

ЛЕКЦІЯ 6

Тема: Базові типи нейронних мереж

Питання лекції

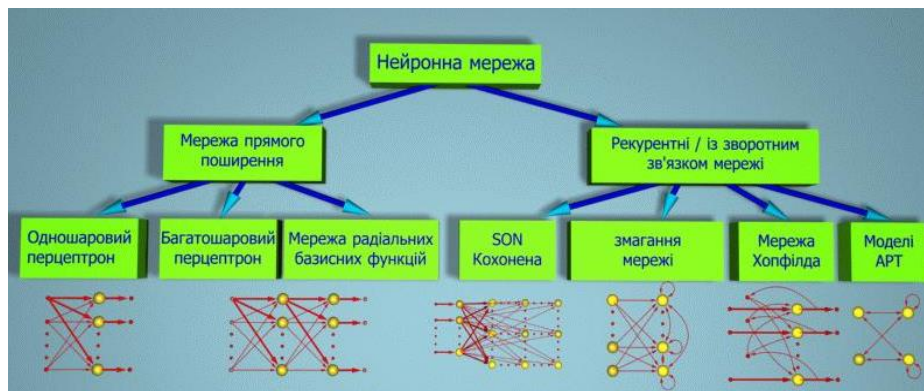
Вступ

1. Мережа Кохонена
2. Мережа квантування навчального вектора (Learning Vector Quantization)
3. Мережа зустрічного поширення (Counter Propagation)
4. Ймовірнісна нейрона мережа
5. Мережа Хопфілда
6. Мережа «Машина Больцмана»
7. Мережа Хемінга
8. Мережа мережної моделі з двонаправленою асоціативною пам'яттю
9. Мережа адаптивної резонансної теорії (ART)

Висновки

Вступ

На попередній лекції ми розглянули основні поняття нейронних мереж та деякі типи мереж прямого поширення, таких як: Перцептрон Розенблата, Нейромережа зворотного поширення похибки (Back Propagation), Мережа Delta Bar Delta.



На сьогоднішній лекції ми розглянемо ще ряд нейронних мереж.

1. Мережа Кохонена

Мережа розроблена Тойво Кохоненом на початку 1980-х рр. і принципово відрізняється від розглянутих вище мереж, оскільки використовує неконтрольоване навчання і навчальна множина складається лише із значень вхідних змінних.

Мережа розпізнає кластери в навчальних даних і розподіляє дані до відповідних кластерів. Якщо в подальшому мережа зустрічається з набором даних, несхожим ні з одним із відомих зразків, вона відносить його до нового кластеру. Якщо в даних містяться мітки класів, то мережа спроможна вирішувати задачі класифікації. Мережі Кохонена можна використовувати і в задачах, де класи є відомими - перевага буде у спроможності мережі виявляти подібність між різноманітними класами.

Мережа Кохонена має лише два прошарки: вхідний і вихідний (рис. 1.) її ще називають **самоорганізовуваною картою**. Елементи карти розташовуються в деякому просторі, як правило, двовимірному. Мережа Кохонена навчається методом послідовних наближень. У процесі навчання на входи подаються дані, але *мережа при цьому підлаштовується не під еталонне значення виходу, а під закономірності у вхідних даних*. Починається навчання з вибраного випадковим чином вихідного розташування центрів.

В процесі послідовної подачі на вхід мережі навчальних прикладів визначається *найбільш схожий нейрон (той, у якого скалярний добуток ваг і поданого на вхід вектора є мінімальним)*. Цей нейрон оголошується переможцем і є центром при підлаштуванні ваг в сусідніх нейронів. Таке правило навчання передбачає "змагальне" (від «змагання») навчання з врахуванням відстані нейронів від "нейрона-переможця".

Навчання при цьому полягає не в мінімізації помилки, а в підлаштуванні ваг (внутрішніх параметрів нейронної мережі) для найбільшого збігу з вхідними даними.

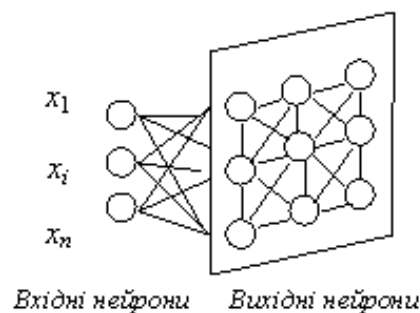


Рис. 1. Мережа Кохонена

Мережа Кохонена навчається методом послідовних наближень. Починаючи з випадковим чином обраного вихідного розташування центрів, алгоритм поступово поліпшується для кластеризації навчальних даних. Основний ітераційний алгоритм Кохонена послідовно проходить ряд епох, на кожній з яких обробляється один приклад з навчальної вибірки. Вхідні сигнали послідовно пред'являються мережі, при цьому бажані вихідні сигнали не визначаються. Після пред'явлення достатнього числа вхідних векторів синаптичні ваги мережі стають здатні визначити кластери.

Ваги організуються так, що топологічно близькі вузли реагують до схожих вхідних сигналів.

В результаті роботи алгоритму центр кластера встановлюється в певній позиції, яка задовольняє кластеризовані приклади, для яких даний нейрон є "переможцем". В результаті навчання мережі необхідно визначити міру сусідства нейронів, тобто окіл нейрона-переможця, який представляє кілька нейронів, що оточують нейрон-переможця.

Для реалізації алгоритму необхідно визначити міру сусідства нейронів (околиця нейрона-переможця). На рис. 2 показані зони топологічного сусідства нейронів на карті ознак у різні моменти часу. $NE_j(t)$ - множина нейронів, що вважаються сусідами нейрона j у момент часу t . Зони сусідства зменшуються з часом.

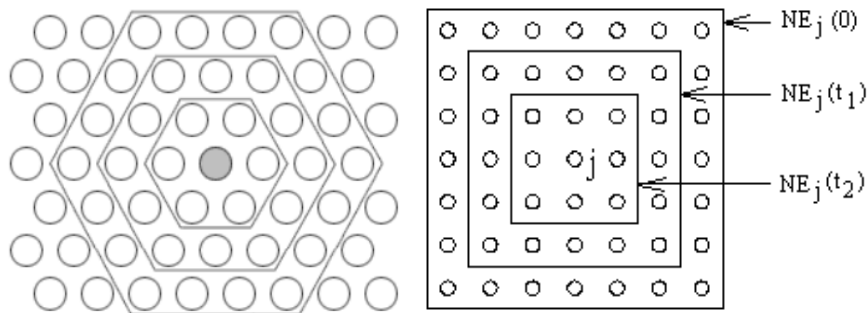


Рис. 2. Зони топологічного сусідства на карті ознак у різні моменти часу

Спочатку до околу належить велике число нейронів, далі її розмір поступово зменшується. Мережа формує топологічну структуру, в якій схожі приклади утворюють групи прикладів, які близько знаходяться на топологічній карті.

Алгоритм функціонування мережі Кохонена:

1. *Ініціалізація мережі.* Ваговим коефіцієнтам мережі надаються малі випадкові значення. Загальне число синаптичних ваг - $M \cdot N$ (див. рис. 1 Початкова зона сусідства показана на рис. 2).

