https://doi.org/10.31891/2307-5732-2025-353-50
УДК 004.4:004.04:004.09

**ЯКОВИН СЕРГІЙ**
Івано-Франківський національний технічний університет нафти і газу
https://orcid.org/0000-0002-3335-2892
e-mail: syakovyn@gmail.com
**МЕЛЬНИЧУК СТЕПАН**
Івано-Франківський національний технічний університет нафти і газу
https://orcid.org/0000-0002-6973-4235
e-mail: stenni@ukr.net

# АПРОКСИМАЦІЯ ІНДИКАТОРНИХ ФУНКЦІЙ У КОНТЕКСТІ ЙМОВІРНІСНОГО ДИСКРЕТНОГО ПЕРЦЕПТРОНА

*Одним із перспективних варіантів застосування перцептронних структур, як самостійних цифрових компонентів,які реалізуються на функціях кореляції та згортки, є опрацювання сигналів і потокових даних. Зокрема у контексті задач розпізнавання шаблонів, виявлення гармонійних та періодичних складових і т.д. Основні аспекти властивості дискретного перцептрона можна застосовувати для вирішення задач розпізнавання різних типів сигналів за їх формою, виявлення частотних компонентів сигналу, що є елементом спектрального аналізу, динамічного налаштування параметрів фільтрації тощо. В загальному випазку перцептрон можна розглядати як окремий самодостатній компонент функціонал якого може змінюватись в залежності від вирішуваної задачі, однак структурні, схемні, алгоритмічні та програмні рішення такого компоненту залишаються незмінними. Фактично це цифровий компонент, який може бути навчений на наборах даних, які містять як необхідні інформаційні ознаки та можливі варіанти їх спотворень і застосований для опрацювання сигналів чи даних спеціалізований компонент комп'ютерної системи.*

*У матеріалах статті розглянуто особливості реалізації перцептронів, який базується на використанні ймовірнісних оцінок дискретних синаптичних сигналів, що попередньо зазнали зміщення. Показано ключові аспекти агрегації дискретних вхідних сигналів, щ реалізуються на операція додання та ймовірнісного оцінювання. Крім того, запропоновано методи апроксимації активаційних функцій, що забезпечують гладкість і диференційованість, необхідні для ефективного навчання нейронних мереж. В ході досліджень встановлено, що використання статистичних оцінок дозволяє значно зменшити обчислювальні витрати, підвищити ефективність опрацювання сигналів та прискорити процес навчання. Отримані результати мають перспективу для використання у комп'ютерних системах та компонентах, що вирішують різноманітні прикладні задачі, зокрема опрацювання сигналів, потокових даних тощо.*

*Ключові слова: штучні нейронні мережі, перцептрон, дискретні сигнали, ймовірнісні оцінки, активаційні функції, опрацювання сигналів.*

**YAKOVYN SERHIY**
**MELNYCHUK STEPAN**
Ivano-Frankivsk National Technical University of Oil and Gas

# APPROXIMATION OF INDICATOR FUNCTIONS IN THE CONTEXT OF PROBABILISTIC DISCRETE PERCEPTRON

*One of the promising applications of perceptron structures, as independent digital components implemented based on correlation and convolution functions, is signal and stream data processing. Specifically, this includes tasks such as pattern recognition, detection of harmonic and periodic components, etc. The main aspects of the properties of the discrete perceptron can be utilized for solving problems related to recognizing various types of signals by their form, identifying frequency components of a signal (an element of spectral analysis), dynamically adjusting filtering parameters, and so on. In general, a perceptron can be viewed as a standalone self-sufficient component, whose functionality may vary depending on the task at hand, although its structural, schematic, algorithmic, and software solutions remain unchanged. Essentially, this is a digital component that can be trained on datasets containing necessary informational features and possible distortion variations and applied for signal or data processing as a specialized component of a computer system.*

*The article's materials examine the specific features of implementing perceptrons based on the use of probabilistic estimates of discrete synaptic signals, which have undergone prior bias. Key aspects of aggregating discrete input signals, realized through addition and probabilistic estimation operations, are shown. Additionally, methods for approximating activation functions that ensure smoothness and differentiability required for efficient neural network training have been proposed. Research has established that using statistical estimates can significantly reduce computational costs, enhance signal processing efficiency, and accelerate the training process. The obtained results have the potential to be utilized in computer systems and components that solve various applied tasks, including signal and stream data processing.*

*Keywords: artificial neural networks, perceptron, discrete signals, probabilistic estimates, activation functions, signal processing.*

Стаття надійшла до редакції / Received 24.04.2025
Прийнята до друку / Accepted 22.05.2025

