УДК 004.8

Т.І. Завалій, Ю.В. Нікольський
Національний університет "Львівська політехніка",
кафедра інформаційних систем та мереж

АНАЛІЗ ДАНИХ ТА ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ТЕОРІЇ НАБЛИЖЕНИХ МНОЖИН

Описано теорію наближених множин та методику використання цього підходу для пошуку правил у таблицях даних. Ці правила утворюють класифікатор, який може класифікувати нові приклади. Описано використання ROC-кривої та коефіцієнта успішності для оцінювання якості таких класифікаторів.

The paper describes the theory of rough sets and application of this approach for mining rules from data tables. These rules serve as a classifier that can perform classification of new examples. The use of success rate and ROC curve for evaluation of classifier's quality is described.

1. Вступ

Задачі, пов'язані з інтелектуальним опрацюванням даних і видобуванням знань, є надзвичайно актуальними у зв'язку з постійним збільшенням обсягів інформації, доступної для людини. Застосуванням спеціальних методів *видобування знань з баз даних* (KDD, knowledge discovery in databases), зокрема, *інтелектуального аналізу даних* (DM, data mining) можна розкрити неявну структуру даних, виявити взаємні впливи різних факторів та закономірності, присутні у масиві даних. Для цього успішно використовують різноманітні методи м'яких обчислень: нейронні мережі й генетичні алгоритми, розмиті та наближені множини [1, 2]. Із застосуванням цих методів отримують добрі результати розв'язання реальних задач, пов'язаних із обробкою великих обсягів зашумлених даних у разі моделювання складних предметних областей.

У цій статті розглянуто проблему дослідження таблиці з даними про діагностування деякої хвороби серця. Ці дані необхідно попередньо опрацювати, проаналізувати і вивести правила, оцінити якість цих правил. Оцінка якості отриманих правил та прийнятих рішень розглядається як актуальна проблема видобування знань. Основним результатом дослідження є побудовані класифікатори та оцінка їхньої якості. Для обробки даних та побудови класифікаторів використано методи наближених множин.

2. Аналіз останніх досліджень

2.1. Теорія наближених множин

Теорія *наближених множин* (rough sets) за останні два десятиліття набула значного поширення та застосування, зокрема у видобуванні знань з баз даних. Розвиток теорії наближених множин спричинив появу нейронаближеного числення [3], наближених розмитих систем [4], окремих розділів гранульного числення [5] тощо. Основи теорії уперше сформульовані Ж. Павлаком (Z. Pawlak) [6]. Вона стала гнучким математичним інструментом подолання суперечливостей у даних та виявлення в них прихованих закономірностей. У межах технології наближених множин застосовують метод *логічного виведення* (boolean reasoning) для редукування даних і побудови правил прийняття рішень. Кінцевим результатом є або набір логічних правил вигляду „якщо ..., то", які здебільшого отримуються генеруванням матриць нерозрізненності та редуктів, або шаблони даних, які можна застосувати для фільтрування та редукування даних.

Дані, що досліджують з використанням наближених множин, подають за допомогою прикладів, зібраних у таблицю прийняття рішень $\mathbf{A}=(U,A\cup\{d\})$, де U – непорожня скінченна множина прикладів, A – непорожня скінченна множина умовних атрибутів, d – атрибут прийняття

рішення з доменом V_d, $|V_d| = k$. Значення v_i атрибута d відносить кожний приклад $x \in U$ до класу прийняття рішення X_i, де $i = 1,2,\ldots,k$. У загальному випадку, таблиця може мати більше одного класифікуючого атрибуту. У табл 1. наведено зразок таблиці прийняття рішень, яку можна було б сформувати, наприклад, у галузі кредитування.

Таблиця 1

Таблиця прийняття рішень
з класифікуючим атрибутом "прогноз"

a	*b*	*c*	*d*	*прогноз*
задовільно	ні	так	10	негативний
добре	так	ні	10	позитивний
задовільно	ні	так	40	позитивний
задовільно	ні	так	10	негативний
добре	ні	так	10	негативний

На множині U прикладів таблиці визначають відношення нерозрізненності. Нехай $B \subseteq A$ – підмножина всієї множини атрибутів таблиці **A**, x та y – приклади таблиці. Тоді $IND(B) = \{(x,y) \in U \times U, \ \forall a \in B \mid a(x)=a(y)\}$ – відношення B-нерозрізненності [7]. Якщо явно не задають множину B, то мають на увазі всю множину A атрибутів таблиці. Отже, приклади x та y нерозрізненні, якщо однаковими є відповідні значення їхніх атрибутів. Відношення нерозрізненності симетричне, рефлексивне й транзитивне, а, отже, є відношенням еквівалентності. Клас еквівалентності, отриманий на основі цього відношення, позначають через $[X]_B$, де $X \subseteq U$. Приклади цього класу еквівалентні на множині атрибутів B.

