

ОДЕГОВ МИКОЛА

Державний університет інтелектуальних технологій і зв'язку

<https://orcid.org/0000-0001-5526-2487>e-mail: onick_64@ukr.net**ГАДЖИЄВ МАТІН**

Державний університет інтелектуальних технологій і зв'язку

<https://orcid.org/0000-0001-7280-3863>e-mail: gadjievmm@ukr.net**ПЕРЕКРЕСТОВ ІГОР**

Державний університет інтелектуальних технологій і зв'язку

<https://orcid.org/0009-0007-3805-8143>e-mail: perekrestov.igor@gmail.com**ЩЕРБА СЕРГІЙ**

Державний університет інтелектуальних технологій і зв'язку

<https://orcid.org/0009-0004-2457-9122>e-mail: sergeyscherba2@gmail.com**КАІН ВАДИМ**

Державний університет інтелектуальних технологій і зв'язку

e-mail: kain.vadim50@gmail.com

АЛГОРИТМ ОПТИМІЗАЦІЇ ПАРАМЕТРІВ ФУНКЦІЙ АКТИВАЦІЇ ПДБ-МЕРЕЖ

Дана робота продовжує дослідження у напрямку розробки швидких алгоритмів вирішення задач кластеризації та класифікації. Розглянуті у попередніх роботах гібридні алгоритми включають обробку вхідних даних за допомогою штучних нейронних мереж, заснованих на принципі далекодії-близькодії (ПДБ-мережі). Суть цього принципу полягає у тому, щоб задавати перехідні матриці перетворень між шарами мережі практично статичними, а основні операції оптимізації мережі покласти на навчання окремих нейронів.

У даній роботі обґрунтовується алгоритм навчання функцій активації нейронів шарів ПДБ-мережі. У якості таких нелінійних функцій запропоновано параметричний варіант відомої функції ReLU – Relu2. Єдиним параметром функцій Relu2 є коефіцієнт відсічення (скалярний параметр). У роботі доведено, що використання функцій Relu2 підвищує роздільну здатність штучної нейронної мережі. Для порівняння запропонованого методу за критеріями точності та швидкості обрано базовий алгоритм K найближчих сусідів. Прийнятий найпростіший та найшвидший варіант одного найближчого сусіда.

Варіант застосування ПДБ-мережі у даній роботі зводиться до використання двошарової структури з одним вхідним та одним вихідним шаром. Функції активації вихідного шару оптимізуються за показником максимальної середньої відстані перетворених зразків з різних класів. Ці перетворені зразки розподіляються по підкласам (кластерам) у рівній кількості цих кластерів для всіх класів. Класифікація тестового набору відбувається за мінімізацією відстані до центроїдів кластерів, тобто використовується метод K -середніх. Виконаний порівняльний аналіз показує, що використаний в роботі гібридний алгоритм суттєво перевищує алгоритм найближчого сусіда за критерієм швидкості, не поступаючи йому за критерієм точності.

Ключові слова: задача класифікації, принцип далекодії-близькодії, функції активації, алгоритм k -середніх, алгоритм k -найближчих сусідів, критерій точності, критерій часу.

NICK ODEGOV, HADZHYIEV MATIN, PEREKRESTOV IHOR, SERHII SHCHERBA, VADYM KAIN

State University of Intellectual Technologies and Telecommunications

ALGORITHM FOR OPTIMIZATION OF LCR-NETWORK ACTIVATION FUNCTIONS PARAMETERS

This paper continues research aimed at developing efficient algorithms for solving clustering and classification problems. The previously considered hybrid algorithms involve processing input data using artificial neural networks based on the Long-and-Close-Range (LCR) principle. According to this principle, the transition matrices between network layers are defined as nearly static, while the primary network optimization effort focuses on training individual neurons.