2. *Пред'явлення мережі нового вхідного сигналу.*

3. *Обчислення відстані до всіх нейронів мережі:*

Відстані d_j від вхідного сигналу до кожного нейрона j визначаються за формулою:

$$d_j = \sum_{i=1}^N (x_i(t) \cdot w_{ij}(t))^2$$

де x_i - i -ий елемент вхідного сигналу в момент часу t , $w_{ij}(t)$ - вага зв'язку від i -го елемента вхідного сигналу до нейрона j у момент часу t .

4. *Вибір нейрона з найменшою відстанню:*

Вибирається нейрон-переможець j^* , для якого відстань d_j найменша.

5. Налаштування ваг нейрона j^* і його сусідів:

Робиться налаштування ваг для нейрона j^* і всіх нейронів з його околу NE. Нові значення ваг:

$$w_{ij}(t+1) = w_{ij}(t) + r(t)(x_i(t) - w_{ij}(t))$$

де $r(t)$ - швидкість навчання, що зменшується з часом (додатне число, менше одиниці).

6. Повернення до кроку 2.

В алгоритмі використовується коефіцієнт швидкості навчання, який поступово зменшується, для тонкішої корекції в новій епосі. В результаті позиція центру встановлюється в певній позиції. Вона задовільним чином кластеризує приклади, для яких даний нейрон є переможцем.

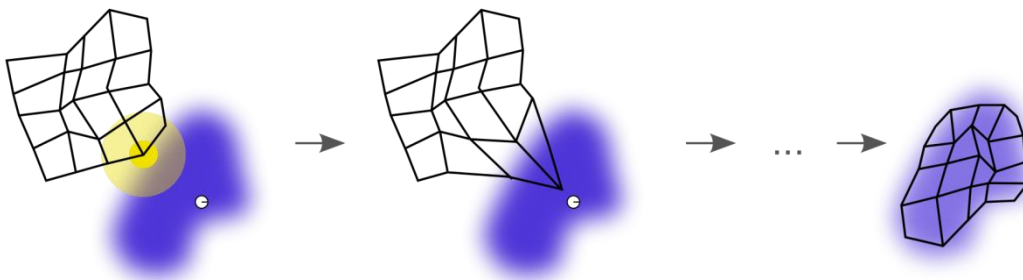
Властивість топологічної впорядкованості досягається в алгоритмі за допомогою використання поняття околу. **Окіл** - це декілька нейронів, що оточують нейрон-переможець. Відповідно до швидкості навчання, розмір околу поступово зменшується, так, що спочатку до нього належить досить велике число нейронів (можливо вся карта), на самих останніх етапах окіл стає нульовим і складається лише з нейрона-переможця.

В алгоритмі навчання корекція застосовується не тільки до нейрона-переможця, але і до всіх нейронів з його поточного околу. В результаті такої зміни околу, початкові доволі великі ділянки мережі мігрують в бік навчальних прикладів. Мережа формує грубу структуру топологічного порядку, при якій схожі приклади активують групи нейронів, що близько знаходяться на топологічній карті.

З кожною новою епохою швидкість навчання і розмір околу зменшуються, тим самим всередині ділянок карти виявляються більш тонкі розходження, що зрештою призводить до точнішого налаштування кожного нейрона.

Часто навчання зумисно розбивають на дві фази: більш коротку, з великою швидкістю навчання і великих околів, і більш тривалу з малою швидкістю навчання і нульовими або майже нульовими околами.

Після того, як мережа навчена розпізнавати структури даних, її можна використовувати як засіб візуалізації при аналізі даних.



Ілюстрація навчання самоорганізаційної карти. Синя область — це розподіл тренувальних даних, а маленький білий диск — поточна навчальна вибірка, отримана з цих даних. Спочатку (ліворуч) вузли карти довільно розташовані в просторі даних. Вибирається вузол, найближчий до тренувального вузла (виділений

жовтим кольором), та переміщується у напрямку тренувальних даних, оскільки є (меншою мірою) її сусідами на сітці. Після багатьох ітерацій сітка має тенденцію наближатися до розподілу даних (праворуч).

Області застосування. Кластерний аналіз, розпізнавання образів, класифікація.

Недоліки. Мережа може бути використана для кластерного аналізу маючи заздалегідь відоме число кластерів.

Переваги. Мережа Кохонена здатна функціонувати в умовах завад, тому що число кластерів фіксоване, ваги модифікуються повільно, налаштування ваг закінчується після навчання.

2. Мережа квантування навчального вектора (Learning Vector Quantization)

Мережа була запропонована Тойво Кохоненом у середині 80-х рр., пізніше, ніж його робота із самоорганізованих карт. Мережа базується на шарі Кохонена, здатному до сортування прикладів у відповідні кластери і використовується як для проблем класифікації, так і для кластеризації зображень.

Мережа містить вхідний шар, самоорганізовану карту Кохонена і вихідний шар. Приклад мережі зображено на рис. 3. Вихідний шар має стільки нейронів, скільки є відмінних категорій або класів. Карта Кохонена має ряд нейронів, згрупованих для кожного з цих класів, кількість яких залежить від складності відношення "вхід-вихід". Звичайно, кожен клас буде мати однакову кількість елементів по всьому шарі. Шар Кохонена навчається класифікації за допомогою навчальної множини.

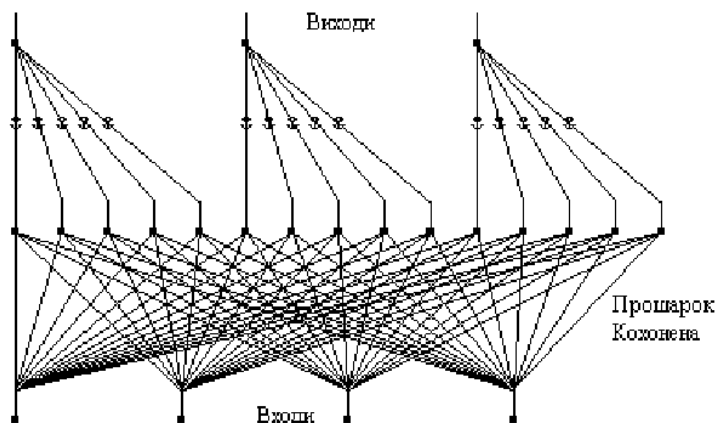


Рис. 3. Приклад мережі з квантуванням навчального вектора

Мережа використовує правила контрольованого навчання. Вхідний шар містить стільки нейронів, скільки є окремих вхідних параметрів. Квантування навчального вектора класифікує свої вхідні дані у визначені групування, тобто відображає n -мірний простір у m -мірний простір (бере n входів і створює m виходів).

Карти зберігають відношення між близькими сусідами в навчальній множині так, що вхідні образи, що не були попередньо вивчені, будуть розподілені по категоріях їхніх найближчих сусідів у навчальних даних.

У режимі навчання, контрольована мережа використовує шар Кохонена, де обчислюється відстань від навчального вектора до будь-якого нейрона і найближчий нейрон з'являється переможцем. Існує лише один переможець на весь шар.