Частина даних в таблиці може бути надмірною або суперечливою, зокрема, якщо певні приклади еквівалентні на множині умовних атрибутів, але мають різні значення атрибута прийняття рішення. Ці приклади неможливо однозначно класифікувати. Кажуть, що вони належать *граничній області*. Якщо гранична область непорожня, то множину U називають *наближеною*; у протилежному випадку вона *точна*. Суперечливі приклади, що належать граничній області, вилучають з таблиці в процесі наближення. Якщо $X \subset U$ – підмножина множини прикладів, то X можна наблизити на множині B атрибутів побудовою так званих B-нижнього та B-верхнього наближень множини X. Їх позначають $\underline{B}X$ та $\overline{B}X$, відповідно, де $\underline{B}X = \{x \mid [x]_B \subset X\}$, $\overline{B}X = \{x \mid [x]_B \cap X \neq 0\}$, $x \in X$. З множини B приклади з $\underline{B}X$ можна впевнено класифікувати як елементи множини X, натомість, приклади з $\overline{B}X$ можна класифікувати лише як можливі елементи X. Множину $BN_B(X) = \overline{B}X - \underline{B}X$ називають B-граничною областю множини X. Вона містить приклади, які неможливо однозначно віднести до множини X на підставі інформації з атрибутів B. Множину $U - \overline{B}X$ називають B-зовнішньою областю X: вона містить приклади, які можна однозначно класифікувати як такі, що не належать множині X. Ілюстрацією введених понять є рис. 1, на якому показано простір прикладів з таблиці прийняття рішень, фрагмент якої неведений у табл. 1. В кожному з квадратів на рисунку приклади нерозрізненні на множині атрибутів $B=\{a,d\}$. На осі ординат відкладено значення атрибута a, на осі абсцис – атрибута d. Приклади з множини X належать класу прийняття рішень з міткою "негативний".

Окрім видалення з таблиці прикладів, що належать граничній області, вилучають надмірні стовпці, значення в яких не впливають на класифікацію прикладів; для побудови правил залишають лише ті атрибути, від яких залежить розрізненність прикладів таблиці. Множину цих атрибутів називають *редуктом*. Іншими словами, редукт – це підмножина $B \subset A$ атрибутів таблиці, яка забезпечує таку саму розрізненність всіх прикладів таблиці, як і вся множина A. Мінімальний набір атрибутів, значення яких дають змогу відрізнити один приклад від усіх інших прикладів таблиці,

називають *об'єктно-залежним редуктом*. З погляду кінцевого результату, ефективною є побудова так званих *динамічних редуктів*, які обчислюють на основі окремих частин таблиці. Для цього довільно формують набір з k підтаблиць, і серед знайдених для них редуктів відбирають той, який зустрічається найчастіше. Цей підхід більшою мірою враховує особливості розподілу даних в таблиці.

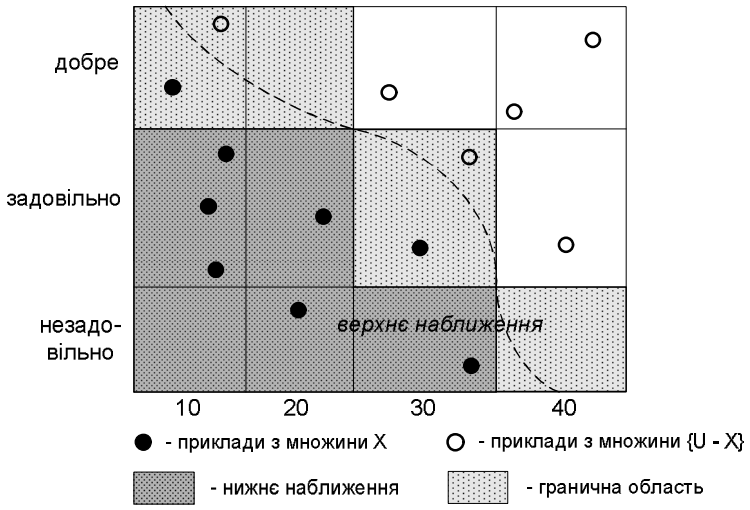

Рис. 1. Множина X, її верхнє та нижнє наближення на множині атрибутів B={a,d}

Редукт таблиці шукають за методом логічного виведення. Він полягає у побудові *функції розрізнення* (discernibility function) і її подальшому спрощенні. Функція розрізнення є булевою функцією $g_A(a_1^*,...,a_m^*) = \wedge\{\vee c_{ij}^* \mid 1 \le j \le i \le n, c_{ij}^* \ne 0\}$, де $i, j = (1,2,\mathbf{K},n)$, m – кількість атрибутів таблиці **A**, n – кількість прикладів, a_m^* – атрибут таблиці, c_{ij}^* – елемент спеціальної *матриці розрізнення* $M(A)$ [7]. Елемент c_{ij}^* є диз'юнкцією атрибутів, за значеннями яких відрізняються приклади x_i та x_j з U. Ці приклади повинні мати різні значення атрибута прийняття рішення. Якщо приклади мають однакове значення атрибута прийняття рішення, то $c_{ij}^* = 0$. Тобто $M(A)$ – це симетрична матриця розмірів $n \times n$ з нульовою діагоналлю. Наступним кроком є зведення функції розрізнення до вигляду досконалої кон'юнктивної нормальної форми, атоми якої є атрибутами редукту. Якщо це не можливо, функцію спрощують до кон'юнктивної нормальної форми мінімальної довжини і переводять у диз'юнктивну нормальну форму, диз'юнкти якої утворюють редукти.