The paper proposes and substantiates an algorithm for optimizing activation functions in neurons of LCR-network layers. As nonlinear activation functions, we introduce a parametric modification of the well-known ReLU function, named Relu2. This function includes a single scalar parameter, the cutoff coefficient. It is demonstrated that applying Relu2 functions enhances the discriminative capability (resolution) of artificial neural networks.

To evaluate the accuracy and computational efficiency of the proposed method, we selected the classical K -nearest neighbors (KNN) algorithm as a baseline. Specifically, we adopted its simplest and fastest variant—one nearest neighbor (1-NN).

In this study, the LCR-network implementation utilizes a two-layer architecture consisting of an input and an output layer. The activation functions of the output layer are optimized to maximize the average distance between transformed samples from different classes. These transformed samples are subsequently divided equally among subclasses (clusters) across all classes. Classification of the test set is performed by minimizing distances to cluster centroids, thus employing the K -means clustering method.

A comparative analysis demonstrates that the hybrid algorithm significantly outperforms the nearest neighbor algorithm in terms of computational speed while achieving comparable accuracy.

Keywords: classification problem, long-and-close-range principle, activation functions, K -means algorithm, K -nearest neighbors algorithm, accuracy criterion, time criterion.

Стаття надійшла до редакції / Received 05.04.2025

Прийнята до друку / Accepted 18.04.2025

Аналіз джерел та постановка проблеми

Дана робота продовжує дослідження алгоритмів штучних нейронних мереж, побудованих за принципом далекоді-близькодії (ПДБ-мережі, або LCR-net) [1, 2]. В якості тренувального полігону для визначення характеристик якості алгоритмів знову використовується структурований та розмічений набір зразків рукописних цифр MNIST. Корисність цього набору полягає у дуже великій варіативності зразків та іноді протиріччя між мітками та людським уявленням про правильне написання звичайних цифр. На рис. 1 наведено деякі приклади зображень з набору MNIST, які підтверджують це положення: даний набір є жорстким полігоном для випробування алгоритмів розпізнавання.

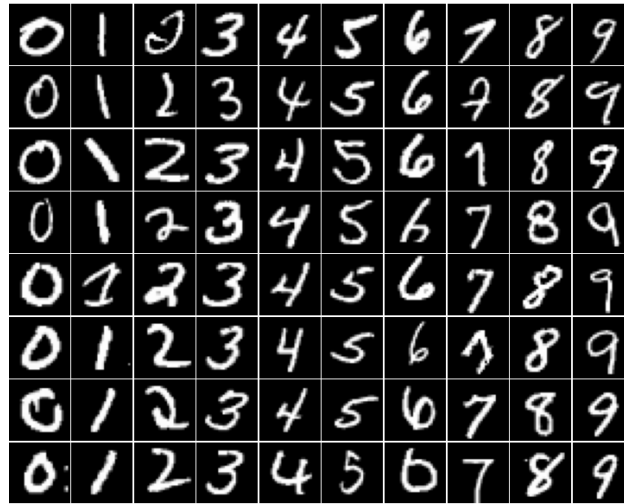


Рис. 1. Приклади зображень рукописних цифр набору MNIST

Дійсно, 2 у першому рядку скоріше нагадує 0, рівно як і 6 в нижньому рядку; 7 у першому рядку мало чим відрізняється від 1 у четвертому рядку; 2 у другому рядку мало чим відрізняється від 1 і т. д. У сучасних наукових дослідженнях цей набір рукописних цифр також використовується як той самий полігон. Так, у роботі [3] досліджуються лінійні алгоритми класифікації MNIST з використанням статистичних критеріїв сумісного використання груп алгоритмів. При цьому у більшості варіантів досягається відсоток правильних рішень задачі класифікації близько 91% (далі – показник точності, або accuracy). Тільки при використанні значної кількості різних алгоритмів (deep ensemble) лінійного перетворення фазового простору з різними параметрами досягається точність близько 96%.

