

РОЗДІЛ 2

НЕЙРОННІ МЕРЕЖІ ПРЯМОГО ПОШИРЕННЯ

Нейронні мережі прямого поширення сигналу – це нейронні мережі, де нейрони розділено на групи – *шари*, а зв'язки між нейронами організовано таким чином, що перший (*вхідний шар*) сприймає зовнішній сигнал, обробляє його і передає значення з виходів своїх нейронів наступному шару, при цьому всередині кожного шару зв'язки між нейронами відсутні, а на входи нейронів *схованих шарів* (усіх інших шарів, крім першого), поступають значення з відповідних виходів нейронів попереднього шару. Останній шар мережі називають *вихідним*. Значення з виходів його нейронів утворюють загальний вихідний сигнал НМ.

2.1 Одношаровий перцептрон. Метод Уїдроу-Хоффа

Одношаровий перцептрон є одним з найпростіших варіантів НМ і містить лише один нейрон (див. рис. 1.2). Будучи самостійною моделлю НМ з одного боку, одношаровий перцептрон (формальний нейрон) є основним конструкційним елементом для більшості моделей НМ, з іншого боку.

Одношаровий перцептрон використовує як дискримінантну функцію зважену суму, а як активаційну – порогову або сигмоїдну, рідше – лінійну функцію. У залежності від типу функції активації розрізняють *дискретні перцептрони*, що використовують порогову функцію активації, і *дійсні перцептрони*, що використовують дійсні функції активації, наприклад, сигмоїдну функцію.

Метод навчання одношарового дискретного перцептрона має наступний вигляд.

Крок 1. Вагам w_i , $i = 1, 2, \dots, N$, та порогу w_0 присвоюються випадкові значення.

Крок 2. Пред'являються черговий екземпляр $x^p = \{x_{1p}, \dots, x_{lp}\}$ з навчальної множини, $p = 1, 2, \dots, m$, де m – кількість екземплярів у навчальній вибірці, і бажаний вихід y^{s*} .

Крок 3. Обчислюється реальне значення на виході перцептрона $y = \psi(\varphi(w, x))$, де $w = \{w_0, w_1, w_2, \dots, w_L\}$ – набір ваг, що утворюють пам'ять нейрона.

Крок 4. Коригуються ваги перцептрона:

$$w_i = w_i + \alpha (y^{s*} - y^s) x_{ip},$$

$$i = 0, 1, \dots, L, x_0 = 1,$$

де α – позитивний коригувальний приріст (крок навчання).

Крок 5. Якщо досягнуто збіжність, то процедура корекції ваг закінчується, інакше – перехід до кроку 2.

Відповідно до даного методу спочатку здійснюється ініціалізація параметрів перцептрона випадковими значеннями. Потім по черзі пред'являються образи з відомою класифікацією, вибрані з навчальної множини, і коригуються ваги відповідно до формул кроків 3 та 4.

Величина корекції визначається позитивним коригувальним приростом, конкретне значення якого вибирається досить великим, щоб швидше проводилася корекція ваг, і в той же час досить малим, щоб не допустити надмірного зростання значень ваг.

Процедура навчання триває до тих пір, поки не буде досягнута збіжність, тобто поки не будуть отримані ваги, що забезпечують правильну класифікацію для всіх образів з навчальної множини.

У тому випадку, коли навчальні вибірки розділити гіперплощиною неможливо, для навчання перцептрона можна використовувати метод *Уїдроу-Хоффа*, здатний мінімізувати середньоквадратичну помилку між бажаними і реальними виходами мережі для навчальних даних. Цей метод також можна застосовувати для *навчання одношарового дійсного перцептрона*. Метод Уїдроу-Хоффа можна записати в тому ж вигляді, що й вищеописаний метод, припускаючи, що у вузлах перцептрона нелінійні елементи відсутні, а коригувальний приріст в процесі ітерацій поступово зменшується до нуля.

Якщо для вирішення задачі розпізнавання образів використовується дискретний перцептрон, вирішальне правило відносить вхідний образ до класу K^1 , якщо вихід перцептрона дорівнює 1, і до класу K^0 – в іншому випадку. У разі, якщо для

вирішення задач розпізнавання образів використовується дійсний перцептрон, вирішальне правило відносить вхідний образ до класу K^1 , якщо вихід мережі більше 0,5; і до класу K^0 – в іншому випадку.

Незважаючи на досить обмежені здібності кожного нейрона окремо, їх об'єднання в мережу дає можливість вирішувати більш складні завдання, невідкладні кожному окремому нейрону.

2.2 Багатощарова нейронна мережа

Багатощарова нейронна мережа (БНМ) прямого поширення (*багатощаровий перцептрон*) складається з формальних нейронів і характеризується наступними параметрами і властивостями: M – кількість шарів мережі, N_μ – кількість нейронів у μ -му шарі, зв'язки між нейронами у шарі відсутні. Виходи нейронів μ -го шару, $\mu=1,2,\dots, M-1$, надходять на входи нейронів тільки наступного $\mu+1$ -го шару. Зовнішній векторний сигнал x надходить на входи нейронів тільки першого шару, виходи нейронів останнього M -го шару утворюють вектор виходів мережі $y^{(M)}$. Схема мережі показана на рис. 2.1.