## Вступ

Аналітичні, структурні, алгоритмічні та програмні реалізації перцептронів, як фундаментальних компонент штучних нейронних мереж, є важливим інструментом в задах розробки комп'ютерних систем для обробки інформації та підтримки прийняття рішень. Доцільно зазначити, що класичні моделі перцептронів зіштовхуються з низкою проблем, серед яких високі обчислювальні витрати, складність навчання та обмежена здатність до узагальнення. Одним із перспективних підходів до подолання згаданих проблем може бути перхід до дискретного базису, а також використання ймовірнісних оцінок дискретних синаптичних сигналів з подальшим зміщенням їхніх значень. Застосування методів статистичної оцінки та апроксимації дозволяє підвищити ефективність опрацювання сигналів, а перехід до дискретного базису - зменшити обчислювальні витрати та прискорити процес навчання нейронних мереж. Водночас, дискретний характер сигналів дозволяє скоротити простір можливих значень, що сприяє швидшому та більш ефективному навчанню нейронів.

Пецептрон, як окремий компонент, традиційно реалізується на основі функцій кореляції та згортки, результат розрахунку яких в подальшому опрацьовуються активаційним функціями. Використання статистичного оцінювання вхідних синалів дозволяє розширити функціонал таких компонентів за межі лінійного розділення. У проведених дослідженнях основну увагу приділено розробці нових методів апроксимації активаційних функцій, що забезпечують гладкість і диференційованість, необхідні для ефективного навчання. Застосування таких методів дозволяє знизити обчислювальну складність та підвищити точність роботи як перцептрону так і нейронної мережі, що відкриває нові перспективи щодо їх використання при вирішенні прикладних задач.

**Метою роботи є**: дослідження можливостей зниження обчислювальної складності дискретних перцептронних структур шляхом використання ймовірнісних підходів до оцінки вхідних сигналів, а також розробці нових методів для апроксимації функцій, що використовуються в процесі навчання нейронних мереж. Результати дослідження можуть знайти застосування у комп’ютерних компонентах та системах, що призначені для вирішення різних прикладних задач.

## Попередні дослідження

У загальному випадку, перцептрон, який реалізує опрацювання дискретних вхідних синаптичних сигналів на основі ймовірнісного оцінювання, можна описати як [1]:

$$y = \sigma(f(\bar{x} + \bar{w})) \tag{1}$$

де $\bar{x} = (x_1, x_2, \dots, x_n)$ – вектор вхідних сигналів, $\bar{w} = (w_1, w_2, \dots, w_n)$ – вектор зміщень вхідних сигналів

$f$ – функція оцінки інформаційної ентропії (чи іншої ймовірнісної оцінки) вхідних сигналів,

$\sigma$– активаційна функція (наприклад порогова функція Хевісайда *H*).

Однією із найпростіший функцій оцінювання інформаційної ентропії є функція кількості унікальних станів вхідних сигналів $Q(\bar{x} + \bar{w})$ [2].

Для побудови нейронної мережі, що буде еквівалентної до (1) необхідно емулювати функцію виду:

$$y = C(x_1, x_2) = \begin{cases} 0, & x_1 \neq x_2 \\ 1, & x_1 = x_2 \end{cases} \tag{2}$$

що нескладно виразити через функцію Хевісайда:

$$y = C(x_1, x_2) = H(x_1 - x_2) + H(x_2 - x_1) - 1 \tag{3}$$

Розширивши (3) на вектор $\bar{x} = (x_1, x_2, \dots, x_n)$ у якому визначається кількість елементів рівних $i$-му елементу отримаємо:

$$C(\bar{x}, i) = \sum_{j=1}^{n}\left[H(x_i - x_j) + H(x_j - x_i) - 1\right] = \sum_{j=1}^{n} H(x_i - x_j) + \sum_{j=1}^{n} H(x_j - x_i) - n \tag{4}$$

Для підрахунку унікальності вхідних сигналів необхідно ввести метрику унікальності:

$$U(\bar{x}, i) = \frac{1}{C(\bar{x}, i)} \tag{5}$$

обернену до кількості елементів рівних елементу і. Скомбінувавши (4) та (5), отримаємо:

$$Q(\bar{x}) = \sum_{i=1}^{n} U(\bar{x}, i) = \sum_{i=1}^{n} \frac{1}{C(\bar{x}, i)} = \sum_{i=1}^{n} \frac{1}{\sum_{j=1}^{n} H(x_i - x_j) + \sum_{j=1}^{n} H(x_j - x_i) - n} \tag{6}$$

де $Q(\bar{x})$ – кількість унікальних елементів вектору $\bar{x}$, що представляє вхідний сигнал на синаптичних входах дискретного перцептрону.