Наприклад, для таблиці прийняття рішень, поданої у табл. 1, функція розрізнення $g_A(a,b,c,d) = (a \vee b \vee c) \wedge (d) \wedge (a \vee b \vee c) \wedge (b \vee c) \wedge (d) \wedge (a \vee d)$. Наступне спрощення $g_A = (a \vee b \vee c) \wedge (b \vee c) \wedge (d) \wedge (a \vee d) = (b \vee c) \wedge (d) \wedge (a \vee d) = (b \vee c) \wedge (d) = (b \wedge d) \vee (c \wedge d)$ утворює два рівноцінні редукти $R_1 = \{b,d\}$ та $R_2 = \{c,d\}$.

Іншим методом спрощення функції розрізнення є жадібний алгоритм Джонсона [8]. Нехай R – множина атрибутів, які утворюють шуканий редукт, S – набір підмножин s_i атрибутів a_m за значеннями яких відрізняються два приклади x та y з U, $a_m \in A$, $s_i \subseteq A$. Ці підмножини атрибутів відповідають кон'юнктам функції розрізнення, яка будується при логічному виведенні. Тобто S є інакшим способом представлення функції g_A. Так, для прикладу з табл. 1 $S = \{\{a,b,c\},\{d\},\{a,b,c\},\{b,c\},\{d\},\{a,d\}\}$, $(a,b,c,d) \in A$. Для кожної підмножини s_i

обчислюють коефіцієнт $w(s_i)$, який позначає вагу s_i в S. У [9] пропонується приймати вагу підмножини рівною кількості її повторень у S. Весь алгоритм пошуку редукту складається з п'яти кроків.

Крок 1. Покласти $R = \varnothing$.

Крок 2. Вибрати в S атрибут a, який максимізує значення $\sum w(s_i)$ всіх s_i, які містять a.

Крок 3. Додати a до R.

Крок 4. Видалити з S всі підмножини s_i, що містять a.

Крок 5. Якщо $S = \varnothing$, то R – шуканий редукт, кінець. Інакше – перехід до кроку 2.

Якщо на кроці 5 припинити побудову редукту тоді, коли з набору S вилучено не всі елементи, то ми отримаємо так званий *наближений редукт*. Цей редукт характеризується *ступенем підтримки* (HF, hitting fraction) – відсотком тих підмножин s_j від загальної кількості s_i, атрибути з яких увійшли до редукту R ($s_j \cap R \neq \varnothing$). Тобто, при побудові наближеного редукту враховуються не всі підмножини s_i, а лише заданий їх відсоток з більшою вагою w. У реальних наборах даних навіть один приклад може вносити у дані шум і змінювати функцію нерозрізненності для таблиці, що, своєю чергою, відобразиться у редукті. Використання ступеня підтримки дає змогу фільтрувати шум у даних й уникати *перенавчання* (overfitting), оскільки наближений редукт містить лише найважливіші атрибути та відображає найсильніші залежності в таблиці прийняття рішень.

На основі атрибутів, що увійшли до редукта генерують правила прийняття рішень вигляду $\alpha \to \beta$, та розраховують їхні числові характеристики. Тут α – умова правила, β – наслідок. Таке правило відображає залежність, можливо, ймовірнісного характеру, між набором значень v атрибутів a та значенням v_d атрибута прийняття рішення d. Елементарною частиною правила є *дескриптор* – вираз вигляду $a = v$, де $a \in A$, $v \in V_a$. Умову правила утворює кон'юнкція дескрипторів $a = v$, а наслідком правила є дескриптор $d = v_d$. Наприклад, (*a*="добрий") $\cup$ (*b*="так") $\cup$ (*c*="ні") $\to$ прогноз="позитивний". Інший спосіб запису – ***ЯКЩО*** (*a*="добрий") ***і*** (*b*="так") ***і*** (*c*="ні") ***ТО*** (прогноз="позитивний"). Правила, які генерують на основі об'єктно-залежних редуктів мають різну довжину умови і називаються мінімальними правилами.

Якість правила оцінюють такими числовими характеристиками [10]:

1. *Підтримка*. Цей параметр позначається *support*(α→β) і вказує кількість прикладів з навчальної таблиці, для яких виконується як умова α правила, так і його наслідок β.

2. *Точність*. Точність правила – це відношення кількості навчальних прикладів, для яких виконується все правило, до кількості навчальних прикладів, для яких виконується лише умова правила:

$$accuracy(\alpha \to \beta) = \frac{support(\alpha \to \beta)}{support(\alpha)}.$$

3. *Покриття*. Цей параметр показує відношення кількості навчальних прикладів, для яких виконується все правило, до кількості навчальних прикладів, для яких виконується наслідок правила:

$$coverage(\alpha \to \beta) = \frac{support(\alpha \to \beta)}{support(\beta)}.$$

4. Допоміжні характеристики, наприклад *coverage*(α) – частина навчальних прикладів, для яких виконується умова правила:

$$coverage(\alpha) = \frac{support(\alpha)}{|U|}.$$

Введені параметри дають не лише певні відомості про характер отриманих правил, а й безпосередньо використовуються при застосуванні цих правил для прийняття рішень.

