

**JERZY STEFANOWSKI**

**ALGORYTMY INDUKCJI  
REGUŁ DECYZYJNYCH  
W ODKRYWANIU WIEDZY**

Rozprawa habilitacyjna  
Wersja z 8 lutego 2001

Wydane przez Wydawnictwo Politechniki Poznańskiej, Seria  
Rozprawy nr 361

## Spis treści

<b>Streszczenie</b>	5
<b>1. Wstęp</b>	7
1.1. Problematyka odkrywania wiedzy z danych	7
1.2. Cel i zakres rozprawy	12
<b>2. Reguły decyzyjne, metody ich indukcji i stosowania w problemach klasyfikacji</b>	15
2.1. Pojęcia podstawowe związane z regułą reprezentacją wiedzy w uczeniu maszynowym	15
2.1.1. Reprezentacja danych	16
2.1.2. Reguły decyzyjne	17
2.1.3. Indukcja reguł decyzyjnych	18
2.2. Uwzględnianie niespójności informacji za pomocą teorii zbiorów przybliżonych oraz reguł decyzyjnych	21
2.3. Stosowanie reguł decyzyjnych do klasyfikowania nowych obiektów	28
2.4. Metody oceny klasyfikatorów regułowych	30
2.4.1. Miary oceny	30
2.4.2. Podstawowe techniki oceny eksperymentalnej	32
2.5. Dyskusja problemów związanych ze stosowaniem reguł decyzyjnych i teorii zbiorów przybliżonych w perspektywie klasyfikowania nowych obiektów	33
<b>3. Uwzględnianie różnych miar oceny reguł decyzyjnych w procesie ich indukcji</b>	37
3.1. Specyfika perspektywy opisu w odkrywaniu reguł decyzyjnych	37
3.2. Opisowe miary oceny reguł	39
3.3. Poszukiwanie zbioru reguł decyzyjnych z uwzględnieniem opisowych i klasyfikacyjnych miar oceny	42
3.3.1. Algorytm EXPLORE	43
3.3.2. Eksperymenty obliczeniowe	46
3.3.3. Automatyczne poszukiwanie ograniczonej liczby silnych reguł o akceptowalnych właściwościach klasyfikacyjnych	49
3.4. Interaktywne poszukiwanie reguł decyzyjnych	51
<b>4. Indukcja reguł decyzyjnych z danych numerycznych</b>	54
4.1. Wprowadzenie	54
4.2. Dyskretyzacja atrybutów liczbowych	55
4.3. Metody dyskretyzacji wstępnej	56
4.4. Bezpośrednia indukcja z danych numerycznych	57
4.5. Metoda globalnej wstępnej dyskretyzacji oparta na minimalizacji entropii warunkowej	58
4.6. Algorytm MODLEM	59
4.7. Eksperymenty obliczeniowe	63
4.8. Dyskusja otrzymanych wyników	65
<b>5. Zastosowanie teorii zbiorów przybliżonych oraz indukcji reguł decyzyjnych do danych zawierających nieznaną wartość atrybutów</b>	67

5.1. Brakujące i niedostępne wartości atrybutów	67
5.2. Uwzględnianie brakujących wartości oparte na relacji tolerancji	70
5.3. Uwzględnianie niedostępnych wartości atrybutów za pomocą relacji podobieństwa	72
5.4. Wartościowana relacja tolerancji	74
5.4.1. Motywacje	74
5.4.2. Definiowanie przybliżeń zbiorów	75
5.4.3. Przykład użycia wartościowanej relacji tolerancji	77
5.4.4. Reguły decyzyjne	80
5.4.5. Dyskusja użycia wartościowanej relacji tolerancji	82
<b>6. Reguły decyzyjne dla wielokryterialnego problemu sortowania</b>	<b>84</b>
6.1. Teoria zbiorów przybliżonych i reguły decyzyjne w wielokryterialnych problemach decyzyjnych	84
6.2. Teoria zbiorów przybliżonych oparta na relacji dominacji	86
6.3. Model ze zmienną spójnością w teorii zbiorów przybliżonych opartej na relacji dominacji	91
6.4. Indukcja reguł decyzyjnych w teorii zbiorów przybliżonych opartej na relacji dominacji	94
6.4.1. Algorytm DOMLEM	95
6.4.2. Algorytm DOMApriori	98
6.4.3. Inne aspekty indukcji reguł dla wielokryterialnego problemu sortowania	102
<b>7. Złożone systemy klasyfikujące</b>	<b>106</b>
7.1. Motywacje	106
7.2. Przegląd metod tworzenia złożonych systemów klasyfikujących	107
7.3. System hybrydowy integrujący uczenie się reguł decyzyjnych oraz wcześniej zapamiętanych przypadków	110
7.3.1. Architektura systemu	110
7.3.2. Proces uczenia się w systemie hybrydowym	111
7.3.3. Ocena eksperymentalna	112
7.4. Klasyfikator wielokrotny $n^2$	114
7.4.1. Architektura i zasada działania	114
7.4.2. Ocena stosowania różnych algorytmów uczących w budowie klasyfikatorów binarnych	117
<b>8. Przykłady zastosowania metod indukcji reguł decyzyjnych w analizie danych medycznych i technicznych</b>	<b>122</b>
8.1. Znaczenie zastosowań praktycznych	122
8.2. Przegląd wybranych zastosowań	123
8.2.1. Analiza danych medycznych	123
8.2.2. Problemy diagnostyki technicznej	127
8.2.3. Analiza danych finansowych	129
8.3. Uwagi na temat realizacji zastosowań praktycznych	130
<b>9. Podsumowanie</b>	<b>133</b>
<b>Literatura</b>	<b>137</b>
<b>Summary</b>	<b>151</b>

## Streszczenie

Rozprawa poświęcona jest odkrywaniu wiedzy z danych metodami indukcji reguł decyzyjnych w perspektywie predykcji i opisu. Dane zawarte są w tzw. tablicy decyzyjnej, której wiersze odpowiadają obiektom, a kolumny atrybutom; każdy element tej tablicy charakteryzuje jakiś obiekt za pomocą odpowiedniego atrybutu. Atrybuty dzielą się na warunkowe i decyzyjne – pierwsze charakteryzują obiekty niezależnie od decyzji określonych przez te drugie. Jeśli chodzi o decyzje, to rozpatruje się problematykę klasyfikacji, zatem wartości atrybutów decyzyjnych odpowiadają klasom decyzyjnym, a wiersze tablicy decyzyjnej (obiekty) – przykładom klasyfikacji (przykładom uczącym). Reguły decyzyjne indukowane są z przykładów uczących i określają, czy obiekt spełniający warunki zdefiniowane na podzbiórze atrybutów warunkowych należy do danej klasy decyzyjnej. Jeżeli dane (przykłady uczące), które podlegały procesowi odkrywania wiedzy, zawierają niespójności, to w procesie indukcji stosuje się elementy teorii zbiorów przybliżonych. W rozprawie zaprezentowano nowe metody oraz dokonano krytycznego przeglądu wykorzystania istniejących algorytmów indukcji reguł decyzyjnych. Przedstawiono oryginalną metodę generowania reguł decyzyjnych uwzględniającą miary oceny reguł związane z perspektywą opisu w odkrywaniu wiedzy – algorytm EXPLORE. Wprowadzono algorytm MODLEM indukcji reguł decyzyjnych z danych numerycznych bez ich wstępnej dyskretyzacji. Zaproponowano dwa nowe uogólnienia teorii zbiorów przybliżonych i metod indukcji reguł decyzyjnych dla niekompletnych tablic decyzyjnych. Pierwsze wykorzystuje niesymetryczną relację podobieństwa i umożliwia analizę przypadku niedostępnych wartości atrybutów. Drugie jest oparte na wartościowanej relacji tolerancji i przeznaczone dla przypadku brakujących wartości atrybutów. Wprowadzono też algorytmy DOMLEM oraz DOMApriori odkrywania reguł decyzyjnych dla wielokryterialnych problemów sortowania. Przedstawiono zasady stosowania algorytmów indukcji reguł do budowy złożonych systemów klasyfikujących. Dokonano także przeglądu zastosowań praktycznych omawianych metod w problemach medycznych oraz w diagnostyce technicznej.

# 1. Wstęp

## 1.1. Problematyka odkrywania wiedzy z danych

Rozwój systemów informatycznych, w szczególności baz danych czy technik elektronicznego rejestrowania informacji, ułatwił gromadzenie dużych zbiorów danych. Bankowość, rynek ubezpieczeniowy, firmy telekomunikacyjne, przedsiębiorstwa handlowe, systemy opieki medycznej to tylko niektóre przykłady miejsc, w których istnieją lub powstają duże bazy danych. Wyzwaniem staje się obecnie nie tylko efektywne przechowywanie takich danych, lecz także ich analiza, zdolność interpretacji i wyciągania użytecznych wniosków, które mogą prowadzić do lepszych decyzji. W związku z tym powstało zapotrzebowanie na nowe metody i narzędzia informatyczne, które mogą wspomóc człowieka w odkrywaniu wiedzy z danych [50]. Jest to obszarem zainteresowania nowej dziedziny nazywanej **odkrywaniem wiedzy** (ang. *knowledge discovery*). Termin ten zgodnie z definicją podaną w [52] oznacza „nietrywialny proces poszukiwania nowych, potencjalnie użytecznych i ostatecznie zrozumiałych wzorców z danych”.

Przez wzorzec (ang. *pattern*) rozumie się pewną regularność pomiędzy elementami danych. Termin ten jest używany w szerokim zakresie i obejmuje różne formy reprezentacji wiedzy; często stosowanymi są: tablice wielodzzielcze (kontyngencji), reguły asocjacyjne, drzewa i reguły decyzyjne, wyrażenia w logice pierwszego lub drugiego rzędu, modele regresji wielowymiarowej, zależności ilościowe wyrażone w postaci równań, skupienia obiektów czy Bayesowskie sieci przekonań [50, 124, 173, 271].

Z uwagi na związek odkrywania wiedzy z bazami danych często używa się określenia „odkrywanie wiedzy w bazach danych” lub skrótu KDD (z języka angielskiego *Knowledge Discovery in Databases*). Innym często spotykanym określeniem jest termin angielski *data mining* – **eksploracja danych** (w literaturze polskojęzycznej używa się także terminu „zglobianie danych”). Zgodnie z [50, 86] samo odkrywanie wiedzy jest jednak pojęciem szerszym i obejmuje eksplorację danych jako jeden z elementów składowych związany z użyciem konkretnego algorytmu w celu ekstrakcji wybranej reprezentacji wiedzy z danych. Według [50] odkrywanie wiedzy jest interaktywnym i iteracyjnym **procesem**, w którym wyróżnia się następujące etapy:

- analiza i poznanie dziedziny zastosowania, identyfikacja dostępnej wiedzy i celów użytkownika,
- wybór danych związanych z celami procesu,
- „czyszczenie” i wstępne przetwarzanie danych oraz ich redukcja,
- wybór zadania i algorytmów odkrywania wiedzy,
- pozyskiwanie wiedzy z danych (podstawowy krok eksploracji danych),

- interpretacja i ocena odkrytej wiedzy,
- przygotowanie wiedzy do użycia.

Obecnie większość badań koncentruje się na algorytmach eksploracji danych, chociaż inne etapy mają równie duże znaczenie dla pomyślnej realizacji całości procesu [18, 35, 86, 137].

Należy podkreślić dużą rolę użytkownika w tym procesie. Jak wskazuje się w [18, 50, 114, 137], to użytkownik systemu odkrywania wiedzy powinien posiadać dobre zrozumienie dziedziny zastosowania, tak aby wybrać właściwy podzbiór danych, określić, jakie są zadania analizy, jaka powinna być reprezentacja poszukiwanej wiedzy, których algorytmów należy użyć. Dlatego system odkrywania wiedzy powinien być oprogramowaniem interaktywnym, a nie narzędziem w pełni automatycznym. Sam proces odkrywania wiedzy jest procesem intensywnego współdziałania człowieka z oprogramowaniem i składa się najczęściej z wielu iteracji obejmujących modyfikacje wstępnych specyfikacji i powtarzanie niektórych kroków [18].

W odkrywaniu wiedzy z danych rozwiązuje się różne zadania. W pracach [50, 86, 262] wyróżnia się następujące podstawowe kategorie zadań: klasyfikację, regresję, analizę szeregów czasowych, analizę skupień, podsumowanie danych poprzez tworzenie opisów charakterystycznych lub wyróżniających klas obiektów, analizę zależności pomiędzy atrybutami opisującymi obiekty, wyszukiwanie anomalii i odchyłeń od wartości typowych w danych.

Podsumowując różne interpretacje odkrywania wiedzy, można stwierdzić, że zadania odkrywania wiedzy mogą być rozważane w dwóch perspektywach: predykcji oraz opisu.

**Predykcja** dotyczy przewidywania przyszłych lub nieznanymi wartości pewnych atrybutów na podstawie dostępnych danych. W szczególności, badania prowadzone w zakresie klasyfikacji obiektów (patrz przeglądy w [50, 262]), gdzie celem jest predykcja przydziału obiektów do pewnych kategorii (klas obiektów) na podstawie wiedzy wynikającej z analizy danych o przykładach klasyfikacji obiektów w przeszłości.

Perspektywa **opisu** w odkrywaniu wiedzy obejmuje zadania związane z wyszukiwaniem różnych wzorców charakteryzujących właściwości danych i z ich prezentacją użytkownikowi w zrozumiałej formie umożliwiającej interpretację [50, s. 12].

Odkrywanie wiedzy ma charakter interdyscyplinarny i wykorzystuje rezultaty pochodzące ze statystyki, uczenia maszynowego, wizualizacji danych, baz danych oraz przetwarzania informacji niepewnej (np. teorii zbiorów rozmytych czy teorii zbiorów przybliżonych). Uczenie maszynowe i statystyka mają obecnie największe znaczenie z uwagi na dostarczanie algorytmów dla odkrywania wiedzy [138]. Z drugiej strony, specyfika odkrywania wiedzy z baz danych dostarcza nowych problemów badawczych, zarówno dla uczenia maszynowego, jak i dla problematyki baz danych (patrz np. [35, 49, 86, 137, 144, 261, 262]).

Pomimo że odkrywanie wiedzy jest stosunkowo młodą dziedziną, istnieje już wiele ważnych zastosowań praktycznych (patrz przeglądy w [49, 86]). Powstają też systemy informatyczne wspomagające proces odkrywania wiedzy. Przykładami systemów komercyjnych są Intelligent Miner z IBM, wywodzący się z projektu badawczego DataQuest, MineSet z Silicon Graphics, Enterprise Miner z SAS Institute, Clementine z Integral Solution czy DBMiner. Przegląd różnych systemów zamieszczono w [86, rozdział 10] oraz [13, dodatek A].

Przedmiotem zainteresowania autora niniejszej rozprawy jest odkrywanie wiedzy z danych metodami indukcji reguł decyzyjnych. Jest to związane z prowadzonymi przez autora badaniami w zakresie uczenia maszynowego oraz teorii zbiorów przybliżonych.

Reguły decyzyjne indukowane są ze zbiorów danych reprezentujących informacje o zbiorze obiektów nazywanych przykładami uczącymi, opisanych za pomocą zbioru atrybutów. Problem rozważany w dziedzinie **uczenia maszynowego** (ang. *machine learning*) polega na znalezieniu reguł określających, czy obiekt należy do pewnego podzbioru nazywanego klasą (decyzyjną) lub pojęciem. Nieformalnie mówiąc, reguły decyzyjne indukowane z przykładów są wyrażeniami logicznymi reprezentowanymi w postaci:

*jeżeli* spełnione określone warunki *to* określona decyzja,

gdzie część warunkowa zawiera koniunkcję warunków elementarnych zbudowanych na wartościach atrybutów opisujących obiekty, a część decyzyjna określa przynależność obiektu spełniającego te warunki do danej klasy decyzyjnej.

Reprezentacja wiedzy w postaci reguł uznawana jest za czytelniejszą dla człowieka niż inne reprezentacje (porównaj dyskusje w [35, 141, 142, 185]). Podkreśla się także jej modularność i przydatność do analizy pojedynczych reguł przez eksperta. Ich forma prezentacji wykorzystywana jest do finalnej reprezentacji innych skutecznych klasyfikatorów, np. drzewa decyzyjnego czy sieci neuronowych. Reguły decyzyjne są wykorzystywane w wielu pomyślnych zastosowaniach uczenia maszynowego i odkrywania wiedzy [50, 86, 126].

Większość algorytmów poszukuje takich reguł drogą indukcyjnego uogólnienia opisu przykładów uczących – patrz dyskusje w [35, 124, 141, 142]. Reguły wygenerowane dla każdej klasy decyzyjnej powinny być spełniane przez przykłady należące do tej klasy (tzw. przykłady pozytywne). Równocześnie reguły nie powinny być spełniane przez żaden lub powinny być spełniane tylko przez niewiele przykładów z innych klas (tzw. przykładów negatywnych). Indukcję reguł w odkrywaniu wiedzy będziemy rozpatrywać w dwóch perspektywach:

- predykcji (klasyfikacji),
- opisu.

Celem indukcji ukierunkowanej na predykcję jest wygenerowanie z danych zbioru reguł, które będą użyte do klasyfikowania nowych obiektów. Celem indukcji ukierunkowanej na perspektywę opisu jest odkrycie z danych reguł reprezentu-

jących regularności (lub czasami anomalie), które charakteryzują dane i są „interesujące” dla różnego rodzaju użytkowników. Tego typu wzorce mogą ułatwić interpretację zależności pomiędzy wartościami atrybutów a definicją klasy decyzyjnej [137]. Zauważmy, że w tej perspektywie reguły mogą być użyte także do reprezentacji ogólniejszych zależności w danych, np. jako reguły asocjacyjne [1]. W rozprawie ograniczamy się jednak do omówionej poprzednio postaci reguł decyzyjnych wskazujących klasy decyzyjne.

Należy zauważyć, że zarówno rozwój algorytmów indukcji reguł, jak i sposób oceny reguł ukierunkowany był przede wszystkim na perspektywę klasyfikowania nowych obiektów. Ponieważ zbiór reguł traktowany jest wtedy jako klasyfikator, poprawność klasyfikowania jak największej liczby obiektów jest główną miarą oceny. Jeżeli proces odkrywania wiedzy ukierunkowany jest na perspektywę opisu danych, konieczne jest rozważanie także innych metod poszukiwania reguł i miar ich oceny. Nie było to dotychczas przedmiotem tak intensywnych badań jak w przypadku klasyfikacji, lecz rozwój dziedziny odkrywania wiedzy zwiększa zainteresowanie tą perspektywą [50, s. 12].

W przypadku budowania klasyfikatorów większość dotychczasowych badań prowadzono nad tworzeniem i eksperymentalną oceną pojedynczych klasyfikatorów. W ostatnich latach obserwuje się zainteresowanie tworzeniem klasyfikatorów złożonych integrujących więcej niż jeden klasyfikator składowy. Ogólnym celem takich systemów jest osiągnięcie lepszej trafności klasyfikowania nowych obiektów niż z zastosowaniem pojedynczych klasyfikatorów wieloklasowych. Badania eksperymentalne potwierdzają skuteczność proponowanych rozwiązań [10, 31, 39, 54, 89, 95, 146, 186, 262].