Переможцеві дозволено збуджувати лише один вихідний нейрон, повідомляючи клас або кластер до якого належить вхідний вектор. Якщо нейрон-переможець знаходиться в очікуваному класі навчального вектора, його ваги підсилюються в напрямку навчального вектора. Якщо нейрон-переможець не знаходиться в класі навчального вектора, ваги з'єднань зменшуються. Ця остання операція згадується як відштовхування (*repulsion*). Під час навчання окремі нейрони, що приписані до часткового класу мігрують до області, зв'язаної з їхнім специфічним класом. Під час режиму функціонування, обчислюється відстань від вхідного вектора до будь-якого нейрона і знову найближчий нейрон з'являється переможцем. Це у свою чергу генерує один вихід, визначаючи частковий клас, знайдений мережею.

Недоліки. Для складної класифікації схожих вхідних прикладів, мережа вимагає великої карти Кохонена з великою кількістю нейронів на клас. Це може бути переборено доцільним вибором навчальних прикладів або розширенням вхідного шару. *Деякі нейрони мають тенденцію до перемоги занадто часто*, тобто набудовують свої ваги дуже швидко, у той час як інші постійно залишаються незадіяними. Це часто трапляється, якщо їх ваги мають значення далекі від навчальних прикладів. Для усунення цього, нейрон, що перемагає занадто часто - *штрафується*, тобто зменшуються ваги його зв'язків з кожним вхідним нейроном. Це зменшення ваг пропорційно до різниці між частотою перемог нейрона і частотою перемог середнього нейрона.

Переваги. Алгоритм граничної корекції використовується для удосконалення рішення навіть якщо було знайдено відносно гарне рішення. Алгоритм здатний діяти, якщо нейрон-переможець знаходиться в неправильному класі, а другий найкращий нейрон у правильному класі. Навчальний вектор повинний бути близько від середньої точки простору, що з'єднує ці два нейрони. Неправильний нейрон-переможець зміщається з навчального вектора, а нейрон з іншого місця просувається до навчального вектора. Ця процедура робить чіткою границю між областями, де можлива невірна класифікація. На початку навчання бажано відключити відштовхування. Нейрон-переможець просувається до навчального вектора лише тоді, коли навчальний вектор і нейрон-переможець знаходяться в одному класі. Такий підхід доцільний, якщо нейрон повинний обійти область, яка має відмінний клас для досягнення необхідної області.

3. Мережа зустрічного поширення (CounterPropagation)

Роберт Хехт-Нильсен (Robert Hecht-Nielsen) розробив мережу Counterpropagation як засіб для об'єднання неконтрольованого шару Кохонена з контрольованим вихідним шаром. Мережа призначена для розв'язання складних класифікацій, при мінімізації числа нейронів і часу навчання. Навчання для мережі Counterpropagation схоже на мережі з квантуванням навчального вектора. Приклад

мережі зображено на рис. 4. Односпрямована мережа CounterPropagation має три шари: вхідний шар, самоорганізована карта Кохонена і вихідний шар, що використовує правило "дельта" для зміни вхідних ваг з'єднань. Цей шар називають шаром Гроссберга.

Перша мережа Counterpropagation складалася з двонаправленого відображення між вхідним і вихідним шарами. Дані надходять на вхідний шар для генерації класифікації на вихідному шарі, вихідний шар по черзі приймає додатковий вхідний вектор і генерує вихідну класифікацію на вхідному шарі мережі. Через такий зустрічно-расповсюджений потік інформації мережа одержала свою назву. Багато розроблювачів використовують односпрямований варіант Counterpropagation, коли існує лише один шлях прямого поширення від вхідного до вихідного шару. У мережі зустрічного поширення об'єднані два алгоритми: самоорганізована карта Кохонена і зірка Гроссберга (Grossberg Outstar). Нейроні мережі, що поєднують різні нейропарадигми як будівельні блоки, більш близькі до мозку по архітектурі, чим однорідні структури. Вважається, що в мозку саме каскадні з'єднання модулів різної спеціалізації дозволяють виконувати необхідні обчислення. Кожен елемент вхідного сигналу подається на всі нейрони шару Кохонена. Ваги зв'язків (w_{mn}) утворюють матрицю W . Кожен нейрон шару Кохонена з'єднаний із усіма нейронами шару Гроссберга. Ваги зв'язків (v_{np}) утворюють матрицю ваг V . У процесі навчання мережі зустрічного поширення вхідні вектори асоціюються з відповідними вихідними векторами (двійковими або аналоговими).

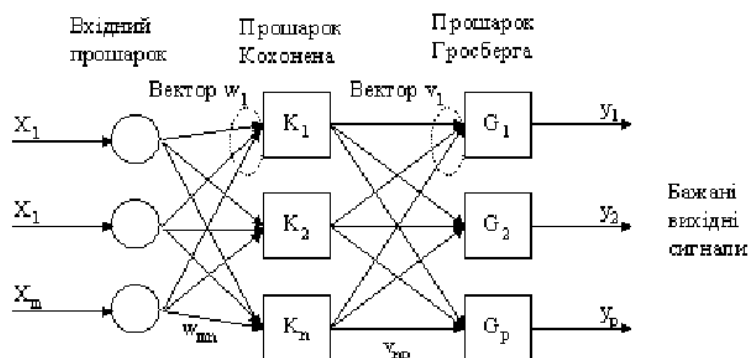


Рис. 4. Мережа зустрічного поширення без зворотних зв'язків

Після навчання мережа формує вихідні сигнали, що відповідають вхідним сигналам. Узагальнююча здатність мережі дає можливість одержувати правильний вихід, коли вхідний вектор неповний або перекручений.

Навчання мережі

У результаті самонавчання шар здобуває здатність розділяти несхожі вхідні вектори. Який саме нейрон буде активуватися при пред'явленні конкретного вхідного сигналу, заздалегідь важко передбачити. При навчанні шару Кохонена на вхід подається вхідний вектор і обчислюються його скалярні добутки з векторами ваг усіх нейронів. *Скалярний добуток* є мірою подібності між вхідним вектором і вектором ваг. Нейрон з максимальним значенням скалярного добутку вважається "переможцем" і його ваги підсилюються (ваговий вектор наближається до вхідного).

$$w_H = w_C + r(x - w_C)$$

де w_H - нове значення ваги, що з'єднує вхідний компонент x з нейроном, що виграв, w_C - попереднє значення ваги, r - коефіцієнт швидкості навчання, що спочатку зазвичай дорівнює 0,7 і може поступово зменшуватися в процесі навчання. Це дозволяє робити великі початкові кроки для швидкого грубого навчання і менші кроки при підході до остаточної величини.

Кожна вага, зв'язана з нейроном-переможцем Кохонена, змінюється пропорційно різниці між його величиною і величиною входу, до якого він приєднаний. Напрямок зміни мінімізує різниця між вагою і відповідним елементом вхідного шару. Навчальна множина може містити багато схожих між собою вхідних векторів, і мережа повинна бути навчена активувати один нейрон Кохонена для кожного з них. Ваги цього нейрона є усередненням вхідних векторів, що його активують. Виходи шару Кохонена подаються на входи нейронів шару Гроссберга. Входи нейронів обчислюються як зважена сума виходів шару Кохонена.

Кожна вага коректується лише в тому випадку, якщо він з'єднаний з нейроном Кохонена, що має ненульовий вихід. Величина корекції ваг пропорційна різниці між вагою і необхідним виходом нейрона Гроссберга. Навчання шару Гроссберга - це навчання "із учителем", алгоритм використовує задані бажані виходи.

Функціонування мережі.