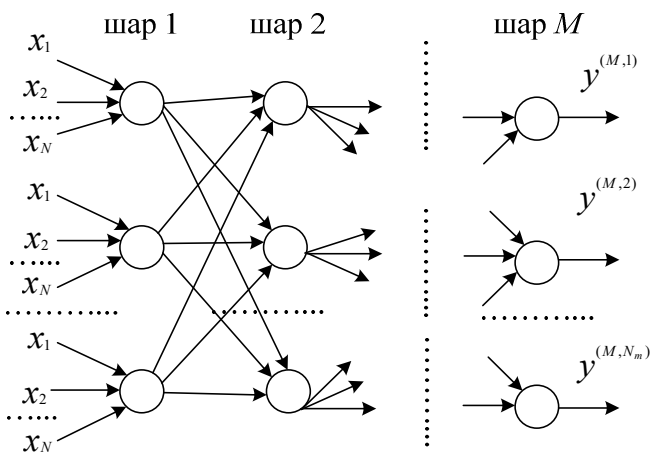


Рисунок 2.1 – Схема БНМ

Кожен i -й нейрон μ -го шару ((μ, i) -й нейрон) перетворює вхідний вектор $x^{(\mu, i)}$ у вихідну скалярну величину $y^{(\mu, i)}$. Це

перетворення складається з двох етапів: спочатку обчислюється дискримінантна функція $\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)})$, яка далі перетворюється на вихідну величину $y^{(\mu,i)} = \psi^{(\mu,i)}(\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)}))$, де $w^{(\mu,i)} = (w_0^{(\mu,i)}, w_1^{(\mu,i)}, \dots, w_{N_\mu}^{(\mu,i)})^T$ – вектор вагових коефіцієнтів нейрона, $x_j^{(\mu,i)}$ – j -та компонента N_μ – вимірного вхідного вектора $x^{(\mu,i)}$.

Якість рішення, одержуваного БНМ, буде істотно залежати від кількості шарів, кількості нейронів у кожному шарі і кількості зв'язків між шарами.

Загальна кількість ваг і порогів у БНМ визначається формулою:

$$N_w = \sum_{\mu=1}^M N_\mu (N_{\mu-1} + 1), N_0 = N.$$

Для різних класів задач, що вирішуються за допомогою НМ, запропоновані евристичні оцінки кількості шарів і нейронів.

Для забезпечення узагальнювальних властивостей необхідно, щоб кількість ваг і порогів БНМ $N_w \ll NS$, де N – розмірність вхідного сигналу (кількість ознак), S – кількість екземплярів у навчальній вибірці.

Виходячи для простоти з того, що розмірність виходу БНМ дорівнює одному (тобто мережа має на останньому шарі тільки один нейрон), а кількість шарів може становити два або три (менше двох не дозволить вирішувати нелінійні задачі, а більше трьох шарів непотрібно, бо відомо, що тришарова БНМ здатна апроксимувати будь-яку обчислювану функцію), а також приймаючи $N \approx S$, отримаємо:

– для двохшарової БНМ:

$$N_w = N_1(N+1) + (N_1+1) = N_1(N+2) + 1 \ll NS,$$

$$N_1 \ll \frac{NS-1}{N+2} \Rightarrow N_1 \ll N;$$

– для тришарової БНМ:

$$N_w = N_1(N+1) + N_2(N_1+1) + (N_2+1) = N_1(N+1) + N_2N_1 + 2N_2 + 1 \ll NS.$$

Прийmemo $N_2=0,5N_1$, а $N_1N \approx N_1^2$ тоді для тришарової БНМ отримаємо: $1,5N_1N_1 + 2N_1 \ll NN - 1 \Rightarrow N_1 \ll N$.

Відомі й інші підходи. Наприклад, для оцінки кількості нейронів у прихованих шарах однорідних двошарових БНМ з сігмоїдними функціями активації можна використовувати формулу для оцінки необхідного числа синаптичних ваг N_w :

$$\frac{N_M S}{1 + \log_2 S} \leq N_w \leq N_M \left(\frac{S}{N_M} + 1 \right) (N + N_M + 1) + N_M,$$

де N_M – розмірність вихідного сигналу.

Оцінивши необхідну кількість ваг, можна розрахувати N_n – кількість нейронів у прихованих шарах. Наприклад, для двошарової мережі вона складе: $N_n = N_w / (N + N_M)$.

