**Теоретичний матеріал до лабораторної роботи №5**

**ДВОШАРОВИЙ ПЕРСЕПТРОН**

**1. Обмеженість одношарового персептрона**

Як вже зазначалося раніше, Ф.Розенблатту вдалося навчити свій персептрон розпізнавати букви алфавіту. Це був колосальний успіх: *Електронний пристрій, створений за образом і подобою людського мозку, навчений подібно до людини, успішно моделював інтелектуальні функції людини.* Це був успіх в пізнанні самої природи людського мислення. Мозок почав розкривати свої таємниці. З'явилася можливість досліджувати мозок методами моделювання, не вдаючись до складних антигуманних і мало що корисних природних експериментів. Це була сенсація, яка прикувала до себе увагу мислячих людей всього світу. Здавалося, що ключ до інтелекту був знайдений і повне відтворення людського мозку і всіх його функцій - всього лише питання часу. Письменникам-фантастам, вченим, інженерам, бізнесменам, політикам бачилися дуже райдужні перспективи практичного застосування ідей штучного інтелекту. Уряд Сполучених Штатів Америки виділив великі субсидії на розвиток нового перспективного наукового напрямку.

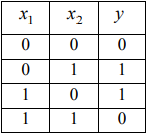
Клас задач, що розв'язували нейромережі розширювався. Але в міру розширення фронту наукових досліджень виникали проблеми. Несподівано виявилося, що багато нових завдань персептрон вирішити не міг. Причому ці нові завдання, зовні нічим не відрізнялися від тих, з якими персептрон успішно справлявся раніше. Виникла необхідність пояснення парадоксів, глибокого аналізу та створення теоретичної бази нейроінформатикі.

Наступний період історії штучного інтелекту почався з появи в 1969 р. книги двох відомих американських математиків *М.Мінського* і *С.Пайперта* «Персептрони». Автори цієї книги математично строго довели, що одношарові персептрони, які використовувалися в той час, в принципі не здатні вирішувати багато простих завдань. Одну з таких задач, що увійшла в історію нейроінформатикі під назвою проблеми «Виключне АБО», ми розглянемо детально.

«Виключне АБО» - це логічна функція двох аргументів, кожен з яких може мати значення «істинно» або «хибно». Сама вона приймає значення «істинно», тільки коли один з аргументів має значення «істинно». У всіх інших випадках ця функція приймає значення «хибно». Якщо закодувати значення «істинно» одиницею, а значення «хибно» - нулем, то необхідну відповідність між аргументами , , і самою функцією *y* можна представити у вигляді табл.1, яка називається таблицею істинності логічної функції.

Таблиця 1

**Таблиця істинності логічної функції «Виключне АБО»**



Завдання полягає в тому, щоб навчитися моделювати функцію «Виключне АБО» за допомогою однонейронного персептрона з двома входами і і одним виходом *y* (рис. 1). При виконанні лабораторної роботи №1 Ви вже намагалися вирішити цю задачу шляхом підбору значень синаптичних ваг , , і порога *θ*, проте зробити це Вам не вдалося. Вам не вдалося це зробити, хоча в інших випадках, при моделюванні логічних функцій «І» та «АБО», у Вас проблем не виникало.

Зовні функції «І», «АБО» і «Виключне АБО» мало чим відрізняються одна від одної, і Вам було не зрозуміло, чому Ваш однонейронний персептрон успішно справлявся з моделюванням двох перших функцій, а з моделюванням третьої функції він впоратися не міг.

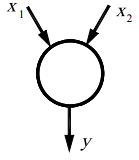
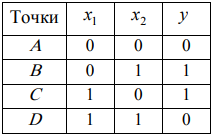


Рис. 1. Однонейронний персептрон з двома входами і одним виходом

Для пояснення цього феномена американськими математиками М.Мінским і С.Пайпертом була запропонована геометрична інтерпретація, яка полягає в наступному. Вони запропонували зобразити на координатній площині , всі можливі комбінації вхідних сигналів у вигляді чотирьох точок: *A, B, C, D,* як показано на рис. 2. Точка *A* має координати = 0, = 0; точка B має координати = 0, = 1 і т.д. згідно табл. 2.