Найкращі результати розпізнавання дають алгоритми штучних нейронних мереж (ШНМ). Слід відзначити сучасні тенденції гібридизації алгоритмів вирішення задач класифікації, в тому числі класифікації образів набору MNIST. Зокрема, у роботі [4] послідовно використовуються алгоритми згорткових нейронних мереж (Convolutional Neural Networks – CNN) та опорних векторів (Support Vector Machine – SVM). За свідченням авторів цієї роботи такий гібридний алгоритм дозволяє отримати точність до 99,28%.

На жаль, як у статті [3], так і у статті [4] немає жодної згадки про інші важливі параметри алгоритму – час навчання на розпізнавання. Серед багатьох робіт за даною тематикою знайшлась рідкісна публікація, де саме параметру часу приділена певна увага [5]. У даній роботі також використано гібридний алгоритм, перша ступень якого зводиться до редукції первинної розмірності фактор-простору до простору з меншою кількістю факторів, що забезпечує поряд з іншими операціями підвищення швидкості навчання більш ніж у 100 разів порівняно з алгоритмами-прототипами. Один з таких прототипів – метод К найближчих сусідів (K nearest neighbors – KNN). Відмічено, що точність з використанням даного порівняльного методу може бути навіть більше 99%.

Гібридний алгоритм з використанням мереж, заснованих на принципі далекоді-близькодії (ПДБ-мережі, або LCR-net) запропоновано у роботі [2]. У той час як у згаданій роботі вирішувалась задача кластеризації, в даній роботі вирішується задача класифікації.

Метою роботи є розробка та обґрунтування алгоритмів навчання нейронів шарів ПДБ-мереж.

База порівняння – алгоритм найближчих сусідів

Алгоритм із застосуванням ПДБ-мережі у даній роботі порівнюється з класичним алгоритмом К найближчих сусідів в рівних умовах:

- алгоритми реалізовано на мові Python без прискорення за допомогою GPU;
- із всього набору зразків MNIST використовується лише частина – тренувальний набір 60000 зразків;
- цей набір розподіляється на $N_{tr} = 50000$ зразків та $N_{tst} = 10000$ зразків;
- в залежності від плану конкретного числового експерименту із першої частини обирається певна кількість зразків для навчальної вибірки $N_{ed} \leq N_{tr}$.

У даній роботі алгоритм найближчих сусідів реалізується у найпростішому варіанті: $K = 1$; кожний зразок із 10000 зразків тестового набору порівнюється із розміченим зразком навчальної вибірки і відноситься до класу найближчого з цих зразків. При цьому порядок алгоритму складає $Ned \times N_{test}$ операцій. Результати числових експериментів для цих умов наведені у табл. 1.

Таблиця 1

Результати класифікації методом одного найближчого сусіда

Кількість зразків Ned	Час класифікації, с	Точність рішення, %
500	31	83,36
1000	62	86,72
2000	112	89,89
3000	168	91,19
5000	285	92,93
10000	547	94,29
20000	1139	95,57
50000	2865	96,72

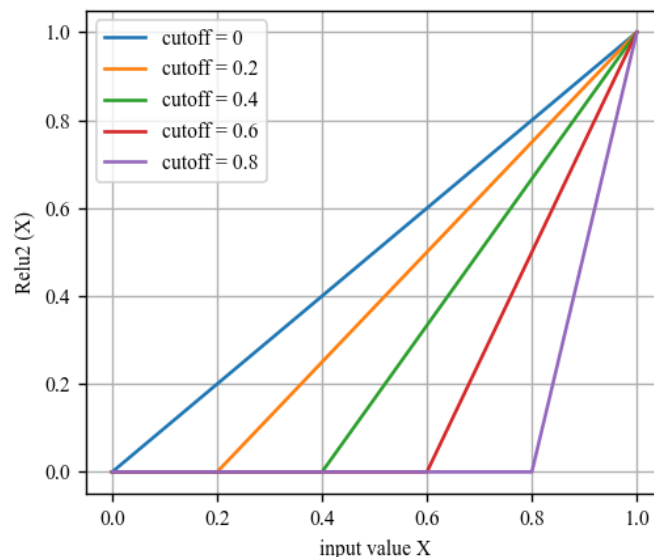
Аналіз даних у табл. 1 показує, що час класифікації практично лінійно залежить від порядку алгоритму – у даному випадку від кількості зразків у навчальній вибірці. Втім, отриманий найкращий результат (96,72% правильних рішень) поступається згаданому у роботі [5] (більше 99% правильних рішень). Можливо, у цих дослідженнях використовувалась більша кількість найближчих сусідів, $K > 1$. Однак, слід також враховувати, що час вирішення задачі при цьому приблизно збільшиться у K разів.