Інший підхід можна реалізувати на основі використання функції підрахунку унікальних станів тільки в одному напрямку, а саме:

$$C_{>}(\bar{x}, i) = \sum_{j=1}^{n}\left[H(x_i - x_j) + H(x_j - x_i) - 1\right] = \sum_{j=1}^{n} H(x_i - x_j) + \sum_{j=1}^{n} H(x_j - x_i) - n + i - 1 \tag{7}$$

Як результат, замість виразу (4) отримаємо вираз (7) і у такому випадку можна записати:

$$Q(\bar{x}) = \sum_{i=1}^{n} H(1 - C_{>}(\bar{x}, i)) = \sum_{i=1}^{n} H\left(n + 2 - i - \sum_{j=1}^{n} H(x_i - x_j) + \sum_{j=1}^{n} H(x_j - x_i)\right) \tag{8}$$

Використовуючи кількість унікальних станів $Q(\bar{x})$ як опосередковану або спрощену функцію оцінки інформаційної ентропії з (1) отримаємо:

$$y = \sigma(Q(\bar{x} + \bar{w})) \tag{9}$$

Функції активації в перцептронах відіграють важливу роль, зокрема додають нелінійність до моделі, а також обмежують вихідні значення нейрона в певному діапазоні, що допомагає у стабілізації навчального процесу.

Основними функціями активації, які використовуються в перцептронах є сходинкова (ступінчата) та сигмоїдна, а також часто використовують гіперболічний тангенс, ReLU, LeakyReLU, ELU та SoftMax., основні аспекти використання яких подано в таблиці 1.

Таблиця 1

**Типові функції активації, що використовують у перцептронних структурах**

| **Функція активації** | **Застосування** |
|---|---|
| Сходинкова (Threshold): $f(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$ | Використовується в класичному перцептроні для задач, де потрібно приймати рішення між двома класами так, як вона забезпечує чітке розділення класів, проте вона не дозволяє оцінити "ймовірність" або ступінь впевненості у рішенні та не підходить для задач з багатьма класами або нелінійними залежностями |
| Сигмоїдна (Sigmoid): $\sigma(x) = \frac{1}{1+e^{x}}$ | Особливістю є область визначення (0;1), що добре підходить для задач класифікації (наприклад, бінарної класифікації), проте у неї є й недоліки: проблема зникнення градієнта через те, що похідна стає дуже малою для великих значень х та вихідне значення не центрованевідносно 0, що може уповільнити навчання |
| Гіперболічний тангенс (Tanh): $tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$ | Основною особливістю є область визначення (-1;1), що надає їй перевагу над сигмоїдною функцією, оскільки вихідне значення відцентроване відносно 0, фактично вилучено можливе зміщення, проте вона також може мати проблему зникнення градієнта |
| ReLU (Rectified Linear Unit): $f(x) = max(0, x)$ | Особливість цієї функції полягає в тому, що вона швидко обчислюється і добре працює для глибоких мереж таусуває проблему зникнення градієнта в позитивній області. Проте її використання призводить до проблеми "мертвих нейронів" (вихідне значення завжди дорівнює 0) для випадку $x \leq 0$ |
| LeakyReLU: $f(x) = \begin{cases} x, & x > 0 \\ ax, & x \leq 0 \end{cases}$ | У порівнянні з RELU, вона вирішує проблему "мертвих" нейронів, допускаючи невеликий градієнт для випадків $x \leq 0$ |
| ELU (Exponential Linear Unit): $f(x) = \begin{cases} x, & x > 0 \\ a(e^{x} - 1), & x \leq 0 \end{cases}$ | Основною особливістю цієї функції є плавний перехід для негативних значень і порівнянні із ReLU |

## Основні результати

В ході дослідження перцептронних структур вираз (2) подавався через порогову функцію Хевісайда, що є незручним рішенням для застосування градієнтного навчання оскільки згадана функція недиференційована. Тому замість неї зазвичай використовують гладкі апроксимації, зокрема логістичну сигмоїду $H_a(x) = \frac{1}{1+e^{-kx}}$, арктангенс $H_a(x) = \frac{1}{2} + \frac{1}{\pi} arctan(kx)$, функцію помилок Гауса $H_a(x) = \frac{1}{2}(1 + er\ \ f(kx))$ тощо [3, 4]. Однак, підстановка цих апроксимацій у вираз (3) показує, що для випадку $x_1 = x_2$ всі вони приймають значення рівне 0 замість 1, що унеможливлює їх застосування у такому контексті без додаткових модифікацій.

Інший підхід, описаний у [5, 6], полягає у використанні індикаторної функції $1_0(x)$, у цьому випадку:

$$y = C(x_1, x_2) = 1_0(x_1 - x_2) \tag{10}$$

Опираючись на вираз (10) структура перцептрону, що обчислює $C(x_1, x_2)$ на два синаптичні входи, матиме вигляд, рис.1.

