, 5] досить докладно описані теоретичні основи нейронних мереж і практичні рекомендації їх застосування. Основними перевагами нейронних мереж перед традиційними обчислювальними алгоритмами є здатність до навчання, адаптація до змін навколишнього середовища, гнучкість структури, стійкість до відмов. Також нейронні мережі є базовим інструментом машинного навчання у штучному інтелекті [6].

Останнім часом нейромережеві технології дедалі частіше знаходять своє застосування у практиці вимірювань. Розробка нейромережевих моделей вимірювальних пристроїв є одним із перспективних напрямів інтелектуалізації засобів вимірювань. Такі моделі дозволять значно покращити метрологічні характеристики та ефективність існуючих вимірювальних перетворювачів.

Метою кваліфікаційної роботи є дослідження адаптивної системи коригування функції перетворення засобу вимірювання, яка буде використовувати у якості пристрою корекції штучну нейронну мережу та забезпечуватиме лінійність функції перетворення в усьому діапазоні можливих значень вимірюваної вхідної величини.

# 1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА ВЛАСТИВОСТІ НЕЙРОННИХ МЕРЕЖ

## 1.1 Поняття про штучні нейронні мережі

Штучна нейронна мережа (ШНМ, *англ.* Artificial Neural Network) – це математична модель, а також її програмна та апаратна реалізація, побудовані за принципом функціонування біологічних нейронних мереж – мереж нервових клітин живого організму. Це поняття виникло при вивченні процесів, які відбуваються в мозку та при намаганні змоделювати ці процеси. Першою такою спробою були нейронні мережі У. Маккалока та У. Піттса. Після розробки алгоритмів навчання отримувані моделі стали використовуватися в практичних цілях: в задачах класифікації, прогнозування, розпізнавання образів, діагностики, керування тощо.

ШНМ є системою з'єднаних між собою взаємодіючих вузлів (штучних нейронів). Ці штучні нейрони зазвичай є доволі простими перетворювачами сигналів. Кожен нейрон подібної мережі має справу лише з сигналами, які він отримує від інших нейронів, і сигналами, які він періодично надсилає іншим нейронам мережі. Проте, якщо з'єднати такі нейрони в достатньо велику мережу з керованою взаємодією, разом вони здатні виконувати досить складні завдання.

Однією з головних переваг нейронних мереж перед традиційними алгоритмами обробки даних є здатність навчання складній поведінці, що дає можливість розв'язувати за допомогою ШНМ різноманітні задачі, для вирішення яких традиційні методи і підходи є неефективними. Навчання полягає у знаходженні оптимальних значень вагових коефіцієнтів зв'язків між нейронами. В процесі навчання ШНМ здатні виявляти складні приховані залежності між вхідними та вихідними даними, а також здійснювати узагальнення. Це означає, що ретельно навчена нейромережа зможе

повернути правильний результат на підставі даних, які були відсутні в датасеті (навчальній вибірці), а також неповних або «зашумлених», частково спотворених даних [4-7].

## 1.2 Математична модель штучного нейрона

Основним обчислювальним елементом ШНМ є штучний (формальний) нейрон, який по суті виступає дуже спрощеною моделлю біологічного нейрона. Це примітивний обчислювальний пристрій, що має декілька входів і один вихід (рис. 1.1).

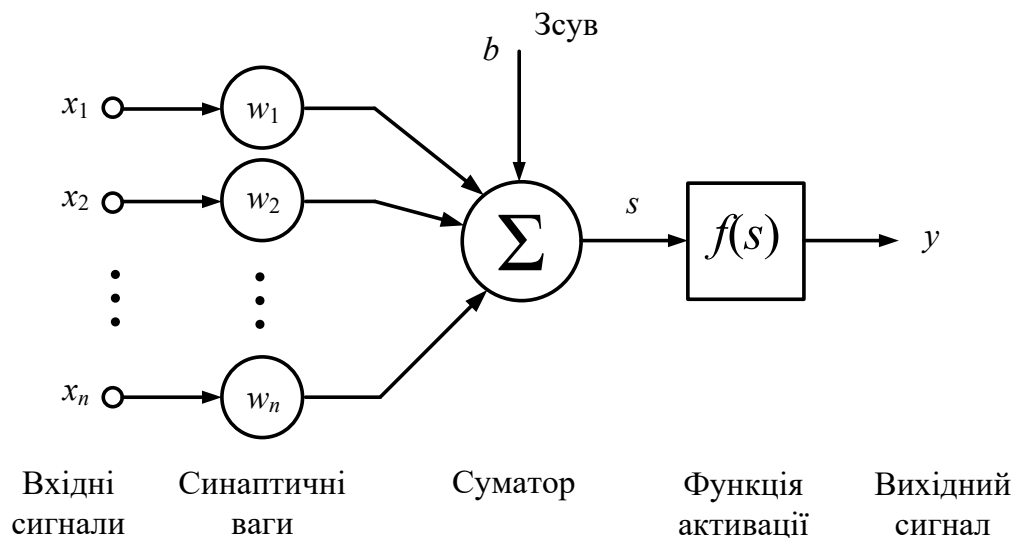


Рисунок 1.1 – Структура штучного нейрона

Формальний нейрон реалізує скалярну функцію векторного аргументу. На вхід подається вектор вхідних сигналів  $x = (x_1, x_2, \dots, x_n)^T$ , кожен з яких множиться на свій ваговий коефіцієнт (синаптичну вагу)  $w_i, i = 1, 2, \dots, n$ . Далі обчислюється зважена сума вхідних сигналів  $s$ , яка є аргументом (вхідною величиною) деякої нелінійної функції  $y = f(s)$ . Ця функція називається функцією активації або передаточною функцією нейрона і вона, власне кажучи, формує вихідний сигнал нейрона  $y$ .

Таким чином, математична модель нейрона описується наступними співвідношеннями

$$s = \sum_{i=1}^n w_i x_i + b, \quad (1.1)$$

$$y = f(s), \quad (1.2)$$

де  $w_i$  – вага синапса ( $i = 1, 2, \dots, n$ );

$b$  – значення зсуву;

$s$  – результат підсумовування;

$x_i$  – компонент вхідного вектору (вхідний сигнал) ( $i = 1, 2, \dots, n$ );

$y$  – вихідний сигнал нейрона;

$n$  – число входів нейрона;

$f$  – нелінійне перетворення (функція активації).

У загальному випадку вхідний сигнал, синаптичні вагові коефіцієнти і значення зсуву можуть набувати дійсних значень. Вихідний сигнал нейрона  $y$  визначається обраною функцією активації й може бути як дійсним, так і цілим. У багатьох практичних задачах входи, вагові коефіцієнти і зсуви можуть мати лише деякі фіксовані значення.

Таким чином, формальний нейрон повністю описується своїми ваговими коефіцієнтами  $w_i$  і функцією активації  $f(s)$ . При подачі на вхід вектору сигналу  $x = (x_1, x_2, \dots, x_n)^T$  нейрон формує деяке числове значення  $y$  на виході.

Описаний вище елемент ШНМ можна вважати суттєво спрощеною математичною моделлю біологічних нейронів – клітин нервової системи людини й тварин. Щоб підкреслити розходження біологічних і математичних нейронів, останні іноді називають нейроподібними елементами або формальними нейронами.

Приклади функцій активації нейронів наведено у табл. 1.1.

Таблиця 1.1 – Функції активації нейронів

Назва	Формула	Область значень
Порогова (Хевісайда)	$f(s) = \begin{cases} 0, & s < 0, \\ 1, & s \geq 0 \end{cases}$	(0, 1)
Знакова (сигнатурна)	$f(s) = \begin{cases} 1, & s > 0, \\ -1, & s < 0 \end{cases}$	(-1, 1)
Сигмоїдальна (логістична)	$f(s) = \frac{1}{1 + e^{-s}}$	(0, 1)
Напівлінійна	$f(s) = \begin{cases} s, & s > 0, \\ 0, & s \leq 0 \end{cases}$	(0, $\infty$ )
Лінійна	$f(s) = s$	( $-\infty, \infty$ )
Радіально-базисна (Гауса)	$f(s) = \exp(-s^2)$	(0, 1)
Напівлінійна з насиченням	$f(s) = \begin{cases} 0, & s \leq 0, \\ s, & 0 < s < 1, \\ 1, & s \geq 1 \end{cases}$	(0, 1)
Лінійна з насиченням	$f(s) = \begin{cases} -1, & s \leq -1, \\ s, & -1 < s < 1, \\ 1, & s \geq 1 \end{cases}$	(-1, 1)
Гіперболічний тангенс (сигмоїдальна)	$f(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}}$	(-1, 1)
Трикутна	$f(s) = \begin{cases} 1 -  s , &  s  \leq 1, \\ 0, &  s  > 1 \end{cases}$	(0, 1)

Розглянута вище проста модель формального нейрона не відображає багатьох властивостей свого біологічного аналога (наприклад, не враховуються затримки у часі, які впливають на динамічні властивості системи). Проте, незважаючи на вказані обмеження, обчислювальні пристрої, побудовані із таких нейронів, мають властивості, що дуже нагадують біологічну систему.

### 1.3 Класифікація нейронних мереж

Розрізняють два великих класи нейромережових архітектур: мережі прямого поширення (feed forward neural network) та рекурентні мережі (зі зворотними зв'язками), які у свою чергу поділяються ще на декілька категорій (рис. 1.2).

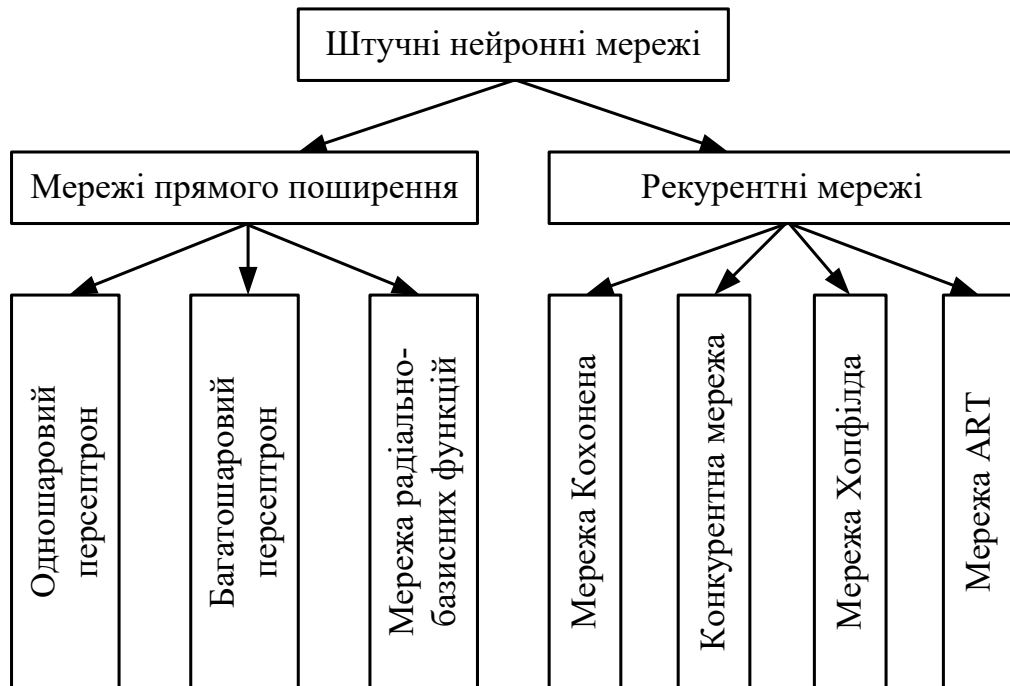


Рисунок 1.2 – Архітектури нейронних мереж

Крім архітектури, можна класифікувати ШНМ за різними ознаками, серед яких можна виділити:

- метод навчання – кероване навчання або навчання з учителем (supervised learning), некероване навчання або навчання без учителя (unsupervised learning), навчання з підкріпленням (reinforcement learning);

- статичні і динамічні мережі. Зміна параметрів статичної ШНМ відбувається за певним алгоритмом у процесі навчання, після завершення навчання параметри мережі не змінюються. В динамічних нейромережах навчання і функціонування залежить від часу, тобто є динамічним процесом;

- бінарні і аналогові нейронні мережі. В бінарних вихід кожного нейрона може набувати тільки двох значень – логічний нуль та логічна одиниця, а у аналогових виходи нейронів можуть набувати неперервних значень;

- асинхронні та синхронні нейронні мережі. В асинхронних у заданий момент часу свій стан змінює тільки один нейрон, а у синхронних – декілька (зазвичай шар нейронів).

Більш розлогу класифікацію ШНМ наведено на рис. 1.3.

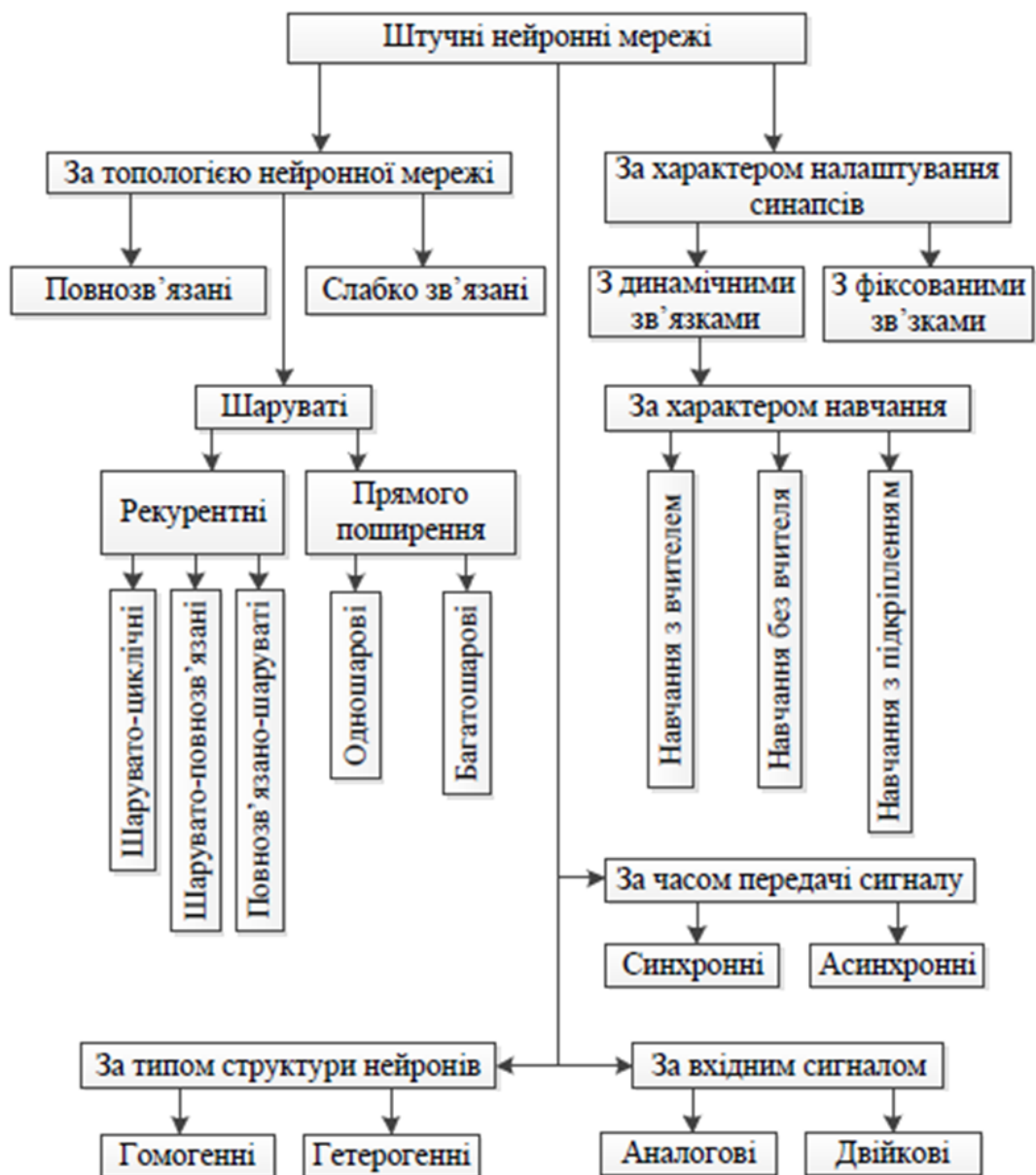


Рисунок 1.3 – Класифікація нейронних мереж

### 1.3.1 Одношарова нейромережа

Архітектуру одношарової нейронної мережі прямого поширення наведено на рис. 1.4. Вхідний шар утворений самим вектором вхідних сигналів  $x = (x_1, x_2, \dots, x_m)^T$  і не виконує ніяких трансформацій, тому його не беруть до уваги. Основні обчислення здійснює вихідний шар нейронів. Кожен нейрон цього вихідного шару зв'язаний з кожним компонентом вектору вхідних сигналів і вага зв'язку між  $i$ -м компонентом вектору вхідних сигналів та  $j$ -м нейроном позначається як  $w_{ij}$ .

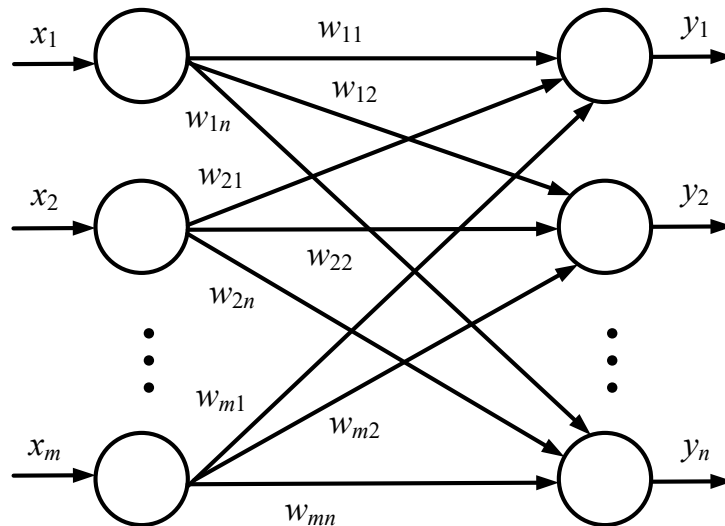


Рисунок 1.4 – Одношарова нейронна мережа

Синаптичні вагові коефіцієнти такої мережі зручно описувати у вигляді матриці  $W$ . Тоді рівняння перетворення одношарової нейронної мережі у матричному вигляді виглядає наступним чином

$$y = F(W^T x), \quad (1.3)$$

де  $y = (y_1, y_2, \dots, y_n)^T$  – вектор вихідних сигналів нейромережі;

$x = (x_1, x_2, \dots, x_m)^T$  – вектор вхідних сигналів нейромережі;

$$W = \begin{pmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \dots & w_{mn} \end{pmatrix} \quad \text{– матриця синаптичних вагових}$$

коефіцієнтів;

$F(\bullet)$  – функція активації.

### 1.3.2 Багат шарова нейромережа

Розвитком описаної вище архітектури одношарової ШНМ стала багат шарова мережа. Дослідження показали, що багат шарові нейромережі мають набагато більші функціональні можливості порівняно з одношаровими і є одним з основних базових елементів штучного інтелекту та машинного навчання.