У своїй найпростішій формі шар Кохонена функціонує за правилом "переможець одержує все". Для даного вхідного вектора один і тільки один нейрон Кохонена видає логічну одиницю, всі інші видають нуль. Шар Гроссберга функціонує в схожій манері. Його вихід є зваженою сумою виходів шару Кохонена. Якщо шар Кохонена функціонує так, що лише один вихід дорівнює одиниці, а інші дорівнюють нулеві, то кожен нейрон шару Гроссберга видає величину ваги, що зв'язує цей нейрон з єдиним нейроном Кохонена, чий вихід відмінний від нуля. У повній моделі мережі зустрічного поширення є можливість одержувати вихідні сигнали по вхідним і навпаки. Цим двом діям відповідають пряме і зворотне поширення сигналів.

Області застосування. Розпізнавання образів, відновлення образів (асоціативна пам'ять), стиск даних (із втратами).

Недоліки. Мережа не дає можливості будувати точні апроксимації (точні відображення). У цьому мережа значно уступає мережам зі зворотним поширенням похибки. До недоліків моделі також варто віднести слабкий теоретичний базис модифікацій мережі зустрічного поширення.

Переваги

- Мережа зустрічного поширення проста. Вона дає можливість одержувати статистичні властивості з множини вхідних сигналів. Кохонен довів, що для навченої мережі ймовірність того, що випадково обраний вхідний вектор буде найближчим до будь-якого заданого вагового вектора, дорівнює $1/k$, k - число нейронів Кохонена.

- Мережа піддається швидкому навчанню. Час навчання в порівнянні зі зворотним поширенням може бути в 100 разів менше.
- По своїх можливостях будувати відображення мережа зустрічного поширення значно перевершує одношарові перцептрони.
- Мережа корисна для додатків, де потрібна швидка початкова апроксимація.
- Мережа дає можливість будувати функцію і зворотно до неї, що знаходить застосування при рішенні практичних задач.

Модифікації. Мережі зустрічного поширення можуть розрізнятися способами визначення початкових значень синаптичних ваг.

- Для підвищення ефективності навчання застосовується додавання шуму до вхідних векторів.

- Ще один метод підвищення ефективності навчання - надання кожному нейрону "почуття справедливості". Якщо нейрон стає переможцем частіше, ніж $1/k$ (k - число нейронів Кохонена), то йому тимчасово збільшують поріг, даючи тим самим учитися й іншим нейронам.

- Крім "методу акредитації", при якому для кожного вхідного вектора активується лише один нейрон Кохонена, може бути використаний "метод інтерполяції", при використанні якого ціла група нейронів Кохонена, що мають найбільші виходи, може передавати свої вихідні сигнали в шар Гроссберга. Цей метод підвищує точність відображень, реалізованих мережею.

4. Ймовірнісна нейрона мережа

Ймовірнісна нейрона мережа була розроблена Дональдом Спехтом (Donald Specht). Ця мережна архітектура була вперше представлена в двох статтях : "Ймовірнісні нейронні мережі для класифікації" (Probabilistic Neural Networks for Classification) 1988, "Відображення або асоціативна пам'ять і ймовірнісні нейронні мережі" (Mapping or Associative Memory and Probabilistic Neural Networks) 1990 р.

Виходи мережі можна інтерпретувати, як оцінки ймовірності приналежності елемента до визначеного класу. Ймовірнісна мережа учить оцінювати функцію щільності ймовірності, її вихід розглядається як очікуване значення моделі в даній точці простору входів. Це значення зв'язане з щільністю ймовірності загального розподілу вхідних і вихідних даних. Задача оцінки щільності ймовірності відноситься до області байєсівської статистики. Звичайна статистика по заданій моделі показує, яка ймовірність того або іншого виходу (наприклад, на гральній кісті 6 очок буде випадати в середньому в одному випадку із шести).

Байєсівська статистика інтерпретує по іншому: правильність моделі оцінюється по наявним достовірним даним, тобто дає можливість оцінювати щільність ймовірності розподілу параметрів моделі по наявним даним. При рішенні задач класифікації можна оцінити щільність ймовірності для кожного класу, порівняти між собою ймовірності приналежності до різних класів і вибрати модель з параметрами, при яких щільність ймовірності буде більшою. Оцінка щільності ймовірності в мережі заснована на ядерних оцінках. Якщо приклад розташований у даній точці простору, тоді в цій точці є визначена щільність ймовірності. Кластери з

поруч розташованих точок, свідчать, що в цьому місці щільність ймовірності велика. Біля спостереження є більша довіра до рівня щільності, а в міру віддалення від нього довіра зменшується і прагне до нуля. У методі ядерних оцінок у точці, що відповідає кожному прикладові, міститься деяка проста функція, потім вони усі додаються й у результаті виходить оцінка для загальної щільності ймовірності. Частіше як ядерні функції беруть дзвоноподібні функції (гаусівські). Якщо є достатня кількість навчальних прикладів, такий метод дає гарні наближення до дійсної щільності ймовірності. *Ймовірнісна мережа має три шари: вхідний, радіальний і вихідний.* Радіальні елементи беруться по одному на кожен приклад. Кожний з них має гауссову функцію з центром у цьому прикладі. Кожному класові відповідає один вихідний елемент. Вихідний елемент з'єднаний лише з радіальними елементами, що відносяться до його класу і підсумовує виходи всіх елементів, що належать до його класу. Значення вихідних сигналів виходять пропорційно ядерним оцінкам ймовірності приналежності відповідним класам.

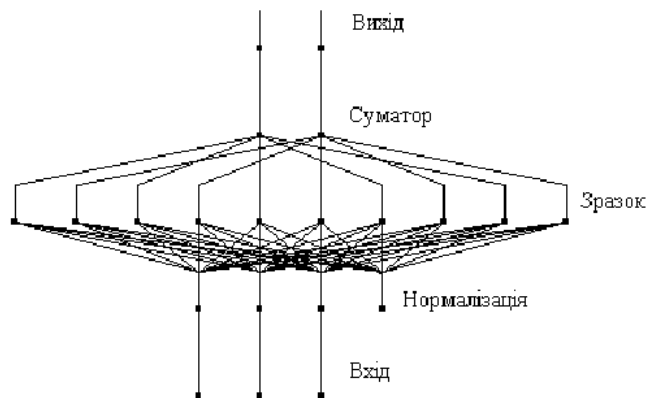


Рис. 5. Схема ймовірнісної нейронної мережі.

Переваги. Навчання ймовірнісної нейронної мережі набагато простіше, ніж BackPropagation.

Недоліки. Недоліком мережі є її розмір, оскільки вона фактично містить у собі всі навчальні дані, вимагає багато пам'яті і може повільно працювати.

Модифікації. Базова модель ймовірнісної нейронної мережі має модифікації. Припустимо, що пропорції класів у навчальній множині відповідають їх пропорціям у всій досліджуваній множині (апріорна ймовірність). Наприклад, якщо серед усіх людей хворими є 2%, то в навчальній множині для мережі, що діагностує захворювання, хворих також повинне бути 2%. Якщо ж апріорні ймовірності відрізняються від пропорції в навчальній вибірці, мережа буде видавати невірний результат. Це можна врахувати, уводячи коригувальні коефіцієнти для різних класів.