Таблиця 2

**Таблиця істинності логічної функції «Виключне АБО», доповнена тачками *A, B, C, D***



Тоді в точці *A* вихід персептрона y має дорівнювати нулю, в точці *B* - одиниці, в якій точці *C* - одиниці і в точці *D* - нулю.

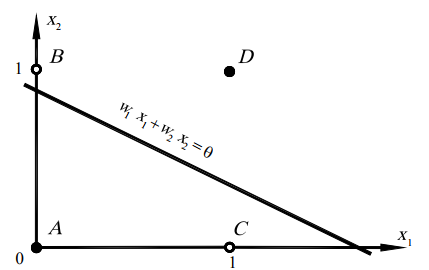


Рис. 2. Графічна інтерпретація до пояснення проблеми «Виключне АБО»

Як нам відомо (див. наприклад лабораторну роботу №1), математичний нейрон Мак-Каллока - Питтса, зображений на рис. 1, здійснює перетворення

(1)

(2)

Замінимо в рівнянні (1) *S* на *θ*:

(1)

Якщо в цьому рівнянні величини і вважати змінними, а *θ*, і - константами, то на координатної площині , розглянуте рівняння відіб'ється у вигляді прямої лінії, положення і нахил якої визначаються значеннями коефіцієнтів і і порога *θ*. Для всіх точок площини і , що лежать на цій лінії, виконується рівність *S = θ* і тому, відповідно до формули (2), вихід персептрона дорівнює одиниці. Для точок, що лежать вище зазначеної лінії сума більша ніж *θ*, і тому за формулами (1)-(2) вихід персептрона також дорівнює одиниці, а для точок, що лежать нижче цієї лінії, сума менша ніж *θ*, і вихід персептрона дорівнює нулю. Тому лінію, яка зображує рівняння (3), називають *пороговою прямою.*

А тепер подивимося на табл. 2. Відповідно до цієї таблиці в точках *A* і *D* вихід персептрона повинен бути нульовим, а в точках *B* і *C* - одиничним. Але для цього треба розташувати порогову пряму так, щоб точки *A* і *D* лежали нижче цієї лінії, а точки *B* і *C* - вище, що неможливо. Це означає, що, скільки б персептрон не навчали, які б значення не надавали його синаптичним вагам і порогові, персептрон в принципі не здатний відтворити співвідношення між входами і виходом, необхідними таблиці істинності функції «Виключне АБО».

Крім проблеми «Виключне АБО» М.Мінський і С.Пайперт привели ряд інших завдань, в яких точки, що зображують вхідні сигнали, не можуть бути розділені пороговою прямою (в багатовимірних випадках - площиною, гиперповерхнею). Такі завдання отримали назву *лінійно нероздільні*.

Після виходу в світ книги М.Минського і С.Пайперта «Персептрони» всім стало ясно, що спроби, які активно вживали в той час щоб навчати персептрони вирішенню багатьох завдань, які, як виявилося, належать до класу лінійно нероздільних, з самого початку були приречені на провал . Це була марна трата часу, сил і фінансових ресурсів.

**------------------------------ Коротко про головне ----------------- -------------**

Однонейронний персептрон в принципі не дозволяє моделювати логічну функцію «Виключне АБО" і вирішувати інші лінійно нероздільні задачі.

**2. Рішення проблеми «Виключне АБО»**

Поява книги М.Мінського і С.Пайперта «Персептрони» викликало шок в науковому світі. Суворі математичні докази М. Мінського і С.Пайперта були невразливі. Загальний ентузіазм замінився не менше загальним песимізмом. В газетах стали з'являтися критичні статті з повідомленнями про те, що вчені в своїх дослідженнях зайшли в глухий кут, даремно витративши величезні державні гроші. Уряд США негайно припинив фінансування нейропроектів і приступив до пошуків винних. Бізнесмени, які втратили надію повернути вкладені капітали, відвернулися від вчених і нейроінформатіка була забута протягом понад 20 років.