В багат шарових мережах шари нейронів розташовані каскадно, тобто виходи нейронів одного шару є входами нейронів наступного шару. Приклад такої мережі зображено на рис. 1.5.

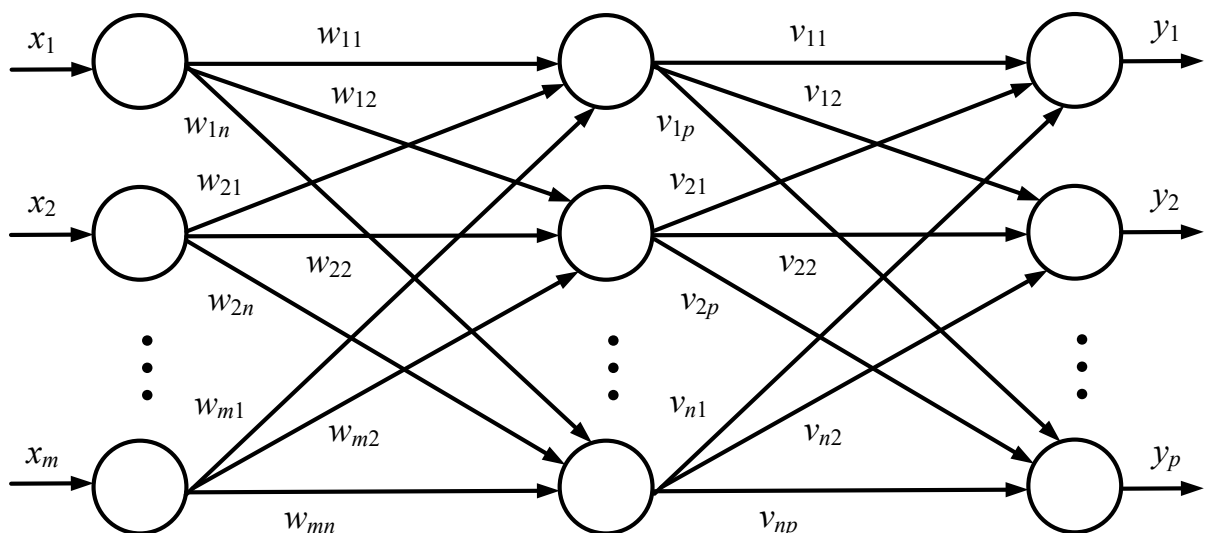


Рисунок 1.5 – Двошарова нейронна мережа

Математичні співвідношення, які описують двошарову нейромережу, мають вигляд

$$y = \Phi(V^T F(W^T x)), \quad (1.4)$$

де  $y = (y_1, y_2, \dots, y_p)^T$  – вектор вихідних сигналів нейромережі;

$x = (x_1, x_2, \dots, x_m)^T$  – вектор вхідних сигналів нейромережі;

$$W = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mn} \end{pmatrix} \quad - \quad \text{матриця синаптичних вагових}$$

коефіцієнтів прихованого шару;

$$V = \begin{pmatrix} v_{11} & v_{12} & \cdots & v_{1p} \\ v_{21} & v_{22} & \cdots & v_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n1} & v_{n2} & \cdots & v_{np} \end{pmatrix} \quad - \quad \text{матриця синаптичних вагових коефіцієнтів}$$

вихідного шару;

$F(\bullet), \Phi(\bullet)$  – функції активації нейронів прихованого та вихідного шару відповідно.

### 1.3.3 Рекурентна нейромережа

В мережах прямого поширення (feed forward neural networks) сигнали проходять в одному напрямку від входу до виходу. Разом з тим існує клас нейромереж, які мають зворотні зв'язки від виходів до входів, вони називаються рекурентними мережами (recurrent neural networks). На відміну від мереж прямого поширення, які не мають пам'яті, тобто вихідний сигнал повністю визначається вхідним сигналом і значенням вагових коефіцієнтів зв'язків, рекурентні нейромережі мають динамічні властивості, тобто

вихідний сигнал залежить від попередніх значень вхідного сигналу та внутрішнього стану системи. Приклад такої мережі наведено на рис. 1.6.

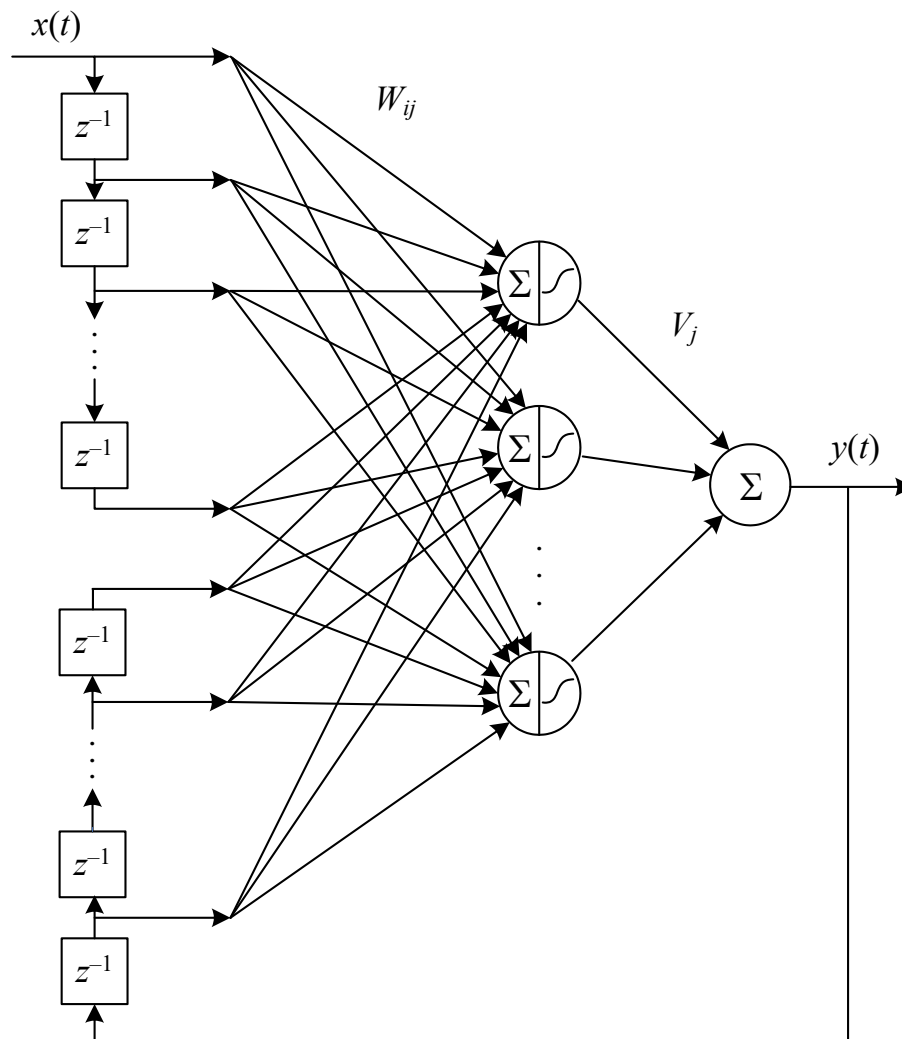


Рисунок 1.6 – Нейронна мережа зі зворотніми зв'язками

Основною перевагою рекурентних ШНМ є можливість побудови на їх основі моделей нелінійних динамічних об'єктів, що значно розширює можливості застосування таких обчислювальних структур. До недоліків слід віднести можливість виникнення нестійких режимів роботи, що призводить до колапсу або неможливості навчання. Крім того, алгоритми навчання рекурентних мереж є більш складними, ніж алгоритми навчання мереж прямого поширення.

## 1.4 Области застосування нейронних мереж

Штучні нейронні мережі найчастіше використовуються для розв'язання таких задач:

- класифікація та розпізнавання образів. Образами можуть виступати різні за своєю природою об'єкти – зображення, літери алфавіту, зразки аудіо чи відео тощо. Використовується навчання з учителем, коли нейромережі пропонуються зразки образів у вигляді вектору значень ознак та вказується, до якого класу вони належать. Після закінчення навчання можна пред'являти мережі невідомі раніше образи і отримати відповідь про приналежність до певного класу [8].

Кількість нейронів у вихідному шарі такої мережі зазвичай дорівнює кількості визначених класів. Кожен вихідний нейрон асоціюється з одним із класів. Коли мережі пред'являється новий образ для класифікації, на одному з її виходів повинен з'явитися «1», а на всіх інших виходах – «0»;

- прийняття рішень і керування. Ця задача є по суті задачею класифікації. Необхідно класифікувати ситуації, характеристики яких надходять на вхід нейронної мережі. На виході мережі при цьому повинна з'явитися ознака прийнятого рішення. При цьому вхідними сигналами є різні критерії опису стану керованої системи;

- кластеризація. Під кластеризацією мається на увазі розбивка множини деяких об'єктів на класи, при цьому ні кількість, ні ознаки класів як правило заздалегідь не відомі. Після навчання така мережа може визначати приналежність вхідного сигналу до одного з класів. Також мережа може показувати, що вхідний сигнал не належить до жодного з виділених класів – це є ознакою присутності нових даних, яких не було у навчальній вибірці. Таким чином, ШНМ здатна виявляти нові, невідомі раніше класи сигналів. Кластеризацію здійснюють, наприклад, нейронні мережі Кохонена;

- прогнозування. Здатність ШНМ до прогнозування прямо впливає із її здатності до узагальнення й виділення прихованих залежностей між

вхідними та вихідними даними. Після навчання мережа здатна передбачити майбутні значення деякої послідовності на основі декількох попередніх значень;

- апроксимація. Нейронні мережі є універсальними апроксиматорами та можуть з будь-якою наперед заданою точністю відтворювати складні нелінійні функціональні залежності. При цьому можуть використовуватись різні функції активації нейронів [9, 10];

- стискання даних і асоціативна пам'ять. Здатність нейромереж до виявлення взаємозв'язків між різними параметрами дає можливість подати більш компактно дані великої розмірності, якщо ці дані тісно взаємозв'язані один з одним. Зворотний процес – відновлення вихідного набору даних із частини інформації – називається асоціативною пам'яттю. Асоціативна пам'ять також дозволяє відновлювати початковий образ або сигнал із зашумлених або пошкоджених вхідних даних.

## 2 БАГАТОШАРОВІ ПЕРСЕПТРОНИ

### 2.1 Структура та властивості багатошарового перцептрона

Багатошарова нейронна мережа прямого поширення, або багатошаровий перцептрон (Multilayer Perceptron, MLP) складається з декількох послідовно з'єднаних між собою шарів формальних нейронів: вхідного, прихованих та вихідного. Нейрони одного шару не зв'язані між собою, виходи нейронів  $n$ -го шару надходять на входи нейронів наступного  $n + 1$ -го шару (рис. 2.1).

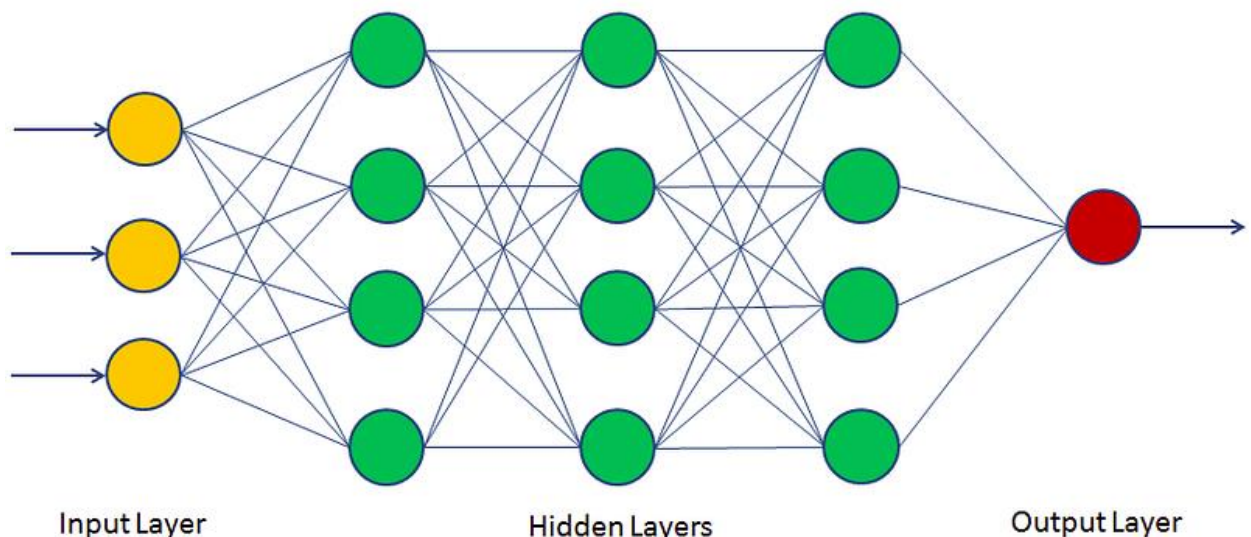


Рисунок 2.1 – Багатошаровий перцептрон

Першим йде шар, який має назву вхідного або розподільного (Input Layer). Нейрони цього шару, які теж називають вхідними, приймають елементи вхідного сигналу (вектору ознак) та розподіляють їх на нейрони наступного шару. Обробка даних нейронами вхідного шару не здійснюється, тому його іноді не беруть в розрахунок при визначення загальної кількості шарів MLP.

Останній шар мережі має назву вихідного (Output Layer), нейрони цього шару формують вихідний векторний сигнал (результат роботи) мережі.

Між вхідним та вихідним шарами MLP розташовуються один або декілька прихованих або проміжних шарів (Hidden Layers). Нейрони прихованих шарів можуть мати різні функції активації. Саме ці нейрони і виконують основні, іноді досить складні, перетворення сигналів і забезпечують отримання потрібного результату на виході нейромережі.

На сьогодні багатошарова архітектура ШНМ є однією з найбільш поширених та добре досліджених, для неї розроблено різноманітні алгоритми навчання. Багатошарові мережі можуть відтворювати функціональні залежності будь-якого ступеня складності, а кількість шарів та нейронів у кожному з них визначають складність такої функції. Конфігурація MLP однозначно визначається кількістю шарів, кількістю нейронів у кожному шарі та функціями активації нейронів.

На вхід кожного нейрона подаються всі вихідні сигнали нейронів попереднього шару, а його вихідний сигнал поступає на входи всіх нейронів наступного шару. Кожен нейрон має свій унікальний набір (вектор) синаптичних вагових коефіцієнтів, за допомогою яких посилюються або послаблюються відповідні вхідні сигнали даного нейрона.

Для вхідних нейронів доволі часто використовується лінійна або напівлінійна функція активації (див. табл. 1.1). Нейрони прихованого шару зазвичай мають порогову або сигмоїдальну (логістичну) функції активації, які є більш зручними для розв'язання задач класифікації або апроксимації. Вихідні нейрони у більшості випадків не використовують функцію активації, тобто просто формують зважену суму виходів нейронів передостаннього шару, або ж використовують порогову чи логістичну функції, коли мова йде про класифікацію у просторі вхідних ознак.

Таким чином, при вирішенні конкретних практичних задач за допомогою MLP виникає необхідність вибору вказаних параметрів нейромережі. Узагальнений алгоритм вирішення такої задачі [11] наведено на рис. 2.2.

### Алгоритм вибору параметрів нейронної мережі.

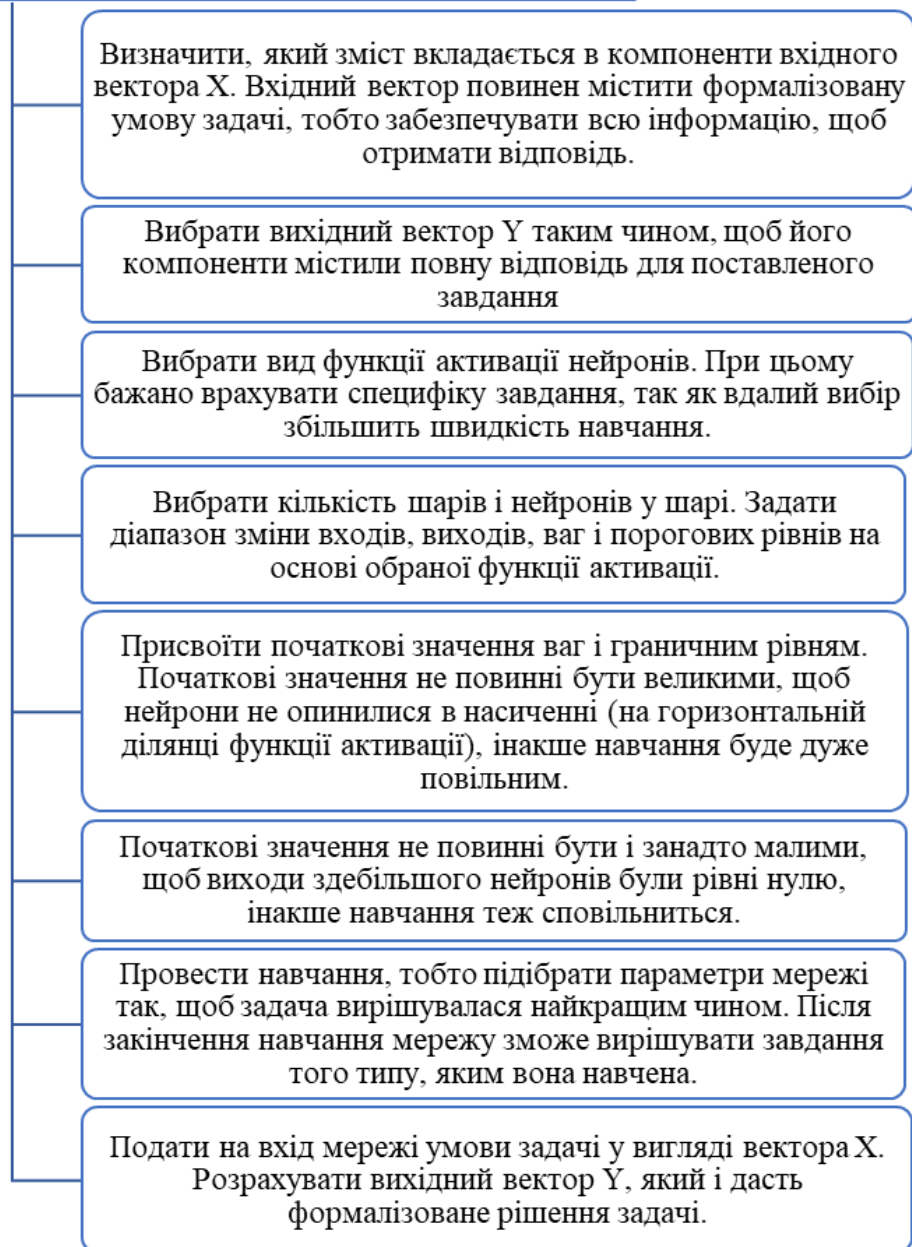


Рисунок 2.2 – Алгоритм вибору параметрів ШНМ

## 2.2 Навчання багатозарового персептрона

Найбільш цікавою і цінною властивістю нейромереж є їх здатність до навчання. Подібно біологічним організмам ШНМ можуть адаптуватися, тобто цілеспрямовано змінювати власну поведінку з метою найкращого вирішення поставленої перед ними задачі.