2.2. Прийняття рішень за допомогою правил

Найпоширенішим випадком прийняття рішення є класифікація. Класифікатором вважатимемо четвірку $C = \langle RUL, P, HF, t \rangle$, де *RUL* – множина правил, *P* – характеристики правил, τ – *порогове значення*, яке є дійсним числом, $\tau \in [0, 1]$. Під час класифікації нового прикладу *x*, необхідно, використовуючи правила і їхні характеристики, розрахувати числову оцінку належності цього приклада кожному з можливих класів. Клас із найбільшим значенням коефіцієнта впевненості "перемагає" в класифікації прикладу. В загальному випадку, розраховують параметр *certainty*(*x*,β) – коефіцієнт впевненості в належності прикладу x класу β. Потім, залежно від знайдених коефіцієнтів, приймають рішення щодо класифікації прикладу. Це так званий процес "голосування". У разі наявності лише двох класів приклад відносять до класу, коефіцієнт впевненості для якого перевищує задане порогове значення τ, і вся процедура класифікації зводиться до розрахунку бінарної класифікаційної функції $d_k(x)$, яку описано нижче.

За методом "голосування" класифікацію виконують так [10]:

1) у множині всіх правил *RUL* шукають правила, умова яких виконується для прикладу *x*. Відібрані таким чином правила утворюють множину $RUL(x) \subseteq RUL$;

2) якщо множина $RUL(x) = \varnothing$, то класифікація неможлива, а як приклад приймають визначене наперед рішення. Як варіант, можна відбирати правила, що наближено відповідають прикладу;

3) етап "виборів". Усуваються конфлікти і рангуються рішення;

а) **к**ожному правилу $r \in RUL(x)$ приписують певну кількість голосів – $\text{votes}(r)$. Зазвичай $\text{votes}(r)$ дорівнює підтримці правила $\text{support}(r)$, але можливе обчислення $\text{votes}(r)$ і на основі деякого інтегрованого показника якості правила;

б) обчислюють коефіцієнт нормалізації $\text{norm}(x)$ як суму голосів, зібраних всіма правилами, відібраними для даного прикладу:

$$\text{norm}(x) = \sum_{i=1}^{|RUL(x)|} \text{votes}(r_i); \quad (1)$$

в) для кожного можливого класу β обчислюють коефіцієнт впевненості $\text{certainty}(x,\beta)$:

$$R_b = \{ r \in RUL(x) \mid b \text{ – наслідок } r \}; \quad (2)$$

$$\text{votes}(\beta) = \sum_{r \in R_\beta} \text{votes}(r); \quad (3)$$

$$\text{certainty}(x,\beta) = \text{votes}(\beta) \,/\, \text{norm}(x) \ . \quad (4)$$

За цією процедурою можливі значні розширення й уточнення, зокрема щодо трактування невідомих значень, якщо вони є, щодо трактування випадків, коли одне правило є узагальненням іншого тощо. Якщо домен атрибута прийняття рішення є двоелементним $V_d = \{X_1, X_2\}$, то процедура прийняття рішення зводиться до обчислення функції $f(x) = certainty(x,X_1)$ належності прикладу *x* до одного з двох класів, зазвичай – X_1. Область визначення $\phi(x)$ – [0, 1], де *certainty*(x,X_1) розраховують за формулами 1–4 та використовують функцію θ, яка інтерпретує значення функції $\phi(x)$ так:

$$\theta(\phi(x)) = \begin{cases} 1, & \textit{якщо} \quad \phi(x) \ge \tau \\ 0, & \textit{якщо} \quad \phi(x) < \tau \end{cases}$$

Отже θ – звичайна гранична функція, для якої потрібно задати порогове значення $t \in [0, 1]$.

Бінарну класифікаційну функцію $d_k(x)$ можна визначити так [10]:

$$d_k : U \xrightarrow{\phi} [0,1] \xrightarrow{\theta} \{0,1\},$$

де U – множина тестових прикладів, які потрібно класифікувати. Треба зазначити, що використання цього типу класифікаційних функцій характерне не лише для класифікаторів на основі правил, а й для інших технологій машинного навчання – нейронних мереж, дерев рішень, рівнянь регресії тощо.

Результати класифікації тестових прикладів подають *матрицею помилок* (МП), елементами якої є кількісні характеристики застосування побудованих правил для класифікації прикладів:

1. $True(X_i)$ – кількість тестових прикладів, правильно класифікованих до класу X_i,
2. $False(X_i)$ – кількість тестових прикладів, помилково класифікованих до класу X_i.

На основі цих показників розраховують *коефіцієнт успішності* (*КУ*) класифікації:

$$КУ = \frac{\sum_{i=1}^{k} True\ (X_i)}{\sum_{i=1}^{k} True\ (X_i) + \sum_{i=1}^{k} False\ (X_i)},$$

де $k = |V_d|$ – кількість класів у таблиці прийняття рішень.

Вигляд матриці помилок у разі бінарної класифікації показано у табл. 2. У табл. 3 показано вигляд матриці помилок у разі, коли домен атрибута прийняття рішень $V_d = \{X_1, X_2, X_3\}$. Матриця помилок якісного класифікатора повинна містити якомога більші значення на головній діагоналі і якомога менші (в ідеальному випадку – нульові) значення в решті комірок.

