Міністерство освіти і науки України

Житомирський державний технологічний університет

**МЕТОДИЧНІ ВКАЗІВКИ**

до виконання лабораторних робіт

з дисципліни

**«Системи штучного інтелекту»**

для студентів спеціальності

"Програмне забезпечення систем"

усіх форм навчання

Житомир

2015

Методичні вказівки до виконання лабораторних робіт з дисципліни " Системи штучного інтелекту " для студентів спеціальності "Програмне забезпечення систем" усіх форм навчання / В.О. Скачков – Житомир: ЖДТУ, 2015.

**ЗМІСТ**

ЗАГАЛЬНІ ПОЛОЖЕННЯ 4

Лабораторна робота №1 5

ОБРОБКА ЕКСПЕРТНИХ ОЦІНОК 5

Завдання до роботи 5

Зміст звіту 5

Контрольні запитання 5

Лабораторна робота № 2 7

БАЄВСЬКИЙ МЕТОД ПРИЙНЯТТЯ РІШЕНЬ 7

Завдання до роботи 7

Зміст звіту 7

Контрольні запитання 7

Лабораторна робота № 3 9

БАЄВСЬКИЙ МЕТОД ПРИЙНЯТТЯ РІШЕНЬ 9

Завдання до роботи 9

Зміст звіту 9

Контрольні запитання 9

Лабораторна робота № 4 11

ПЕРСЕПТРОН РОЗЕНБЛАТТА 11

Завдання до роботи 11

Зміст звіту 11

Контрольні запитання 12

Лабораторна робота № 5 13

НАВЧАННЯ БАГАТОШАРОВОГО ПЕРСЕПТРОНА 13

Завдання до роботи 13

Зміст звіту 13

Контрольні запитання 14

ЛІТЕРАТУРА 15

Додаток А 16

ПРИКЛАДИ ВИКОНАННЯ ЕЛЕМЕНТІВ ЗВІТІВ 16

З ЛАБОРАТОРНИХ РОБІТ 16

Додаток Б 19

ОБРОБКА ЕКСПЕРТНИХ ОЦІНОК 19

Додаток В 26

БАЄВСЬКИЙ МЕТОД ПРИЙНЯТТЯ РІШЕНЬ 26

Додаток Г 30

ПЕРСЕПТРОН РОЗЕНБЛАТТА 30

Додаток Д 33

БАГАТОШАРОВИЙ ПЕРСЕПТРОН 33

# ЗАГАЛЬНІ ПОЛОЖЕННЯ

Дане видання призначене для вивчення та практичного освоєння студентами усіх форм навчання програмного забезпечення інтелектуальних систем.

Відповідно до графіка студенти перед виконанням лабораторної або самостійної роботи повинні ознайомитися з конспектом лекцій та рекомендованою літературою.

Для одержання заліку по кожній роботі студент здає викладачу цілком оформлений звіт з текстами розроблених програм, файлами програм, що виконуються, файлами даних.

Звіт (приклад оформлення - див. додаток А) має містити:

- титульний аркуш (на ньому вказують назву міністерства, назву університету, назву кафедри, номер, вид і тему роботи, виконавця та особу, що приймає звіт, рік);

- мету, варіант і завдання роботи;

- лаконічний опис теоретичних відомостей;

- текст програми, що обов'язково містить коментарі;

- вхідні та вихідні дані програми;

- змістовний аналіз отриманих результатів та висновки.

Звіт виконують на білому папері формату A4 (210 x 297 мм). Текст розміщують тільки на однієї стороні листа. Поля сторінки з усіх боків – 20 мм. Аркуші скріплюють за допомогою канцелярських скріпок. Для набору тексту звіту використовують редактор MS Word 2000-2003: шрифт Times New Roman, 14 пунктів. Міжрядковий інтервал: полуторний – для тексту звіту, одинарний – для листінгів програм, таблиць і роздруківок даних.

Під час співбесіди студент повинний виявити знання про мету роботи, по теоретичному матеріалу, про методи виконання кожного етапу роботи, по змісту основних розділів розробленого звіту з демонстрацією результатів на конкретних прикладах. Студент повинний вміти правильно аналізувати отримані результати. Для самоперевірки при підготовці до виконання і здачі роботи студент повинний відповісти на контрольні питання, приведені наприкінці опису відповідної роботи. Загальний залік студент одержує після виконання і здачі останньої роботи.

# Лабораторна робота №1

## ОБРОБКА ЕКСПЕРТНИХ ОЦІНОК

Мета роботи: Скласти програму яка підраховувала групові експертні оцінки при безпосередньому оцінюванні об’єктів;

### Завдання до роботи

1. Ознайомитися з конспектом лекцій та рекомендованою літературою, а також додатком Б.
2. Написати програму зі зручним та понятим інтерфейсом для користувача, яка б обробляла експерті оцінки.
3. Програма повинна давати можливість вводити: кількість експертів *m*, кількість об’єктів *n*, точність обчислення Е, оцінки експертів (оцінки можуть бути довільними на деякій визначеній шкалі оцінювання), а також враховувати постійне значення h=1.
4. Оформити звіт з роботи.
5. Відповісти на контрольні запитання.

### Зміст звіту

1. Тема та мета роботи.
2. Технічне завдання.
3. Короткі теоретичні відомості.
4. Текст програми, що реалізує обробку експертних оцінок при безпосередньому оцінюванні об’єктів.
5. Приклади роботи програми.
6. Висновки, що містять відповіді на контрольні запитання, а також відображують результати виконання роботи.

Приклади виконання елементів звіту наведено у додатку А.

### Контрольні запитання

1. На чому ґрунтується можливість одержання групової експертної оцінки шляхом додавання індивідуальних оцінок з вагами компетентності й важливості?
2. Як можна обчислити коефіцієнти компетентності експертів?
3. Що є основною ідеєю обчислення коефіцієнтів компетентності експертів?
4. Який вигляд має алгоритм обчислення групових оцінок і коефіцієнтів компетентності експертів для випадку, коли групове оцінювання об’єктів проводиться на основі тільки одного показника (h=1)?