У процесі навчання багатошарова ШНМ здатна виявляти складні залежності між вхідними та вихідними векторами ознак та здійснювати узагальнення. Саме навчання полягає у цілеспрямованій зміні значень синаптичних вагових коефіцієнтів нейронів з метою досягнення мінімуму деякої наперед заданої цільової функції, яка найчастіше є квадратичною функцією помилки мережі. Для навчання використовується так звана навчальна вибірка або датасет. Узагальнено процес навчання нейромережі наведено на рис. 2.3. На вхід нейромережі послідовно подаються приклади навчальної вибірки, здійснюється розрахунок вихідного сигналу та помилки мережі і проводиться коригування значень вагових коефіцієнтів нейронів. Коли середньоквадратична похибка стає досить малою, мережа вважається навченою і процедура навчання зупиняється.

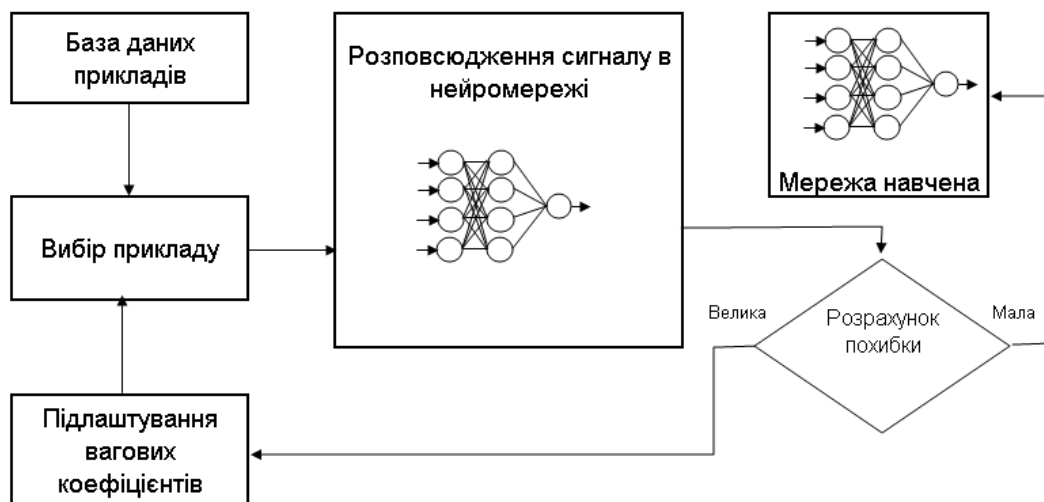


Рисунок 2.3 – Процес навчання ШНМ

На сьогодні застосовують три парадигми навчання ШНМ, що відомі також як парадигми машинного навчання:

- навчання з учителем або кероване навчання (supervised learning);
- навчання без учителя або некероване навчання (unsupervised learning);
- навчання з підкріпленням (reinforcement learning).

При використанні парадигми навчання з учителем датасет, на якому навчається нейромережа, повинен бути розмічений або промаркований, тобто містити правильні відповіді (виходи нейромережі) на кожен вхідний зразок навчальної послідовності. Сама ця навчальна послідовність утворена парами значень  $(x(1), y^*(1)), (x(2), y^*(2)), \dots, (x(N), y^*(N))$  вхідних та очікуваних вихідних даних.

Найбільш відомим алгоритмом керованого навчання багат шарового пересептрону є алгоритм зворотного поширення помилки (backpropagation). Він належить до градієнтних алгоритмів оптимізації і основна його ідея полягає у тому, що сигнали помилки поширюються від виходів до входів ШНМ, тобто у напрямку, протилежному прямому поширенню сигналів у нейромережі. Проте алгоритму зворотного поширення помилки властиві недоліки, які можуть зробити процес навчання невизначено довгим:

- параліч мережі – в процесі навчання може виникнути ситуація, коли вихідні сигнали більшості нейронів набувають великих значень, що призводить до «залипання» або значного сповільнення процедури корекції вагових коефіцієнтів. Це пояснюється тим, що при великих значеннях сигналів ми потрапляємо в область, де похідна функції активації стає близькою до нуля, а крок пошуку процедури оптимізації пропорційний значенню цієї похідної;

- локальні мінімуми – оскільки поверхня цільової функції, що використовується при оптимізації, є складною і містить пагорби, долини, яри, складки тощо у просторі високої розмірності, то завжди є досить висока ймовірність потрапити в локальний мінімум, з якого алгоритм не може вийти, хоча поруч можуть знаходитись більш глибокі мінімуми;

- збіжність алгоритму – в теорії для забезпечення збіжності градієнтної процедури пошуку корекції вагових коефіцієнтів повинні бути нескінченно малими, але це неможливо реалізувати, оскільки ми матимемо безкінечно великий час навчання. Тому розмір кроку повинен бути скінченним, але якщо він буде фіксованим і занадто малим, то навчання

відбуватиметься надто повільно. Якщо крок пошуку буде фіксованим і занадто великим, то є небезпека виникнення паралічу або нестійкого режиму роботи мережі.

Можливо також виникнення ситуації перенавчання (*overfitting*), яка проявляється у тому що навчена нейромережа занадто точно відпрацьовує всі приклади з датасету, але не дає правильних результатів на тестовій вибірці, що містить нові для мережі дані, які не були присутні у навчальній вибірці. Тобто у процесі навчання модель починає відпрацьовувати випадкові похибки і шуми та втрачає здатність до узагальнення. Причиною перенавчання може бути недостатній обсяг навчальної вибірки, занадто тривале навчання або переускладнена структура нейромережевої моделі.

Кращі результати, порівняно з *backpropagation*, дають алгоритми другого порядку, такі як метод спряжених градієнтів та алгоритм Левенберга-Марквардта, які забезпечують вищу швидкість збіжності.

Неконтрольоване навчання або навчання без учителя використовує алгоритми самонавчання – вони навчаються без будь-яких позначок або попереднього навчання. Натомість нейромережевій моделі надаються необроблені дані без міток, і вона має виводити власні правила та структурувати інформацію на основі подібностей, відмінностей і шаблонів без чітких інструкцій щодо роботи з кожним фрагментом даних.

Алгоритми неконтрольованого навчання краще підходять для складніших завдань обробки, таких як організація великих наборів даних у кластери. Вони корисні для ідентифікації раніше невиявлених шаблонів у даних і можуть допомогти визначити функції, корисні для категоризації даних. До задач некерованого навчання відносяться також зниження розмірності та оцінювання закону розподілу випадкової величини.

Деякі алгоритми некерованого навчання здатні змінювати структуру самої мережі, тобто кількість нейронів та взаємозв'язки між ними, іншими словами здійснювати самоорганізацію.

До алгоритмів навчання без вчителя можна віднести метод *k*-середніх,

алгоритм просторової кластеризації DBSCAN (Density-Based Spatial Clustering Of Applications With Noise), метод головних компонентів (Principal Component Analysis, PCA), алгоритм Кохонена тощо.

Навчання з підкріпленням використовується для управління діями деякого інтелектуального агента в динамічному середовищі з метою максимізації сигналу винагороди. Базове навчання з підкріпленням розглядається як марковський процес прийняття рішень. Мета такого навчання полягає у тому, щоб агент навчився оптимальній (або майже оптимальній) стратегії, яка максимізує функцію винагороди або інший наданий сигнал підкріплення, який накопичується з миттєвих винагород.

### 2.3 Метод зворотного поширення помилки

Навчальна вибірка задається множиною пар векторів  $\{(x(i), y^*(i))\}$ ,  $i = 1, 2, \dots, N$ , де  $x = (x_1, x_2, \dots, x_m)$  – вектор вхідних сигналів нейромережі,  $y^* = (y_1^*, y_2^*, \dots, y_p^*)$  – вектор очікуваних значень вихідних сигналів для вхідного вектору  $x$ . Тоді функція помилки, яку потрібно мінімізувати в результаті навчання, матиме вигляд

$$E = \frac{1}{2} \sum_{k=1}^p (y_k^* - y_k)^2 = \frac{1}{2} \sum_{k=1}^p e_k^2. \quad (2.1)$$

Мінімізацію виразу (2.1) за ваговими коефіцієнтами  $w_{ij}$  будемо здійснювати за допомогою градієнтної процедури

$$w_{ij}(t) = w_{ij}(t-1) - \mu \frac{\partial E}{\partial w_{ij}}, \quad (2.2)$$

де  $\mu$  – невід’ємний множник, який визначає крок пошуку.

Розглянемо спочатку обчислення похідної для вагових коефіцієнтів

нейронів вихідного шару. Зазначимо, що ваговий коефіцієнт  $w_{ij}$  впливає на вихідний сигнал  $y_j$ , оскільки є частиною суми  $S_j = \sum_i w_{ij} O_i$ , де  $O_i$  позначає вихід  $i$ -го нейрона попереднього (передостаннього) шару. Тому можна записати

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial S_j} \frac{\partial S_j}{\partial w_{ij}} = \frac{\partial E}{\partial S_j} O_i. \quad (2.3)$$

Припускаючи, що вихідний шар нейронів має сигмоїдальну (логістичну) функцію активації  $y_j = \sigma(S_j) = \frac{1}{1 + e^{-S_j}}$ , отримаємо

$$\frac{\partial E}{\partial S_j} = \frac{\partial E}{\partial y_j} \cdot \frac{\partial y_j}{\partial S_j} = \frac{\partial}{\partial y_j} \left( \frac{1}{2} \sum_{j=1}^p (y_j^* - y_j)^2 \right) \cdot \frac{\partial \sigma(S_j)}{\partial S_j} = -y_j(1 - y_j)e_j, \quad (2.4)$$

де  $e_j = y_j^* - y_j$  – помилка  $j$ -го виходу мережі.

Якщо ж  $j$ -й нейрон знаходиться не у вихідному шарі, то

$$\begin{aligned} \frac{\partial E}{\partial S_j} &= \sum_k \frac{\partial E}{\partial S_k} \cdot \frac{\partial S_k}{\partial S_j}, \\ \frac{\partial S_k}{\partial S_j} &= \frac{\partial S_k}{\partial O_j} \cdot \frac{\partial O_j}{\partial S_j} = w_{ij} O_j (1 - O_j). \end{aligned} \quad (2.5)$$

Тут індекс  $k$  позначає виходи  $j$ -го нейрона. Позначимо  $\delta_k = \frac{\partial E}{\partial S_k}$  поправку, яка попередньо обчислена для вузла наступного рівня. Узагальнюючи вищесказане, для вузла вихідного шару

$$\delta_j = -y_j(1 - y_j)e_j,$$

а для внутрішнього вузла мережі

$$\delta_j = O_j(1 - O_j) \sum_{k \in \text{Outputs}(j)} \delta_k w_{jk}.$$

Запишемо послідовність дій алгоритму зворотного поширення помилки (backpropagation).

Крок 1. Ініціалізувати вагові коефіцієнти мережі  $w_{ij}$  випадковими значеннями, близькими до нуля.

Крок 2. Вибрати черговий приклад  $(x(k), y^*(k))$  з навчальної вибірки, подати вхідний вектор сигналів на вхід мережі.

Крок 3. Розрахувати вихідні сигнали кожного нейрона (включаючи виходи мережі). Обчислити різницю між виходом MLP та очікуваним значенням  $e = y^* - y$ .

Крок 4. Скоригувати синаптичні вагові коефіцієнти нейронів згідно таких співвідношень:

$$w_{ij}(t) = w_{ij}(t-1) - \mu \delta_j O_i,$$

$$\delta_j = \begin{cases} -y_j(1-y_j)e_j, & \text{для вузлів вихідного шару;} \\ O_j(1-O_j) \sum_{k \in \text{Outputs}(j)} \delta_k w_{jk}, & \text{для вузлів прихованого шару.} \end{cases}$$

Крок 5. Повторювати кроки 2-4 для кожного прикладу навчальної вибірки, поки помилка не досягне досить малого рівня, визначеного критерієм зупинки навчання, або по закінченню заданої кількості ітерацій (epoch).

Метод зворотного поширення помилки має ряд суттєвих недоліків, які були описані вище у розділі 2.2. Тому на практиці для навчання MLP більшого поширення набув алгоритм Левенберга-Марквардта.

## 2.4 Алгоритм Левенберга-Марквардта

Алгоритм Левенберга-Марквардта (Levenberg-Marquardt Algorithm, LMA) належить до градієнтних методів оптимізації другого порядку і використовується найчастіше для розв'язання нелінійних задач найменших квадратів [12, 13]. LMA по суті є комбінацією методів Гауса-Ньютона та градієнтного спуску. Важливою перевагою цього алгоритму перед градієнтними процедурами першого порядку є висока швидкість збіжності. Проте, як і більшість чисельних методів, алгоритм Левенберга-Марквардта знаходить лише локальний мінімум цільової функції, який не обов'язково буде глобальним.

Слід зауважити, що LMA було розроблено спеціально для мінімізації цільової функції у вигляді суми квадратів інших нелінійних функцій. Тому для задачі навчання MLP ми будемо розглядати середньоквадратичну функцію втрат

$$E(w) = \frac{1}{2} \sum_{i=1}^m e_i^2(w) = \frac{1}{2} e^T(w) e(w), \quad (2.6)$$

де  $m$  – кількість прикладів навчальної вибірки;

$e(w) = (e_1(w), e_2(w), \dots, e_m(w))^T$  – вектор помилок мережі;

$w = (w_1, w_2, \dots, w_n)^T$  – вектор параметрів мережі (вагових коефіцієнтів, що настроюються).

Градієнт цільової функції (2.6) можна записати у матричній формі як

$$\nabla_w E(w) = J^T(w) e(w), \quad (2.7)$$

$$\text{де } J(w) = \begin{bmatrix} \frac{\partial e_1(w)}{\partial w_1} & \frac{\partial e_1(w)}{\partial w_2} & \dots & \frac{\partial e_1(w)}{\partial w_n} \\ \frac{\partial e_2(w)}{\partial w_1} & \frac{\partial e_2(w)}{\partial w_2} & \dots & \frac{\partial e_2(w)}{\partial w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_m(w)}{\partial w_1} & \frac{\partial e_m(w)}{\partial w_2} & \dots & \frac{\partial e_m(w)}{\partial w_n} \end{bmatrix} \text{ – матриця Якобі, яка містить}$$

всі часткові похідні помилки мережі за всіма параметрами. Кожен рядок тут відповідає одному прикладу навчальної вибірки.

Особливістю методу найменших квадратів є те, що маючи матрицю Якобі, легко отримати гесіан  $\nabla_w^2 E(w)$ , якщо апроксимувати  $e_i(w)$  лінійними наближеннями [14]. У цьому випадку він буде мати вигляд

$$\nabla_w^2 E(w) = J^T(w)J(w). \quad (2.8)$$

Користуючись наведеними вище позначеннями, запишемо формулу методу Гауса-Ньютона для нашої задачі

$$w(t+1) = w(t) - \left[ J^T(w(t))J(w(t)) \right]^{-1} J^T(w(t))e(w(t)). \quad (2.9)$$

Тут може виникнути проблема з тим, що гесіан виявиться виродженою матрицею, яка не матиме оберненої. Для подолання цього до гесіана додають одиничну матрицю  $I$ , помножену на додатний параметр регуляризації  $\mu$ . Це і є алгоритм Левенберга-Марквардта

$$w(t+1) = w(t) - \left[ J^T(w(t))J(w(t)) + \mu I \right]^{-1} J^T(w(t))e(w(t)). \quad (2.10)$$

Коли параметр  $\mu$  має великі значення, алгоритм (2.10) наближається до методу градієнтного спуску, а коли  $\mu$  прямує до нуля, ми матимемо метод Гауса-Ньютона.

Починають навчання з малого значення множника  $\mu$ . Якщо значення нев'язки (2.6) не зменшується, то параметр  $\mu$  збільшують (зазвичай в 10 разів). Оскільки це наближає процес оптимізації до градієнтного спуску, то врешті-решт ми досягнемо зменшення середньоквадратичної помилки. Якщо таке зменшення відбулось, значення  $\mu$  зменшують (теж в 10 разів), задіюючи метод Гауса-Ньютона для швидшої збіжності до точки мінімуму. В цьому і полягає основна ідея LMA – постійно переключатися між методом найскорішого спуску і методом Гауса-Ньютона.

Треба сказати, що хоча LMA є не оптимальним, а лише евристичним алгоритмом, він дуже добре працює на практиці. Єдиний його недолік полягає у необхідності обернення матриці Гессе на кожній ітерації. Проте й тут можна застосувати певну хитрість, а саме вважати гесіан діагональною матрицею, тобто обнулити всі змішані часткові похідні. Обернення діагональної матриці, як відомо, здійснюється досить просто.

## 3 МЕРЕЖІ РАДІАЛЬНИХ БАЗИСНИХ ФУНКЦІЙ

### 3.1 Архітектура радіально-базисної мережі

Ще одним універсальним апроксиматором складних функціональних залежностей є мережа радіальних базисних функцій (Radial Basis Function Network, RBFN). За допомогою радіально-базисних функцій можна як завгодно точно апроксимувати довільну функцію, відповідно RBFN, як і MLP, є універсальним апроксиматором [15]. Саме цей аспект і буде нас цікавити у контексті задачі, яка розглядається в кваліфікаційній роботі.

Математичну основу RBFN складає метод потенційних функцій, що дозволяє подати деяку задану функцію  $y(x)$  у вигляді розкладання у базисі потенційних функцій

$$y(x) = \sum_{i=1}^P a_i f_i(x) = a^T f(x), \quad (3.1)$$

де  $a = (a_1, a_2, \dots, a_P)^T$  – вектор параметрів, які підлягають визначенню;

$f(x) = (f_1(x), f_2(x), \dots, f_P(x))^T$  – вектор базисних функцій.

В RBFN в якості базисних функцій розглядаються функції відстані між вектором вхідних сигналів  $x$  та векторами так званих центрів базисних функцій  $c_i$ :  $f_i(x) = f(\|x - c_i\|)$ . Зазвичай відстань між векторами оцінюється або евклідовою нормою

$$\|x - c_i\| = \sqrt{\sum_{j=1}^n (x_j - c_{ij})^2},$$

або манхетенською метрикою

$$\|x - c_i\| = \sum_{j=1}^n |x_j - c_{ij}|.$$

Архітектура такої мережі складається з трьох шарів:

- вхідного, на який поступає вектор вхідних сигналів;
- прихованого, що складається з нейронів радіального типу;
- вихідного, який формує зважену лінійну комбінацію виходів нейронів прихованого шару (рис. 3.1).

