

Міністерство освіти і науки України
Сумський державний університет

А. С. Довбиш, І. В. Шелехов

ОСНОВИ ТЕОРІЇ РОЗПІЗНАВАННЯ ОБРАЗІВ

Навчальний посібник

У двох частинах

Частина 1

Рекомендовано вченою радою Сумського державного університету



Суми
Сумський державний університет
2015

УДК 004.93'1(072)

ББК 32.81я7

Д58

Рецензенти:

С. Ф. Теленик – доктор технічних наук, професор (Національний технічний університет України «Київський політехнічний інститут»);

Є. П. Путьатін – доктор технічних наук, професор (Харківський національний університет радіоелектроніки)

*Рекомендовано до видання вченою радою
Сумського державного університету як навчальний посібник
(протокол № 3 від 12 листопада 2015 року)*

Довбиш А. С.

Д58 Основи теорії розпізнавання образів : навч. посіб. : у 2 ч. /
А. С. Довбиш, І. В. Шелехов. – Суми : Сумський державний
університет, 2015. – Ч. 1. – 109 с.

ISBN 987-966-657-596-1

ISBN 987-966-657-597-8 (частина 1)

Метою цього навчального посібника є ознайомлення студентів з основними класичними та сучасними методами розпізнавання образів. При цьому особливу увагу приділено формуванню інтуїтивного розуміння теоретичних основ кожного з методів і особливостей їх практичного застосування. Це дозволяє студентові одержати знання, достатні для самостійної роботи у сфері інтелектуального аналізу даних.

УДК 004.93'1(072)

ББК 32.81я7

ISBN 987-966-657-596-1

© Довбиш А. С., Шелехов І. В., 2015

ISBN 987-966-657-597-8 (частина 1)

© Сумський державний університет, 2015

Зміст

	С.
Вступ	4
1 Системологія розпізнавання образів	5
1.1 Основні визначення, положення та принципи теорії розпізнавання образів	5
1.2 Постановка задачі інформаційного синтезу систем розпізнавання образів	13
1.3 Вирішальні правила	15
1.4 Критерії якості розпізнавання образів	26
Контрольні запитання та завдання	33
2 Детерміновані методи розпізнавання образів	37
2.1 Метод еталонів	37
2.2 Метод еталонів, що дробляться	46
2.3 Лінійні вирішальні правила	54
2.4 Метод К-найближчих сусідів	62
2.5 Метод потенціальних функцій	68
2.6 Структурні методи	71
2.7 Метод опорних векторів	73
2.8 Детерміновані методи автоматичної класифікації	91
Контрольні запитання та завдання	101
Список літератури	105

Вступ

Навчальний посібник створено на основі багаторічного досвіду, накопиченого авторами при викладанні навчальної дисципліни «Теорія розпізнавання образів» магістрам спеціальності «Комп'ютерні науки та інформаційні технології» в Сумському державному університеті.

Мета, яку автори намагалися реалізувати під час написання посібника, полягала в ознайомленні студентів не лише з основними класичними та сучасними методами розпізнавання образів, а й із перспективами і тенденціями їх розвитку. Досягнення поставленої мети здійснювалося шляхом порівняльного аналізу методів розпізнавання образів, виявлення їх переваг і недоліків. При цьому основний акцент у навчальному посібнику зроблено на аналізі та синтезі систем розпізнавання образів, побудованих на ідеях і методах машинного навчання та самонавчання, які найбільшою мірою моделюють механізм і функції когнітивних процесів, властивих людині під час прийняття рішень.

У посібнику не розглядаються методи синтезу штучних нейронних мереж і нечіткі методи подання та виведення знань, оскільки вони вивчаються в навчальній дисципліні бакалаврського рівня «Інтелектуальні системи».

Навчальний посібник складається з двох частин. У першій частині розглядаються визначення, принципи, системологічні положення і основні детерміновані методи розпізнавання образів. Загальне редагування, підрозділи 1.1 і 1.4 виконано Довбишем А.С.

Системологія розпізнавання образів

1

1.1 Основні визначення, положення та принципи теорії розпізнавання образів

Теорія розпізнавання образів (*Pattern recognition*) – розділ кібернетики, присвячений створенню теоретичних основ класифікації та ідентифікації предметів, явищ, процесів, сигналів, ситуацій тощо, які будемо узагальнено називати об'єктами, а процеси класифікації та ідентифікації – розпізнаванням.

Здатність до розпізнавання є важливою особливістю усіх живих організмів. Як відомо, в процесі набуття досвіду прийняття рішень у правій півкулі головного мозку людини формуються когнітивні (понятійні) образи, які разом із методами їх розпізнавання утворюють за термінологією комп'ютерних наук базу знань. При цьому ліва півкуля виконує функції операційної системи. Таким чином, як доведено за результатами експериментальних досліджень будь-яка діяльність біонічної системи здійснюється на основі образного мислення [1, 2].

Визначення 1.1.1 Образом (*Pattern*) називається модель, яка відбиває властивості об'єкта, що розпізнається.

Образ характеризується множиною ознак розпізнавання, які утворюють структурований вектор-реалізацію образу.

Визначення 1.1.2 Ознакою (*Feature*) розпізнавання називається характеристика певної властивості об'єкта, що аналізується.

Визначення 1.1.3 Вектором-реалізацією образу називається структурована, тобто впорядкована, послідовність ознак розпізнавання, яку будемо подавати у вигляді вектора-рядка.

$$\mathbf{x} = (x_1, x_2, \dots, x_N)$$

або вектора-стовпця

$$\mathbf{x} = (x_1, x_2, \dots, x_N)^T.$$

Далі замість терміна «вектор-реалізація образу» часто будемо вживати термін «реалізація образу», а для позначення належності реалізації \mathbf{x} до класу розпізнавання X_m^o будемо використовувати запис $\mathbf{x} \in X_m^o$, або

$$\mathbf{x}_m = (x_{m,1}, \dots, x_{m,i}, \dots, x_{m,N}).$$

Крім того, в теорії розпізнавання часто замість терміна «образ» вживають термін «клас розпізнавання». Наприклад, класом розпізнавання можна вважати множину зірок, що належать до одного сузір'я, або множину студентів, які показали певний рівень знань під час складання сесії. Якщо розглянути групи студентів з оцінками «задовільно» та «добре», різниця між якими досить розмита і визначається кожним викладачем суб'єктивно, то виникає поняття нечітких класів розпізнавання. Такі класи здатні перетинатися в просторі ознак розпізнавання, а їх реалізації

характеризуються певним ступенем належності до декількох класів розпізнавання.

Із початком комп'ютерної ери теорія розпізнавання образів набула свого стрімкого розвитку, результатом якого стало широке використання в усіх галузях соціально-економічної сфери суспільства систем розпізнавання різного призначення [3-5].

Визначення 1.1.4 Система розпізнавання образів – електронно-обчислювальний комплекс, здатний моделювати розумові процеси, властиві людині під час прийняття рішень із метою виявлення аналогій серед досліджуваних об'єктів.

Для розпізнавання образів необхідно розв'язати дві основні задачі: розбиття простору ознак розпізнавання на області, що відповідають певному класу об'єктів, та визначення належності реалізації образу, що розпізнається, до відповідного класу. Найчастіше система розпізнавання образів за аналогією з біонічною системою функціонує в двох режимах: навчання й екзамену. На етапі навчання з метою побудови вирішальних правил здійснюється розбиття простору ознак на класи розпізнавання.

Визначення 1.1.5 Вирішальним правилом (або класифікатором) називається математичний вираз або алгоритм визначення належності реалізації образу, що розпізнається, одному з класів розпізнавання із заданого алфавіту.

Як приклад розглянемо задачу розподілу символів на два класи: клас X_1^o – символи, видовжені вертикально (рис. 1.1.1 а), і клас X_2^o – символи, видовжені горизонтально (рис. 1.1.1 б). Для кожного символу необхідно визначити його належність до одного з цих класів.

А Е І О У Ж М Ф Ш Щ

а

б

Рисунок 1.1.1 – Графічне зображення реалізацій класів розпізнавання: а – символи, видовжені вертикально (клас X_1^o); б – символи, видовжені вертикально (клас X_2^o)

Під час формування простору ознак розпізнавання виконують інформаційний аналіз характеристик символів. Для кожної з характеристик оцінюють її *релевантність* – ступінь відповідності до поставленої задачі, та *інформативність* – вплив майбутньої ознаки на ефективність вирішальних правил. Наприклад, те, що символи класу X_1^o складаються з голосних літер, а класу X_2^o – з приголосних, не має відношення до поставленої задачі, тобто відповідна ознака не буде релевантною.

На рисунку 1.1.2 показано графічне зображення основних геометричних характеристик символу «А», який є однією з реалізацій класу X_1^o .

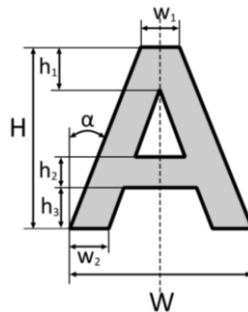


Рисунок 1.1.2 – Графічне зображення геометричних ознак розпізнавання символу

Аналіз рис. 1.1.2 засвідчує, що для кожного символу можна сформувати власну множину геометричних характеристик.

Наявність неточностей та шумів (*noise*) під час вимірювання значень ознак розпізнавання обумовлює необхідність їх попередньої обробки (*preprocessing*). При цьому виконується як фільтрація, метою якої є видалення шумів, так і відновлення значень ознак. Окремою групою методів попередньої обробки є сегментація, яка полягає у відокремленні об'єктів, що розпізнаються, від фону. Наприклад, у системах відеоспостереження при автоматичній ідентифікації особи попередня обробка полягає в пошуку та виділенні обличчя або його частини на зображенні зі змінним фоном [6].

Відомо, що в рамках синтаксичного підходу корисність інформації визначається її кількістю, тобто чим більший обсяг має реалізація образу, тим вона корисніша. Тому на етапі апріорного проектування системи розпізнавання, якщо відсутній її експериментальний зразок, розробники намагаються при формуванні вхідного інформаційного опису системи враховувати найбільшу кількість ознак різної природи.

Позначимо простір ознак розпізнавання як Ω^M , де N – вимірність простору, що визначається кількістю ознак. Тоді сформований на етапі апріорного проектування словник ознак розпізнавання здебільшого буде надлишковим. Можливість оптимізації (тут і далі в інформаційному розумінні) словника ознак та інших параметрів функціонування виникає на етапі апостеріорного проектування, коли за результатами фізичного моделювання

системи розпізнавання здійснюється максимізація її функціональної ефективності.

При функціонуванні системи розпізнавання в режимі навчання формування оптимальних вирішальних правил виконується шляхом аналізу вхідних даних, які подаються у вигляді навчальної матриці.

Визначення 1.1.5 Репрезентативна множина випадкових реалізацій образу, що використовується на етапі навчання системи розпізнавання, називається навчальною матрицею (*training set*).

Наприклад, навчальна матриця типу «об’єкт-властивість» для класу розпізнавання X_m^o може бути подана відповідно до формату вектора-реалізації образу у вигляді двовимірного масиву розмірністю $N \times n$:

$$\| \mathbf{x}_m^{(j)} \| = \left\| \begin{array}{c} \mathbf{x}_m^{(1)} \\ \mathbf{x}_m^{(2)} \\ \dots \\ \mathbf{x}_m^{(j)} \\ \dots \\ \mathbf{x}_m^{(n)} \end{array} \right\| = \left\| \begin{array}{cccc} x_{m,1}^{(1)} & x_{m,2}^{(1)} & \dots & x_{m,N}^{(1)} \\ x_{m,1}^{(2)} & x_{m,2}^{(2)} & \dots & x_{m,N}^{(2)} \\ & & \dots & \\ x_{m,1}^{(j)} & x_{m,2}^{(j)} & \dots & x_{m,N}^{(j)} \\ & & \dots & \\ x_{m,1}^{(n)} & x_{m,2}^{(n)} & \dots & x_{m,N}^{(n)} \end{array} \right\| \quad (1.1.1)$$

або розмірністю $n \times N$

$$\| \mathbf{x}_m^{(j)} \| = \| \mathbf{x}_m^{(1)}, \mathbf{x}_m^{(2)} \dots \mathbf{x}_m^{(j)} \dots \mathbf{x}_m^{(n)} \| = \left\| \begin{array}{cccc} x_{m,1}^{(1)} & x_{m,1}^{(2)} & \dots & x_{m,1}^{(j)} \dots x_{k,1}^{(n)} \\ x_{m,2}^{(1)} & x_{m,2}^{(2)} & \dots & x_{m,2}^{(j)} \dots x_{k,2}^{(n)} \\ & & \dots & \\ x_{m,N}^{(1)} & x_{m,N}^{(2)} & \dots & x_{m,N}^{(j)} \dots x_{k,N}^{(n)} \end{array} \right\|, \quad (1.1.2)$$

де N – кількість ознак розпізнавання; n – кількість реалізацій.

Таким чином, навчальна матриця для алфавіту із M класів розпізнавання складається з M двовимірних масивів (1.1.1) або (1.1.2).

Важливим розділом теорії розпізнавання образів є автоматична класифікація або кластер-аналіз вхідних даних [7-9]. Кластер складається з множини схожих (подібних) реалізацій образу, які можна відділити за певними критеріями від інших об'єктів. Оскільки при кластер-аналізі масив ідентифікаторів класу реалізацій відсутній, то іншою назвою цього процесу є «навчання без учителя» (*unsupervised learning*). У процесі кластеризації виконується не лише пошук подібних реалізацій, але й формування вирішальних правил для кожного кластера. Априорна інформація про кількість кластерів та їх розміщення в просторі ознак розпізнавання значно спрощує цей процес.

Основними підходами, що відрізняються галуззю знань і науковим напрямком вирішення проблем розпізнавання образів, є:

1) алгебраїчний, основною перевагою якого є прості вирішальні правила.

Основний недолік цього підходу полягає в невисокій достовірності розпізнавання, оскільки він не враховує неконтрольовані фактори, які впливають на процес розпізнавання;

2) геометричний, що характеризується універсальністю, наочністю та простотою інтерпретації алгоритмів розпізнавання;

3) статистичний, у рамках якого використовують статистичні характеристики аналізу даних;

4) біологічний, до якого відносять штучні нейронні мережі. Алгоритми в рамках цього підходу моделюють когнітивні процеси, що відбуваються у нервових клітинах мозку людини.

Основний недолік біологічного підходу – це висока чутливість до багатовимірності простору ознак розпізнавання;

5) мережевий (семантичні мережі, фрейми, мережі Петрі, дерево рішень тощо).

Перевагами цього підходу є простота моделі, можливість її розширення та ускладнення, а основний недолік – складність побудови вирішальних правил;

6) нечіткий, який створено на базі алгебраїчного підходу, і він є конкурентом статистичного підходу. Цей підхід дозволяє моделювати процеси розпізнавання образів, що апріорно перетинаються в просторі ознак розпізнавання. Але він не пристосований до оптимізації параметрів функціонування системи розпізнавання;

7) теоретико-ігровий підхід, вирішальні правила у рамках якого характеризуються високою складністю та невисокою достовірністю розпізнавання.

Незважаючи на те що наведені підходи відрізняються один від одного рівнем і видом математичної формалізації слабоформалізованих процесів прийняття рішень, між ними не існує чіткої межі, а самі підходи часто доповнюють один одного. Оскільки всі основні підходи, окрім алгебраїчного, перетинаються з геометричним, то саме в рамках

геометричного підходу формування загальної теорії прийняття рішень є найбільш виправданим.

В рамках геометричного підходу теорія розпізнавання образів базується на двох основних принципах [10, 11]:

- *максимально-дистанційний принцип*, за яким вирішальні правила будуються шляхом максимізації середньої міжкласової відстані;

- *мінімально-дистанційний принцип*, за яким вирішальні правила будуються за умови мінімізації середньої відстані реалізації образу до центру свого класу.

Виконання цих принципів є необхідною умовою одержання максимальної достовірності розпізнавання, яка визначається повною ймовірністю правильного прийняття рішень

$$P_i = p_1 D_1 = p_2 D_2,$$

де p_1, p_2 - безумовні ймовірності, D_1, D_2 - перша та друга достовірності відповідно.

Таким чином, історія становлення теорії розпізнавання образів характеризується великою різноманітністю методів і підходів до вирішення її проблем.

1.2 Постановка задачі інформаційного синтезу систем розпізнавання образів

Для сприйняття образу необхідно виділити множину релевантних інформативних ознак та, вимірявши їх значення, сформуванати вектор-реалізацію образу. При цьому на практиці, як правило, виникає необхідність використання

додаткових процедур попередньої обробки, фільтрації та відновлення образів.

Важливу роль під час розв'язання задачі інформаційного синтезу відіграє апріорна інформація про об'єкти розпізнавання, еобхідна для формування алфавіту класів розпізнавання, словника ознак і навчальної матриці. Крім того, для побудови високодостовірних вирішальних правил на етапі формування вхідного математичного опису системи розпізнавання необхідно здійснювати так званий розвідувальний аналіз із метою забезпечення статистичної стійкості й однорідності реалізацій образу [12-14].

Методи розпізнавання поділяють на дві основні групи:

1) методи, які для формування вирішальних правил використовують класифіковану навчальну матрицю (методи навчання з учителем);

2) методи автоматичної класифікації, здатні обробляти некласифіковані дані (методи навчання без учителя).

Всередині кожної з цих основних груп виконується подальший поділ методів на підгрупи залежно від апріорної інформації, необхідної для їх застосування.

Розглянемо в рамках геометричного підходу узагальнену постановку задачі інформаційного синтезу системи розпізнавання образів, здатної навчатися [15].

Нехай дано алфавіт класів розпізнавання $\{X_m^o | m = 1..M\}$, де M – потужність алфавіту.

На етапі навчання необхідно:

1) побудувати деяким оптимальним (тут і далі в інформаційному розумінні) способом розбиття \mathfrak{R} простору ознак $\Omega^{|M|}$ на класи розпізнавання, що відповідає умовам:

$$\begin{aligned}
& (\forall X_m^o \in \mathfrak{R})[X_m^o \neq \emptyset], \\
& (\forall X_k^o \in \mathfrak{R})(\forall X_l^o \in \mathfrak{R})[X_k^o \neq X_l^o \rightarrow X_k^o \cap X_l^o = \emptyset], \\
& \cup X_m^o \in \Omega^{|N|}, k, l, m = 1..M; \quad (1.2.1)
\end{aligned}$$

2) за геометричними параметрами побудованого оптимального розбиття \mathfrak{R}^* простору ознак на класи розпізнавання побудувати безпомилкові за навчальною матрицею вирішальні правила.

На етапі екзамену, тобто безпосереднього розпізнавання, прийняти високодостовірне рішення про належність реалізації образу, що розпізнається, одному з класів заданого алфавіту.

Таким чином, у топологічному розумінні класом розпізнавання є область у просторі ознак розпізнавання, елементи якої еквівалентно співвідносяться.

1.3 Вирішальні правила

Вирішальні правила необхідні для прийняття рішень про належність образу, що розпізнається, одному з класів із заданого алфавіту. Вони можуть будуватися з використанням математичних формул, геометричних об'єктів, лінгвістичних структур тощо.

Розглянемо етапи побудови вирішальних правил у рамках геометричного підходу, що характеризується найбільшою універсальністю та наочністю:

- 1) вимірювання ознак розпізнавання;
- 2) нормалізація значень ознак розпізнавання, яка полягає у зведенні значень до вигляду, зручного для оброблення на

ЕОМ; формування на кожному спостереженні об'єкта векторів-реалізацій образів;

3) формування навчальної матриці;

4) допустимі перетворення навчальної матриці в субпарацептуальному просторі;

5) розбиття деяким оптимальним способом простору ознак на класи розпізнавання;

б) за геометричними параметрами оптимального (тут і далі в інформаційному розумінні) розбиття простору ознак на класи розпізнавання будуються вирішальні правила.

Таким чином, для побудови вирішальних правил (або класифікатора) необхідно оптимізувати параметри функціонування системи розпізнавання.

Визначення 1.3.1 Параметрами функціонування називаються характеристики системи розпізнавання, які впливають на її функціональну ефективність.

У системах розпізнавання, що навчаються, параметри їх функціонування часто називаються параметрами навчання.

Розглянемо декілька прикладів формування вирішальних правил. Нехай X_1^o і X_2^o – два класи розпізнавання, розподіл реалізацій яких показано на рис. 1.3.1.

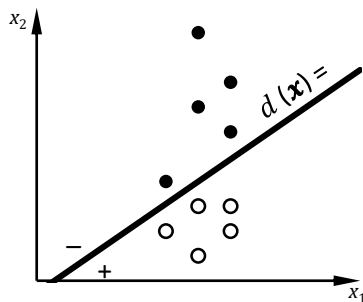


Рисунок 1.3.1 – Розподіл реалізацій двох класів розпізнавання

Кожна реалізація є вектором ознак розпізнавання $\mathbf{x} = (x_1, x_2)^T$, заданим у просторі ознак $x_1 - x_2$. На рисунку 1.3.1 реалізації, що належать до класу X_1^o , позначено світлими кружочками, а які належать до класу X_2^o – темними.