Відомі й інші формули для оцінки, наприклад:

$$2(N + N_n + N_M) \leq S \leq 10(N + N_n + N_M); 0,1S - N - N_M \leq N_n \leq 0,5S - N - N_M.$$

2.3 Навчання БНМ. Метод зворотного поширення помилки

Процес навчання БНМ здійснюється у результаті мінімізації цільової функції – певного критерію якості $E(Q(\varepsilon, s))$, що характеризує інтегральну міру близькості виходів мережі $y^{(M)}$ і вказівок вчителя $y^* = \{y^{s*}\}$, $s = 1, 2, \dots, S$, де s – номер поточного екземпляра навчаючої вибірки, S – кількість екземплярів у навчальній вибірці, $Q(\varepsilon, s)$ – миттєвий критерій якості, що залежить від вектора помилки мережі для i -ї вихідної змінної s -го екземпляра $\varepsilon(s, i, w) = y^{(M, i)} - y_i^s$, $i = 1, 2, \dots, N_M$, де w – множина ваг мережі, $y^{(M, i)}$ та y_i^{s*} – відповідно, розраховане мережею і бажане значення i -ї вихідної змінної для s -го екземпляра навчаючої вибірки.

Як миттєвий критерій якості найбільш часто використовують суму квадратів помилки

$$Q(\varepsilon, s) = \sum_{i=1}^{N_M} \varepsilon(s, i, w)^2$$

або суму модулів помилки для s -го екземпляра вибірки

$$Q(\varepsilon, s) = \sum_{i=1}^{N_M} |\varepsilon(s, i, w)|.$$

Інтегральну міру близькості E визначають за формулою:

$$E = \frac{1}{\nu} \sum_{s=1}^S Q(\varepsilon, s),$$

де ν – деякий коефіцієнт (якщо $\nu=1$, то E визначає сумарну миттєву помилку; якщо $\nu = S$, то E визначає середню миттєву помилку).

Для кожного вхідного вектора x^s з навчальної множини повинен бути визначений вектор бажаних виходів мережі y^{s*} . Якщо навчальна БНМ використовується в якості класифікатора, то зазвичай бажані виходи мають низький рівень (0 або менше 0,1), крім виходу вузла, відповідного класу, до якого належить x ; цей вихід у даному випадку має високий рівень (1 або більше 0,9).

Для початку навчання ваги БНМ заданої структури встановлюють рівними випадковим числам.

Для мінімізації цільової функції навчання вирішують завдання багатомірної нелінійної оптимізації, для чого традиційно використовують *градієнтні методи*. Ці методи носять ітеративний характер, оскільки компоненти градієнта виявляються нелінійними функціями. Градієнтні методи засновані на ітераційній процедурі корекції значень ваг, реалізованої у відповідності з формулою:

$$w_{k+1} = w_k + \alpha_k p(w_k), k = 0, 1, 2, \dots$$

де w_k , w_{k+1} – поточне і нове наближення значень ваг і порогів НМ до оптимального рішення, відповідно; α_k – крок збіжності; $p(w_k)$ – напрям пошуку в багатовимірному просторі ваг. Спосіб визначення $p(w_k)$ та α_k на кожній ітерації залежить від особливостей конкретного методу.

Оскільки градієнтні методи вимагають обчислення похідних цільових функції для визначення напрямку пошуку, для розрахунку частинних похідних цільової функції за вагами мережі використовують *метод зворотного поширення помилки* (error backpropagation method), який був розроблений як

узагальнення методу Уїдроу-Хоффа для БНМ з нелінійними диференційовними функціями активації.

Метод зворотного поширення помилки для настроювання ваг НМ використовує градієнт функції помилки таким чином, щоб мінімізувати помилку на навчальній вибірці. Для цього мережі послідовно надаються вхідні вектори (екземпляри) з навчальної вибірки, і, починаючи з останнього шару, обчислюються *градієнти функції помилки* для кожного нейрона:

$$\frac{\partial E}{\partial w_j^{(\mu,i)}} = \frac{\partial E}{\partial \psi^{(\mu,i)}} \frac{\partial \psi^{(\mu,i)}}{\partial \phi^{(\mu,i)}} \frac{\partial \phi^{(\mu,i)}}{\partial w_j^{(\mu,i)}},$$

де $\frac{\partial \psi^{(\mu,i)}}{\partial \phi^{(\mu,i)}} = \psi'^{(\mu,i)}(\phi^{(\mu,i)})$.

Якщо в якості дискримінантної функції нейронів мережі використовується зважена сума, то $\frac{\partial \phi^{(\mu,i)}}{\partial w_j^{(\mu,i)}} = x_j^{(\mu,i)} = \psi^{(\mu-1,j)}$.

Для нейронів вихідного шару: $\frac{\partial E}{\partial \psi^{(M,i)}} = y_i^s - \psi^{(M,i)}$,

для нейронів інших шарів:

$$\frac{\partial E}{\partial \psi^{(\mu,i)}} = \sum_j \frac{\partial E}{\partial \psi^{(\mu+1,j)}} \frac{\partial \psi^{(\mu+1,j)}}{\partial \phi^{(\mu+1,j)}} \frac{\partial \phi^{(\mu+1,j)}}{\partial \psi^{(\mu,i)}},$$

де $\frac{\partial \phi^{(\mu+1,j)}}{\partial \psi^{(\mu,i)}} = w_j^{(\mu,i)}$.