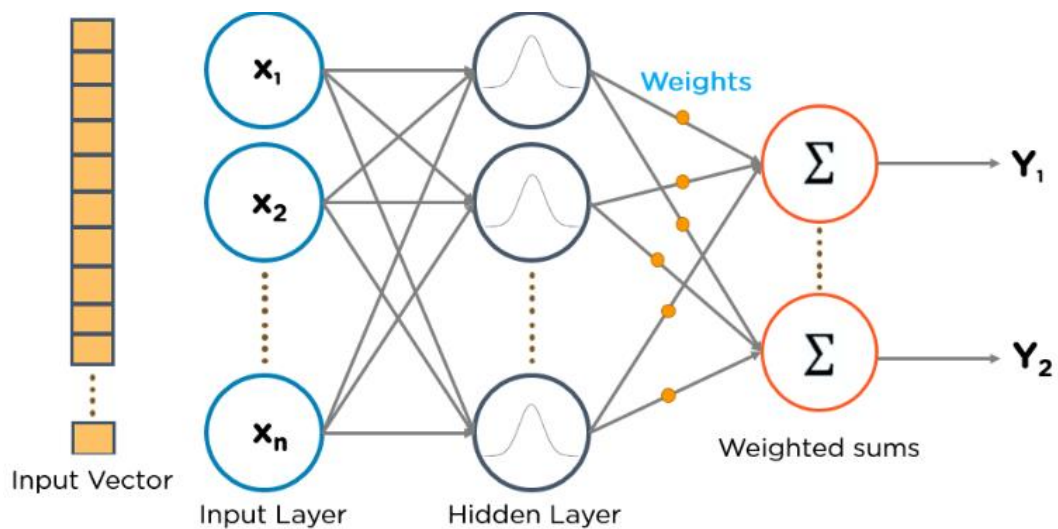


Рисунок 3.1 – Радіально-базисна нейромережа

У якості функції активації нейронів прихованого шару найчастіше використовується функція Гауса

$$\varphi_i(x) = \varphi_i(\|x - c_i\|) = \exp\left(-\frac{\|x - c_i\|^2}{2\sigma_i^2}\right), \quad (3.2)$$

де  $\varphi_i(x)$  – функція активації  $i$ -го нейрона прихованого шару;

$c_i$  – центр функції Гауса  $i$ -го нейрона прихованого шару;

$\sigma_i$  – ширина або радіус функції Гауса  $i$ -го нейрона прихованого шару.

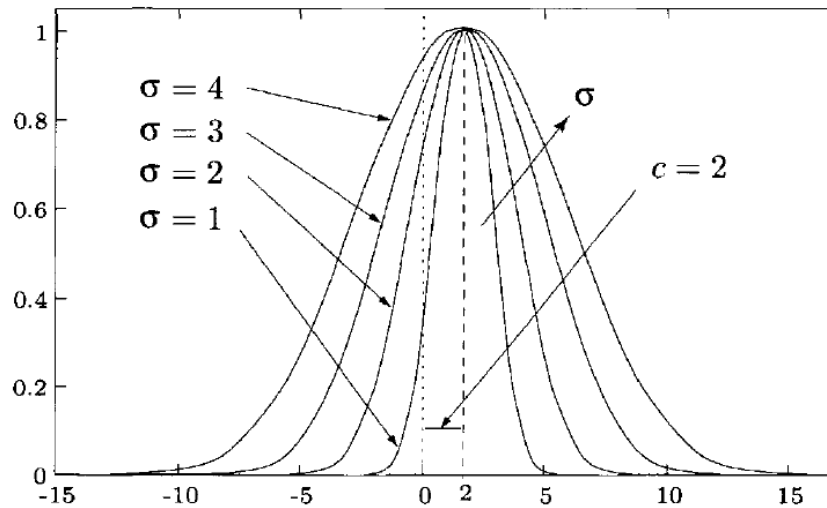


Рисунок 3.2 – Радіальна базисна функція Гауса

Таким чином, вихідний сигнал  $j$ -го нейрона вихідного шару описується співвідношенням

$$y_j(x) = \sum_{i=1}^P w_{ij} \exp\left(-\frac{\|x - c_i\|^2}{2\sigma_i^2}\right). \quad (3.3)$$

### 3.2 Навчання радіально-базисних нейромереж

Навчання RBFN полягає у визначенні кількості нейронів у прихованому шарі  $N$ , лінійних вагових коефіцієнтів  $w_{ij}$  вихідних нейронів, центрів  $c_i$  та радіусів  $\sigma_i$  нейронів прихованого шару. При цьому можливі такі варіанти:

- задаються фіксовані значення центрів та радіусів нейронів прихованого шару, а вагові коефіцієнти вихідних нейронів визначаються шляхом навчання;

- центри  $c_i$  і радіуси  $\sigma_i$  визначаються шляхом самонавчання (найчастіше за допомогою методів кластеризації), а потім корекцією ваг вихідного шару мінімізується цільова функція (2.1);

- всі параметри мережі визначаються за допомогою навчання з учителем, наприклад методом зворотного поширення помилки.

Навчання RBFN здійснюється поетапно. Спочатку необхідно визначити кількість нейронів у прихованому шарі та центри і радіуси функції Гауса для цих нейронів. Найпростішим способом є випадковий вибір центрів  $c_i$  серед прикладів навчальної вибірки. Проте більш поширеним варіантом є використання процедур кластеризації, наприклад  $k$ -середніх. Він полягає у розбивці множини вхідних даних на кластери з подальшим вибором їх центрів таким чином, щоб точки, які відносяться до одного кластера, знаходились ближче до центру цього кластера, ніж до центрів інших кластерів. Стандартний алгоритм  $k$ -кластеризації складається з наступних кроків.

Крок 1. Визначають кількість кластерів  $S < N$ .

Крок 2. Із навчальної вибірки  $\{x_1, x_2, \dots, x_N\}$  обирають  $S$  даних у якості центрів  $c_i = x_j, j = 1, 2, \dots, S$ .

Крок 3. Додають нові вектори  $x_i, i = S + 1, S + 2, \dots, N$  в один із кластерів за критерієм мінімуму відстані до центру кластера, тобто приписують  $x_i$  до  $j^*$ -го кластера, якщо виконується умова  $\|x_i - c_{j^*}\| = \min_j \|x_i - c_j\|, j = 1, 2, \dots, S$ .

Крок 4. Перевизначають центри кластерів  $c_j = \frac{1}{N_j} \sum_{i \in C_j} x_i, j = 1, 2, \dots, S$ ,

$N_j$  – кількість елементів навчальної вибірки, які належать  $j$ -му кластеру з центром  $c_j$ .

У результаті виконання вищенаведеного алгоритму отримуємо множину центрів  $\{c_1, c_2, \dots, c_S\}$ .

Після визначення центрів переходять до розрахунку радіусів функцій Гауса. Якщо вибрати радіуси занадто короткими, то мережа не зможе здійснювати інтерполяцію між заданими точками і втратить здатність до узагальнення. Якщо ж радіуси, навпаки, виявляться занадто великими, мережа не

зможє відслідковувати дрібні деталі. Розрахунок радіусів функцій Гауса можна здійснювати з використанням методу  $k$ -найближчих сусідів ( $k$ -nearest neighbor method). У цьому випадку параметр  $\sigma_i$  обирається як середньоквадратична відстань між центром  $c_i$  та його  $K$  найближчими сусідами

$$\sigma_i = \sqrt{\frac{1}{K} \sum_{k=1}^K \|c_i - c_k\|^2}.$$

Після того, як визначено значення центрів та радіусів функцій Гауса нейронів прихованого шару, переходять до настроювання вагових коефіцієнтів  $w_{ij}$  нейронів вихідного шару. Оскільки ці коефіцієнти входять у рівняння перетворення (3.3) лінійно, для розрахунку їх значень можна скористатися методом найменших квадратів, який буде мінімізувати цільову функцію (2.1).

## 4 ПОБУДОВА ФУНКЦІЙ ПЕРЕТВОРЕННЯ ЗАСОБІВ ВИМІРЮВАННЯ

### 4.1 Функція перетворення засобу вимірювання та підходи до її визначення

Однією з основних метрологічних характеристик засобу вимірювань, згідно з ГОСТ 8.009-84, є його номінальна статична функція перетворення  $y = F(x)$ , яка є функціональною залежністю інформативного параметра вихідного сигналу вимірювального пристрою  $y$  від інформативного параметра його вхідного сигналу  $x$ . Найчастіше до функції перетворення пред'являється вимога лінійності робочому діапазоні вимірюваної величини.

Для експериментального визначення функції перетворення здійснюються сумісні вимірювання. Сумісними називаються одночасно проведені вимірювання двох або декількох неоднойменних величин з метою визначення функціональної залежності між ними:

$$Y = f(X_1, \dots, X_k). \quad (4.1)$$

При цьому спільно вимірюють значення аргументів  $(X_{1i}, \dots, X_{ki})$  і відповідні значення величини  $Y$ :  $Y_i = f(X_{1i}, \dots, X_{ki})$ ; по отриманим експериментальним даним  $(x_{1i}, \dots, x_{ki}, y_i)$  будують функціональну залежність (4.1). Найбільш важливим для практичного застосування є залежність величини  $Y$  тільки від одного аргументу  $X$ :

$$Y = f(X). \quad (4.2)$$

Можливі два основних варіанти реалізації сумісних вимірювань величин  $X$  і  $Y$ : активний і пасивний експеримент. В активному (або

планованому) експерименті значення аргументу  $(x_1, \dots, x_m)$  вибирають заздалегідь і послідовно відтворюють ці значення, виконуючи при кожному фіксованому значенні аргументу  $x_i$  вимірювання відповідної величини  $Y_i$ . У цьому випадку  $X$  називають контрольованою змінною. Оскільки значення аргументу  $x_i$  відтворюються з похибками  $\delta_{x_i}$ , істинні значення  $X_i$  при цьому невідомі:  $x_i = X_i + \delta_{x_i}$ . Істинні значення величини  $Y$  відповідають  $Y_i = f(X_i)$ , а в результаті вимірювань одержують:

$$y_i = f(X_i) + \delta_{y_i}. \quad (4.3)$$

У пасивному (непланованому) експерименті значення аргументу заздалегідь не вибирають, а вимірюють ті значення  $X_i$ , які задані яким-небудь способом або довільно обрані із числа можливих; одержують результати вимірювань:  $x_i = X_i + \delta_{x_i}$ , де  $\delta_{x_i}$  – похибки вимірювання. При тих же значеннях  $X_i$  вимірюють також відповідні величини  $Y_i$  і отримують результати (4.3).

У багатьох випадках активний експеримент є кращим, оскільки за рахунок раціонального вибору значень  $x$  можна одержати більш точні оцінки залежності, а для обробки можна застосовувати більш широкий клас оцінок. Однак на практиці це не завжди можна реалізувати.

Шукана залежність (4.2), що будується по набору експериментальних даних  $(x_i, y_i)$ ,  $i = 1, \dots, m$ , може бути представлена в різному вигляді: аналітично (формулою), графічно або у вигляді таблиці. Вибір способу представлення залежить від функціонального вигляду й складності залежності, а також від способу її подальшого використання на практиці. Звичайно прагнуть задавати її в аналітичному вигляді формулою:

$$Y = f(X; a_1, \dots, a_k). \quad (4.4)$$

Така форма подання є найбільш компактною і дозволяє вирішувати широке коло практичних задач. Однак у тих випадках, коли залежність не можна досить точно апроксимувати простою функцією або аналітична побудова виявляється занадто трудомісткою, доводиться задавати її за допомогою графіка або таблиці.

Спосіб завдання залежності визначає обробку даних при її побудові. Якщо залежність задається таблицею  $(x_i, y_i)$ ,  $i = 1, \dots, m$ , то при її складанні лише виконують обробку результатів спостережень  $x_{ij}, y_{ij}$  у кожній точці окремо відповідно до звичайних правил обробки даних при прямих або непрямих вимірюваннях. Якщо залежність задається аналітично, то на додаток до цього необхідна обробка всього набору даних у цілому. При цьому задають певний функціональний вигляд залежності і обчислюють оцінки її параметрів.

Для графічного способу завдання залежності можливі два варіанти:

- якщо графік будується по точках (без згладжування), то виконується лише обробка даних у кожній із точок; у проміжках між ними залежність визначається, наприклад, шляхом інтерполяції;
- якщо графік будують зі згладжуванням, орієнтуючись на певний функціональний вигляд залежності, то обробка виконується, як і при аналітичному представленні.

При побудові залежності в аналітичному вигляді варто дотримуватися наступного порядку дій:

- визначити функціональний вигляд залежності. Вигляд залежності може бути приблизно відомий або з фізичних закономірностей, що описують дане явище, або на основі попереднього досвіду й попереднього аналізу даних. Іноді він буває заданий заздалегідь як бажаний або необхідний;

- вибрати метод побудови залежності за експериментальним даними. При цьому варто враховувати обраний вигляд залежності й апріорні відомості про похибки вимірювання  $\delta_{x_i}$  і  $\delta_{y_i}$ ;

- обчислити оцінки параметрів залежності обраного вигляду;
- оцінити похибки побудованої залежності. При цьому використовуються відомі характеристики випадкових і систематичних похибок вимірювання  $x_i$  і  $y_i$ ;

- перевірити правильність вибору вигляду залежності або оцінити ступінь відхилення дійсної залежності від обраного вигляду. У деяких випадках необхідно також перевірити спільність побудованої залежності з деякою заданою або раніше побудованою. Наприклад, при перевірці засобів вимірювань порівнюють побудовану градувальну характеристику з номінальною або раніше побудованою. Якщо розбіжність двох залежностей допустима, то засіб вимірювання вважають придатним, у протилежному випадку – його бракують.

#### 4.2 Побудова лінійної функції перетворення методом найменших квадратів

У сучасній математиці розроблені численні методи вирішення завдань побудови функціональних залежностей за експериментальним даними, насамперед – статистичні методи, засновані на імовірнісних моделях для похибок вимірювань [16]. Найпоширенішим з них є метод найменших квадратів (МНК). У цьому методі оцінки параметрів залежності визначають із умови, що сума квадратів відхилень розрахункових значень від експериментальних значень  $y_i$  мінімальна. Стосовно до залежності вигляду

$Y = \sum_{i=1}^k a_i \varphi_i(X)$  це означає:

$$\min Q = \sum_{i=1}^m \left[ y_i - \sum_{j=1}^k a_j \varphi_j(X_i) \right]^2. \quad (4.5)$$

Умова мінімізації  $Q$  приводить до системи рівнянь:

$$\frac{\partial Q}{\partial a_l} = -2 \sum_{i=1}^m \left[ y_i - \sum_{j=1}^k a_j \varphi_j(X_i) \right] \varphi_l(X_i) = 0, \quad (4.6)$$

з якої утворюється лінійна система щодо коефіцієнтів  $a_j$ :

$$\sum_{j=1}^k a_j b_{jl} = g_l, \quad l = 1, \dots, k, \quad (4.7)$$

де

$$b_{jl} = \sum_{i=1}^m \varphi_j(X_i) \varphi_l(X_i), \quad (4.8)$$

$$g_l = \sum_{i=1}^m y_i \varphi_l(X_i). \quad (4.9)$$

У результаті розв'язання цієї системи одержують оцінки МНК [16].

При обґрунтуванні методу найменших квадратів у математичній статистиці передбачається, що результати вимірювань, задовольняють наступним умовам:

- значення аргументів  $X_i$  відомі точно;
- результати вимірювань  $y_i$  містять лише випадкові похибки, які незалежні, мають нульові середні і однакові дисперсії  $\sigma^2$ ;
- похибки  $y_i$  мають нормальний розподіл.

За цих умов МНК дає незміщені оцінки коефіцієнтів залежності, що мають мінімальні дисперсії. При виконанні тільки першої й другої умов МНК-оцінки незміщені й мають мінімальні дисперсії серед лінійних оцінок коефіцієнтів.

Однак на практиці перераховані умови виконуються далеко не завжди; замість них часто зустрічаються наступні:

- похибки вимірювання аргументів  $X_i$  істотні. Наявність похибок вимірювання аргументів призводить до появи істотних зсувів оцінок параметрів залежності, тобто оцінки не є незміщеними і спроможними. Однак для планованого експерименту при побудові лінійної залежності все-таки можна застосовувати МНК із відповідними уточненнями при оцінці похибок. Для непланованого експерименту необхідно застосовувати методи конфлюентного аналізу, у яких ураховується наявність похибок  $X_i$ ;

- похибки вимірювання величин  $y_i$  містять систематичні складові. Наявність систематичних похибок вимірювань  $y_i$  приводить до того, що МНК не є строго оптимальним, однак це не перешкоджає його застосуванню. Оцінка систематичних похибок отриманих результатів досить проста й звичайно не викликає труднощів;

- похибки вимірювання  $y_i$  можуть бути залежні або мати різні дисперсії. Наявність залежностей або неоднорідність дисперсій також не є перешкодою для використання МНК. Якщо кореляційна матриця похибок відома (хоча б приблизно), то можна використати узагальнений МНК із вагами, що дає оптимальні оцінки. Однак часто вона невідома, і тоді використовують звичайний МНК, що дає незміщені, але не найефективніші оцінки;

- розподіли випадкових похибок вимірювання  $y_i$  відмінні від нормальних. Якщо розподіли випадкових похибок  $y_i$  різко відрізняються від нормальних, то формальне застосування МНК може привести до значних похибок залежностей; до цього ж приводить і наявність грубих похибок (промахів) в експериментальних даних. Тому треба при використанні МНК перевіряти гіпотезу про нормальний розподіл похибок або використати один з робастних методів, які малочутливі до зазначених відхилень.

### 4.3 Побудова нелінійних залежностей методом найменших квадратів

Хоча на практиці вимірювань найпоширенішими є лінійні залежності, але нерідко доводиться використовувати й нелінійні залежності, які звичайно належать до однієї із двох груп:

- функція  $Y = f(X)$  перетворюється в лінійну функцію  $\tilde{Y} = a + b\tilde{X}$  шляхом заміни змінних:

$$\tilde{X} = \varphi(X); \tilde{Y} = \psi(Y); \quad (4.10)$$

- функція  $Y = f(X)$  є лінійною комбінацією відомих функцій  $\varphi_1, \dots, \varphi_k$ :

$$Y = \sum_{j=1}^k a_j \varphi_j(X). \quad (4.11)$$

З нелінійних функцій першої групи найчастіше зустрічаються степеневі, показникові і дробово-лінійні; із другої групи слід зазначити алгебраїчні і тригонометричні поліноми. Крім того, залежність можна задавати декількома аналітичними виразами, що відповідають різним піддіапазонам аргументу.

Для побудови залежностей, що приводять до лінійних, застосовні всі методи, розроблені для побудови лінійних залежностей, і в першу чергу – МНК. При побудові лінійної залежності  $\tilde{Y} = a + b\tilde{X}$  використовуються перетворені результати спостережень  $(\tilde{X}_i, \tilde{y}_i)$ , де  $\tilde{X}_i = \varphi(X_i)$ ,  $\tilde{y}_i = \psi(y_i)$ ; зокрема, у МНК мінімізується сума квадратів відхилень для нових змінних:

$$Q = \sum_{i=1}^m [\tilde{y}_i - (a + b\tilde{X}_i)]^2. \quad (4.12)$$

Тому при переході до нових змінних варто змінити вагові коефіцієнти результатів спостережень. Якщо вихідні ваги результатів спостережень  $y_i$  були  $\omega_i$ , то для  $\tilde{y}_i$  варто прийняти ваги

$$\tilde{\omega}_i = \omega_i |\psi'(y_i)|^{-2}. \quad (4.13)$$

Це приведе до того, що буде мінімізуватися сума  $Q$  для вихідних змінних.

Після побудови лінійної залежності  $\tilde{Y} = a + b\tilde{X}$  для вихідних змінних виходить залежність вигляду

$$\psi(Y) = a + b\varphi(X) \quad (4.14)$$

або

$$Y = \psi^{-1}(a + b\varphi(X)), \quad (4.15)$$

де  $\psi^{-1}$  – функція, обернена до  $\psi$ .

Найпоширеніші на практиці види нелінійних функцій, що приводять до лінійних, і відповідні лінеаризуючі перетворення наведені в табл. 4.1.