# 

# Лабораторна робота № 2

## БАЙЄСОВСЬКИЙМЕТОД ПРИЙНЯТТЯ РІШЕНЬ

Мета роботи: Навчитися вирішувати задачі діагностики, використовуючи формулу Байеса.

### Завдання до роботи

1. Ознайомитися з конспектом лекцій та рекомендованою літературою, а також додатком В.
2. Написати першу програму, яка реалізує алгоритм обчислення апостеріорних ймовірностей гіпотез при означиванні одного симптому. (Для означівання користувач повинен відповісти на питання про наявність або відсутність у нього симптому. Третя відповідь – «не знаю» )
3. Написати другу програму, яка реалізує алгоритм обчислення апостеріорних ймовірностей однієї гіпотези при означиванні кількох симптомів. Результатом роботи програми є вивід одного з трьох повідомлень: «гіпотеза підтверджується», «гіпотеза спростовується» або «гіпотеза лишається актуальною».
4. Програма повинна давати можливість вводити: назву хвороби і симптомів. Ймовірності підтвердження гіпотези симптомом обчислюються із статистичних даних, а ймовірності спростування гіпотези і  підраховуються як загальні експертні оцінки. (див. лабораторну роботу №1). Апостеріорні ймовірності гіпотез
5. Результати роботи програм
6. Оформити звіт з роботи.
7. Відповісти на контрольні запитання.

### Зміст звіту

1. Тема та мета роботи.
2. Технічне завдання.
3. Короткі теоретичні відомості.
4. Текст програми, що реалізує байєсовський метод прийняття рішень.
5. Приклади роботи програми.
6. Висновки, що містять відповіді на контрольні запитання, а також відображують результати виконання роботи.

Приклади виконання елементів звіту наведено у додатку Б.

### Контрольні запитання

1. На чому ґрунтується Байєсовський метод прийняття рішень?
2. Які бувають гіпотези?
3. Що являють собою апріорні ймовірності гіпотез?
4. Що являють собою апостеріорні вірогідності гіпотез?
5. Яким чином в Байєсовському методі прийняття рішень обраховуються значення апостеріорних вірогідностей?
6. Яким чином проходить процес прийняття рішення для однієї гіпотези?
7. Відношення яких абсолютних значень відіграє вирішальну роль в Байєсовському методі прийняття рішення?

# Лабораторна робота № 3

## БАЙЄСОВСЬКИЙМЕТОД ПРИЙНЯТТЯ РІШЕНЬ

Мета роботи: Навчитися вирішувати задачі діагностики і тестування в навчанні, що потребують врахування неконкретності відповідей тих хто навчається.

### Завдання до роботи

1. Ознайомитися з конспектом лекцій та рекомендованою літературою, а також додатком В.
2. Написати програму зі зручним та понятим інтерфейсом для користувача.
3. Програма повинна розглядати випадок з декількома хворобами і симптомами, різними варіантами відповіді користувача, а також давати відповідь на питання «Вірогідність захворювання на кожну з даних хвороб».
4. Програма повинна давати можливість вводити: назву хвороби, симптоми,  і .
5. Виводити графік можливих траєкторій, що обраховуються.
6. Оформити звіт з роботи.
7. Відповісти на контрольні запитання.

### Зміст звіту

1. Тема та мета роботи.
2. Технічне завдання.
3. Короткі теоретичні відомості.
4. Графік можливих траєкторій, що обраховується і його характеристика.
5. Текст програми, що реалізує байєсовський метод прийняття рішень.
6. Приклади роботи програми.
7. Висновки, що містять відповіді на контрольні запитання, а також відображують результати виконання роботи.

Приклади виконання елементів звіту наведено у додатку А.

### Контрольні запитання

1. На чому ґрунтується Байєсовський метод прийняття рішень?
2. Які бувають гіпотези?
3. Що являють собою апріорні вірогідності гіпотез?
4. Що являють собою апостеріорні вірогідності гіпотез?
5. В якому випадку гіпотеза вважається прийнятою, а в якому - відторгнутою?
6. На скільки класів діляться траєкторії, що приводять до прийняття, відторгнення чи невизначеності гіпотези?
7. На основі чого для кожної гіпотези встановлюються значення верхнього і нижнього порога визначеності гіпотези?
8. Як означуються симптоми?
9. За допомогою чого обраховується нове значення вірогідності гіпотези?

# Лабораторна робота № 4

## ПЕРСЕПТРОН РОЗЕНБЛАТТА

Мета роботи: Навчитися запропонованим методом Ф. Розенблаттом, тобто методом навчання, що складається в ітераційному підстроюванні матриці ваг, що послідовно зменшує помилку у вихідних векторах навчати персептрон.

### Завдання до роботи

1. Ознайомитися з конспектом лекцій та рекомендованою літературою, а також додатком Г.
2. Написати програму зі зручним та понятим інтерфейсом для користувача.
3. Програма повинна давати можливість навчати персептрон. Персептрон визначає парні чи непарні числа введені користувачем.
4. Програма повинна давати можливість вводити числа. Для введення чисел використовується сітка 3х5 (рис. 4.1).



Рис. 4.1 Сітка введення чисел

1. Програма повинна працювати в двох режимах: навчання (навчання персептрона парне чи непарне число) і визначення (відповідати користувачу парне чи непарне число він ввів).
2. Оформити звіт з роботи.
3. Відповісти на контрольні запитання.

### Зміст звіту

1. Тема та мета роботи.
2. Технічне завдання.
3. Короткі теоретичні відомості.
4. Текст програми, що реалізує метод навчання, що складається в ітераційному підстроюванні матриці ваг, що послідовно зменшує помилку у вихідних векторах.
5. Приклади роботи програми.
6. Висновки, що містять відповіді на контрольні запитання, а також відображують результати виконання роботи.

Приклади виконання елементів звіту наведено у додатку А.

### Контрольні запитання

1. Які елементи містить найпростіший класичний персептрон?
2. Чим характеризується однощаровий персептрон?
3. В чому заключається теорема про навчання персептрона?
4. Які кроки включає алгоритм запропонований Ф. Розенблаттом, який грунтується на підстроюванні матриці ваг, що послідовно зменшує помилкуу вихідних векторах?
5. Які обставини враховує формула використана на кроці 3?