У методі зворотного поширення помилки ваги і пороги змінюються у напрямку, протилежному градієнту. Тому базовий метод зворотного поширення часто називають *методом градієнтного спуску* (gradient descent). Базовий метод зворотного поширення помилки складається з таких кроків.

Крок 1. Задати навчальну вибірку екземплярів x і зіставлених ним значень цільового параметру y^* . Ініціалізувати всі ваги і пороги мережі. Задати значення прийнятного рівня помилки ξ . Встановити покажчик поточного екземпляра вибірки $s = 1$.

Крок 2. Подати на вхід БНМ s -й екземпляр з навчальної вибірки $x^s = \{x_{ij}^s\}$, а на вихід – зіставлений йому бажаний вихідний вектор $y^{s*} = \{y_{ij}^{s*}\}$, $s = 1, 2, \dots, S$; $i = 1, 2, \dots, N$; $j = 1, 2, \dots, N_M$.

Крок 3. Рухаючись від першого шару до останнього, розрахувати значення на виходах нейронів і фактичний вихід мережі $y^{(M)} = \{y^{(M, i)}\}$, $i = 1, 2, \dots, N_M$, при подачі на вхід мережі екземпляра x^s .

Крок 4. Розрахувати помилку мережі E . Якщо $E > \xi$ (умова закінчення навчання для s -го екземпляра не виконується), тоді перейти на крок 5; в іншому випадку – перейти на крок 6.

Крок 5. Рухаючись від вихідного шару до першого шару мережі провести настроювання ваг і порогів за формулою $w_{k+1} = w_k + \alpha_k p(w_k)$, визначивши напрямок пошуку в багатовимірному просторі ваг за формулою:

$$p(w_{j,k}^{(\mu,i)}) = -g_{j,k}^{(\mu,i)},$$

де $g_{j,k}^{(\mu,i)} = \frac{\partial E}{\partial w_j^{(\mu,i)}}$ – градієнт помилки за j -ю вагою i -го нейрона μ -го шару БНМ на k -й ітерації навчання, який визначається за формулою:

$$g_{j,k}^{(M,i)} = \psi_k^{\prime(M,i)} \left(\varphi_k^{(M,i)} \right) \left(y_i^s - \psi_k^{(M,i)} \left(\varphi_k^{(M,i)} \right) \right) \psi_k^{(M-1,j)} \left(\varphi_k^{(M-1,j)} \right),$$

а для нейронів інших шарів за рекурентною формулою:

$$g_{p,k}^{(\mu,i)} = \psi_k^{\prime(\mu,i)} \left(\varphi_k^{(\mu,i)} \right) \left(\sum_{j=1}^{N_{\mu+1}} g_{i,k}^{(\mu+1,j)} w_{i,k}^{(\mu+1,j)} \right) x_{p,k}^{(\mu,i)},$$

$$\mu = M-1, \dots, 1; p = 0, \dots, N_{\mu-1}.$$

Якщо у мережі використовуються нейрони з сигмоїдною функцією активації, то для них:

$$\psi^{(\mu,i)} = \left(\frac{1}{1 + e^{-\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)})}} \right)' = \frac{e^{-\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)})}}{\left(1 + e^{-\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)})}\right)^2} =$$

$$= \frac{1}{1 + e^{-\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)})}} \left(1 - \frac{1}{1 + e^{-\varphi^{(\mu,i)}(w^{(\mu,i)}, x^{(\mu,i)})}} \right) = \psi^{(\mu,i)}(1 - \psi^{(\mu,i)}).$$

Перейти на крок 4.

Крок 6. Встановити: $s = s + 1$. Якщо $s > S$, то перейти на крок 7, у протилежному випадку – перейти на крок 2.

Крок 7. Визначити помилку мережі E . Якщо вона є прийнятною ($E \leq \xi$), то завершити пошук і видати як результат роботи значення ваг і порогів w , у протилежному випадку – встановити $s = 1$ та перейти до кроку 2.

Як додаткові *критерії завершення пошуку* у методі зворотного поширення помилки можуть використовуватися досягнення ліміту часу або ліміту кількості циклів корекції ваг (*epoch навчання*) або досягнення заданого мінімального значення градієнта цільової функції.

Стандартний метод зворотного поширення часто збігається дуже повільно, рухаючись вздовж плоских ділянок поверхні помилки. Тому на практиці його часто використовують лише для розрахунку приватних похідних цільової функції помилки за вагами НМ, а пошук в просторі ваг НМ роблять на основі більш швидких градієнтних методів.

Основним недоліком традиційно використовуваних градієнтних методів навчання БНМ є обумовлена ітераційною корекцією ваг низька швидкість збіжності навчання, яка серйозно обмежує практичне застосування нейронного керування. Іншим не менш важливим недоліком даних методів є невизначеність у виборі початкової точки пошуку, параметрів архітектури та топології мережі.