Оцінка СКВ похибок і побудова довірчих границь виконується для перетворених змінних  $\tilde{Y}$ ,  $\tilde{X}$  відповідно до наведеного вище співвідношення. При цьому необхідно враховувати зміни ваг. Потім шляхом зворотної заміни змінних можна одержати довірчі інтервали й оцінки похибок для вихідних змінних.

Таблиця 4.1 – Види нелінійних функцій і відповідні лінеаризуючі перетворення

Функція	Формула	Заміна змінних при лінеаризації	Отримана лінійна функція	Обернена функція
Показникова I	$Y = ae^{\beta X}$	$\tilde{Y} = \ln Y$	$\tilde{Y} = \ln \alpha + \beta X$	$Y = e^{\tilde{Y}}$
Дробово-лінійна I	$Y = (a + \beta X)^{-1}$	$\tilde{Y} = 1/Y$	$\tilde{Y} = \alpha + \beta X$	$Y = 1/\tilde{Y}$
Логарифмічна I	$Y = \alpha + \beta \ln X$	$\tilde{X} = \ln X$	$Y = \alpha + \beta \tilde{X}$	$Y = \tilde{Y}$
Степенева	$Y = \alpha X^\beta$	$\tilde{Y} = \ln Y;$ $\tilde{X} = \ln X$	$\tilde{Y} = \ln \alpha + \beta \tilde{X}$	$Y = e^{\tilde{Y}}$
Гіперболічна	$Y = \alpha + \beta/X$	$\tilde{X} = 1/X$	$Y = \alpha + \beta \tilde{X}$	$Y = \tilde{Y}$
Дробово-лінійна II	$Y = X/(\alpha + \beta X)$	$\tilde{Y} = 1/Y;$ $\tilde{X} = 1/X$	$\tilde{Y} = \beta + \alpha \tilde{X}$	$Y = 1/\tilde{Y}$
Показникова II	$Y = ae^{\beta/X}$	$\tilde{Y} = \ln Y;$ $\tilde{X} = 1/X$	$\tilde{Y} = \ln \alpha + \beta \tilde{X}$	$Y = e^{\tilde{Y}}$
Логарифмічна II	$Y = \alpha + \beta/\ln X$	$\tilde{X} = 1/\ln X$	$Y = \alpha + \beta \tilde{X}$	$Y = \tilde{Y}$

Досить поширені на практиці нелінійні залежності вигляду (4.11). Для їхньої побудови також використовується МНК; при цьому ваги результатів  $\omega_i$  визначають відповідно до загальних правил, наведених вище. Оцінки МНК знаходять із умови

$$\min Q = \sum_{i=1}^m \omega_i \left[ y_i - \sum_{j=1}^k a_j \varphi_j(X_i) \right]^2, \quad (4.16)$$

яка приводить до лінійної системи рівнянь відносно  $a_j$ :

$$\sum_{j=1}^k a_j b_{lj} = g_l, l = 1, \dots, k, \quad (4.17)$$

$$\text{де } b_{lj} = \sum_{i=1}^m \omega_i \varphi_l(X_i) \varphi_j(X_i);$$

$$g_l = \sum_{i=1}^m \omega_i y_i \varphi_l(X_i).$$

Найпоширенішими із цієї групи є алгебраїчні поліноми  $Y = \sum_{j=0}^k a_j X^j$ .

#### 4.4 Апроксимація функцій за допомогою штучних нейронних мереж

Завдання апроксимації функції також можна вирішити й за допомогою нейронної мережі. Розглянемо мережу, нейрони якої як функція активації мають не сходинок, а деяку безперервну сигмоїдальну функцію. У цьому випадку вихід мережі буде деякою безперервною функцією її входів. Конкретний вид цієї функції визначається ваговими коефіцієнтами кожного з нейронів.

Узагальнену теорему Стоуна, не вдаючись у її математичні тонкощі, можна інтерпретувати як твердження про універсальні апроксимаційні можливості довільної нелінійності: за допомогою лінійних операцій і каскадного з'єднання можна з довільного нелінійного елемента одержати пристрій, що обчислює будь-яку безперервну функцію з будь-якою наперед заданою точністю.

Тобто нейромережа із довільною функцією активації може апроксимувати довільну безперервну функцію. При цьому, завжди достатньо тришарової мережі. Потрібна більша точність – просто додають нейронів.

Одним із самих популярних алгоритмів навчання ШНМ є алгоритм зворотного поширення. Це один з варіантів навчання із учителем. Нехай є багатошарова мережа прямого поширення з випадковими ваговими

коефіцієнтами. Є деяка навчальна множина, що складається з пар вхід мережі – бажаний вихід  $\{X, D\}$ . Через  $Y$  позначають реальне вихідне значення мережі, що на початку є практично випадковим через випадковість вагових коефіцієнтів.

Навчання полягає в тому, щоб підібрати вагові коефіцієнти таким чином, щоб мінімізувати деяку цільову функцію. Як цільову функцію розглянемо суму квадратів помилок мережі на прикладах з навчальної множини

$$E(w) = \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2, \quad (4.18)$$

де  $y_{j,p}^{(N)}$  – реальний вихід  $N$ -го вихідного шару мережі для  $p$ -го нейрона на  $j$ -му навчальному прикладі;

$d_{j,p}$  – бажаний вихід.

Тобто, мінімізувавши такий функціонал, одержуємо рішення по методу найменших квадратів.

Оскільки вагові коефіцієнти в залежність  $y_{j,p}^{(N)}(x)$  входять нелінійно, для знаходження мінімуму використовують метод найшвидшого спуску. Тобто на кожному кроці навчання змінюють вагові коефіцієнти по формулі

$$\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial E}{\partial w_{ij}^{(n)}}, \quad (4.19)$$

де  $w_{ij}^{(n)}$  – ваговий коефіцієнт  $j$ -го нейрона  $n$ -го шару для зв'язку з  $i$ -м нейроном  $(n-1)$ -го шару;

$\eta$  – параметр швидкості навчання.

Таким чином, потрібно визначити частинні похідні цільової функції (4.18) по всіх вагових коефіцієнтах мережі. Відповідно до правил

диференціювання складної функції

$$\frac{\partial E}{\partial w_{ij}^{(n)}} = \frac{\partial E}{\partial y_j^{(n)}} \cdot \frac{dy_j^{(n)}}{ds_j^{(n)}} \cdot \frac{\delta_j^{(n)}}{\partial w_{ij}^{(n)}}, \quad (4.20)$$

де  $y_j^{(n)}$  – вихід;

$s_j^{(n)}$  – зважена сума входів  $j$ -го нейрона  $n$ -го шару.

Знаючи функцію активації, можна обчислити  $\frac{dy_j^{(n)}}{ds_j^{(n)}}$ . Наприклад, для

сигмоїди відповідно ця величина буде дорівнювати

$$\frac{dy_j^{(n)}}{ds_j^{(n)}} = \alpha y_j^{(n)} (1 - y_j^{(n)}). \quad (4.21)$$

Третій співмножник  $\delta_j^{(n)} / \partial w_{ij}^{(n)}$  є не що інше, як вихід  $i$ -го нейрона  $(n-1)$ -го шару, тобто

$$\frac{\delta_j^{(n)}}{\partial w_{ij}^{(n)}} = y_i^{(n-1)}. \quad (4.22)$$

Частинні похідні цільової функції по вагам нейронів вихідного шару тепер можна легко обчислити

$$\frac{\partial E}{\partial w_{ij}^{(N)}} = (y_j^{(N)} - d_j) \cdot \frac{dy_j^{(N)}}{ds_j^{(N)}} \cdot y_i^{(N-1)}. \quad (4.23)$$

Вводиться позначення

$$\delta_j^{(n)} = \frac{\partial E}{\partial y_j^{(n)}} \cdot \frac{dy_j^{(n)}}{ds_j^{(n)}}. \quad (4.24)$$

Тоді для нейронів вихідного шару

$$\delta_j^{(N)} = (y_j^{(N)} - d_j) \cdot \frac{dy_j^{(n)}}{ds_j^{(n)}}. \quad (4.25)$$

Для вагових коефіцієнтів нейронів внутрішніх шарів можна записати, чому дорівнює перший співмножник з (4.24), однак його можна представити в такий спосіб:

$$\frac{\partial E}{\partial y_j^{(n)}} = \sum_k \frac{\partial E}{\partial y_k^{(n+1)}} \cdot \frac{dy_k^{(n+1)}}{ds_k^{(n+1)}} \cdot \frac{\partial s_k^{(n+1)}}{\partial y_j^{(n)}} = \sum_k \frac{\partial E}{\partial y_k^{(n+1)}} \cdot \frac{dy_k^{(n+1)}}{ds_k^{(n+1)}} \cdot w_{jk}^{(n+1)}. \quad (4.26)$$

У цій формулі перші два співмножники  $\delta_k^{(n+1)}$ . Можна виразити величини  $\delta_j^{(n)}$  для нейронів  $n$ -го шару через  $\delta_k^{(n+1)}$  для нейронів  $(n+1)$ -го. Оскільки для останнього шару  $\delta_j^{(N)}$  легко за допомогою рекурсивної формули

$$\delta_j^{(n)} = \left[ \sum_k \delta_k^{(n+1)} \cdot w_{jk}^{(n+1)} \right] \cdot \frac{dy_j}{ds_j} \quad (4.27)$$

одержати значення  $\delta_j^{(n)}$  для всіх нейронів всіх шарів.

Остаточно формулу для модифікації вагових коефіцієнтів можна записати у вигляді

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}. \quad (4.28)$$

Таким чином, повний алгоритм навчання нейронної мережі за допомогою алгоритму зворотного поширення матиме наступний вигляд.

Крок 1. Присвоїти всім ваговим коефіцієнтам мережі випадкові початкові значення. При цьому мережа буде здійснювати якесь випадкове перетворення вхідних сигналів і значення цільової функції (4.18) будуть великі.

Крок 2. Подати на вхід мережі один із вхідних векторів з навчальної множини. Обчислити вихідні значення мережі, запам'ятовуючи при цьому вихідні значення кожного з нейронів.

Крок 3. Розрахувати  $\delta_j^{(N)}$ . Потім за допомогою рекурсивної формули підраховуються всі інші  $\delta_j^{(n)}$ .

Крок 4. Скорегувати ваги мережі  $w_{ij}^{(n)} = w_{ij}^{(n)} + \Delta w_{ij}^{(n)}$ .

Крок 5. Розрахувати цільову функцію. Якщо вона досить мала, вважаємо навчання мережі успішно завершеним. Інакше повертаємося на крок 2.

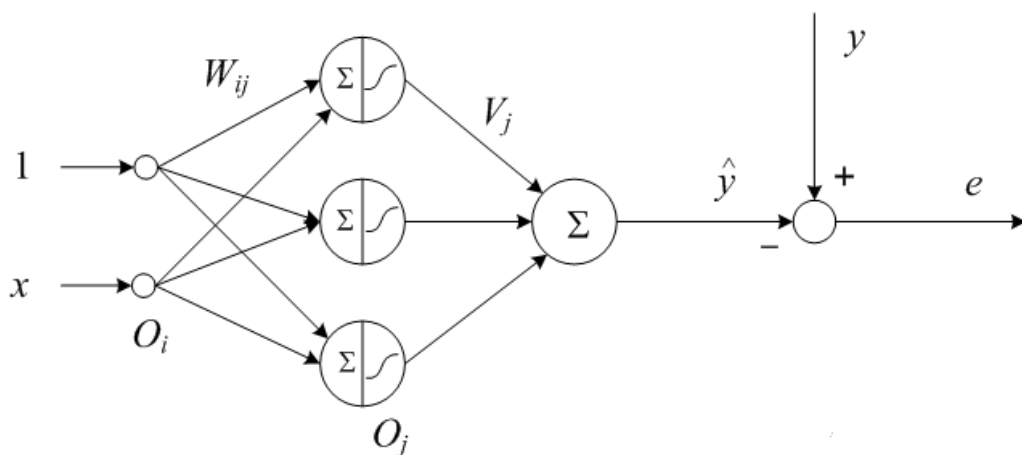


Рисунок 4.1 – Персептрон для моделювання нелінійної функції

Процедура настроювання синаптичних вагів персептрона, зображеного на рис. 4.1, буде мати вигляд:

$$W_{ij}(n) = W_{ij}(n-1) - \eta_W \frac{\partial E}{\partial W_{ij}} = W_{ij}(n-1) + \eta_W e(n) V_j(n) O_j (1 - O_j) O_i,$$

$$V_j(n) = V_j(n-1) - \eta_V \frac{\partial E}{\partial V_j} = V_j(n-1) + \eta_V e(n) O_j.$$

Існують деякі проблеми, що виникають при застосуванні цього методу:

- локальні мінімуми. Алгоритм зворотного поширення реалізує градієнтний спуск по поверхні помилки в просторі вагових коефіцієнтів і тому може застрягати в локальних мінімумах. При цьому поруч може бути інший, значно більш глибокий мінімум.

Для подолання цих труднощів навчання мережі проводять кілька разів і потім вибирають той варіант навченої мережі, що дає найкращі результати. Інший метод полягає в тому, що мережу можна вивести з локального мінімуму, на короткий час, збільшивши швидкість навчання. Іноді, до зміни вагових коефіцієнтів, обчислений по алгоритму, додають шум. Це також дозволяє мережі «вистрибувати» з локальних мінімумів;

- параліч мережі. У процесі навчання мережі значення ваг можуть у результаті корекції стати дуже великими величинами. Це може привести до того, що всі або більшість нейронів будуть функціонувати при дуже великих аргументах функції активації  $s_j^{(n)}$ , в області, де похідна функції дуже мала (як, наприклад для сигмоїдальної функції активації). Тому що величина корекції ваг пропорційна цій похідній, то процес навчання може практично завмерти. Для подолання такої ситуації існують модифікації алгоритму, що враховують лише напрямок антиградієнта й не враховуючу його величину.

## 5 КОРИГУВАННЯ НЕЛІНІЙНОСТІ ФУНКЦІЇ ПЕРЕТВОРЕННЯ ЗАСОБІВ ВИМІРЮВАННЯ ЗА ДОПОМОГОЮ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ

### 5.1 Постановка задачі дослідження

Кілька останніх десятиліть ознаменувались стрімким розвитком інформаційних технологій, які радикально і докорінно змінили не тільки звичний уклад людського життя, а й цілі сектори економіки у більшості розвинених країн світу. Безумовним досягненням наукової думки стали успіхи в розробці штучного інтелекту. Дослідження в царині штучних нейронних мереж, глибокого навчання, фаззі-систем дозволили реалізувати досить складні методи математичної обробки та аналізу великих масивів даних та розв'язувати цілий ряд задач класифікації, оптимізації, управління, розпізнавання образів, ідентифікації, апроксимації складних нелінійних залежностей, діагностики тощо.

Галузь вимірювань та вимірювальної техніки теж не залишається осторонь і намагається активно використовувати більш ефективні методи обробки інформації, які дозволять підвищити точність і достовірність результатів вимірювань. Роль таких методів стає більш важливою з ускладненням вимірювань і підвищенням вимог до їх точності.

Нові напрямки прикладної математики, такі як теорія штучних нейронних мереж, інтервальний аналіз, робасна та непараметрична статистика, нечітка логіка, вейвлет-аналіз та ряд інших, дають математичний апарат для розв'язання тих задач, для яких класичні методи обробки даних є непридатними або малоефективними. До розробки нових методів обробки вимірювальної інформації спонукають потреби сучасної вимірювальної практики, для якої є характерним ускладнення вимірювальних задач і засобів вимірювання.

Розширення можливостей інтелектуальної обробки даних та

результатів вимірювання тісно пов'язано з розвитком методології вимірювань, удосконаленням математичних методів, а також широким впровадженням засобів обчислювальної техніки у вимірювальний ланцюг, включаючи мікропроцесори і мікроконтролери.

Для засобу вимірювання одними з найбільш важливих є метрологічні характеристики, що впливають на результат вимірювання та точність цього результату. Однією з таких характеристик є номінальна статична функція перетворення засобу вимірювання (інші її назви – рівняння перетворення, градувальна характеристика). Вона встановлює залежність  $y = F(x)$  інформативного параметра вихідного сигналу  $y$  від значення інформативного параметра вхідного сигналу  $x$ .

Зазвичай до функції перетворення висувається вимога лінійності у робочому діапазоні вимірювань. Однак, досить часто при вирішенні практичних вимірювальних задач доводиться мати справу з вимірювальними пристроями, які мають суттєво нелінійну функцію перетворення. Так, наприклад, аналогові електронні мегаомметри в режимі вимірювання великих опорів (одиниці і десятки мегаомів) мають зворотну нерівномірну шкалу. Чутливість приладу на різних ділянках такої шкали буде різною, що не зовсім зручно і при взятті відліку, і при оцінюванні похибки результату вимірювання. Окремі елементи вимірювального тракту, такі як напівпровідникові терморезистори або діоди, теж мають суттєво нелінійну робочу характеристику, що відбивається на функції перетворення приладу. Іноді наблизити функцію перетворення до лінійної вдається за допомогою схемотехнічних рішень, але не завжди.

Зазвичай на практиці функцію перетворення апроксимують лінійною залежністю  $\hat{y} = a + bx$ , невідомі коефіцієнти якої  $a$  і  $b$  знаходять за допомогою методу найменших квадратів. Проте у випадку, коли нелінійність функції перетворення є значною, такий підхід не дає бажаного результату, оскільки внаслідок відхилення номінальної функції перетворення від реальної виникає велика методична похибка (рис. 5.1).

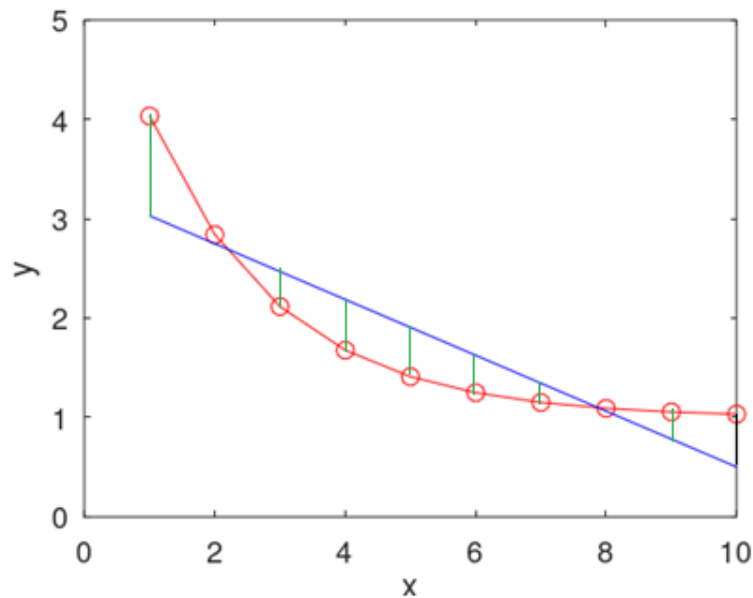


Рисунок 5.1 – Похибки лінеаризації функції перетворення

Для зменшення цієї похибки можна спробувати привести нелінійну функцію  $y = F(x)$  до лінійного вигляду  $\tilde{y} = a + b\tilde{x}$  шляхом заміни змінних

$$\tilde{y} = \varphi(y), \quad \tilde{x} = \psi(x)$$

з подальшим визначенням коефіцієнтів  $a$  і  $b$  лінійної функції за методом найменших квадратів [16]. Але у цьому випадку потрібно заздалегідь знати, як виглядає нелінійна функція перетворення, тобто мати апріорну інформацію про структуру математичної моделі засобу вимірювання. Обґрунтований вибір виду цієї нелінійної залежності є доволі складною задачею, яка погано піддається формалізації і здійснюється, виходячи з відомих фізичних законів або особистого досвіду фахівця, який займається розв'язанням задачі.