5. Мережа Хопфілда

Джон Хопфілд вперше представив свою асоціативну мережу у 1982 р. у Національній Академії Наук. На честь Хопфілда та нового підходу до моделювання, ця мережна парадигма згадується як мережа Хопфілда. Мережа базується на

аналогії фізики динамічних систем. Початкові застосування мережі включали асоціативну, або адресовану за змістом пам'ять та вирішували задачі оптимізації.

Мережа Хопфілда використовує три прошарки: вхідний, прошарок Хопфілда та вихідний прошарок. Кожен прошарок має однакову кількість нейронів. Виходи нейронів вхідного прошарку надходять до входів відповідних нейронів прошарку Хопфілда. Тут, зв'язки мають фіксовані вагові коефіцієнти. Виходи прошарку Хопфілда під'єднуються до входів всіх нейронів прошарку Хопфілда, за винятком самого себе, а також до відповідних елементів у вихідному прошарку. Під час навчання, мережа скеровує дані з вхідного прошарку до прошарку Хопфілда. Прошарок Хопфілда коливається, поки не буде завершена певна кількість циклів, і поточний стан сигналів нейронів прошарку передається на вихідний прошарок. Цей стан відповідає образу, який буде запам'ятовано в мережі.

Навчання мережі Хопфілда вимагає, щоб навчальний образ був представлений на вхідному та вихідному прошарках одночасно. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань. Для правильного навчання мережі відповідні пари "вхід-вихід" мають відрізнитися між собою.

Якщо мережа Хопфілда використовується як пам'ять, що адресується за змістом вона має два головних обмеження.

Число образів, що можна зберегти та точно відтворити є строго обмеженим. Якщо зберігається занадто багато образів, мережа може збігтись до нового неіснуючого образу, відмінного від всіх запрограмованих образів, або не збігтись взагалі. Межа має ємність пам'яті приблизно 15% від числа нейронів у прошарку Хопфілда.

Якщо навчальні приклади є занадто подібними, прошарок Хопфілда може стати нестабільним. Зразок образу вважається нестабільним, якщо він застосовується за нульовий час і мережа збігається до деякого іншого образу з навчальної множини. Ця проблема може бути вирішена вибором навчальних прикладів більш ортогональних між собою.

Структурна схема мережі Хопфілда приведена на рис. 6

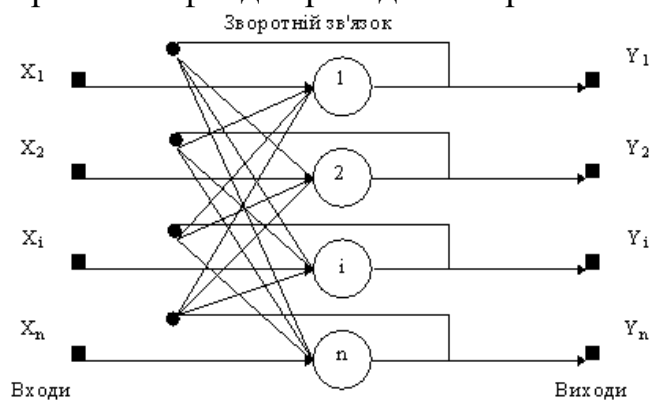


Рис.6. Структурна схема мережі Хопфілда

Для вирішення завдання асоціативної пам'яті є деякий набір двійкових сигналів (зображень, звукових оцифровок, інших даних, що описують об'єкти або характеристики процесів), який вважається зразковим. Мережа повинна вміти з зашумленого сигналу, поданого на її вхід, виділити ("пригадати" за частковою

інформацією) відповідний зразок або "дати висновок" про те, що вхідні дані не відповідають жодному із зразків.

В загальному випадку, будь-який сигнал може бути описаний вектором x_1, x_i, x_n, \dots , n - число нейронів у мережі і величина вхідних і вихідних векторів. Кожний елемент x_i дорівнює або +1, або -1. Позначимо вектор, що описує k -ий зразок, через X_k , а його компоненти, відповідно, - $x_{ik}, k=0, \dots, m-1, m$ - число зразків. Якщо мережа розпізнає (або "пригадує") якийсь зразок на основі пред'явлених їй даних, її виходи будуть містити саме його, тобто $Y = X_k$, де Y - вектор вихідних значень мережі: y_1, y_i, y_n . У протилежному випадку, вихідний вектор не збігається з жодним зразковим.

Якщо, наприклад, сигнали є певним образом зображення, то, відобразивши у графічному виді дані з виходу мережі, можна буде побачити картинку, що цілком збігається з однією зі зразкових (у випадку успіху) або ж "вільну імпровізацію" мережі (у випадку невдачі).

Алгоритм функціонування мережі

1. На стадії ініціалізації мережі вагові коефіцієнти синапсів встановлюються таким чином:

$$w_{ij} = \begin{cases} \frac{\sum_{k=1}^m x_i^k x_j^k}{m}, & i \neq j \\ 0, & i = j \end{cases}$$

Тут i і j - індекси, відповідно, передсинаптичного і постсинаптичного нейронів; x_i^k, x_{jk}^k - i -ий і j -ий елементи вектора k -ого зразка.

2. На входи мережі подається невідомий сигнал (t - номер ітерації). Його поширення безпосередньо встановлює значення виходів:

$$y_i(0) = x_i, i = 0 \dots n-1,$$

тому позначення на схемі мережі вхідних сигналів у явному виді носить чисто умовний характер. Нуль у дужках справа від y_i означає нульову ітерацію в циклі роботи мережі.

3. Обчислюється новий стан нейронів

$$S(t+1) = \sum_{i=1}^n w_{ij} y_j(t), j=0 \dots n-1$$

і нові значення виходів

$$y_j(t+1) = f[S_j(t+1)]$$

де f - передатна функція у виді порогової, приведена на рис.7

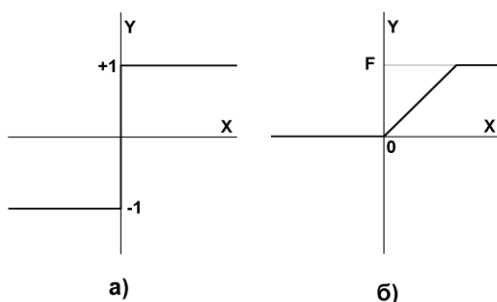


Рис.7. Порогові функції

4. Перевіряємо чи змінилися вихідні значення виходів за останню ітерацію. Якщо так - перехід до пункту 2, інакше (якщо виходи стабілізувались) - кінець. При цьому вихідний вектор являє собою зразок, що найкраще відповідає вхідним даним.

Іноді мережа не може провести розпізнавання і видає на виході неіснуючий образ. Це пов'язано з проблемою обмеженості можливостей мережі. Для мережі Хопфілда число збережених образів m не повинно перевищувати $0.15 \cdot n$ (n – кількість нейронів вихідного прошарку) Крім того, якщо два образи А и Б сильно схожі, вони, можливо, будуть викликати в мережі перехресні асоціації, тобто пред'явлення на входи мережі вектора А призведе до появи на її виходах вектори Б и навпаки.

6. Мережа «Машини Больцмана»

Мережа «Машини Больцмана» (Boltzmann machine) схожа по функції і дії на мережу Хопфілда і включає поняття "модельованого відпалу" для пошуку в просторі станів шару образів глобального мінімуму.