Проте, роботи в області нейромережевих і нейрокомп'ютерних технологій продовжувалися окремими ентузіастами. Роботи тривали в засекречених науково-дослідних інститутах Радянського Союзу, який відокремлювався в той час від Заходу «залізною завісою». Не маючи інформації про настрої зарубіжних колег, радянські вчені спокійно продовжували займатися темою, яка захопила їхні розуми і до початку 80-х рр. здивували світ ракетами і літаками, керованими комп'ютерами нового покоління - нейрокомп'ютерами. Радянські комп'ютери, на відміну від американських, стійко переносили досить серйозні пошкодження, продовжуючи працювати в складних умовах, що було особливо важливо для об'єктів військового призначення. Виявилася ще одна властивість нейрокомп'ютерів, успадкована  від мозку - властивість *живучості*.

Радянським вченим *С.О.Мкртчаном* було показано, що за допомогою багатошарових персептронів може бути змодельована будь-яка логічна функція, якщо відома її логічна формула. Більше того, ним був розроблений спеціальний математичний апарат, що дозволяє конструювати такі персептрони. Виявилося, що проблема «Виключне АБО», що стала каменем спіткання для однонейронного персептрона, може бути вирішена за допомогою нейронної мережі, що складається з трьох нейронів - технейронного персептрона, зображеного на рис. 3.

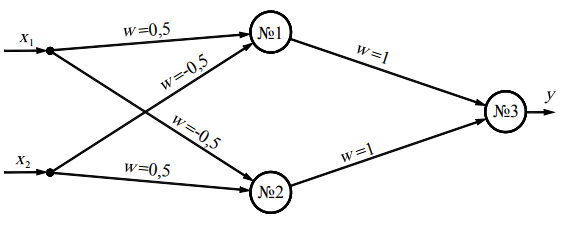
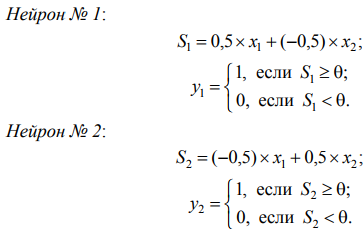
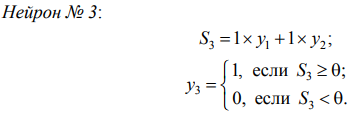


Рис. 3. Нейронна мережа, що моделює функцію «Виключне АБО»

Робота цього персептрона відбувається за наступним алгоритмом.





Поставивши за значення порога θ = 0,5 і заповнивши за допомогою цих формул табл. 3, легко переконатися, що тринейронной персептрон успішно моделює функцію «Виключне АБО».

Таблиця 3

**Процес формування сигналів в трехнейронной персептрони**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | *y* |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | -0,5 | 0,5 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0,5 | -0,5 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Згодом було показано, що і інші лінійно нероздільні завдання, наведені в книзі М.Мінського і С.Пайперта, можуть бути вирішені за допомогою нейромереж, що містять один або кілька прихованих нейронних шарів, тобто шарів нейронів, розташованих між вхідним і вихідним шарами.

Багато дослідників розуміли, що потрібно створювати нейромережі більш складної архітектури, що містять приховані шари нейронів, але не уявляли, як такі мережі навчати. Правила Хебба і їх узагальнення – дельта-правило, годилися тільки для коригування синаптичних ваг нейронів вихідного шару, тоді як питання про налаштування параметрів прихованих нейронних шарів залишалося відкритим.

------------------------------ **Коротко про головне** ----------------- -------------

Логічну функцію «Виключне АБО» може моделювати нейронна мережа, що складається з трьох нейронів, зображена на рис. 3.