Визначення параметрів функцій активації ПДБ-мережі та алгоритм їх оптимізації

Алгоритми навчання ПДБ-мереж передбачають оптимізацію функцій активації окремих нейронів. Для цього функції активації мають мати хоча б один вільний параметр, який і підлягає оптимізації за певним критерієм. Такі функції можна сконструювати на основі відомих нелінійних функцій активації. У роботах [1, 2] було використано параметричні сигмоїди. У даній роботі пропонується функція Relu2, яка є параметризованим варіантом звичайної функції ReLU. Дана функція залежить від одного вільного параметру – коефіцієнту відсічення $0 \leq cutoff < 1$ і визначається як:

$$\text{Relu2}(x, cutoff) = \begin{cases} 0, & x \leq cutoff \\ \frac{x - cutoff}{1 - cutoff}, & x > cutoff \end{cases} \quad (1)$$

Декілька варіантів графіків функції Relu2 при різних значеннях $cutoff$ наведено на рис. 2.

Рис. 2. Графіки функції Relu2 при різних значеннях $cutoff$

Як видно із графіків на рис.1 та із визначення (1), при $cutoff = 0$ нейрон виконує функцію лінійного підсилювача з коефіцієнтом передачі 1. В інших випадках функція Relu2 подавляє вхідні значення, які не перевищують коефіцієнта відсічення $cutoff$. Певним недоліком таких функцій є необхідність приведення значень вхідних сигналів у діапазон від 0 до 1. Втім, амплітуди пікселів вхідних зображень завжди позитивні, а перетворення за допомогою перехідних матриць ПДБ-мереж не призводить до появи значень менше нуля. Нормалізація верхніх значень до одиниці потребує певного часу, але сприяє збереженню енергетичного балансу мережі на всіх ступенях перетворень.

Застосування функції Relu2 підвищує роздільну здатність ПДБ-мережі. Покажемо це на прикладі елементарної задачі дихотомії.

Лема про збільшення відстаней. Нехай на вхід нейрону діють два різні сигнали з амплітудами x_1 та x_2 та визначено модульну метрику між ними: $R_x = |x_1 - x_2|$. Тоді для вихідних сигналів $y_1 = \text{Relu2}(x_1, \text{cutoff})$ та $y_2 = \text{Relu2}(x_2, \text{cutoff})$ знайдеться таке значення коефіцієнту відсічення cutoff , що $R_y = |y_1 - y_2| > R_x$.

Дійсно, якщо $\text{cutoff} \leq x_1, x_2$, то згідно визначення (1) $R_y = R_x / (1 - \text{cutoff})$. Але $1 - \text{cutoff} < 1$ і відстань R_y буде більша ніж R_x у $1/(1 - \text{cutoff})$ разів.

Дане просте положення елементарно доводиться у випадку квадратичної або Евклідової метрики. Різниця лише у тому, що у випадку квадратичної метрики співвідношення між відстанню на вході та виході нейрону буде збільшено у $1/(1 - \text{cutoff})^2$ разів.