Реалізації показаних на рис. 1.3.1 класів розпізнавання можна розділити прямою:

$$d(\mathbf{x}) = w_0 + w_1x_1 + w_2x_2 = 0. \quad (1.3.1)$$

При цьому значення коефіцієнтів w_0 , w_1 та w_2 визначено так, що для всіх реалізацій класу X_1^o виконується нерівність $d(\mathbf{x}) > 0$, а для всіх реалізацій класу X_2^o – $d(\mathbf{x}) < 0$. Крім того, для будь-якої реалізації \mathbf{x} , для якої *априорно* відомо, що вона належить одному з класів X_1^o або X_2^o , можна обчислити значення $d(\mathbf{x})$ і прийняти рішення $\mathbf{x} \in X_1^o$, якщо $d(\mathbf{x}) > 0$, або $\mathbf{x} \in X_2^o$, якщо $d(\mathbf{x}) < 0$. Сформовану в такий спосіб роздільну функцію $d(\mathbf{x})$ називають лінійним вирішальним правилом класу X_1^o .

Для N -вимірною випадку $\mathbf{x} = (x_1, x_2, \dots, x_N)^T$ пряма (1.3.1) перетворюється на гіперплощину (1.3.2):

$$d(\mathbf{x}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_Nx_N = w_0 + \mathbf{w}^T\mathbf{x} = 0, \quad (1.3.2)$$

де $\mathbf{w} = (w_1, w_2, \dots, w_N)$ – вектор коефіцієнтів, до значень яких ставляться вимоги, аналогічні до випадку (1.3.1).

Зазвичай вектор-реалізація \mathbf{x} розширюється до

$$\mathbf{x} = (1, x_1, x_2, \dots, x_N)^T,$$

а вектор коефіцієнтів – до

$$\mathbf{w} = (w_0, w_1, w_2, \dots, w_N).$$

Тоді лінійні вирішальні правила (1.3.1) та (1.3.2) набувають вигляду:

$$d(\mathbf{x}) = \mathbf{w}^T \mathbf{x} = 0. \quad (1.3.3)$$

Визначення 1.3.2 Нехай множина $\{X_m^o | m = 1, M\}$ – алфавіт класів розпізнавання. Якщо функція $d(\mathbf{x}) = 0$, де

$$\mathbf{x} = (1, x_1, x_2, \dots, x_N)^T,$$

є роздільною для класу X_i^o , тобто

$$d_i(\mathbf{x}) = \begin{cases} > 0, & \mathbf{x} \in X_i^o, \\ < 0, & \mathbf{x} \in X_j^o, j \neq i, \end{cases} \quad (1.3.4)$$

то така функція називається *вирішальним правилом* для класу X_i^o .

Якщо для кожного класу X_i^o з алфавіту класів розпізнавання можна сформулювати лінійне вирішальне правило, тобто

$$d_i(\mathbf{x}) = \mathbf{w}^T \mathbf{x} = \begin{cases} > 0, & \mathbf{x} \in X_i^o, \\ < 0, & \mathbf{x} \in X_j^o, j \neq i, \end{cases}$$

то має місце абсолютна лінійна роздільність класів.

На рисунку 1.3.3 подано приклад абсолютної лінійної роздільності трьох класів розпізнавання, реалізації яких позначено відповідно: $\circ (\mathbf{x} \in X_1^o)$, $\bullet (\mathbf{x} \in X_2^o)$ та $\Delta (\mathbf{x} \in X_3^o)$.

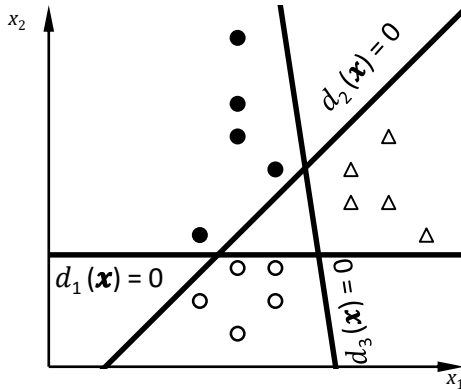


Рисунок 1.3.3 – Приклад абсолютної лінійної роздільності класів розпізнавання

При цьому для визначення належності будь-якої реалізації \mathbf{x} , для якої апріорно відомо, що вона належить одному з класів заданого алфавіту, можна обчислити значення $d_1(\mathbf{x})$, $d_2(\mathbf{x})$, $d_3(\mathbf{x})$ і прийняти рішення $\mathbf{x} \in X_1^o$, якщо $d_1(\mathbf{x}) > 0$, $d_2(\mathbf{x}) < 0$, $d_3(\mathbf{x}) < 0$; $\mathbf{x} \in X_2^o$, якщо $d_1(\mathbf{x}) < 0$, $d_2(\mathbf{x}) > 0$, $d_3(\mathbf{x}) < 0$, та $\mathbf{x} \in X_3^o$, якщо $d_1(\mathbf{x}) < 0$, $d_2(\mathbf{x}) < 0$, $d_3(\mathbf{x}) > 0$.

Визначення 1.3.3 Нехай для класів розпізнавання $\{X_m^o | m = 1, M\}$ має місце абсолютна лінійна роздільність та сформовані відповідні лінійні вирішальні правила $d_1(\mathbf{x}) = 0$, $d_2(\mathbf{x}) = 0, \dots, d_m(\mathbf{x}) = 0$, тоді множина реалізацій

$$D_i(\mathbf{x}) = \{\mathbf{x} | d_i(\mathbf{x}) > 0, d_j(\mathbf{x}) < 0, i \neq j\}, 1 \leq j \leq m, \quad (1.3.5)$$

називається *областю рішень* для класу X_i^o , $1 \leq i \leq m$.

На рисунку 1.3.4 показано область рішень для класів, наведених на рисунку 1.3.3.

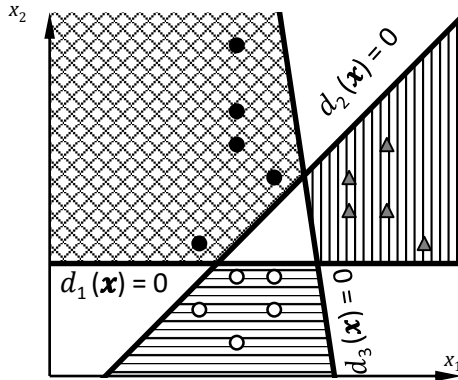


Рисунок 1.3.4 – Приклад області рішень при абсолютній лінійній роздільності класів розпізнавання

На рисунку 1.3.4 область рішень класу X_1^o позначена горизонтальним штрихуванням, класу X_2^o – подвійним штрихуванням, класу X_3^o – вертикальним штрихуванням. При абсолютній лінійній роздільності класів розпізнавання такі області на відміну від класів можуть не мати загальних меж.

Лінійні вирішальні правила застосовуються й тоді, коли абсолютна лінійна роздільність класів відсутня. При цьому розглядається випадок попарної лінійної роздільності класів. Вирішальні правила (1.3.3) формуються для кожної пари класів X_i^o і X_j^o у вигляді функцій $d_{i,j}(x) = 0$:

$$d_{k,l}(x) = \mathbf{w}^T \mathbf{x} = \begin{cases} d_{k,l}(x) > 0, & \mathbf{x} \in X_k^o; \\ d_{k,l}(x) < 0, & \mathbf{x} \in X_l^o. \end{cases} \quad (1.3.6)$$

У результаті кожний клас X_i^o характеризується множиною з $m-1$ лінійного вирішального правила вигляду

(1.3.6). При цьому для будь-яких двох класів X_i^o та X_j^o виконується рівність

$$d_{i,j}(\mathbf{x}) = -d_{j,i}(\mathbf{x}).$$

Для визначення належності будь-якої реалізації \mathbf{x} до класу X_i^o необхідно обчислити значення

$$d_{i,1}(\mathbf{x}), d_{i,2}(\mathbf{x}), \dots, d_{i,i-1}(\mathbf{x}), \dots, d_{i,i+1}(\mathbf{x}), \dots, d_{i,m}(\mathbf{x})$$

і перевірити виконання всіх $m-1$ умов:

$$d_{i,j}(\mathbf{x}) > 0, \text{ для всіх } i \neq j.$$

При цьому зміниться й визначення області рішень.

Визначення 1.3.4 Нехай для класів розпізнавання $\{X_m^o | m = 1, M\}$ мають місце попарна лінійна роздільність та сформовані відповідні лінійні вирішальні правила $\{d_{ij}(\mathbf{x}) = 0\}$, тоді множина реалізацій

$$D_i(\mathbf{x}) = \{\mathbf{x} | d_{ij}(\mathbf{x}) > 0, i \neq j\}, 1 \leq j \leq m, \quad (1.3.7)$$

називається *областю рішень* для класу X_i^o , $1 \leq i \leq m$.

На рисунку 1.3.5 *a* подано приклад попарної лінійної роздільності трьох класів розпізнавання: \circ ($\mathbf{x} \in X_1^o$), \bullet ($\mathbf{x} \in X_2^o$) та Δ ($\mathbf{x} \in X_3^o$), а на рисунку 1.3.5 *б* – відповідних областей рішень.

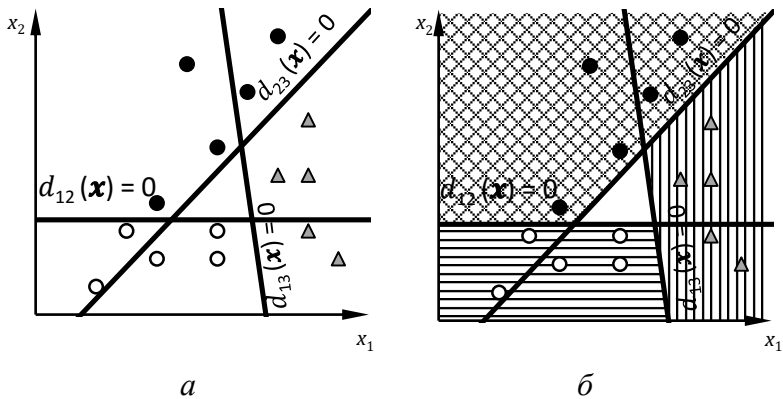


Рисунок 1.3.5 – Попарна лінійна роздільність класів розпізнавання

Аналіз рис. 1.3.4 та 1.3.5 б засвідчує, що як при абсолютній, так і при попарній роздільності об'єднання областей рішень не покриває всього простіру ознак. Таким чином, належність частини реалізацій не може бути визначена, тобто існують такі реалізації, що не належать жодному з класів розпізнавання.

Лінійні вирішальні правила (1.3.6), що використовуються при попарній роздільності класів X_i^o , можна сформулювати й на базі аналогічних правил (1.3.4), що використовуються при абсолютній роздільності. Нехай $d_1(x) = 0$, $d_2(x) = 0$, ..., $d_m(x) = 0$ – лінійні вирішальні правила типу (1.3.4), тоді відповідні їм правила типу (1.3.6) мають вигляд

$$d_{i,j}(x) = d_i(x) - d_j(x), \quad 1 \leq i, j \leq m. \quad (1.3.8)$$

На рисунку 1.3.6 подано приклад областей рішень, що сформованих за виразом (1.3.8) із використанням лінійних вирішальних правил, показаних на рис. 1.3.3.

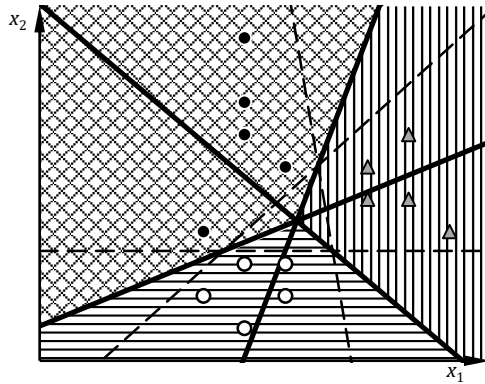


Рисунок 1.3.6 – Перетворення лінійних вирішальних правил

Аналіз рис. 1.3.6 показує, що після перетворення вирішальних правил типу (1.3.4) (штриховані лінії) на правила типу (1.3.6) (неперервні лінії) об'єднання областей рішень усіх класів розпізнавання повністю збігається з простором ознак розпізнавання.

За реальних умов лінійні вирішальні правила використовуються досить рідко через відсутність як абсолютної, так і попарної лінійної роздільності класів розпізнавання.

Вирішальні правила можуть бути й нелінійними, як це показано на рис. 1.3.7.

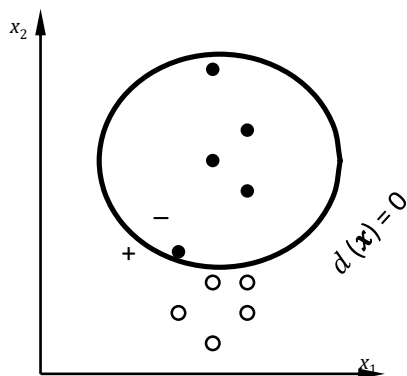


Рисунок 1.3.7 – Нелінійне вирішальне правило для двох класів розпізнавання

На рисунку 1.3.7 два класи розділяються кривою другого порядку:

$$d(\mathbf{x}) = 1 - \frac{(x_1 - w_{11})^2}{w_{12}^2} - \frac{(x_2 - w_{21})^2}{w_{22}^2} = 0.$$

Для випадків нелінійної роздільності поняття вирішальної функції (1.3.2) узагальнюють до вигляду

$$d(\mathbf{x}) = w_0 + w_1 f(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \dots + w_K f_K(\mathbf{x}) = 0, \quad (1.3.9)$$

де $(f_i(\mathbf{x}))$ – послідовність нелінійних або лінійних функцій, $1 \leq i \leq K$.

При використанні розширеної форми векторів-реалізацій, векторів-коефіцієнтів (1.3.3) та розширення $(f_i(\mathbf{x}))$ функцією $f_0(\mathbf{x}) = 1$ вираз (1.3.9) набуває вигляду

$$d(\mathbf{x}) = \sum_{i=0}^K w_i f_i(\mathbf{x}) = 0. \quad (1.3.10)$$

Зазвичай під час формування нелінійних вирішальних правил використовують поліноміальний вигляд функцій $(f_i(\mathbf{x}))$. Наприклад, якщо $f_i(\mathbf{x})$ обмежені поліномами не вище другого степеня ($t \leq 2$), а розмірність простору ознак $N = 2$, то (1.3.10) перетворюється на

$$\begin{aligned} d^{(N,t)}(\mathbf{x}) &= d^{(2,2)}(\mathbf{x}) = \\ &= w_0 + w_1x_1 + w_2x_2 + w_3x_1^2 + w_4x_1x_2 + w_5x_2^2 = 0, \end{aligned}$$

тобто

$$\begin{aligned} \mathbf{w} &= (w_0, w_1, w_2, \dots, w_5), \\ (f_i(\mathbf{x})) &= (1, x_1, x_2, x_1^2, x_1x_2, x_2^2). \end{aligned}$$

Необхідно зауважити, що для всіх нелінійних вирішальних правил типу $d^{(N,2)}(\mathbf{x}) = 0$ використовують матричну форму їх подання:

$$d^{(N,2)}(\mathbf{x}) = \mathbf{x}^T A \mathbf{x} + \mathbf{x}^T \mathbf{b} + c = 0, \quad (1.3.11)$$

де матриця $A = (a_{ij})$, вектор $\mathbf{b} = (b_1, b_2, \dots, b_N)^T$ та c відповідають коефіцієнтам \mathbf{w} . Якщо матриця A – додатньо-визначена, то $d^{(N,2)}(\mathbf{x}) = 0$ є гіпереліпсоїдом. Якщо A – від’ємно-визначена, то $d^{(N,2)}(\mathbf{x}) = 0$ – гіпергіперboloїд. В окремому випадку, коли матриця A – одинична, то $d^{(N,2)}(\mathbf{x}) = 0$ – гіперсфера.

Збільшення значень N та t приводить до стрімкого зростання кількості коефіцієнтів \mathbf{w} вирішальних правил. Зменшення оперативності та великі ресурсні витрати як на етапі формування таких вирішальних правил, так і при їх

використанні в системах розпізнавання образів значно звузили коло їх практичного застосування. Більше того, сучасні алгоритми розпізнавання зазвичай використовують нескладні вирішальні правила, що поєднуються в певну структуру, як, наприклад, у нейромережевому чи ієрархічному підході, або застосовують додаткові адаптивні процедури попередньої обробки вхідних даних, як, наприклад, в інтелектуальних системах на базі нечіткої логіки.

1.4 Критерії якості розпізнавання образів

Якість класифікації об'єктів істотно залежить від вибору міри їх близькості (подібності).

Визначення 1.4.1 Мірою близькості називається величина, що має граничне значення і здатна збільшуватися із зменшенням відстані між реалізаціями класів розпізнавання.

Розглянемо основні способи визначення близькості між об'єктами для кількісних шкал вимірювань [16-18].

Найбільш поширеною мірою близькості в теорії розпізнавання образів є евклідова відстань:

$$d_E(x_m, x_l) = \sqrt{\sum_{i=1}^N (x_m^i - x_l^i)^2}, \quad (1.4.1)$$

де x_m – реалізація класу X_m^o ; x_l – реалізація класу X_l^o ; x_m^i – i -та ознака розпізнавання класу X_m^o ; x_l^i – i -та ознака розпізнавання класу X_l^o ; N – кількість ознак розпізнавання; M – кількість класів розпізнавання.

Відстань (1.4.1) відповідає інтуїтивному уявленню про геометричну близькість двох точок у багатовимірному просторі ознак і особливо ефективно розділяє вектори-реалізації класів розпізнавання при обґрунтуванні гіпотези компактності їх розподілу. Ця міра широко використовується в методах кластер-аналізу, які ґрунтуються на детермінованих дистанційних критеріях близькості.

Для надання більшої ваги значно віддаленим одна від одної реалізаціям класів розпізнавання застосовують квадрат евклідової відстані

$$d_E^2(x_m, x_l) = \sum_{i=1}^N (x_m^i - x_l^i)^2. \quad (1.4.2)$$

Міра (1.4.2) дозволяє збільшувати внесок віддалених реалізацій, оскільки вони підносяться до квадрата.

Для зменшення впливу великих різниць застосовують лінійну (манхеттенську) міру близькості:

$$d(x_m, x_l) = \sum_{i=1}^N |x_m^i - x_l^i|, \quad m, l = \overline{1, M}. \quad (1.4.3)$$

Застосування міри (1.4.3) виправдано в ситуаціях, пов'язаних із багатокритеріальним оцінюванням (наприклад, якості продукції), де для порівняння об'єктів використовують модулі відхилень.

У разі, коли необхідно зменшити або збільшити вагу реалізацій образів, що сильно відрізняються один від одного, застосовують узагальнену степеневу відстань:

$$d(x_m, x_l) = r \sqrt{\sum_{i=1}^N (x_m^i - x_l^i)^p}, \quad (1.4.4)$$

де r – параметр, який відповідає за прогресивне зваження великих відстаней між об'єктами; p – параметр, що відповідає за поступове зваження різниць за окремими координатами.

Якщо у виразі (1.5.4) обидва параметри дорівнюють двом, то ця відстань збігається з відстанню Евкліда.

На практиці має місце гіпотеза нечіткої компактності реалізацій образів, оскільки апіорно класи перетинаються і мають нечіткі (розмиті) межі. Тому застосування в задачах класифікації вищенаведених детермінованих дистанційних критеріїв близькості не дозволяє в загальному випадку побудувати чітке розбиття простору ознак на класи розпізнавання. Для усунення цього недоліку в задачах розпізнавання образів набула застосування міра Махаланобіса [18]:

$$d_M(x_m, x_l) = (x_m - x_l)^T W^{-1} (x_m - x_l), \quad (1.4.5)$$

де T – символ транспонування вектора-стовпчика; W^{-1} – обернена коваріаційна матриця.

При дисперсії $\sigma^2 = 1$ міра (1.4.5) переходить у детерміновану дистанційну міру.

Під час аналізу та синтезу здатних навчатися систем розпізнавання як загальна міра близькості (подібності) образів широко використовується інформаційна міра у вигляді

$$C_l = 1 - E, \quad (1.4.6)$$

де E – нормована інформаційна міра, що є мірою різноманітності класів розпізнавання.

Найбільшого поширення в практичних задачах інформаційного синтезу здатних навчатися систем розпізнавання набули ентропійна (за Шенноном) міра [19-21] та інформаційна міра Кульбака [22].

Нормований ентропійний критерій функціональної ефективності за Шенноном має вигляд

$$E = \frac{H_0 - H(\gamma)}{H_0}, \quad (1.4.7)$$

де H_0 – апіорна (безумовна) середня ентропія:

$$H_0 = -\sum_{l=1}^M p(\gamma_l) \log_2 p(\gamma_l), \quad (1.4.8)$$

$H(\gamma)$ – апостеріорна умовна ентропія, що характеризує залишкову невизначеність після прийняття рішень:

$$H(\gamma) = -\sum_{l=1}^M p(\gamma_l) \sum_{m=1}^M p(\mu_m / \gamma_l) \log_2 p(\mu_m / \gamma_l). \quad (1.4.9)$$

У виразах (1.4.8) і (1.4.9) прийняті такі позначення: $p(\gamma_l)$ – безумовна (апіорна) ймовірність прийняття гіпотези γ_l ; $p(\mu_m / \gamma_l)$ – апостеріорна умовна ймовірність прийняття гіпотези μ_m за умови, що апіорно була прийнята гіпотеза γ_l ; M – кількість альтернативних гіпотез.