2.4 Радіально-базисні нейромережі

Радіально-базисні нейромережі (РБНМ – Radial Basis Function Net, RBFN) були запропоновані для апроксимації функцій багатьох змінних. За допомогою радіально-базисних функцій можна як завгодно точно апроксимувати задану функцію. Як і багат шаровий

персептрон, радіально-базисна мережа є універсальним апроксиматором.

Математичну основу РБНМ складає метод потенційних функцій, розроблений М. А. Айзерманом, Е. М. Браверманом і Л. І. Розоноєром, що дозволяє подати певну функцію $y(x)$ у вигляді суперпозиції потенційних або базисних функцій $f_i(x)$:

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

де $a_i = (a_1, a_2, \dots, a_N)^T$ – вектор підлягаючих визначенню параметрів; $f(x) = (f_1(x), f_2(x), \dots, f_N(x))^T$ – вектор базисних функцій.

В РБНМ в якості *базисних функцій* обираються деякі функції відстані між векторами $f_i(x) = f(\|x - c_i\|)$.

Вектори c_i називають *центрами базисних функцій*. Функції $f(x)$ вибираються невід’ємними і зростаючими при збільшенні $\|x - c_i\|$. В якості міри близькості векторів x та c_i вибираються, як правило, такі метрики:

– *евклідова метрика*: $\|x - c_i\| = \sqrt{\sum_{j=1}^N (x_j - c_{ij})^2}$;

– *манхеттенська метрика*: $\|x - c_i\| = \sum_{j=1}^N |x_j - c_{ij}|$

де $|x_j - c_{ij}| = (x_j - c_{ij}) \text{sign}(x_j - c_{ij})$, $\text{sign}(x_i - c_{ij}) = \begin{cases} 1, (x_i - c_{ij}) > 0; \\ 0, (x_i - c_{ij}) = 0; \\ -1, (x_i - c_{ij}) < 0. \end{cases}$

Структурно РБНМ є окремим випадком двошарової БНМ, нейрони першого шару якої здійснюють локалізовану реакцію на вхідний стимул і використовують як дискримінантну функцію Евклідову відстань, а як функцію активації – радіально-базисну

(ядерну) функцію Гауса, а нейрони другого шару формують зважену лінійну комбінацію з радіально-базисних функцій, обчислених нейронами першого шару (використовують як дискримінантну функцію зважену суму, а як функцію активації – лінійну функцію). Нейрони другого шару РБНМ відповідають вихідним класам, у той час як нейрони першого шару відповідають кластерам, на які розбивається вхідний простір ознак. Кількість кластерів заздалегідь, як правило, невідома. На практиці її визначають на основі методів кластер-аналізу або задають рівним двом і далі нарощують, навчаючи щоразу мережу заново, поки помилка мережі не досягне прийнятного значення.

2.5 Навчання радіально-базисних неймереж

РБНМ характеризується трьома типами параметрів:

– лінійні вагові параметри вихідного шару (входять в опис мережі лінійно);

– центри кластерів – нелінійні (входять в опис нелінійно) параметри прихованого шару;

– відхилення (радіуси базисних функцій) – нелінійні параметри прихованого шару.

Навчання РБНМ, що складається у визначенні цих параметрів, може зводиться до одного з таких варіантів:

1) задаються центри і відхилення, а обчислюються тільки ваги вихідного шару;

2) визначаються шляхом самонавчання (за допомогою методів кластеризації) центри і відхилення, а для корекції ваг вихідного шару використовується навчання з учителем. Тобто методом кластеризації формується фіксована множина центрів кластерів. Потім мінімізацією помилки E отримують асоціації центрів кластерів з виходом;

3) визначаються всі параметри мережі за допомогою навчання з учителем, наприклад за допомогою методу зворотного поширення помилки.

Перші два варіанти застосовуються в мережах, що використовують базисні функції з жорстко заданим радіусом (відхиленням). Третій варіант, будучи найбільш складним і

трудомістким в реалізації, передбачає використання будь-яких базисних функцій.



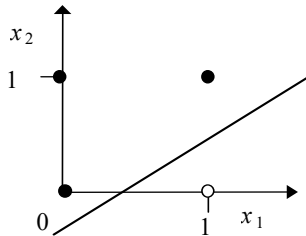
2.6 Приклади виконання завдань

Приклад 1. Чи можна навчити дискретний одношаровий перцептрон обчислювати значення функції: $y = \text{not}(x_1 \text{ and not}(x_2))$?

Спочатку обчислимо таблицю істинності для функції y .

x_1	x_2	$\text{not } x_2$	$x_1 \text{ and not } (x_2)$	y
0	0	1	0	1
0	1	0	0	1
1	0	1	1	0
1	1	0	0	1

Зобразимо графічно функцію y .



На рисунку закрашені круги відповідають значенням $y=1$, а білий круг – значенню $y=0$. Як видно з рисунку, білий круг можна відділити однією гіперплощиною (у проекції – прямою лінією) від чорних кругів, тобто задача є лінійно роздільною.

Тому одношаровий дискретний перцептрон можна навчити безпомилково обчислювати значення цієї функції.