Дана Лема розповсюджується на випадок кількох класів. Наприклад, є різні сигнали з на вході нейрону з трьох класів. Ранжуємо їх, наприклад із порядком зростання: $x_1 < x_2 < x_3$ і встановимо $\text{cutoff} = x_1$. Тоді прийдемо до того ж самого результату: відстань між сигналами на виході збільшиться порівняно з відстанями між ними на вході нейрону.

Лему нескладно узагальнити на статистичні моделі. А саме: якщо вхідні значення двох класів мають унімодальні розподілення, а їх математичні очікування різні, то в результаті перетворення за функцією Relu2 відстані між вихідними значеннями будуть у середньому більше, ніж відстані між вхідними значеннями.

Безпосередньо із Лематикає наступна **Теорема про збільшення роздільної здатності шару ПДБ-мережі**. Нехай на вхід шару з кількістю нейронів N поступають два різні вектори $X1$ та $X2$ тієї ж розмірності. Тоді знайдеться такий вектор Cutoff розмірності N , що $R(Y1, Y2) > R(X1, X2)$ де: $Y1 = \text{Relu2}(X1, \text{Cutoff})$, $Y2 = \text{Relu2}(X2, \text{Cutoff})$.

Для доказу скористуємось алгоритмічним способом (тобто, методом індукції). Отже спочатку встановимо всі значення компонентів вектора Cutoff у нульові стани: $\text{cutoff}_i = 0$, $i \in [0, 1, \dots, N]$. Тоді згідно визначення (1): $Y1 \equiv X1, Y2 \equiv X2 \Rightarrow R_0(Y1, Y2) \equiv R(X1, X2)$. На наступному кроці залишимо налаштування всіх нейронів окрім першого. Оптимізуємо параметр першого нейрону cutoff_1 згідно Лемати так, щоб $R(y_{1,1}, y_{2,1}) > R(x_{1,1}, x_{2,1})$. Оскільки відстані вхідних та вихідних векторів є сумою складових відстаней по нейронам, то $R_1(Y1, Y2) > R_0(Y1, Y2)$.

Далі оптимізуємо коефіцієнт відсічки для другого нейрону і $R_2(Y1, Y2) > R_1(Y1, Y2)$.

Продовжуючи цей процес визначаємо, що послідовне налаштування окремих нейронів лише збільшує відстань між вихідними сигналами. Більш того, якщо складові вектора Cutoff визначені найбільш оптимально, то кращого результату підвищення роздільної здатності ПДБ-мережі досягнути неможливо, оскільки нейрони функціонально відокремлені.

Дані елементарні математичні положення та хід їх доказу дозволяють декомпонувати вирішення задач великої розмірності. Розглянемо простий випадок, якщо шар мережі має відносно невелику кількість 1000 нейронів. Нехай визначення параметрів вектору Cutoff виконується методом повного перебору лише з двох варіантів для кожного нейрону: або $\text{cutoff} = 0$, або, наприклад, $\text{cutoff} = 0.5$. Таких можливих варіантів (порядок алгоритму) буде 2^{1000} – астрономічне число. Такий ефект колись Р. Белман назвав «прокляттям розмірності». Але при послідовній оптимізації порядок алгоритму зменшується до дуже незначної кількості операцій: лише 2000.

Окрім цього, для прискорення визначення оптимальних значень коефіцієнтів відсічки можна використовувати найменші з конкурентних значень сигналів з різних класів. Дані результати використані у задачі оптимізації ПДБ-мережі за наступним алгоритмом.