# Лабораторна робота № 5

## НАВЧАННЯ БАГАТОШАРОВОГО ПЕРСЕПТРОНА

Мета роботи: Написати програму навчання двошарового персептрона методом зворотного поширення помилок. Використати персептрон для розпізнавання образів (цифр, літер).

### Завдання до роботи

1. Ознайомитися з конспектом лекцій та рекомендованою літературою, а також додатком Д.
2. Написати програму зі зручним та понятим інтерфейсом для користувача.
3. Програма повинна давати можливість вводити наступні вхідні дані:

* темп навчання;
* кількість ітерацій;
* малу повну похибку.

1. Програма повинна давати можливість вводити числа аналогічно лабораторній роботі №4, тобто:



Рис. 5.1 Сітка введення чисел

1. Програма повинна працювати в двох режимах: навчання і розпізнавання.
2. Оформити звіт з роботи
3. Відповісти на контрольні запитання.

### Зміст звіту

1. Тема та мета роботи.
2. Технічне завдання.
3. Короткі теоретичні відомості.
4. Текст програми, що реалізує навчання двошарового персептрона методом зворотного поширення
5. Приклади роботи програми.
6. Висновки, що містять відповіді на контрольні запитання, а також відображують результати виконання роботи.

Приклади виконання елементів звіту наведено у додатку А.

### Контрольні запитання

1. Яким чином влаштовані міжнейронні синаптичні зв’язки мережі?
2. Який напрямок поширення нейроімпульсів?
3. Що представляє собою персептрон?
4. В чому полягає функція нейрона?
5. Які є моделі дослідження нейронів?
6. Чи можна застосовувати для навчання багатошарового персептрона вже відоме  -правило Розенблатта? І чому?
7. Яким методом проводиться навчання багатошарового персепрона?
8. В чому полягає основна ідея методу зворотнього поширення помилок?
9. Який вигляд має загальна структура алгоритму зворотнього поширення помилок?

# ЛІТЕРАТУРА

1. ?????

# Додаток А

## ПРИКЛАДИ ВИКОНАННЯ ЕЛЕМЕНТІВ ЗВІТІВ

## З ЛАБОРАТОРНИХ РОБІТ

|  |
| --- |
| **Міністерство освіти і науки України**  **Житомирський державний технологічний університет**  *кафедра ПЗОТ*  **ЗВІТ**  з лабораторної роботи № 1  з дисципліни "Основи проектування систем штучного інтелекту" на тему:  **"ОБРОБКА ЕКСПЕРТИХ ОЦІНОК"**  Виконала: ст. гр. АК-21 \_\_\_\_\_\_\_\_\_\_\_\_\_ Л. С. Савченко  Прийняв: \_\_\_\_\_\_\_\_\_\_\_\_\_ В. О. Скачков  Житомир  2007 |

Рисунок А.1 - Титульний лист звіту

*Приклад.* Розглянемо виконання певних елементів звіту на прикладі лабораторної роботи №1.

Мета: Скласти програму яка б обробляла експертні оцінки двома методами:

* групова експертна оцінка при безпосередньому оцінюванні;
* обробка парних порівнянь.

**Конкретизація вхідних та вихідних даних**

*Вхідними параметрами є*: кількість експертів m, кількість подій n, точність обчислення Е, оцінки експертів (оцінки можуть бити любими на деякій визначеній шкалі оцінювання), а також постійне значення h1=1.

*Вихідним параметром є* значення групової оцінки.

**Короткі теоретичні відомості**

…

**Текст програми,** що реалізує метод групової експертної оцінки при безпосередньому оцінюванні.

…

**Текст програми**, що реалізує метод обробки парних повідомлень

…

**Результати роботи програми:**

*Групова експертна оцінка при безпосередньому оцінюванні*

# 

Рис. 1 Вхідні дані

# 

Рис. 2 Оцінки експертів

# 

Рис. 3 Значення групової оцінки

*Обробка парних порівнянь* (аналогічно попередньому методу) …

**Висновок:** …

# Додаток Б

## ОБРОБКА ЕКСПЕРТНИХ ОЦІНОК

**Групова експертна оцінка об'єктів при безпосередньому оцінюванні**

Існує багато підходів до рішення даної задачі. Розглянемо один з найпростіших. Нехай ***m*** експертів провели оцінку ***n*** об'єктів по ***l*** показниках. Результати оцінювання представлені величинами  , де ***і*** - номер об'єкта, ***j***- номер експерта, ***h*** - номер показника. Величини , отримані методом безпосереднього оцінювання, являють собою числа з деякого відрізка числової осі, або бали.

За групову оцінку для кожного з об'єктів приймемо середнє зважене значення його оцінок



де *qh* - коефіцієнти ваг показників порівняння об'єктів, *kj* - коефіцієнти компетентності експертів. Величини *qh* й *kj* є нормованими, тобто



Коефіцієнти *qh* можуть бути визначені експертним шляхом, як середній коефіцієнт ваги *h*-ого показника по всіх експертах, тобто



Можливість одержання групової експертної оцінки шляхом додавання індивідуальних оцінок з вагами компетентності й важливості ґрунтується на виконанні:

* аксіом теорії корисності фон Неймана-Моргенштерна для індивідуальних і групових оцінок [3];
* і умов нерозрізненості об'єктів у груповому відношенні, якщо вони нерозрізнені у всіх індивідуальних оцінках (частковий принцип Парето) [4].

Коефіцієнти компетентності експертів можна обчислити за апостеріорними даними, тобто за результатами оцінки об'єктів. Основною ідеєю цього обчислення є припущення про те, що компетентність експерта повинна оцінюватися по ступені погодженості його оцінок із груповою оцінкою об'єктів.

В подальшого обмежимось розглядом випадку *h*=1. Тобто коли групове оцінювання об'єктів проводиться на основі тільки одного показника. Алгоритм обчислення групових оцінок і коефіцієнтів компетентності експертів для цього випадку має вигляд:

а) початкові умови при *t*=0



тобто початкове значення коефіцієнтів компетентності для всіх експертів приймається однаковим і рівним.