Innym ważnym problemem jest zdolność radzenia sobie z niedoskonałymi danymi. Jednym ze źródeł trudności w zadaniach opisu czy klasyfikacji jest istnienie niespójności w dostępnych danych. Obiekty posiadające identyczne (lub podobne) opisy, lecz zaliczone do różnych pojęć, uniemożliwiają stworzenie jednoznacznej definicji tychże pojęć. Niespójności nie powinny być traktowane wyłącznie jako wynik błędu czy szumu informacyjnego. Mogą one także wynikać z niedostępności części informacji, naturalnej granularności i niejednoznaczności języka reprezentacji. Przykładowo w analizie decyzyjnej takie niespójności mogą odzwierciedlać nieprecyzyjność stosowanego modelu, wahania decydenta czy niestabilny charakter jego preferencji. Dlatego niespójności tego typu nie powinny być pomijane czy amalgamowane z informacją pewną, lecz raczej przetwarzane w taki sposób, aby odróżnić wnioski pewne od możliwych [217, 224].

**Teoria zbiorów przybliżonych** (ang. rough sets) zaproponowana przez Pawlaka [164, 165] jest dogodnym narzędziem analizy tego typu niespójności informacji. W przypadku niemożliwości precyzyjnego zdefiniowania zbioru obiektów (pojęcia, klasy decyzyjnej) tworzy ona **dolne i górne przybliżenie** tego zbioru na podstawie klas **relacji nierozróżnialności** pomiędzy obiektami. Teoria zbiorów przybliżonych dostarcza spójnych podstaw metodologicznych dla wielu problemów odkrywania wiedzy związanych między innymi z analizą danych na różnych po-



ziomach reprezentacji, granularnością dostępnej informacji, oceną jakości dostępnych danych, tworzeniem przybliżonej klasyfikacji obiektów, badaniem zależności zachodzących między podzbiorem atrybutów opisujących obiekty, selekcją atrybutów czy tworzeniem nowych atrybutów oraz budową samych reguł decyzyjnych [110, 165, 169].

Uwzględnianie niespójności przykładów jest głównym elementem wykorzystania teorii zbiorów przybliżonych w indukcji reguł. Teoria ta jednoznacznie oddziela pewną i niespójną informację. W związku z tym zadanie indukcji reguł z niespójnych danych jest zamienione na indukcję z przybliżeń klas decyzyjnych. Reguły decyzyjne generowane dla każdego przybliżenia w sposób jednoznaczny odróżniają obiekty należące do danego przybliżenia (przykłady pozytywne) od obiektów nie należących do tego przybliżenia (przykłady negatywne). Sama konstrukcja reguł wykorzystuje zasadę indukcji z przykładów, co jest elementem wspólnym z algorytmami uczenia maszynowego. W konsekwencji użycia przybliżeń klas decyzyjnych można wyróżnić reguły pewne indukowane z dolnych przybliżeń oraz reguły możliwe indukowane z górnych przybliżeń. Proponuje się więc nowe algorytmy indukcji takich reguł. Niektóre z nich inspirowane są bezpośrednio technikami maszynowego uczenia się, np. [73, 82, 234], a inne wykorzystują oryginalne rozwiązania, np. oparte na funkcji rozróżnialności i wnioskowaniu bołowskiem [11, 12, 202, 203]. Aktualny stan teorii zbiorów przybliżonych przedstawiono między innymi w [110, 169, 175, 176, 179].

Pomimo intensywnego rozwoju tej teorii, nadal istnieje wiele zagadnień, które wymagają dalszych badań. Wymieńmy kierunki badawcze sugerowane w [110, s. 68], które są związane z problematyką rozprawy.

Postuluje się dalsze badania nad metodami indukcji reguł. W [110] stwierdzono, że potrzebne są badania nad generowaniem reguł z danych zdefiniowanych zarówno na podstawie atrybutów jakościowych, jak i liczbowych. Dotychczasowe algorytmy w zakresie teorii zbiorów przybliżonych ukierunkowane były na pierwszy typ atrybutów, a dane numeryczne podlegały wstępnej dyskretyzacji przed samą indukcją [33, 128, 154, 155, 156]. Innym rozwiązaniem może być stworzenie algorytmu generacji reguł, w którym dane nie są poddawane takiej dyskretyzacji przed procesem indukcji. Kolejnym kierunkiem jest analiza niekompletnych danych zawierających nieznaną wartość atrybutów. Pewne prace w tym zakresie już rozpoczęto [65, 66, 72, 76, 118, 242]. Odrębną klasę problemów stanowi zastosowanie teorii zbiorów przybliżonych do analizy danych dotyczących wielokryterialnych problemów decyzyjnych. Trudnością jest tutaj niemożność zastosowania klasycznej teorii zbiorów przybliżonych do opisów obiektów za pomocą kryteriów, tzn. atrybutów z dziedzinami uporządkowanymi według preferencji. W ramach klasycznej teorii zbiorów przybliżonych nie jest także możliwe wykrycie niespójności w opisie obiektów związanych z naruszeniem fundamentalnej zasady dominacji w zbiorze obiektów. Greco, Matarazzo i Słowiński [59] zaproponowali uogólnienie teorii zbiorów przybliżonych oparte na zastosowaniu relacji dominacji,

które zapoczątkowało liczne prace badawcze [60–70]. Także w tym zakresie istnieje potrzeba zaproponowania algorytmów indukcji nowego typu reguł o uogólnionej składni.

## 1.2. Cel i zakres rozprawy

Rozprawa poświęcona jest odkrywaniu wiedzy z danych metodami indukcji reguł decyzyjnych z wykorzystaniem teorii zbiorów przybliżonych. Przegląd powyższej tematyki w zakresie perspektyw predykcji i opisu danych przedstawiony w rozdziale 1.1 wskazuje, że problemy indukcji reguł decyzyjnych oraz teorii zbiorów przybliżonych wymagają dalszych systematycznych badań. Potrzeba ta wynika zarówno z luk metodologicznych, jak i z zastosowań praktycznych.

W rozprawie zaprezentowane zostaną nowe metody opracowane przez autora, które wraz z przeprowadzoną dyskusją krytyczną wykorzystania istniejących algorytmów tworzą spójne metodycznie podstawy dla indukcji reguł decyzyjnych jako podejścia do odkrywania wiedzy z danych. W związku z tym sformułowano następujące cele szczegółowe rozprawy:

1. Zaproponowanie metod generowania reguł decyzyjnych uwzględniających miary oceny reguł związane z perspektywą opisu w odkrywaniu wiedzy.
2. Przedstawienie algorytmu indukcji reguł decyzyjnych bezpośrednio z danych numerycznych bez ich wstępnej dyskretyzacji.
3. Uogólnienie teorii zbiorów przybliżonych i metod indukcji reguł decyzyjnych dla przypadku tablic informacyjnych zawierających nieznanne wartości atrybutów.
4. Wprowadzenie algorytmów odkrywania reguł decyzyjnych dla wielokryterialnych problemów sortowania.
5. Przedstawienie zasad budowy dwóch złożonych klasyfikatorów: integrującego algorytm indukcji reguł decyzyjnych z uczeniem się z wcześniej zapamiętanych przypadków oraz klasyfikatora dla problemów wieloklasowych.
6. Dokonanie przeglądu zastosowań niektórych omawianych metod w problemach analizy danych medycznych oraz diagnostyki technicznej.

Praca składa się z 9 rozdziałów. Rozdziały od 3 do 8 dotyczą kolejnych celów szczegółowych rozprawy. Omówimy je krótko poniżej, cytując prace autora związane z poszczególnymi celami. Podkreślimy, że ze względu na ograniczoną objętość rozprawy niektóre opublikowane wyniki szczegółowe nie będą prezentowane w sposób dokładny. Przykładowo dotyczy to części wyników eksperymentalnych, które są dokładnie przedstawione w cytowanych pracach autora.

W rozdziale 2 zawarto krytyczny przegląd podstawowych (z punktu widzenia problematyki dyskutowanej w rozprawie) pojęć związanych z uczeniem maszynowym, reprezentacją przykładów uczących, reguł decyzyjnych oraz algorytmów ich indukcji. Równocześnie krótko omówiono uwzględnianie niespójności opisu obiektów

tów za pomocą teorii zbiorów przybliżonych i wprowadzono klasyfikację algorytmów indukcji reguł wykorzystujących elementy tej teorii [234]. Przeprowadzono też dyskusję sposobów formalnej oceny regułowych klasyfikatorów [221, 233] i zaprezentowano dyskusję krytyczną problemów związanych z perspektywą klasyfikacyjną w indukcji reguł.

W rozdziale 3 omówiono sposoby uwzględniania różnych miar oceny w indukcji reguł. W perspektywie opisu uwzględnia się inne miary oceny reguł niż tylko trafność klasyfikowania, np. dla pojedynczych reguł ocenia się jej ogólność, dokładność czy liczbę przykładów spełniających regułę. Pomimo że perspektywy klasyfikacyjna i opisowa indukcji reguł mogą być postrzegane jako różne, interesujące jest jednak poszukiwanie reguł pozwalających na osiągnięcie satysfakcjonującego kompromisu pomiędzy wartościami różnych miar oceny. Wprowadzono oryginalną metodę indukcji ograniczonego liczbowo zbioru reguł ogólnych, wspartych dostatecznie licznym zbiorem przykładów oraz posiadających akceptowalne właściwości klasyfikujące [147, 245]. Przedstawiono sposób automatyzacji doboru parametrów tej metody i eksperymentalnie oceniono jej działanie na wielu rzeczywistych zbiorach danych [234, 246]. Omówiono także wykorzystanie algorytmu używanego w tej metodzie w celu interaktywnego poszukiwania reguł spełniających wymogi użytkownika co do ich składni [239].

W rozdziale 4 przedstawiono nowy algorytm indukcji minimalnego zbioru reguł z danych zdefiniowanych zarówno na podstawie atrybutów nominalnych, jak i liczbowych. W algorytmie nazywanym MODLEM [235] dane numeryczne nie są poddawane wcześniejszej dyskretyzacji przed procesem indukcji reguł. Rozważania przeprowadzono na tle dotychczasowych propozycji i wykonano studium porównawcze z niektórymi z nich [81].

W rozdziale 5 rozważa się problematykę analizy niekompletnych tablic decyzyjnych za pomocą teorii zbiorów przybliżonych. Na tle przeglądu dotychczasowych propozycji wprowadza się rozróżnienie dwóch różnych semantyk nieznanymi wartościami atrybutów: wartości niedostępnych oraz wartości brakujących. W celu ich formalnego uwzględnienia zaproponowano dwa uogólnienia teorii zbiorów przybliżonych wykorzystujące odpowiednio relacje podobieństwa i wartościowaną relację tolerancji [242]. Przeanalizowano właściwości podstawowych pojęć w proponowanych uogólnieniach. Następnie rozważano uogólnioną postać reguł i zbadano ich właściwości [243]. Omówiono indukcję reguł i ich stosowanie do celów klasyfikacji nowych obiektów [244]. Porównano także proponowane uogólnienia teorii zbiorów przybliżonych z innym dotychczas wprowadzonym uogólnieniem opartym na relacji tolerancji.

Rozdział 6 poświęcono odkrywaniu wiedzy w wielokryterialnych problemach decyzyjnych. Szczególną uwagę zwrócono na problem sortowania [224]. Polega on na przydziale obiektów reprezentujących warianty decyzyjne i opisanych kryteriami do ustalonych a priori i uporządkowanych preferencyjnie klas decyzyjnych. Korzystając z rozwiązań zaproponowanych w rozdziałach 3 i 4, przedstawiono nowe algorytmy indukcji reguł o uogólnionej składni uwzględniającej relację dominacji,

tzn. minimalnego i satysfakcjonującego zbioru reguł [71, 238]. Wprowadzono także rozszerzenie dotychczasowego uogólnienia opartego na relacji dominacji na model o zmiennej spójności, który jest przydatny do analizy dużych zbiorów danych zawierających wiele niespójnych opisów przykładów [70].

W rozdziale 7 omówiono dwie propozycje architektur złożonych systemów klasyfikujących [97, 99, 100, 237, 247]. Są to: system hybrydowy integrujący uczenie się reguł i przypadków oraz klasyfikator wielokrotny  $n^2$ . Systemy te oprócz reguł decyzyjnych wykorzystują algorytmy uczenia się z wcześniej zapamiętanych przypadków. W konstruowaniu części regułowej stosuje się algorytmy indukcji reguł zaproponowane dla realizacji innych celów szczegółowych rozprawy. Przedstawia się także ocenę eksperymentalną obu proponowanych systemów.

Rozdział 8 zawiera dyskusję zastosowania przedstawionych metod do problemów praktycznych. W celu ilustracji takich możliwości rozważa się analizę danych medycznych i technicznych. Autor i współpracownicy analizowali wiele problemów praktycznych w tym zakresie [117, 159, 160, 210–214, 225, 240, 241]. Przedstawiono syntetyczny przegląd niektórych osiągniętych rezultatów oraz dyskusję aspektów metodycznych wynikających z zastosowań praktycznych.

Rozdział 9 zawiera wnioski, uwagi końcowe oraz wskazanie kierunków dalszych badań.

## 2. Reguły decyzyjne, metody ich indukcji i stosowania w problemach klasyfikacji

### 2.1. Pojęcia podstawowe związane z regułą reprezentacją wiedzy w uczeniu maszynowym

Uczenie maszynowe (ang. *machine learning*<sup>1</sup>) obejmuje problematykę konstruowania programów komputerowych, które uczą się rozwiązywać zadania na podstawie doświadczenia reprezentowanego przez zbiór przykładów [149]. Z punktu widzenia niniejszej rozprawy ograniczamy się do zagadnień uczenia nadzorowanego, choć dziedzina uczenia maszynowego obejmuje szerszy zakres, np. uczenia nienadzorowanego, dedukcyjnego uczenia się z intensywnym wykorzystaniem wiedzy czy uczenia się ze wzmocnieniem (przeglądy są dostępne np. w [34, 35, 124, 125, 149, 199]).

W **uczeniu nadzorowanym** (ang. *supervised learning*) proces uczenia się dotyczy poszukiwania hipotez opisujących tzw. pojęcia. Termin **pojęcie** (ang. *concept*) rozumiany jest jako „ogólne określenie oznaczające zbiór (klasę) obiektów posiadających pewne wspólne właściwości, które odróżniają je od innych pojęć” [124]. Nauczyciel (nadzorca) dostarcza przykładów i kontrprzykładów wybranego pojęcia (nazywanych także przykładami pozytywnymi i negatywnymi). Niektórzy autorzy zamiast terminu „pojęcie” używają terminu „kategoria” lub „klasa”. Przez hipotezę rozumie się funkcję przypisującą przykładom ich kategorie. Wynikiem nadzorowanego uczenia się pojęć jest wybór pewnej hipotezy z przestrzeni możliwych hipotez, która jest uznana za najlepiej opisującą pojęcia na podstawie dostarczonych przykładów uczących [34].

W przypadku odkrywania wiedzy z danych informacja historyczna reprezentująca doświadczenie w pewnej dziedzinie przechowywana jest w bazie danych, stąd często znana jest klasyfikacja obiektów pełniących funkcję przykładów. W tym przypadku realny nauczyciel wskazujący przykłady nie jest konieczny. Jeśli wiedza o dziedzinie jest ograniczona i nie dysponuje się znaną klasyfikacją obiektów, to możliwe jest użycie metod analizy danych, np. analizy skupień, w celu odnalezienia klas podobnych obiektów [35].

Funkcje odwzorowujące przykłady na odpowiedni zbiór pojęć mogą być poszukiwane wieloma metodami (patrz przeglądy [13, 149, 263]). W przypadku uczenia maszynowego podkreśla się, że poszukiwana reprezentacja wiedzy klasyfikacyjnej powinna dodatkowo spełniać postulat zrozumiałości (ang. *comprehensibility*) [141, 148], tzn. powinna być wyrażona w formie pozwalającej na zrozumienie, ocenę

---

<sup>1</sup> W literaturze polskojęzycznej można spotkać także inne tłumaczenia tego terminu jako „uczenie się maszyn” lub „maszynowe uczenie się”; porównaj [17, 34, 142].

i dalsze użycie przez człowieka. Postulat ten uwzględnia się w tych metodach uczenia maszynowego inspirowanych badaniami sztucznej inteligencji, które tworzą symboliczne reprezentacje wiedzy, takie jak np. drzewa i reguły decyzyjne czy wyrażenia relacyjne w logice pierwszego rzędu [46].

W ogólności proces budowy reprezentacji wiedzy o klasyfikowaniu, np. w postaci reguł decyzyjnych, obejmuje także fazę testowania, w której reguły weryfikowane są na zbiorze przykładów testowych. Następnie reguły mogą być użyte do klasyfikowania nowych obiektów, dla których nie jest znana decyzja o przydziale obiektów do określonej klasy.

Proces uczenia się scharakteryzowany jest przez: język reprezentacji danych, język reprezentacji hipotez (wiedzy) oraz sam algorytm uczenia się.

### 2.1.1. Reprezentacja danych

Zbiór analizowanych danych, tutaj przykładów uczących, może być reprezentowany na kilka sposobów; najczęściej stosowane są tablice informacyjne lub decyzyjne, zapis w logice pierwszego rzędu. W rozprawie będziemy rozważać wyłącznie reprezentacje w postaci tablic informacyjnych lub tablic decyzyjnych.

Formalnie wprowadza się pojęcie **tablicy informacyjnej** [164, 165]<sup>2</sup> jako pary

$$IT = (U, A) \quad (2.1)$$

gdzie  $U$  jest niepustym i skończonym zbiorem obiektów,  $A$  jest niepustym i skończonym zbiorem atrybutów. Zbiór  $V_a$  jest dziedziną atrybutu  $a \in A$ ,  $V = \bigcup_{a \in A} V_a$  oraz  $f : U \times A \rightarrow V$  jest funkcją informacyjną, taką że  $\forall a \in A, x \in U f(a, x) \in V_a$ .

Atrybuty opisujące obiekty mogą być różnego typu. Najczęściej stosuje się atrybuty typu:

- nominalnego, których dziedziny są zbiorami nieuporządkowanymi (tzn. dla dwóch wartości możliwe jest określenie wyłącznie relacji równości lub różności),
- porządkowego, których dziedziny są zbiorami uporządkowanymi (tzn. dla wartości atrybutu możliwe jest określenie relacji równości, różności, mniejszości lub większości),
- liczbowego, których dziedziny są zdefiniowane na liczbowych skalach pomiarowych: przedziałowych lub ilorazowych (na tych skalach, w odróżnieniu od skal porządkowych, możliwe jest także wykonywanie operacji arytmetycznych).