На практиці часто приймаються допущення:

1) рішення є двохальтернативними ($M = 2$);

2) оскільки система розпізнавання функціонує за умови апіорної невизначеності, то згідно з принципом Бернуллі – Лапласа виправданим є прийняття рівноймовірних гіпотез

$$p(\gamma_1) = \dots = p(\gamma_m) = \dots = \frac{1}{M}. \quad (1.4.10)$$

Тоді критерій (1.4.7) із урахуванням виразів (1.4.8) – (1.4.10) набирає вигляду

$$E = 1 + \frac{1}{2} \sum_{l=1}^2 \sum_{m=1}^2 p(\mu_m / \gamma_l) \log_2 p(\mu_m / \gamma_l). \quad (1.4.11)$$

Таким чином, вираз (1.4.11) за прийнятих допущень збігається з формулою для обчислення кількості середньої умовної інформації [13]. У праці [19] запропонована модифікація ентропійного КФЕ, здатної навчатися системи розпізнавання для радіально-базисних вирішальних правил. У цьому випадку ентропійний критерій оптимізації параметрів разі системи розпізнавання після заміни за рівноймовірних гіпотез відповідних апостеріорних імовірностей на апіорні за формулою Байєса має вигляд

$$\begin{aligned}
E_m^{(k)} = 1 + \frac{1}{2} & \left(\frac{\alpha_m^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \log_2 \frac{\alpha_m^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} + \right. \\
& + \frac{\beta_m^{(k)}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} \log_2 \frac{\beta_m^{(k)}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} + \\
& + \frac{D_{1,m}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} \log_2 \frac{D_{1,m}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} + \\
& \left. + \frac{D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \log_2 \frac{D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \right), \quad (1.4.12)
\end{aligned}$$

де $\alpha_m^{(k)}(d)$ – помилка першого роду прийняття рішення на k -му кроці навчання; $\beta_m^{(k)}(d)$ – помилка другого роду; $D_{1,m}^{(k)}(d)$ – перша достовірність; $D_{2,m}^{(k)}(d)$ – друга достовірність; d – дистанційна міра, що визначає відстань контейнера, який відновлюється в процесі навчання в радіальному базисі простору ознак розпізнавання, від геометричного центру класу розпізнавання.

Визначення 1.4.2 Контейнером класу розпізнавання називається радіально-базисна роздільна гіперповерхня, для якої деяким способом визначено геометричний центр.

Таким чином, критерій (1.4.12) є нелінійним і взаємно-неоднозначним функціоналом від точнісних характеристик, що потребує знаходження в процесі навчання робочої (допустимої) області для його визначення.

Розглянемо модифікацію диференціальної інформаційної міри Кульбака, що подається в праці [19] як добуток логарифмічного відношення повної ймовірності $P_{t,m}^{(k)}$ правильного прийняття рішень про належність реалізацій

класу X_m^o до повної ймовірності помилкового прийняття рішень $P_{f,m}^{(k)}$ на їх відхилення. Для двохальтернативних рішень за допущення згідно із принципом Бернуллі-Лапласа, що $p(\mu_m) = p(\mu_c) = 0,5$, модифікована міра Кульбака має вигляд

$$\begin{aligned}
 E_{Km}^{(k)} &= \log_2 \frac{P_{t,m}^{(k)}}{P_{f,m}^{(k)}} * [P_{t,m}^{(k)} - P_{f,m}^{(k)}] = \left| \begin{array}{l} P_{t,m}^{(k)} = 0,5D_{1,m}^{(k)}(d) + 0,5D_{2,m}^{(k)}(d) \\ P_{f,m}^{(k)} = 0,5\alpha_m^{(k)}(d) + 0,5\beta_m^{(k)}(d) \end{array} \right| = \\
 &= \log_2 \left(\frac{D_{1,m}^{(k)}(d) + D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + \beta_m^{(k)}(d)} \right) \left[(D_{1,m}^{(k)}(d) + D_{2,m}^{(k)}(d)) - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d)) \right] = \\
 &= \log_2 \left(\frac{2 - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))}{\alpha_m^{(k)}(d) + \beta_m^{(k)}(d)} \right) * [1 - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))]. \quad (1.4.13)
 \end{aligned}$$

Критерій (1.4.13) можна подати у нормованому вигляді

$$E_{K,m}^{(k)} = \frac{E_{Km}^{(k)}}{E_{K \max}^{(k)}}, \quad (1.4.14)$$

де $E_{K \max}^{(k)}$ – значення інформаційного критерію, обчисленого за формулою (1.4.13) за значень:

$$\begin{aligned}
 D_{1,m}^{(k)}(d) &= D_{2,m}^{(k)}(d) = 1 \\
 \alpha_m^{(k)}(d) &= \beta_m^{(k)}(d) = 0.
 \end{aligned}$$

При інформаційному синтезі здатної навчатися системи розпізнавання нормування критеріїв оптимізації не є обов'язковим, оскільки тут розв'язується задача пошуку екстремальних значень параметрів навчання, що відповідають

глобальному максимуму КФЕ у робочій області визначення його функції. Однак нормування критеріїв оптимізації є доцільним при порівняльному аналізі результатів досліджень і під час оцінювання ступеня близькості реальної системи до потенційної.

Використання статистичних інформаційних критеріїв як КФЕ здатних навчатися систем розпізнавання образів дозволяє здійснювати їх інформаційний аналіз і синтез для побудови безпомилкових за навчальною матрицею вирішальних правил. При цьому основна перевага статистичних інформаційних критеріїв перед детермінованими дистанційними критеріями полягає в можливості трансформувати в процесі навчання системи апіорно нечітке розбиття простору ознак на класи розпізнавання в чітке розбиття класів еквівалентності.

Контрольні запитання та завдання до першого розділу

- 1 Що називається образом?
- 2 Що називається ознакою розпізнавання?
- 3 Що називається вектором-реалізацією образу?
- 4 Що називається системою розпізнавання образів?
- 5 Що називається вирішальним правилом?
- 6 Що називається навчальною матрицею?
- 7 Що називається навчальною вибіркою?
- 8 Які основні підходи до аналізу й синтезу систем розпізнавання?
- 9 Що називається параметром функціонування системи розпізнавання?

10 Які основні етапи побудови вирішальних правил?

11 Який вигляд має лінійне вирішальне правило для класу розпізнавання X_1^o ?

12 Що називається областю рішень для класу розпізнавання X_m^o , $1 \leq m \leq M$?

13 Для класу X_1^o , реалізації якого знаходяться всередині кола одиничного радіуса із центром в (1; 2), та класу X_2^o , реалізації якого знаходяться всередині кола одиничного радіуса із центром в (4; 0), було запропоновано такі лінійні вирішальні правила:

а) $d_1(x) = 2 - x_1$;

б) $d_1(x) = 3 - x_1$.

Поясніть, яке з правил доцільніше використовувати.

14 Нехай $x, y \in X_1^o$. Доведіть, що для X_1^o та X_2^o можливо або неможливо сформулювати лінійне вирішальне правило, якщо

а) $z = \frac{x+y}{2} \in X_2^o$;

б) $z = 2x - y \in X_2^o$;

в) $z = x - y \in X_2^o$.

15 Нехай класи $X_1^o, X_2^o, \dots, X_m^o$ відповідають окремим вершинам випуклого багатокутника. Покажіть, що для таких класів спостерігається абсолютна лінійна роздільність. Випуклим називається багатокутник, усі точки якого знаходяться по один бік від будь-якої прямої, що проходить через дві його сусідні вершини.

16 Нехай усі реалізації класу X_1^o знаходяться всередині кола одиничного радіуса із центром в (1; 0), а реалізації класу X_2^o – кола одиничного радіуса із центром в (4; 0). Крім того, одна реалізація класу X_2^o є точкою, що знаходиться за

межами відповідного кола. Визначте область, яка є «забороненою» для цієї точки, якщо для класів X_1^o та X_2^o спостерігається лінійна роздільність.

17 Нехай сформовані лінійні вирішальні правила трьох класів X_1^o , X_2^o та X_3^o , що відповідно мають вигляд

$$d_1(\mathbf{x}) = x_1 - 1, d_2(\mathbf{x}) = x_2 - 2 \text{ і } d_3(\mathbf{x}) = x_1 - x_2 - 1.$$

Відобразити область рішень для випадку:

- а) абсолютної лінійної роздільності класів;
- б) попарної лінійної роздільності класів з

$$d_{i,j}(\mathbf{x}) = d_i(\mathbf{x}) - d_j(\mathbf{x}).$$

18 Розглянемо випадок, коли кожен із класів розпізнавання складається лише з однієї реалізації:

а) визначте чи буде спостерігатись абсолютна лінійна роздільність для трьох таких класів розпізнавання;

б) відобразіть нетривіальний приклад неабсолютної лінійної роздільності для чотирьох таких класів;

в) чи можна сформулювати п'ять таких класів за умови, що для будь-яких чотирьох із них спостерігається абсолютна лінійна роздільність, але вона відсутня для всіх п'яти класів?

19 Нехай реалізації класів X_1^o та X_2^o мають вигляд $\mathbf{x} = (x_1, x_2, x_3)$. Сформовано відповідне вирішальне правило для цих класів:

$$d(\mathbf{x}) = 1 + 4x_1 - 2x_2 + 2x_1^2 + x_2x_3 + x_3^2 = 0.$$

При цьому $d(\mathbf{x}) > 0$, якщо $\mathbf{x} \in X_1^o$, $d(\mathbf{x}) < 0$, якщо $\mathbf{x} \in X_2^o$:

- а) зведіть $d(\mathbf{x})$ до вигляду (1.3.10);

б) визначте належність таких реалізацій:

$(1, 1, 1)$, $(1, 10, 0)$, $(0, 0.5, 0)$.

20 Що називається мірою близькості образів?

21 Що називається евклідовою мірою близькості образів?

22 Що називається манхеттинською мірою близькості образів?

23 Що називається мірою близькості Махалобіса?

24 Що називається інформаційною мірою близькості?

25 Який вигляд має нормований ентропійний критерій Шеннона?

26 Що називається першою достовірністю?

27 Що називається другою достовірністю?

28 Що називається помилкою першого роду?

29 Що називається помилкою другого роду?

30 Що називається контейнером класу розпізнавання?

31 Який вигляд має інформаційна міра Кульбака?

32 Який вигляд має нормована міра Кульбака?

2.1 Метод еталонів

Основним критерієм визначення детермінованості методу розпізнавання образів є наявність детермінованих (чітких) вирішальних правил. При цьому в рамках геометричного підходу вирішальні правила задаються геометричними параметрами контейнерів класів розпізнавання, побудованими в просторі ознак. Якщо два вектори-реалізації образу знаходяться на невеликій відстані один від одного, то спостерігається їх топологічна «подібність», або компактність.

Визначення 2.1.1 Компактністю називається розподіл реалізацій образу, що знаходиться в обмеженій області простору ознак розпізнавання.

Компактності реалізацій образів поділяють на унімодальні та полімодальні.

Визначення 2.1.2 Унімодальною називається компактність, у якій реалізації образу розподілені навколо єдиного центру їх розсіювання.

Визначення 2.1.3 Полімодальною називається компактність, у якій реалізації образу розподілені навколо декількох центрів їх розсіювання.

Крім того, компактність реалізацій образів поділяють на чітку та нечітку.

Визначення 2.1.4. Чіткою називається компактність, у якій реалізації належать класам розпізнавання, що в просторі ознак не перетинаються.

Визначення 2.1.4. Нечіткою називається компактність, у якій реалізації належать різним класам розпізнавання.

На рис. 2.1.5 показано приклад розподілу реалізацій двох класів розпізнавання.

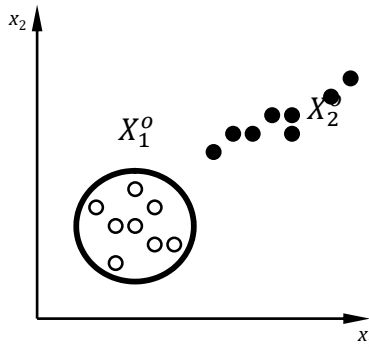


Рисунок 2.1.1 – Розподіл реалізацій двох класів розпізнавання

Аналіз рис. 2.1.1 показує, що реалізації класу X_1^o розподілені рівномірно навколо його єдиного центру. При цьому компактність їх розподілу є чіткою. Контейнер класу X_1^o , що відображений на рис. 2.1.1 у вигляді кола, визначає межі відповідного класу розпізнавання. Частина векторів-реалізацій класу X_2^o не є «подібними» в топологічному сенсі, оскільки відстань між ними значно більша, ніж між іншими. Фактично реалізації таких класів займають декілька суміжних кластерів із центрами, що не збігаються. Для відображення вирішального правила для такого класу необхідно використовувати контейнери більш складної форми.

Якщо в задачі розпізнавання образів розглядаються виключно класи типу X_1^o (рис. 2.1.1), для кожного з яких можна визначити типовий вектор-реалізацію, або *еталон* (еталонний вектор) [23-24], то прийняття рішень про належність реалізації $\mathbf{y} = (y_1, y_2, \dots, y_N)$ відповідному класу здійснюються за дистанційним критерієм *dist*:

$$\min_{k=1..m} dist_k(\mathbf{y}) = \min_{k=1..m} \sqrt{\sum_{i=1}^N (x_{k,i} - y_i)^2}, \quad (2.1.1)$$

де $dist_k(\mathbf{y})$ – відстань між еталоном класу X_k^o і реалізацією \mathbf{y} ; $x_{k,i}$ – реалізація i -ї ознаки еталонного вектора класу X_k^o , y_i – значення i -ї координати реалізації \mathbf{y} .

У разі якщо декілька еталонів одночасно знаходяться на мінімальній відстані від реалізації \mathbf{y} , то можна прийняти рішення про її належність будь-якому з відповідних класів або визначити її як таку, для якої встановити цю належність неможливо.

Задача мінімізації $dist_k^2$ еквівалентна задачі (2.1.1):

$$\begin{aligned} \min_{k=1..m} dist_k^2 &= \min_{k=1..m} \sum_{i=1}^N (x_{k,i} - y_i)^2 = \\ &= \min_{k=1..m} \sum_{i=1}^N (x_{k,i}^2 - 2x_{k,i}y_i + y_i^2). \end{aligned}$$

З урахуванням того, що сума $\sum_{i=1}^N y_i^2$ не залежить від k , така задача зводиться до

$$\min_{k=1..m} \sum_{i=1}^N (x_{k,i}^2 - 2x_{k,i}y_i) = \min_{k=1..m} \left(\sum_{i=1}^N x_{k,i}^2 - \sum_{i=1}^N 2x_{k,i}y_i \right)$$

або

$$\max_{k=1..m} \left(\sum_{i=1}^N 2x_{k,i}y_i - \sum_{i=1}^N x_{k,i}^2 \right). \quad (2.1.2)$$

При використанні коефіцієнтів

$$\mathbf{w}_k = \left(- \sum_{i=1}^N x_{k,i}^2, 2x_{k,1}, 2x_{k,2}, \dots, 2x_{k,N} \right)$$

і розширеної форми вектора-реалізації $\mathbf{y} = (1, y_1, y_2, \dots, y_N)$ задачу (2.1.2) можна подати у вигляді

$$\max_{k=1..m} \mathbf{w}_k \cdot \mathbf{y}^T = \max_{k=1..m} d_k(\mathbf{y}),$$

де $d_k(\mathbf{y})$ – ліва частина лінійного вирішального правила (1.3.3).

При цьому роздільні поверхні формуються для кожної пари класів X_k^o і X_l^o у вигляді (1.3.6):

$$d_{k,l}(\mathbf{y}) = d_k(\mathbf{y}) - d_l(\mathbf{y}) = 0.$$

Геометричне місце точок, що задовольняють таку умову, є гіперплощиною, яка проходить через середину відрізка між еталонними векторами класів X_k^o і X_l^o та перпендикулярна до нього.

У загальному випадку метод еталонів здатний сформулювати вирішальні правила і для класу типу X_2^o (рис. 2.1.1), якщо його реалізації можна подати у вигляді множини підкласів типу X_1^o , для кожного з яких формується власний еталонний вектор.

Розглянемо алгоритм формування вирішальних правил за методом еталонів для загального випадку.

Вхідні дані:

N – потужність словника ознак розпізнавання;

M – потужність алфавіту класів розпізнавання;

l_m – кількість підкласів класу X_m^o ;

$\|x_{l_m}^{(j)}\|$ – навчальна матриця l_m -го підкласу класу X_m^o
 ($l_m = 1, 2, \dots, L, j = 1, 2, \dots, n_{l_m}$);

n_{l_m} – обсяг навчальної матриці l_m -го підкласу класу X_m^o ;

$y = (y_1, y_2, \dots, y_N)$ – вхідна реалізація, для якої необхідно прийняти рішення про її належність до класу із заданого алфавіту $\{X_m^o\}$.

Вихідними даними є номер класу m^* , до якого належить вхідна реалізація y .

Крок 1 Визначасмо координати еталонних двійкових векторів x_{l_k} :

$$x_{l_m,i} = \frac{1}{n_{l_m}} \sum_{j=1}^{n_{l_m}} x_{l_m,i}^{(j)}.$$

Крок 2 Обчислюємо значення $d_{l_m}(y)$:

$$d_{l_m}(y) = 2 \sum_{i=1}^N x_{l_m,i} y_i - \sum_{i=1}^N x_{l_m,i}^2.$$

Крок 3 Визначаємо максимальне значення $d_{l_k}(y)$ і відповідний йому номер класу, до якого належить y :

$$k^* = \arg \max_k \left[\max_{l_k} d_m(y) \right].$$

Крок 4 ЗУПИН.

Як приклад розглянемо задачу класифікації реалізації $y = (y_1, y_2) = (13; 7)$ за методом еталонів.

Вхідна навчальна матриця подана в табл. 3.1.1.

Таблиця 3.1.1 – Вхідні дані

		Клас X_1^o				Клас X_2^o				Клас X_3^o	
		x_1	x_2			x_1	x_2			x_1	x_2
Перший підклас		14	12	Перший підклас		11	14	Перший підклас		12	15
		12	14			11	13			14	16
		13	14			10	12			13	17
		12	12			9	12		Другий підклас	10	11
		14	13			9	14			9	9
Другий підклас		12	9	Другий підклас		17	14	Третій підклас		11	10
		14	11			20	15		6	14	
		14	9			20	11		6	13	
		13	11			18	12		8	13	
		12	10			20	13		8	12	

Аналіз табл. 2.1.1 показує, що потужність словника ознак розпізнавання дорівнює $N = 2$, потужність алфавіту класів розпізнавання – $M = 3$, а кількість підкласів класів X_1^o , X_2^o і X_3^o дорівнює відповідно $m_1 = 2$, $m_2 = 2$ і $m_3 = 3$. При цьому обсяги навчальних матриць відповідно дорівнюють

$$\begin{aligned}
 n_{1_1} &= n_{2_1} = n_{1_2} = n_{2_2} = 5, \\
 n_{1_3} &= n_{2_3} = 3, \\
 n_{3_3} &= 4.
 \end{aligned}$$

Визначаємо координати еталонних двійкових векторів:

$$\begin{aligned}
 \mathbf{x}_{1_1} &= (13, 13), \mathbf{x}_{2_1} = (13, 10), \\
 \mathbf{x}_{1_2} &= (10, 13), \mathbf{x}_{2_2} = (19, 13), \\
 \mathbf{x}_{1_3} &= (13, 16), \mathbf{x}_{2_3} = (10, 10), \\
 \mathbf{x}_{3_3} &= (7, 13).
 \end{aligned}$$

Обчислюємо значення $d_{l_m}(\mathbf{y})$:

$$\begin{aligned}
 d_{1_1} &= 2(13y_1 + 13y_2) - 338 = 182, \\
 d_{2_1} &= 2(13y_1 + 10y_2) - 269 = 209, \\
 d_{1_2} &= 2(10y_1 + 13y_2) - 269 = 173, \\
 d_{2_2} &= 2(19y_1 + 13y_2) - 530 = 146, \\
 d_{1_3} &= 2(13y_1 + 16y_2) - 425 = 137, \\
 d_{2_3} &= 2(10y_1 + 10y_2) - 200 = 200, \\
 d_{3_3} &= 2(7y_1 + 13y_2) - 218 = 146.
 \end{aligned}$$

Оскільки максимальне значення $d_{l_k}(\mathbf{y}) = d_{2_1} = 209$, то реалізація $\mathbf{y} = (13; 7)$ належить до класу X_1^o .