Екли (Ackley), Хинтон (Hinton) і Сейновски (Sejnowski) розробили правило больцмановського навчання в 1985 р. Подібно мережі Хопфілда, машина Больцмана має простір станів, що базується на вагах з'єднань у шарі образів. Процеси навчання мережі, наповненої образами, включає згладжування рельєфу простору станів. *Машини Больцмана моделює відпал металу, що додається до процесу навчання мережі.* Як і при фізичному відпалі, температура починається з великих значень і зменшується з часом. Збільшена температура додає збільшений шумовий коефіцієнт до будь-якого нейрона в шарі образів.

Переважно, кінцевою температурою буде нуль. Для досягнення оптимального рішення доцільно на нижчих температурах додавати більше ітерацій. Машина Больцмана, навчаючись на високій температурі, поводить як випадкова модель, а на низьких температурах вона як детермінована модель. Через випадковий компонент в відпаловому навчанні, нейрон може прийняти нове значення стану, що збільшується швидше, ніж зменшується загальний простір станів. Імітація фізичного відпалу дозволяє просуватися до глобального мінімуму, уникаючи локальний. Як і в мережі Хопфілда, мережі може бути представлений частковий образ для відновлення відсутньої інформації.

Алгоритм функціонування мережі

1. Визначити змінну T , що представляє штучну температуру.
2. Пред'явити мережі множину входів і обчислити виходи і цільову функцію.
3. Дати випадкову зміну вагам і перерахувати вихід мережі і зміну цільової функції відповідно до зміни ваг.
4. Якщо цільова функція зменшилася, тоді нова множина ваг зберігається. Якщо зміна ваг приводить до збільшення цільової функції, то ймовірність збереження цієї зміни обчислюється за допомогою розподілу Больцмана:

$$P(c) = \exp\left(-\frac{c}{kT}\right)$$

де $P(s)$ - ймовірність зміни s у цільовій функції; k - константа, аналогічна константі Больцмана, вибирається в залежності від задачі; T - штучна температура.

Вибирається випадкове число r з рівномірного розподілу від нуля до одиниці. Якщо $P(s)$ більше, ніж r , то зміна зберігається, у противному випадку величина ваги повертається до попереднього значення. Ця процедура дає можливість системі робити випадковий крок у напрямку, що псує цільову функцію, дозволяючи їй вириватися з локальних мінімумів. Кроки 3 і 4 повторюються для усіх ваг мережі, поступово зменшуючи температуру T , поки не буде досягнуто припустиме низьке значення цільової функції. У цей момент пред'являється інший вхідний вектор і процес навчання повторюється. Мережа учить на усіх векторах навчальної множини, поки цільова функція не стане припустимою для усіх з них. Швидкість зменшення температури повинна бути обернено пропорційна логарифмові часу. При цьому мережа сходиться до глобального мінімуму.

Області застосування.

Розпізнавання образів, класифікація.

Недоліки. Повільний алгоритм навчання.

Преваги. Алгоритм дає можливість мережі вибиратися з локальних мінімумів адаптивного рельєфу простору станів.

Модифікації. Випадкові зміни можуть проводитися не тільки для окремих ваг, але і для всіх нейронів шарів у багатошарових мережах або для всіх нейронів мережі одночасно. Ці модифікації алгоритму дають можливість скоротити загальне число ітерацій навчання.

7. Мережа Хемінга

Мережа Хемінга (*Hamming*) є розширенням мережі Хопфілда. Ця мережа була розроблена Річардом Ліппманом (*Richard Lippman*) у середині 80-х рр. Мережа Хемінга реалізує класифікатор, що базується на найменшій похибці для векторів двійкових входів, де похибка визначається відстанню Хемінга. Відстань Хемінга визначається як число бітів, які відрізняються між двома відповідними вхідними векторами фіксованої довжини. Один вхідний вектор є незашумленим прикладом образу, інший є спотвореним образом. Вектор виходів навчальної множини є вектором класів, до яких належать образи. У режимі навчання вхідні вектори розподіляються до категорій для яких відстань між зразковими вхідними векторами та біжучим вхідним вектором є мінімальною.

Мережа Хемінга має три прошарки: вхідний прошарок з кількістю вузлів, скільки є окремих двійкових ознак; прошарок категорій (прошарок Хопфілда), з кількістю вузлів, скільки є категорій або класів; вихідний прошарок, який відповідає числу вузлів у прошарку категорій.

Мережа є простою архітектурою прямого поширення з вхідним рівнем, повністю під'єднаним до прошарку категорій. Кожен нейрон у прошарку категорій є зворотно під'єднаним до кожного нейрона у тому ж самому прошарку і прямо під'єднаним до вихідного нейрону. Вихід з прошарку категорій до вихідного прошарку формується через конкуренцію.

Навчання мережі Хемінга є подібним до методології Хопфілда. На вхідний прошарок надходить бажаний навчальний образ, а на виході вихідного прошарку надходить значення бажаного класу, до якого належить вектор. Вихід містить лише значення класу до якої належить вхідний вектор. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань.

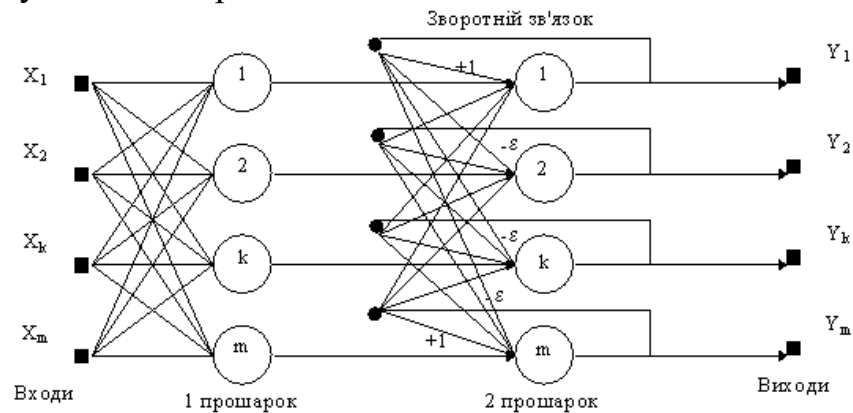


Рис.8.

Алгоритм функціонування мережі Хемінга

1. На стадії ініціалізації ваговим коефіцієнтам першого прошарку і порогу передатної функції присвоюються такі значення:

$$W_{ik} = x_i^k / 2, \quad i=0 \dots n-1, \quad k=0 \dots m-1$$

$$b_k = n / 2, \quad k = 0 \dots m-1$$

Тут x_i^k - i -ий елемент k -ого зразка.

Вагові коефіцієнти гальмуючих синапсів у другому прошарку беруть рівними деякій величині $0 < v < 1/m$. Синапс нейрона, пов'язаний із його ж виходом має вагу +1.