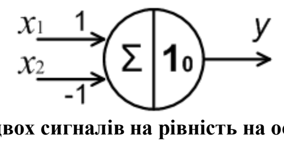


**Рис.1. Перцептрон для порівняння двох сигналів на рівність на основі індикаторної функції**

В такому випадку, замість виразів (4) та (6), відповідно отримаємо вирази (11) та (12):

$$C(\bar{x}, i) = \sum_{j=1}^{n} 1_0(x_j - x_i) \tag{11}$$

$$Q(\bar{x}) = \sum_{i=1}^{n} \frac{1}{C(\bar{x}, i)} = \sum_{i=1}^{n} \frac{1}{\sum_{j=1}^{n} 1_0(x_j - x_i)} \tag{12}$$

Базуючись на виразі (12), можна отримати нейромережу, здатну підраховувати кількість унікальних елементів, структуру якої подано на рис. 2.

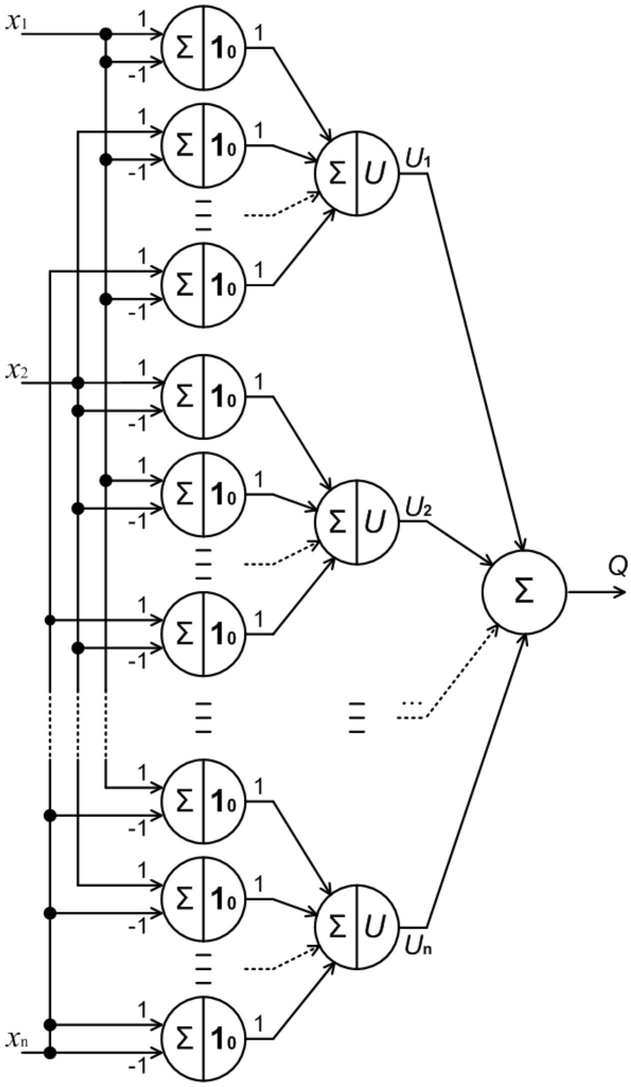


**Рис.2. Структура нейронної мережі, що визначає кількість унікальних станів**

Аналогічно, для виразів (7) та (8), отримаємо вирази (13) та (14):

$$C_>(\bar{x}, i) = \sum_{j=i}^{n} 1_0(x_j - x_i) \tag{13}$$

$$Q(\bar{x}) = \sum_{i=1}^{n} H(1 - C_>(\bar{x}, i)) = \sum_{i=1}^{n} H\left(1 - \sum_{j=1}^{n} 1_0(x_j - x_i)\right) \tag{14}$$

Однак, у виразі (14) все ще використовується функція Хевісайда, провівши її заміну на індикаторну функцію, отримаємо:

$$Q(\bar{x}) = \sum_{i=1}^{n} 1_0(C_>(\bar{x}, i)) = \sum_{i=1}^{n} 1_0\left(\left[\sum_{j=1}^{n} 1_0(x_j - x_i)\right] - 1\right) \tag{15}$$

Доцільно зазначити, що індикаторна функція $1_0(x)$ також непридатна для градієнтного навчання, тому її необхідно замінювати гладкою диференційованою апроксимацією. Як приклади таких апроксимацій можна використати функцію Гауса $f(x) = e^{-kx^2}$, функцію Лоренца $f(x) = \frac{1}{1+kx^2}$, гіперболічний секанс $f(x) = \mathrm{sech}(kx) = \frac{2}{e^{kx}+e^{-kx}}$, функцію Лапласа $f(x) = e^{-k|x|}$ тощо [7, 8].

Аналіз графіків цих функцій при $k$=10 в околі $x$=0, як показано на рис. 3, показує, що форму індикаторної функції найточніше відтворює гіперболічний секанс і функція Лапласа. Оскільки остання містить модуль і, відповідно,немає похідної в точці 0, доцільно обрати саме гіперболічний секанс як основну апроксимацію у подальших розрахунках.