Одним з відомих підходів до зменшення впливу нелінійності функції перетворення на похибку результату вимірювання, який можна вважати універсальним, є алгоритмічна корекція функції перетворення. Послідовно із засобом вимірювання включається спеціальний пристрій-коректор, який

здійснює зворотне по відношенню до його характеристики перетворення  $\hat{x} = F^{-1}(y)$ . В результаті такої корекції ми отримуємо оцінку  $\hat{x}$  вхідного сигналу (вимірюваної величини)  $x$ , а результуюча функція перетворення стає лінійною. Важливою додатковою умовою є інваріантність такого перетворювача до виду нелінійної функції, що коригується, тобто можливість адаптації до довільної функції перетворення.

Метою даної кваліфікаційної роботи є дослідження адаптивної системи коригування функції перетворення засобу вимірювання, яка буде використовувати у якості пристрою корекції штучну нейронну мережу та забезпечуватиме лінійність функції перетворення в усьому діапазоні можливих значень вимірюваної вхідної величини. Узагальнену структуру такої системи наведено на рисунку 5.2. Дію внутрішніх та зовнішніх випадкових факторів на вимірювальний процес представлено адитивним шумом  $\xi(t)$  на виході засобу вимірювання, де  $t$  – поточний час. Вихідний сигнал  $y(t)$  подається на вхід нейромережі-коректора, яка здійснює зворотне перетворення та формує оцінку вхідного сигналу (вимірюваної величини)  $\hat{x}(t)$ .

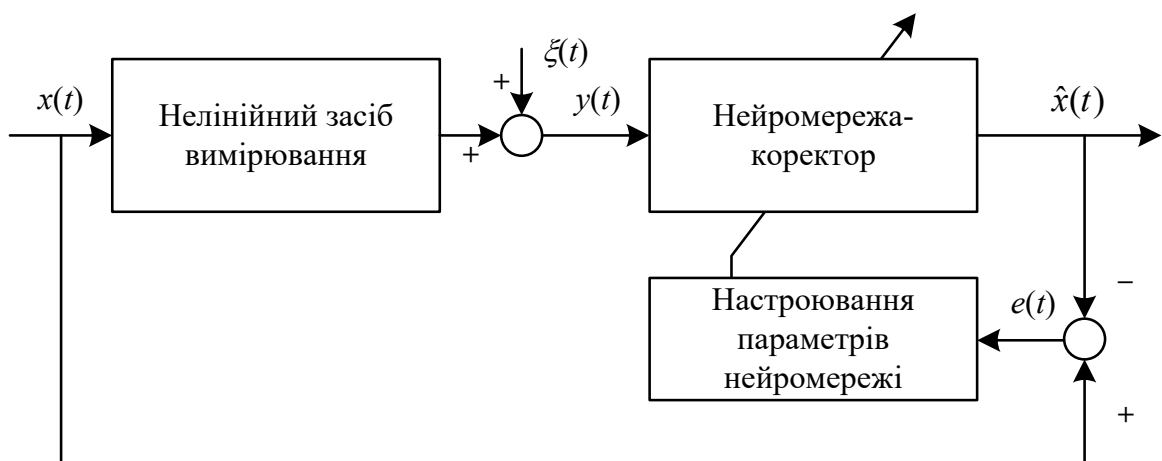


Рисунок 5.2 – Структура адаптивної системи коригування нелінійності функції перетворення засобу вимірювання

Алгоритм навчання здійснює настроювання синаптичних вагових коефіцієнтів нейромережі таким чином, щоб забезпечити найкраще наближення її вихідного сигналу  $\hat{x}(t)$  до відомого значення вхідного сигналу  $x(t)$ . При цьому мінімізується середньоквадратичне значення помилки корекції  $e(t) = x(t) - \hat{x}(t)$  за допомогою однієї з відомих градієнтних процедур оптимізації.

Відмінною особливістю ШНМ є здатність до навчання, що зробило їх пріоритетним напрямком досліджень в області штучного інтелекту. Теорія ШНМ стрімко розвивається в останні роки, що сприяє підвищенню зацікавленості до їх використання в різноманітних галузях науки, техніки, економіки, медицини, військової справи тощо. Завдяки таким властивостям, як висока надійність функціонування, завадостійкість, здатність до узагальнення, можливість реалізації складних багатовимірних відображень, нейромережі широко застосовуються для розпізнавання образів і класифікації, прийняття рішень і управління, оптимізації, апроксимації функцій, прогнозування, фільтрації, організації пам'яті [1-11].

Виходячи з вищевказаних особливостей ШНМ, зокрема здатності бути універсальним апроксиматором складних нелінійних залежностей, саме нейромережу найбільш доцільно використовувати у якості коригуючого пристрою, як було показано у ряді досліджень [17-27].

## 5.2 Пристрій корекції на базі багатошарового персептрона

Багатошарова нейронна мережа прямого поширення, або багатошаровий персептрон (Multilayer Perceptron, MLP) складається з декількох послідовно з'єднаних між собою шарів формальних нейронів: вхідного, прихованих та вихідного (рис. 5.3). Нейрони одного шару не зв'язані між собою, виходи нейронів  $n$ -го шару надходять на входи нейронів наступного  $n+1$ -го шару. Вхідний векторний сигнал надходить на входи нейронів першого шару, а вихідні сигнали останнього шару формують

вихідний векторний сигнал мережі. Конфігурація MLP визначається кількістю шарів, кількістю нейронів у кожному шарі та функціями активації нейронів. Навчання полягає у коригуванні коефіцієнтів зв'язків між нейронами таким чином, щоб забезпечити необхідні значення вихідних сигналів мережі. У процесі навчання багатошарова нейромережа здатна виявляти складні залежності між вхідними та вихідними сигналами та здійснювати узагальнення. При використанні парадигми навчання «з учителем» датасет, на якому навчається нейромережа, повинен бути розмічений або промаркований, тобто містити правильні відповіді (виходи нейромережі) на кожен вхідний зразок навчальної послідовності.

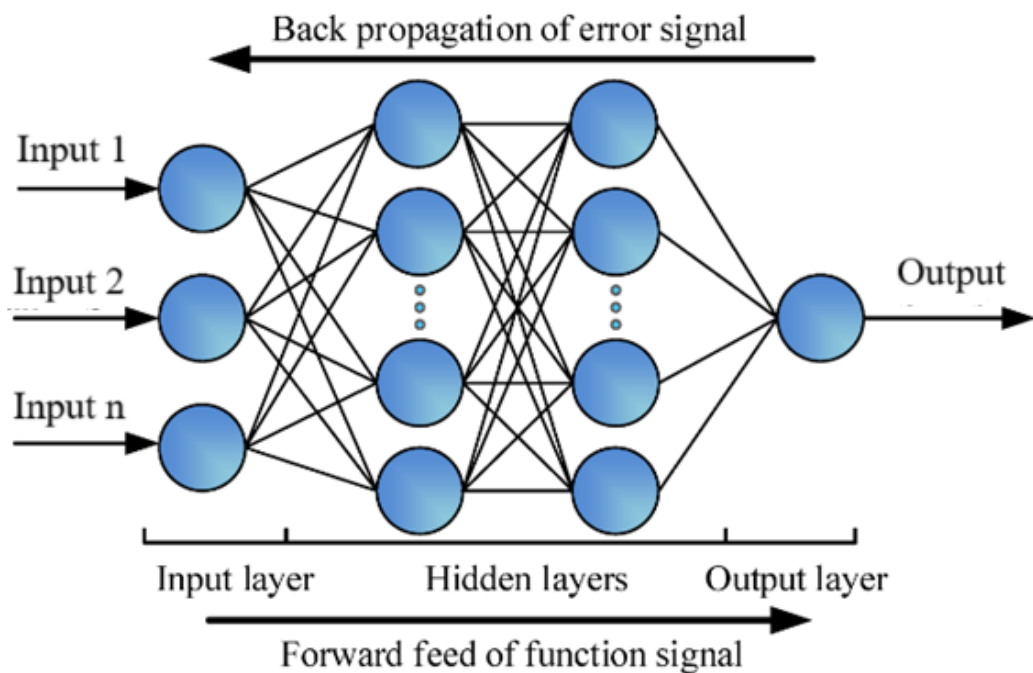


Рисунок 5.3 – Багатошаровий перцептрон

Пропонується реалізувати пристрій для коригування нелінійності функції перетворення засобу вимірювання на базі тришарового перцептрона, структуру якого наведено на рис. 5.4.

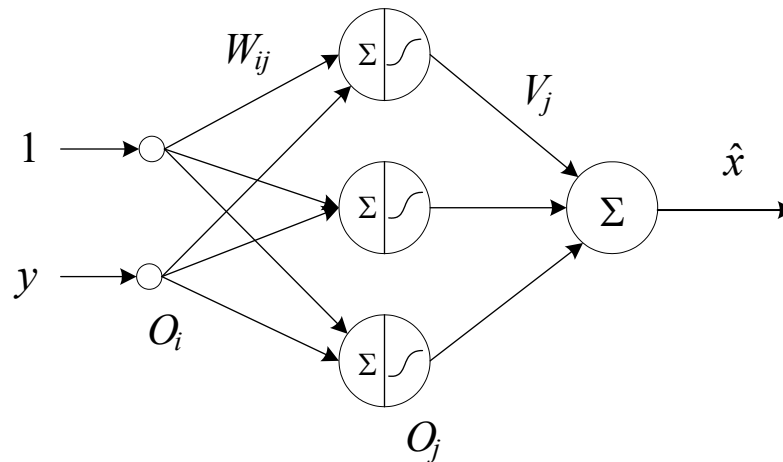


Рисунок 5.4 – MLP-коректор

Вихідний шар персептрона утворено одним нейроном, який формує сигнал  $\hat{x}$  як зважену суму вихідних сигналів нейронів прихованого шару

$$\hat{x} = \sum_{j=1}^n V_j O_j, \quad (5.1)$$

де  $O_j$  – вихідний сигнал  $j$ -го нейрона прихованого шару;

$V_j$  – синаптична вага  $j$ -го входу нейрона вихідного шару;

$n$  – кількість нейронів у прихованому шарі.

Прихований шар MLP утворено нейронами з сигмоїдальними функціями активації. Кожен нейрон цього шару описується такими рівняннями

$$O_j = \frac{1}{1 + e^{-S_j}}, \quad S_j = \sum_{i=1}^m W_{ij} O_i,$$

де  $O_i$  – вихідний сигнал  $i$ -го нейрона вхідного шару;

$W_{ij}$  – синаптична вага  $i$ -го входу  $j$ -го нейрона прихованого шару;

$m$  – кількість нейронів у вхідному шарі.

Вхідний шар нейронів утворено самими вхідними сигналами нейромережі, якими у рамках задачі, що розглядається, є вихідний сигнал засобу вимірювання  $y$  та постійний сигнал, який дорівнює одиниці і вводить для опису постійного зміщення.

Навчання MLP базується на мінімізації середньоквадратичної помилки

$$E = \frac{1}{2} \sum_{k=1}^N e^2(k) = \frac{1}{2} \sum_{k=1}^N (x(k) - \hat{x}(k))^2 \quad (5.2)$$

шляхом настроювання синаптичних вагових коефіцієнтів нейронів за допомогою градієнтних методів. Найбільш популярними є два алгоритми навчання:

- метод зворотного поширення помилки (back propagation);
- алгоритм Левенберга-Марквардта (Levenberg-Marquardt algorithm, LMA), який є комбінацією методу Гауса-Ньютона та методу градієнтного спуску.

У нашому випадку для навчання MLP доцільніше використовувати алгоритм Левенберга-Марквардта, оскільки він має вищу швидкість збіжності порівняно з іншими градієнтними методами оптимізації.

Датасет для процедури навчання нейромережі-коректора буде складатися з пар значень  $\{<y(1), x(1)>, <y(2), x(2)> \dots <y(N), x(N)>\}$ , які отримують при калібруванні засобу вимірювання, подаючи на вхід зразковий сигнал  $x(k)$  і отримуючи відповідне йому значення вихідного сигналу  $y(k)$ .

### 5.3 Пристрій корекції на базі радіально-базисної нейромережі

Ще одним універсальним апроксиматором складних функціональних залежностей є радіально-базисна нейронна мережа (Radial Basis Function Network, RBFN). Архітектура такої мережі складається з трьох шарів: вхідного, на який поступає вектор вхідних сигналів; прихованого, що

складається з нейронів радіального типу; вихідного, який формує зважену лінійну комбінацію виходів нейронів прихованого шару (рис. 5.5).

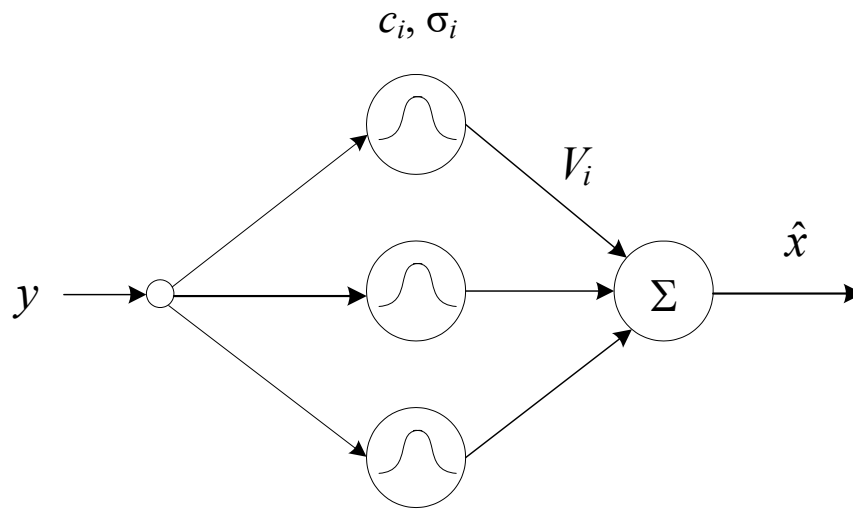


Рисунок 5.5 – RBFN-коректор

Якщо в якості функції активації нейронів прихованого шару використовувати функцію Гауса, то рівняння такого перетворювача матиме вигляд

$$\hat{x} = \sum_{i=1}^n V_i \exp\left(-\frac{(y - c_i)^2}{2\sigma_i^2}\right), \quad (5.3)$$

де  $c_i$  – центр  $i$ -ої базисної функції;

$\sigma_i$  – радіус (ширина)  $i$ -ої базисної функції.

Навчання RBFN полягає у визначенні лінійних вагових коефіцієнтів  $V_i$  вихідного нейрона, центрів  $c_i$  та радіусів  $\sigma_i$  нейронів прихованого шару.

При цьому можливі такі варіанти:

- задаються фіксовані значення центрів та радіусів нейронів прихованого шару, а вагові коефіцієнти вихідного нейрона визначаються шляхом навчання;

- центри і радіуси визначаються шляхом самонавчання (найчастіше за допомогою методів кластеризації), а потім корекцією ваг вихідного нейрона мінімізується цільова функція (5.2);

- всі параметри мережі визначаються за допомогою навчання з учителем, наприклад методом зворотного поширення помилки.

#### 5.4 Результати експериментальних досліджень

Для вивчення властивостей запропонованих нейромережевих систем коригування функції перетворення засобів вимірювання застосовувалось комп'ютерне моделювання в середовищі MATLAB з використанням бібліотеки Neural Network Toolbox. Текст програми моделювання наведено у додатку Б. У процесі моделювання досліджувався вплив виду нелінійності на якість коригування функції перетворення, при цьому для моделювання нелінійного засобу вимірювання використовувались такі види функцій [16]:

- синусоїдальна  $y = \sin(ax + b)$ ;
- степенева  $y = ax^b$ ;
- гіперболічна  $y = a + b/x$ ;
- дробово-лінійна I  $y = \frac{1}{a + bx}$ ;
- дробово-лінійна II  $y = \frac{cx}{a + bx}$ ;
- показникова I  $y = ae^{bx}$ ;
- показникова II  $y = ae^{b/x}$ ;
- логарифмічна I  $y = a + b \ln x$ ;
- логарифмічна II  $y = a + b/\ln x$ .

Адитивний шум на виході засобу вимірювання моделювався гаусівським випадковим процесом з нульовим середнім і стандартним відхиленням 0,005. Якість коригування оцінювалась середньоквадратичним

відхиленням оцінки сигналу після коригування від його дійсного значення

$$s = \sqrt{\frac{1}{N-1} \sum_{k=1}^N (x(k) - \hat{x}(k))^2}. \quad (5.4)$$

Для порівняння здійснювалось моделювання аналогічної системи коригування функції перетворення на базі поліноміального апроксиматора 5-го степеню. Результати моделювання наведено на рисунках 5.6-5.15 та в таблиці 5.1. Кружечками на графіку функції позначено точки навчальної вибірки.