Таблиця 2

Вигляд матриці помилок при бінарній класифікації

Справжній клас	**Прогнозований клас**	
	X_1	X_2
X_1	True(X_1)	False(X_2)
X_2	False(X_1)	True(X_2)

Таблиця 3

Вигляд матриці помилок при небінарній класифікації

Справжній клас	**Прогнозований клас**		
	X_1	X_2	X_3
X_1	True(X_1)	False(X_2)	False(X_3)
X_2	False(X_1)	True(X_2)	False(X_3)
X_3	False(X_1)	False(X_2)	True(X_3)

Оцінку якості побудованого класифікатора також виконують за допомогою ROC-кривої (ROC, receiver operating characteristic) [11]. Ця крива показує результати роботи класифікатора для різних порогових значень τ. Площа *AUC* (area under curve) під ROC-кривою характеризує якість розрізнення класифікатором класів X_1 та X_2, $AUC \in [0, 1]$. За наявності більше ніж двох класів один з них вважають класом X_1, а решту об'єднують у клас X_2. Якість розрізнення класів називають дискримінаційною здатністю; вона за змістом є здатністю класифікатора правильно визначати заданий клас. Нульовій дискримінаційній здатності класифікатора відповідає значення площі *AUC*

≤ 0.5. Нульова дискримінаційна здатність означає, що класифікатор прогнозує клас прикладу не краще ніж звичайне підкидання монети. Ідеальна дискримінаційна здатність – це функція, графік якої складається з двох відрізків: перший з'єднує точки з координатами (0;0) та (0;1), а другий – (0;1) та (1;1). Площа під таким графіком дорівнює 1. Отже, що більша площа *AUC*, то краща якість класифікатора. Значення площі під ROC-кривою можна вважати статистичною оцінкою правильності прогнозу належності тестового прикладу певному класу.

Для побудови ROC-кривої для кожного значення t обчислюють матрицю помилок розмірів 2×2 і розраховують частину прикладів правильно (*TPR*, true positive rate) та неправильно (*FPR*, false positive rate) віднесених класифікатором до класу X_1 за формулами

$$TPR(t) = \frac{True(X_1)}{True(X_1) + False(X_2)},\ FPR(t) = \frac{False(X_1)}{False(X_1) + True(X_2)}.$$

Частину правильно класифікованих прикладів відкладають на осі ординат, а неправильно – на осі абсцис. На рис. 2 зображено приклад ROC-кривої деякого класифікатора. Результати класифікації отримані в системі Rosetta для таблиці прийняття рішень з результатами діагностування певного захворювання (табл. 5) у 3532 пацієнтів. Для навчання використано 3355 прикладів, а решту 177 прикладів використано для тестування класифікатора і побудови ROC-кривої. В системі Rosetta значення t змінюється від 0 до 1 з кроком 0.004. При цьому, для кожного t обчислювалась матриця помилок.

Таблиця 4

Результати класифікації тестових прикладів класифікатором з 47 правил

Порогове значення, t	**Частина неправильних класифікацій,** ***FPR***(t)	**Частина правильних класифікацій,** ***TPR***(t)	**Коефіцієнт успішності,** ***КУ***
0.892	0.000	0.000	0.657
0.868	0.000	0.118	0.697
0.512	0.000	0.294	0.758
0.480	0.077	0.500	0.778
0.388	0.092	0.500	0.768
0.288	0.169	0.794	0.818
0.072	0.215	0.912	0.828
0.064	0.415	0.941	0.707
0.012	0.462	1.000	0.697
0.008	0.569	1.000	0.626
0.004	0.662	1.000	0.566
0.000	1.000	1.000	0.343

Точки кривої відповідають результатам застосування класифікатора з різними пороговими значеннями t (див. табл. 4) для визначення класу X_1, що відповідає наявності хвороби. Так, в точці кривої (0;0), видно, що при t = 0.892 класифікатор не відніс до X_1 жодного прикладу x, тобто для даної множини тестових прикладів значення коефіцієнта certainty(x,X_1) ніколи не перевищувало 0.892. Із зменшенням t до 0.868 класифікатор "розпізнає" 0.118×100%=11.8% прикладів класу X_1. Зі зменшенням t до 0.512 класифікатор "розпізнає" 29.4% прикладів класу X_1. При t = 0.48 класифікатор відносить до X_1 вже 50% тестових прикладів з класу X_1 і 7.7% тестових прикладів з класу X_0. З табл. 4 видно, що для даного класифікатора оптимальним є значення t = 0.072 в точці кривої (0.215;0.912). При цьому пороговому значенні правильно класифіковано 91.2% прикладів класу X_1 і 78.5% класу X_2, а коефіцієнт успішності становив *КУ* = 0.828. Значення площі *AUC* = 0.9.