Osobną kategorię stanowią atrybuty, których dziedziny są uporządkowane według preferencji – atrybuty te nazywa się kryteriami. Rozróżnienie porządkowych, przedziałowych i ilorazowych skal kryteriów wiąże się z ich interpretacją w sensie

---

<sup>2</sup> W pracy [165] użyto nazwy „system informacyjny”.

intensywności preferencji. W niektórych pracach rozważa się także jeszcze inne typy atrybutów, np. strukturalne, których wartości tworzą taksonomie [90, 141].

Niech  $K = \{K_j : j = 1, \dots, r\}$  oznacza zbiór pojęć (klas decyzyjnych) rozważanych w uczeniu nadzorowanym. Określenie pojęć może być podane przez nauczyciela (użytkownika systemu) lub zdefiniowane za pomocą dodatkowego atrybutu opisującego obiekty w tablicy decyzyjnej.

**Tablica decyzyjna**  $DT$  jest tablicą informacyjną o postaci:

$$IT = (U, A \cup \{d\}) \quad (2.2)$$

gdzie  $d \notin A$  jest atrybutem decyzyjnym. Elementy zbioru  $A$  nazywane są atrybutami warunkowymi. Jeżeli tablica decyzyjna zawiera przykłady pojęć, to dla każdego przykładu  $x \in U$  wartość funkcji informacyjnej  $f(d, x) \in K$ .

Dla każdego pojęcia  $K_j$  podaje się zbiór przykładów uczących  $E_{K_j}$  złożony z obiektów  $x \in U$  opisanych ciągiem wartości atrybutów warunkowych  $a_i \in A$ , tj.  $(f(a_1, x), f(a_2, x), \dots, f(a_m, x))$  oraz wartością atrybutu decyzyjnego  $f(d, x) = K_j$ .

### 2.1.2. Reguły decyzyjne

Indukcja reguł decyzyjnych dotyczy zbioru pojęć  $K$ . Zbiór obiektów  $U$  podzielony jest na niepuste podzbiory  $E_{K_i}, E_{K_j}$  parami rozłączne, tzn.  $\forall i \neq j E_{K_i} \cap E_{K_j} = \emptyset$  oraz  $\bigcup_{j=1}^r E_{K_j} = U$ . Niech  $U = E_{K_j}^+ \cup E_{K_j}^-$  dla danego pojęcia  $K_j$  oznacza podział zbioru  $U$  na podzbiór przykładów pozytywnych  $E_{K_j}^+ = E_{K_j}$  oraz przykładów negatywnych  $E_{K_j}^- = U \setminus E_{K_j}$  (tj. pozostałych przykładów).

**Reguła decyzyjna**  $r$  opisująca pojęcie  $K_j$  zdefiniowana jest jako wyrażenie postaci:

$$\text{jeżeli } P \text{ to } Q \quad (2.3)$$

gdzie  $P$  jest częścią warunkową (przesłanką) reguły oraz  $Q$  jest częścią decyzyjną (konkluzją) reguły oznaczającą, że obiekt  $x$  spełniający  $P$  należy do  $K_j$ <sup>3</sup>. W literaturze reguła decyzyjna jest zapisywana także w postaci  $P \rightarrow Q$ .

Część warunkowa  $P$  jest koniunkcją warunków elementarnych  $w$  i jest reprezentowana w postaci:  $P = w_1 \wedge w_2 \wedge \dots \wedge w_k$ , gdzie  $k$  jest liczbą wykorzystanych warunków. Liczbę  $k$  nazywać będziemy długością reguły i oznaczać  $DI(P)$ . Niektórzy autorzy koniunkcję  $P$  nazywają złożeniem warunków lub kompleksem.

**Warunek elementarny**  $w_i$  (selektor [141]) reguły  $r$  jest zdefiniowany jako<sup>4</sup>:

$$(f(a_i, x) \propto \text{term}(a_i)) \quad (2.4)$$

<sup>3</sup> W literaturze stosowane są różne formy zapisu  $Q$ , np. samo  $K_j$ ,  $x \in K_j$ , lub  $f(d, x) = K_j$  w przypadku wyspecyfikowania atrybutu  $d$ .

<sup>4</sup> W literaturze uczenia maszynowego stosuje się także zapis  $(a_i \propto \text{term})$ , traktując atrybut jako zmienną, która może przyjmować wartości [141].

gdzie  $f(a_i, x)$  oznacza wartość atrybutu  $a_i$  dla obiektu  $x$ ;  $\infty$  oznacza operator relacji należący do zbioru  $\{=, \neq, <, \leq, >, \geq, \in\}$ <sup>5</sup>, a *term* oznacza tzw. term elementarny, który jest stałą będącą wartością z dziedziny atrybutu  $a_i$  lub może być uogólniony do podzbioru wartości w przypadku użycia operatora  $\in$  [224].

**Pokryciem** koniunkcji warunków elementarnych  $P$  w danej tablicy decyzyjnej  $DT$ , oznaczonym przez  $[P]$ , jest zbiór obiektów z tablicy  $DT$  spełniających logicznie warunki elementarne reprezentowane przez  $P$ . Niektórzy autorzy nazywają pokrycie  $[P]$  znaczeniem wyrażenia  $P$  w ustalonej tablicy decyzyjnej [110]. Pokrycie  $[P]$  można podzielić na część pozytywną  $[P]_{K_j}^+ = [P] \cap E_{K_j}^+$  oraz negatywną  $[P]_{K_j}^- = [P] \cap E_{K_j}^-$ .

Reguła decyzyjna  $r$  jest dyskryminująca, tj. odróżnia przykłady należące do  $K_j$  od przykładów negatywnych wtedy i tylko wtedy, gdy jej część warunkowa spełnia warunek niesprzeczności  $[P]_{K_j}^- = \emptyset$ . Wymaga się równocześnie, aby  $[P]_{K_j}^+ \neq \emptyset$ .

Dyskryminująca reguła decyzyjna  $r$  jest minimalna, jeśli usunięcie jakiegokolwiek warunku elementarnego  $w$  z jej części warunkowej  $P$  prowadzi do  $[P]_{K_j}^- \neq \emptyset$ .

Zbiór reguł decyzyjnych  $R$  jest kompletnym opisem pojęcia  $K_j$ , jeśli każdy przykład  $x \in E_{K_j}^+$  spełnia część warunkową i decyzyjną przynajmniej jednej reguły  $r \in R$ . Michalski [141, 142] podał, że tworzenie regułowego opisu klas może przyjmować postać tzw. opisu dyskryminującego, który musi spełniać wymóg zupełności oraz zawierać dyskryminujące i minimalne reguły. Pojęcie minimalnego opisu dyskryminującego oznacza, że zbiór reguł jest minimalny, tzn. nie istnieje żaden jego podzbiór, który spełnia warunki zupełności i niesprzeczności.

### 2.1.3. Indukcja reguł decyzyjnych

Problem znalezienia minimalnego zbioru reguł, który pokrywa zbiór przykładów i poprawnie je klasyfikuje, jest NP-zupełny. W dowodach wykorzystuje się transformację tego problemu do problemu minimalnego pokrycia zbioru [4, 206, 245]. Podobny wynik podano wcześniej dla indukcji drzew decyzyjnych [93]. Dlatego stosuje się algorytmy przybliżone. Najbardziej znane z dotychczas zaproponowanych opierają się na zasadzie generowania kolejnych pokryć (ang. *sequential covering*). Polega ona na uczeniu się pojedynczej reguły, usuwaniu przykładów, które ona pokrywa, i powtarzaniu procesu dla pozostałych przykładów. W rezultacie powstaje zbiór reguł pokrywających rozważany zbiór przykładów. W [149] podaje się ogólny schemat takiego algorytmu:

---

<sup>5</sup>Operatory inne niż „=”, „≠” i „∈” stosuje się do atrybutów, których dziedziny są uporządkowane.



**Sequential covering** ( $K_j$  klasa;  $A$  atrybuty;  $E$  przykłady,  $\tau$  próg akceptacji);

**begin**

$R := \emptyset;$     {zbiór poszukiwanych reguł}

$r := \text{learn-one-rule}(\text{klasa } K_j; \text{ atrybuty } A; \text{ przykłady } E)$

**while**  $\text{evaluate}(r, E) > \tau$  **do**

**begin**

$R := R \cup r;$

$E := E \setminus R;$     {usuń przykłady pozytywne pokryte przez  $R$ }

$r := \text{learn-one-rule}(\text{klasa } K_j; \text{ atrybuty } A; \text{ przykłady } E);$

**end;**

**return**  $R$

**end.**

Zakładamy, że dostępna jest funkcja *learn-one-rule*, która dla danego zbioru przykładów znajduje jedną regułę pokrywającą możliwie jak najwięcej przykładów pozytywnych i jak najmniej negatywnych. Funkcja może być zrealizowana różnymi sposobami w zależności od algorytmu (przykłady w [25, 36, 37, 73, 141, 145, 235]). W wielu z tych algorytmów początkowy kandydat na część warunkową reguły pokrywa zbiór wszystkich przykładów, w tym negatywnych. Następnie podlega specjalizacji poprzez dodawanie warunków elementarnych, dopóki nie zostanie spełniony warunek akceptacji reguły (np. niesprzeczności w algorytmie AQ). *Evaluate* jest zależną od użytkownika funkcją oszacowania jakości zbioru reguł. Na przykład, kolejnych reguł poszukuje się, dopóki, pozostają przykłady pozytywne z klasy  $K_j$  nie pokryte przez żadną z dotychczas wygenerowanych reguł [73, 141]. Można też za pomocą progu  $\tau$  zdefiniować warunek wcześniejszego zakończenia poszukiwania zbioru reguł [37]. Algorytm powtarza się iteracyjnie oddzielnie dla każdej klasy  $K_j$  lub dla pełnej klasyfikacji. W drugim przypadku należy odpowiednio rozróżniać przykłady pozytywne [36, 37]. Należy zauważyć, że przedstawiony schemat wykorzystuje heurystykę zachłanną, gdyż usuwa przykłady pokryte przez wygenerowaną regułę. Podobnie heurystycznie ogranicza się przestrzeń przeszukiwania warunków elementarnych w trakcie budowy pojedynczej reguły [245]. W poszukiwaniu koniunkcji części warunkowej reguł można także wykorzystać metody wnioskowania boolowskiego [207].

Do oceny kandydatów na części warunkowe reguł stosuje się różne miary w zależności od algorytmu. Michalski [145] proponuje rozważać szereg kryteriów, które ustawione są w porządku leksykograficznym. W [124] jako najczęściej stosowane kryteria oceny wymienia się:

- maksymalizację liczby przykładów pozytywnych pokrywanych przez koniunkcję  $P$ , tj.  $|[P]_{K_j}^+|$ , gdzie  $|\cdot|$  oznacza licznosc zbioru,
- maksymalizację stosunku liczby pokrywanych przykładów pozytywnych do ogólnej liczby wszystkich przykładów, tj.  $|[P]_{K_j}^+|/|[P]|$ ,

- minimalizację liczby użytych warunków elementarnych, tj. minimalizację długości  $DI(P)$ .

Inne algorytmy (np. [25, 37]) wykorzystują do oceny koniunkcji warunków  $P$  miarę entropii informacji (porównaj jej użycie podczas indukcji drzew decyzyjnych [185]). Niech  $S$  oznacza zbiór przykładów pokrytych przez  $P$ . Miara **entropii** zdefiniowana jest jako:

$$\text{Ent}(S) = -\sum_{i=1}^r p_i \cdot \log_2 p_i, \quad (2.5)$$

gdzie  $p_i$  jest prawdopodobieństwem wystąpienia klasy  $K_i$  w zbiorze przykładów  $S$ , a  $r$  jest liczbą klas decyzyjnych. Im mniejsza wartość, tym lepsza koniunkcja  $P$ .

Inną miarą oceniającą zależność wyrażen  $P$  i  $Q$  jest  $m$ -estymata prawdopodobieństwa podana przez Cestnika [27]:

$$me_{K_i}(P) = \frac{n_c + m \cdot p_i}{n + m}, \quad (2.6)$$

gdzie  $n_c$  jest liczbą przykładów pozytywnych pokrywanych przez koniunkcję  $P$ ,  $n$  jest ogólną liczbą wszystkich przykładów pokrywanych przez koniunkcję  $P$ ,  $p_i$  jest prawdopodobieństwem a priori klasy  $K_i$ ,  $m$  – stałą zależną od danych.

Szczególnym przypadkiem miary  $me_{K_i}(P)$  jest **estymata Laplace'a**

$$Lp(P) = \frac{n_c + r - 1}{n + r}, \quad (2.7)$$

gdzie  $r$  jest liczbą klas decyzyjnych. Właściwości tej miary omówiono w [27, 36].

W podejmowaniu decyzji o wyborze najlepszych warunków elementarnych podczas indukcji reguł można także korzystać z zasady **minimalnej długości opisu** (ang. *Minimal Description Length*) [191; 34, rozdz. 5.5; 149 s. 171]. Zasada ta jest związana z kodowaniem danych oraz reprezentacji hipotez (patrz przykłady podane w [34]). W łącznym kodzie wyróżnia się część związaną z dopasowaniem hipotezy do przykładów uczących (interpretowaną jako błąd na zbiorze uczącym) oraz część związaną ze złożonością hipotezy (rozumianej jako długość jej optymalnego kodu). Zasada minimalności opisu mówi, że dla danego zbioru przykładów i różnych hipotez należy wybrać hipotezę, która minimalizuje łączną długość kodu. W myśl typowej interpretacji [34, 149] uznaje się, że ta zasada uwzględnia dokładność i złożoność hipotez w jednym uniwersalnym kryterium. Stosowana jest ona do wielu problemów w uczeniu maszynowym (patrz [34, rozdz. 5.5]), w szczególności znane są dobre wyniki w zakresie wyboru warunków w konstrukcji drzew decyzyjnych oraz redukcji rozmiarów wygenerowanych drzew [139, 188].

Historycznie pierwszym algorytmem opartym na zasadzie pokryć był algorytm AQ podany przez Michalskiego [140]. Jego najważniejszym operatorem budowy hipotez była tzw. gwiazda specjalizująca maksymalnie ogólne warunki złożone pokrywające wybrany początkowy przykład pozytywny w celu odróżnienia ich od

przykładów negatywnych. W pierwotnej postaci AQ tworzył minimalne opisy dyskryminujące pokrywające wszystkie przykłady. Podejście to było rozwijane w szeregu algorytmów AQ11, AQ15 i AQ17 oraz w systemie INLEN [145, 146].

Algorytm CN2 [36] łączy inspiracje ideą algorytmu AQ z technikami znanymi z indukcji drzew decyzyjnych w celu lepszego uwzględniania „zazsumionych” przykładów (patrz [148, s. 73]). Stosuje także inny sposób tworzenia hipotez od ogólnej do szczegółowej. Algorytm ten jest rozwijany w [37, 45] i wykorzystywany w systemie odkrywania wiedzy MLC++.

Przeszukiwanie przestrzeni hipotez od ogólnej do szczegółowej realizowane jest także w algorytmie Itrule, który korzysta z zasad wnioskowania probabilistycznego. W rezultacie powstają reguły probabilistyczne o postaci „jeśli ... to ... z prawdopodobieństwem ...” [229]. Inne znane algorytmy to PRISM [25] zgodny z zasadą kolejnych pokryć, lecz w inny sposób poszukujący najlepszej reguły, oraz HCV wykorzystujący tzw. macierze rozszerzeń. Cios i Liu [35] podają algorytm hybrydowy łączący koncepcje budowy reguł i drzew, stosując do uczenia się pokryć model programowania całkowitoliczbowego. Jeszcze inne zasady przeszukiwania przestrzeni hipotez zaproponowano w algorytmie PVM, gdzie poszukuje się reguł o ustalonej maksymalnej długości, wybierając takie, które maksymalizują zdolności predykcyjne [263]. Poszukiwanie prostych kandydatów na reguły o dobrych właściwościach klasyfikacyjnych zrealizował Holte w [92] (porównaj także [4, 189, 197]). Powyższy przegląd nie wyczerpuje wszystkich prac, lecz pokazuje najczęściej stosowane algorytmy.

Podsumowując, należy wspomnieć o możliwości zamiany struktury drzewa decyzyjnego na zbiór reguł. Drzewa decyzyjne mogą być generowane na przykład za pomocą algorytmu ID3 lub C4.5 [185]. Każde drzewo może być zamienione na zbiór reguł, tzn. ścieżki prowadzące od korzenia do liścia reprezentują koniunkcję testów odpowiadających części warunkowej reguły. W algorytmie C4.5 stosuje się bardziej złożone techniki upraszczające strukturę reguł, które są ostatecznie prezentowane w postaci listy uporządkowanej, gdzie reguły są ułożone zgodnie z ich przydatnością do klasyfikowania nowych przypadków [185].

## **2.2. Uwzględnianie niespójności informacji za pomocą teorii zbiorów przybliżonych oraz reguł decyzyjnych**

Teoria zbiorów przybliżonych (ang. *rough sets*) została zaproponowana przez Pawłaka [164, 165] jako narzędzie analizy informacji granularnej. Granularność dostępnej informacji może powodować niespójność opisu obiektów, a teoria zbiorów przybliżonych dostarcza właśnie podstaw do uwzględniania tego typu niespójności. Teoria oparta jest na założeniu, że posiadając informację reprezentowaną za pomocą atrybutów i ich wartości na obiektach, możliwe jest określenie relacji zachodzącej pomiędzy tymi obiektami. Obiekty posiadające ten sam opis, wyrażony za pomocą atrybutów, są nierozróżnialne ze względu na dostępną informację.

Zakładamy, że informacja o obiektach dostępna jest w postaci tablicy informacyjnej  $IT = (U, A)$  zgodnej z definicją (2.1). W przypadku uczenia nadzorowanego informacja o klasyfikacji obiektów może być wyrażona za pomocą atrybutu decyzyjnego  $d$ . Prowadzi to do reprezentacji w postaci tablicy decyzyjnej  $DT = (U, A \cup \{d\})$  zgodnej z definicją (2.2).

Niech  $IT = (U, A)$ ; z każdym podzbiorem atrybutów  $B \subseteq A$  związana jest **relacja nierozróżnialności** (ang. *indiscernibility*)  $IND(B)$  zdefiniowana jako:

$$IND(B) = \{(x, y) \in U \times U : \forall a \in B \ f(a, x) = f(a, y)\} \quad (2.8)$$

Jeśli  $(x, y) \in IND(B)$ , to obiekty  $x$  i  $y$  są nierozróżnialne ze względu na podzbiór  $B$ . Relacja nierozróżnialności jest zwrotna, symetryczna i przechodnia. Rodzinę wszystkich klas relacji  $IND(B)$ , tj. podział  $U$  za pomocą  $B$ , oznacza się przez  $U / IND(B)$ . Klasy nazywane są zbiorami  $B$ -elementarnymi.  $I_B(x)$  oznacza klasę relacji  $IND(B)$  zawierającą obiekt  $x$ .

Relacja nierozróżnialności została użyta w oryginalnej wersji teorii zbiorów przybliżonych. Nie jest to jedyna możliwa relacja pomiędzy obiektami. Relacje **tolerancji** oraz **podobieństwa** są wykorzystywane w rozszerzeniach teorii zbiorów przybliżonych [178, 208, 226, 227, 248, 249] (patrz także rozdział 5).