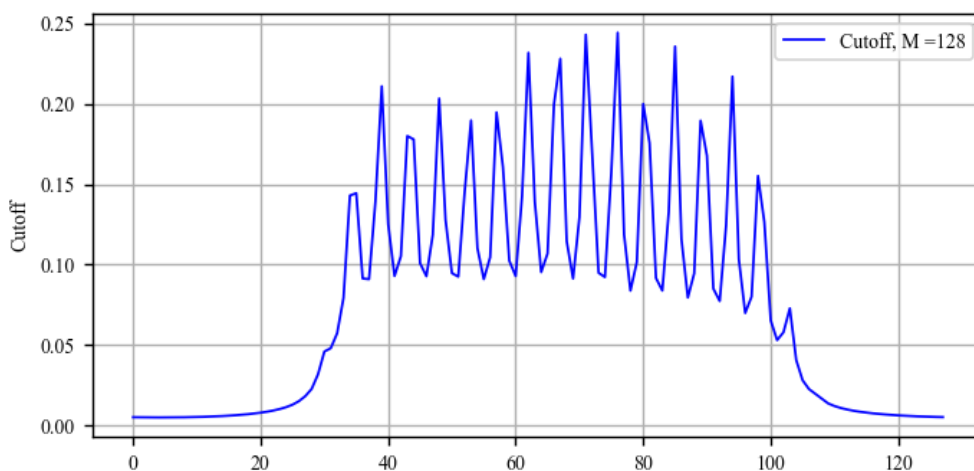
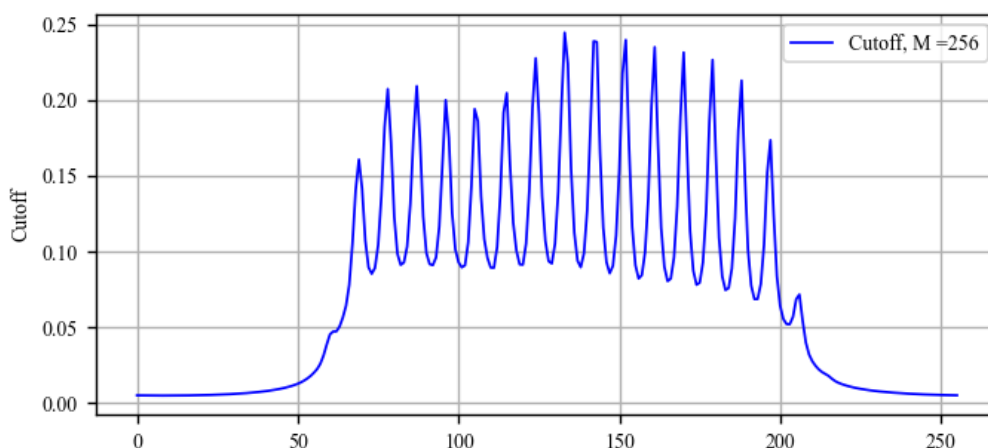
Набір тренувальних даних розподіляється на мікро батчі – блоки, які включають 10 зразків різних класів: зображення цифр 0, 1, ..., 9. Для наступного блоку з номером L визначається поточне значення індивідуального оптимального вектора $\text{Cutoff}(L)$.

Критерій оптимальності – максимум середньої відстані між зразками різних класів, для чого по суті і призначені нелінійні функції типу Relu2 . Оптимальний вектор значень $\text{Cutoff_opt}(L+1)$ на наступному кроці визначається за рекурентним алгоритмом:

$$\text{Cutoff_opt}(L+1) = \frac{L \cdot \text{Cutoff_opt}(L) + \text{Cutoff}(L)}{L+1} \quad (2)$$

В цілому алгоритм навчання ПДБ-мережі у конкретному випадку включає редукцію розмірності вхідних зразків з $N = 784$ пікселів до розмірності $M = \text{out_size}$ за допомогою перехідної матриці, що визначається за принципом далекодії-близькодії. Кількість нейронів вихідного шару M встановлюється у конкретному числовому експерименті. Результати конкретних експериментів також залежать від кількості елементів навчальної вибірки N_{ed} . Результати оптимізації масивів Cutoff в двох випадках наведено на рис. 3 та рис. 4.