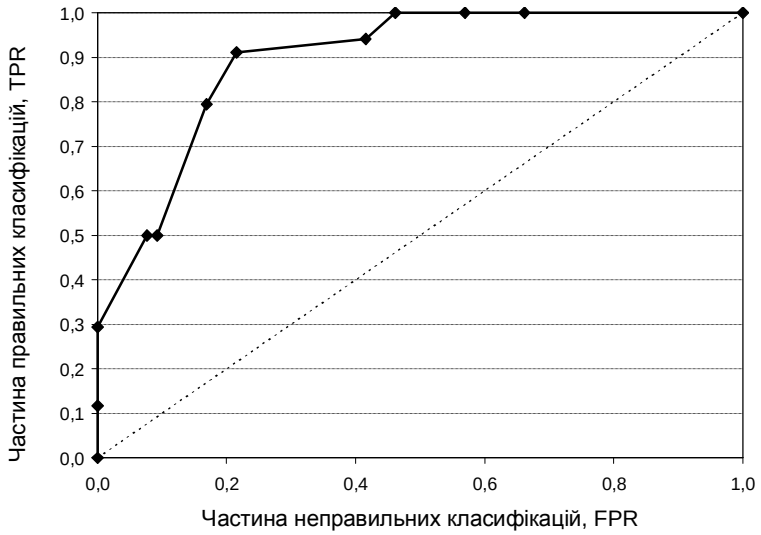

Рис. 2. Приклад ROC-кривої, побудованої на основі табл. 4

Початково, ROC-криві використовували в галузі обробки сигналів, зокрема для фільтрування шумів. У машинному навчанні використання ROC-кривих для оцінки класифікаторів теж дає багато можливостей. Значення площі під ROC-кривою може слугувати інтегральним показником якості класифікатора, не залежним від обраного порогового значення функції класифікації, і використовуватись при порівнянні класифікаторів. Використання ROC-кривих дає змогу налаштувати класифікатор залежно від ціни помилки при класифікації. Наприклад, якщо ціна діагностування хвороби чи додаткового обстеження здорової людини невелика, то можна підібрати таке значення t, яке б давало змогу виявляти всі випадки захворювання у дійсно хворих (розпізнавати всі приклади класу X_1) за рахунок певного відсотка помилок щодо здорових людей.

3. Цілі статті

Мета роботи полягала в описі технології наближених множин та основних кроків з її застосування для вирішення задач аналізу даних та машинного навчання. Потрібно було приділити увагу питанням класифікації та оцінки якості класифікаторів. Також, необхідно було проаналізувати масив медичних даних з результатами діагностування певної хвороби серця. Вивести з таблиці даних правила і застосувати їх для класифікації нових прикладів захворювання. З використанням ROC-кривої та коефіцінта успішності можна оцінити якість класифікатора.

У результаті проведених досліджень розв’язано такі основні задачі:

- описано методику навчання та класифікації з використанням наближених множин;
- описано методику оцінювання класифікатора за допомогою ROC-кривої;
- знайдено атрибути, які найбільше впливають на прийняття рішень;
- побудувано правила та оцінено якість класифікації.

4. Методика дослідження

Методика дослідження даних з використанням наближених множин складається з такої послідовності кроків [10]:

1. Доповнення або вилучення прикладів з невідомими значеннями.
2. Поділ таблиці на навчальну та тестову частини.
3. Дискретизація числових атрибутів навчальної таблиці.
4. Дискретизація числових атрибутів тестової таблиці.
5. Знаходження редуктів для навчальної таблиці.
6. Генерування правил на основі редуктів.
7. Тестування правил на прикладах тестової таблиці.

Алгоритм експериментів зображено орієнтованим графом, вершини якого позначені номерами кроків, а дуги – даними, які опрацьовують на наступному кроці (рис. 3). Кроки 1-4 утворюють етап попереднього оброблення даних в процесі видобування знань, кроки 5 та 6 – етап інтелектуального аналізу даних, а крок 7 – етап оцінки та інтерпретації результатів. Дугам графа відповідають: A – початкова таблиця прийняття рішень розмірів $m \times n$; A_1, A_2, A_3, A_4, A_5 – таблиці, отримані в результаті виконання кроків алгоритму; $CUTS$ – множина зрізів, отриманих у результаті дискретизації; RED – множина отриманих редуктів; RUL – множина правил; $МП$ – матриця помилок з результатами класифікації.

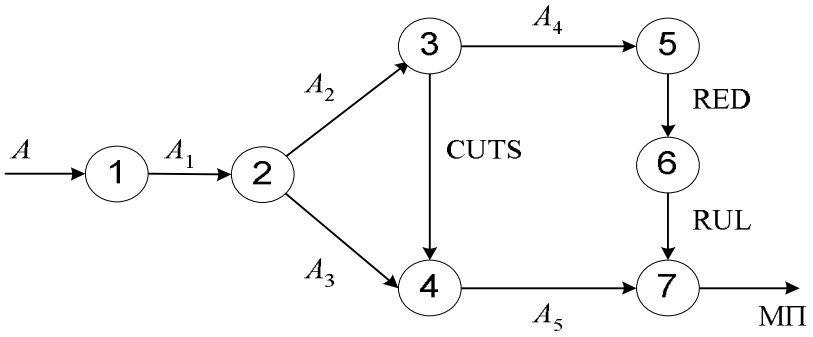

Рис. 3. Граф алгоритму начання та оцінки якості класифікації

5. Основний матеріал

Задача, яку розглянуто в практичній частині статті, полягає у побудові на основі медичних даних класифікатора у вигляді набору правил та його подальшій оцінці. Дані медичних спостережень та діагностування пацієнтів з хворобою серця подані у вигляді таблиці розмірів 3532×15 (табл. 5).