Relacja pomiędzy obiektami jest użyta do zdefiniowania pojęcia **zbioru przybliżonego**. Niech  $IT = (U, A)$  będzie tablicą informacyjną,  $X$  niepustym podzbiorem  $U$  oraz  $B \subseteq A$ .  **$B$ -dolne przybliżenie** i  **$B$ -górne przybliżenie**  $X$ , oznaczone odpowiednio  $\underline{B}(X)$  i  $\overline{B}(X)$ , jest definiowane jako:

$$\underline{B}(X) = \{x \in U : I_B(x) \subseteq X\} \quad (2.9)$$

$$\overline{B}(X) = \bigcup_{x \in X} I_B(x) \quad (2.10)$$

Zbiór  $BN_B(X) = \overline{B}(X) - \underline{B}(X)$  jest nazywany  **$B$ -granicą** (lub obszarem wątpliwym) zbioru  $X$ . Pomiedzy przybliżeniami zachodzi relacja  $\underline{B}(X) \subseteq X \subseteq \overline{B}(X)$ .

Dolne przybliżenie  $\underline{B}(X)$  jest zbiorem obiektów, które można z pewnością zaliczyć do  $X$  na podstawie dostępnego opisu, podczas gdy obiekty z  $\overline{B}(X)$  mogą być tylko uznane za możliwie należące do  $X$ . Granica  $BN_B(X)$  zawiera obiekty, których nie można jednoznacznie przydzielić do  $X$  z uwagi na sprzeczny opis. Natomiast obiekty z  $U \setminus \overline{B}(X)$  nie należą do  $X$ . O zbiorze  $X$  mówi się, że jest przybliżony (ang. *rough*), jeśli  $BN_B(X) \neq \emptyset$ ; w przeciwnym razie jest  $B$ -definiowalny [165].

Zbiór przybliżony  $X$  może być scharakteryzowany ilościowo za pomocą współczynnika **dokładności przybliżenia**:

$$\alpha_B(X) = \frac{|\underline{B}(X)|}{|\overline{B}(X)|} \quad (2.11)$$

gdzie  $|X|$  oznacza licznosc niepustego zbioru  $X$ .

Definicje powyższe mogą być rozszerzone na klasyfikacje obiektów. Niech będzie dana tablica decyzyjna  $DT = (U, A \cup \{d\})$ , gdzie zbiór  $V_d$  wartości atrybutu decyzyjnego  $d$  jest równy  $\{v_d^1, \dots, v_d^r\}$ . Atrybut decyzyjny  $d$  definiuje podział  $\text{CLASS}_{DT}(d) = \{X_1, X_2, \dots, X_r\}$  zbioru obiektów  $U$ , gdzie  $X_k = \{x \in U : f(d, x) = v_d^k\}$  dla  $1 \leq k \leq r$ . Podział  $\text{CLASS}_{DT}(d)$  jest nazywany klasyfikacją obiektów w  $DT$ , a  $X_k$  jest  $k$ -tą **klasą decyzyjną**. Zbiór  $\underline{B}(X_1) \cup \underline{B}(X_2) \cup \dots \cup \underline{B}(X_r)$  jest  $B$ -dolnym przybliżeniem klasyfikacji  $\text{CLASS}_{DT}(d)$ . Nazywany jest także  $B$ -pozytywnym obszarem klasyfikacji i oznaczany  $\text{POS}_B(d)$ . Współczynnik

$$\gamma(B, d) = \frac{|\text{POS}_B(d)|}{|U|} = \frac{\sum_{k=1}^r |\underline{B}(X_k)|}{|U|} \quad (2.12)$$

jest nazywany **jakością przybliżenia klasyfikacji**  $\text{CLASS}_{DT}(d)$ . Współczynnik  $\gamma(B, d)$  przyjmuje wartości z zakresu  $[0, 1]$ . Jeśli  $\gamma(B, d) = 1$ , to klasyfikacja może być w pełni opisana na podstawie zbioru atrybutów  $B$ , natomiast gdy  $\gamma(B, d) < 1$ , to może być opisana tylko w sposób przybliżony. Współczynnik ten odpowiada także pojęciu poziomu zależności atrybutu  $d$  od zbioru atrybutów  $B$  [110].

Współczynnik jakości może być wykorzystany do zdefiniowania tablic decyzyjnych spójnych (ang. *consistent*) i niespójnych (ang. *inconsistent*). Tablica decyzyjna  $DT = (U, A \cup \{d\})$  jest spójna (lub deterministyczna) wtedy i tylko wtedy, gdy  $d$  zależy jednoznacznie od zbioru atrybutów  $A$  ( $\gamma(A, d) = 1$ ). Gdy ten warunek jest niespełniony ( $\gamma(A, d) < 1$ ), to tablica decyzyjna jest niespójna. Łatwo zauważyć, że tablica jest zgodna, gdy  $\text{POS}_A(d) = U$ . Tablica jest niespójna, jeśli zawiera niespójne przykłady.

Formalnie kryterium zgodności tablicy decyzyjnej może być określone za pomocą funkcji  $\delta_A : U \rightarrow P(V_d)$  nazywanej uogólnioną decyzją i zdefiniowanej dla  $\forall x \in U$  następująco [202]:

$$\delta_A(x) = \{i : \exists x' \in U \ x' \text{IND}(A)x \text{ oraz } f(d, x') = i\} \quad (2.13)$$

Tablica jest spójna, jeśli dla każdego obiektu  $x \in U$  zachodzi  $|\delta_A(x)| = 1$ . W przeciwnym przypadku tablica jest niespójna.

Teoria zbiorów przybliżonych jest także przydatna do wykrywania nadmiarowości danych w tablicy informacyjnej. Wykorzystuje się pojęcie tzw. reduktu. **Reduktem** w tablicy  $IT = (U, A)$  nazywamy podzbiór  $B \subseteq A$  wtedy i tylko wtedy, gdy:

- dla każdego  $x \in U$  zachodzi  $I_B(x) = I_A(x)$ ,
- dla każdego podzbioru  $C \subset B$  pierwszy warunek jest niespełniony.

W przypadku tablicy decyzyjnej  $DT = (U, A \cup \{d\})$  można podać inną definicję reduktu. Podzbiór  $B \subseteq A$  nazywany jest **reduktem względnym** w tablicy  $DT$  wtedy i tylko wtedy, gdy:

- $\gamma(B, x) = \gamma(A, x)$ ,

– dla każdego podzbioru  $C \subset B$  pierwszy warunek jest niespełniony.

W danej tablicy może wystąpić więcej niż jeden redukt. Część wspólną wszystkich reduktów nazywa się jądrem (ang. *core*). Jądro zawiera te atrybuty, których usunięcie z tablicy decyzyjnej powoduje spadek jakości przybliżenia klasyfikacji [110, 165]. Należy jednak pamiętać, że pojęcie reduktu jest zdefiniowane wyłącznie dla obiektów obecnych w tablicy decyzyjnej (tzw. założenie świata zamkniętego [165]) i nie uwzględnia obiektów znajdujących się poza tablicą decyzyjną (założenie świata otwartego). Przydatność reduktu do selekcji atrybutów w zadaniach klasyfikowania nowych obiektów nie jest jednoznaczna.

Pojęcie reduktu może być lokalnie ograniczone do obiektu  $x$ . Reduktem związanym z obiektem  $x \in U$  nazywamy minimalny podzbiór atrybutów  $B \subseteq A$ , który pozwala odróżnić  $x$  (precyzyjniej – jego klasę nierozróżnialności  $I_B(x)$ ) od innych obiektów z  $U$ .

Poszukiwanie wszystkich reduktów jest problemem o złożoności wykładniczej [110]. Ponadto problem poszukiwania minimalnego reduktu, tj. reduktu o minimalnej liczbie atrybutów spośród wszystkich reduktów, jest NP-zupełny [206]. Zaproponowano kilka algorytmów dokładnych [123, 206, 252]. Często stosowane jest tzw. wnioskowanie boolowskie [110, 202, 206]. Wykorzystuje ono konstrukcję macierzy rozróżnialności oraz przekształcenia odpowiadającej jej funkcji rozróżnialności. Zbiór pierwszych implikantów tej funkcji pozwala na wyznaczenie globalnych reduktów (względnych) lub lokalnych reduktów (względnych) związanych z pojedynczymi obiektami. Oprócz algorytmów dokładnych poszukiwania reduktów proponuje się także algorytmy przybliżone, np. wykorzystujące algorytmy genetyczne [267].

Na podstawie tablicy decyzyjnej  $DT = (U, A \cup \{d\})$  można skonstruować zbiór reguł decyzyjnych  $R$ . **Reguły decyzyjne** mają następującą postać<sup>6</sup>:

$$\text{jeżeli } P \text{ to } (f(d, x) = v_d^j) \quad (2.14)$$

gdzie część warunkowa  $P$  jest koniunkcją warunków  $(f(a_i, x) = v_{ai})$ , a atrybut decyzyjny  $d$  przyjmuje wartości z dziedziny  $V_d$ . W uogólnieniach zbiorów przybliżonych stosuje się także inne formy reprezentacji reguł zbliżone do formuły (2.3).

Reguły decyzyjne konstruowane są na podstawie opisów przykładów zawartych w tablicach decyzyjnych. Jeśli tablica jest niespójna, to reguły decyzyjne generowane są dla przybliżeń klas decyzyjnych. Dla wybranego przybliżenia (dolnego lub górnego) klasy decyzyjnej  $X_k$  ( $k = 1, \dots, r$ ) reguły decyzyjne są spełnione przez przykłady z tego przybliżenia i równocześnie nie są spełnione przez przykłady nie należące do tego przybliżenia. W konsekwencji użycia do indukcji przybliżeń klas decyzyjnych jako zbioru przykładów pozytywnych wyróżnia się dwa podstawowe typy reguł:

<sup>6</sup> W definicji podanej przez Skowrona [202] reguła decyzyjna jest podana w postaci  $\phi \rightarrow (d = v)$ , gdzie  $\phi$  jest koniunkcją wyrażeń atomowych  $(a_i = w)$ .

- pewne (ang. *certain*),
- możliwe (ang. *possible*).

Reguły pewne generowane są z przykładów należących do dolnych przybliżeń, a reguły możliwe z przykładów należących do górnych przybliżeń klas decyzyjnych (porównaj [73, 82, 234]). Poniższe oznaczenia użyte są do omówienia tych typów reguł.

Niech  $[w_i]$  oznacza zbiór obiektów spełniających wyrażenie  $w_i$ , tzn.  $\{x \in U : f(a_i, x) = v_{ai}\}$ . Zauważmy, że  $[w_i \wedge w_j] = [w_i] \cap [w_j]$ . Dla uproszczenia regułę (2.14) będziemy zapisywać w postaci:

$$\text{jeżeli } P \text{ to } Q,; \quad \text{gdzie } Q = (f(d, x) = v_d^j). \quad (2.15)$$

Mówimy, że obiekt  $x$  jest dopasowany do części warunkowej reguły  $r$  lub reguła **pokrywa** (ang. *covers*) obiekt, jeśli:

$$x \in [P] \quad (2.16)$$

Obiekt  $x$  **wspiera** (ang. *supports*) regułę  $r$ , jeśli:

$$x \in [P \wedge Q] \quad (2.17)$$

Niech  $\text{CLASS}_{DT}(d) = \{X_1, X_2, \dots, X_r\}$  będzie klasyfikacją zbioru obiektów  $U$ , gdzie  $X_j = \{x \in U : f(d, x) = v_d^j\}$  dla  $1 \leq j \leq r$ .

Reguła decyzyjna  $r$  (2.15) jest regułą pewną w tablicy  $DT$ , jeśli:

$$[P] \subseteq [Q] \quad (2.18)$$

Każdy obiekt spełniający część warunkową należy do dolnego przybliżenia klasy decyzyjnej  $X_j$ .

Niech  $X_j = [Q]$ ; reguła decyzyjna  $r$  jest regułą możliwą w tablicy  $DT$ , jeśli:

$$[P] \subseteq \overline{A}(X_j). \quad (2.19)$$

Oznacza to, że każdy obiekt spełniający część warunkową należy do górnego przybliżenia klasy decyzyjnej  $X_j$ .

Pewna reguła decyzyjna  $r$  jest **minimalna**, jeśli żadna inna reguła  $r'$  spełniająca warunek (2.18) nie może być skonstruowana z podzbioru warunków wykorzystywanych w części warunkowej reguły  $r$ . Analogiczną definicję minimalności można sformułować dla reguły możliwej.

W literaturze [170, 234] rozważa się także reguły niedeterministyczne zamiast reguł możliwych. Reguły te pokrywają wyłącznie niespójne obiekty. Niech  $Y_l$  oznacza zbiór obiektów posiadających identyczny zbiór wartości uogólnionej decyzji  $\delta_A(x) = \theta$ , gdzie  $|\theta| > 1$ . Reguły niedeterministyczne pokrywają wyłącznie obiekty należące do  $Y_l$  i mają następującą postać:

$$\text{jeżeli } P \text{ to } (\delta_A(x) = \theta). \quad (2.20)$$

Część decyzyjna zapisywana jest także w postaci  $\vee(f(d,x) = v_d)$  [224]. Podobnie jak w przypadku reguł pewnych i możliwych wymaga się, aby  $[P] \subseteq [Y_i]$  oraz aby reguła była minimalna. Niektórzy autorzy używają także innych postaci reguł, np. częściowo dyskryminujących [274]; porównaj także [118, 129, 254, 259].

Teoria zbiorów przybliżonych jest wykorzystywana w różnym stopniu zarówno w indukcji reguł, jak i we wstępnym przetwarzaniu danych. W rezultacie zaproponowano wiele algorytmów indukcji reguł korzystających z elementów zbiorów przybliżonych (porównaj przeglądy w [11, 73, 110, 203, 234]). Można je podzielić na trzy podstawowe kategorie:

1. Algorytmy indukcji minimalnego zbioru reguł.
2. Algorytmy indukcji wyczerpującego zbioru reguł.
3. Algorytmy indukcji satysfakcjonującego zbioru reguł.

Pierwsza kategoria ukierunkowana jest na pokrycie zbioru wszystkich obiektów za pomocą minimalnego zbioru reguł. Druga kategoria ukierunkowana jest na poszukiwanie zbioru wszystkich reguł w przyjętej składni, które można wygenerować z danej tablicy. Trzecia kategoria algorytmów generuje ograniczony zbiór wszystkich reguł spełniających pewne warunki, np. minimalna liczba przykładów wspierających, maksymalna długość części warunkowej [234].

Poszukiwanie optymalnego minimalnego zbioru reguł jest problemem NP-zupełnym [245]. Poszukiwanie wyczerpującego zbioru wszystkich reguł jest problemem o złożoności wykładniczej [206]. W przypadku trzeciej kategorii koszty obliczeniowe mogą być zredukowane poprzez użycie właściwych ograniczeń generowanych reguł [121, 245].

W zakresie pierwszej kategorii proponuje się najczęściej algorytmy przybliżone o złożoności wielomianowej. Najpopularniejszy jest algorytm LEM2, zaproponowany przez Grzymałę [73]. Tworzy on minimalny opis dyskryminujący przybliżenia każdej klasy decyzyjnej za pomocą oryginalnej propozycji zbliżonej do zasady generowania kolejnych pokryć. Dostępne są także liczne modyfikacje tego algorytmu dostosowane do różnych niedoskonałości danych [76–78]. Przykładami innych algorytmów, które można zaliczyć do tej kategorii, są LEM1 [83], CPLA [85], MODLEM [235], algorytmy oparte na współczynniku dyskryminującym [266] i w pewnym sensie Probrough [129]. Możliwe jest także zastosowanie do znajdowania tego typu reguł metod wywodzących się z wnioskowania boolowskiego [207].

Druga kategoria algorytmów opiera się na obserwacji, że reduktory różnego typu (zwłaszcza względne związane z obiektami) mogą być przydatne do syntezy zbioru reguł decyzyjnych [202, 165, 170]. Taka redukcja tablicy decyzyjnej może być wykonana wieloma sposobami, lecz najczęściej stosuje się metodę wnioskowania boolowskiego opartą na funkcjach rozróżnialności, zaproponowaną przez Skowrona [202]. Przegląd różnych algorytmów wykorzystujących wnioskowanie boolowskie podany jest w [11] (patrz także [110, 170, 203, 248]). Zbiór wszystkich reguł



wygenerowanych z danych zawierających tzw. „szum” lub różne „utrudnienia” może być jednak zbyt „dopasowany” do przykładów uczących i nieskuteczny w klasyfikacji nowych obiektów. Proponuje się więc rozszerzenia podstawowych metod wnioskowania boolowskiego dla takich sytuacji. Jedną z możliwości jest stosowanie tzw. reduktów przybliżonych (ang. *approximate reducts*), które tylko do pewnego stopnia zachowują relację nierozróżnialności; patrz np. [110, 254]. Innym rozwiązaniem jest poszukiwanie tzw. reduktów dynamicznych [11, 12]. W tym rozwiązaniu z początkowej tablicy decyzyjnej w sposób losowy wybiera się określoną liczbę różnych podzbiorów obiektów. Dla każdej tablicy decyzyjnej zdefiniowanej na podstawie wylosowanego podzbioru obiektów poszukuje się zbioru reduktów. Ostatecznie wybiera się najbardziej stabilne redukty, które występują w większości tabel. Wyniki eksperymentalne [11] wskazują, że zbiory reguł tworzone z wykorzystaniem powyższych dynamicznych reduktów są bardziej odporne na różnego rodzaju „zaburzenia” danych i mogą prowadzić do lepszej trafności klasyfikowania nowych obiektów. Część algorytmów opartych na wnioskowaniu boolowskim jest zaimplementowana w systemie ROSETTA [161]. Wnioskowanie boolowskie wykorzystywane jest także w poszukiwaniu reguł w modelu zmiennej precyzji [274], choć Ziarko proponuje przyrostową wersję algorytmu [198]. Inne algorytmy zaliczane do tej kategorii, a nie wykorzystujące wnioskowania boolowskiego, to np. „wykładnicze opuszczanie warunków” [73].

Tworzenie satysfakcjonującego zbioru reguł wymaga wyspecyfikowania ograniczeń, które muszą spełniać reguły. Odnoszą się one do akceptowalności przez użytkownika wartości wybranych miar oceny reguły (zostanie to omówione szczegółowo w rozdziale 3). Taki zbiór reguł może być zbudowany na dwa sposoby. Pierwszy polega na wygenerowaniu zbioru wszystkich reguł i następnie jego przefiltrowaniu w celu odnalezienia interesujących reguł [109, 245]. Drugi sposób polega na wprowadzeniu bezpośrednio do algorytmu ograniczeń przeszukiwania przestrzeni hipotez i generowaniu interesujących reguł bezpośrednio z danych, np. za pomocą algorytmu Explore [147, 234]. Do tej grupy można także zaliczyć modyfikacje algorytmu poszukiwania reguł asocjacyjnych [1] na potrzeby poszukiwania reguł pewnych i możliwych [16, 121, 130].