Приклад 2. Яке буде значення на виході y одношарового перцептрона із пороговою функцією активації, дискримінантною функцією зважена сума та двома зовнішніми входами при поданні на його входи сигналів $x_1=1$ та $x_2=0,6$ при відомих значеннях ваг входів $w_1=1$ та $w_2=-1$ і порога $w_0=0$?

Одношаровий перцептрон складається усього з одного нейрона. Функціонування штучного нейрона описується формулою: $y = \psi(\varphi(w, x))$.

Спочатку визначимо значення дискримінантної функції:

$$\varphi(w, x) = \sum_{j=1}^N w_j x_j + w_0 = 1 \cdot 1 + (-1) \cdot 0,6 + 0 = 0,4.$$

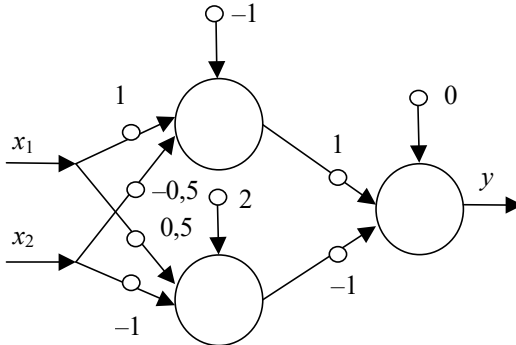
Після чого розрахуємо значення функції активації:

$$\psi(x) = \begin{cases} 1, & x \geq 0; \\ 0, & x < 0; \end{cases}$$

Підставляючи $x=0,4$, отримаємо: $\psi(x) = 1$.

Таким чином, $y = 1$.

Приклад 3. Нехай задано БНМ, структуру і значення параметрів якої зображено на рисунку. Для екземпляра $x=(1 \ 2)^T$ обчислити значення на виході БНМ, виходячи з того, що усі її нейрони використовують дискримінантну функцію зважена сума та функцію активації – пороговоу.



Спочатку для кожного нейрона першого шару визначимо значення на його виході:

– для першого нейрона першого шару:

$$\varphi^{(1,1)}(w^{(1,1)}, x^{(1,1)}) = 1 \cdot 1 + (-0,5) \cdot 2 - 1 = -1; \psi^{(1,1)}(\varphi^{(1,1)}) = 0;$$

– для другого нейрона першого шару:

$$\varphi^{(1,2)}(w^{(1,2)}, x^{(1,2)}) = 0,5 \cdot 1 + (-1) \cdot 2 + 2 = 0,5; \psi^{(1,2)}(\varphi^{(1,2)}) = 1.$$

Потім визначимо значення на виході єдиного нейрона другого шару БНМ:

$$\varphi^{(2,1)}(w^{(2,1)}, x^{(2,1)}) = 1 \cdot 0 + (-1) \cdot 1 + 0 = -1; \psi^{(2,1)}(\varphi^{(2,1)}) = 0.$$

Отже отримаємо значення на виході БНМ: $y = 0$.

? 2.7 Контрольні питання

1. Багатошаровий перцептрон: модель і принципи побудови архітектури.
2. Градієнтні методи навчання багатошарових нейромереж.
3. Дайте порівняльну характеристику відомих Вам методів навчання багатошарових нейромереж.
4. Застосування кластер-аналізу при навчанні радіально-базисних нейромереж.
5. Критерії завершення пошуку у методі зворотного поширення помилки.
6. Математична основа РБНМ.
7. Метод зворотного поширення помилки.
8. Метод Уїдроу-Хоффа.
9. Методи навчання радіально-базисних нейромереж.
10. Моделі та принципи синтезу архітектури радіально-базисних нейромереж.
11. Можливості і властивості одношарових перцептронів.
12. Навчання радіально-базисних нейромереж.
13. Навчання одношарового перцептрона.
14. Недоліки методу зворотного поширення помилки.
15. Опишіть функціонування багатошарового перцептрона.
16. Опишіть функціонування одношарового перцептрона.
17. Опишіть функціонування РБНМ.
18. Переваги і недоліки радіально-базисних мереж.
19. Порівняння моделей та методів навчання нейромереж прямого поширення.
20. Призначення шарів РБНМ.
21. Структура РБНМ.
22. У чому подібність і відмінність радіально-базисних нейромереж і багатошарових нейромереж?
23. У чому полягає відмінність вхідного, вихідного та схованих шарів у БНМ?
24. Чи впливає величина кроку навчання на час навчання одношарового перцептрона, багатошарової мережі? Відповідь обґрунтуйте теоретично та доведіть експериментально.

25. Чи впливає вид функції активації нейрона на тривалість навчання і роботи перцептрона, величину помилки навчання і класифікації (оцінювання) і, якщо впливає, то яким чином?

26. Чи впливає кількість використаних ознак на швидкість навчання перцептрона? Відповідь обґрунтуйте теоретично та доведіть експериментально.

27. Чи впливає обсяг навчальної вибірки на швидкість навчання нейромереж?