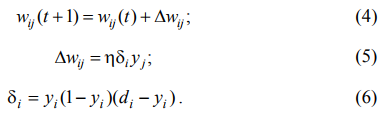
**3. Алгоритм зворотного поширення помилки**

Ефективний алгоритм навчання багатошарових персептронов, який відкрив шлях до їхнього широкого практичного застосування, став відомий тільки в 1986 р. завдяки публікаціям *Д.Румельхарта, Г.Хілтона і Р.Вільямса*. Ідея цього алгоритму полягає в тому, що помилки нейронів вихідного шару використовуються для обчислення помилок нейронів, розташованих в прихованих шарах. Значення помилок поширюються від вихідного шару нейронів всередину мережі від подальших нейронних шарів до попередніх. Звідси назва методу *алгоритм зворотного поширення помилки* *(back propagation).*

Цікаво відзначити, що алгоритм зворотного поширення помилки був запропонований на один рік раніше в роботах *А.Паркера* і *А.Ле-Кана*, виданих незалежно одна від одної. Більше того, ще в 1974 році цей простий і витончений алгоритм був захищений *П.Вербосом* в його докторській дисертації. Однак тоді він залишився непоміченим, і тільки через більш ніж десять років був «перевідкритий» заново і отримав загальне визнання і застосування. Залишилися непоміченими і роботи радянських вчених, які ще раніше розробляли подібні алгоритми в своїх засекречених інститутах і успішно застосовували їх при побудові систем управління об'єктами військового призначення.

Розглянемо ідею алгоритму зворотного поширення помилки, спробувавши узагальнити дельта-правило на випадок навчання *двошарового* персептрона, що має *N* входів, *I* виходів і *прихований* шар з *J* нейронів (рис. 4). Цей персептрон насправді має три шари, проте в літературі його називають двошаровим, оскільки нейрони вхідного шару мають всього один вхід, не мають синаптичних ваг і не виконують підсумовування вхідних сигналів, а лише передають один єдиний вхідний сигнал нейронам наступного шару.

Алгоритм коригування синаптичних ваг нейронів вихідного шару залишимо таким, як для одношарового персептрона (див. Узагальнене дельта-правило: теоретичний матеріал до лабораторної роботи №4), замінивши на :



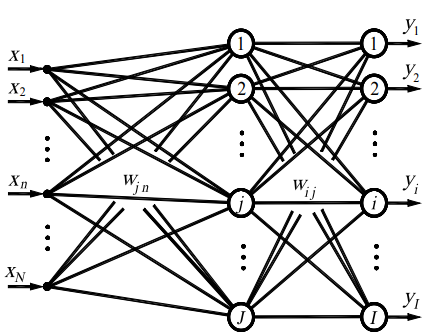
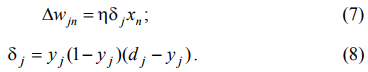


Рис. Двошаровий персептрон з *N* входами, *I* виходами і прихованим шаром з *J* нейронів

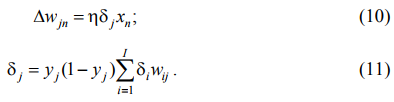
Синаптичні ваги нейронів прихованого шару спробуємо коригувати за допомогою тих же формул, в яких індекс *i* замінимо на *j*, а індекс *j* замінимо на індекс *n*:



При використанні цих формул виникає питання про обчислення нейронної помилки , яка для прихованого шару невідома. Ідея авторів розглянутого алгоритму полягала в тому, щоб в якості цієї помилки використовувати сумарні нейронні помилки з вихідного шару, помножені на сили відповідних синаптичних зв'язків, тобто



Отже, для прихованого шару остаточно маємо



Використовуючи цю ідею, нескладно розписати алгоритм зворотного поширення помилки для навчання персептрона, що має довільну кількість прихованих шарів. Однак перш за все зазначимо, що ми будемо використовувати нейрони, що мають сигмоїдальну активаційну функцію (див. Теоретичний матеріал до лабораторної роботи №4), і виконують операцію підсумовування з урахуванням нейронного зміщення (див. Теоретичний матеріал до лабораторної роботи №1):



Тут - вага додаткового входу *i*-го нейрона, що імітує його зміщення , а = 1 - величина сигналу додаткового входу.