Пропонований алгоритм класифікації фактично заводиться до гібридного алгоритму застосування ПДБ-мережі та алгоритму К середніх (K-means). Алгоритм навчання ПДБ-мережі визначає оптимальні параметри налаштування нейронів вихідного шару. Образи зразків вихідного шару усереднюються і розподіляються по рівній кількості $C > 1$ кластерів для кожного класу. Далі зразки із тестового набору перетворюються у навченої ПДБ-мережі, тобто переходять у простір меншої розмірності $M = \text{out_size}$. Далі знаходиться найменша відстань редуктованого зразка з одним із центротидів кластерів.

Рис. 3. Графіки значень масивів Cutoff для $M=128$ Рис. 4. Графіки значень масивів Cutoff для $M=128$

Результат розпізнавання (класифікації) приймається як правильний, якщо мітка кластеру (клас кластеру) дорівнює мітці тестового зразка. Результати ряду числових експериментів наведено у табл.2.

Аналіз даних у табл. 2 доводить, що використаний гібридний алгоритм з доволі високою точністю вирішує задачу класифікації. Важливо, що час вирішення цієї задачі є дуже прийнятним для вирішення практичних задач. Дійсно, якщо лише за 44 секунди правильно класифікуються більше 90% зразків з 10000, то з тим самим рівнем ймовірності правильних вирішень за 1 хвилину алгоритм класифікує більше 1 363 636 зразків. Це приблизно 2 томи тексту по 250 сторінок у кожному із розрахунку 4000 символів на сторінці. Навчання алгоритму також відбувається дуже швидко, оскільки використовуються рекурентні формули типу (2), які мають дуже високі показники збіжності.

Таблиця 2

Результати класифікації гібридним методом з використанням ПДБ-мережі та K-means

Кількість нейронів у вихідному шарі M	Кількість кластерів у класах C	Кількість середніх K	Час класифікації, с	Точність рішення, %
64	10	100	5	84,39
128	10	100	7	86,10
80	100	1000	44	91,29
128	100	1000	53	91,62
200	200	2000	102	92,94
512	200	2000	109	94,84
392	500	5000	260	95,19
128	1000	10000	472	95,74
256	1000	10000	511	96,63

Обговорення, висновки та рекомендації

Порівняння різних алгоритмів може бути коректним лише за рівних умов. У даній роботі обидва алгоритми реалізовано в одному й тому ж програмному та апаратному середовищі. Але вони значно відрізняються за параметрами проведених числових експериментів. Так, алгоритм найближчого

сусіда взагалі не потребує навчання, а в алгоритмах, заснованих на використанні ПДБ-мереж, певний час втрачається на оптимізацію параметрів цих мереж. Втім, ці два алгоритми можна порівняти за часом класифікації 10000 зразків тренувального набору за умови приблизно рівної точності. Таке порівняння дозволяють зробити окремі співставні результати, що наведено у табл. 3.

Таблиця 3

Порівняння часу класифікації для досягнення приблизно рівної точності з використанням конкурентних методів: метода найближчого сусіда та гібридного метода з використанням ПДБ-мережі

Метод найближчого сусіда		Гібридний метод з використанням ПДБ-мережі	
Час класифікації, с	Точність рішення, %	Час класифікації, с	Точність рішення, %
62	86,72	7	86,10
168	91,19	44	91,29
285	92,93	102	92,94
547	94,29	109	94,84
1139	95,57	472	95,74
2865	96,72	511	96,63

Аналіз даних у табл. 3 дозволяє зробити однозначний висновок: приблизно однакові результати класифікації за критерієм точності досягаються гібридним алгоритмом з використанням навченої ПДБ-мережі за значно менший час. Так, найкращі показники – більше 96% правильних вирішень досягаються гібридним алгоритмом у 5,6 разів швидше, ніж алгоритмом найближчого сусіда.