28. Чи впливає репрезентативність навчальної вибірки на точність класифікації екземплярів тестової вибірки?

29. Чи впливає репрезентативність тестової вибірки на точність класифікації екземплярів тестової вибірки?

30. Чи впливає репрезентативність тестової вибірки на точність навчання перцептрона по навчальній вибірці?

31. Чи доцільно застосовувати одношаровий перцептрон для класифікації складно (нелінійно) роздільних образів? Обґрунтуйте і доведіть відповідь. Наведіть приклади.

32. Чи доцільно застосовувати радіально-базисних нейромереж для класифікації складно (нелінійно) роздільних образів?

33. Чи завжди збігається метод зворотного поширення помилки для багатошарового перцептрона?

34. Чи завжди збігаються методи навчання одношарового перцептрона?

35. Чи завжди збігаються методи навчання радіально-базисних нейромереж?

36. Чи залежить якість навчання нейромереж від якості та обсягу навчальної вибірки?

37. Чи можливе використання неградієнтних методів багатовимірної безумовної оптимізації для настроювання ваг багатошарових нейромереж, і, якщо можливе, то наскільки це доцільно робити?

38. Що таке крок навчання?

39. Що таке нейронні мережі прямого поширення сигналу?

40. Що таке шар нейронів?

41. Як визначається цільова функція навчання у методі зворотного поширення помилки?

42. Як визначити кількість шарів, нейронів та ваг у БНМ?

43. Як визначити часткову похідну цільової функції за керованою змінною в методі зворотного поширення помилки?

44. Який з градієнтних методів навчання на основі зворотного поширення помилки Ви рекомендували б для застосування, якщо критерієм оптимальності навчання НМ є: а) швидкість навчання, б) простота обчислень, в) універсальність?

45. Які задачі можна вирішувати на основі багат шарових перцептронів, а які не можна? Обґрунтуйте і доведіть відповідь. Приведіть приклади.


46. Які задачі можна вирішувати на основі одно шарових перцептронів, а які не можна?


47. Які задачі можна вирішувати на основі радіально-базисних нейромереж, а які не можна? Обґрунтуйте і доведіть відповідь. Наведіть приклади.


48. Які методи навчання багат шарових нейромереж Вам відомі?

49. Які функції активації нейронів найчастіше використовують у нейромережах прямого поширення?

2.8 Практичні завдання

 *Завдання 1.* Чи може дійсний багат шаровий перцептрон безпомилково або із незначною помилкою навчитися моделювати функцію $y = x_1^2 + 0,5x_1 + x_2$, якщо число його шарів дорівнює: а) 1; б) 2; в) 3? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

 *Завдання 2.* Чи можна навчити дискретний багат шаровий перцептрон безпомилково або із незначною помилкою обчислювати значення функцій: 1) $y = x_1$ and x_2 ; 2) $y = x_1 \text{ xor } x_2$; 3) $y = \text{not}((\text{not } x_1) \text{ and } x_2)$, а дійсний багат шаровий перцептрон – обчислювати значення функцій: 1) $y = 3x_1 - 0,05x_2$; 2) $y = \sin(x_1) + 0,3x_2$; 3) $y = 0,5x_1 + 2x_2 - 2,5(x_1 - x_2) + 5,5$; 4) $y = x_1 / (x_2 \sin(\pi))$? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

 *Завдання 3.* Чи можна навчити дискретний одно шаровий перцептрон обчислювати значення функцій: $y = (x_1 \text{ and } x_2)$, $y = (x_1 \text{ xor } x_2)$, $y = \text{not}((\text{not } x_1) \text{ and } x_2)$, а дійсний одно шаровий перцептрон – обчислювати значення функцій: 1) $y = 3x_1 - 0,05x_2$;

2) $y = \sin(x_1) + 0,3x_2$; 3) $y = 0,5x_1 + 2x_2 - 2,5(x_1 - x_2) + 5,5$? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

✍ Завдання 4. Чи можна навчити радіально-базисну нейромережу обчислювати значення функцій: 1) $y = x_1$ and x_2 ; 2) $y = x_1 \text{ xor } x_2$; 3) $y = \text{not}((\text{not } x_1) \text{ and } x_2)$, а дійсний багатoshаровий перцептрон – обчислювати значення функцій: 1) $y = 3x_1 - 0,05x_2$; 2) $y = \sin(x_1) + 0,3x_2$; 3) $y = 0,5x_1 + 2x_2 - 2,5(x_1 - x_2) + 5,5$; 4) $y = x_1 / (x_2 \sin(\pi))$? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

✍ Завдання 5. Яке буде значення на виході у одношарового перцептрона із двома зовнішніми входами при поданні на його входи сигналів $x_1 = -1$ та $x_2 = 0,3$ при відомих значеннях ваг входів $w_1 = -1$ та $w_2 = 2$ і порога $w_0 = 1$, якщо він має:

а) дискримінантну функцію зважена сума та функцію активації – лінійну?

б) дискримінантну функцію зважена сума та функцію активації – порогову?