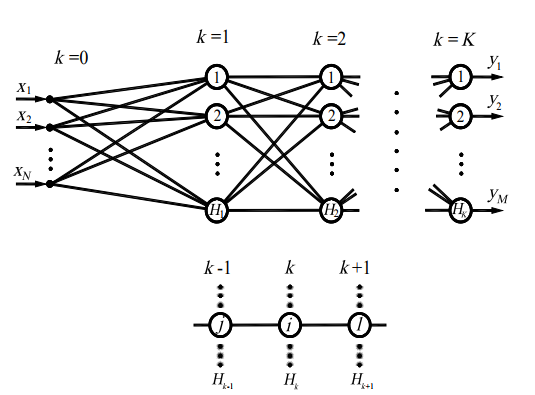


Рис. 5 Багатошаровий персептрон (MLP - MultiLayerPerseptron)

Розпишемо алгоритм зворотного поширення помилки для багатошарового персептрона, що має вхідний шар *k* = 0, кілька прихованих шарів *k* = 1, 2, ..., *K* -1 і вихідний шар *k = K* (рис. 5).

Нейрони вхідного шару не виконують математичних перетворень, а лише передають вхідні сигнали нейронам першого шару. Будемо вважати, що кожен *k*-й шар містить нейронів. Таким чином, персептрон має *N* = входів і *M* = виходів. В алгоритмі будемо використовувати такі позначення: *i* - порядковий номер нейрона *k* -го шару; *j* - порядковий номер нейрона *(k -1)*-го шару; *l* - порядковий номер нейрона *(k +1)* -го шару (див. рис. 5, внизу).

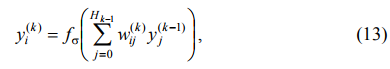
***Крок 1***. Ініціалізація синаптичних ваг і зміщень.

У циклах по *k = 1,2,...,K*; i = 1,2,...,; j = 0,1,2,..., синаптичним вагам і зміщенням датчиком випадкових чисел присвоюються малі величини, наприклад, з інтервалу від -1 до 1.

***Крок 2.*** Відкриття циклу по *q = 1, 2, ..., Q*. Подання з навчальної множини прикладів чергового вхідного вектора і відповідного йому бажаного вихідного вектора , де *q* - номер прикладу в навчальній множині.

***Крок 3.*** Прямий прохід.

У циклах по *k = 1, 2, ..., K*; i = 1,2,..., обчислюються вихідні сигнали *i-го* нейрона в *k-му* шарі



де ; ; ; вихідні сигали персептрона

***Крок 4.*** Зворотний прохід.

У циклах по *k = K,K-1,...,1;* i = 1,2,...,; *j* = 1,2,..., обчислюються синаптичні ваги на новій епосі



де



причому для вихідного шару *k = K* згідно (8)



а для всіх інших прихованих шарів згідно (11)



***Крок 5.*** Закриття циклу по *q*.

***Крок 6.*** Повторення кроків 2 - 5 необхідну кількість разів.

Вектори навчальних прикладів  **і**  на кроці 2 алгоритму зазвичай представляються послідовно від першого до останнього, тобто *q = 1,2, ..., Q*, де *Q* - загальна кількість прикладів. Наприклад, в разі розпізнавання букв російського алфавіту *Q = 33*. Після того, як для кожного навчального прикладу будуть скориговані вагові коефіцієнти персептрона, тобто кроки 2-4 будуть повторені 33 рази, на кроці 6 алгоритму обчислюється середньоквадратична помилка, усереднена по всіх навчальних прикладів:



Крім середньоквадратичної помилки може бути також оцінена максимальна різниця між бажаним і прогнозованим (те, що обчислив персептрон) виходами персептрона:



Ітераційний процес, що задається кроком 6, закінчується після того, як помилка *ε*, що обчислюється за формулами (16) або (17), досягне заданої величини, або коли буде досягнуто граничну кількість епох навчання. В результаті персептрон навчиться виконувати потрібне відображення будь-якого вхідного вектора  на вихідний вектор , що відрізняється від бажаного вектора  на деяку малу величину.