Використавши sech($kx$) у виразах (12) та (15), отримаємо апроксимації $Q(\bar{x})$ придатні для градієнтного навчання, а саме:

$$Q(\bar{x}) \approx \sum_{i=1}^{n} \frac{1}{\sum_{j=1}^{n} \mathrm{sech}(k(x_j - x_i))} \tag{16}$$

$$Q(\bar{x}) \approx \sum_{i=1}^{n} \mathrm{sech}\left(k\left(\left[\sum_{j=1}^{n} \mathrm{sech}(x_j - x_i)\right] - 1\right)\right) \tag{17}$$

Таким чином, отримано гладку апроксимацію дискретної характеристики $Q(\bar{x})$, яка підраховує кількість унікальних станів на входах, що придатна для градієнтного спуску. На наступному етапі необхідно реалізувати перетворення $Q(\bar{x})$ у вихідний сигнал (реакцію) перцептрона за допомогою функції активації.

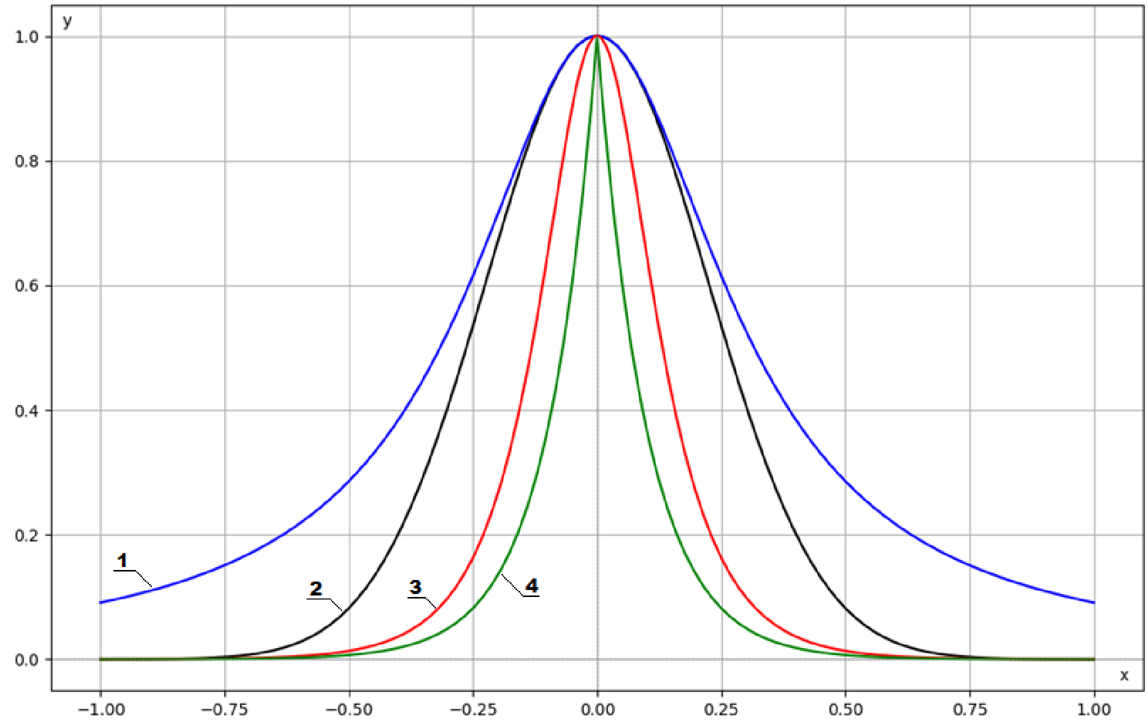


**Рис.3. Функції, які апроксимують індикаторну функцію $1_0(x)$: 1 - Лоренца, 2 – Гауса 3 - гіперболічний секанс, 4 - Лапласа**

Перед вибором активаційних функцій слід розглянути поведінку функції $Q(\bar{x})$ [5,6], що визначає кількість унікальних станів. Така функція є дискретною та приймає цілі значення на проміжку [1; $n$], де $n$ – кількість входів перцептрону. На відміну від лінійної суми, яка зростає із зростанням компонент (та корелюється із "силою сигналу"), $Q(\bar{x})$ може поводитися нелінійно. Тобто однакова зміна вагових коефіцієнтів $W$ може як збільшити так і зменшити кількість унікальних станів або залишити кількість цих станів без змін. Зокрема, якщо кілька вхідних сигналів (координат) $x_i$ стали рівними після операції зміщення (додавання $w_i$) то загальна кількість унікальних станів зменшилася, і навпаки.

Щоб ефективно маніпулювати вихідними сигналами, перцептрон повинен мати можливість адекватно змінювати коефіцієнти зміщення $W$. Однак результат маленької зміни кожного $w_i$ не завжди плавно змінює $Q$. Часто при невеликих зміщеннях вихід перцептрону залишається незмінним, доки певні компоненти не співпадуть/розійдуться. Тобто функція агрегації змінюється стрибкоподібно, при чому в багатьох точках залишається сталою, що ускладнює або й унеможливлює використання методів градієнтного навчання.