Додатково формуємо роздільні поверхні для кожної пари класів X_k^o і X_l^o у вигляді (1.3.6):

$$\begin{aligned}
 d_{1_1,1_2}(\mathbf{y}) &= -d_{1_2,1_1}(\mathbf{y}) = d_{1_1}(\mathbf{y}) - d_{1_2}(\mathbf{y}) = 6y_1 - 69 = 0; \\
 d_{1_1,2_2}(\mathbf{y}) &= -d_{2_2,1_1}(\mathbf{y}) = d_{1_1}(\mathbf{y}) - d_{2_2}(\mathbf{y}) = \\
 &= -12y_1 + 192 = 0; \\
 d_{1_1,1_3}(\mathbf{y}) &= -d_{1_3,1_1}(\mathbf{y}) = d_{1_1}(\mathbf{y}) - d_{1_3}(\mathbf{y}) = -6y_2 + 87 = 0;
 \end{aligned}$$

$$\begin{aligned}
d_{1_1,2_3}(\mathbf{y}) &= -d_{2_3,1_1}(\mathbf{y}) = d_{1_1}(\mathbf{y}) - d_{2_3}(\mathbf{y}) = \\
&= 6y_1 + 6y_2 - 138 = 0; \\
d_{1_1,3_3}(\mathbf{y}) &= -d_{3_3,1_1}(\mathbf{y}) = d_{1_1}(\mathbf{y}) - d_{3_3}(\mathbf{y}) = 12y_1 - 120; \\
d_{2_1,1_2}(\mathbf{y}) - d_{1_2,2_1}(\mathbf{y}) &= d_{2_1}(\mathbf{y}) - d_{1_2}(\mathbf{y}) = \\
&= 6y_1 - 6y_2 = 0; \\
d_{2_1,2_2}(\mathbf{y}) &= -d_{2_2,2_1}(\mathbf{y}) = d_{2_1}(\mathbf{y}) - d_{2_2}(\mathbf{y}) = \\
&= -12y_1 - 6y_2 + 261 = 0; \\
d_{2_1,1_3}(\mathbf{y}) &= -d_{1_3,2_1}(\mathbf{y}) = d_{2_1}(\mathbf{y}) - d_{1_3}(\mathbf{y}) = \\
&= -12y_2 + 156 = 0; \\
d_{2_1,2_3}(\mathbf{y}) &= -d_{2_3,2_1}(\mathbf{y}) = d_{2_1}(\mathbf{y}) - d_{2_3}(\mathbf{y}) = 6y_1 - 69 = 0; \\
d_{2_1,3_3}(\mathbf{y}) &= -d_{3_3,2_1}(\mathbf{y}) = d_{2_1}(\mathbf{y}) - d_{3_3}(\mathbf{y}) = \\
&= 12y_1 - 6y_1 - 51 = 0; \\
d_{1_2,1_3}(\mathbf{y}) &= -d_{1_3,1_2}(\mathbf{y}) = d_{1_2}(\mathbf{y}) - d_{1_3}(\mathbf{y}) = \\
&= -6y_1 - 6y_2 + 156 = 0; \\
d_{1_2,2_3}(\mathbf{y}) &= -d_{2_3,1_2}(\mathbf{y}) = d_{1_2}(\mathbf{y}) - d_{2_3}(\mathbf{y}) = 6y_2 - 69 = 0; \\
d_{1_2,3_3}(\mathbf{y}) &= -d_{3_3,1_2}(\mathbf{y}) = d_{1_2}(\mathbf{y}) - d_{3_3}(\mathbf{y}) = 6y_1 - 51 = 0; \\
d_{2_2,1_3}(\mathbf{y}) &= -d_{1_3,2_2}(\mathbf{y}) = d_{2_2}(\mathbf{y}) - d_{1_3}(\mathbf{y}) = \\
&= 12y_1 - 6y_2 - 105 = 0; \\
d_{2_2,2_3}(\mathbf{y}) &= -d_{2_3,2_2}(\mathbf{y}) = d_{2_2}(\mathbf{y}) - d_{2_3}(\mathbf{y}) = \\
&= 18y_1 + 6y_2 - 330 = 0; \\
d_{2_2,3_3}(\mathbf{y}) &= -d_{3_3,2_2}(\mathbf{y}) = d_{2_2}(\mathbf{y}) - d_{3_3}(\mathbf{y}) = 24y_1 - 312 = 0.
\end{aligned}$$

При застосуванні таких вирішальних правил для визначення належності реалізації \mathbf{y} умова (1.3.7) виконується лише для всіх роздільних поверхонь $d_{2_1,l_m}(\mathbf{y})$ за $l \neq 2, m \neq 1$. Таким чином, реалізація $\mathbf{y} = (13, 7)$ належить до другого підкласу класу X_1^0 .

На рис. 2.1.2 *а* подане графічне зображення векторів-реалізацій (табл. 2.1.1), сформовані еталонні вектори (рис. 2.1.2 *б*), області рішень (рис. 2.1.2 *в*) і положення реалізації у відносно них (рис. 2.1.2 *г*).

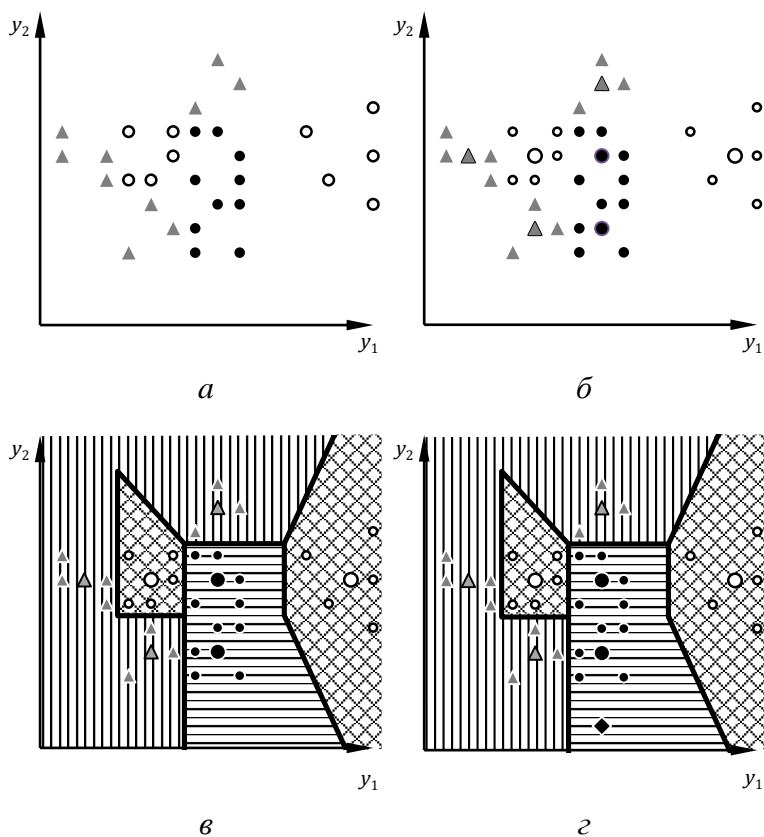


Рисунок 2.1.2 – Графічне зображення етапів застосування методу еталонів: *а* – навчальні матриці; *б* – еталонні вектори; *в* – області рішень; *г* – положення реалізації y

На рис. 2.1.2 для позначення належності вхідних реалізацій використані такі позначки $\circ (x \in X_1^o)$, $\bullet (x \in X_2^o)$ та $\Delta (x \in X_3^o)$. При цьому позначки для еталонних векторів

класів мають більший розмір. Реалізацію у позначено маркером \blacklozenge . Область рішень класу X_1^o позначена горизонтальним штрихуванням \equiv , класу X_2^o – подвійним штрихуванням \boxtimes , класу X_3^o – вертикальним штрихуванням \equiv . При цьому об'єднання областей рішень усіх класів повністю збігається з простором ознак розпізнавання.

Перевага методу – простота вирішальних правил.

Основний недолік – метод носить модельний характер, оскільки виключає перетин класів розпізнавання.

2.2 Метод еталонів, що дробляться

Метод еталонів, що дробляться, спрямований на усунення модельності методу еталонів шляхом наближення до практичних задач, для яких є характерним перетин класів [24]. Процес навчання за методом еталонів, що дробляться, складається із двох етапів. На першому етапі за навчальною вибіркою для кожного класу будується контейнер у вигляді гіперсфери якомога меншого радіуса, до якої входять усі реалізації класу. При цьому формується еталон кожного класу, обчислюється відстань від еталона до всіх реалізацій цього класу. На другому етапі обирається максимальна із цих відстаней r_{max} . Відтворюється гіперсфера з центром у еталоні й радіусом $R=r_{max}+\epsilon$, що охоплює всі реалізації свого класу. Аналогічна процедура проводиться для всіх класів із заданого алфавіту. На рис. 2.2.1 наведений приклад розподілу реалізацій двох класів розпізнавання у двовимірному просторі ознак.

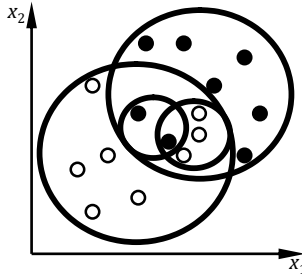


Рисунок 2.2.1 – Вирішальні правила сформовані за методом еталонів, що дробляться

Якщо гіперсфери різних образів перетинаються й в області перетину містяться реалізації більш ніж одного класу, то для них будуються гіперсфери другого рівня, потім третього і т. д. доти, поки гіперсфери не будуть перетинатися або в області перетину міститимуться реалізації лише одного класу.

Розпізнавання проводиться таким чином. Визначається місцезнаходження реалізації, належність якої необхідно встановити, щодо гіперсфер першого рівня. Якщо така реалізація розміщується в частині гіперсфери, що відповідає одному й лише одному класу, то процедура розпізнавання зупиняється. Якщо реалізація розміщується в області перетину гіперсфер, що під час навчання містить реалізації більш ніж одного класу, то переходимо до гіперсфер другого рівня й проводимо процедуру розпізнавання, аналогічну до гіперсфер першого рівня. Такий процес триває, поки належність реалізації до певного класу не буде встановлена однозначно. Хоча така подія може й не настати. Зокрема, невідома реалізація може не потрапити до жодної з гіперсфер будь-якого рівня. У таких випадках виникає необхідність включити у вирішальне правило відповідні дії. Наприклад,

система може приймати гіпотезу про неналежність реалізації до жодного із класів або визначати належність за іншим методом, наприклад побудови еталонів.

Розглянемо алгоритм формування вирішальних правил за методом еталонів, що дробляться.

Вхідні дані:

N – потужність словника ознак розпізнавання;

$m_{(1)}$ – потужність алфавіту класів розпізнавання першого рівня;

$\|x_{(1)k}^{(j)}\|$ – навчальна матриця класу X_k^o першого рівня

($k = 1, 2, \dots, m_{(1)}, j = 1, 2, \dots, n_{(1)k}$);

$n_{(1)k}$ – обсяг навчальної матриці класу X_k^o першого рівня;

ε – константа, що використовується під час обчислення радіусів;

M_{lev} – максимальна кількість рівнів.

Вихідні дані:

$x_{(lev),k}$ – еталонні вектори класу X_k^o рівня lev ;

$R_{(lev)k}$ – радіуси гіперсфер класу X_k^o рівня lev .

Крок 1 Формуємо вирішальні правила першого рівня ($lev = 1$).

Крок 2 Визначаємо координати еталонних векторів $x_{(lev),k}$:

$$x_{(lev)k,i} = \frac{1}{n_k} \sum_{j=1}^{n_{(lev)k}} x_{(lev)k,i}^{(j)}$$

Крок 3 Обчислюємо значення $dist_k(x_{(lev)k}^{(j)})$:

$$dist_k(\mathbf{x}_{(lev)k}^{(j)}) = \sqrt{\sum_{i=1}^N (x_{(lev)k,i} - x_{(lev)k,i}^{(j)})^2}.$$

Крок 4 Визначаємо максимальні значення $dist_k(\mathbf{x}_{(lev)k}^{(j)})$ та відповідні радіуси гіперсфер $R_{(lev)k}$:

$$R_{(lev)k} = \max_j [dist_k(\mathbf{x}_{(lev)k}^{(j)})] + \varepsilon.$$

Крок 5 Обчислюємо значення $dist_k(\mathbf{x}_{(lev)l}^{(j)})$ за $l \neq k$:

$$dist_k(\mathbf{x}_{(lev)l}^{(j)}) = \sqrt{\sum_{i=1}^N (x_{(lev)k,i} - x_{(lev)l,i}^{(j)})^2}.$$

Крок 6 Визначаємо множини реалізацій класів, що знаходяться в гіперсферах інших класів:

$$\|\mathbf{x}_{(lev+1)k}^{(p)}\| = \{\mathbf{x}_{(lev)k}^{(j)} \mid \exists (l \neq k) dist_l(\mathbf{x}_{(lev)k}^{(j)}) \leq R_{(lev)l}\},$$

де $p = 1, 2, \dots, n_{(lev+1)k} \leq n_{(lev)k}$.

Крок 7 Визначаємо параметри навчальної матриці для вирішальних правил такого рівня ($lev = lev + 1$):

$$m_{(lev)} = card \{ \|\mathbf{x}_k^{(p)}\| \neq \emptyset \},$$

$$n_{(lev)k} = t_k \text{ для } \|\mathbf{x}_k^{(p)}\| \neq \emptyset.$$

Крок 8 Якщо $m_{(lev)} \geq 2$ та $lev \leq M_{lev}$, то крок 2, інакше ЗУПИН.

Як приклад розглянемо задачу формування вирішальних правил за методом еталонів, що дробляться ($\varepsilon = 0,5$). Вхідна навчальна матриця подана в табл. 2.2.1

Таблиця 2.2.1 – Вхідні дані

	Клас X_1^o					Клас X_2^o				
	1	2	3	4	5	1	2	3	4	5
x_1	10	10	10	11	10	10	11	9	11	12
x_2	11	16	15	15	13	9	12	10	10	13

Аналіз табл. 2.2.1 показує, що потужність словника ознак розпізнавання $N = 2$, потужність алфавіту класів розпізнавання першого рівня $m_{(1)} = 2$, обсяг навчальної матриці класів X_1^o та X_2^o першого рівня $n_{(1)1} = n_{(1)2} = 5$.

Ураховуючи те, що $\varepsilon = 0,5$, проведимо обчислення параметрів класифікатора з точністю до одного знака після коми.

Визначаємо еталони $\mathbf{x}_{(1),k}$ класів X_k^o ($k = 1, 2$):

$$\mathbf{x}_{(1),1} = (10.2, 14.0), \mathbf{x}_{(1),2} = (10.6, 10.8).$$

Обчислюємо відстань $dist_k(\mathbf{x}_{(1)k}^{(j)})$ від еталонів класів X_k^o до їх реалізацій:

$$dist_1(\mathbf{x}_{(1)1}^{(j)}) = (3.0, 2.0, 1.0, 1.3, 1.0);$$

$$dist_2(\mathbf{x}_{(1)2}^{(j)}) = (1.9, 1.3, 1.8, 0.9, 2.6).$$

Відтворюємо гіперсфери першого рівня класів X_k^o із центрами в еталонах $x_{(1),k}$ і радіусами, що більше максимальної відстані $dist_k(x_{(1)k}^{(j)})$ на ε :

$$R_{(1)1} = 3.0 + \varepsilon = 3.5; R_{(1)2} = 2.6 + \varepsilon = 3.1.$$

Обчислюємо відстань від еталонів класів X_k^o до реалізацій інших класів X_l^o ($l \neq k$):

$$dist_1(x_{(1)2}^{(j)}) = (5.0, 2.2, 4.2, 4.1, 2.1);$$

$$dist_2(x_{(1)1}^{(j)}) = (0.6, 5.2, 4.2, 4.2, 2.3).$$

Реалізації $x_{(1)1}^{(1)}$, $x_{(1)1}^{(5)}$ класу X_1^o та $x_{(1)2}^{(2)}$, $x_{(1)2}^{(5)}$ класу X_2^o входять в область перетину гіперсфер першого рівня, оскільки для них виконується умова

$$dist_l(x_{(1)k}^{(j)}) \leq R_{(1)l} \text{ за } l \neq k.$$

Таким чином, до області перетину входять реалізації декількох класів, що вказує на необхідність формування гіперсфер другого рівня. При цьому навчальна матриця має вигляд (табл. 3.2.2).

Таблиця 2.2.1 – Вхідні дані (другий рівень)

	Клас X_1^o		Клас X_2^o	
	1	2	1	2
x_1	10	10	11	12
x_2	11	13	12	13

Визначаємо еталонні вектори $\mathbf{x}_{(2),k}$ класів X_k^o ($k = 1, 2$):

$$\mathbf{x}_{(2),1} = (10.0, 12.0), \mathbf{x}_{(2),2} = (11.5, 12.5).$$

Обчислюємо відстань $dist_k(\mathbf{x}_{(2)k}^{(j)})$ від еталонних векторів класів X_k^o до їх реалізацій:

$$dist_1(\mathbf{x}_{(2)1}^{(j)}) = (1.0, 1.0); dist_2(\mathbf{x}_{(2)2}^{(j)}) = (0.7, 0.7).$$

Відтворюємо гіперсфери першого рівня класів X_k^o із центрами в еталонах $\mathbf{x}_{(2),k}$ і радіусами, що більше максимальної відстані $dist_k(\mathbf{x}_{(2)k}^{(j)})$ на величину ε :

$$R_{(2)1} = 1.0 + \varepsilon = 1.5; R_{(2)2} = 0.7 + \varepsilon = 1.2.$$

Обчислюємо відстань від еталонів класів X_k^o до реалізацій інших класів X_l^o ($l \neq k$):

$$dist_1(\mathbf{x}_{(2)2}^{(j)}) = (1.0, 2.2); dist_2(\mathbf{x}_{(2)1}^{(j)}) = (2.1, 1.6).$$

До області перетину гіперсфер другого рівня входить лише реалізація $\mathbf{x}_{(2)2}^{(1)}$ класу X_2^o , оскільки для неї виконується умова

$$dist_1(\mathbf{x}_{(2)2}^{(1)}) \leq R_{(2)1}.$$

Таким чином, до області перетину входять реалізації лише одного з класів, що надає можливість завершити формування детермінованих вирішальних правил. На

рисунку 2.2.2 сформовані правила, подані у графічному вигляді.

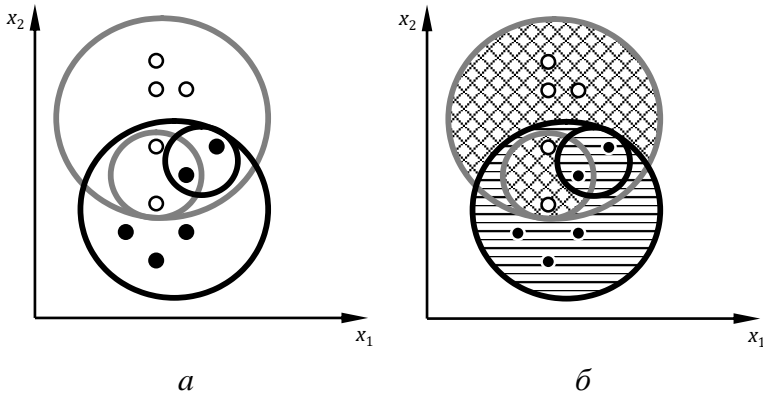


Рисунок 2.2.2 – Результати формування вирішальних правил за методом еталонів, що дробляється: *a* – гіперсфери класів; *б* – області рішень

На рис. 2.1.2 для позначення належності вхідних реалізацій використані такі позначення: \circ ($x \in X_1^o$), \bullet ($x \in X_2^o$). Область рішень класу X_1^o позначена горизонтальним штрихуванням \equiv , класу X_2^o – подвійним штрихуванням \boxtimes .

Крім того, правила можна подати і у такому вигляді:

- 1) якщо $d((11.5, 12.5); \mathbf{y}) \leq 1.2$, то \mathbf{y} належить класу X_2^o .
- 2) якщо $d((10.0, 12.0); \mathbf{y}) \leq 1.5$, то \mathbf{y} належить класу X_1^o .
- 3) якщо $d((10.6, 10.8); \mathbf{y}) \leq 3.1$, то \mathbf{y} належить класу X_2^o .
- 4) якщо $d((10.2, 14.0); \mathbf{y}) \leq 3.5$, то \mathbf{y} належить класу X_1^o .

При цьому

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2},$$

а пріоритет вирішального правила заданий його номером. При цьому вирішальне правило з більшим пріоритетом має менший номер.

Перевагами методу еталонів, що дробляться, є:

- метод враховує реальні апріорні розбиття класів розпізнавання, де вони перетинаються в просторі ознак;

- метод є відносно простим.

Метод має такі основні недоліки:

- контейнери різних рівнів не є оптимальними в інформаційному розумінні;

- оскільки реалізації на практиці є випадковими, то фактична достовірність розпізнавання на екзамені може істотно відрізнятися від асимптотичної достовірності, одержаної на етапі навчання, тобто цей метод не характеризується високою достовірністю розпізнавання.

2.3 Лінійні вирішальні правила

За цим методом роздільну гіперповерхню задають у вигляді лінійної функції (1.3.1 – 1.3.2), для якої необхідно визначити коефіцієнти $\mathbf{w} = (w_1, w_2, \dots, w_N)$.

Одним із прикладів реалізації лінійних вирішальних правил є перцептрон Розенблата (рис. 2.3.1).

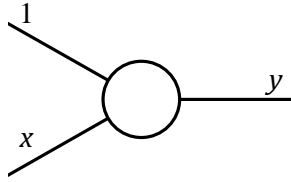


Рисунок 2.3.1 – Модель персептрона

При цьому

$$\begin{aligned} \mathbf{x} &\in R^N \text{ або } \mathbf{x} \in \{-1, 1\}^N; \\ y &\in R \text{ або } y \in \{-1, 1\}. \end{aligned}$$

Таку модель використовується для розв'язання задачі класифікації для двох класів. Вона є ідентичною задачі

$$y \in \{0, 1\}.$$

Будемо розглядати випадок

$$\mathbf{x} \in R^N; y \in \{-1, 1\}.$$

Функціонування персептрона описується такою залежністю:

$$y = \text{sign}(\mathbf{w}^T \mathbf{x} - \tau) = f(\mathbf{x}, \mathbf{w}), \quad (2.3.1)$$

де τ – деякий поріг; \mathbf{w} – вектор вагових коефіцієнтів персептрона.