Таблиця 5

Фрагмент таблиці з даними про результати діагностування

№	Age	Gender	PIK	KV	SK	UA	AA	BE	OH	REW	R_AK	R_MK	R_AKMK	GH	KHKS
1	53	1	0	0	0	0	0	0	0	0	0	0	0	1	**1**
2	65	1	1	0	0	0	0	0	0	1	0	0	1	0	**0**
3	63	1	1	0	0	0	0	0	0	0	0	0	0	1	**1**
4	62	1	1	0	0	0	0	0	0	0	0	0	0	1	**1**
5	70	1	0	0	0	0	0	0	0	0	0	0	0	0	**1**
<...>															
3531	44	1	1	0	0	0	0	0	0	0	0	0	0	0	**0**
3532	62	1	1	0	0	0	0	0	0	0	0	0	0	1	**0**

Таблиця прийняття рішень містить атрибути Age (вік), Gender (стать: 0 – жін., 1 – чол.), інші атрибути, що представляють результати тестів, і класифікуючий атрибут KHKS (0 – захворювання немає, 1 – захворювання є). Атрибут Age містить значення з діапазону [15, 88], середнє значення віку – 58. Всі інші атрибути – двійкові. Пацієнтів за статтю розподілено так: чоловіків – 2201 (62.32%), жінок – 1331 (37.68%). Дані є повними, тобто таблиця не містить порожніх значень.

Дослідження виконано у системі Rosetta [8, 9]. Rosetta призначена для аналізу даних та побудови правил за технологією наближених множин. Відповідно до методики дослідження було проведено дві серії експериментів. У кожній з них для побудови класифікаторів використано 1766 прикладів як навчальну множину, решта 1766 – як тестову. Кожна серія проводилась з метою визначити, як змінюється якість класифікатора залежно від значення ступеня підтримки, заданого

під час дискретизації атрибута Age та побудови редукта за алгоритмом Джонсона. У першій серії експериментів в результаті дискретизації за алгоритмом *Boolean reasoning* було виділено 10 вікових груп пацієнтів. У другій серії експериментів в результаті наближеної дискретизації зі ступенем підтримки *HF*=0.9 алгоритмом виділено лише дві вікові групи – [15..44) та [44..88] років. Всі результати експериментів узагальнені у табл. 6.

Таблиця 6

Результати експериментів

Серія експери-ментів	Кількість вікових груп	Номер класи-фікатора	Ступінь підтримки редукта, *HF*	Кількість атрибутів у редукті	Кількість правил	Коефіцієнт успішності, *КУ*	Площа, *AUC*
1	10	1	1.0	11	114	0.836	0.904
		2	0.92	5	57	0.806	0.854
2	2	3	1.0	11	79	0.844	0.904
		4	0.91	5	27	0.813	0.853

Застосування наближеної дискретизації дало змогу істотно зменшити розмір домена атрибута Age, внаслідок чого зменшилась кількість знайдених правил – зі 114 до 79, та з 57 до 27, відповідно. При цьому незначно збільшилась успішність класифікаторів №3-4 порівняно з класифікаторами №1–2, але майже не змінились значення площі під ROC-кривою. Це означає, що наближена дискретизація не вплинула на якість класифікаторів. Бачимо, що побудова наближених редуктів зменшила успішність класифікаторів №2 та №4. Можна зробити висновок, що атрибути у таблиці прийняття рішень підібрані дуже вдало, штучне зменшення їх кількості не покращує результатів. Лише три атрибути не увійшли до редуктів і є надмірними: R_AK, R_MK, R_AKMK.

Частину знайдених правил та їхні характеристики наведено у табл. 7. У табл. 8 наведено матрицю помилок для класифікатора №3. Цей класифікатор правильно визначив клас 1490 прикладів і взагалі не класифікував 19.

Таблиця 7

Частина правил класифікатора №3

№	Правило	***support*(a ® b)**	***accuracy*(a ® b)**	***coverage*(a ® b)**
1	AGE([44, 88]) AND GENDER(1) AND PIK(0) AND KV(0) AND SK(0) AND UA(0) AND AA(0) AND BE(0) AND OH(0) AND REW(0) AND GH(0) => KHKS(0)	243	0.93	0.18
2	AGE([44, 88]) AND GENDER(1) AND PIK(1) AND KV(0) AND SK(0) AND UA(0) AND AA(0) AND BE(0) AND OH(0) AND REW(0) AND GH(1) => KHKS(1)	94	0.42	0.25
3	AGE([44, 88]) AND GENDER(1) AND PIK(0) AND KV(0) AND SK(0) AND UA(0) AND AA(0) AND BE(0) AND OH(0) AND REW(0) AND GH(0) => KHKS(1)	18	0.07	0.05
< . . . >				
79	AGE([44, 88]) AND GENDER(0) AND PIK(1) AND KV(0) AND SK(0) AND UA(0) AND AA(0) AND BE(0) AND OH(0) AND REW(1) AND GH(1) => KHKS(0)	1	0.001	0.001

Таблиця 8

Результати класифікації для класифікатора №3

Справжній клас	Прогнозований клас	
	0	1
0	1298	65
1	192	192

Дискримінаційну якість класифікаторів №3–4 порівняно за допомогою ROC-кривих на рис. 4. Бачимо, що крива класифікатора №3 домінує над кривою класифікатора №4 і має більше значення площі *AUC*=0.904. Це означає, що класифікатор №3 краще визначає клас X_1=1, тобто діагностує наявність хвороби.