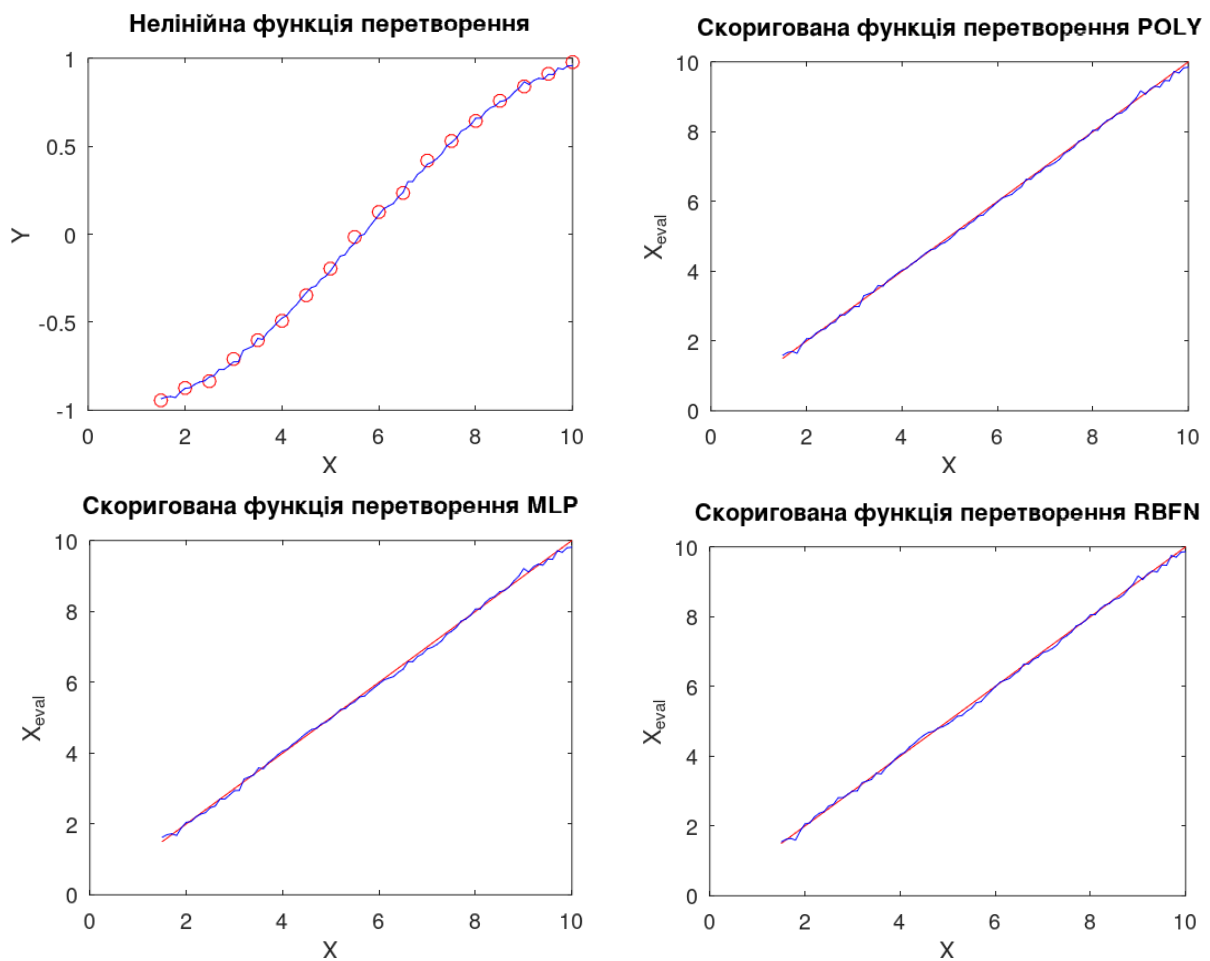


Рисунок 5.6 – Коригування синусоїдальної функції перетворення

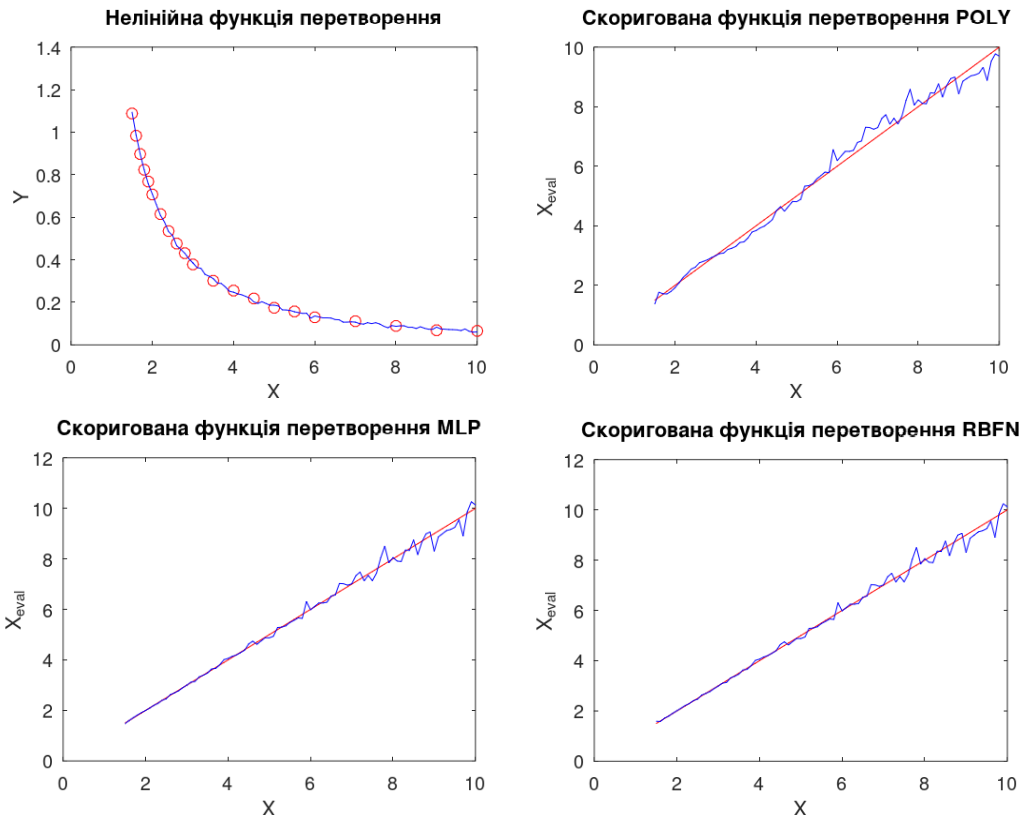


Рисунок 5.7 – Коригування степеневої функції перетворення

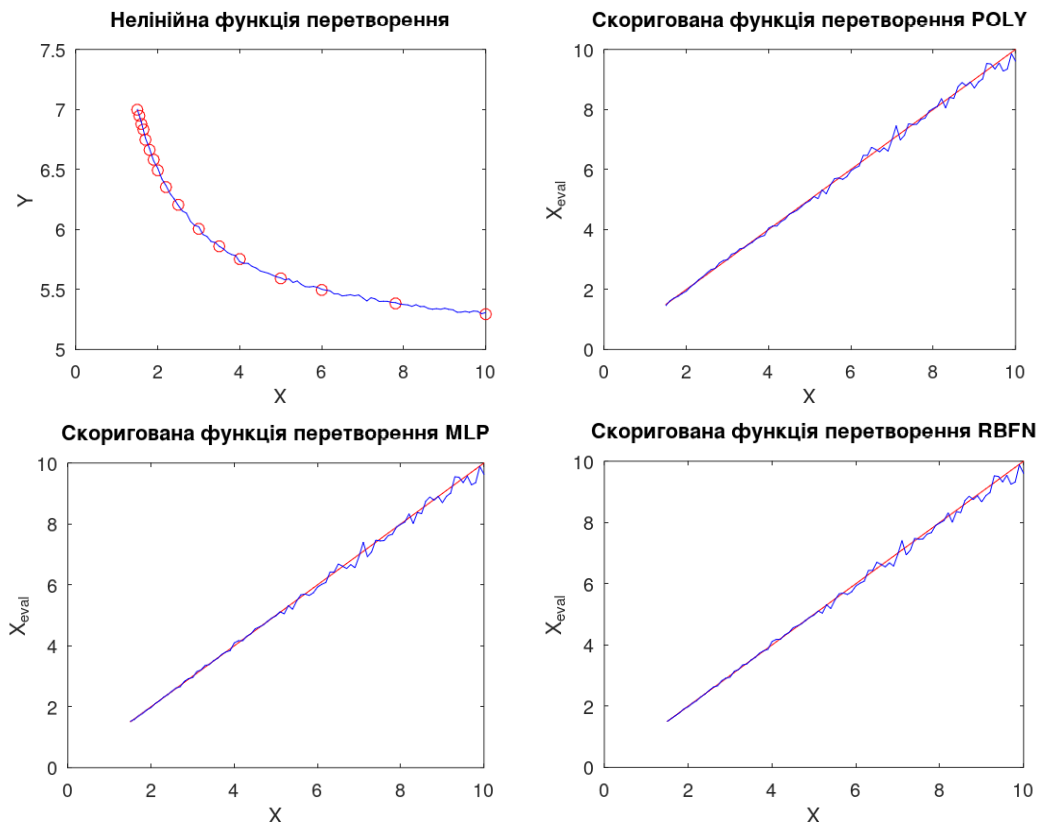


Рисунок 5.8 – Коригування гіперболічної функції перетворення

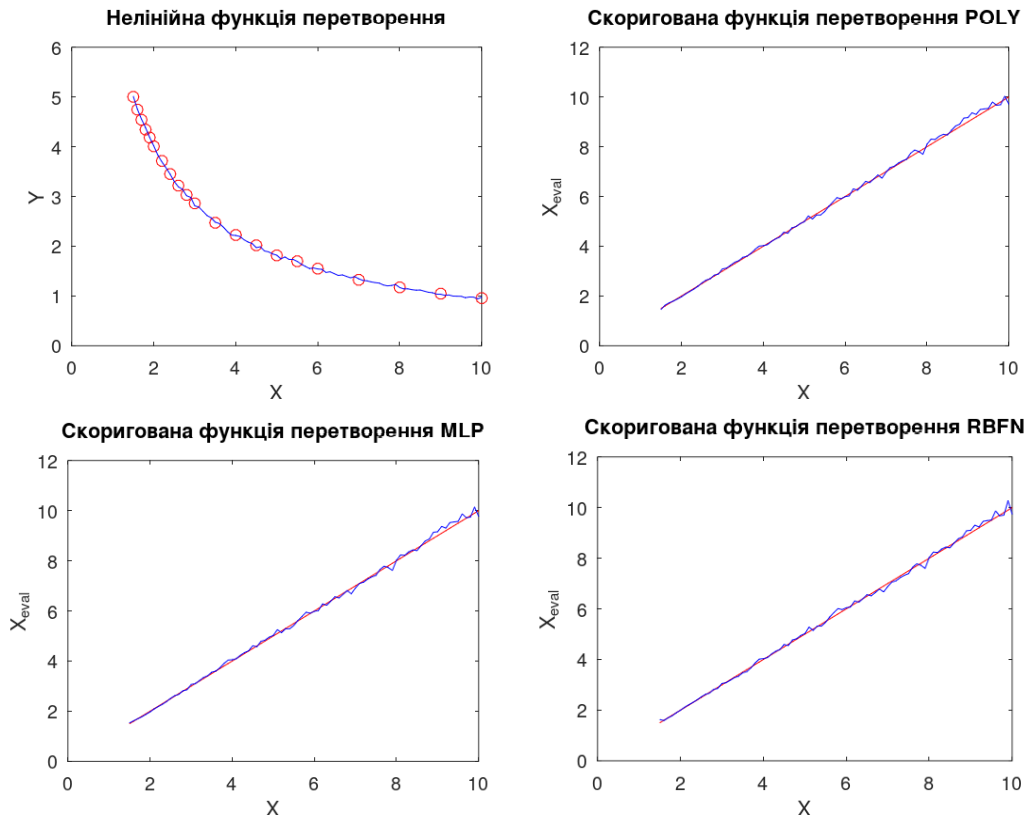


Рисунок 5.9 – Коригування дробово-лінійної I функції перетворення

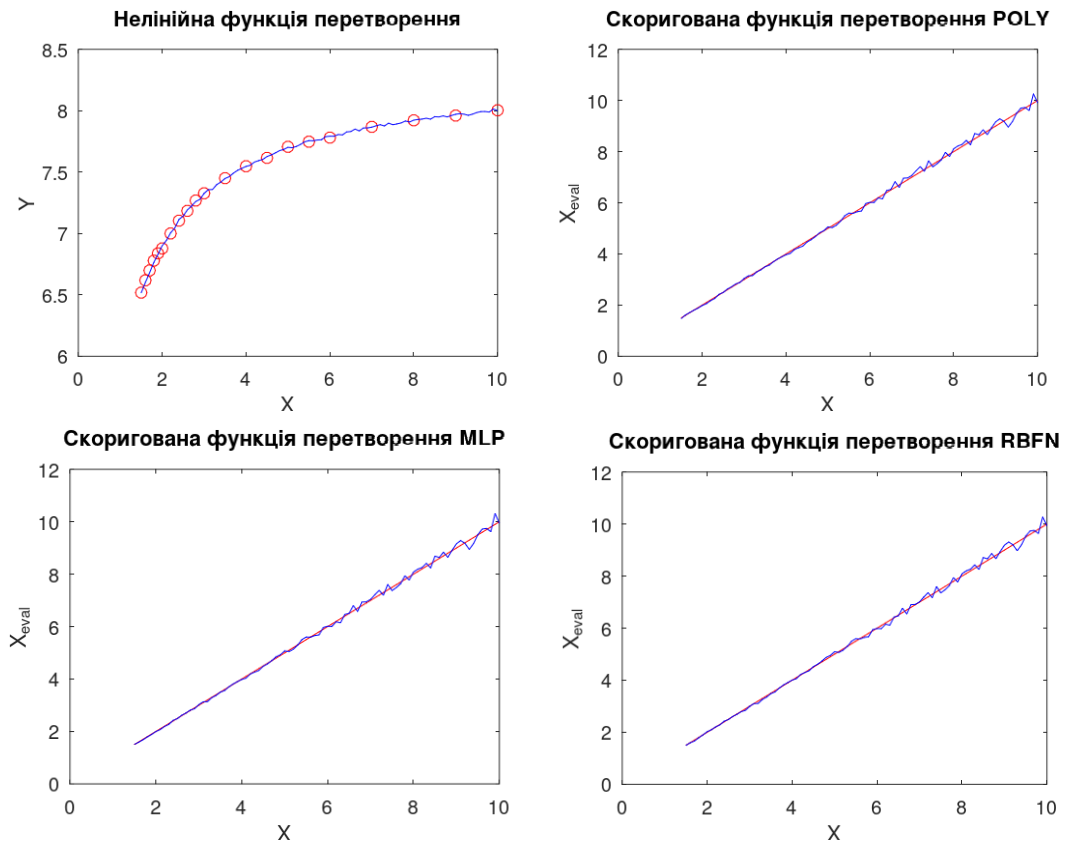


Рисунок 5.10 – Коригування дробово-лінійної II функції перетворення

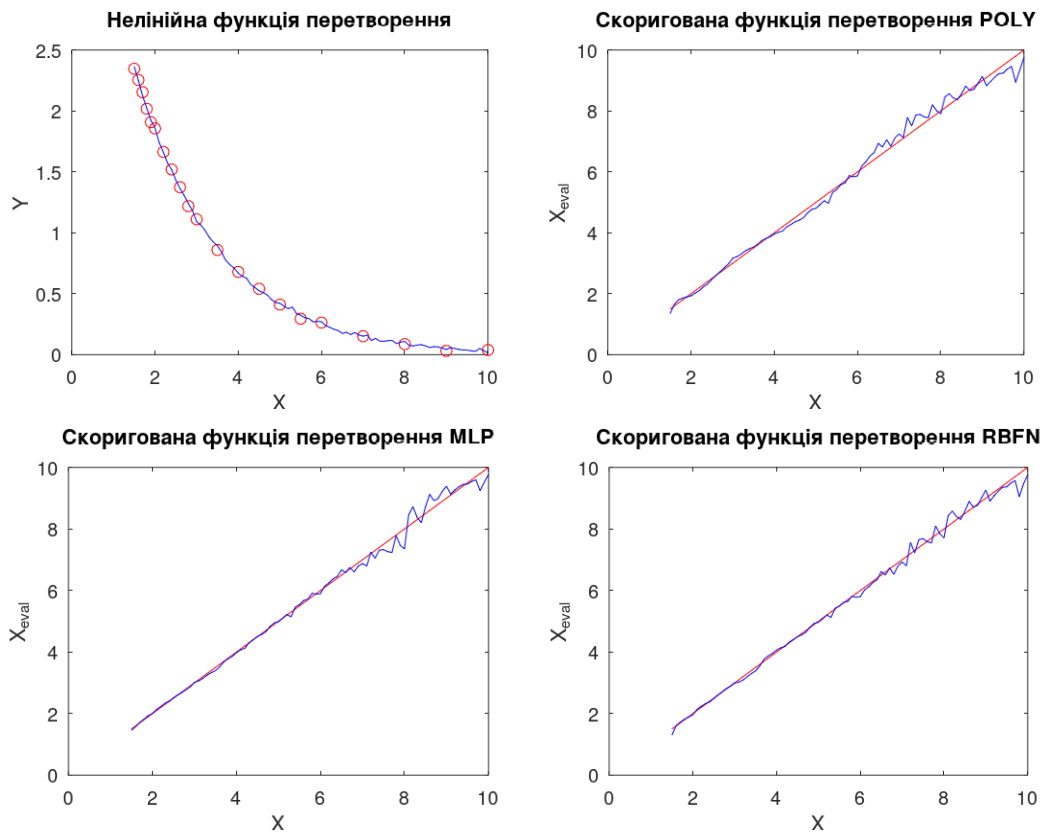


Рисунок 5.11 – Коригування показникової I функції перетворення

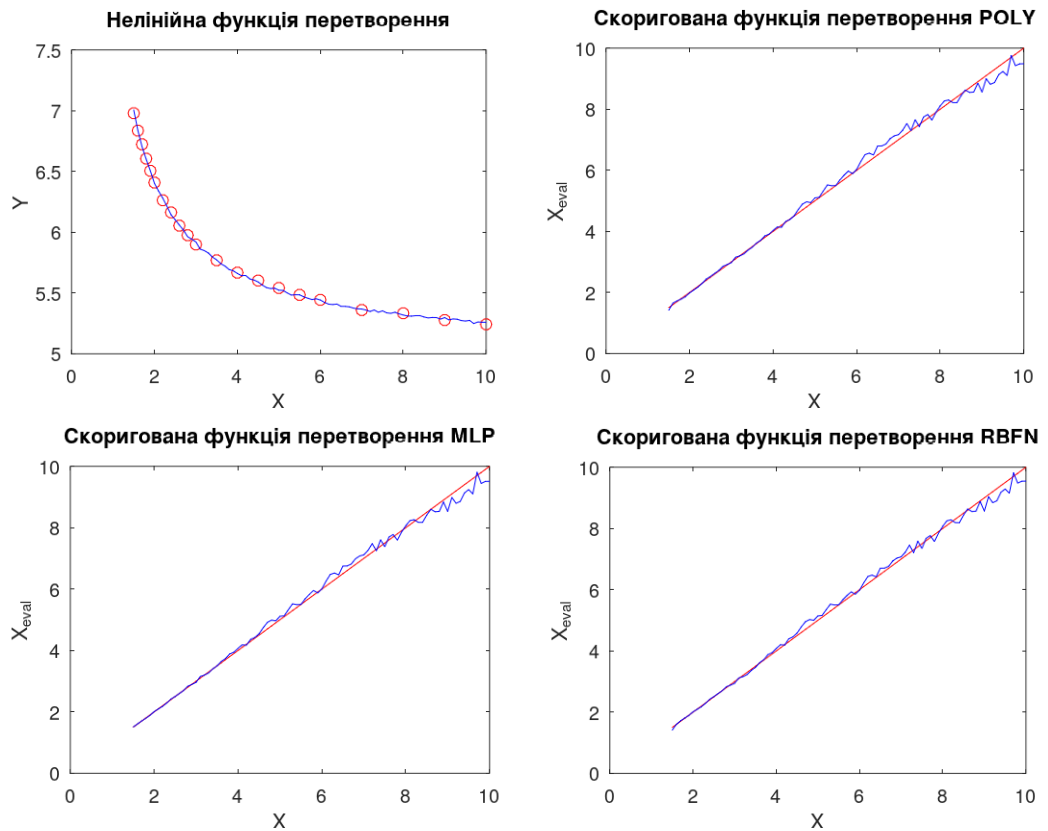


Рисунок 5.12 – Коригування показникової II функції перетворення

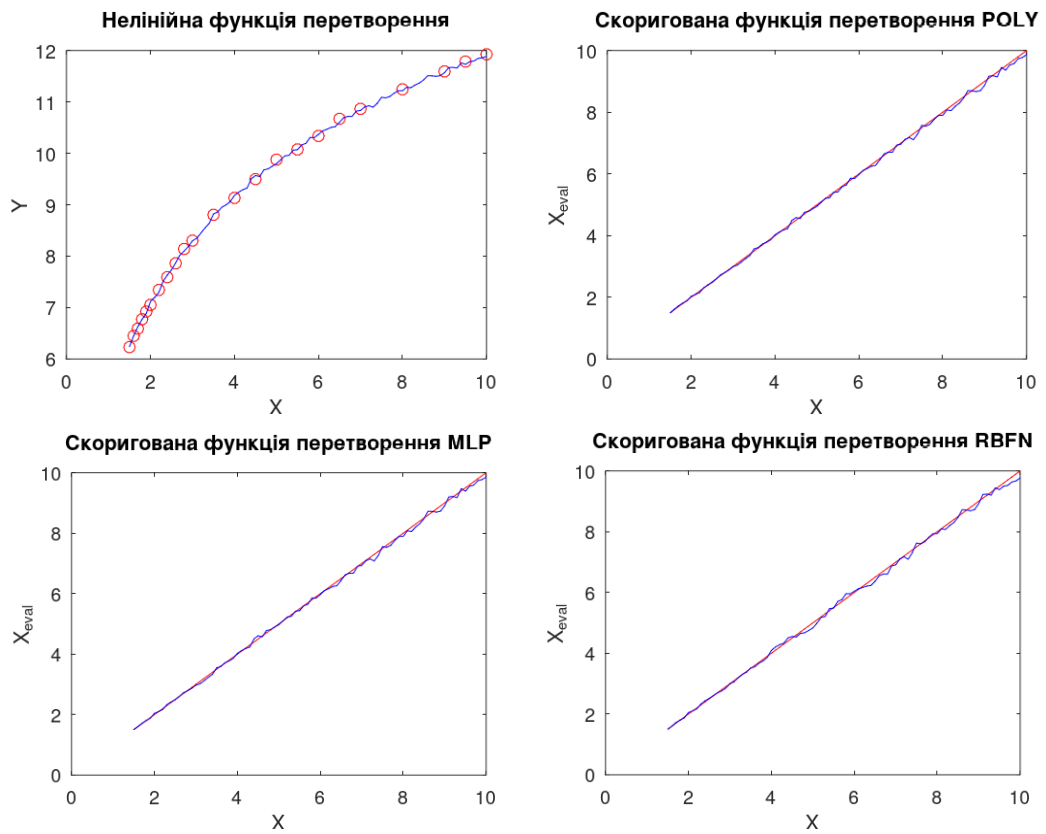


Рисунок 5.13 – Коригування логарифмічної I функції перетворення

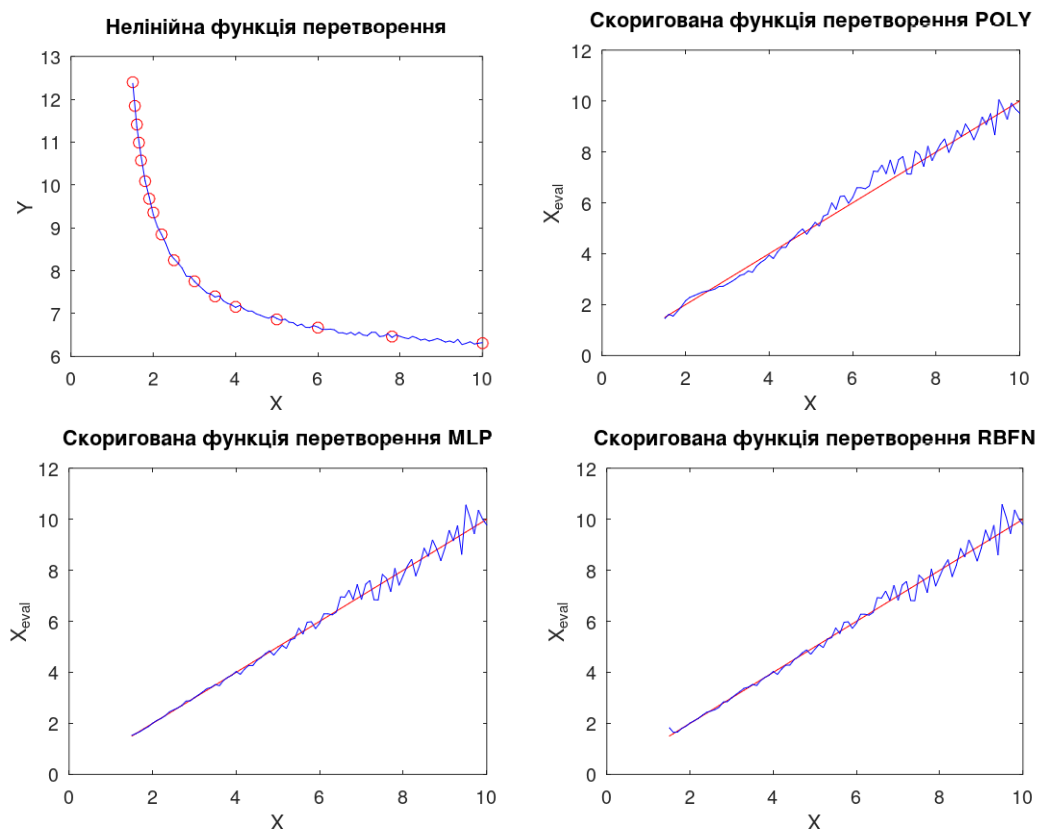


Рисунок 5.14 – Коригування логарифмічної II функції перетворення

Таблиця 5.1 – Середньоквадратична похибка корекції для різних нелінійних функцій і коректорів

Функція	Формула	Середньоквадратична похибка корекції		
		POLY	MLP	RBFN
1. Синусоїдальна	$y = \sin(0,3x - 1,7)$	0,0615	0,0752	0,0700
2. Степенева	$y = 2 \cdot x^{-1,5}$	0,2621	0,1992	0,1983
3. Гіперболічна	$y = 5 + 3/x$	0,1360	0,1374	0,1414
4. Дробово-лінійна I	$y = \frac{1}{0,05 + 0,1x}$	0,0974	0,0985	0,1016
5. Дробово-лінійна II	$y = \frac{10x}{0,5 + 1,2x}$	0,1076	0,1061	0,1053
6. Показникова I	$y = 5 \cdot e^{-0,5 \cdot x}$	0,2160	0,1870	0,1611
7. Показникова II	$y = 5 \cdot e^{0,5/x}$	0,1780	0,1727	0,1590
8. Логарифмічна I	$y = 5 + 3 \cdot \ln x$	0,0704	0,0715	0,0911
9. Логарифмічна II	$y = 5 + 3/\ln x$	0,3099	0,2814	0,2849

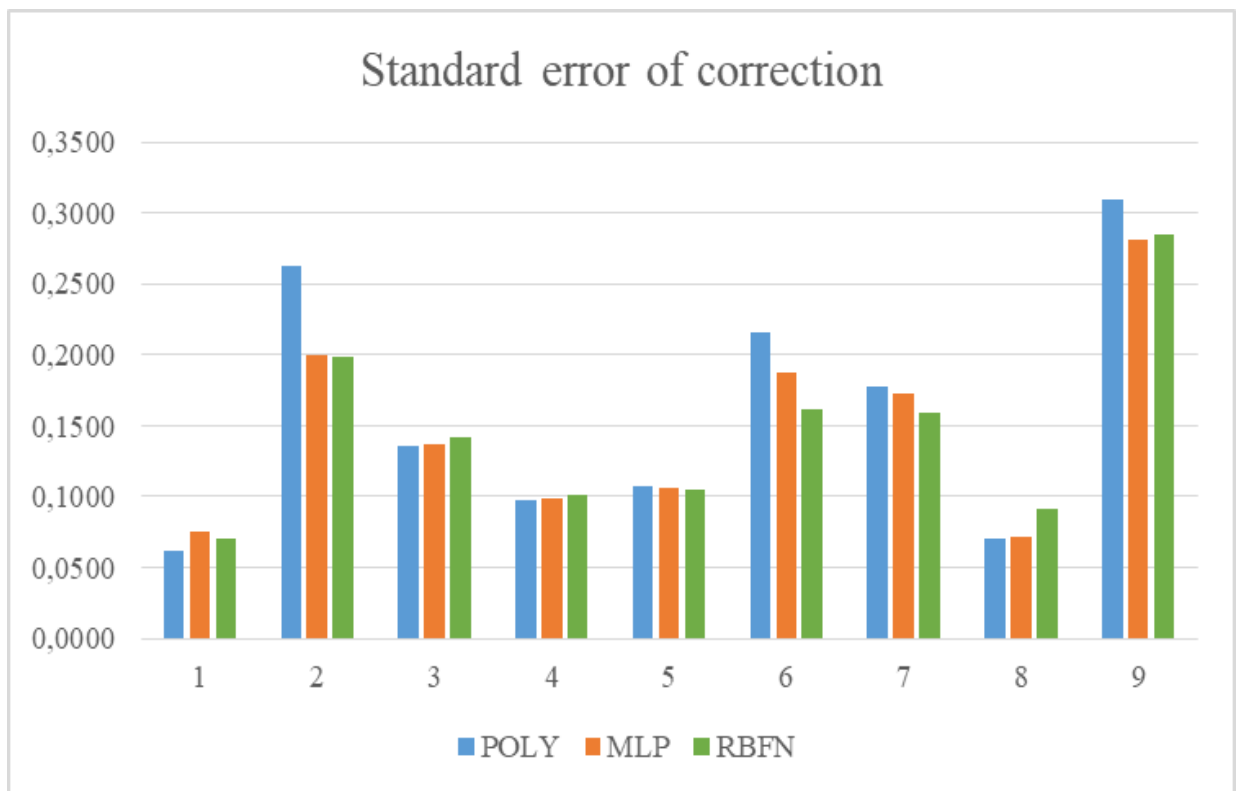


Рисунок 5.15 – Похибки коригування

Аналіз результатів моделювання, наведених на рисунках 5.6-5.15 та в табл. 5.1, свідчить про те, що запропонована нейромережева система коригування нелінійності функції перетворення засобу вимірювання на базі MLP та RBFN є придатною для коригування досить широкого класу нелінійних функцій перетворення, які зустрічаються у вимірювальних задачах. Порівняння характеристик нейромережевих коректорів і аналогічної системи на базі поліноміального апроксиматора показує, що середньоквадратична похибка корекції для деяких видів нелінійної функції перетворення у традиційного та нейромережевих коректорів відрізняється несуттєво, а для інших нейромережевий підхід показує кращі результати.

З цього можна зробити висновок, що результати моделювання повністю підтверджують працездатність запропонованої адаптивної системи коригування нелінійності функції перетворення на базі ШНМ і узгоджуються з теоретичними припущеннями [28]. Перевагою запропонованого підходу є інваріантність нейромережевих коректорів до виду нелінійної характеристики засобу вимірювання та можливість синтезувати такі системи через навчання, без залучення складних методів проектування. Це значно розширює можливості застосування таких систем у метрологічній практиці, зокрема в каналах інформаційно-вимірювальних систем, де досить просто реалізувати методи цифрової обробки вимірювальної інформації.

Використання запропонованого коректора на базі нейромережевого підходу дозволить суттєво зменшити систематичну похибку вимірювання, викликану невідповідністю номінальної та реальної функції перетворення засобу вимірювання.

## ВИСНОВКИ

В кваліфікаційній роботі розглянуто загальну характеристику та властивості штучних нейронних мереж, їх класифікацію та області застосування. Детально розглянуто структуру та властивості багатошарового персептрона та методи його навчання, зокрема метод зворотного поширення помилки та алгоритм Левенберга-Марквардта. Також у роботі розглянуто архітектуру та методи навчання радіально-базисних нейромереж.

Проаналізовано підходи до побудови лінійних та нелінійних функцій перетворення засобів вимірювання, зокрема традиційний підхід з використанням методу найменших квадратів та нейромережеві методи апроксимації нелінійних функцій.

Запропоновано узагальнену структуру адаптивної системи коригування нелінійності функції перетворення засобу вимірювання за допомогою штучних нейронних мереж. Пристрій корекції реалізовано у двох варіантах: на базі багатошарового персептрона та на базі радіально-базисної нейромережі. Для вивчення властивостей запропонованих нейромережевих систем коригування функції перетворення засобів вимірювання застосовувалось комп'ютерне моделювання в середовищі MATLAB з використанням бібліотеки Neural Network Toolbox. У процесі моделювання досліджувався вплив виду нелінійності на якість коригування функції перетворення.

Порівняння характеристик нейромережевих коректорів і аналогічної системи на базі поліноміального апроксиматора показує, що середньоквадратична похибка корекції для деяких видів нелінійної функції перетворення у традиційного та нейромережевих коректорів відрізняється несуттєво, а для інших нейромережевий підхід показує кращі результати.

Результати комп'ютерного моделювання повністю підтверджують працездатність запропонованої адаптивної системи коригування нелінійності

функції перетворення на базі ШНМ і узгоджуються з теоретичними припущеннями. Перевагою запропонованого підходу є інваріантність нейромережових коректорів до виду нелінійної характеристики засобу вимірювання та можливість синтезувати такі системи через навчання, без залучення складних методів проектування. Це значно розширює можливості застосування таких систем у метрологічній практиці, зокрема в каналах інформаційно-вимірювальних систем, де досить просто реалізувати методи цифрової обробки вимірювальної інформації.