Poza omawianymi powyżej algorytmami istnieje grupa algorytmów inspirowana modelami probabilistycznymi, np. PRIMOSE [259] oraz ProbRough [129]. W szczególności ProbRough pozwala uzyskiwać dobrą skuteczność klasyfikowania. Kowalczyk zaproponował także specjalne podejście do eksploracji dużych baz danych nazywane przybliżonymi modelami danych (ang. *rough data modelling*) [114]. Polega ono na poszukiwaniu w tablicy decyzyjnej wielu modeli opartych na ograniczonej liczbie skupień – klas relacji równoważności na podzbiórach atrybutów  $C \subseteq A$  (na ogół rozważa się podzbiory zawierające 2–5 atrybutów). Na podstawie opisu uporządkowanych skupień buduje się reguły decyzyjne. Spośród wielu przeszukiwanych prostych modeli wybiera się najskuteczniejszy z uwagi na wybraną miarę oceny; patrz [114, 134].

Przegląd implementacji różnych algorytmów indukcji reguł decyzyjnych wykorzystujących elementy teorii zbiorów przybliżonych jest dostępny w [176].

### 2.3. Stosowanie reguł decyzyjnych do klasyfikowania nowych obiektów

W perspektywie predykcji reguły decyzyjne wygenerowane z przykładów uczących używane są do klasyfikowania nowych obiektów. Przez nowe obiekty rozumie się obiekty, które nie służyły do indukcji. Znany jest ich opis za pomocą wartości atrybutów, natomiast celem klasyfikowania jest określenie przydziału takiego obiektu do klasy decyzyjnej. Jeżeli dla klasyfikowanego obiektu znana jest ponadto jego rzeczywista klasyfikacja to nazywamy go przykładem testowym, gdyż możliwe jest wtedy porównanie proponowanej decyzji klasyfikacyjnej z rzeczywistą. Klasyfikowanie obiektów opiera się na dopasowaniu opisu obiektu do części warunkowych reguł decyzyjnych. Wyróżniamy dopasowanie całkowite i częściowe.

Niech  $R$  oznacza zbiór reguł uzyskany w wyniku indukcji oraz niech  $R_k \subset R$  oznacza podzbiór reguł odpowiadający pojedynczej klasie decyzyjnej  $X_k$ . Reguły są reprezentowane w najogólniejszej postaci (2.3). Rozpatrzmy zbiór nowych obiektów  $E^n$ . Niech  $e \in E^n$  oznacza obiekt, który ma zostać sklasyfikowany. Jego opis można zapisać jako  $\bigwedge_{a_i \in A} (f(a_i, e) = v_{ie})$ , gdzie  $v_{ie}$  jest wartością atrybutu  $a_i$  z dziedziny  $V_{a_i}$ . Zbiór atrybutów wykorzystany do skonstruowania części warunkowej reguły  $r$  oznaczmy przez  $C_r$ . W przypadku nie wyspecyfikowania któregoś z atrybutów najczęściej przyjmuje się, że obiekt może przyjąć dowolną wartość z dziedziny tego atrybutu podczas dopasowania do reguły.

Mówimy o całkowitym dopasowaniu (ang. *complete matching*) obiektu  $e$  do części warunkowej reguły  $r$ , jeśli:

$$\forall a_i \in C_r \quad (f(a_i, e) \propto \text{term}(a_i)) \quad (2.21)$$

W przypadku składni reguł zgodnej z (2.14) zapis upraszcza się do  $(f(a_i, e) = v_{ai})$ .

Mówimy o częściowym dopasowaniu (ang. *partial matching*) obiektu  $e$  do części warunkowej reguły  $r$ , jeśli istnieje przynajmniej jeden atrybut  $a_j \in C_r$ , dla którego nie jest spełniony warunek:

$$(f(a_j, e) \propto \text{term}(a_j)) \quad (2.22)$$

Dopasowanie reguł decyzyjnych do opisu klasyfikowanego przykładu realizowane jest różnie w zależności od tego, czy reguły decyzyjne uporządkowane są w postaci listy decyzyjnej, czy tworzą nieuporządkowany zbiór reguł.

W przypadku listy decyzyjnej dokonuje się dopasowania obiektu do kolejnych reguł na liście [37, 185]. Pierwsza dopasowana reguła na liście wyznacza przydział obiektu  $e$  do klasy decyzyjnej. Nie przegląda się wtedy dalszych reguł. Ostatnią regułą na liście jest tzw. reguła domyślna (ang. *default rule*), stosowana wtedy, gdy żadna z poprzednich reguł nie dopasowała się do obiektu. Reguła domyślna najczęściej wskazuje klasę większościową w zbiorze uczącym, tj. najliczniejszą klasę w zbiorze uczącym. Innym rozwiązaniem jest wymuszenie sytuacji, gdy części wa-

runkowe reguł tworzą rozłączny podział przestrzeni przykładów, który prowadzi do jednoznacznych klasyfikacji, np. [129]. W przypadku klasyfikowania obiektu za pomocą nieuporządkowanego zbioru reguł występuje jedna z następujących sytuacji:

1. Część warunkowa dokładnie jednej reguły jest całkowicie dopasowana do obiektu, który jest zaklasyfikowany do klasy wskazywanej przez tę regułę.
2. Część warunkowa więcej niż jednej reguły jest całkowicie dopasowana do obiektu.
3. Część warunkowa żadnej reguły nie jest dopasowana do obiektu.

Sytuacja 1 jest jednoznaczna, natomiast sytuacje 2 oraz 3 rozwiązuje się różnymi sposobami, np. [11, 36, 37, 74, 110, 145, 221, 233, 253]. Poniżej krótko opiszemy dwie techniki, podane odpowiednio przez Grzymałę [74] oraz Stefanowskiego [231], gdyż są one stosowane w dalszej części rozprawy.

Pierwsza z tych technik [74] proponuje rozwiązanie obu sytuacji (2 i 3) na podstawie dodatkowych miar charakteryzujących regułę  $r$ . Grzymała wprowadza współczynnik siły reguły  $MR(r) = |[P \wedge Q]|$ . Jest to łączna liczba przykładów wspierających regułę. Rozważa także tzw. specyficzność reguły  $SR(r)$ , zdefiniowaną jako liczbę warunków elementarnych w regule.

W sytuacji 2 dla reguł, których części warunkowe są całkowicie dopasowane do obiektu, określa się poparcie klasy decyzyjnej  $X_k$ , zdefiniowane jako:

$$SUP(X_k) = \sum_{r \in R_k^-} MR(r) \cdot SR(r), \quad (2.23)$$

gdzie  $R_k^-$  oznacza reguły z  $R_k$  całkowicie dopasowane do obiektu  $e$ . Obiekt jest przydzielany do tej klasy decyzyjnej  $X_k$ , dla której poparcie  $SUP(X_k)$  osiąga największą wartość.

Rozwiązanie sytuacji 3 polega na zidentyfikowaniu reguł częściowo dopasowanych do obiektu. Dla przypadku częściowego dopasowania określa się dodatkową miarę stopnia dopasowania reguły do obiektu  $MF(r, e)$ . Jest ona równa stosunkowi liczby warunków reguły dokładnie dopasowanych do opisu obiektu do ogólnej liczby warunków tworzących część warunkową reguły. Przykład  $e$  jest klasyfikowany do tej klasy  $X_k$ , dla której wyrażenie (2.24) osiąga największą wartość:

$$SUPP(X_k) = \sum_{r \in R_k^{\approx}} MF(r, e) \cdot MR(r) \cdot SR(r), \quad (2.24)$$

gdzie  $R_k^{\approx}$  oznacza reguły z  $R_k$  częściowo dopasowane do obiektu  $e$ . Ocena eksperymentalna tego podejścia zawarta jest między innymi w [84].

Druga technika rozwiązywania sytuacji 2 i 3 została podana w [231]. Możliwe jest uwzględnienie użycia reguł niedyskryminujących (2.20). Dla sytuacji 2 niech  $R^-(e, k)$  oznacza te reguły całkowicie dopasowane do  $e$ , które pokrywają przykłady uczące z klasy  $X_k$ ; liczba takich przykładów oznaczana jest przez  $CR(r)$ .

Nowy obiekt  $e$  jest przydzielany do klasy  $X_k$ , dla której wyrażenie  $\sum_{r \in R^{\approx}(e,k)} CR(r)$  osiąga największą wartość. W sytuacji 3 proponuje się użycie zbioru reguł bliskich względem klasyfikowanego obiektu. Przez reguły bliskie rozumie się te reguły, których część warunkowa różni się jak najmniej od opisu klasyfikowanego obiektu w sensie miary odległości zdefiniowanej następująco:

$$DR(r, e) = \frac{1}{m} \cdot \left[ \sum_{a_i \in C_r} k_i d_i^p \right]^{1/p} \quad (2.25)$$

gdzie  $m = |C_r|$ ,  $k_i$  jest wagą przypisaną atrybutowi  $a_i$  – domyślnie równą 1,  $p$  jest wykładnikiem miary odległości:  $p = 1, 2, \dots$ ; a  $d_i$  jest częściową odległością pomiędzy wartością atrybutu  $a_i$  w opisie obiektu i w warunku elementarnym reguły  $v_{ir}$ . W przypadku atrybutów liczbowych  $d_i = \text{abs}(f(a_i, e) - v_{ir}) / \text{abs}(v_{i\max} - v_{i\min})$ , gdzie  $\text{abs}$  oznacza wartość bezwzględną różnicy, a  $v_{i\max}$  i  $v_{i\min}$  oznaczają minimalne i maksymalne wartości w dziedzinie atrybutu. Dla atrybutów nominalnych:  $d_i = 1$ , gdy występuje różnica pomiędzy obiektem a warunkiem reguły, i  $d_i = 0$  w przeciwnym razie (inne definicje odległości częściowej rozważa się w [40, 247]).

Obiekt  $e$  jest klasyfikowany do takiej klasy  $X_k$ , dla której poniższe wyrażenie osiąga największą wartość:

$$\sum_{r \in R^{\approx}(e,k)} (1 - DR(r, e)) \cdot CR^{\approx}(r), \quad (2.26)$$

gdzie  $R^{\approx}(e,k)$  oznacza te reguły częściowo dopasowane do  $e$ , które pokrywają przykłady uczące z klasy  $X_k$ ; liczba takich przykładów oznaczana jest przez  $CR^{\approx}(r)$ .

W pracach [216, 221, 232] rozważano także inną postać miary bliskości opartą na relacji podobieństwa. Zastosowanie jej do klasyfikowania z wykorzystaniem wiedzy dziedzinowej od użytkownika przedstawiono w [222]. Wykorzystanie obu miar bliskości badano eksperymentalnie dla wielu danych testowych [232].

## 2.4. Metody oceny klasyfikatorów regułowych

### 2.4.1. Miary oceny

Ocenę przydatności zbioru reguł do klasyfikowania nowych obiektów dokonuje się przez estymację błędu klasyfikowania lub trafności klasyfikowania w odniesieniu do zbioru przykładów testowych, których przynależność do klas jest określona. Zakłada się, że rozkład przykładów jest losowy i reprezentatywny [263].

Niech  $E^t$  będzie zbiorem przykładów testowych, a  $d_R(x)$  oznacza decyzje klasyfikatora regułowego dla  $x \in E^t$ . Wtedy  $E_{cor}^t = \{x \in E^t : d_R(x) = f(d, e)\}$  jest zbiorem przykładów poprawnie sklasyfikowanych, a  $E_{err}^t = \{x \in E^t : d_R(x) \neq f(d, e)\}$  jest zbiorem przykładów błędnie sklasyfikowanych. Ponadto,  $n_{cor} = |E_{cor}^t|$  oraz  $n_{err} = |E_{err}^t|$  są licznosciami tych zbiorów.

**Łączny błąd klasyfikowania** (ang. *overall error rate*) jest zdefiniowany jako:

$$\varepsilon_{ov} = \frac{n_{err}}{|E^t|} \quad (2.27)$$

**Miara trafności klasyfikowania** określona jest jako uzupełnienie do jedynki łącznego błędu, czyli:

$$\eta_{ov} = 1 - \varepsilon_{ov} = 1 - \frac{n_{err}}{|E^t|} = \frac{n_{cor}}{|E^t|} \quad (2.28)$$

Miary te alternatywnie wyraża się w procentach. Zamiast nazwy „trafność klasyfikowania” używa się także nazw „dokładność” (z ang. *classification accuracy*) lub „sprawność”. Im większa wartość  $\eta_{ov}$  (im mniejsza  $\varepsilon_{ov}$ ), tym skuteczniejszy jest klasyfikator.

Klasyfikator jest oceniany za pomocą definicji miar (2.27) lub (2.28) wtedy, gdy błędy niesłusznego zaklasyfikowania przykładu do innej klasy niż znana wartość atrybutu decyzyjnego, mają równe znaczenie. W niektórych zastosowaniach rozróżnienie pomiędzy różnymi błędnymi (niesłusznymi) zaklasyfikowaniami jest istotne. Przykładowo w medycynie zaliczenie chorego pacjenta do grupy zdrowych jest dużo bardziej niebezpieczne niż sytuacja odwrotna. Podobnie różne są koszty błędnych decyzji w analizie finansowej, np. przy ocenie kredytobiorców. Ma to także znaczenie dla analizy zbiorów danych o nierównomiernym rozkładzie przykładów pomiędzy klasami, gdy użytkownikowi zależy na poprawnym klasyfikowaniu przykładów z klasy mniejszościowej. W takich sytuacjach należy posługiwać się **macierzą pomylek** (ang. *confusion matrix*). Jest to macierz kwadratowa  $r \times r$ , gdzie wiersze odpowiadają poprawnym klasom decyzyjnym, a kolumny – decyzjom przewidywanym przez klasyfikator; na przecięciu wiersza  $i$  oraz kolumny  $j$  umieszcza się liczbę przykładów testowych  $n_{ij}$  należących oryginalnie do klasy  $i$ -tej, a zaliczonych do klasy  $j$ -tej. Dla danej tablicy dokładność klasyfikowania  $\eta_{ov} = \sum_{i=1}^r n_{ii} / |E^t|$ . Liczby w pozostałych polach  $n_{ij}$  ( $i \neq j$ ) mogą być wykorzystane do analizy niesłusznych zaklasyfikowań obiektów. Najczęściej proponuje się zastosowanie kosztu błędnej klasyfikacji (ang. *misclassification cost*)  $C(\varepsilon_{ov})$  zamiast łącznego błędu  $\varepsilon_{ov}$ . Niech  $c_{ij}$  będzie kosztem błędnego zaklasyfikowania do klasy  $j$ -tej przykładu należącego do klasy  $i$ -tej; koszt poprawnej klasyfikacji  $c_{ii}$  przyjmuje się równy 0. Koszt błędnej klasyfikacji  $C(\varepsilon_{ov})$  jest łączną sumą wszystkich błędnych klasyfikacji:

$$C(\varepsilon_{ov}) = \sum_{i=1}^r \sum_{j=1}^r n_{ij} \cdot c_{ij} \quad (2.29)$$

Inne propozycje miar oceny błędnych klasyfikacji można znaleźć np. w [35, 77, 150, 263].

W niektórych problemach uczenia się rozważa się klasyfikację binarną lub problem wieloklasowy sprowadza się do problemu binarnego – przykładowo w analizie danych medycznych częste jest wyróżnianie jednej szczególnie ważnej dla diagnostyki klasy decyzyjnej związanej z niebezpieczną chorobą. W takich sytuacjach działanie klasyfikatora przedstawia się w postaci macierzy  $2 \times 2$ .

**Tabela 2.1.** Macierz pomyłek dla klasyfikacji binarnej

Oryginalne klasy	Przewidywane klasy decyzyjne	
	Pozytywna	Negatywna
Pozytywna	$TP$	$FN$
Negatywna	$FP$	$TN$

Nazewnictwo jest inspirowane technikami oceny testów medycznych [162], przykładowo  $TP$  (ang. *true positive*) oznacza liczbę poprawnie sklasyfikowanych przykładów z wybranej klasy, a  $FN$  (ang. *false negative*) liczbę błędnie sklasyfikowanych przykładów z tej klasy. Dla takiej macierzy definiuje się dwie dodatkowe miary:

$$- \text{wrażliwość (ang. } sensitivity) = TP/(TP+FN), \quad (2.30)$$

$$- \text{specyficzność (ang. } specificity) = TN/(FP+TN). \quad (2.31)$$

Preferowane są jak najwyższe wartości tych miar.

Wnikliwszej analizy działania klasyfikatorów binarnych dokonuje się na podstawie analizy krzywej ROC (ang. *Receiver Operating Characteristic*) [87]; jej wykorzystanie dyskutowane jest także w [35, 77, 109, 180].

#### 2.4.2. Podstawowe techniki oceny eksperymentalnej

Dla realnych problemów uczenia się nie dysponuje się często zbiorem wszystkich możliwych przykładów. Dostępny jest zbiór przykładów o ograniczonych rozmiarach, który należy traktować jako próbę losową pobraną z populacji wszystkich możliwych przykładów. Taki ograniczony liczbowo zbiór przykładów musi być użyty do estymacji miar oceny klasyfikatora. Wykonuje się to na drodze eksperymentalnej. Dostępny zbiór przykładów należy podzielić na część uczącą  $E^u$ , na podstawie której buduje się klasyfikator, oraz testującą  $E^t$  służącą do estymacji wybranej miary – najczęściej (2.27) lub (2.28)<sup>7</sup>. Takie postępowanie nazywane jest paradygmatem *train-and-test* [263]. Podział zbioru przykładów na części uczącą i testującą jest losowy i realizuje się go w różny sposób w zależności od liczby dostępnych przypadków. Dla dużej liczby przykładów (więcej niż kilka tysięcy przypadków) wykonuje się jednokrotny podział (ang. *hold-out*). Dla mniejszej liczby

<sup>7</sup> W niektórych klasyfikatorach, których parametry dostraja się za pomocą oszacowania trafności klasyfikacji, zbiór  $E^u$  może być także dzielony wewnętrznie na dwie części: właściwy zbiór uczący i zbiór „walidujący” [108, 195].

przykładów stosuje się wielokrotne podziały i prowadzi się estymację dla każdego podziału. W [263] wyróżnia się następujące techniki podziału i estymacji błędu lub trafności klasyfikowania (zachowano oryginalne anglojęzyczne nazwy tych technik):

- *hold-out*,
- *random subsampling*,
- *k-fold cross validation*,
- *leaving-one-out*,
- *bootstrapping*.