У геометричній інтерпретації рівняння (2.3.1) визначає два підпростори

$$\begin{aligned} \{\mathbf{x}: y = +1\} &\Leftrightarrow H^+ = \{\mathbf{x}: \mathbf{w}^T \mathbf{x} \geq \tau\}, \\ \{\mathbf{x}: y = -1\} &\Leftrightarrow H^- = \{\mathbf{x}: \mathbf{w}^T \mathbf{x} < \tau\} \end{aligned} \quad (2.3.2)$$

з роздільною гіперплощиною (афінний підпростір розмірності $N - 1$):

$$H = \{\mathbf{x}: \mathbf{w}^T \mathbf{x} - \tau = 0\}. \quad (2.3.3)$$

Збільшимо розмірність простору ознак розпізнавання:

$$\mathbf{x} \in R^N \Rightarrow \tilde{\mathbf{x}} \in R^{N+1}, \quad (2.3.4)$$

де $\tilde{\mathbf{x}} = (1, x_1, x_2, \dots, x_N)$,

$$\mathbf{w} \in R^N \Rightarrow \tilde{\mathbf{w}} \in R^{N+1}, \quad (2.3.5)$$

де $\tilde{\mathbf{w}} = (\tau, w_1, w_2, \dots, w_N)$.

Ураховуючи (2.3.4) та (2.3.5), можна записати

$$\mathbf{w}^T \mathbf{x} - \tau = \tilde{\mathbf{w}}^T \tilde{\mathbf{x}}.$$

Навчання перцептрона є процесом налаштування вагових коефіцієнтів $\tilde{\mathbf{w}}$. Під час навчання нейронної мережі, як правило, математичні вирази для роздільних поверхонь відсутні. Тому навчання виконується лише на навчальній вибірці.

Навчальна вибірка (скінченна) задається множиною, що складається з пар вхід-вихід:

$$T = \{(\mathbf{x}^{(1)}, t^{(1)}), (\mathbf{x}^{(2)}, t^{(2)}), \dots, (\mathbf{x}^{(n)}, t^{(n)})\}, \quad (2.3.6)$$

де $t^{(j)} \in \{-1, 1\}$, $j = 1, 2, \dots, n$.

Мета навчання – налаштувати вагові коефіцієнти $\tilde{\mathbf{W}}$ таким чином, щоб для будь-яких $(\mathbf{x}^{(j)}, t^{(j)})$ виконувалося $y = \text{sign}(\tilde{\mathbf{W}}^T \tilde{\mathbf{x}}^{(j)}) = t^{(j)}$.

Розглянемо алгоритм навчання перцептрона Розенблата:

Вхідні дані:

N – потужність словника ознак розпізнавання;

$m = 2$ – потужність алфавіту класів розпізнавання;

$\|\mathbf{x}_k^{(j)}\|$ – навчальна матриця класу X_k^o ($k = 1, 2$,
 $j = 1, 2, \dots, n_k$);

n_k – обсяг навчальної матриці класу X_k^o .

Вихідні дані:

$\tilde{\mathbf{W}} = (w_0, w_1, w_2, \dots, w_N)$ – вагові коефіцієнти перцептрона.

Крок 1 Збільшуємо розмірність простору ознак розпізнавання:

$$\tilde{\mathbf{x}}_k^{(j)} = (1, x_{k,1}^{(j)}, x_{k,2}^{(j)}, \dots, x_{k,N}^{(j)}).$$

Крок 2 Формуємо множину

$$\begin{aligned} \mathbf{z}^{(j)} &= \tilde{\mathbf{x}}_1^{(j)}; \\ \mathbf{z}^{(n_k+j)} &= \tilde{\mathbf{x}}_2^{(j)}. \end{aligned}$$

Крок 3 Обираємо будь-який елемент $\mathbf{z}_k^{(j)}$ як початкове наближення для $\tilde{\mathbf{W}}$.

Крок 4 Обираємо випадкове значення $\mathbf{z}^{(l)}$. Якщо $\tilde{\mathbf{W}}^T \mathbf{z}^{(l)} > 0$, переходимо до кроку 4, інакше – до кроку 5.

Крок 5 Модифікація вагових коефіцієнтів $\tilde{\mathbf{W}} = \tilde{\mathbf{W}} + \varphi \mathbf{z}^{(l)}$, де $\varphi = 1$.

Крок 6 Якщо умова $\tilde{\mathbf{w}}^T \mathbf{z}^{(j)} > 0$ виконується для всіх векторів навчальної вибірки ($j = 1, 2, \dots, n_1, n_1 + 1, n_1 + 2, \dots, n_2$), то ЗУПИН, інакше переходимо до кроку 4.

Крок 5 обумовлений тим, що значення додатка $\tilde{\mathbf{w}}^T \mathbf{z}^{(l)}$ буде збільшуватися при додаванні до коефіцієнтів $\tilde{\mathbf{w}}$ значень $\mathbf{z}^{(l)}$, що дозволить перетворити від'ємні значення додатка $\tilde{\mathbf{w}}^T \mathbf{z}^{(l)}$ на додатні.)

Лінійні вирішальні правила можуть застосовуватися для нормальних образів, тобто для класів розпізнавання, що не перетинаються. Якщо класи перетинаються, для них навчальна вибірка не є нормально розподіленою, а вектори-реалізації образів не є лінійно роздільними.

Як приклад розглянемо задачу формування лінійних вирішальних правил у вигляді одношарового перцептрона. Вхідна навчальна матриця наведена в табл. 2.3.1.

Таблиця 2.3.1 – Вхідні дані

	Клас X_1^o					Клас X_2^o				
	1	2	3	4	5	1	2	3	4	5
x_1	10	10	10	11	9	10	11	9	11	9
x_2	18	16	15	14	12	9	11	10	10	10

Аналіз табл. 2.3.1 показує, що потужність словника ознак розпізнавання $N = 2$, потужність алфавіту класів розпізнавання першого рівня $m_{(1)} = 2$, обсяг навчальної матриці класів X_1^o та X_2^o першого рівня $n_{(1)1} = n_{(1)2} = 5$.

Збільшуємо розмірність простору ознак (табл. 2.3.2):

Таблиця 2.3.2 – Результати першого кроку навчання перцептрона

	Клас X_1^o					Клас X_2^o				
	1	2	3	4	5	1	2	3	4	5
x_1	10	10	10	11	9	10	11	9	11	9
x_2	18	16	15	14	12	9	11	10	10	10
x_3	1	1	1	1	1	1	1	1	1	1

Формуємо множину $\mathbf{z}^{(j)}$ (табл. 2.3.3).

Таблиця 2.3.3 – Результати другого кроку навчання перцептрона

j	Клас X_1^o					Клас X_2^o				
	1	2	3	4	5	6	7	8	9	10
z_1	10	10	10	11	9	-10	-11	-9	-11	-9
z_2	18	16	15	14	12	-9	-11	-10	-10	-10
z_3	1	1	1	1	1	-1	-1	-1	-1	-1

Вибираємо деякий елемент $\mathbf{z}^{(l)}$ як початкове наближення для $\tilde{\mathbf{W}}$:

$$\tilde{\mathbf{w}} = \mathbf{z}^{(1)} = \begin{pmatrix} 10 \\ 18 \\ 1 \end{pmatrix}.$$

Перевіряємо виконання умови $\tilde{\mathbf{w}}^T \mathbf{z}^{(j)} > 0$ для всіх векторів навчальної вибірки:

$$\tilde{\mathbf{w}}^T \mathbf{z} = (425 \quad 389 \quad 371 \quad 363 \quad 307 \quad -263 \quad -309 \quad -271 \quad -291 \quad -271).$$

Проводимо модифікацію вагових коефіцієнтів

$$\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(l)},$$

де $\varphi = 1, l = 6$.

Тоді

$$\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(6)} = \begin{pmatrix} 10 \\ 18 \\ 1 \end{pmatrix} + \begin{pmatrix} -10 \\ -9 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 9 \\ 0 \end{pmatrix}.$$

Перевіряємо виконання умови $\tilde{\mathbf{w}}^T \mathbf{z}^{(j)} > 0$ для всіх векторів навчальної вибірки:

$$\tilde{\mathbf{w}}^T \mathbf{z} = (\ 162 \ 144 \ 135 \ 126 \ 108 \ -81 \ -99 \ -90 \ -90 \ -90 \).$$

Проводимо модифікацію вагових коефіцієнтів $\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(l)}$, де $\varphi = 1, l = 6$.

$$\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(6)} = \begin{pmatrix} 0 \\ 9 \\ 0 \end{pmatrix} + \begin{pmatrix} -10 \\ -9 \\ -1 \end{pmatrix} = \begin{pmatrix} -10 \\ 0 \\ -1 \end{pmatrix}.$$

Перевіряємо виконання умови $\tilde{\mathbf{w}}^T \mathbf{z}^{(j)} > 0$ для всіх векторів навчальної вибірки:

$$\tilde{\mathbf{w}}^T \mathbf{z} = (\ -101 \ -101 \ -101 \ -111 \ -91 \ 101 \ 111 \ 91 \ 111 \ 91 \).$$

Проводимо модифікацію вагових коефіцієнтів $\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(l)}$, де $\varphi = 1, l = 1$.

$$\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(1)} = \begin{pmatrix} -10 \\ 0 \\ -1 \end{pmatrix} + \begin{pmatrix} 10 \\ 18 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 18 \\ 0 \end{pmatrix}.$$

Перевіряємо виконання умови $\tilde{\mathbf{w}}^T \mathbf{z}^{(j)} > 0$ для всіх векторів навчальної вибірки:

$$\tilde{\mathbf{w}}^T \mathbf{z} = (\quad 324 \quad 288 \quad 270 \quad 252 \quad 216 \quad -162 \quad -198 \quad -180 \quad -180 \quad -180 \quad).$$

Проводимо модифікацію вагових коефіцієнтів $\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(l)}$, де $\varphi = 1$, $l = 6$.

$$\tilde{\mathbf{w}} = \tilde{\mathbf{w}} + \varphi \mathbf{z}^{(6)} = \begin{pmatrix} 0 \\ 18 \\ 0 \end{pmatrix} + \begin{pmatrix} -10 \\ -9 \\ -1 \end{pmatrix} = \begin{pmatrix} -10 \\ 9 \\ -1 \end{pmatrix}.$$

Перевіряємо виконання умови $\tilde{\mathbf{w}}^T \mathbf{z}^{(j)} > 0$ для всіх векторів навчальної вибірки:

$$\tilde{\mathbf{w}}^T \mathbf{z} = (\quad 61 \quad 43 \quad 34 \quad 15 \quad 17 \quad 20 \quad 12 \quad 1 \quad 21 \quad 1 \quad).$$

Таким чином, лінійне вирішальне правило для розпізнавання двох образів має вигляд

$$y = \text{sign}(-10x_1 + 9x_2 - 1).$$

Графічний вигляд сформованого правила подано на рис. 2.3.2.

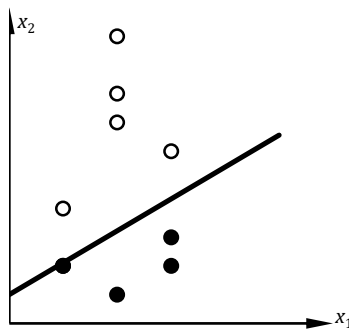


Рисунок 2.3.2 – Графічний вигляд лінійного вирішального правила

Лінійні вирішальні правила можуть застосовуватися для чітких образів, тобто для класів розпізнавання, що не перетинаються. Якщо класи перетинаються, для них навчальна вибірка не є нормально розподіленою, а вектори-реалізації образів не є лінійно розділеними.

2.4 Метод K -найближчих сусідів

Нехай навчальна матриця $\|x_m^{(j)}\|$, де $m = 1, 2, \dots, M$; $j = 1, 2, \dots, n$, сформована для алфавіту з M класів розпізнавання $X_1^o, X_2^o, \dots, X_M^o$. Метод найближчих сусідів визначає належність вхідної реалізації y до одного з цих класів відповідно до належності найближчої (сусідньої) реалізації з навчальної матриці:

$$\begin{aligned} & \arg \min_{k=1..m} \min_{j=1..n} \text{dist}(x_k^{(j)}, y) = \\ & = \arg \min_{k=1..m} \min_{j=1..n} \sqrt{\sum_{i=1}^N (x_{k,i}^{(j)} - y_i)^2}. \end{aligned}$$

Узагальненням такого методу є метод K -найближчих сусідів [18, 25], який полягає в тому, що при визначенні належності вектора y аналізуються не одна, а декілька найближчих реалізацій навчальної матриці.

Вхідні дані алгоритму методу K -найближчих сусідів:

N – потужність словника ознак розпізнавання;

m – потужність алфавіту класів розпізнавання;

n_k – обсяг навчальної матриці класу X_k^o ;

$\|\mathbf{x}_k^{(j)}\|$ – навчальна матриця класу X_k^o ($k = 1, 2, \dots, m$,
 $j = 1, 2, \dots, n_k$);

$\mathbf{y} = (y_1, y_2, \dots, y_N)$ – вхідна реалізація, для якої необхідно прийняти рішення про її належність до одного з класів X_k^o ;

K – кількість найближчих реалізацій, які аналізуються при визначенні належності \mathbf{y} .

Вихідні дані:

K^* – номер класу, до якого належить вхідна реалізація \mathbf{y} .

Крок 1 Ініціалізуємо масив s_k ($k = 1, 2, \dots, m$), елементи якого дорівнюють нулю.

Крок 2 Визначаємо відстань між реалізацією \mathbf{y} та елементами навчальної матриці $\|\mathbf{x}_k^{(j)}\|$:

$$\text{dist}(\mathbf{x}_k^{(j)}, \mathbf{y}) = \sqrt{\sum_{i=1}^N (x_{k,i}^{(j)} - y_i)^2}.$$

Крок 3 Ініціалізуємо лічильник $t = 0$.

Крок 4 Визначаємо мінімальний елемент масиву $\text{dist}(\mathbf{x}_k^{(j)}, \mathbf{y})$ та номер класу, що йому відповідає:

$$(k^*, j^*) = \arg \min_{k=1..m} \min_{j=1..n_k} \text{dist}(\mathbf{x}_k^{(j)}, \mathbf{y}).$$

Крок 5 Збільшуємо значення елемента s_{k^*} на одиницю:

$$s_{l^*} = s_{l^*} + 1.$$

Крок 6 Видаляємо елемент $dist(x_{k^*}^{(j^*)}, y)$ з масиву відстаней:

$$\begin{aligned} dist(x_{k^*}^{(j^*)}, y) &= \emptyset, \\ n_{k^*} &= n_{k^*} - 1. \end{aligned}$$

Крок 7 Збільшуємо лічильник $t = t + 1$.

Крок 8 Якщо $t < K$, то виконується крок 4, інакше – крок 9.

Крок 9 Визначаємо номер класу, до якого належить вхідна реалізація y :

$$K^* = arg \max_{k=1..m} s_k.$$

Крок 10 ЗУПИН.

Як приклад розглянемо задачу класифікації реалізації

$$y = (y_1, y_2) = (13, 16),$$

що була визначена в прикладі з розділу 2.1 як еталон одного з підкласів класу X_3^0 .

У табл. 2.4.1 наведена навчальна матриця, яка за структурою аналогічна матриці, наведеній у табл. 2.1.1. При цьому на відміну від табл. 2.1.1 розділ реалізацій на підкласи класів розпізнавання не здійснювався.

Таблиця 2.4.1 – Вхідні дані

Клас X_1^o		Клас X_2^o		Клас X_3^o	
x_1	x_2	x_1	x_2	x_1	x_2
14	12	11	14	12	15
12	14	11	13	14	16
13	14	10	12	13	17
12	12	9	12	10	11
14	13	9	14	9	9
12	9	17	14	11	10
14	11	20	15	6	14
14	9	20	11	6	13
13	11	18	12	8	13
12	10	20	13	8	12

Аналіз табл. 2.4.1 показує, що потужність словника ознак розпізнавання дорівнює $N = 2$, а потужність алфавіту класів розпізнавання – $m = 3$. При цьому обсяги навчальних матриць відповідно становлять $n_1 = n_2 = n_3 = 10$.

Сформуємо масив відстаней між реалізацією \mathbf{y} та елементами навчальної матриці (табл. 2.4.2).

Таблиця 2.4.2 – Матриця відстаней до реалізації \mathbf{y}

$dist(\mathbf{x}_1^{(j)}, \mathbf{y})$	4.12	2.24	2.00	4.12	3.16	7.07	5.10	7.07	5.00	6.08
$dist(\mathbf{x}_2^{(j)}, \mathbf{y})$	2.83	3.61	5.00	5.66	4.47	4.47	7.07	8.60	6.40	7.62
$dist(\mathbf{x}_3^{(j)}, \mathbf{y})$	1.41	1.00	1.00	5.83	8.06	6.32	7.28	7.62	5.83	6.40

Аналіз таблиці 2.4.2 показує, що найбличою до реалізації \mathbf{y} є реалізація $\mathbf{x}_3^{(2)}$ та $\mathbf{x}_3^{(3)}$. При цьому

$$\begin{aligned} & \text{dist}(x_3^{(3)}, y) = \\ & = \text{dist}(x_3^{(3)}, y) = \min_{k=1..m} \left[\min_{j=1..n_k} \text{dist}(x_k^{(j)}, y) \right] = 1.00. \end{aligned}$$

Таким чином, при $K = 1$ та $K = 2$ реалізація $y = (13, 16)$ належить до класу X_3^0 . При збільшенні значення K номер класу, до якого належить реалізація y , змінюється. Так, при $K = 9$ або $K = 10$ реалізація $y = (13, 16)$ належить до класу X_1^0 . На рис. 2.4.1 наведено області рішень, що формуються за методом найближчих сусідів за навчальною матрицею (табл. 2.4.1).

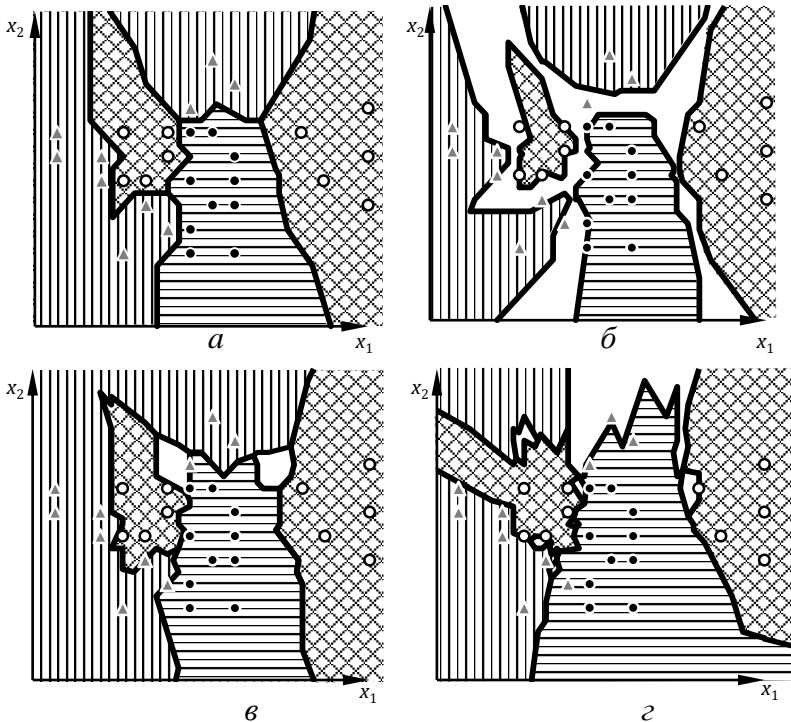


Рисунок 2.4.1 – Результати формування вирішальних правил k -найближчих сусідів: $a - K = 1$, $b - K = 2$, $v - K = 3$, $z - K = 9$

На рис. 2.4.1 для позначення належності вхідних реалізацій використано такі позначення $\circ (x \in X_1^o)$, $\bullet (x \in X_2^o)$ та $\Delta (x \in X_3^o)$. Область рішень класу X_1^o позначена горизонтальною штриховкою Ξ , класу X_2^o – подвійною штриховкою \boxtimes , класу X_3^o – вертикальною штриховкою III . При цьому об'єднання областей рішень усіх класів розпізнавання повністю збігається з простором ознак розпізнавання лише для випадку, коли параметр $K = 1$ (рис. 2.4.1 а). При $K > 1$ з'являються області, в яких прийняти рішення про належність реалізацій неможливо, оскільки вони мають однакову кількість найближчих сусідів із різних класів. При цьому така особливість має більший прояв при використанні парних значень K . У нашому випадку при збільшенні значення K із двох (рис. 2.4.2 б) до трьох (рис. 2.4.2 в) обсяг таких областей простору ознак розпізнавання значно скорочується. Використання значень K , які наближаються до значень обсягів навчальних матриць класів (рис. 2.4.2 з), зменшує ефективність вирішальних правил, сформованих за методом найближчих сусідів. Особливо це проявляється за наявності в алфавіті класів, реалізації яких характеризуються різним ступенем компактності. Для класів, наведених у навчальній матриці «подібними» векторами-реалізаціями, вирішальні правила будуть характеризуватися високим значенням помилки другого роду, тобто високою ймовірністю віднесення до такого класу реалізацій інших класів. Для класів, наведених у навчальній матриці «неподібними» векторами-реалізаціями, високою буде ймовірність прийняття рішення про те, що вони не належать до відповідного класу, тобто є помилкою першого роду.

2.5 Метод потенціальних функцій

Нехай $m = 2$, тобто маємо алфавіт із двох класів: $\{X_1^o, X_2^o\}$. Відома координата вершини вектора-реалізації $\mathbf{x}^{(j)}$ в просторі ознак. Надамо точки, що знаходиться в цій координаті, потенціал $+q^{(j)}$, якщо $\mathbf{x}^{(j)} \in X_1^o$, і потенціал $-q^{(j)}$, якщо $\mathbf{x}^{(j)} \in X_2^o$. Тоді може мати місце таке вирішальне правило, що створюється n потенціалами [18, 24, 25]:

$$g(\mathbf{x}) = \sum_{j=1}^n q^{(j)} K(\mathbf{x}, \mathbf{x}^{(j)}), \quad (2.5.1)$$

де $K(\mathbf{x}, \mathbf{x}^{(j)})$ – потенціальна функція.