б) рекурентні співвідношення для *t*=1,2,3 ...

 - групова оцінка для ***і***-ого об'єкта на ***t***-ому кроці на основі індивідуальних оцінок ***xіj .***

 - нормувальний коефіцієнт

 - коефіцієнти компетентності ***j***-ого експерта на ***t***-ому кроці

 - коефіцієнти компетентності ***m***-ого експерта з умови нормування.

в) умова закінчення ітераційного процесу



Збіжність даної ітераційної процедури доведена в літературі для випадку, коли індивідуальні оцінки невід’ємні, а експерти й об'єкти не розпадаються на окремі групи (тобто коли кожна група експертів не оцінює об'єкти своєї групи). У більшості практичних завдань ці умови виконуються, що доводить збіжність алгоритму [5].

Приклад. Дві фірми (*n*=2) з розробки програмного забезпечення створили програмний продукт, який рішає одну і ту ж задачу. Три експерти (*m*=3) оцінили програмний продукт кожної фірми за одним критерієм (*l*=1). Результатами експертизи виявились нормовані оцінки *x*1j+*x*2j=1, *j*=1,2,3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *xіj* | Експерт 1 | Експерт 2 | Експерт 3 |  |
| Фірма 1 | 0,3 | 0,5 | 0,2 |  |
| Фірма 2 | 0,7 | 0,5 | 0,8 |  |

Обчислимо групові оцінки й коефіцієнти компетентності кожного з експертів. Для цього скористаємося наведеним вище алгоритмом, взявши точність обчислення *ε* =0,001.

Середні оцінки об'єктів першого наближення (при *t*=1) будуть рівні:

 *x*1 =(0,333;0,667)

Обчислимо нормувальний коефіцієнт *λ*1



Значення коефіцієнтів компетентності першого наближення приймуть значення:





 і тоді *k*1 =(0,34;0,30;0,36)

Обчислюючи групові оцінки другого й т.д. наближення, одержимо:

Результат третього кроку задовольняє умові закінчення ітераційного процесу й за значення групової оцінки приймається x ≈ x3 = (0,3235; 0,6765).

**Обробка парних порівнянь**

При встановленні причинно-наслідкових залежностей між об'єктами предметної області, експертам у ряді випадків складно виразити їх чисельно. Тобто важко встановити кількісно ступінь впливу тієї або іншої причини (об'єкта) на конкретний наслідок. Особливо психологічно це складно, якщо таких об'єктів багато.

Разом з тим, експерти порівняно легко вирішують завдання парного порівняння. Це завдання полягає в тім, що експерт встановлює переваги об'єктів при порівнянні всіх можливих пар. Тобто експерт, розглядаючи всі можливі пари об'єктів, у кожній з них встановлює ту причину, що на його думку дуже впливає на наслідок. Виникає питання, як одержати ***оцінку всієї сукупності об'єктів*** на основі результатів парного порівняння, виконаного групою експертів.

Нехай кожний з ***m*** експертів робить оцінку впливу на результат всіх пар об'єктів, даючи числову оцінку



, якщо об’єкт Oi більше значимий, ніж Oj

, об’єкти Oi и Oj рівноправні

, якщо об’єкт Oi менш значимий, ніж Oj

де h=1,2,...m - номер експерта, і,j=1,2,...n - номера об'єктів, досліджуваних при експертизі. Т. е. за результатами експертизи маємо m-таблиць (матриць) виду (мал.7):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Rm | O1 | ... | Oj | ... | On |  |  |  |  |  |  |  |  |  |
|  | R2 | O1 | ... | Oj | ... | On |  |  |  | **O1** | **...** | **Oj** | **...** | **On** |  | **K** |
| **R1** | **O1** | **...** | **Oj** | **...** | **On** |  |  |  | **O1** |  |  |  |  |  |  | **K1** |
| **O1** |  |  |  |  |  |  |  |  | **...** |  |  |  |  |  |  | **...** |
| **...** |  |  |  |  |  |  |  | **⇒** | **Oі** |  | ***xіj=M[rіj]*** | | |  | **⇒** | **Kі** |
| **Oі** |  |  | ***rіj1*** |  |  |  |  |  | **...** |  |  |  |  |  |  | **...** |
| **...** |  |  |  |  |  |  |  |  | **On** |  |  |  |  |  |  | **Kn** |
| **On** |  |  |  |  |  |  |  | *m* |  |  |  |  |  |  |  |  |

Рис.1. Послідовність обробки парних порівнянь

Як випливає з рис.1послідовність обробки парних порівнянь полягає в тому, що на підставі таблиць парних порівнянь *m*-експертів будується матриця математичних очікувань оцінок всіх пар об'єктів. Потім по цій матриці обчислюється вектор коефіцієнтів відносної важливості об'єктів.

Якщо при оцінці пари ***Oіj*** із загальної кількості експертів ***mі*** висловилися на користь ***Oі*** , ***mj*** експертів на користь ***Oj*** , а ***mp*** вважає ці об'єкти рівноправними, то оцінка математичного очікування дискретної випадкової величини ***rіj*** буде дорівнювати:



Так як загальна кількість експертів  , то визначаючи звідси ***mp*** і підставляючи його у вищенаведений вираз, одержимо



Очевидно, що ***хіj***+ ***хjі*** = 1. Сукупність величин ***хіj*** утворять матрицю ***Х***=||***хіj***|| розмірності ***n*** ***n***, на основі якої можна побудувати ранжування всіх об'єктів і визначити коефіцієнти відносної важливості об'єктів, тобто вектор



*k =* [*k1, k2, ... kn*]*T*

Одним зі способів визначення значень елементів вектора *К* є ітераційний алгоритм виду:

а) початкова умова t=0



б) рекурентні співвідношення



де ***Х*** - матриця математичних очікувань оцінок пар об'єктів, ***kt*** - вектор

коефіцієнтів відносної важливості об'єктів порядку ***t***.