2. На входи мережі подається невідомий вектор $x_1, x_i, x_n \dots$. Розраховуються стани нейронів першого прошарку (верхній індекс у дужках указує номер прошарку):

$$y_j^{(1)} = S_j^{(1)} = \sum_{i=1}^n w_{ij} x_i + b_j, \quad j=0 \dots m-1$$

Після цього отримання значення ініціалізують значення виходів другого прошарку:

$$y_j^{(2)} = y_j^{(1)}, \quad j = 0 \dots m-1$$

3. Обчислюються нові стани нейронів другого прошарку:

$$S_j^{(2)}(t+1) = y_j(t) - \varepsilon \sum_{k=1}^m y_k^{(2)}(t), \quad k \neq j, \quad j = 1 \dots m$$

і значення їх виходів:

$$y_j^{(1)}(t+1) = f | S_j^{(2)}(t+1) |$$

Передатна функція f має вид порога, причому величина b повинна бути достатньо великою, щоб будь-які можливі значення аргументу не призводили до насичення.

4. Перевіряється, чи змінилися виходи нейронів другого прошарку за останню ітерацію. Якщо так - перейти до кроку 3. Інакше - кінець.

Роль першого прошарку є умовною: скориставшись один раз на першому кроці значеннями його вагових коефіцієнтів, мережа більше не вертається до нього, тому перший прошарок може бути взагалі виключений із мережі.

Мережа Хемінга має ряд переваг над мережею Хопфілда. Вона реалізує оптимальний класифікатор мінімуму похибки, якщо похибки вхідних бітів є випадковими та незалежними. Для функціонування мережі Хемінга потрібна менша кількість нейронів, оскільки середній прошарок вимагає лише один нейрон на клас, замість нейрону на кожен вхідний вузол. І, нарешті, мережа Хемінга не страждає від неправильних класифікацій, які можуть трапитись у мережі Хопфілда. В цілому, мережа Хемінга є як швидшою, так і точнішою за мережу Хопфілда.

8. Мережа мережної моделі з двонаправленою асоціативною пам'ятю

Ця мережна моделі була розроблена Бартом Козко (Bart Kosko) і розширює модель Хопфілда. Множина парних образів учить по образах, що представлені як біполярні вектори. Подібно мережі Хопфілда, коли представляється зашумлена версія одного образа, визначається найближчий образ, асоційований з ним. На рис. 9 показаний приклад двонаправленої асоціативної пам'яті.

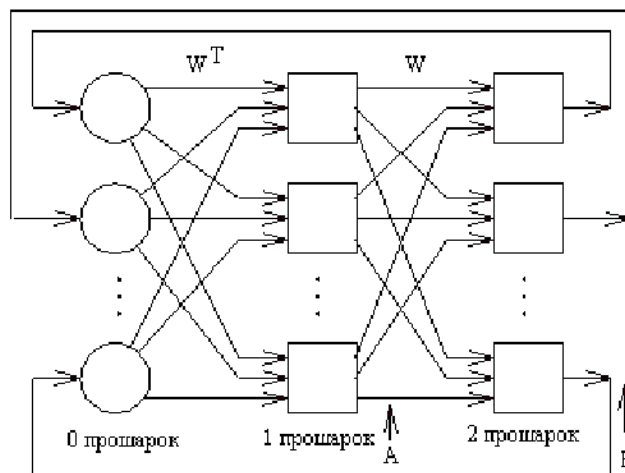


Рис. 9. Двонаправлена асоціативна пам'ять

Вона має стільки входів, скільки є виходів. Два схованих шари містяться на двох окремих асоціативних елементах пам'яті і представляють подвоєний розмір вхідних векторів. Середні шари цілком з'єднані між собою. Вхідний і вихідний шари потрібні для реалізації засобів введення і відтворення інформації з мережі. Середні шари розроблені для збереження асоційованих пар векторів. Якщо на вхід мережі надходить зашумлений вектор образа, середні шари коливаються до досягнення стабільного стану рівноваги, що відповідає найближчій навченій асоціації і на виході генерується образ з навчальної множини. Подібно мережі Хопфілда, двонаправлена асоціативна пам'ять схильна до неправильного відшукування навченого образа, якщо надходить невідомий вхідний вектор, що не був у складі навчальної множини.

Двонаправлена асоціативна пам'ять відноситься до гетероасоціативної пам'яті. Вхідний вектор надходить на один набір нейронів, а відповідний вихідний вектор продукується на іншому наборі нейронів. Вхідні образи асоціюються з вихідними.

Для порівняння: мережа Хопфілда автоасоціативна. Вхідний образ може бути відновлений або виправлений мережею, але не може бути асоційований з іншим образом. У мережі Хопфілда використовується одношарова структура асоціативної пам'яті, у якій вихідний вектор з'являється на виході тих же нейронів, на які надходить вхідний вектор. Двонаправлена асоціативна пам'ять, як і мережа Хопфілда, здатна до узагальнення, виробляючи правильні вихідні сигнали, незважаючи на ушкоджені входи.

Розглянемо схему двонаправленої асоціативної пам'яті.

Вхідний вектор A обробляється матрицею ваг W мережі, у результаті чого продукується вектор вихідних сигналів мережі B . Вектор B обробляється транспонованою матрицею W^T ваг мережі, що продукує сигнали, які представляють новий вхідний вектор A . Цей процес повторюється доти, поки мережа не досягне стабільного стану, у якому ні вектор A , ні вектор B не змінюються.

Нейрони в шарах 1 і 2 функціонують, як і в інших парадигмах, обчислюючи суму зважених входів і значення передатної функції F :

$$b_j = F \sum_i a_i w_{ij}$$

або у векторній формі:

$$B = F(AW)$$

де B - вектор вихідних сигналів нейронів шару 2, A - вектор вихідних сигналів нейронів шару 1, W - матриця ваг зв'язків між шарами 1 і 2, F - передатна функція.

Аналогічно

$$A = F(BW^T),$$

де W^T є транспозицією матриці W .

Як передатну функцію використовується експонентна сігмоїда. Шар 0 не виконує обчислень і не має пам'яті. Він служить лише засобом розподілу вихідних сигналів шару 2 до елементів матриці W^T . Формула для обчислення значень синаптичних ваг:

$$W = \sum_j A_j^T B_j$$

де A_j і B_j - вхідні і вихідні сигнали навчальної множини.

Вагова матриця обчислюється як сума добутків усіх векторних пар навчальної множини. Системи зі зворотним зв'язком мають тенденцію до коливань. Вони можуть переходити від стану до стану, ніколи не досягаючи стабільності. Доведено, що двонаправлена асоціативна пам'ять безумовно стабільна при будь-яких значеннях ваг мережі.

Області застосування.

Асоціативна пам'ять, розпізнавання образів.

Недоліки. Ємкість двонаправленої асоціативної пам'яті жорстко обмежена. Якщо n - кількість нейронів у вхідному шарі, то число векторів, що можуть бути запам'ятовані в мережі не перевищує $L = n/2 \log_2 n$. Так, якщо $n = 1024$, то мережа здатна запам'ятати не більш 25 образів. Двонаправлена асоціативна пам'ять має деяку непередбачуваність у процесі функціонування, можливі помилкові відповіді.

Переваги

- У порівнянні з автосоціативною пам'яттю (наприклад, мережею Хопфілда), двонаправлена асоціативна пам'ять дає можливість будувати асоціації між векторами A і B , що у загальному випадку мають різні розмірності. За рахунок таких можливостей гетероасоціативна пам'ять має більш широкий спектр застосувань, чим автоасоціативна пам'ять.