За визначенням потенціальна функція є спадною при збільшенні відстані поточної координати \mathbf{x} від вершини $\mathbf{x}^{(j)}$ і монотонно спадною до 0 при $|\mathbf{x} - \mathbf{x}^{(j)}| \rightarrow \infty$.

Вибір потенціальної функції не є простою задачею, яку доводиться розв'язувати розробнику інформаційного забезпечення систем розпізнавання. Від вибору потенціальної функції залежить як збіжність алгоритму, так і його точність.

На рисунку 2.5.1 наведено приклад побудови вирішального правила за методом потенціальних функцій.

Таким чином, процес навчання за методом потенційних функцій полягає в побудові вирішального правила (3.5.1). Тоді процес розпізнавання за цим методом може здійснюватися за таким алгоритмом: якщо в точці \mathbf{x} , де

знаходиться реалізація, що розпізнається, обчислюємо $g(x)$, отримуємо правило

$$IF g(x) > 0 THEN x \in X_1^o ELSE x \in X_2^o .$$

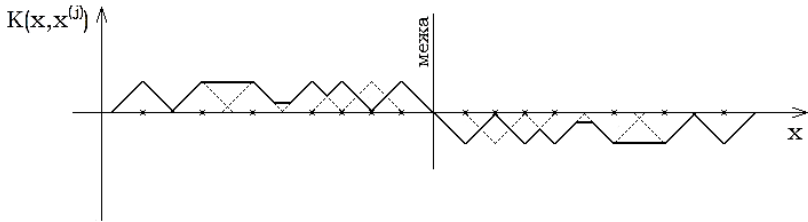


Рисунок 2.5.1 – Графічне відображення вирішального правила, сформованого за методом потенціальних функцій

При великих обсягах навчальної вибірки доцільно обчислювати не вирішальне правило (2.5.1), а оцінювати розподільну межу для класів розпізнавання (рис. 2.5.1). У цьому випадку для підвищення оперативності розпізнавання достатньо лише визначити, де знаходиться реалізація – справа чи зліва від межі.

Розглянемо алгоритм формування вирішальних правил за методом потенціальних функцій для загального випадку.

Вхідні дані:

N – потужність словника ознак розпізнавання;

m – потужність алфавіту класів розпізнавання;

$\|x_k^{(j)}\|$ – навчальна матриця класу X_k^o ($l_k = 1, 2, \dots, m$, $j = 1, 2, \dots, n_k$);

n_k – обсяг навчальної матриці класу X_k^o ;

$y = (y_1, y_2, \dots, y_N)$ – вхідна реалізація, для якої необхідно прийняти рішення про її належність до одного з класів X_k^o .

Вихідні дані:

k^* – номер класу, до якого належить вхідна реалізація \mathbf{y} .

Крок 1 Визначаємо значення потенційної функції

$$g_k^{(j)}(\mathbf{y}) = K(\mathbf{x}, \mathbf{x}_k^{(j)}).$$

Крок 2 Обчислюємо значення потенціалу, що формують реалізації класу X_k^o :

$$G_k(\mathbf{y}) = \sum_{j=1}^{n_k} g_k^{(j)}(\mathbf{y}).$$

Крок 3 Визначаємо максимальне значення $G_k(\mathbf{y})$ та відповідний йому номер класу, до якого належить \mathbf{y} :

$$k^* = \arg \max_k G_k(\mathbf{y}).$$

Крок 4 ЗУПИН

Подальше узагальнення методу потенційних функцій полягає в його рандомізації шляхом побудови роздільної межі за навчальною вибіркою із використанням процедур стохастичної апроксимації.

Переваги:

- простота реалізації, що ґрунтується на фізичних законах електростатики (чим далі від заряду, тим вплив його менший).

Недоліки:

- необхідність обґрунтування вибору потенціальної функції, що впливає безпосередньо на збіжність алгоритму навчання та на достовірність розпізнавання в режимі екзамену;

- метод носить модельний характер, оскільки орієнтований на класи, що не перетинаються у просторі ознак розпізнавання.

2.6 Структурні методи

При структурних методах, реалізації образу характеризуються не множиною їх значень (наприклад, навчальна матриця типу об'єкт-властивість) або їх відношень (наприклад, навчальна матриця типу відношення-подібність), а їх структурою [26, 27].

Наприклад, на рисунку 2.6.1. наведено зображення, які необхідно описати їх структурою.

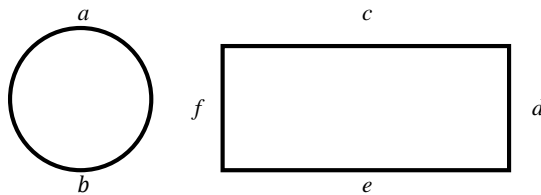


Рисунок 2.6.1 – Вхідні дані для структурного методу

На рисунку 2.6.2 показано ієрархічну структуру зображень (рис. 2.6.1).

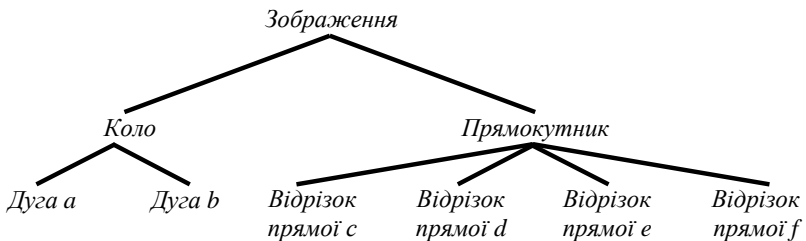


Рисунок 2.6.2 – Ієрархічна структура

Структурний підхід ґрунтується на аналогії між структурою реалізації образу та синтаксисом мов, тому його часто називають *лінгвістичним*.

Етап розпізнавання за структурними методами полягає у розпізнаванні непохідних елементів реалізацій образу і синтаксичному аналізі (граматичний розбір) “речення”, що описує образ. Приклад формування “речення” показано на рис. 2.6.3.

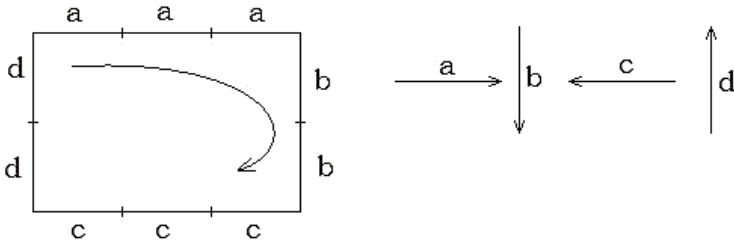


Рисунок 2.6.3 – Граматичний розбір структури

Структурні методи знайшли широке використання при сегментації (при визначенні меж) різних текстів, при усномовному розпізнаванні за послідовністю фонем тощо.

Головна перевага структурних методів – це можливість подати велику кількість реалізацій у вигляді малопотужної множини непохідних елементів і граматичних правил.

Недоліки:

- відсутність прямих вирішальних правил;
- обмеженість використання через те, що аналізується не весь образ, а лише його фрагмент, що ускладнює процес прийняття рішень.

2.7 Метод опорних векторів

Метод опорних векторів (*Support Vector Machine, SVM*) набув популярності при розв'язанні задач класифікаційного, факторного класифікаційного та регресійного аналізу кілька років тому [28-31]. Характерними властивостями цього методу є те, що формування вирішальних правил розглядається як задача опуклої оптимізації, в якій широко використовуються множники Лагранжа. Існує також альтернативний метод релевантних векторів (*Relevance Vector Machine, RVM*), що на відміну від методу опорних векторів використовує апостеріорні оцінки точнісних характеристик вирішальних правил.

Розглянемо задачу формування *лінійних* вирішальних правил у вигляді (1.3.9) для двох класів розпізнавання X_1^o та X_2^o за методом опорних векторів. При цьому $f_i(\mathbf{x})$ є певними (фіксованими) функціями, що трансформують простір первинних ознак розпізнавання. Навчальна матриця подається в альтернативному вигляді $\|\mathbf{x}^{(j)}\|$, тобто реалізації класів X_1^o та X_2^o знаходяться в одному двовимірному масиві розмірністю $(n_1 + n_2) \times N$, де n_k – кількість реалізацій класу X_k^o . Додатковий одновимірний масив ідентифікаторів класу $\|t^{(j)}\|$ розмірністю $(n_1 + n_2)$ формується за правилом

$$t^{(j)} = \mathbf{sign}(d(\mathbf{x}^{(j)})) = \begin{cases} +1, & d(\mathbf{x}^{(j)}) > 0, \\ -1, & d(\mathbf{x}^{(j)}) < 0, \end{cases} = \begin{cases} +1, & \mathbf{x}^{(j)} \in X_1^o, \\ -1, & \mathbf{x}^{(j)} \in X_2^o. \end{cases} \quad (2.7.1)$$

Аналогічну постановку задачі можна спостерігати, наприклад, для такого типу штучних нейронних мереж, як одношаровий перцептрон, алгоритм навчання якого дозволяє сформувати вирішальні правила за скінченну кількість кроків оптимізації вагових коефіцієнтів. При цьому значення цих коефіцієнтів залежить не лише від вмісту навчальної матриці, але й, наприклад, від початкових значень коефіцієнтів, а також від порядку реалізацій у навчальній матриці. Таким чином, у випадку лінійної роздільності класів можна сформувати множину лінійних вирішальних правил, що будуть безпомилковими за навчальною матрицею. Метод опорних векторів дозволяє серед таких правил обрати таке, що характеризується мінімальною помилкою класифікації реалізацій, що не увійшли до навчальної матриці (помилкою узагальнення). При цьому використовується поняття функціонального зазору (*functional margin*) – найменшої відстані від роздільної функції $d(\mathbf{x}) = 0$ до найближчої реалізації з навчальної матриці (рис. 2.7.1).

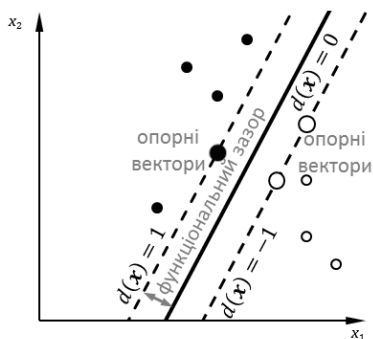


Рисунок 2.7.1 – Роздільна функція $d(\mathbf{x}) = 0$, сформована за методом опорних векторів

У процесі оптимізації параметрів роздільної функції $d(\mathbf{x}) = 0$ за методом опорних векторів функціональний зазор максимізується, а найближчі до неї реалізації з навчальної матриці одержують назву опорних векторів (на рис. 2.7.1 вони позначені збільшеними точками). Один із підходів до розв'язання задачі максимізації функціонального зазору використовував статистичну модель розподілу векторів-реалізацій \mathbf{x} на базі оцінювання щільності Парцель-Розенблата для гаусівських ядрових функцій, що мають загальний параметр σ^2 . Максимальний функціональний зазор досягався шляхом мінімізації ймовірності помилки класифікації, що проводилася аналогічно до байесівських класифікаторів.

Відстань від вектора $\mathbf{x}^{(j)}$ до роздільної функції $d(\mathbf{x}) = 0$ обчислюється за формулою

$$\frac{|d(\mathbf{x}^{(j)})|}{\sqrt{\sum_{i=1}^K w_i^2}} = \frac{|\sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)})|}{\sqrt{\sum_{i=1}^K w_i^2}}, \quad (2.7.2)$$

що з урахуванням (2.7.1) перетворюється на

$$\frac{|\sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)})|}{\sqrt{\sum_{i=1}^K w_i^2}} = \frac{t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)})}{\sqrt{\sum_{i=1}^K w_i^2}}, \quad (2.7.3)$$

а задача максимізації функціонального зазору набирає вигляду

$$\begin{aligned} & \arg \max_{\mathbf{w}} \left\{ \min_j \left[\frac{t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)})}{\sqrt{\sum_{i=1}^K w_i^2}} \right] \right\} = \\ & = \arg \max_{\mathbf{w}} \left\{ \frac{\min_j \left[t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)}) \right]}{\sqrt{\sum_{i=1}^K w_i^2}} \right\}. \end{aligned} \quad (2.7.4)$$

Застосуємо лінійне перетворення коефіцієнтів $\mathbf{w} = \lambda \cdot \mathbf{w}$, де λ – таке число, що для всіх опорних векторів $\mathbf{x}^{(o)}$ буде виконуватися рівність

$$t^{(o)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(o)}) = 1. \quad (2.7.5)$$

При цьому всі реалізації навчальної матриці $\|\mathbf{x}^{(j)}\|$ будуть задовольняти обмеження

$$t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)}) \geq 1. \quad (2.7.6)$$

Таким чином, задача (2.7.4) зводиться до задачі

$$\arg \max_{\mathbf{w}} \left\{ \frac{1}{\sqrt{\sum_{i=1}^K w_i^2}} \right\} = \arg \min_{\mathbf{w}} \left\{ \sqrt{\sum_{i=1}^K w_i^2} \right\},$$

яку можна подати у вигляді

$$\arg \min_{\mathbf{w}} \left\{ \frac{1}{2} \sum_{i=1}^K w_i^2 \right\}. \quad (2.7.7)$$

Для розв'язання задачі (2.7.7) з обмеженнями (2.7.6) застосовується механізм множників Лагранжа $\mathbf{a} = \|a^{(j)}\|$, ($a^{(j)} \geq 0$). При цьому кількість множників збігається з кількістю обмежень (2.7.6) або кількістю реалізацій навчальної матриці $\|\mathbf{x}^{(j)}\|$. Відповідна функція Лагранжа, що потребує подальшої максимізації, має вигляд

$$L(\mathbf{w}, \mathbf{a}) = \frac{1}{2} \sum_{i=1}^k w_i^2 - \sum_{j=1}^{n_1+n_2} a^{(j)} \left\{ t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)}) - 1 \right\}. \quad (2.7.8)$$

Прирівнюючи часткові похідні функції за w_i до нуля, визначаємо, що

$$w_i = \sum_{j=1}^{n_1+n_2} a^{(j)} \cdot t^{(j)} \cdot f_i(\mathbf{x}^{(j)}), \text{ якщо } i = 1..K; \quad (2.7.9)$$

$$0 = \sum_{j=1}^{n_1+n_2} a^{(j)} \cdot t^{(j)} \cdot f_0(\mathbf{x}^{(j)}) = \sum_{j=1}^{n_1+n_2} a^{(j)} \cdot t^{(j)}, \text{ якщо } i = 0. \quad (2.7.10)$$

Таким чином, після заміни w_i функція (2.7.8) набирає вигляду

$$L(\mathbf{a}) = \sum_{j=1}^{n_1+n_2} a^{(j)} -$$

$$-\frac{1}{2} \sum_{i=1}^K \sum_{j=1}^{n_1+n_2} \sum_{l=1}^{n_1+n_2} a^{(j)} \cdot t^{(j)} \cdot a^{(l)} \cdot t^{(l)} \cdot f_i(\mathbf{x}^{(l)}) \cdot f_i(\mathbf{x}^{(j)}), \quad (2.7.11)$$

а при її максимізації враховуються такі обмеження:

$$\begin{cases} a^{(j)} \geq 0; \\ \sum_{j=1}^{n_1+n_2} a^{(j)} \cdot t^{(j)} = 0. \end{cases} \quad (2.7.12)$$

Сума добутків $\sum_{i=1}^K f_i(\mathbf{x}) \cdot f_i(\mathbf{x}')$ визначає функцію ядра $K(\mathbf{x}, \mathbf{x}')$.

Як приклад розглянемо задачу формування лінійного вирішального правила $d(\mathbf{x}) = \sum_{i=0}^K w_i f_i(\mathbf{x})$ за методом опорних векторів для задачі подання функції логічного множення AND. Проведемо такі перетворення таблиці істинності функції AND (табл. 2.7.1):

Таблиця 2.7.1 – Формування вхідних даних

	x_1	x_2	$x_1 AND x_2$
$\mathbf{x}^{(1)}$	0	0	0
$\mathbf{x}^{(2)}$	0	1	0
$\mathbf{x}^{(3)}$	1	0	0
$\mathbf{x}^{(4)}$	1	1	1

 \Rightarrow

x_1	x_2	t
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

Аналіз табл. 2.7.1 показує, що кількість ознак розпізнавання дорівнює $N = 2$, кількість векторів-реалізацій

навчальної матриці для класу X_1^0 дорівнює $n_1 = 1$, для класу $X_2^0 - n_2 = 3$. Таким чином, при формуванні функції Лагранжа необхідно ввести $n_1 + n_2 = 4$ множники $a^{(j)}$, де $j = 1, 2, \dots, (n_1 + n_2 = 4)$. Нехай $f_i(x) = x_i$ для $i = 1, (K = N = 2)$ та $f_0(x) = 1$, функція (2.7.11) набирає вигляду:

$$\begin{aligned}
 L(\mathbf{a}) &= \sum_{j=1}^4 a^{(j)} - \frac{1}{2} \sum_{i=1}^2 \sum_{j=1}^4 \sum_{l=1}^4 a^{(j)} \cdot t^{(j)} \cdot a^{(l)} \cdot t^{(l)} \cdot x_i^{(l)} \cdot x_i^{(j)} = \\
 &= a^{(1)} + a^{(2)} + a^{(3)} + a^{(4)} - (a^{(1)})^2 - (a^{(2)})^2 - (a^{(3)})^2 - \\
 &\quad - (a^{(4)})^2 - 2a^{(1)}a^{(4)} + 2a^{(2)}a^{(3)}, \quad (2.7.13)
 \end{aligned}$$

а умови (2.7.12) –

$$\left\{ \begin{array}{l} a^{(1)} \geq 0, \\ a^{(2)} \geq 0, \\ a^{(3)} \geq 0, \\ a^{(4)} \geq 0, \\ -a^{(1)} - a^{(2)} - a^{(3)} + a^{(4)} = 0. \end{array} \right. \quad (2.7.14)$$

Подамо останню умову (2.7.14) у вигляді $a^{(4)} = a^{(1)} + a^{(2)} + a^{(3)}$, тоді (2.7.13) спрощується до

$$\begin{aligned}
 L(\mathbf{a}) &= 2a^{(1)} + 2a^{(2)} + 2a^{(3)} - 4(a^{(1)})^2 - 2(a^{(2)})^2 - \\
 &\quad - 2(a^{(3)})^2 - 4a^{(1)}a^{(2)} - 4a^{(1)}a^{(3)}. \quad (2.7.15)
 \end{aligned}$$

Аналіз (2.7.15) показує, що квадратична форма функції цілі

$$-4(a^{(1)})^2 - 2(a^{(2)})^2 - 2(a^{(3)})^2 - 4a^{(1)}a^{(2)} - 4a^{(1)}a^{(3)} \leq 0$$

є невід'ємною, а рівність можлива лише для нульових значень $a^{(j)}$. Таким чином, квадратична форма є від'ємно-визначеною, і максимізація (2.7.15) за умов $a^{(j)} \geq 0$ є задачею *квадратичного* програмування. Зазначимо необхідні і достатні умови існування сідлової точки у вигляді:

$$\begin{cases} \frac{\partial L(a^{(1)})}{\partial a^{(1)}} = 2 - 8a^{(1)} - 4a^{(2)} - a^{(3)} \leq 0, \\ \frac{\partial L(a^{(2)})}{\partial a^{(2)}} = 2 - 4a^{(1)} - 4a^{(2)} \leq 0, \\ \frac{\partial L(a^{(3)})}{\partial a^{(3)}} = 2 - 4a^{(1)} - 4a^{(3)} \leq 0, \end{cases} \quad (2.7.16)$$

та

$$\begin{cases} a^{(1)} \frac{\partial L(a^{(1)})}{\partial a^{(1)}} = a^{(1)}(2 - 8a^{(1)} - 4a^{(2)} - a^{(3)}) = 0, \\ a^{(2)} \frac{\partial L(a^{(2)})}{\partial a^{(2)}} = a^{(2)}(2 - 4a^{(1)} - 4a^{(2)}) = 0, \\ a^{(3)} \frac{\partial L(a^{(3)})}{\partial a^{(3)}} = a^{(3)}(2 - 4a^{(1)} - 4a^{(3)}) = 0. \end{cases} \quad (2.7.17)$$

Введемо додаткові невід'ємні змінні $v^{(1)}$, $v^{(2)}$, $v^{(3)}$ у вираз (2.7.16):

$$\begin{cases} 2 - 8a^{(1)} - 4a^{(2)} - 4a^{(3)} + v^{(1)} = 0, \\ 2 - 4a^{(1)} - 4a^{(2)} + v^{(2)} = 0, \\ 2 - 4a^{(1)} - 4a^{(3)} + v^{(3)} = 0, \end{cases} \quad (2.7.18)$$

тоді рівняння (2.7.17) набувають вигляду

$$a^{(1)}v^{(1)} = 0, a^{(2)}v^{(2)} = 0, a^{(3)}v^{(3)} = 0. \quad (2.7.19)$$

Таким чином, задача щодо функції цілі $z = 0 \rightarrow \max$ за умов (2.7.18), (2.7.19) та $a^{(j)} \geq 0, v^{(j)} \geq 0$ є задачею лінійного програмування.