Використання запропонованого коректора на базі нейромережевого підходу дозволить суттєво зменшити систематичну похибку вимірювання, викликану невідповідністю номінальної та реальної функції перетворення засобу вимірювання.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Wasserman P. D. *Neural Computing: Theory and Practice*. N.Y., Van Nostrand Reinhold, 1989. 230 p.
2. Haykin S. *Neural Networks. A Comprehensive Foundation*. New Jersey, Prentice Hall, 2006. 842 p.
3. Rojas R. *Neural Networks: A Systematic Introduction*. Springer-Verlag, Berlin, 1996. 502 p.
4. Бодянский Е. В., Руденко О. Г. *Искусственные нейронные сети: архитектуры, обучение, применение*. Харьков : ТЕЛТЕХ, 2004. 369 с.
5. Руденко О. Г., Бодянский Е. В. *Искусственные нейронные сети*. Харьков : ООО “Компания СМІТ”, 2005. 408 с.
6. Haykin S. *Neural Networks and Learning Machines*. 3rd ed. Pearson, Prentice Hall, 2009. 906 p.
7. Субботін С. О. *Нейронні мережі : теорія та практика: навч. посіб.* Житомир : Вид. О. О. Євенок, 2020. 184 с.
8. Bishop C.M. *Pattern Recognition and Machine Learning*. Springer. 2006. 738 p.
9. *Теорія та практика нейронних мереж : навч. посіб.* /Л. М. Добровська, І. А. Добровська. К.: НТУУ «КПІ» Вид-во «Політехніка», 2015. 396 с.
10. Hornik K., Stinchcombe M., White H. *Multilayer feedforward networks are universal approximators*. *Neural Networks*. 1989. 2(5). PP. 359-366.
11. Ткаліченко С. В. *Штучні нейронні мережі : навч. посібник*. Кривий Ріг : Державний університет економіки і технологій, 2023. 150 с.
12. Levenberg K. *A Method for the Solution of Certain Problems in Least Squares*. *Quart. Appl. Math.* 1944. Vol. 2. P. 164–168.
13. Marquardt D. *An Algorithm for Least-Squares Estimation of Nonlinear Parameters*. *SIAM Journal on Applied Mathematics*. 1963. No 11 (2). P. 431–441.

14. Ranganathan A. The Levenberg-Marquardt algorithm. Georgia Tech College of Computing, 2008. P. 1–5.
15. Li M., Liu X. Radial Basis Function Networks for Function Approximation. *Journal of Intelligent Information Systems*. 2017. No 48(2). PP. 267–283.
16. Грановский В. А., Сирая Т. Н. Методы обработки экспериментальных данных при измерениях. Л. : Энергоатомиздат, 1990. 288 с.
17. Запорожец О. В., Коротенко В. А., Овчарова Т. А. Компенсация нелинейности функции преобразования измерительных устройств с помощью искусственной нейронной сети. *Системи управління, навігації та зв'язку*. 2010. Вип. 4(16). С. 99–103.
18. Дегтярев А. В., Запорожец О. В., Овчарова Т. А. Адаптивная система компенсации нелинейности функции преобразования измерительных устройств на базе трехслойного персептрона. *Електротехнічні та комп'ютерні системи*. 2012. № 06(82). С. 235–241.
19. Дегтярев А. В., Запорожец О. В., Овчарова Т. А. Идентификация нелинейной функции преобразования с помощью искусственной нейронной сети. *Український метрологічний журнал*. 2013. Вип. № 2. С. 4–8.
20. Запорожец О. В., Овчарова Т. А. Компенсация нелинейности динамических измерительных преобразователей при помощи нейросетевых моделей. *Метрологія та прилади*. 2014. Вип. № 1/II/(45). С. 74–77.
21. Запорожец О. В., Овчарова Т. А., Руженцев И. В. Компенсация нелинейности полупроводникового терморезистора с помощью искусственной нейронной сети. *Системи обробки інформації*. 2015. №6 (131). С. 64–67.
22. Дегтярев А. В., Запорожец Н. О., Запорожец О. В. Коррекция функции преобразования измерительного канала с помощью радиально-базисной нейросети // Сборник трудов VI Международной научно-технической конференция «Метрология, информационно-измерительные технологии и системы» (24-25 октября 2017 г.). Харьков, 2017. С. 42–43.

23. Дегтярев А. В., Запорожец Н. О., Запорожец О. В. Компенсация нелинейности функции преобразования канала информационно-измерительной системы с помощью радиально-базисной нейросети // Тези доповідей п'ятої міжнародної науково-технічної конференції «Проблеми інформатизації» (13-15 листопада 2017 р.). Черкаси – Баку – Бельсько-Бяла – Полтава, 2017. С. 85–86.

24. Дегтярьов О. В., Запорожець Н. О., Запорожець О. В. Адаптивна корекція функції перетворення вимірювального каналу за допомогою радіально-базисної нейромережі // Матеріали восьмої міжнародної науково-технічної конференції «Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління» (26-27 квітня 2018 р.). Полтава – Баку – Харків – Жиліна, 2018. С. 98–99.

25. Zaporozhets O. V., Shtefan N. V. Using Artificial Neural Network for Compensation of Semiconductor Thermistor Nonlinearity // 2019 IEEE 8th International Conference on Advanced Optoelectronics and Lasers (CAOL), Sozopol, Bulgaria, 6-8 Sept. 2019. PP. 703–706.

26. Zaporozhets O. V., Zaporozhets N. O. Using artificial neural network for compensation of thermistor conversion function nonlinearity // Тези доповідей сьомої міжнародної науково-технічної конференції «Проблеми інформатизації» (13-15 листопада 2019 р.). Черкаси – Харків – Баку – Бельсько-Бяла, 2019. С. 111.

27. Al Rawashdeh L., Zakharov I., Zaporozhets O. Nonlinearity Correction in Dynamic Measuring Devices Using Neural Network Models. *Pomiary Automatyka Robotyka*. R 24. Nr 4/2020. PP. 57–60.

28. Авакін С. М., Довгополий С. О., Запорожець О. В., Мощенко І. О. Коригування функції перетворення засобів вимірювання з використанням нейромережевого підходу. *Системи управління, навігації та зв'язку*. 2025. Вип. 2(80).