Таким чином, якщо використовувати для навчання підходи, що ґрунтуються на методах зворотного поширення помилки, то часто виникатиме ситуація з постійними нульовими "градієнтами", оскільки майже всюди похідна за дискретним зсувом буде нуль, а в точках перестановок буде розрив.

Отже, запропонована структура в [1] дискретного перцептрону, на поточному етапі досліджень, є перспективною для дискретної, зокрема бінарної, логіки чи комбінаторних задач, а не для класичного безперервного навчання. Якщо ж навчання реалізувати не градієнтними методами (перебором, генетичними алгоритмами чи іншими еволюційними/комбінаторними методами), то подібні активації можуть бути актуальні і для інших прикладних задач.

В ході досліджень розглядалися такі функції активації (18, 20, 22, 24).

$$f(x, A) = \begin{cases} 0, & x < A \\ 1, & x \geq A \end{cases} \tag{18}$$

фактично це загальновідома функція Хевісайда, що ділить множину значень на дві підмножини, які дають значення 0 та 1, вона стрибкоподібно змінюється на пороговому значенні $A$ і може бути записана так:

$$f(x, A) = H(x - A) \tag{19}$$

Порогова функція у вихідному шарі нейронної мереж для задач двокласової класифікації така:

$$f(x, A) = \begin{cases} 0, & x < A \wedge A \geq 0 \vee x \geq -A \wedge A < 0 \\ 1, & x \geq A \wedge A \geq 0 \vee x < -A \wedge A < 0 \end{cases} \tag{20}$$

Ця функція є модифікованою пороговою функцією, що враховує знак параметру $A$. Таким чином, послуговуючись рівністю (17) її можна представити так:

$$f(x, A) = \begin{cases} H(-x - A), & x < A \\ H(x - A), & x \geq A \end{cases} \tag{21}$$

В обох режимах ($A \geq 0$ та $A < 0$) така функція залишається бінарною і при $A \geq 0$ збігається з класичною пороговою функцією (17). Для випадку *A*<0 отримуємо інвертовану порогову функцію.

Для типових реалізацій перцептронів така функція активації не використовується, оскільки не надає жодних переваг у порівнянні із класичною пороговою функцією. Однак, у разі застосування порогової функції Хевісайда *H(x – A)* до величини *x*, яка відповідає кількості унікальних значень вхідних дискретних сигналів, виникає природне обмеження на діапазон *A*, оскільки мова йде про мінімально допустиму кількість різних станів у векторі сигналу (синаптичних входах). Слід зауважити, що таке обмеження може бути надто звуженим для задач, у яких важливо, визначати не тільки "чи вистачає унікальних значень?", але й "чи не є їх забагато?". У подібних випадках корисно мати можливість задати від'ємне *A*, щоб інтерпретувати від'ємні пороги як "верхню межу" кількості унікальних значень або реалізувати інверсну логіку активації.

Таким чином, модифікована порогова функція, яка коректно реагує і на додатний, і на від'ємний поріг *A*, істотно розширює сферу її застосувань. Додавання підтримки від'ємних значень *A* дає змогу: керувати поведінкою перцептрона у випадках, коли слід вимикати активацію при надмірно великій кількості унікальних значень та забезпечувати більшу гнучкість у побудові логічних умов. Зокрема, поріг *A>0* відповідає умові "активуватися, якщо унікальних елементів не менше *A*", тоді як поріг *A<0* задає протилежний сенс – "активуватися, якщо унікальних елементів не більше –*A*". Таким чином, одна й та ж функція активації може універсально задовольняти обидва типи порогових перевірок.

З погляду практичної реалізації, таке узагальнення не змінює основних дискретних властивостей функції Хевісайда, проте надає додаткові можливості для логічної інтерпретації виходу. Зокрема, в контексті нетрадиційних перцептронних моделей, що працюють із визначенням унікальних значень, від'ємні *A* перетворюють бінарне рішення "більше/менше за поріг" на більш універсальну умову "виходу кількості унікальних ознак за межі діапазону". Такий підхід суттєво спрощує побудову комбінаторних чи логічних правил без упровадження додаткових елементів у модель.

$$f(x,A) = \begin{cases} 0, & x \neq A \\ 1, & x = A \end{cases} \quad (22)$$

Функція (22) це індикаторна функція рівності (23), яка перевіряє чи вхід *x* точно рівний заданому параметру *A*:

$$f(x,A) = 1_0(x - A) \quad (23)$$

Подана функція не використовується як активаційна у класичних перцептронах через вузький діапазон активації (єдине значення):

$$f(x,A) = \begin{cases} 0, & x \neq A \wedge A \geq 0 \vee x = -A \wedge A < 0 \\ 1, & x = A \wedge A \geq 0 \vee x \neq -A \wedge A < 0 \end{cases} \quad (24)$$

Ця функція також є модифікованою індикаторною функцією, що поводиться по різному залежно від знаку параметру *A*. В результаті рівність (24) можна представити так:

$$f(x,A) = \begin{cases} \&\ \ 1_0(x - A), & x \geq 0 \\ 1 - 1_0(x + A), & x < A \end{cases} \quad (25)$$

Як і попередня, подана модифікована індикаторна функція теж не використовується як активаційна у класичних перцептронах через її занадто широкий діапазон активації (виключене тільки єдине значення).