Розв'яжемо таку задачу квадратичним симплекс-методом, в якому в симплекс-таблицях на кожному кроці оптимізації при виборі нового базису обов'язково вимагається виконання умов (2.7.19). Введемо штучний базис

$$\{p^{(1)}, p^{(2)}, p^{(3)}\},$$

де $p^{(j)} \geq 0$,

тоді

$$z = -Mp^{(1)} - Mp^{(2)} - Mp^{(3)} \rightarrow \max$$

$$\begin{cases} 8a^{(1)} + 4a^{(2)} + 4a^{(3)} - v^{(1)} + p^{(1)} = 2, \\ 4a^{(1)} + 4a^{(2)} - v^{(2)} + p^{(2)} = 2, \\ 4a^{(1)} + 4a^{(3)} - v^{(3)} + p^{(3)} = 2, \\ a^{(j)} \geq 0, v^{(j)} \geq 0, p^{(j)} \geq 0, \text{ де } j = 1..3. \end{cases} \quad (2.7.20)$$

Оберемо $M = 100$, оскільки таке значення набагато більше за інші коефіцієнти задачі. При формуванні симплекс-таблиць збережемо позначення (2.7.20) та введемо дві додаткові умови:

1. Згідно з (2.7.19) на кожному кроці оптимізації до базису не можна одночасно поміщати змінні $a^{(j)}$ та $v^{(j)}$ з однаковими індексами.

2. Оптимальний план не повинен містити штучних змінних $p^{(j)}$.

У табл. 2.7.2–2.7.6 наведені симплекс-таблиці розв'язку задачі.

Таблиця 2.7.2 – Симплекс-таблиця першого кроку оптимізації

C_i	0	0	0	0	0	0	-100	-100	-100			
B	$a^{(1)}$	$a^{(2)}$	$a^{(3)}$	$v^{(1)}$	$v^{(2)}$	$v^{(3)}$	$p^{(1)}$	$p^{(2)}$	$p^{(3)}$	RHS	R	
-100	$p^{(1)}$	8	4	4	-1	0	0	1	0	0	2	0.25
-100	$p^{(2)}$	4	4	0	0	-1	0	0	1	0	2	0.5
-100	$p^{(3)}$	4	0	4	0	0	-1	0	0	1	2	0.5
	z_i	-1600	-800	-800	100	100	100	-100	-100	-100	-600	
	$C_i - z_i$	1600	800	800	-100	-100	-100	0	0	0		

Аналіз табл. 2.7.2 показує, що розв'язувальними є перший стовпець ($a^{(1)}$) та перший рядок ($p^{(1)}$).

Таблиця 2.7.3 – Симплекс-таблиця другого кроку оптимізації

C_i	0	0	0	0	0	0	-100	-100	-100			
B	$a^{(1)}$	$a^{(2)}$	$a^{(3)}$	$v^{(1)}$	$v^{(2)}$	$v^{(3)}$	$p^{(1)}$	$p^{(2)}$	$p^{(3)}$	RHS	R	
0	$a^{(1)}$	1	0.5	0.5	-0.1	0	0	0.13	0	0	0.25	-2
-100	$p^{(2)}$	0	2	-2	0.5	-1	0	-0.5	1	0	1	2
-100	$p^{(3)}$	0	-2	2	0.5	0	-1	-0.5	0	1	1	2
	z_i	0	0	0	-100	100	100	100	-100	-100	-200	
	$C_i - z_i$	0	0	0	100	-100	-100	-200	0	0		

Аналіз табл. 2.7.3 показує, що розв'язувальним є четвертий стовпець ($v^{(1)}$), який згідно з умовою (2.7.19) можна помістити в базис лише за відсутності в ньому ($a^{(1)}$).

На наступному кроці проведемо заміну першого стовпця ($a^{(1)}$) на стовпець ($v^{(1)}$).

Таблиця 2.7.4 – Симплекс-таблиця третього кроку оптимізації

C_i	0	0	0	0	0	0	0	-100	-100	-100		
B	$a^{(1)}$	$a^{(2)}$	$a^{(3)}$	$v^{(1)}$	$v^{(2)}$	$v^{(3)}$	$p^{(1)}$	$p^{(2)}$	$p^{(3)}$	RHS	R	
0	$v^{(1)}$	-8	-4	-4	1	0	0	-1	0	0	-2	-
-100	$p^{(2)}$	4	4	0	0	-1	0	0	1	0	2	0.5
-100	$p^{(3)}$	4	0	4	0	0	-1	0	0	1	2	-
	z_i	-800	-400	-400	0	100	100	0	-100	-100	-400	
	$C_i - z_i$	800	400	400	0	-100	-100	-100	0	0		

Аналіз табл. 2.7.4 показує, що розв'язувальним знову є перший стовпець ($a^{(1)}$), тобто має місце зациклювання оптимізації. Для виходу з такої ситуації оберемо розв'язувальними другий стовпець ($a^{(2)}$) та другий рядок ($p^{(2)}$).

Таблиця 2.7.5 – Симплекс-таблиця четвертого кроку оптимізації

C_i	0	0	0	0	0	0	0	-100	-100	-100		
B	$a^{(1)}$	$a^{(2)}$	$a^{(3)}$	$v^{(1)}$	$v^{(2)}$	$v^{(3)}$	$p^{(1)}$	$p^{(2)}$	$p^{(3)}$	RHS	R	
0	$v^{(1)}$	-4	0	-4	1	-1	0	-1	1	0	0	-
0	$a^{(2)}$	1	1	0	0	-0.3	0	0	0.25	0	0.5	-
-100	$p^{(3)}$	4	0	4	0	0	-1	0	0	1	2	0.5
	z_i	-400	0	-400	0	0	100	0	0	-100	-200	
	$C_i - z_i$	400	0	400	0	0	-100	-100	-100	0		

Аналіз табл. 2.7.5 показує, що розв'язувальним є третій стовпець ($a^{(3)}$) та третій рядок ($p^{(3)}$).

Таблиця 2.7.6 – Симплекс-таблиця п'ятого кроку оптимізації

C_i	0	0	0	0	0	0	0	-100	-100	-100		
B	$a^{(1)}$	$a^{(2)}$	$a^{(3)}$	$v^{(1)}$	$v^{(2)}$	$v^{(3)}$	$p^{(1)}$	$p^{(2)}$	$p^{(3)}$	RHS		
0	$v^{(1)}$	0	0	0	1	-1	-1	-1	1	1	2	
0	$a^{(2)}$	1	1	0	0	-0.3	0	0	0.25	0	0.5	
0	$a^{(3)}$	1	0	1	0	0	-0.3	0	0	0.25	0.5	
	z_i	0	0	0	0	0	0	0	0	0	0	
	$C_i - z_i$	0	0	0	0	0	-100	-100	-100			

Аналіз табл. 2.7.6 показує, що був знайдений оптимальний план у вигляді $v^{(1)} = 2$, $a^{(2)} = 0.5$, $a^{(3)} = 0.5$. Тоді за (2.7.16) $a^{(1)} = 0$, $v^{(2)} = 0$, $v^{(3)} = 0$. Для такого плану оптимальне значення функції цілі $L(\mathbf{a}) = 1$.

Для визначення значення змінної a_4 застосуємо останню умову (2.7.14)

$$a^{(4)} = a^{(1)} + a^{(2)} + a^{(3)} = 1.$$

Для визначення значень коефіцієнтів w_1 та w_2 застосуємо (2.7.9):

$$w_1 = \sum_{j=1}^4 a^{(j)} \cdot t^{(j)} \cdot x_1^{(j)} = 1,$$

$$w_2 = \sum_{j=1}^4 a^{(j)} \cdot t^{(j)} \cdot x_2^{(j)} = 1.$$

Для визначення значення коефіцієнта w_0 необхідно встановити, які з векторів є опорними $\mathbf{x}^{(o)}$, та застосувати формулу

$$w_0 = \frac{1}{n_o} \sum_{j \in S} \left(t^{(j)} - \sum_{l \in S} a^{(l)} \cdot t^{(l)} \cdot \sum_{i=1}^m f_i(\mathbf{x}^{(l)}) \cdot f_i(\mathbf{x}^{(j)}) \right), \quad (2.7.21)$$

де S – множина опорних векторів, а n_o – їх кількість.

Склад множини опорних векторів визначають за відповідними ненульовими множниками $a^{(j)}$, оскільки при формуванні функції Лагранжа (2.7.8) на $a^{(j)}$ накладаються обмеження, аналогічні до додаткових умов (2.7.19) для змінних $v^{(j)}$:

$$\begin{cases} a^{(j)} \geq 0, \\ t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)}) - 1 \geq 0, \\ a^{(j)} \left\{ t^{(j)} \cdot \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)}) - 1 \right\} = 0. \end{cases} \quad (2.7.22)$$

Остання умова з (2.7.22) свідчить про те, що для кожного вектора-реалізації $\mathbf{x}^{(j)}$ виконується або $a^{(j)} = 0$, або $t^{(j)} \sum_{i=0}^K w_i f_i(\mathbf{x}^{(j)}) - 1 = 0$. Тоді за (2.7.5) $a^{(j)} > 0$ тільки для опорних векторів $\mathbf{x}^{(o)}$.

Таким чином, для задачі (табл. 2.7.1) при $\mathbf{a} = (0; 0.5; 0.5; 1)$

$$S = \{\mathbf{x}^{(2)}, \mathbf{x}^{(3)}, \mathbf{x}^{(4)}\}, w_0 = -1 \text{ за (2.7.21), а } d(\mathbf{x}) = x_1 + x_2 - 1.$$

До цього часу ми припускали, що класи X_1^o та X_2^o є лінійно-роздільними в просторі вторинних ознак розпізнавання $\{f_i(\mathbf{x})\}$. При цьому за методом опорних векторів формуються безпомилкові за навчальною матрицею лінійні вирішальні правила. На практиці класи можуть перекриватися, і в цьому випадку застосовується нелінійний метод опорних векторів. Основна ідея такої модифікації полягає у відображенні векторів-реалізацій навчальної матриці в такий простір вторинних ознак розпізнавання, який дозволить перетворити класи на лінійно-роздільні. При цьому відповідні функції $f_i(\mathbf{x})$ та ядро $\mathbf{K}(\mathbf{x}, \mathbf{x}')$ будуть нелінійними. Для ілюстрації такого підходу розглянемо характеристику складності (ємності) алгоритмів для розв'язання задач класифікації для двох класів, що має назву розмірність Вапніка–Червоненкіса [14].

Розмірність Вапніка–Червоненкіса для функцій $\{g_i(\mathbf{x})\}$ дорівнює максимальній кількості векторів-реалізацій, які можуть бути розділені на два класи всіма можливими способами. Наприклад, будь-які три вектори-реалізації $\{(x_1, x_2)\}$, якщо вони не колінеарні, можуть бути розділені за допомогою лінійної функції $\{g(x_1, x_2)\}$ на площині $x_1 \times x_2$ (рис. 2.7.2).

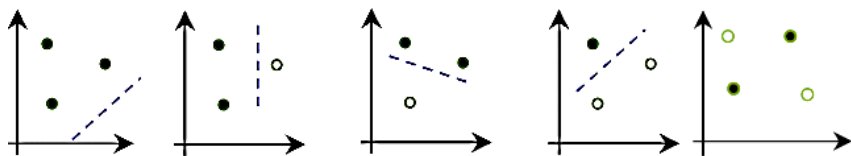


Рисунок 2.7.2 – Варіанти лінійних роздільних функцій

При цьому для чотирьох векторів-реалізацій на площині $x_1 0 x_2$ це правильно не завжди. Таким чином, розмірність Вапніка–Червоненкіса для лінійних функцій у двовимірному просторі дорівнює 3. Для загального випадку було доведено теорему про те, що розмірність Вапніка–Червоненкіса для лінійних функцій у n -вимірному просторі дорівнює $n + 1$. Таким чином, для скінченної множини векторів-реалізацій існує простір, де вони розділяються на два класи всіма можливими способами за допомогою лінійних функцій. Відтворення лінійного вирішального правила з використанням методу опорних векторів можливе після відображення векторів-реалізацій у відповідний простір вторинних ознак. На жаль, використання простору більшої розмірності призводить до істотного збільшення обчислювальних витрат та зменшення оперативності розрахунків. Тому на практиці намагаються зменшити розмірність простору вторинних ознак шляхом використання одного із стандартних ядер $K(x, x')$:

- $K(x, x') = (x \cdot x' + 1)^p$ – поліноміальне ядро;
- $K(x, x') = e^{-\frac{\sum_{i=1}^K (x_i - x'_i)^2}{2\sigma^2}}$ – радіальна базисна функція Гауса;
- $K(x, x') = \tanh(k \cdot x \cdot x' - \delta)$ – сигмоїдальне ядро ($k > 0$; $\delta < 0$).

Приклад.

Нехай $f_i(x) = x_i$ для $i = 1, (K = N = 2)$ і використовується ядро $K(x, x') = (x \cdot x')^2 = (x_1)^2 \cdot (x'_1)^2 + 2 \cdot x_1 \cdot x'_1 \cdot x_2 \cdot x'_2 + (x_2)^2 \cdot (x'_2)^2$.

Для відображення векторів-реалізацій з простору R^2 у простір R^3 сформуємо таку множину функцій:

$$\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x})),$$

що

$$(\mathbf{x} \cdot \mathbf{x}')^2 = \mathbf{f}(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}').$$

Наприклад,

$$\begin{aligned} f_1(\mathbf{x}) &= x_1^2, \\ f_2(\mathbf{x}) &= \sqrt{2} \cdot x_1 \cdot x_2, \\ f_3(\mathbf{x}) &= x_2^2, \end{aligned}$$

або

$$\begin{aligned} f_1(\mathbf{x}) &= \frac{1}{\sqrt{2}}(x_1^2 - x_2^2), \\ f_2(\mathbf{x}) &= 2 \cdot x_1 \cdot x_2, \\ f_3(\mathbf{x}) &= \frac{1}{\sqrt{2}}(x_1^2 + x_2^2). \end{aligned}$$

При збільшенні розмірності простору до R^4 можна запропонувати такі функції:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1^2, \\ f_2(\mathbf{x}) &= x_1 \cdot x_2, \\ f_3(\mathbf{x}) &= x_1 \cdot x_2, \\ f_4(\mathbf{x}) &= x_2^2. \end{aligned}$$

Критерієм, за яким визначаються функції ядра, є умова Мерсера. За цією умовою $K(x, x')$ є ядром, якщо виконується:

а) $K(x, x') = K(x', x)$;

б) $\iint K(x, x')g(x)g(x')dxdx' > 0$;

для будь-якої $g(x)$ такої, що $\int g(x)^2 dx$ – скінченне.

При цьому б) можна замінити на еквівалентну

$$\forall(x, x')[(x, x')^T \cdot K(x, x') \cdot (x, x') > 0].$$

Усі наведені раніше стандартні ядра сформовані з урахуванням умови Мерсера.

Як приклад розглянемо задачу формування вирішального правила $d(x) = \sum_{i=0}^K w_i f_i(x)$ за методом опорних векторів для задачі подання функції логічного додавання за модулем два XOR. Проведемо такі перетворення таблиці істинності функції XOR (табл. 2.7.7):

Таблиця 2.7.7 – Формування вхідних даних

	x_1	x_2	$x_1 XOR x_2$
$x^{(1)}$	0	0	0
$x^{(2)}$	0	1	1
$x^{(3)}$	1	0	1
$x^{(4)}$	1	1	0

 \Rightarrow

x_1	x_2	t
-1	-1	-1
-1	1	1
1	-1	1
1	1	-1

Аналіз табл. 2.7.7 показує, що кількість ознак розпізнавання дорівнює $N = 2$, кількість векторів-реалізацій навчальної матриці для класу X_1^0 дорівнює $n_1 = 2$, для класу

$X_2^0 - n_2 = 2$. Графічне зображення функції XOR подано на рис. 2.7.3.

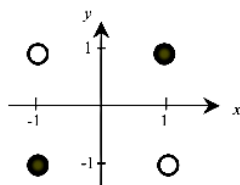


Рисунок 2.7.3 – Графічне зображення функції XOR

Аналіз рис. 2.7.3 показує, що на відміну від функції AND функція XOR не може бути подана у вигляді лінійного вирішального правила, оскільки множини векторів-реалізацій нульових та одиничних наборів цієї функції перетинаються.

Будемо використовувати поліноміальне ядро

$$\begin{aligned} K(\mathbf{x}, \mathbf{x}') &= (\mathbf{x} \cdot \mathbf{x}' + 1)^2 = (x_1 \cdot x'_1 + x_2 \cdot x'_2 + 1)^2 = \\ &= (x_1 \cdot x'_1)^2 + (x_2 \cdot x'_2)^2 + 2 \cdot x_1 \cdot x'_1 \cdot x_2 \cdot x'_2 + 2 \cdot x_1 \cdot x'_1 + \\ &\quad + 2 \cdot x_2 \cdot x'_2 + 1 \end{aligned}$$

і відобразимо двовимірні вектори-реалізації у шестивимірний простір ($K = 6$) вторинних ознак розпізнавання (табл. 2.7.8)

$$\mathbf{f}(\mathbf{x}) = (x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1),$$

що задовольняє умову $(\mathbf{x} \cdot \mathbf{x}')^2 = \mathbf{f}(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}')$. Крім того, таке ядро відносять до стандартних і воно задовольняє також умову Мерсера.

Таблиця 2.7.8 – Результат відображення вхідних даних у шестивимірний простір вторинних ознак розпізнавання

x_1	x_2	t	\Rightarrow	x_1^2	x_2^2	$\sqrt{2}x_1x_2$	$\sqrt{2}x_1$	$\sqrt{2}x_2$	1	t
-1	-1	-1		1	1	$\sqrt{2}$	$-\sqrt{2}$	$-\sqrt{2}$	1	-1
-1	1	1		1	1	$-\sqrt{2}$	$-\sqrt{2}$	$\sqrt{2}$	1	1
1	-1	1		1	1	$-\sqrt{2}$	$\sqrt{2}$	$-\sqrt{2}$	1	1
1	1	-1		1	1	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{2}$	1	-1

Аналіз табл. 2.7.8 показує, що після відображення вхідні вектори-реалізацій стали лінійно-роздільними по осі $\sqrt{2}x_1x_2$. Таким чином, використання вектора коефіцієнтів $\mathbf{w} = (0; 0; \sqrt{2}; 0; 0; 0)$ дозволяє визначити оптимальну роздільну поверхню:

$$d(\mathbf{x}) = \sum_{i=1}^K w_i f_i(\mathbf{x}^{(j)}) = x_1^{(j)} x_2^{(j)} = 0.$$

Крім того, всі вектори-реалізації будуть опорними, оскільки вони задовольняють умову (2.7.5). Зрозуміло, що при застосуванні методу опорних векторів у визначенні вторинних ознак розпізнавання $\mathbf{f}(\mathbf{x})$ та формуванні табл. 2.7.8 немає необхідності, достатньо визначитися зі структурою ядра $\mathbf{K}(\mathbf{x}, \mathbf{x}')$ у (2.7.11).

2.8 Детерміновані методи автоматичної класифікації

Автоматична класифікація (кластер-аналіз даних або самонавчання системи розпізнавання) дозволяє здійснювати

розбиття простору ознак на класи розпізнавання (кластери) без застосування апріорно класифікованих навчальних матриць, які формуються при розпізнаванні образів «з учителем» [32-34].

В основу автоматичної класифікації покладено гіпотезу компактності, згідно з якою у просторі ознак розпізнавання існує множина згущень реалізацій образів із відповідними центрами їх розсіювання. Реалізації, що належать до одного згущення, є найближчими до еталонного вектора-реалізації, який визначає центр розсіювання реалізацій відповідного класу розпізнавання. При цьому поняття «близькість» розглядається у загальному випадку як міра різноманітності реалізації образів, яка є функціоналом як від дистанційних критеріїв, так і критеріїв зв'язності між реалізаціями.

Основними етапами автоматичної класифікації є:

1) формування у просторі ознак розпізнавання вибіркової некласифікованої множини реалізацій класів розпізнавання;

2) визначення параметрів функціонування системи розпізнавання, що впливають на її функціональну ефективність і що будуть оптимізуватися у процесі кластеризації;

3) вибір міри подібності об'єктів як критерію оптимізації;

4) реалізація методу автоматичної класифікації для створення груп подібних об'єктів (класів);

5) визначення моменту зупину алгоритму автоматичної класифікації даних;

6) подання результатів автоматичної класифікації і перевірка їх стійкості.

7) аналіз одержаних результатів на їх валідність, залежно від якого приймається рішення про необхідність корекції обраної метрики та методу автоматичної класифікації.

Методи автоматичної класифікації даних залежно від структури вирішальних правил поділяються на дві групи:

- лінійні (неієрархічні, або однорівневі);
- ієрархічні;

а залежно від ступеня перетину класів розпізнавання на:

- чіткі (класи розпізнавання не перетинаються);
- нечіткі (класи розпізнавання перетинаються).

Лінійна автоматична класифікація передбачає розбиття вхідних даних на визначену кількість класів розпізнавання. Існує два основних підходи до реалізації лінійних методів. Перший підхід полягає у визначеності меж класів розпізнавання як найбільш щільних ділянок у багатовимірному просторі ознак розпізнавання, тобто топологічно клас визначається там, де має місце найбільше згущення реалізацій. Другий підхід полягає у мінімізації міри відмінності реалізацій об'єктів. Оскільки задача кластеризації полягає у побудові оптимального розбиття простору ознак на класи розпізнавання, за критерій оптимізації можна використати середньоквадратичну похибку розбиття

$$e^2 = \sum_{m=1}^M \sum_{j=1}^{n_m} \|x_m - x_m^{(j)}\|^2 \rightarrow \min_{\{m\}}, \quad (2.9.1)$$

де x_m – центр ваги класу X_m^o , який визначається як вершина усередненого вектора-реалізацій цього класу;

$x_m^{(j)}$ – j -та реалізація класу X_m^o ; n_m – кількість реалізацій класу X_m^o .