- умова нормування.

в) ознака закінчення ||k*t* - k*t*-1||<*E*.

Якщо матриця ***Х*** невід’ємна й нерозкладна (тобто шляхом перестановки рядків і стовпців її не можна привести до трикутного виду), то при збільшенні порядку *t* *→ ∞* величина *λt* сходиться до максимального власного числа матриці ***Х***, тобто



Це твердження випливає з теореми Перрона-Фробеніуса й доводить збіжність наведеного вище алгоритму [6].

Приклад. Припустимо, що в результаті опитування трьох (*m*=3) експертів про ступінь впливу на результат трьох (*n*=3) різних факторів (об'єктів) отримані наступні таблиці парних порівнянь:

Експетр 1(R1) Експерт 2(R2) Експерт 3(R3)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **О1** | **О2** | **О3** |  |  | **О1** | **О2** | **О3** |  |  | **О1** | **О2** | **О3** |
| **О1** | 0,5 | 1 | 1 |  | **О1** | 0,5 | 0,5 | 0,5 |  | **О1** | 0,5 | 1 | 0,5 |
| **О2** | 0 | 0,5 | 0 |  | **О2** | 0,5 | 0,5 | 0,5 |  | **О2** | 0 | 0,5 | 0 |
| **О3** | 0 | 1 | 0,5 |  | **О3** | 0,5 | 0,5 | 0,5 |  | **О3** | 0,5 | 1 | 0,5 |

Для одержання групової оцінки ступеня впливу кожного з об'єктів на результат, побудуємо матрицю математичних очікувань оцінок кожної з пар об'єктів, що для розглянутого прикладу буде мати вигляд:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **О1** | **О2** | **О3** |
| **О1** | 3/6 | 5/6 | 4/6 |
| **О2** | 1/6 | 3/6 | 1/6 |
| **О3** | 2/6 | 5/6 | 3/6 |

Значення елементів цієї матриці отримані з наступних виразів:



Скористаємося вищеописаним алгоритмом для одержання вектора відносної важливості об'єктів. Для наочності, кожний із кроків представимо у вигляді:

*крок 0:*



*крок 1:*



*крок 2:*





Продовжуючи ітераційний процес доти , поки норма оцінки не буде менше заданої ( (|*Kіt*- *Kіt-1*|) < 0,001) одержимо



На четвертому кроці виконується умова виходу, що дозволяє за групову оцінку ступеня впливу на результат прийняти вектор коефіцієнтів відносної важливості об'єктів виду:



# Додаток В

## БАЙЄСОВСЬКИЙМЕТОД ПРИЙНЯТТЯ РІШЕНЬ

Вирішення задач діагностики і тестування в навчанні потребує врахування неконкретності відповідей тих хто навчається. Врахувати цю неконкретність дозволяє байєсовський метод прийняття рішень. Байєсовський метод має строге математичне обґрунтування, і на його основі можна реалізувати механізм вивода, що дозволяє вирішувати вказані вище задачі.

Байєсовський метод заснований на розумінні вірогідності деякої події як деякої оцінки, яка приписується цій події людиною і може змінюватись при надходженні будь-яких додаткових відомостей. Математичний фундамент цього методу складає теорема Байєса. Вона розглядає подію S з вірогідністю P(S)>0 і множину попарно несумісних подій  , повну в тому розумінні, що одна з цих подій неодмінно настане. Тоді, виходячи з теореми, вірогідність події при умові, що наступила подія S, може бути розрахована по формулі:

,

де - апріорні (завчасно відомі) ймовірності гіпотез,

- апостеріорні (отримані в процесі) ймовірності гіпотез,

- симптом,

- вірогідність підтвердження гіпотези  симптомом ,

- вірогідність не підтвердження гіпотези  симптомом .

Для оцінки впливу симптому  на гіпотезу  необхідно означити, симптом, наприклад, спитавши у користувача чи має місце подія . Якщо подія має місце (відповідь «Так»), то для обрахування нової (апостеріорної) вірогідності  використовується формула:

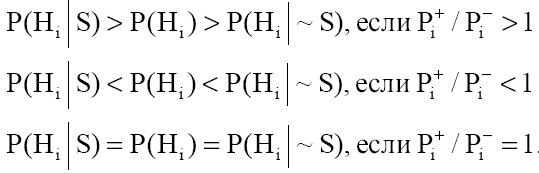


а якщо не має місце (відповідь «Ні»),то:



У випадку відповіді «Не знаю» апостеріорна ймовірність гіпотези  не змінюється.

Можна побачити, що справедливе співвідношення:



Для врахування невизначеності знання користувача при означувані симптомів необхідно розширити список його можливих відповідей. Це легко зробити користувачу означити симптом на шкалі від  до , ,  - відповідає відповіді «Ні», 0 - відповіді «Не знаю», а  - відповіді «Так». Наприклад, при відповідь користувача «4» відповідала б степені впевненості «Дуже навіть може бути, що «Так», а відповідь «-3» - «схоже, що «Ні» . Значення апостеріорних вірогідностей в цьому випадку, обраховують використовуючи шматково-лінійну інтерполяцію між значеннями , , .

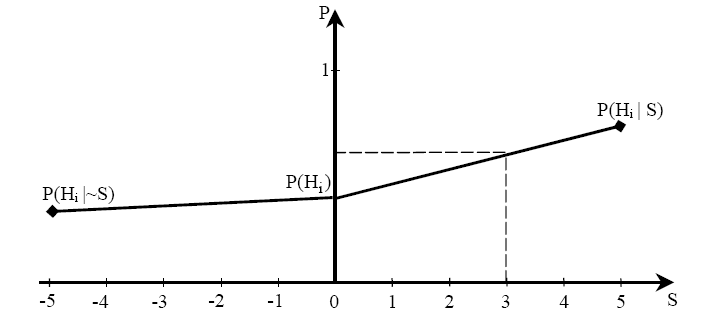
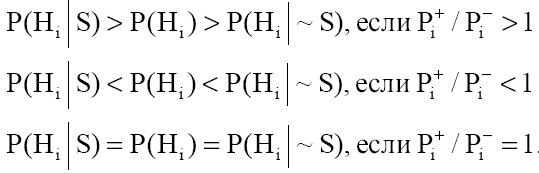


Рис. Для обліку невизначеностей знань користувача

**2.** Процес прийняття рішення для однієї гіпотези проходить наступним чином. Нехай - гіпотеза, якій приписана апостеріорна вірогідність і яка залежить від симптомів , де . З кожним симптомом  пов’язані дві вірогідності:  і .