Основним обмеженням розглянутих функцій є їх негладкість, що вимагає спеціальних методів навчання (зокрема, методи нескінченно малих збурень, евристики типу прямопрохідного оцінювача або еволюційні алгоритми), оскільки стандартний підхід, що ґрунтується на зворотному поширенні помилки не оновлює ваги при нульових похідних.

Однак, для випадку запропонованого дискретного перцептрону, такі функції активації мають суттєву перевагу у простоті та інтерпретованості, що фактично дозволяє розглядати перцептрон як окремий поліфункціональний логічний компонент. Мається на увазі, що емулюючи будь-які логічні функції запропонований дискретний перцептрон, що використовує одну з вище описаних (18, 20, 22, 24) функцій активації, структурно залишається незмінним.

Така особливість дозволяє розглянути питання створення програмованих логічних інтегральних схем (ПЛІС) на базі ідентичних компонентів, які в залежності від потребу будуть реалізовувати будь-яку функцію бінарної логіки.

З поміж двох сценаріїв використання дискретного перцептрона: 1 - як програмованого логічного елементу, де негладкі активації зручні та цілком виправдані; 2 - як частини нейронної мережі, що навчається градієнтно - другий знову зумовлює появу проблеми відсутності похідної [7, 8]. Тобто, якщо вийти за межі чисто логічної інтерпретації і перейти до оптимізації ваг методом зворотного поширення помилки, негладкі функції (19, 21, 23, 25) стають непридатні. В такій ситуації доцільно зосередитись на пороговій функції Хевісайда (19) та індикаторній функції (23) і дослідити, як стандартно її замінити гладкими неперервними апроксимаціями, зокрема такими як логістична сигмоїда, арктангенс тощо.

Оскільки функція (18) – це функція Хевісайда (19), то відомо багато варіантів її емуляції за допомогою неперервних функцій, зокрема логістичної функції (26), арктангенса (27) [3, 4]:

$$H(x) \approx \sigma_k(x) = \frac{1}{1+e^{-kx}} = \frac{1}{2}\left(1 + tanh\left(\frac{kx}{2}\right)\right) \quad (26)$$

$$H(x) \approx \frac{1}{2} + \frac{1}{\pi} arctan(kx) \tag{27}$$

де $k$ – коефіцієнт крутизни нахилу, збільшення якого дозволяє отримати кращу апроксимацію, проте за рахунок збільшення значень похідних утруднюємо застосування методу градієнтного спуску для пошуку оптимумів.

Для емуляції модифікованої функції Хевісайда неперервною необхідно позбутися умови. Для цього скористаємося функцією знаку (28) та тим, що $A$=0 не є цікавою у нашому випадку (так як кількість унікальних станів завжди більша 0), підставивши її у (21), отримаємо (29):

$$sign(x) = \begin{cases} -1, & x < 0 \\ 0 & x = 0 \\ 1, & x > 0 \end{cases} \tag{28}$$

$$f(x, A) = H(sign(A)x - A) \tag{29}$$

Скориставшись тим, що $sign(x) \approx tanh(kx)$, отримаємо:

$$f(x, A) \approx \frac{1}{2}\left(1 + tanh\left(\frac{k(x \cdot tanh(kA) - A)}{2}\right)\right) \tag{30}$$

Індикаторну функцію (23), як було продемонстровано вище, доцільно емулювати за допомогою гіперболічного секанса, в такому випадку:

$$f(x, A) = 1_0(x - A) \approx \text{sech}(k(x - A)) \tag{31}$$

Для емуляції модифікованої індикаторної функції неперервною необхідно позбутися умови. Скориставшись знаковою функцією (28) та виразом (25) для модифікованої індикаторної функції, отримаємо:

$$f(x, A) = \begin{cases} 1_0(x - sign(A)A), & A \geq 0 \\ 1 - 1_0(x + sign(A)A), & A < 0 \end{cases} \tag{32}$$

Оскільки $A$=0 не є цікавою у нашому випадку, так як кількість унікальних станів завжди більша 0, тоді отримаємо:

$$f(x, A) \approx \frac{1}{2} + sign(A)\left(1_0(x - sign(A)A - \frac{1}{2}\right) \tag{33}$$

Скориставшись тим, що $sign(x) \approx tanh(kx)$, отримаємо:

$$f(x, A) = \frac{1}{2} + tanh(kA)\left(\text{sech}(k(x - tanh(kA)A)) - \frac{1}{2}\right) \tag{34}$$

В результаті, отримані функції (26), (30), (31) та (34) дозволяють скористатися методом градієнтного спуску за рахунок емуляції дискретних функцій (18), (20), (22) та (24) неперервними.