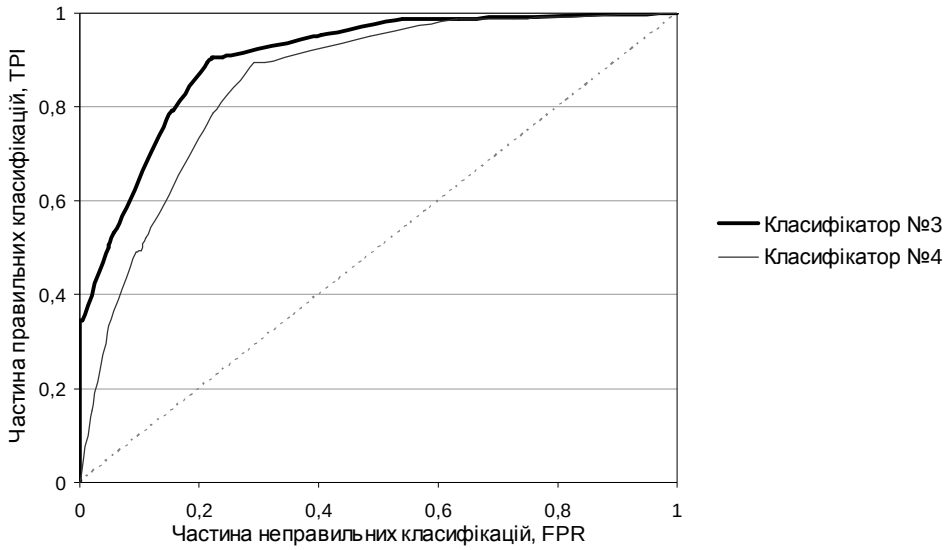

Рис. 4. ROC-криві, що ілюструють якість класифікаторів №3-4

6. Висновки

У статті описано теорію наближених множин та її застосування для побудови класифікаторів на основі правил. Розглянуто питання оцінки якості таких класифікаторів. Основними результатами дослідження є порівняння класифікаторів, побудованих для визначення діагнозу хвороби серця. При використанні наближеної дискретизації, зі ступенем наближення 0.9, було досягнуто успішної класифікації 84.4% тестових прикладів. Показано, що успішність класифікації зменшується на 3% при штучному зменшенні множини атрибутів редукта. Надалі необхідно покращити отримані результати побудовою динамічних редуктів та застосувати описану методику в інших предметних областях.

1. Заде Л. А. Роль мягких вычислений и нечеткой логики в понимании, конструировании и развитии информационных/интеллектуальных систем. // Новости Искусственного Интеллекта. РАИИ. – 2001. – №2–3. – С. 7–11. 2. Mitra S., Pal S. K., Mitra P. Data mining in soft computing framework: a survey // IEEE Transactions on Neural Networks. − 2002. − Vol. 13. – P. 3–14. 3. Pal S. K., Polkowski L., Skowron A., eds. Rough-Neuro Computing: Techniques for Computing with Words. − Springer-Verlag. − Heidelberg. − 2003. 4. Pal S. K., Skowron A., eds. Rough Fuzzy Hybridization: A New Trend in Decision Making. − Springer-Verlag. − 1998. 5. Bargiela A., Pedrycz W. Granular Computing. An Introduction. − Springer. − 2002. 6. Pawlak Z. Rough Sets. // International Journal of Computer and Information Sciences. − Plenum Press New York. − 1982. − Vol. 11/5. − P. 341−356. 7. Komorowski J., Polkowski L., Skowron A. Rough Sets: A Tutorial. // Eds. S. K. Pal and A. Skowron, Rough Fuzzy Hybridization: A New Trend in Decision-Making. − Springer-Verlag. − 1998. – P. 3–98. 8. Øhrn A. ROSETTA Technical Reference Manual. – 2001. (http://www.idi.ntnu.no/~aleks/). 9. Øhrn A., Komorowski J., Skowron A., Synak P. The design and implementation of a knowledge discovery toolkit based on rough sets: The ROSETTA system. In Polkowski L. and Skowron A., eds. Rough Sets in Knowledge Discovery 1: Methodology and Applications, volume 18 of Studies in Fuzziness and Soft Computing. − Physica-Verlag. − Heidelberg. − 1998. 10. Øhrn A. Discernibility and Rough Sets in Medicine: Tools and Applications, PhD thesis. Norwegian University of Science and Technology, Department of Computer and Information Science, − 1999. 11. Hastie T., Tibshirani R., Friedman J. The Elements of Statistic Learning; Data Mining, Inference, and Predicting. – Springer-Verlag. – 2001.