- Двонаправлена асоціативна пам'ять - проста мережа, що може бути реалізована у виді окремої СБІС або оптоелектроним способом.

- Процес формування синаптичних ваг простий і швидкий. Мережа швидко сходиться в процесі функціонування.

- Сигнали в мережі можуть бути як дискретними, так і аналоговими. Для обох випадків доведена стабільність мережі.

9. Мережа адаптивної резонансної теорії (ART)

Розроблена Стивеном Гроссбергом і Карпентером у середині 80-х рр. Парадигма використовує неконтрольоване навчання, аналізує значимі вхідні дані, виявляє можливі ознаки і класифікує образи у вхідному векторі.

Мережа адаптивної резонансної теорії складається з двох взаємозалежних шарів нейронів, розташованих між вхідним і вихідним шарами. Кожен вхідний образ нижчого шару резонансу стимулює очікуваний образ на вищому шарі, що пересилається до нижчого шару, щоб впливати на наступний вхід. Це створює "резонанс" між нижчим і вищим шарами для полегшення мережної адаптації образів. Мережа переважно використовується в біологічному моделюванні, проте існують деякі технічні застосування. Головним обмеженням мережної архітектури є її шумова чутливість. Навіть невелика кількість шуму на вхідному векторі плутає узагальнюючі можливості навченої мережі. Мережа ART-1 реалізує алгоритм кластеризації, дуже схожий на алгоритм "послідовного лідера". Дотримуючись цього алгоритму перший вхідний сигнал вважається зразком першого кластера. Наступний вхідний сигнал порівнюється зі зразком першого кластера. Говорять, що вхідний сигнал "направляється за лідером" і належить першому кластеру, якщо відстань до зразка першого кластера менша порога. У протилежному випадку другий вхідний сигнал - зразок другого кластера. Цей процес повторюється для всіх наступних вхідних сигналів. Таким чином, число кластерів росте з часом і залежить як від значення порога, так і від метрики відстані, використовуваної для порівняння вхідних сигналів і зразків класів. Основна частина мережі ART-1 схожа на мережу Хеммінга. За допомогою послідовних зв'язків обчислюється відповідність

вхідних сигналів і зразків кластерів. Максимальне значення відповідності підсилюється за допомогою латеральних зв'язків вихідних нейронів.

Мережа ART-1 відрізняється від мережі Хеммінга зворотними зв'язками від вихідних нейронів до вхідних, крім того є можливість виключати вихідний нейрон з максимальним значенням відповідності і проводити тестування відповідності вхідного сигналу і зразків кластерів, як того вимагає алгоритм "послідовного лідера".

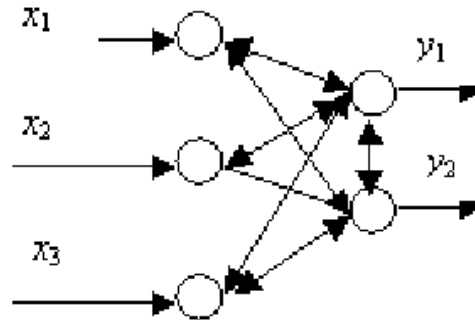


Рис. 10. Основні компоненти класифікатора Карпентер/Гроссберга

Алгоритм функціонування мережі

1. Ініціалізація мережі:

$$v_{ij}(0)=1; W_{ij}(0)=1/(1+N)$$

$$0 \leq i \leq N-1; 0 \leq j \leq M-1; 0 \leq b \leq 1;$$

де $w_{ij}(t)$ - синаптична вага зв'язку від i -го нейрона першого шару до j -го нейрона другого шару в момент часу t , $v_{ij}(t)$ - синаптична вага зв'язку від j -го нейрона другого шару до i -го нейрона першого шару в момент часу t , b - значення порога. Ваги $v_{ij}(t)$ і $w_{ij}(t)$ визначають зразок, що відповідає нейрону j . Поріг b показує, наскільки повинен вхідний сигнал збігатися з одним із запам'ятованих зразків, щоб вони вважалися схожими. Близьке до одиниці значення порога вимагає майже повного збігу. При невеликих значеннях порога вхідний сигнал і зразок, навіть, якщо вони сильно розрізняються, вважаються приналежними до одного кластера.

2. Пред'явлення мережі нового бінарного вхідного сигналу:

Вхідні сигнали пред'являються вихідному шарові нейронів аналогічно мережі Хеммінга.

3. Обчислення значень відповідності:

$$y_j = \sum_{i=1}^N w_{ij}(t) x_i$$

Значення відповідності обчислюються паралельно для всіх зразків, запам'ятованих у мережі, аналогічно мережі Хеммінга.

4. Вибір зразка з найбільшою відповідністю:

$$y_j = \max(y_j)$$

Ця операція виконується за допомогою латерального гальмування.

5. Порівняння з порогом:

$$\|X\| = \sum_{j=1}^M x_j \quad \|V \cdot X\| = \sum_{j=1}^M v_{ij} \times x_j$$

якщо $\frac{\|V \cdot X\|}{\|X\|} > b$ перехід до кроку 7, інакше до кроку 6.

На цьому кроці обчислюється відношення скалярного добутку вхідного сигналу і зразка з найбільшим значенням відповідності до числа одиничних біт вхідного сигналу. Значення відношення порівнюється з порогом, уведеному на першому кроці. Якщо значення відношення більше порога, то вхідний сигнал вважається схожим на зразок найбільшому значенню відповідності. У цьому випадку зразок модифікується шляхом виконання операції AND (логічне "І"). Новий зразок є зразком на попередньому кроці + вхідний сигнал. Якщо значення відношення менше порога, то вважається, що вхідний сигнал відрізняється від усіх зразків і розглядається як новий зразок. У мережу вводиться нейрон, що відповідає новому зразкові, і обчислюються значення синаптичних ваг.

6. Виключення приклада з найбільшим значенням відповідності:

Вихід нейрона з найбільшим значенням відповідності тимчасово встановлюється рівним нулеві і більш не приймає участі в кроці 4.

7. Адаптація приклада з найбільшим значенням відповідності:

$$v_{ij}(t+1) = v_{ij}(t)x_j$$

$$w_{ij}(t+1) = \frac{v_{ij}(t)x_j}{0,5 + \sum_{j=1}^M v_{ij}(t)x_j}$$

8. Включення усіх виключених на кроці 6 зразків.

Повернення до кроку 2.

Вхідні сигнали в цій моделі бінарні.

Розмірності входу і виходу обмежені при програмній реалізації тільки можливостями обчислювальної системи, на якій моделюється нейрона мережа, при апаратній реалізації - технологічними можливостями. Ємкість мережі збігається з числом нейронів другого шару і може збільшуватися в процесі функціонування мережі.

Області застосування. Розпізнавання образів, кластерний аналіз.

Недоліки. Необмежене збільшення числа нейронів у процесі функціонування мережі. З появою шуму виникають значні проблеми, зв'язані з неконтрольованим ростом числа зразків.

Переваги. Навчання без учителя.

Модифікації. Існує модель ART-2 з аналоговими значеннями вхідними сигналами.

Висновки

Серед усіх різновидів і типів нейронних мереж необхідно чітко розрізняти мережі з прямими зв'язками та рекурентні мережі. Також слід розрізняти мережі що навчаються з учителем, та такі що здатні навчатися самі.