**Висновок**

За результатами проведених досліджень запропоновано новий підхід до структурної оргвнізації перцептрона, що базується на використанні ймовірнісних оцінок дискретних сигналів і апроксимації активаційних функцій. Проведені дослідження показали, що застосування імовірнісних оцінок дозволяє значно підвищити ефективність обробки сигналів, зменшити обчислювальні витрати а також дещо прискорити процес навчання нейронних мереж. Використання гладких диференційованих апроксимацій замість класичних активаційних функцій дозволяє подолати проблеми, пов'язані з недиференційованістю, і забезпечити ефективне навчання мережі. Запропоновані рішення мають перспективи для практичної імплементації, оскільки дозволяють уніфікувати, а також суттєво знизити схемну, алгоритмічну та програмну складність цифрових компонентів комп'ютерних систем. Це, своєю чергою, відкриває нові можливості для використання перцептронних структур для створення програмованих логічних інтегральних схем (ПЛІС) на базі ідентичних компонентів, що реалізовувати будь-яку функцію бінарної логіки на ідентичних схемних рішеннях.

Таким чином, отримані результати підтверджують перспективність запропонованого підходу і вказують на можливість подальших досліджень для удосконалення методів опрацювання дискретних сигналів та підвищення ефективності штучних нейронних мереж.

**Література**


1. Патент України UA № 126753, МПК G06N3/08, G06N20/00, 2023 р., Бюл. № 4
2. Tian, X. (2017). Transfer Entropy Estimation via Copula. . https://doi.org/10.2991/MECS-17.2017.159.
3. Han, X., & Kloeden, P. (2020). Sigmoidal approximations of Heaviside functions in neural lattice models. *Journal of Differential Equations*, 268, 5283-5300. https://doi.org/10.1016/j.jde.2019.11.010.
4. Wang, X., Yang, M., & Kloeden, P. (2020). Sigmoidal approximations of a delay neural lattice model with Heaviside functions. *Communications on Pure & Applied Analysis*. https://doi.org/10.3934/cpaa.2020104.
5. Xing, D., Yan, T., Han, Z., & Liu, J. (2023). Supplier evaluation model based on entropy-TOPSIS and 0-1 programming algorithm. , 12645, 126453J - 126453J-6. https://doi.org/10.1117/12.2680793.
6. Tu, Q., Xu, T., Fang, T., Wang, W., Jiang, J., & Zhu, P. (2020). An Entropy evaluation method of hierarchical clustering. *2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, 226-230. https://doi.org/10.1109/DCABES50732.2020.00066.
7. Berahas, A., Cao, L., Choromanski, K., & Scheinberg, K. (2019). A Theoretical and Empirical

Comparison of Gradient Approximations in Derivative-Free Optimization. *Foundations of Computational Mathematics*, 22, 507 - 560. https://doi.org/10.1007/s10208-021-09513-z.

8. Chaudhuri, S., & Solar-Lezama, A. (2010). Smooth interpretation. , 279-291. https://doi.org/10.1145/1806596.1806629.

**References**

1. Патент України UA № 126753, МПК G06N3/08, G06N20/00, 2023 р., Бюл. № 4
2. Tian, X. (2017). Transfer Entropy Estimation via Copula. . https://doi.org/10.2991/MECS-17.2017.159.
3. Han, X., & Kloeden, P. (2020). Sigmoidal approximations of Heaviside functions in neural lattice models. *Journal of Differential Equations*, 268, 5283-5300. https://doi.org/10.1016/j.jde.2019.11.010.
4. Wang, X., Yang, M., & Kloeden, P. (2020). Sigmoidal approximations of a delay neural lattice model with Heaviside functions. *Communications on Pure & Applied Analysis*. https://doi.org/10.3934/cpaa.2020104.
5. Xing, D., Yan, T., Han, Z., & Liu, J. (2023). Supplier evaluation model based on entropy-TOPSIS and 0-1 programming algorithm. , 12645, 126453J - 126453J-6. https://doi.org/10.1117/12.2680793.
6. Tu, Q., Xu, T., Fang, T., Wang, W., Jiang, J., & Zhu, P. (2020). An Entropy evaluation method of hierarchical clustering. *2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, 226-230. https://doi.org/10.1109/DCABES50732.2020.00066.
7. Berahas, A., Cao, L., Choromanski, K., & Scheinberg, K. (2019). A Theoretical and Empirical Comparison of Gradient Approximations in Derivative-Free Optimization. *Foundations of Computational Mathematics*, 22, 507 - 560. https://doi.org/10.1007/s10208-021-09513-z.
8. Chaudhuri, S., & Solar-Lezama, A. (2010). Smooth interpretation. , 279-291. https://doi.org/10.1145/1806596.1806629.