**------------------------------ Коротко про головне ----------------- -------------**

Першим алгоритмом навчання нейронної мережі були правила Хебба, призначені для навчання одношарового персептрона з нейронами, що мають ступінчасті активаційні функції. Потім було введено поняття нейронної помилки як різниці між необхідним виходом нейрона і його реальним значенням . В результаті алгоритм навчання персептрона за допомогою правил Хебба був узагальнений у вигляді алгоритму дельта-правила. В ітераційних формулах алгоритму дельта-правила з'явився коефіцієнт швидкості навчання *η*, що дозволяє впливати на величину ітераційного кроку. Потім була запропонована сигмоїдальна активаційна функція і було введено поняття квадратичної помилки навчання персептрона. В результаті з'явилося узагальнене дельта-правило, яке реалізує метод градієнтного спадання і дозволяє працювати не тільки з бінарними, але і з безперервними сигналами. Алгоритм зворотного поширення помилки є наступним узагальненням узагальненого дельта-правила і дозволяє навчати не лише одношарові, а й багатошарові персептрони.

**4. Види активаційних функцій**

В сучасних нейронних мережах і нейропакетах найбільш часто застосовуються наступні види активаційних функцій.

**Порогові активаційні функції (функції-сходинки)**. Порогові активаційні функції-сходинки можуть мати як несиметричний (рис. 6, *а*), так і симетричний (рис. 6, *б*) вид відносно початку координат.

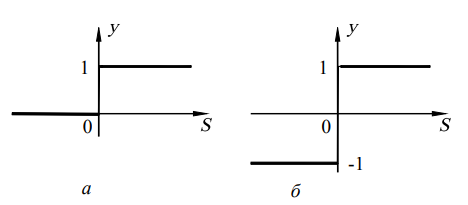


Рис. 6 Порогові активаційні функції-сходинки

Їх аналітичне подання відповідно для рис. 6, *а* і рис. 6, *б*:

де , причому – величина сигналу додаткового входу, а – це його вага, що імітує нейронне зміщення *b* (що дорівнює порогу чутливості нейрона, взятому з протилежним знаком: *b = -θ*).

Ступінчасті активаційні функції зазвичай використовуються в задачах розпізнавання образів. Персептрони зі ступінчастими активаційними функціями можуть навчатися за допомогою правил Хебба і дельта-правила. Узагальнене дельта правило і алгоритм зворотного поширення помилки для навчання таких персептронов не годяться, тому що в ці алгоритми включають знаходження похідних від активаційних функцій, що неможливо для функцій, які мають розрив.

**Лінійні активаційні функції**. На рис. 7, *а* зображено графік лінійної активаційної функції

*y = S.*

Область зміни цієї функції необмежена.

Такі активаційні функції зазвичай застосовуються в нейронах вхідного шару. Вони також непогано працюють при вирішенні простих лінійно роздільних завдань, причому з навчанням таких персептронов можуть справлятися алгоритми дельта-правила, узагальненого дельта-правила і алгоритм зворотного поширення помилки.