Перша вірогідність характеризує степінь участі симптому в даній гіпотезі (наприклад, вірогідність високої температури (симптом ) при хворобі грип (гіпотеза )), а друга вірогідність характеризує ступінь участі симптому в других гіпотезах (продовжуючи приклад, можна виділити, що це може бити вірогідність високої температури, якщо пацієнт хворий не грипом). Вірогідності  і  незалежні, тобто кожна із цих вірогідностей незалежно від другої може приймати значення від 0 до 1. Якщо значення , як правило, можна оцінити використовуючи накопичені данні (історії хвороб, журнали обліку успіхів тощо), то значення  оцінити практично неможливо, тому оцінка  практично завжди залежить від експерта чи людини, яка відіграє роль експерта. Як випливає з формул



в баєвському методі вирішальну роль відіграє не абсолютне значення  і , а їх відношення . Саме ці відношення на ряду з апріорними вірогідностями гіпотез і представляють собою експертні знання.

**3.** Симптоми означуються в порядку їх слідування, і за допомогою цих значень симптомів обраховується нове значення вірогідності гіпотези. Якщо на питання вірогідності гіпотези допустимі відповіді «Так», «Ні», «Не знаю», то всього буде  послідовностей можливих відповідей, які задають різні траєкторії обчислення. Якщо відповідати завжди так, щоб вірогідність збільшувалась (зменшувалась), то отримаємо траєкторію, яка приводить до максимальної (мінімальної) можливості вірогідності даної гіпотези. На основі максимальної і мінімальної вірогідності для кожної гіпотези встановлюються значення верхнього і нижнього порога визначеності гіпотези. Наприклад, в якості верхнього порога визначеності гіпотези  - можна взяти значення максимальної вірогідності, яке помножене на коефіцієнт 0,8, а в якості нижнього порогу визначеності - значення максимальної вірогідності, яке помножено на коефіцієнт 0,2. Гіпотеза вважається прийнятною, якщо траєкторія, що обчислюється дає значення вірогідності гіпотези, таке що перевищує значення верхнього порогу, і відторгнутою, якщо значення вірогідності гіпотези менше значення нижнього порогу. Таким чином всі траєкторії, що обчислюються діляться на 3 класи траєкторія, що приводять до прийняття, відторгнення чи невизначеності гіпотези.

# Додаток Г

## ПЕРСЕПТРОН РОЗЕНБЛАТТА

Однієї з перших штучних мереж, здатних до перцепції (сприйняттю) і формуванню реакції на сприйнятий стимул, з'явився PERCEPTRON Розенблатта (F.Rosenblatt, 1957). Персептрон розглядався його автором не як конкретний технічний обчислювальний пристрій, а як модель роботи мозку. Потрібно помітити, що після декількох десятиліть досліджень сучасні роботи зі штучних нейронних мереж рідко переслідують таку мету.

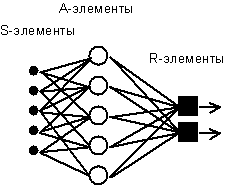


Рис. 1. Елементарний персептрон Розенблатта.

Найпростіший класичний персептрон містить нейроподібні елементи трьох типів (див. Рис. 1), призначення яких у цілому відповідає нейронам рефлекторної нейронної мережі. S-елементи формують сітківку сенсорних кліток, що приймають двійкові сигнали від зовнішнього миру. Далі сигнали надходять у шар асоціативних або A-елементів (для спрощення зображення частина зв'язків від вхідних S-кліток до A-кліток не показана). Тільки асоціативні елементи, що представляють собою формальні нейрони, виконують нелінійну обробку інформації й мають змінювані ваги зв'язків. R-елементи з фіксованими вагами формують сигнал реакції персептрона на вхідний стимул.

Розенблатт називав таку нейронну мережу тришарової, однак по сучасній термінології, представлена мережа звичайно називається одношарової, тому що має тільки один шар нейропроцесорних елементів. Одношаровий персептрон характеризується матрицею синаптических зв'язків W від S- до A-елементів. Елемент матриці відповідає зв'язку, що веде від і-го S-елемента до j-му A-елемента.

У Корнельской авіаційної лабораторії була розроблена електротехнічна модель персептрона MARK-1, що містила 8 вихідних R-елементів й 512 A-елементів, які можна було з'єднувати в різних комбінаціях. На цьому персептроні була проведена серія експериментів по розпізнаванню букв алфавіту й геометричних образів.

У роботах Розенблатта був зроблений висновок про те, що нейронна мережа розглянутої архітектури буде здатна до відтворення будь-якої логічної функції, однак, як було показано пізніше М. Мінським і С. Пейпертом (M.Mіnsky, S.Papert, 1969), цей висновок виявився неточним. Були виявлені принципові непереборні обмеження одношарових персептронів, і в наслідку став в основному розглядатися багатошаровий варіант персептрона, у якому є кілька шарів процесорних елементів.

Із сьогоднішніх позицій одношаровий персептрон представляє скоріше історичний інтерес, однак на його прикладі можуть бути вивчені основні поняття й прості алгоритми навчання нейронних мереж.

Теорема про навчання персептрона.

Навчання мережі складається в підстроюванні вагових коефіцієнтів кожного нейрона. Нехай є набір пар векторів (x, y),  = 1..p, називаний навчальною вибіркою. Будемо називати нейронну мережу навченої на даній навчальній вибірці, якщо при подачі на входи мережі кожного вектора x, на виходах щораз виходить відповідний вектор y.