Omówimy krótko technikę *k-fold cross validation* (w języku polskim można ją nazwać walidacją krzyżową) wykorzystywaną w dalszej części rozprawy. Opis pozostałych i zasady ich stosowania można znaleźć np. w [21, 148, 263]. Zbiór przykładów (powyżej 100) jest losowo podzielony na  $k$  podzbiorów  $U = E_1 \cup \dots \cup E_k$ . W  $i$ -tej iteracji ( $1 \leq i \leq k$ ), zbiór  $E^u = U \setminus E_i$  jest stosowany jako zbiór przykładów uczących, a sam zbiór  $E_i$  jako zbiór przykładów testowych. Trafność klasyfikowania jest wyliczana jako wartość średnia z trafności estymowanych w każdej iteracji  $\eta(E_i)$ , tzn. jest zdefiniowana jako:

$$\bar{\eta}_{ov} = \frac{1}{k} \sum_{i=1}^k \eta(E_i) \quad (2.32)$$

*Leaving-one-out* jest szczególnym przypadkiem, dla którego liczba iteracji jest równa liczbie przykładów, a w kolejnych iteracjach zbiór  $E_i$  tworzy pojedynczy przykład; podejście to jest stosowane dla małej liczby przykładów (mniej niż 100). W ogólności dobór parametru  $k$  powinien być dokonany w zależności od rozmiarów analizowanego zbioru danych. W [21, 195] podaje się, że wartość  $k = 10$  jest najbardziej zalecana.

Więcej informacji na temat estymacji miar oceny klasyfikatorów, dyskusji poprawności tych technik, zasad wykonywania eksperymentów porównywania różnych klasyfikatorów można znaleźć w [21, 148, 180, 195].

## 2.5. Dyskusja problemów związanych ze stosowaniem reguł decyzyjnych i teorii zbiorów przybliżonych w perspektywie klasyfikowania nowych obiektów

Dokonamy krótkiej dyskusji problemów związanych z użyciem reguł decyzyjnych pozyskanych metodami zarówno uczenia maszynowego, jak i teorii zbiorów przybliżonych w perspektywie klasyfikacyjnej dla realnych, dużych zbiorów danych.

W tej perspektywie celem jest najczęściej osiągnięcie jak największej trafności klasyfikowania  $\eta_{ov}$ . Dla badanego algorytmu zaleca się wykonywanie eksperymentów porównawczych z innymi algorytmami na danych testowych – pewnym stan-

dardem jest korzystanie z danych zgromadzonych w [14]. W pracach [148, 263] przedstawiono metodykę i przykłady takich studiów porównawczych. Rezultaty tych i innych badań wskazują, że nie ma jednego uniwersalnego algorytmu dla wszystkich danych. Zależy to zarówno od charakteru danych, jak i od założeń czy strategii używanych w algorytmie (tzw. indukcyjne ukierunkowanie [149]). W pracy [148] w opisie wyników eksperymentów porównawczych w projekcie Statlog czytamy, że „działanie (reguł) było praktycznie nierozróżnialne od drzew decyzyjnych, lecz oferowało lepsze dopasowanie do umysłu człowieka”. Podobnie w [46] podkreśla się, że metody regułowe w odróżnieniu od technik typu „czarna skrzynka” (np. funkcje dyskryminacyjne, sieci neuronowe) oferują jawną reprezentację wiedzy ułatwiającą zrozumienie i wyjaśnienie problemów decyzyjnych (w języku angielskim porównanie obu grup metod określono jako *data fit vs. mental fit*).

W odniesieniu do metod wykorzystujących teorię zbiorów przybliżonych w pracy [110] stwierdza się, że możliwe jest osiągnięcie trafności klasyfikowania porównywalnej ze standardowymi klasyfikatorami symbolicznymi wywodzącymi się z uczenia maszynowego. Dla metod opartych na wnioskowaniu boolowskim wyniki eksperymentalne omawiano w [11, 110, 153], dla algorytmu LEM2 i podobnych – w [74, 82, 233], skuteczność Probrough omówiono w [129], a TRANCE w [114].

W ocenie eksperymentalnej najczęściej używa się miary trafności klasyfikowania. W przypadku zbiorów danych o nierównomiernym rozkładzie przykładów pomiędzy klasami, a także w odniesieniu do klasyfikacji binarnej korzystniejsze jest stosowanie innych miar, takich jak: koszt, wrażliwość, specyficzność, omówionych w rozdziale 2.4.1. W szczególności w ostatnich latach stosuje się analizę krzywej ROC [180]. Praca [77] przedstawia interesującą propozycję modyfikacji klasyfikatora poprzez zamianę wsparcia reguł klasy mniejszościowej na podstawie analizy zmian na krzywej ROC.

Oprócz oceny trafności klasyfikowania rozważa się także inne kryteria związane z efektywnością uczenia się (czas, pamięć, prędkość uczenia się wraz ze zmieniającą się liczbą przykładów), odpornością na zmiany w danych czy przejrzystością reprezentacji wiedzy (porównaj [124]).

W pracy [35] podaje się następujące ograniczenia metod symbolicznych: tworzone klasyfikatory nie są z natury probabilistyczne, istnienie wielu identycznie opisanych obiektów nie wpływa dostatecznie znacząco na hipotezy, generowane klasyfikatory mogą być wrażliwe na małe zmiany w danych wejściowych, a ponadto poszukiwanie hipotez z bardzo dużych danych jest dość kosztowne.

W odniesieniu do drugiej z powyższych uwag krytycznych należy zauważyć, że tego typu klasy nierozróżnialnych obiektów można wykorzystać w teorii zbiorów przybliżonych, np. mogą być uwzględniane w trakcie budowania modelu przybliżonego [114]. Problem zwiększenia odporności modelu na zmiany czy „trudności” w danych może być w pewnym stopniu uwzględniany przez techniki redukcji zbiorów reguł (tzw. *rule truncation* podane w [145]) oraz za pomocą technik takich jak



redukty przybliżone lub dynamiczne czy model zmiennej precyzji w teorii zbiorów przybliżonych [11, 12, 274].

Problem zastosowania algorytmów indukcji reguł do analizy dużych zbiorów danych wiąże się ponadto z następującymi trudnościami:

- bardzo dużą liczbą obiektów,
- dużą liczbą atrybutów,
- nieistotnością części atrybutów dla klasyfikacji obiektów,
- wzajemną współzależnością atrybutów warunkowych,
- równoczesną obecnością atrybutów różnego typu,
- występowaniem nieznanymi wartości atrybutów.

Dwa ostatnie zagadnienia będą szczegółowo omawiane w rozdziałach 4 i 5. Rozwiązaniem problemu dużych zbiorów danych jest często ich redukcja. Może ona dotyczyć zarówno liczby obiektów, jak i atrybutów. W pierwszym przypadku często spotykanym rozwiązaniem jest losowe wybranie z oryginalnej tablicy próby o odpowiedniej liczności i uczenie klasyfikatora na jej podstawie [35, 262]. Wyróżnia się tutaj trzy sposoby tworzenia zbioru:

- pojedynczej próby,
- przyrostowego tworzenia zbioru uczącego,
- losowego podziału danych na podzbiory.

W drugim z powyższych sposobów tworzony klasyfikator jest systematycznie weryfikowany na części przykładów nie należących do wylosowanej próby. Jeśli jego działanie nie jest satysfakcjonujące, to zawartość próby uczącej jest modyfikowana z uwzględnieniem przykładów trudnych do nauczenia – takie rozwiązanie stosowane jest w technice okien (ang. *windowing*) w algorytmie C4.5 [185]. W trzecim podejściu zbiór danych jest dzielony na podzbiory (niekoniecznie rozłączne); z każdego nich uczy się klasyfikator. Pojedyncze klasyfikatory łączone są następnie w klasyfikator złożony [31, 95].

Jeszcze inne podejście do redukcji tablic decyzyjnych o dużych rozmiarach zaprezentowano w [152, 153], gdzie początkowa tablica decyzyjna podlega dekompozycji do zbioru „podtabel” o akceptowalnych rozmiarach. Dekompozycja jest wykonywana na podstawie znalezionych tzw. wzorców (ang. *template*); są to „dość mocno długie” wektory par (atrybut, wartość) wsparte przez wystarczająco dużą liczbę przykładów. Metody wnioskowania boolowskiego mogą być wykorzystane do generowania reguł decyzyjnych z podtabel. Przegląd różnych strategii poszukiwania wzorców zawarto w [153].

Problem redukcji liczby atrybutów rozważa się także na różne sposoby. Z uwagi na koszty obliczeniowe najszybsze są techniki usuwania atrybutów nieistotnych dla klasyfikacji (ang. *irrelevant*) w trakcie wstępnego przetwarzania danych. Istnieje wiele propozycji wywodzących się ze statystyki, np. testy  $\chi^2$  [142, 148], względ-

na entropia między atrybutem warunkowym a decyzyjnym, krokowe techniki pochodzące z analizy dyskryminacyjnej [262] i inne. Wiele z nich ocenia pojedyncze atrybuty. Ocena całych podzbiorów jest kosztowniejsza obliczeniowo; jednym z rozwiązań jest wybranie losowo (lub w inny sposób) rodziny wielu podzbiorów atrybutów i systematyczne przeszukanie ich w celu odnalezienia podzbioru prowadzącego do skutecznego klasyfikatora [35, 114]. W przypadku danych o mniejszych rozmiarach dobre rezultaty pozwala osiągnąć model *wrapper* [108]. Wyniki eksperymentalne wskazują, że selekcja atrybutów może prowadzić do poprawy trafności klasyfikacji różnych klasyfikatorów [101, 108]. Ponadto wskazuje się na dobre działanie klasyfikatorów regułowych zbudowanych z wykorzystaniem małego podzbioru atrybutów [4, 92, 114, 134]. Do selekcji atrybutów można także wykorzystać współpracę z ekspertami dziedzinowymi, którzy potrafią ukierunkować proces odkrywania wiedzy [126].

Jak wskazano w [148], dobrze zdefiniowany problem uczenia nadzorowanego powinien zawierać atrybuty warunkowe, które są wzajemnie niezależne. Badania nad wykrywaniem i eliminacją atrybutów zależnych prowadzi się zarówno w zakresie uczenia maszynowego [111], jak i teorii zbiorów przybliżonych [211].

### 3. Uwzględnianie różnych miar oceny reguł decyzyjnych w procesie ich indukcji

#### 3.1. Specyfika perspektywy opisu w odkrywaniu reguł decyzyjnych

Indukcja reguł decyzyjnych może być prowadzona zgodnie z dwoma perspektywami: predykcji klasyfikacji nowych obiektów oraz opisu. Przypomnijmy, że celem indukcji zorientowanej na klasyfikację jest zbudowanie z przykładów uczących zbioru reguł decyzyjnych, które będą użyte do klasyfikowania nowych obiektów. Celem indukcji zorientowanej na tworzenie opisu jest odkrywanie z danych zbiorów reguł decyzyjnych reprezentujących regularności (lub czasami anomalie czy wyjątki), które są interesujące, użyteczne i zrozumiałe dla różnego rodzaju użytkowników [50]. Wzorce opisowe są więc ukierunkowane bardziej na interpretację przez człowieka wiedzy pozyskiwanej z danych [114, 138].

Tradycyjne algorytmy uczenia maszynowego były tworzone w perspektywie klasyfikacyjnej (patrz dyskusja w rozdziale 2). Druga perspektywa indukcji reguł nie była dotychczas tak intensywnie badana, lecz rozwój dziedziny odkrywania wiedzy zwiększa zapotrzebowanie na poszukiwanie wzorców w tej perspektywie. W przeglądowej pracy [50, s. 12] pisze się, że „względna ważność predykcji i opisu dla poszczególnych zastosowań eksploracji danych może ulegać zmianie. Jednak w kontekście odkrywania wiedzy w bazach danych opis staje się ważniejszy niż predykcja. Jest to przeciwieństwem zastosowań uczenia maszynowego lub rozpoznawania wzorców (np. rozpoznawania mowy), gdzie predykcja jest często najważniejszym celem”. Zauważmy, że wzorce opisowe mogą obejmować także szerszą klasę reguł, np. reguły asocjacyjne [1]. Reguły asocjacyjne reprezentują istotne współzależności pomiędzy wartościami atrybutów, przy czym nie rozważa się pojęcia klas decyzyjnych; powiązania dotyczą wyłącznie atrybutów warunkowych. W tym rozdziale rozważamy jednak wyłącznie postać reguły (2.3), opisującą zależność pomiędzy wartościami atrybutów a klasą decyzyjną.

Wzorce klasyfikacyjne i opisowe oparte są na tym samym typie wiedzy, tj. zbiorze reguł decyzyjnych. Jednakże ze względu na różne cele obu perspektyw zarówno kryteria oceny reguł, jak i sposoby ich indukcji mogą być odmienne.

Najważniejsza różnica dotyczy kryteriów oceny reguł. W perspektywie klasyfikacyjnej ocena dotyczy całego zbioru reguł i jest oparta na jednym dobrze zdefiniowanym kryterium, takim jak trafność klasyfikowania (2.28) lub wrażliwość wybranej klasy decyzyjnej (2.30). Preferuje się wysoką skuteczność klasyfikowania, nawet kosztem obniżenia jego interpretowalności czy pominięcia w jego strukturze „ważnych” reguł. W perspektywie opisu każda pojedyncza reguła jest oceniana indywidualnie jako potencjalny reprezentant „interesującego” wzorca odnalezionego w danych. Należy podkreślić, że ocena ta jest trudniejsza i częściowo subiek-

tywna. W ogólności zależy od kontekstu zastosowania, zainteresowania i wiedzy użytkownika. Możliwe jest, że dwóch różnych użytkowników nie wskaże tych samych reguł jako interesujących. Pomimo tych trudności prowadzone są badania nad bardziej formalnymi miarami oceny reguł. Ma to szczególne znaczenie dla reguł asocjacyjnych, gdzie konieczne jest silne ograniczenie przestrzeni przeszukiwań kandydatów na reguły [106].

Niektórzy autorzy proponują definiować „interesujące” reguły częściowo na podstawie wymagań podawanych przez użytkownika [106, 133, 201], podczas gdy inni [53, 56, 102, 136, 174] proponują tzw. ilościowe miary oceny reguł. Miary ilościowe, nazywane przez niektórych autorów obiektywnymi, oceniają znaczenie pojedynczej reguły na podstawie jej związku ze zbiorem danych, z którego ją wyprowadzono. Przeglądy dotychczas zaproponowanych miar dostępne są np. w [5, 22, 270]. Interpretacja niektórych z tych miar w kontekście teorii zbiorów przybliżonych i reguł decyzyjnych podana jest także przez Pawlaka w [166, 167].

Drugą zasadniczą różnicą pomiędzy obiema perspektywami dotyczy samego sposobu poszukiwania reguł. Bezpośrednie stosowanie algorytmów wypracowanych dla perspektywy klasyfikacyjnej może prowadzić do nie satysfakcjonujących rezultatów w perspektywie opisu. Pomimo że podkreśla się „naturalną czytelność” regułowej reprezentacji wiedzy [141, 143] typowe algorytmy indukcji reguł klasyfikacyjnych faworyzują osiąganie wysokiej trafności klasyfikowania nowych (lub testowych) obiektów. W rezultacie wygenerowany zbiór reguł może zawierać specyficzne, mało interesujące dla użytkownika reguły. Ponadto taki zbiór może zawierać tylko ograniczoną część interesujących reguł. Jest to konsekwencją stosowanego heurystycznego schematu generowania kolejnych pokryć (patrz rozdział 2.2.1), gdyż część ogólnych i silnych reguł może pozostać ukryta w danych. Z drugiej strony, użycie metod indukcji wyczerpującego zbioru reguł (np. [202, 203]) może prowadzić do zbiorów reguł, które są zbyt skomplikowane. Zbiór składający się z setek czy tysięcy reguł jest niezrozumiały dla człowieka. Ponadto większość reguł opisuje najczęściej tylko pojedyncze obiekty i nie reprezentuje ogólnych wzorców w danych. Możliwe jest wprowadzenie filtrowania takich zbiorów w celu selekcji najbardziej interesujących reguł [106, 109], lecz wiąże się to z dużymi kosztami obliczeniowymi pozyskania najpierw wyczerpującego zbioru reguł.

Perspektywa opisu w odkrywaniu wiedzy wymaga nowych algorytmów indukcji reguł, które mogą w większym stopniu uwzględniać wymagania użytkownika wobec wybranych ilościowych miar oceny reguł. W tej perspektywie powinno się poszukiwać ogólnych, wiarygodnych i względnie prostych zbiorów reguł, gdyż ich interpretacja przez człowieka jest łatwiejsza niż złożonych zbiorów reguł. W dalszych rozdziałach takie cechy zbioru reguł będziemy wiązać z następującymi kryteriami: wsparciem reguły przez przykłady uczące, siłą zależności części decyzyjnej reguły od części warunkowej, liczbą użytych warunków elementarnych oraz całkowitej liczby reguł.

Zauważmy, że równoczesne uwzględnianie kilku kryteriów i określanie ich wartości satysfakcjonujących użytkownika ma charakter wielokryterialny i jest

trudniejsze niż w przypadku budowy klasyfikatorów. Ponadto osiągnięcie zadowalających użytkownika wartości wybranych kryteriów opisowej oceny reguł może być częściowo konfliktowe z osiągnięciem wysokiej trafności klasyfikacyjnej zbioru odkrytych reguł. Wynika to z różnych celów obu perspektyw. Z drugiej strony interesujące jest badanie stopnia tego konfliktu. Eksperymentalna analiza tego problemu jest pierwszym celem niniejszego rozdziału.

Pozytywna odpowiedź na pytanie, w jakim stopniu reguły o dobrych właściwościach opisowych są przydatne także do klasyfikowania, prowadzi do drugiego celu, tj. do zaprezentowania metody budowy prostego zbioru reguł (w sensie liczby reguł, ich ogólności i wsparcia) zapewniającego równocześnie akceptowalną trafność klasyfikacji. Możliwość zbudowania takiego zbioru reguł ma znaczenie także dla perspektywy klasyfikacyjnej, gdyż zwiększa uzasadnienie i zrozumienie decyzji klasyfikacyjnych dla nowych przykładów.

Realizacja dwóch pierwszych celów związana jest z użyciem nowego algorytmu indukcji reguł uwzględniającego oczekiwania wobec satysfakcjonujących wartości miar opisowych. Wprowadza się oryginalną propozycję algorytmu EXPLORE [147, 245]. W części problemów odkrywania wiedzy potrzebne może być także uwzględnianie wymagań użytkownika co do preferowanej składni reguł. W ostatnim podrozdziale przedstawia się modyfikacje algorytmu EXPLORE dokonane w celu uwzględnienia w trakcie indukcji reguł zarówno ilościowych miar oceny, jak i wymogów użytkownika co do składni reguł.

### 3.2. Opisowe miary oceny reguł

Miary oceny używane w dalej dyskutowanej perspektywie opisu mogą być oprócz tego przydatne także do [5, 22]:

- ograniczenia liczby potencjalnych kandydatów w trakcie przeszukiwania danych, tak aby wygenerować tylko najbardziej „użyteczne” reguły,
- wybrania najlepszych reguł z większego zbioru,
- wykorzystania w systemach indukcji konstruktywnej do modyfikacji reprezentacji przykładów uczących,
- rozstrzygnięcia konfliktów podczas klasyfikowania nowych przykładów.