Іноді застосовують лінійні активаційні функції з обмеженою областю зміни (рис. 7, б):

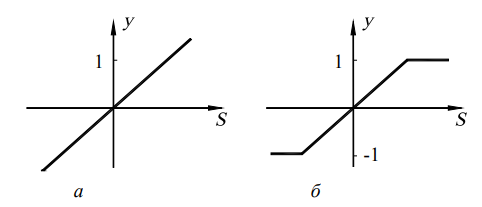


Рис. 7. Лінійні активаційні функції з необмеженою (а) і обмеженою (б) областями зміни

**Сигмоїдальні активаційні функції.** На рис. 8, а зображений графік сигмоїдальної функції, заданої рівнянням

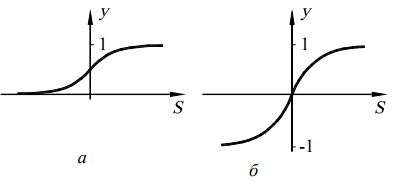


а на рис. 8, б - графік функції, заданої рівнянням



У цих рівняннях коефіцієнт α впливає на кут нахилу ліній до осі *S*.

Аналогічний представленому на останньому графіку вид мають функції арктангенса і гіперболічного тангенса , а також функція , які теж називають сигмоїдальними.



Мал. 8. Сигмоїдальні активаційні функції з несиметричною (а) і симетричною (б) областями зміни

Персептрони з сигмоїдальними активаційними функціями добре навчаються за допомогою алгоритму зворотного поширення помилки, а також за допомогою дельта правила і узагальненого дельта-правила (якщо персептрон одношаровий).

**Логарифмічні активаційні функції**. На рис. 9 представлений графік активаційної функції, заданої рівнянням



На відміну від сигмоїдальної ця функція має необмежену область зміни. Іноді це зручно, тому що не вимагається масштабування вихідних сигналів персептрона (докладніше про це див. теоретичний матеріал до лабораторної роботи №8). Крім того, логарифмічні активаційні функції дозволяють уникати небажаного ефекту, званого *паралічем мережі* - втратою чутливості мережі до варіацій вагових коефіцієнтів і, як наслідок, завмирання процесу навчання при попаданні зважених сум вхідних сигналів нейрона в область насичення сигмоїд (див. також лабораторну роботу №8).

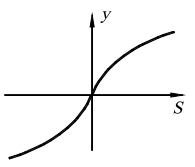


Рис. 9. Логарифмічна активаційна функція

Радіально-базисні активаційні функції. Останнім часом набувають поширення нейромережі, нейрони яких мають активаційні функції в формі функції Гаусса (див. Рис. 10):



де - евклідова відстань між вхідним вектором **X** і *центром активаційної функції* **C**; σ - параметр кривої Гауса, що зветься шириною вікна. Такі активаційні функції називають *радіально-базисними* *(RBF)*, а відповідні нейронні мережі - *RBF-мережами*. Методи навчання RBF-мереж, їх переваги та недоліки в нашому навчальному курсі не розглядаються.

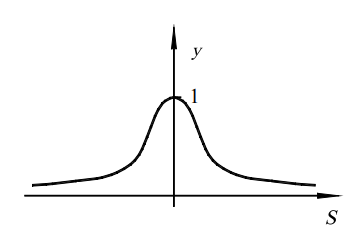


Рис. 10. Радіально-базисна активаційна функція

Відзначимо, що всі наведені вище активаційні функції, за винятком порогової і лінійної, є безперервно диференційованими. Лінійна функція і логарифмічна функція виконують перетворення нескінченної вхідної множини значень змінних *S* в множину змінних *y*. Порогові активаційні функції перетворюють безліч значень *S* в бінарні множини *y = 0* і *y = 1* або *y=-1* і *y=1*. Решта активаційні функції перетворюють нескінченну вхідну множину значень *S* в обмежені вихідні множини: *y ∈ (0,1), y ∈ (-1,1) і y ∈ (0,1]*. Від виду активаційних функцій, які використовуються, залежать функціональні можливості нейронних мереж, а також вибір способів їх навчання.

------------------------------ **Коротко про головне** ----------------- -------------

Активаційні функції здійснюють перетворення зваженої суми вхідних сигналів нейрона в його вихідний сигнал. Від виду активаційних функцій залежать функціональні можливості нейронних мереж, а також вибір способів їх навчання.