Запропонований Ф.Розенблаттом метод навчання складається в ітераційному підстроюванні матриці ваг, що послідовно зменшує помилку у вихідних векторах. Алгоритм включає кілька кроків:

|  |  |
| --- | --- |
| Крок 0. | Початкові значення ваг всіх нейронів Image16покладаються випадковими. |
| Крок 1. | Мережі пред'являється вхідний образ x, у результаті формується вихідний образ Image17 |
| Крок 2. | Обчислюється вектор помилки Image18, що робиться мережею на виході. Подальша ідея полягає в тому, що зміна вектора вагових коефіцієнтів в області малих помилок повинне бути пропорційно помилці на виході, і дорівнює нулю якщо помилка дорівнює нулю. |
| Крок 3. | Вектор ваг модифікується по наступній формулі: Image19.  Тут Image20- темп навчання. |
| Крок 4. | Кроки 1-3 повторюються для всіх навчальних векторів. Один цикл послідовного пред'явлення всієї вибірки називається епохою. Навчання завершується після закінчення декількох епох:  а) коли ітерації зійдуться, тобто вектор ваг перестає змінюватися;  б) коли повна сумування по всіх векторах абсолютна помилка стане менше деякого малого значення. |

Використовувана на кроці 3 формула враховує наступні обставини:

а) модифікуються тільки компоненти матриці ваг, що відповідають ненульовим значенням входів;

б) знак приросту ваги відповідає знаку помилки, тобто позитивна помилка ( > 0, значення виходу менше необхідного) проводить до посилення зв'язку;

в) навчання кожного нейрона відбувається незалежно від навчання інших нейронів, що відповідає важливому з біологічної точки зору, принципу локальності навчання.

Даний метод навчання був названий Ф. Розенблаттом “методом корекції зі зворотною передачею сигналу помилки”. Пізніше більш широко стала відомо назва “ - правило ”. Представлений алгоритм ставиться до широкого класу алгоритмів навчання із учителем, оскільки відомі як вхідні вектора, так і необхідні значення вихідних векторів (є вчитель, здатний оцінити правильність відповіді учня).

Доведена Розенблаттом теорема про збіжність навчання по  - правилу говорить про те, що персептрон здатний навчиться будь-якому навчальному набору, що він здатний представити.

# Додаток Д

## БАГАТОШАРОВИЙ ПЕРСЕПТРОН

Розглянемо ієрархічну мережеву структуру, у якій зв'язані між собою нейрони (вузли мережі) об'єднані в кілька шарів (Рис. 1). На можливість побудови таких архітектур указав ще Ф.Розенблатт, однак їм не була вирішена проблема навчання. Міжнейронні синаптичні зв'язки мережі влаштовані таким чином, що кожен нейрон на даному рівні ієрархії приймає й обробляє сигнали від кожного нейрона більше низького рівня. Таким чином, у даній мережі є виділений напрямок поширення нейроімпульсів - від вхідного шару через один (чи декілька) прихованих шарів до вихідного шару нейронів. Нейромережу такої топології ми будемо називати узагальненим багатошаровим персептроном або, якщо це не буде викликати непорозумінь, просто персептроном.

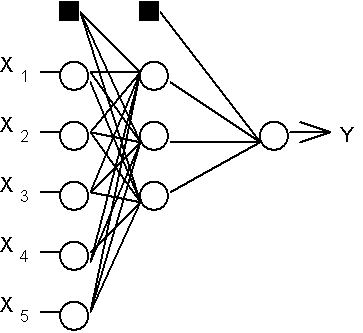


Рис. 1. Структура багатошарового персептрона з п'ятьма входами, трьома нейронами в схованому шарі, і одним нейроном вихідного шару.

Персептрон представляє собою мережу, що складається з декількох послідовно з'єднаних шарів формальних нейронів МакКаллока й Питтса. На нижчому рівні ієрархії перебуває вхідний шар, що складається із сенсорних елементів, задачею якого є тільки прийом і поширення по мережі вхідної інформації. Далі є один або, рідше, кілька схованих шарів. Кожен нейрон на схованому шарі має кілька входів, з'єднаних з виходами нейронів попереднього шару або безпосередньо із вхідними сенсорами X1..Xn, і один вихід. Нейрон характеризується унікальним вектором вагових коефіцієнтів w. Ваги всіх нейронів шаруючи формують матрицю, що ми будемо позначати V або W. Функція нейрона складається в обчисленні зваженої суми його входів з подальшим нелінійним перетворенням її у вихідний сигнал:

Image28 (5.1)

Виходи нейронів останнього, вихідного, шару описують результат класифікації Y=Y(X). Особливості роботи персептрона полягають у наступному. Кожен нейрон підсумовує сигнали, що до нього поступили від нейронів попереднього рівня ієрархії з вагами, обумовленими станами синапсів, і формує відповідний сигнал (переходить у збуджений стан), якщо отримана сума вище граничного значення. Персептрон перетворює вхідний образ, що визначає ступені порушення нейронів самого нижнього рівня ієрархії, у вихідний образ, обумовлений нейронами самого верхнього рівня. Число останніх, звичайно, порівняно невелике. Стан порушення нейрона на верхньому рівні говорить про приналежність вхідного образа до тієї або іншої категорії.

Традиційно розглядається аналогова логіка, при якій припустимі стани синаптичних зв'язків визначаються довільними дійсними числами, а ступені активності нейронів - дійсними числами між 0 й 1. Іноді досліджуються також моделі з дискретною арифметикою, у якій синапс характеризується двома булевими змінними: активністю (0 або 1) і полярністю (-1 або +1), що відповідає тризначній логіці. Стану нейронів можуть при цьому описуватися однієї булевої змінної. Даний дискретний підхід робить конфігураційний простір станів нейронної мережі кінцевим (не говорячи вже про переваги при апаратній реалізації).

**Навчання методом зворотного поширення помилок**

Для навчання багатошарової мережі в 1986 р. Руммельхартом і Хинтоном (Rummelhart D.E., Hіnton G.E., Wіllіams R.J., 1986) був запропонований алгоритм зворотного поширення помилок (error back propagatіon). Численні публікації про промислові застосування багатошарових мереж із цим алгоритмом навчання підтвердили його принципову працездатність на практиці.