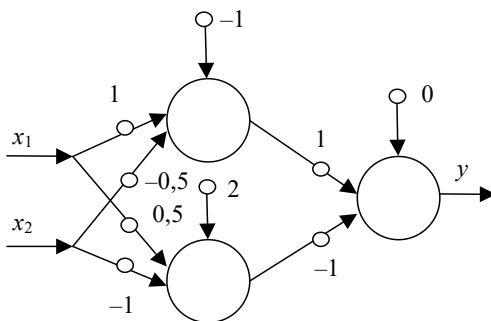
в) дискримінантну функцію зважена сума та функцію активації – сигмоїдну?

г) дискримінантну функцію евклідова відстань та функцію активації – радіально-базисну (Гауса)?


д) дискримінантну функцію евклідова відстань та функцію активації – порогову?

е) дискримінантну функцію евклідова відстань та функцію активації – сигмоїдну?


✍ Завдання 6. Нехай задано БНМ, структуру і значення параметрів якої зображено на рисунку.



Для екземпляра $x = (-1 \ 3)^T$ обчислити значення на виході БНМ, виходячи з того, що усі її нейрони використовують дискримінантну функцію зважена сума та функцію активації – лінійну.

 *Завдання 7.* Підготуйте реферат на одну з таких тем.

- 1.Методи навчання багат шарових нейромереж прямого поширення сигналу.
- 2.Каскадно-кореляційні мережі прямого поширення.
- 3.Апаратні моделі нейромереж прямого поширення.
- 4.Історичні моделі і методи навчання персептронів 60-х років ХХ сторіччя.
- 5.Порівняльна характеристика нейромереж прямого поширення.
- 6.Застосування нейромереж прямого поширення.

 *Завдання 8.* Емуляція та навчання одношарового персептрона.

1. Задати функції активації та дискримінантні функції нейрона, кількість входів одношарового персептрона, навчальну і контрольну вибірки (набори пар значень входів і бажаних виходів персептрона), значення максимальної припустимої помилки навчання, значення параметра збіжності навчання.

2. Написати і відлагодити програму на алгоритмічній мові високого рівня, що моделює формальний нейрон (одношаровий персептрон) і реалізує алгоритми навчання дискретного і дійсного персептронів. Передбачити в програмі відображення на екран і запис у файл на диску поточного стану параметрів персептрона і результатів його роботи: матриці ваг, помилки класифікації (оцінювання), значень на входах і виході персептрона, а також часу навчання і класифікації на основі персептрона.

3. Для вибірки даних здійснити навчання персептрона і зберегти у файлі на диску результати його роботи для навчальної і контрольної вибірок.

4. Кілька разів змінити вид функції активації нейрона і виконати п. 3.

5. Результати виконання п. п. 3–4 занести у таблицю, стовпці якої повинні мати назви: вид функції активації, метод навчання,

час навчання, час класифікації (оцінювання) для навченого персептрона (для навчальної і контрольної вибірок), помилка класифікації (оцінювання) для навчальної і контрольної вибірок. Проаналізувати отримані результати і зробити висновки про те, як впливає вид функції активації формального нейрона на час навчання і класифікації (оцінювання), а також величину помилки навчання і класифікації (оцінювання) одношарового персептрона.

6. На основі розробленої програми, змінюючи значення кроку навчання, для персептрона з заданою функцією активації дослідити, як впливає величина кроку навчання на час навчання. Побудувати графік залежності часу навчання персептрона від величини кроку навчання.



Завдання 9. Емуляція та навчання багатошарового персептрона.

1. Задати число шарів і число нейронів у кожному шарі, вид функції активації нейронів, кількість входів і виходів персептрона, набір пар значень входів і бажаних виходів персептрона, максимально припустиму помилка навчання; крок навчання.

2. Розробити на алгоритмічній мові програмування високого рівня програму, що моделює багатошаровий персептрон та здатна навчати його на основі методу зворотного поширення помилки.

3. Здійснити навчання і моделювання багатошарового персептрона для заданих даних.

4. Кілька разів змінити вид функції активації нейрона і виконати п. 3.

5. Результати моделювання занести у таблицю, стовпці якої повинні мати назви: вид функції активації нейронів, помилка навчання, помилка розпізнавання, час навчання, час класифікації персептрона.

6. Проаналізувати отримані результати і зробити висновки про те, як впливають вид функції активації на час навчання і класифікації (оцінювання), а також величину помилки навчання / класифікації багатошарового персептрона.



Завдання 10. Емуляція та навчання радіально-базисної нейромережі.

1. Задати навчальну і контрольну вибірки даних.

2. Розробити програму на алгоритмічній мові високого рівня, що реалізує навчання і моделювання РБНМ.

3. Здійснити навчання і моделювання РБНМ для навчальних і тестових даних.

4. Результати моделювання занести в таблицю, стовпці якої повинні мати назви: помилка навчання, помилка розпізнавання, час навчання, час класифікації.

5. Проаналізувати отримані результати.