W celu przedstawienia ilościowych miar oceny reguł używa się interpretacji reguły w kategoriach teorii zbiorów. Wiąże ona regułę ze zbiorem obiektów, na podstawie którego została wygenerowana [270]. Niech  $U$  oznacza zbiór tych obiektów, a reguła będzie reprezentowana w postaci (2.4), tj. *jeżeli*  $P$  *to*  $Q$ . Reguła taka generuje dwa następujące podziały zbioru  $U$ :

$$U = U_P \cup U_{\neg P} = [P] \cup (U - [P]) \quad (3.1)$$

$$U = U_Q \cup U_{\neg Q} = [Q] \cup (U - [Q]) \quad (3.2)$$

gdzie  $U_P$  oznacza zbiór obiektów spełniających przesłankę reguły  $P$ , a  $U_{\neg P}$  oznacza zbiór obiektów nie spełniających  $P$ ,  $U_Q$  jest zbiorem przykładów pozytywnych  $Q$ , a  $U_{\neg Q}$  jest zbiorem przykładów negatywnych. Większość ilościowych miar oceny reguł wyprowadza się na podstawie tablicy kontyngencji  $2 \times 2$  zawierającej informacje o liczebności odpowiednich zbiorów.

**Tabela 3.1.** Tablica kontyngencji dla reguły *jeżeli Q to P*

	$Q$	$\neg Q$	Suma
$P$	$n_{PQ}$	$n_{P\neg Q}$	$n_P$
$\neg P$	$n_{\neg PQ}$	$n_{\neg P\neg Q}$	$n_{\neg P}$
Suma	$n_Q$	$n_{\neg Q}$	$n$

W tabeli 3.1  $n_{PQ}$  oznacza liczbę obiektów spełniających zarówno  $Q$ , jak i  $P$ ,  $n_{P\neg Q}$  liczbę obiektów spełniających  $P$ , lecz nie spełniających  $Q$  itd.;  $n_P = n_{PQ} + n_{P\neg Q}$  oznacza liczbę obiektów pokrytych przez regułę ( $n_P = |U_P|$ );  $n_Q = n_{PQ} + n_{\neg PQ}$  oznacza liczbę obiektów spełniających  $Q$  ( $n_Q = |U_Q|$ ) a  $n = |U| = n_P + n_{\neg P} = n_Q + n_{\neg Q}$  jest liczebnością wszystkich obiektów. Tablica może być przedstawiana także w postaci zawierającej częstości (względem  $|U|$ ).

Zgodnie z klasyfikacją podaną w [270] podstawowe ilościowe miary oceny reguł definiowane na podstawie tablicy kontyngencji można podzielić na trzy podstawowe kategorie:

- ogólności,
- jednostronnej implikacji,
- dwustronnej implikacji.

Omówimy krótko najczęściej stosowane miary w tych kategoriach.

**Ogólność** części warunkowej  $P$  jest definiowana jako:

$$G(P) = \frac{n_P}{n} \quad (3.3)$$

Wyrażenie  $P$  jest bardziej ogólne, jeśli pokrywa więcej obiektów. Jeżeli  $G(P) = \alpha$  to znaczy, że  $(100 \cdot \alpha)\%$  obiektów w  $U$  spełnia  $P$ . Oczywiście,  $0 \leq G(P) \leq 1$ .

Inną miarą w tej kategorii jest **wsparcie reguły** (ang. *support*) zdefiniowane jako:

$$G(P \wedge Q) = \frac{n_{PQ}}{n} \quad (3.4)$$

Jest ona często używana w ocenie odkrytych reguł [1, 234, 276].

Miary jednostronnej implikacji charakteryzują stopień, w jakim prawdziwość  $P$  implikuje prawdziwość  $Q$ . Według [270] obejmują dwie podstawowe miary: bezwzględne wsparcie reguły oraz zmianę bezwzględnego wsparcia.

**Bezwzględne wsparcie** (ang. *absolute support*) konkluzji reguły  $Q$  pod warunkiem prawdziwości przesłanki  $P$  definiuje się jako:

$$AS(Q|P) = \frac{n_{PQ}}{n_P} \quad (3.5)$$

Miara ma zakres zmienności  $0 \leq AS(Q|P) \leq 1$  i określa stopień, w jakim  $P$  implikuje  $Q$ . Można interpretować ją jako prawdopodobieństwo warunkowe, że losowo wybrany obiekt, który spełnia wyrażenie  $P$ , spełnia również  $Q$ . W teorii mnogości jest to stopień, w jakim  $U_P$  jest zawarte w  $U_Q$ .

**Zmiana bezwzględnego wsparcia** (ang. *change of support*)  $Q$  przez  $P$  zdefiniowana jest jako:

$$CS(Q|P) = AS(Q|P) - G(Q) \quad (3.6)$$

Wielkość ta przyjmuje wartości od  $-1$  do  $1$ .

Miara (3.5) jest używana w eksploracji danych, np. dla indukcji reguł asocjacyjnych, jako wiarygodność (ang. *confidence*). Pawlak pisze w [167], że miara (3.5) po raz pierwszy została wprowadzona przez Łukasiewicza jako współczynnik pewności implikacji logicznej [135]. Przegląd literatury wskazuje także, że jest ona często wykorzystywana przez innych autorów, choć nadają jej inne nazwy, np. dokładność reguły (ang. *accuracy*) [94], stopień dyskryminacji [147, 234], *true positive rate* [35, 270].

Użycie miary  $CS(Q|P)$  jest dyskutowane w [174]. Ma ona także znaczenie w analizie danych o nierównomiernym rozkładzie przykładów w klasach (patrz także [112]).

Inną miarą jest **względne pokrycie reguły** (ang. *coverage*) zdefiniowane jako:

$$AS(P|Q) = \frac{n_{PQ}}{n_Q} \quad (3.7)$$

Reguła ma wysokie pokrycie, jeśli obiekty spełniające konkluzję reguły  $Q$  równocześnie spełniają wyrażenie  $P$ . Może być także interpretowana w kategoriach prawdopodobieństwa warunkowego [167]. Miara ta nazywana jest także siłą względną reguły. Jest ona używana przez wielu autorów, np. Michalski wprowadził ją w [141], także Tsumoto używał jej do oceny reguł w [257] oraz Pawlak omawiał ją w kontekście badania związku reguł z twierdzeniem Bayesa [167].

**Miary dwustronnej implikacji** (ang. *double implication* lub *mutual support*) określają siłę podwójnej implikacji  $P \leftrightarrow Q$ , tzn. stopień, w jakim  $P$  powoduje i wyłączenie powoduje  $Q$ . W [270] definiuje się przykład takiej miary jako:

$$MS(P,Q) = \frac{n_{PQ}}{n_{PQ} + n_{P-Q} + n_{-PQ}} \quad (3.8)$$

Inną miarą w tej kategorii jest stopień niezależności  $IND(P,Q) = AS(Q|P) / G(Q)$ . Szersza dyskusja tych miar zawarta jest w [270].

Przegląd literatury wskazuje, że oprócz różnych wariantów powyższych miar (nazywać je będziemy pojedynczymi) stosuje się także **miary złożone**. Miary złożone agregują w różny sposób pojedyncze miary. Wśród nich wyróżnia się tzw. miary empiryczne i miary oparte na podstawach teoretycznych. Przykłady miar empirycznych to propozycje Michalskiego [141], Torgo [256] i Brazdila [19]. Często stosowana miara Michalskiego jest zdefiniowana jako  $\mu \cdot AS(Q|P) + (1-\mu) \cdot AS(P|Q)$ , gdzie współczynnik  $\mu$  jest określany w zależności od kontekstu zastosowania. Miary teoretyczne dzieli się na miary związku (pomiędzy klasyfikacjami kolumn i wierszy w tablicy kontyngencji) obejmujące miarę Pearsona,  $G^2$  [229] oraz miary uzgodnienia dotyczące głównej przekątnej tablicy kontyngencji, np. formuły Cohena i Colemana. Inne miary teoretyczne to miary logicznej konieczności  $Q_{LS}$  oraz wywodzące się z teorii informacji. Przegląd miar teoretycznych zawarty jest w [5, 270]. W literaturze twierdzi się że formuły empiryczne, pomimo że nie mają silnych podstaw teoretycznych, są bardziej intuicyjne i łatwiejsze do interpretacji niż miary teoretyczne [5]. Ponadto w badaniach eksperymentalnych działają równie skutecznie. Szerszy przegląd propozycji różnych miar, a także ich porównania, przedstawiono w [5, 21, 270].

### 3.3. Poszukiwanie zbioru reguł decyzyjnych z uwzględnieniem opisowych i klasyfikacyjnych miar oceny

Założmy, że jesteśmy zainteresowani poszukiwaniem ogólnych i względnie prostych zbiorów reguł zawierających wiarygodne i silne reguły. Przy takich założeniach spośród wielu miar oceny reguł szczególną uwagę poświęcamy następującym:

- pokrycie reguły  $AS(P|Q)$  (reguła powinna pokrywać wystarczająco dużą liczbę przykładów uczących),
- dokładność reguły  $AS(Q|P)$  (wiarygodność zależności prezentowanej przez regułę oznacza, że reguła powinna pokrywać jak najmniejszą liczbę negatywnych przykładów),
- prostota reguły lub długość reguły  $DI(P)$ , oceniana jako liczba użytych warunków elementarnych (preferowane są proste reguły),
- całkowita liczba reguł  $N(R)$  (mniejsza liczba reguł łatwiej poddaje się analizie).

Jeśli równocześnie rozpatrujemy perspektywę klasyfikacyjną, uwzględniamy także kryterium trafności klasyfikowania  $\eta_{ov}$ , przy czym nie ma ono tutaj charakte-



ru dominującego. Uważamy, że zbiór reguł powinien wtedy prowadzić do trafności porównywalnej z trafnością standardowych klasyfikatorów regułowych.

Przypomnijmy, że algorytmy typowo stosowane w uczeniu maszynowym dla perspektywy wyłącznie klasyfikacyjnej nie pozwalają na odnalezienie z danych wszystkich reguł charakteryzujących się satysfakcjonującymi wartościami powyższych kryteriów. Dyskusję ograniczeń tych algorytmów przeprowadzono w rozdziale 3.1.

Oznacza to konieczność użycia innych algorytmów, które powinny umożliwić odkrycie najlepiej wszystkich reguł satysfakcjonujących wybrane kryteria; w rozważanym przypadku oczekuje się przynajmniej satysfakcjonującego wsparcia reguły, jej dokładności i prostoty. Jest to związane z miarami  $AS(P|Q)$ ,  $AS(Q|P)$ ,  $DI(P)$ ,  $N(R)$  oraz  $\eta_{ov}$ . Może to być osiągnięte przez specjalne sposoby eksploracji przestrzeni potencjalnych reguł. Powinno się rozważać algorytmy, które stopniowo generują reguły od prostszych jednoelementowych do bardziej złożonych, stosując przy tym techniki redukcji przestrzeni przeszukiwań oraz warunki zatrzymania, które umożliwiają jak najszybsze odrzucenie niepotrzebnych kandydatów na reguły.

Zasady powyższe dla miar wsparcia reguły  $G(P \wedge Q)$  i jej dokładności  $AS(Q|P)$  są szczególnie efektywnie zrealizowane w algorytmie Agrawala poszukiwania reguł asocjacyjnych (zwłaszcza w fazie generowania silnych wzorców) [1]. Modyfikacje tego algorytmu mające na celu poszukiwanie silnych reguł decyzyjnych (o postaci zgodnej w formułami 2.14) w kontekście teorii zbiorów przybliżonych były przedstawione w [121, 130]. Także autor wspólnie z Bobinem [16] zaproponował inną modyfikację algorytmu Agrawala służącą do poszukiwania reguł zgodnych z formułą (2.4). Ponadto Nguyen i Ślęzak badali w [158] związki pomiędzy poszukiwaniem reguł asocjacyjnych a pojęciami reduktów różnego typu w teorii zbiorów przybliżonych. W szczególności pokazali, jak zastosować redukt przybliżone do generowania reguł asocjacyjnych z wcześniej znalezionych silnych wzorców. Jeszcze inne spojrzenie na podobny problem intensywnego przeszukiwania przestrzeni hipotez w celu tworzenia zbioru jednorodnych prostych reguł o dobrych właściwościach klasyfikujących zaprezentowano w algorytmie BRUTE [189, 197]. Ponadto specjalną technikę przeszukiwania wszcz wykorzystano też w systemie MONET do równoległego poszukiwania wzorców charakterystycznych w dużych bazach danych [91]. W dalszych rozważaniach używać się będzie jeszcze innego podejścia realizującego silne ograniczanie przestrzeni kandydatów, tj. własnego algorytmu EXPLORE wprowadzonego przez autora i Vanderpootena w połowie lat dziewięćdziesiątych [147, 245].

### 3.3.1. Algorytm EXPLORE

Algorytm ma na celu wygenerowanie z danych wszystkich reguł, których wartości miar oceny pokrycia  $AS(P|Q)$ , długości części warunkowej  $DI(P)$ , dokładności  $AS(Q|P)$  muszą spełniać narzucone wymogi. Ponadto można wprowadzić dodatkowe ograniczenia co do składni części warunkowych, co jest przedstawione

w rozdziale 3.4. Uwzględnianie niespójnych przykładów jest możliwe z wykorzystaniem teorii zbiorów przybliżonych [234] lub przez odpowiednie dobranie wartości progu wymagań do minimalnej dokładności pojedynczej reguły.

Przeszukiwanie przestrzeni potencjalnych kandydatów na reguły wykonuje się za pomocą procedury, która jest powtarzana iteracyjnie dla każdej klasy decyzyjnej, lub jej przybliżenia. Zasadniczą część algorytmu oparta jest na generowaniu kandydatów o rosnącym rozmiarze, zaczynając od pojedynczych warunków elementarnych. Rozbudowa pojedynczego kandydata-koniunkcji jest zatrzymywana, gdy odnajdzie się regułę satysfakcjonującą wybrane wymogi co do wartości miary oceny lub spełni się tzw. warunek zatrzymania  $SC$ , który odzwierciedla niemożność osiągnięcia zadowalających wartości miar opisowych reguły. Zapis algorytmu przedstawiony jest poniżej. Stosuje się oznaczenia wprowadzone w rozdziale 2.1. W przypadku użycia miar oceny reguł ocenia się kandydatów na ich część warunkową, dlatego w ich notacji używa się symbolu kandydata  $P$  zamiast reguły  $r$ .

**procedure** EXPLORE

(**input**  $LS$ : lista warunków elementarnych;  $SC$ : warunki zatrzymania;

**output**  $R$ : zbiór reguł);

**begin**

$R := \emptyset$ ;

$GoodCandidates(LS, R)$ ;  $\{LS - \text{uporządkowana lista warunków } w_1, w_2, \dots, w_n\}$

$Kl := LS$ ;  $\{\text{stwórz kolejkę kandydatów} - Kl\}$

**while**  $Kl \neq \emptyset$  **do**

**begin**

wybierz pierwszą koniunkcję  $P$  z  $Kl$  ;

$Kl := Kl \setminus \{P\}$ ;  $\{\text{usuń koniunkcję z kolejki } Kl\}$

$Extend(P, LC)$ ;  $\{LC - \text{lista rozszerzonych koniunkcji}\}$

$GoodCandidates(LC, R)$ ;

$Kl := Kl \cup LC$ ;  $\{\text{umieść koniunkcje z } LC \text{ na końcu kolejki } Kl\}$

**end**

**end.**

**procedure**  $Extend(C$  : koniunkcja, **var**  $L$  : lista kandydatów);

{Procedura umieszcza na liście  $L$  rozszerzenia koniunkcji  $C$ ,

które są możliwymi kandydatami na reguły}

**begin**

Niech  $k$  będzie długością  $C$  i  $h$  niech będzie najwyższym indeksem warunków elementarnego występującego w  $C$ ;

$L := \{C \wedge c_{h+i} \text{ gdzie } c_{h+i} \in LS, \text{ taki że wszystkie } k \text{ podzbiory } C \wedge c_{h+i} \text{ o rozmiarze } k \text{ i uwzględniające } c_{h+i} \text{ także należą do } Kl, i = 1, \dots, n-h\}$

**end;**

**procedure**  $GoodCandidates$  (**var**  $L$  : lista koniunkcji będących kandydatami,

**var**  $R$  : zbiór reguł);

{Procedura usuwa z listy  $L$ :

- wszystkie koniunkcje, które mogą stać się regułami ze względu na warunki zatrzymania  $SC$ ,
- koniunkcje spełniające warunek utworzenia reguły, które umieszcza się w  $R$

```

begin
  for  $P \in L$  do
    begin
      if  $P$  spełnia  $SC$  then  $L := L \setminus \{P\}$ ;
      else
        if  $AS(Q|P) \geq \alpha$  then {jeśli spełniony jest warunek akceptacji kandydata na regułę,}
          begin { $\alpha = 1$  dla reguł pewnych}
            utwórz regułę  $r$  na podstawie  $P$ ;
            if  $r$  jest minimalna then  $R := R \cup \{P\}$ ;
             $L := L \setminus \{P\}$ ;
          end
        end
      end
    end
  end;

```

Początkowa lista warunków elementarnych  $LS$  jest tworzona przed uruchomieniem procedury. Każdy warunek  $w_i$  musi pokrywać choć jeden przykład pozytywny. Początkowa lista warunków poddawana jest redukcji, aby odrzucić te warunki, które prowadzą do zadowalających wartości miar oceny reguł lub spełniają warunki zatrzymania  $SC$  (patrz procedura *GoodCandidates*). Warunki pozostałe na liście  $LS$  są sortowane leksykograficznie i następnie łączone ze sobą, aby utworzyć warunki złożone, które są kandydatami o rozmiarze większym o jeden. Jest to wykonywane w procedurze *Extend*, gdzie w iteracji  $k$ -tej tworzy się warunki złożone o rozmiarze  $k+1$  przez rozszerzanie warunków ze struktury kolejki  $KI$  za pomocą kolejnych warunków elementarnych z listy  $LS$ , przestrzegając ograniczeń na „dobre” złożenia. Powstali kandydaci oceniani są znowu za pomocą procedury *GoodCandidates*. Dokładniejszy opis algorytmu oraz dyskusja technik przyspieszających jego implementacje dostępna jest w [147, 246].

Warunki zatrzymania  $SC$  są zdefiniowane zgodnie z wymaganiami sprecyzowanymi wobec pokrycia reguły  $AS(P|Q)$ , jej długości  $DI(P)$ , możliwej liczby reguł  $N(R)$  czy innych miar. Wymóg wobec minimalnej dokładności reguły  $AS(Q|P)$  może być wyrażony za pomocą progu rozróżnialności  $\alpha$ , który musi spełnić kandydat na regułę, tj.  $AS(Q|P) \geq \alpha$ .