Показники точності двох розглянутих алгоритмів поступаються кращим опублікованим показникам – більше 99% правильних вирішень. Втім, ПДБ-мережа у даному випадку моделюється у найпростішому варіанті: її структура складається лише із вхідного та вихідного шарів. Як показують попередні дослідження [1, 2], збільшення глибини ПДБ-мережі може сприяти підвищенню її ефективності за критерієм точності.

Втім, аналіз таких мереж є предметом наступних досліджень.

Література

1. Одегов М.А. Принцип далекодії-близькодії у задачах структуризації та навчання штучних нейронних мереж / М.А. Одегов, Ю.О. Бабіч // Вісник Хмельницького національного університету. Технічні науки. - 2024. - №3. - С. 357 - 365. [HTTPS://DOI.ORG/10.31891/2307-5732-2024-337-3-54](https://doi.org/10.31891/2307-5732-2024-337-3-54).
2. Одегов М.А. Порівняльний аналіз алгоритмів кластеризації / М.А. Одегов, Д.Г. Багачук, І.С. Перекрестов, Я.О. Петрович // Вісник Хмельницького національного університету. Технічні науки. - 2024. - №4. - С. 406-413. [HTTPS://DOI.ORG/10.31891/2307-5732-2024-339-4-61](https://doi.org/10.31891/2307-5732-2024-339-4-61).
3. Magnus Malmström, Isaac Skog, Daniel Axehill, Fredrik Gustafsson, Uncertainty quantification in neural network classifiers—A local linear approach, Automatica, Volume 163, 2024, 111563, ISSN 0005-1098, <https://doi.org/10.1016/j.automatica.2024.111563>.
4. Savita Ahlawat, Amit Choudhary, Hybrid CNN-SVM Classifier for Handwritten Digit Recognition, Procedia Computer Science, Volume 167, 2020, Pages 2554-2560, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2020.03.309>.
5. Ekaterina Plesovskaya, Sergey Ivanov, Hierarchical Classification on the MNIST Dataset Using Truncated SVD and Kernel Density Estimation, Procedia Computer Science, Volume 212, 2022, Pages 368-377, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2022.11.021>.

References

1. Odegov M.A. Princip dalekodiyyi-blizykodiyyi u zadachah strukturizatsiyi ta navchannya shtuchnih neyronnih merezh / M.A. Odegov, Yu.O. Babich // Visnik Hmel'nickogo nacionalnogo universitetu. Tehnichni nauki. - 2024. - №3. - S. 357 - 365. [HTTPS://DOI.ORG/10.31891/2307-5732-2024-337-3-54](https://doi.org/10.31891/2307-5732-2024-337-3-54).
2. Odegov M.A. Porivnyalnij analiz algoritmiv klasterizatsiyi / M.A. Odegov, D.G. Bagachuk, I.S. Perekrstov, Ya.O. Petrovich // Visnik Hmel'nickogo nacionalnogo universitetu. Tehnichni nuki. - 2024. - №4. - S. 406-413. [HTTPS://DOI.ORG/10.31891/2307-5732-2024-339-4-61](https://doi.org/10.31891/2307-5732-2024-339-4-61).
3. Magnus Malmström, Isaac Skog, Daniel Axehill, Fredrik Gustafsson, Uncertainty quantification in neural network classifiers—A local linear approach, Automatica, Volume 163, 2024, 111563, ISSN 0005-1098, <https://doi.org/10.1016/j.automatica.2024.111563>.
4. Savita Ahlawat, Amit Choudhary, Hybrid CNN-SVM Classifier for Handwritten Digit Recognition, Procedia Computer Science, Volume 167, 2020, Pages 2554-2560, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2020.03.309>.
5. Ekaterina Plesovskaya, Sergey Ivanov, Hierarchical Classification on the MNIST Dataset Using Truncated SVD and Kernel Density Estimation, Procedia Computer Science, Volume 212, 2022, Pages 368-377, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2022.11.021>.