У рамках другого підходу найбільш поширеним серед лінійних методів кластеризації є алгоритм K -середніх. Цей алгоритм буде задану кількість класів, розміщених як можна далі один від одного. Схема алгоритму складається з таких етапів:

1) випадково вибираються k реалізацій, вершини яких є початковими центрами ваги класів розпізнавання;

2) збільшується обсяг області класу розпізнавання на заданий крок зміни обраної дистанційної міри близькості;

3) перевизначаються центри ваги класів розпізнавання з урахуванням реалізацій образів, що потрапили до них ;

4) якщо критерій (2.9.1) не перевищує значення критерію зупину, то виконується пункт 2, інакше – пункт 5;

5) якщо число реалізацій образів, що належать до побудованого розбиття, не менше заданого $n_{зад} \leq n$, де n – повна кількість реалізацій у розподілі, то виконується пункт 6, інакше – пункт 1 і починається побудова нового розбиття простору ознак на кластери;

б) ЗУПИН.

Таким чином, у розглянутій схемі алгоритму K -середніх як критерій зупину роботи алгоритму вибирається мінімальне значення середньоквадратичної помилки (2.11.1) і задане число реалізацій образів, що належать до побудованого розбиття.

Перевагою методу K -середніх є простота алгоритму автоматичної класифікації, що здійснюється в рамках геометричного підходу, а основним недоліком можна

вважати необхідність апіорного задання кількості класів розбиття та відсутність гарантії побудови чіткого розбиття.

За лінійними алгоритмами будується одне розбиття простору ознак на класи розпізнавання, а за ієрархічними – система вкладених розбиттів. У результаті реалізації ієрархічного алгоритму кластеризації створюється дерево класів розпізнавання, коренем якого є вся вибірка, а листками – фінальні класи розбиття. Ієрархічні алгоритми використовуються при розбитті вхідних даних на велику кількість кластерів. При цьому вони дозволяють на кожному ієрархічному рівні розбивати кластери на страти, для яких вирішальні правила будуються за лінійними алгоритмами.

Алгоритми ієрархічної автоматичної класифікації поділяються на два основні типи: агломеративні (висхідні) і дивізімні (низхідні).

В агломеративних методах обмежена множина класів поступово об'єднується в більші множини, тому такі методи в літературі класифікуються як висхідні. Основою цих методів є обчислення критерію якості розподілу множини S на підмножини S_1, \dots, S_k , де k – кількість підмножин, на які поділяється множина S . Для цього випадку за критерій якості розподілу беруть міру близькості множин. Найбільш вживаними мірами близькості двох підмножин є:

- відстань за принципом ближнього сусіда

$$\rho_{\min}(S_1, S_2) = \min_{x_m^{(j)} \in S_1, x_c^{(j)} \in S_2} d(x_m^{(j)}, x_c^{(j)}),$$

де $x_m^{(j)}$ – вектор-реалізація класу $X_m^o \subseteq S_1$; $x_c^{(j)}$ – вектор-реалізація класу $X_c^o \subseteq S_2$;

- відстань за принципом дальнього сусіда

$$\rho_{\max}(S_1, S_2) = \max_{x_m^{(j)} \in S_1, x_c^{(j)} \in S_2} d(x_m^{(j)}, x_c^{(j)});$$

- відстань за центрами ваги двох множин

$$\rho(S_1, S_2) = d(x_m, x_c),$$

де x_m – еталонний (усереднений) вектор-реалізація, вершина якого визначає центр ваги класу $X_m^o \subseteq S_1$; x_c – еталонний вектор-реалізація, вершина якого визначає центр ваги класу $X_c^o \subseteq S_2$.

У випадку агломеративних алгоритмів, на вхід яких подається розподіл підмножин множини S , принцип побудови загальної ієрархічної структури має вигляд

$$(S_1^*, S_2^*) = \arg \min_{\substack{S_1 \neq S_2 \\ S_1, S_2 \in S^{(k)}}} \rho(S_1, S_2),$$

де $S^{(k)}$ – підмножина, одержана на k -му кроці агломерації підмножин.

Однією із переваг використання агломеративних методів побудови ієрархічних структур є простота візуалізації результатів класифікації.

На рисунку 2.9.1 наведено приклад візуалізації графа ієрархії.

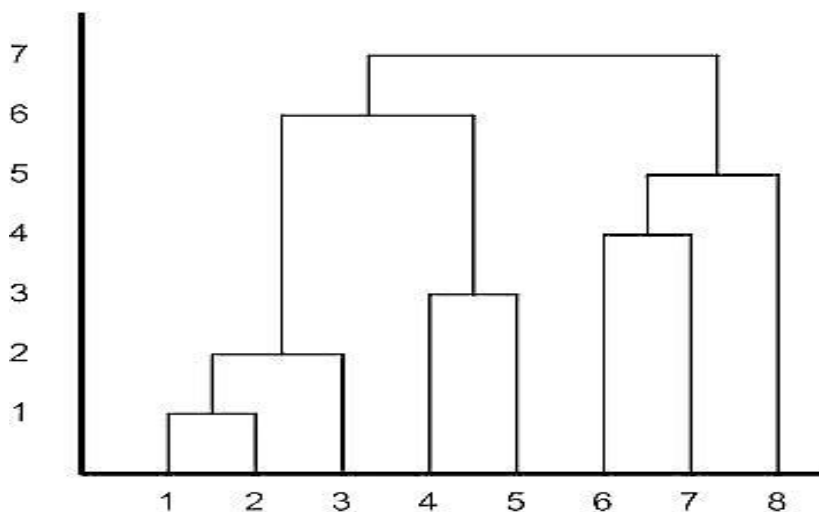


Рисунок 2.9.1 – Візуалізація графа ієрархії

Аналіз рис. 2.9.1 показує, що на кожному кроці підмножини меншої потужності об'єднуються в одну підмножину, і при цьому у загальному випадку вершини об'єднання двох підмножин можуть знаходитися на різних рівнях. Такий випадок при візуалізації графа ієрархії називається інверсією.

У дивізімних методах множина класів розпізнавання поступово розділяється на менші підмножини, що утворюють у результаті відповідні алфавіти класів розпізнавання меншої потужності. Методи цієї групи ще називають спадними.

У випадку дивізімних методів розбиття ієрархічної структури алгоритм побудови підмножин має вигляд

$$(S_1^*, S_2^*) = \arg \max_{S_1 \cup S_2 = S} \rho(S_1, S_2),$$

де S_1^* – оптимальна в розумінні критерію якості розбиття, що застосовується, підмножина ієрархічної структури S ; S_2^* – оптимальна в розумінні критерію якості розбиття підмножина ієрархічної структури S ($S_1^* \neq S_2^*$).

Чіткі алгоритми автоматичної класифікації ставлять кожній реалізації простору ознак розпізнавання у відповідність номер класу, тобто кожна реалізація належить лише до одного класу. У нечітких алгоритмах кожна реалізація належить до кожного класу з деякою ймовірністю.

Розглянемо методи кластер-аналізу в геометричній інтерпретації. Класичним представником таких методів є метод FOREL [33]. Ідея цього методу полягає в побудові в просторі ознак гіперсфери радіусом $R_1 = R_{\max}$, яка охоплює реалізації класу розпізнавання X_1^o .

За множиною точок вершин реалізацій, що належать до гіперсфери, визначається середнє значення координат, в яке пересувається центр гіперсфери. Оскільки в поточну гіперсферу попали додатково інші точки, то процедура їх усереднення повторюється, і відповідно центр гіперсфери не пересувається у нову усереднену координату і так доти, поки не відбудеться зупин процедури. Тоді всі точки, що належать до цієї гіперсфери не розглядаються і процес повторюється доти, поки не будуть використані всі точки.

Таким чином, результатом таксономії є множина гіперсфер (формальних елементів) радіусом R_1 із центрами визначеними за вищенаведеним алгоритмом. Такий процес називається циклом з формальним елементом радіусом R_1 ($R_1 = const$). У наступному циклі аналогічно будуються

гіперсфери радіусом $R_2 = R_1 - \Delta R$, де ΔR – параметр функціонування системи розпізнавання, який впливає як на оперативність алгоритму, так і на точність розбиття простору ознак на таксони.

Метод FOREL не дозволяє побудувати чітке розбиття еквівалентності класів розпізнавання. Основний недолік алгоритму FOREL полягає в тому, що результати таксономії залежать від початкового вибору центра гіперсфери радіусом R_1 . Але цей алгоритм може бути корисним при побудові апіорного нечіткого розбиття простору ознак на класи.

Методом кластер-аналізу, що базується на критеріях зв'язності, є метод KRAV [33]. В основу методу покладено обчислення інтегрованого критерію якості таксономії:

$$F = \ln \frac{ah}{\lambda\rho} \rightarrow \max, \quad (2.9.2)$$

де a – середня довжина ребер, що з'єднують таксони:

$$a = \frac{1}{Q-1} \sum_{q=1}^{Q-1} a_q, \quad (2.9.3)$$

У формулі (2.9.3) a_q – довжина ребер одного таксона:

$$a_q = \frac{1}{m-1} \sum_{l=1}^{m-1} a_l,$$

де a_l – довжина l -го ребра q -го таксона; m – кількість точок у таксоні.

Параметри h і λ у виразі (2.9.1) визначаються відповідно за такими формулами:

$$h = \prod_{m=1}^Q \frac{m_q}{M},$$

де m_q – кількість точок q -го таксона; M – загальна кількість усіх точок;

$$\lambda = \frac{1}{M-1} \sum_{i=1}^{M-1} \lambda_i.$$

Якщо довжина l -го ребра a_l , а довжина найближчого суміжного з ним ребра β_{\min} , то чим меншою буде λ_i , тим більше є причин вважати, що саме по ребру з довжиною a_l пройде межа між таксонами, тобто

$$\lambda_i = \frac{\beta_{\min}}{a_l},$$

де β_{\min} – довжина найближчого суміжного ребра.

У виразі (2.9.2) параметр p – це усереднена за всіма таксонами міра близькості точок, що визначається за формулою

$$\rho = \frac{1}{Q} \sum_{q=1}^Q \rho_q,$$

де

$$\rho_q = \frac{1}{m_q} \sum_{l=1}^{m_q-1} a_l -$$

усереднена міра близькості в середині q -го таксона.

Таким чином, стверджується, що чим більше значення функціонала F , тим буде вищою якість розбиття.

Практична цінність таксономічних методів автоматичної класифікації полягає в можливості побудови нечіткого класифікованого розбиття простору ознак на класи розпізнавання. Але ці методи не дозволяють будувати безпомилкові за навчальною матрицею вирішальні правила для класів розпізнавання, що перетинаються в просторі ознак.

Контрольні запитання та завдання до другого розділу

- 1 Що називається компактністю реалізацій образу?
- 2 Що називається унімодальною компактністю реалізацій образу?
- 3 Що називається полімодальною компактністю реалізацій образу?
- 4 Що називається чіткою компактністю реалізацій образу?
- 5 Що називається нечіткою компактністю реалізацій образу?
- 6 Нехай у просторі ознак розпізнавання R^2 було сформовано еталони чотирьох класів розпізнавання:

$$\mathbf{x}_1 = (1, 1), \mathbf{x}_2 = (-1, -1), \mathbf{x}_3 = (-1, 1), \mathbf{x}_4 = (0, 0).$$

Відтворіть область рішень кожного з класів за методом еталонів графічно.

7 Нехай у просторі ознак розпізнавання R^2 було сформовано чотири еталони двох класів розпізнавання:

$$\mathbf{x}_{1_1} = (1, 1), \mathbf{x}_{1_2} = (0, 0), \mathbf{x}_{2_1} = (-1, 1), \mathbf{x}_{2_2} = (-1, -1).$$

Відтворіть область рішень кожного з класів за методом еталонів графічно.

8 Нехай навчальна матриця класу X_1^0 складається з шести реалізацій:

$$(0, 3), (0, 2), (0, 1), (0, 0), (-1, 0), (-2, 0),$$

а класу X_2^0 – із чотирьох реалізацій:

$$(1, 3), (1, 1), (1, 0), (0, -1).$$

Сформуйте вирішальне правило за методом k -найближчих сусідів та визначте належність реалізації $(1, 4)$:

- а) при $k = 1$;
- б) при $k = 3$;
- в) при $k = 5$.

9 Для вирішальних правил, сформованих у завданні 8 при $k = 1$, знайдіть реалізації, для яких неможливо прийняти рішення щодо їх належності.

10 Які переваги і недоліки має метод еталонів?

10 Які переваги і недоліки має метод еталонів, що дробляться?

11 Розробіть модифікацію методу еталонів, що дробляться, для випадку, коли класи розпізнавання розбиті на підкласи.

12 Які переваги і недоліки мають лінійні вирішальні правила?

13 Побудуйте лінійні вирішальні правила у вигляді одношарового перцептрона для випадку, коли алфавіт класів розпізнавання складається з трьох класів.

14 Яка основна ідея методу найближчого сусіда?

15 Яка основна ідея методу K-найближчих сусідів?

16 Які переваги і недоліки має метод K-найближчих сусідів?

17 Яка основна ідея методу потенціальних функцій?

18 Сформуйте вирішальне правило за даними завдання 8 для потенціальної функції

$$K(\mathbf{x}, \mathbf{x}^{(j)}) = \frac{1}{(\mathbf{x} - \mathbf{x}')(\mathbf{x} - \mathbf{x}')^T}$$

19 Які переваги і недоліки має метод потенціальних функцій?

20 Які переваги і недоліки мають структурні методи розпізнавання образів?

21 Наведіть приклади практичних задач, де доцільно використовувати структурні методи розпізнавання образів.

22 Побудуйте лінійні вирішальні правила за методом опорних векторів для задачі подання функції логічного додавання OR.

22 Побудуйте лінійні вирішальні правила за методом опорних векторів для задачі подання логічної функції імплікація.

23 Побудуйте вирішальні правила за методом опорних векторів для задачі подання функції логічна рівнозначність.

24 Як поділяти методи автоматичної класифікації за структурою вирішальних правил?

25 Як поділяти методи автоматичної класифікації за ступенем перетину класів розпізнавання?

26 Поясніть основну ідею методу К-середніх.

27 Які переваги і недоліки має метод К-середніх?

28 Яка особливість агломеративних алгоритмів автоматичної класифікації?

29 Яка особливість дивізимних алгоритмів автоматичної класифікації?

30 Яка схема алгоритму ФОРЕЛЬ?

31 Яка ідея покладена в основу алгоритму КРАБ?

32 Які переваги і недоліки мають методи таксономічні методи автоматичної класифікації?

Список літератури

1. Эшби У. Р. Конструкция мозга. Происхождение адаптивного поведения: пер. с англ.; под. ред. П. К. Анохина и В. А. Шидловского / У. Р. Эшби. – М. : ИИЛ, 1962. – 308 с.

2. Анохин П. К. Биология и нейрофизиология условного рефлекса / П. К. Анохин. – М.: Медицина, 1968. – 547 с.

3. Иванов А. И. Нейросетевые алгоритмы биометрической идентификации личности : монография / А. И. Иванов. – М. : Радиотехника, 2004. – 144 с.

4. Shalkoff R. J. Digital image processing and computer vision / R. J. Shalkoff. – New York-Chichester-Brisbane-Toronto-Singapore: John Wiley & Sons, 1989. – 489 p.

5. Duda R. O. Pattern Classification, second ed. / R. O. Duda, P. E. Hart, D. G. Stork. – John Wiley & Sons, New York, 2001. – 738 p.

6. Путятин Е. П. Обработка изображений в робототехнике / Е. П. Путятин, С. И. Аверин. – М. : Машиностроение, 1990. – 320 с.

7. Олдендерфер М. С. Кластерный анализ. Факторный, дискриминантный и кластерный анализ: пер. с англ.; под. ред. И. С. Енюкова / М. С. Олдендерфер, Р. К. Блэшфилд. – М. : Финансы и статистика, 1989. – 215 с.

8. Загоруйко Н. Г. Алгоритмы обнаружения эмпирических закономерностей / Н. Г. Загоруйко, В. Н. Елкина, Г. С. Лбов. – Новосибирск : Наука. – 1985. – 110 с.

9. Зайченко Ю. П. Основы проектування інтелектуальних систем: навчальний посібник / Ю. П. Зайченко. – К. :

Видавничий Дім «Слово», 2004. – 352 с.

10. Ивахненко А. Г. О принципах построения обучающихся систем управления сложными процессами / А. Г. Ивахненко. – М. : Наука, 1970. – 252 с.

11. Ту Дж. Принципы распознавания образов: пер. с англ. / Дж. Ту, Р. Гонсалес. – М. : Мир, 1978. – 416 с.

12. Hastie T. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. – 2nd ed. / T. Hastie, R. Tibshirani, J. Friedman. – Springer-Verlag, 2009. – 746 p.

13. Краснопоясовський А. С. Класифікаційний аналіз даних : навчальний посібник / А. С. Краснопоясовський. – Суми : Видавництво СумДУ, 2002. – 159 с.

14. Vapnik V. N. The nature of statistical learning theory / V. N. Vapnik. – Springer, 1999. – P. 314.

15. Краснопоясовський А. С. Інформаційний синтез інтелектуальних систем керування: підхід, що ґрунтується на методі функціонально–статистичних випробувань / А. С. Краснопоясовський. – Суми : Видавництво СумДУ, 2004. – 261 с.

16. Айвазян С. А. Прикладная статистика: классификация и снижение размерности : справ. изд./ С. А. Айвазян, В. М. Бухштабер, И. С. Енюков, Л. Д. Мешалкин. – М. : Финансы и статистика, 1989. – 607 с.

17. Беллман Р. Вопросы анализа и процедуры принятия решений / Р. Беллман, Л. Заде. – М. : Мир, 1976. – С. 172-216.

18. Васильев В. И. Распознающие системы: справочник 2-е изд., перераб. и доп. / В. И. Васильев. – Киев : Наукова думка, 1983. – 422 с.

19. Довбиш А. С. Основи проектування інтелектуальних систем : навчальний посібник / А. С. Довбиш. – Суми : Видавництво СумДУ, 2009. – 171 с.

20. Кузьмин И. В. Основы теории информации и кодирования / И. В. Кузьмин, В. А. Кедрус. – Киев : Вища школа, 1978. – 256 с.

21. Кузьмин И. В. Оценка эффективности и оптимизация автоматизированных систем контроля и управления / И. В. Кузьмин. – М. : Сов. радио, 1971. – 296 с.

22. Кульбак С. Теория информации и статистика: пер. с англ. / С. Кульбак. – М. : Наука, 1967. – 408 с.

23. Волошин Г. Я. Методы распознавания образов / Г. Я. Волошин. – Владивосток : ВГУЭС, 2000. – 74 с.

24. Местецкий Л. М. Математические методы распознавания образов: курс лекций / Л. М. Местецкий. – М. : МГУ, 2004. – 85 с.

25. Симаков В. С. Адаптивное управление сложными системами на основе теории распознавания образов / В. С. Симаков, Е. В. Луценко. – Краснодар : КГТУ, 1999. – 318 с.

26. Фу К. Структурные методы в распознавании образов : пер. с англ. / К. Фу. – М. : Мир, 1977. – 320 с.

27. Новикова Н. М. Структурное распознавание образов : учебное пособие для вузов / Н. М. Новикова. – Воронеж : Издательство ВГУ, 2008. – 30 с.

28. Burges C. J. C. A tutorial on support vector machines for pattern recognition / C. J. C. Burges // Knowledge Discovery and Data Mining. – 1998. – № 2(2). – P. 121–167.

29. Cristianini N. Support vector machines and other kernel-based learning methods / N. Cristianini, J. Shawe-Taylor. – Cambridge University Press, 2000. – P. 204.

30. Muller K. R. An introduction to kernel based learning algorithms / K. R. Muller, S. Mika, G. Ratsch, et al. // IEEE

Transactions on Neural Networks. – 2001.– № 12(2). – P. 181–202.

31 Scholkopf B. New support vector algorithms / B. Scholkopf, A. Smola, R. C. Williamson, P. L. Bartlett // Neural Computation. – 2000. – 12(5). – P. 1207–1245.

32. Мандель И. Д. Кластерный анализ / И. Д. Мандель. – М. : Финансы и статистика. – 1988. – 176 с.

33. Загоруйко Н. Г. Прикладные методы анализа данных и знаний / Н. Г. Загоруйко. – Новосибирск : ИМ СО РАН, 1999. – 270 с.

34. Han J. Data mining: concepts and techniques. – 3rd ed. / J. Han, M. Kamber, J. Pei. – Morgan Kaufmann / Elsevier, 2012. – 744 p.

Навчальне видання

Довбиш Анатолій Степанович,
Шелехов Ігор Володимирович

ОСНОВИ ТЕОРІЇ РОЗПІЗНАВАННЯ ОБРАЗІВ

Навчальний посібник

У двох частинах

Частина 1

Художнє оформлення обкладинки І. В. Шелехова
Редактори: С. М. Симоненко, Н. З. Ключко
Комп'ютерне верстання І. В. Шелехова

Формат 60x84/16. Ум. друк. арк. 6,51. Обл.-вид. арк. 5,24. Тираж 300 пр. Зам. №

Видавець і виготовлювач
Сумський державний університет,
вул. Римського-Корсакова, 2, м. Суми, 40007
Свідоцтво суб'єкта видавничої справи ДК № 3062 від 17.12.2007.