На початку виникає резонне питання - а чому для навчання багатошарового персептрона не можна застосувати вже відоме  -правило Розенблатта? Відповідь полягає в тому, що для застосування методу Розенблатта необхідно знати не тільки поточні виходи нейронів y, але й необхідні *правильні* значення **Y**. У випадку багатошарової мережі ці правильні значення є тільки для нейронів *вихідного* шару. Необхідні значення виходів для нейронів схованих шарів невідомі, що й обмежує застосування  -правила.

Основна ідея зворотного поширення полягає в тому, як одержати оцінку помилки для нейронів схованих шарів. Помітимо, що *відомі* помилки, що роблять нейронами вихідного шару, виникають внаслідок *невідомих* поки помилок нейронів схованих шарів. Чим більше значення синаптичного зв'язку між нейроном схованого шару й вихідним нейроном, тим сильніше помилка першого впливає на помилку другого. Отже, оцінку помилки елементів схованих шарів можна одержати, як зважену суму помилок наступних шарів. При навчанні інформація поширюється від нижчих шарів ієрархії до вищих, а оцінки помилок, що робляться мережею - у зворотному напрямку, що й відбито в назві методу.

Перейдемо до докладного розгляду цього алгоритму. Для спрощення позначень обмежимося ситуацією, коли мережа має тільки один схований шар. Матрицю вагових коефіцієнтів від входів до схованого шару позначимо W, а матрицю ваг, що з'єднують схований і вихідний шар - як V. Для індексів приймемо наступні позначення: входи будемо нумерувати тільки індексом і, елементи схованого шару - індексом j, а виходи, відповідно, індексом k.

Нехай мережа навчається на вибірці (X,Y),  =1..p. Активності нейронів будемо позначати малими буквами y з відповідним індексом, а сумарні зважені входи нейронів - малими буквами x.

Загальна структура алгоритму зворотного поширення помилки:

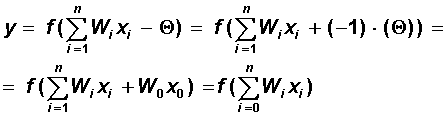
|  |  |
| --- | --- |
| Крок 0. | Початкові значення ваг всіх нейронів всіх шарів V(t=0) і W(t=0) покладаються випадковими числами. |
| Крок 1. | Мережі пред'являється вхідний образ X,, у результаті формується вихідний образ y Y. При цьому нейрони послідовно від шару до шару функціонують по наступних формулах:  схований шар  Image29  вихідний шар  Image30  Де f(x) - сигмоідальная функція, обумовлена по формулі (5.1) |
| Крок 2. | Функціонал квадратичної помилки мережі для даного вхідного образа має вигляд:  Image31  Даний функціонал підлягає мінімізації. Класичний градієнтний метод оптимізації складається в ітераційному уточненні аргументу відповідно до формули:  Image32  Функція помилки в явному виді не містить залежності від ваги Vjk, тому скористаємося формулами неявного диференціювання складної функції:  Image33  Тут враховане корисна властивість сигмоідальной функції f(x): її похідна виражається тільки через саме значення функції, f'(x)=f(1-f). Таким чином, всі необхідні величини для підстроювання ваг вихідного шару V отримані. |
| Крок 3. | На цьому кроці виконується підстроювання ваг схованого шару. Градієнтний метод як і раніше дає:  Image34  Обчислення похідних виконуються по тих же формулам, за винятком деякого ускладнення формули для помилки  j.  Image35  При обчисленні  j тут і був застосований принцип зворотного поширення помилки: частки похідні беруться тільки по змінним *наступного* шару. По отриманих формулах модифікуються ваги нейронів схованого шару. Якщо в нейронній мережі є кілька схованих шарів, процедура зворотного поширення застосовується послідовно для кожного з них, починаючи із шару, що передує вихідному, і далі до шару, що випливає за вхідним. При цьому формули зберігають свій вид із заміною елементів вихідного шару на елементи відповідного схованого шару. |
| Крок 4. | Кроки 1-3 повторюються для всіх навчальних векторів. Навчання завершується по досягненні малої повної помилки або максимально припустимого числа ітерацій, як й у методі навчання Розенблатта. |

Як видно з опису кроків 2-3, навчання зводиться до рішення задачі оптимізації функціонала помилки градієнтним методом. Вся “сіль” зворотного поширення помилки полягає в тому, що для її оцінки для нейронів схованих шарів можна прийняти зважену суму помилок наступного шару.

Параметр h має сенс темпу навчання й вибирається досить малим для збіжності методу. Про збіжність необхідно зробити кілька додаткових зауважень. По-перше, практика показує що збіжність методу зворотного поширення досить повільна. Невисокий темп збіжності є “генетичною хворобою” всіх градієнтних методів, тому що локальний напрямок градієнта аж ніяк не збігається з напрямком до мінімуму. По-друге, підстроювання ваг виконується незалежно для кожної пари образів навчальної вибірки. При цьому поліпшення функціонування на деякій заданій парі може, загалом кажучи, приводити до погіршення роботи на попередніх образах. У цьому змісті, *немає* достовірних (крім досить великої практики застосування методу) гарантій збіжності.

Дослідження показують, що для подання довільного функціонального відображення, що задається навчальною вибіркою, досить усього *два шари* нейронів. Однак на практиці, у випадку складних функцій, використання більш ніж одного схованого шару може давати економію повного числа нейронів.

У завершення лекції зробимо зауваження щодо настроювання порогів нейронів. Легко помітити, що поріг нейрона може бути зроблений еквівалентним додатковій вазі, з'єднаному з фіктивним входом, рівним -1. Дійсно, вибираючи W0=, x0=-1 і починаючи підсумовування з нуля, можна розглядати нейрон з нульовим порогом й одним додатковим входом:



Додаткові входи нейронів, що відповідають порогам, зображені на Рис. 1 темними квадратиками. З урахуванням цього зауваження, всі викладені в алгоритмі зворотного поширення формули підсумовування по входах починаються з нульового індексу.