W dalszych eksperymentach obliczeniowych rozważamy wyłącznie wymóg wobec minimalnego pokrycia reguły wyrażony za pomocą progu  $l$  – jest to minimalny procent przykładów pozytywnych, który musi pokryć tworzona reguła. Odpowiadający temu warunek zatrzymania jest wyrażony jako:

$$SC: AS(P|Q) < l \quad (3.9)$$

W eksperymentach będzie się także poszukiwać reguł pewnych, tzn. pokrywających wyłącznie przykłady pozytywne, tj.  $AS(Q|P) = 1$ . Dydaktyczne przykłady ilustrujące użycie tego algorytmu są przedstawione w [147, 234].

Ocena złożoności obliczeniowej algorytmu wiąże się z uwzględnieniem wykorzystania warunków zatrzymania  $SC$ . Niech podstawową operacją będzie sprawdzenie, jakie obiekty pokrywa koniunkcja  $P$ . W ogólności funkcja złożoności jest rzędu  $O(NF)$ , gdzie  $N = n \cdot m$  jest rozmiarem danych wejściowych,  $F$  jest liczbą koniunkcji zbiorów warunków elementarnych będących kandydatami do sprawdzenia. Jeżeli użytkownik wykorzystuje parametry  $SC$ , dostosowując je do zbioru danych, to powinny one silnie ograniczać wielkość  $F$ . Natomiast jeśli użytkownik zrezygnuje z ich stosowania, co odpowiada opcji szukania wszystkich reguł, to wielkość  $F$  może być zbyt duża. Zauważmy, że w najgorszym przypadku szukania wszystkich reguł, gdy reguły pokrywają pojedyncze obiekty i wykorzystują wszystkie atrybuty w budowie warunków elementarnych, liczba potencjalnych kandydatów rośnie wykładniczo. Jest to wielkość rzędu  $\sum_{k=1}^g \binom{m}{k} \cdot W^k$ , gdzie  $W$

jest średnią liczbą wartości w dziedzinie atrybutu,  $m$  – liczbą atrybutów, a  $g$  – maksymalną wielkością zbioru kandydującego. Należy jednak uznać, że wielkość ta jest oszacowana bardzo pesymistycznie, gdyż dla rzeczywistych danych część połączeń warunków elementarnych często nie pokrywa żadnych przykładów pozytywnych, więc nie będą one mogły być rozwijane przez funkcję *Extend*.

Zauważmy, że w wyniku działania algorytmu pewne przykłady uczące mogą pozostać niepokryte wygenerowanymi regułami. Nie jest to wadą, gdyż w ten sposób można wskazać przykłady, które są trudne do opisanie. Ich dalsza analiza może być interesująca, gdyż pozwoli sprawdzić, czy są efektem pewnego „zaszumienia” lub czy mogą być zinterpretowane jako możliwe nietypowe przypadki lub wyjątki. Podejście wykorzystujące taką interpretację będzie omawiane w rozdziale 7.3.

### 3.3.2. Eksperymenty obliczeniowe

Studium eksperymentalne wykonano w dwóch głównych celach:

1. Zbadanie zakresu zmian wartości wybranych miar oceny reguł ( $AS(P|Q)$ ,  $DI(P)$ ,  $N(R)$  oraz  $\eta_{ov}$ ) i stopnia konfliktu pomiędzy miarami ważnymi dla perspektywy opisu i miarą ważną dla perspektywy klasyfikacyjnej. Odbywać się to będzie za pomocą systematycznych zmian parametrów algorytmu EXPLORE (progu  $l$ ). W szczególności jesteśmy zainteresowani znalezieniem takich wartości parametrów, które mogą prowadzić do akceptowalnego kompromisu pomiędzy wartościami wybranych miar oceny reguł.
2. Sprawdzenie, w jakim stopniu reguły o dobrych właściwościach opisowych są także efektywne w klasyfikowaniu. Dodatkowym celem może być zaprezentowanie sposobu budowy ograniczonego liczbowo zbioru reguł pokrywających wystarczającą liczbę przykładów oraz zapewniającego równocześnie akceptowalną trafność klasyfikowania porównywalną ze standardowymi algorytmami.

Zauważmy, że powyższą motywację do poszukiwania ograniczonych zbiorów prostych reguł o dobrych właściwościach klasyfikujących można interpretować jako zgodną z zasadą minimalności długości [34, 149]. Przypomnijmy, że w myśl tej zasady przy podejmowaniu decyzji o wyborze hipotezy z ograniczonego zbioru różnych hipotez należy wybrać tę hipotezę, która ma minimalną długość łącznego kodu. Innymi słowy, wybiera się hipotezę (tutaj rozumianą jako zbiór reguł), która minimalizuje zarówno złożoność hipotezy (tzn. długość kodu odpowiadającego reprezentacji wiedzy), jak i błąd klasyfikowania.

W przypadku oceny skuteczności klasyfikacyjnej dokona się porównania z algorytmem LEM2 [73], który jest popularnym algorytmem tworzenia minimalnego zbioru reguł dogodnych dla celów klasyfikacyjnych metodą kolejnych pokryć. Ponadto dokona się także porównania z podejściem tworzenia wyczerpującego zbioru wszystkich reguł decyzyjnych (porównaj rozdział 2.2).

Algorytm EXPLORE był parametryzowany wyłącznie za pomocą progu  $l$  wyrażającego minimalną akceptowalną wartość względnego pokrycia reguły  $AS(P|Q)$ . Dokładność reguły  $AS(Q|P)$  nie była parametryzowana, tzn. jest równa 1 (reguły pewne według teorii zbiorów przybliżonych), w celu zachowania zgodności z porównywanymi algorytmami i utrzymania prostoty eksperymentu (rezultaty eksperymentów ze zmianą parametru minimalnej akceptowalnej dokładności reguły są dostępne w [236]). Z podobnych przyczyn nie wykorzystywano parametrów odnoszących się do długości reguły czy liczby reguł. Wybór parametru pokrycia reguły umożliwia silne zróżnicowanie reguł. Ponadto inne badania eksperymentalne wskazują, że opierając się na względnie prostych zbiorach reguł pokrywających odpowiednio dużo przykładów możliwe jest zbudowanie skutecznych klasyfikatorów [92, 114]. Składnia tworzonych warunków elementarnych została ograniczona do postaci (2.14) ze względu na porównywane algorytmy<sup>8</sup>.

Zbiór reguł decyzyjnych oceniano za pomocą następujących miar:

- liczby reguł,
- średniego pokrycia reguły (wyrażonego w bezwzględnej liczbie obiektów),
- średniej długości reguły,
- trafności klasyfikowania.

Wartość ostatniej miary oceniano za pomocą techniki 10-krotnej walidacji krzyżowej. W przypadku niejednoznaczności klasyfikowania obiektów testowych użyto metody odległościowej omówionej w rozdziale 2.4.1.

Poniżej prezentujemy wyniki eksperymentu na czterech różnych zbiorach danych pochodzących z [14]. Szersze eksperymenty na innych danych omawia się także w [234]. Zbiór *Iris* oryginalnie zdefiniowany na atrybutach ilościowych został zdyskretyzowany za pomocą metody [48].

---

<sup>8</sup> Poszukiwanie reguł z warunkami elementarnymi o ogólniejszej składni (2.4) za pomocą algorytmu EXPLORE rozważano w [236].

**Tabela 3.2.** Charakterystyka zbiorów reguł otrzymanych za pomocą algorytmu EXPLORE przy zmianach progu minimalnego pokrycia  $l$ 

Zbiór danych	Warunek zatrzymania SC: – $l$ próg minimalnego pokrycia reguły	Liczba reguł $N(R)$	Średnie pokrycie reguły $n_{PQ}$ [# obiektów]	Średnia długość reguły $DI(P)$ [# warunków]	Trafność klasyfikowania $\eta_{ov}$ [%]
<i>Iris</i>	wszystkie reg.	80	6,03	2,1	92,67
	5%	35	12,23	1,89	92,67
	10%	22	17,27	1,86	92
	15%	20	18,4	1,85	90
	20%	15	21,6	1,8	83,33
	25%	14	22,36	1,79	78,67
	30%	6	33,83	1,83	60,67
	LEM2	23	11	1,91	95,33
<i>Tic-tac-toe</i>	wszystkie reg.	2858	4,27	4,63	91,35
	5%	16	60,25	3	97,19
	10%	16	60,25	3	97,19
	15%	2	50	3	15,24
	20%	0	–	–	–
	30%	0	–	–	–
	LEM2	24	40,83	3,67	98,96
<i>Voting</i>	wszystkie reg.	1502	10,61	4,723	95,87
	5%	231	45,86	3,6	94,51
	10%	138	66,96	3,3	94,5
	15%	104	79,61	3,1	93,8
	20%	82	89,87	3,1	94
	25%	67	96,99	3,1	93,32
	30%	50	104,7	3,1	93,31
	40%	21	133	2,76	80,23
LEM2	26	43,77	3,69	95,87	
<i>Election</i>	wszystkie reg.	>260 000	–	–	–
	10%	828	26,91	3,48	89,39
	15%	87	33,82	3,05	87,37
	20%	8	53,75	2,38	73,88
	25%	2	79	1,5	32,96
	30%	1	105	1	23,64
	LEM2	48	21,176	3,27	89,41

Kolejno używając algorytmu EXPLORE, systematycznie testowano następujące wartości progu  $l$ : 0%, 5%, 10%, 15%, 20%, 30%. Informacja o charakterystyce otrzymanych reguł przedstawiona jest w tabeli 3.2.

Wyniki wskazują, że dla wszystkich zbiorów danych jest możliwe zidentyfikowanie przynajmniej jednego zbioru reguł otrzymanego za pomocą algorytmu EXPLORE, który ma dobre właściwości opisowe (ze względu na wybrane miary).

Takie zbiory składają się z ograniczonej liczby reguł (tj. porównywalnej z liczbą reguł otrzymaną za pomocą algorytmu LEM2) charakteryzujących się średnim pokryciem większym przynajmniej dwukrotnie od reguł z porównywanych algorytmów. Ich średnia długość jest także krótsza. Ponadto wybrane zbiory reguł pozwalają na osiągnięcie trafności klasyfikowania porównywalnej z najlepszą osiągniętą w wyniku użycia algorytmu LEM2 lub podejścia wszystkich reguł.

Przykładowo, dla zbioru *Iris*, interesujący jest zakres progu  $l$  pomiędzy 10% i 15%. Wybierając  $l = 15\%$ , otrzymujemy zbiór 20 reguł o lepszych właściwościach opisowych i akceptowalnej trafności klasyfikowania. Podobnie dla pozostałych zbiorów danych następujące wartości progu  $l$  prowadzą do interesujących rezultatów: 5–10% dla *Tic-tac-toe*, 20–30% dla *Voting* i ok. 15% dla *Election*. Możliwe jest dokładniejsze przebadanie tych podzakresów (patrz [147]).

Zauważmy, że tworzenie zbioru wszystkich reguł jest nieefektywne zarówno ze względu na złożoność obliczeń, nieczytelność modelu, jak i na słabe wartości miar opisowych. Względnie atrakcyjna jest wyłącznie trafność klasyfikacji. Koszt obliczeniowy uzyskania takiego zbioru reguł jest jednak wyższy niż w przypadku innych rozważanych metod.

Dokładniejsza analiza wpływu progu  $l$  na „jakość” wygenerowanych reguł wskazuje, że przekroczenie pewnych wartości powoduje silne pogorszenie przynajmniej jednej z miar opisujących reguły. W szczególności dotyczy to liczby reguł i trafności klasyfikowania (porównaj np. wartości  $l = 15\%$  dla *Tic-tac-toe* czy 25% dla *Election*). Wartości te zależą niewątpliwie od zbioru danych, ale dla wszystkich obserwujemy gwałtowną zmianę miar opisowych. Podobne obserwacje poczyniono dla innych zbiorów danych w [234, 236].

Pomimo występowania powyższego zjawiska jest możliwe dla każdego z badanych zbiorów danych określenie pewnych zakresów zmienności wartości progu  $l$ , które prowadzą do akceptowalnego kompromisu pomiędzy wartościami czterech miar ( $N(R)$ ,  $n_{PO}$ ,  $DI(P)$ , oraz  $\eta_{ov}$ ). Z jednej strony, wartości te mogą być użyte jako początkowe do dalszego przeszukiwania przestrzeni reguł w interaktywnym i iteracyjnym procesie odkrywania wiedzy. Z drugiej strony, wskazuje to, że jest możliwe zbudowanie w miarę prostego i ogólnego zbioru silnych reguł decyzyjnych pozwalających równocześnie na osiągnięcie akceptowalnej trafności klasyfikowania.

### 3.3.3. Automatyczne poszukiwanie ograniczonej liczby silnych reguł o akceptowalnych właściwościach klasyfikacyjnych

Wyniki eksperymentów świadczą o tym, że jest możliwe zbudowanie zbioru reguł o dobrych właściwościach opisowych i równocześnie o akceptowalnych właściwościach klasyfikacyjnych. Poszukiwanie takiego zbioru reguł oparte było na systematycznej zmianie parametru  $l$  i analizie charakterystyk miar oceny dla kolejnych zbiorów reguł, tak aby zidentyfikować interesujące zakresy wartości tych miar. W tym podejściu jest możliwy współdział (choć potencjalny) człowieka – analityka. Ponadto, w przypadku analizy dużych rozmiarów danych podejście mo-

że napotkać ograniczenia związane ze zbyt wysokimi kosztami obliczeniowymi. Wiązą się one z koniecznością wygenerowania dużej liczby zbiorów reguł, kierunkiem zmian parametru  $l$  od 0% do wyższych wartości, a także ze złożonością obliczeniową samego algorytmu EXPLORE.

Można rozważyć inny sposób budowy zbioru reguł. Powinien się on charakteryzować przynajmniej pewnym stopniem zautomatyzowania oraz uwzględniać wymóg zmniejszania kosztów obliczeń. Proponujemy takie podejście, kierując się motywacją rozpoczęcia poszukiwań od możliwie najprostszyc zbiorów reguł, tzn. zawierających jak najmniej najsilniejszych reguł. Jeśli taki zbiór nie jest satysfakcjonujący ze względu na wybrane miary oceny, to poszukuje się kolejnego zbioru reguł o większym stopniu złożoności. Odpowiada to krokowej zmianie parametru  $l$  od wysokich wartości do niższych. Warunki zatrzymania takiej procedury mogą obejmować osiągnięcie satysfakcjonujących wartości wybranych miar oceny lub silne pogorszenie wartości innych kryteriów związanych z warunkami zatrzymania algorytmu EXPLORE. Ogólny schemat procedury jest następujący:

**procedure** *Tunethresholds* ( **var**  $\mathcal{L}$ : zbiór progów względnego pokrycia dla wszystkich klas;  
**var**  $l(K)$ : próg względnego pokrycia dla klasy decyzyjnej  $K$ );

**begin**

Przypisz początkowe wartości progom z  $\mathcal{L}$ ;

Wygeneruj zbiór reguł  $R$  i oblicz wybrane miary oceny;

$best := R$ ;

$bestL := \mathcal{L}$ ;

**for** każda klasa decyzyjna  $K$  **do**

**repeat**

zmniejsz wartość  $l(K)$ ;

Wygeneruj zbiór reguł  $R$  i oblicz miary oceny;

**if** *classification*( $R$ ) lepsza niż *classification*( $best$ ) **and** not(*stop*) **then**

**begin**

$best := R$ ;

$bestL := \mathcal{L}$

**end**

**until** (*classification*( $best$ ) - *classification*( $R$ )  $\leq \delta$ ) **and** *stop*;

$\mathcal{L} := bestL$ ;

**end.**

*Classification* jest funkcją oceniającą zdolności klasyfikacyjne zbioru reguł, np. trafność klasyfikowania, wrażliwość klasy decyzyjnej. Jeśli dwa różne zbiory reguł prowadzą do podobnej oceny klasyfikacyjnej, za lepszy uznaje się ten zbiór, który zawiera mniej reguł; próg rozróżnialności  $\delta$  oznacza maksymalne dopuszczalne pogorszenie trafności klasyfikowania. Funkcja *stop* związana jest z warunkami zatrzymania. Jeśli choć jeden z nich zostanie spełniony, to proces poszukiwania zbioru reguł będzie zakończony. Warunki te mogą być definiowane w szerokim sensie; przykładowo użytkownik może dla każdej klasy decyzyjnej określić mak-



symalną liczbę reguł decyzyjnych, maksymalną długość reguły czy spadek wartości wrażliwości dla wybranych klas decyzyjnych.

Dobór wartości progowych  $l$  pokrycia reguły wykonywany jest indywidualnie dla każdej klasy decyzyjnej. Ma to pozwolić na lepsze dostosowanie się do specyfikacji zbioru danych, w szczególności dla danych o nierównomierniej liczności klas decyzyjnych. Użycie standardowych algorytmów indukcji dla takich danych prowadzi do zbioru reguł, gdzie dominującą funkcję pełnią reguły związane z większościową klasą decyzyjną. Początkowe wartości progów  $\mathcal{L}$  powinny być dobierane jako wysokie, tak aby pierwszy ze zbiorów reguł zawierał nieliczne reguły (nawet pojedyncze) dla każdej klasy decyzyjnej. Następnie należy wartości  $l(K)$  systematycznie zmniejszać, przy czym krok powinien być ustalany zgodnie ze specyfikacją dostępnych danych. Początkowe  $l(K)$  wartości mogą być ustalone na podstawie wyniku działania standardowego algorytmu indukcji minimalnego zbioru reguł, np. LEM2 [247].

Zauważmy, że przedstawiony schemat postępowania prowadzi do rozważania najprostszycy zbiorów reguł o jak najniższym koszcie obliczenia jako pierwszych. Możliwe jest dodatkowo narzucenie ograniczenia, że długość reguły nie może przekraczać pewnej niewielkiej liczby warunków. Doświadczenia eksperymentalne i prace innych autorów wskazują, że ograniczone zbiory względnie prostych reguł wykorzystujących małą liczbę atrybutów (rekomendacja literaturowa 2–5 atrybutów) lepiej uogólniają dane i mają dobre właściwości klasyfikacyjne na wielu testowych zbiorach danych (interesujące rezultaty podane są przez Holtego [92] i Kowalczyka [114]; inne badania są zawarte w [4, 134, 234]).

Omawiana automatyczna procedura *Tunethresholds* zostanie użyta w budowie hybrydowego systemu klasyfikującego omawianego w rozdziale 7.3.

### 3.4. Interaktywne poszukiwanie reguł decyzyjnych

W części zadań odkrywania wiedzy można oczekiwać bardziej intensywnego współdziałania użytkownika z systemem. Jeżeli użytkownik jest ekspertem w danej dziedzinie, to może wyspecyfikować więcej wymagań w stosunku do składni reguł niż zwykły użytkownik zainteresowany tylko odnalezieniem najsilniejszych i najprostszycy reguł. Można zakładać, że ekspert chciałby poszukiwać bardziej specyficznych reguł oraz że dysponuje dostatecznie bogatą wiedzą o danym problemie, aby wskazać systemowi obszary danych i przestrzeni hipotez, w których indukcja reguł może przynieść interesujące rezultaty. Takie poszukiwanie reguł może się odbywać w sposób iteracyjny, co pozwala na stopniową ocenę wyników przez użytkownika i doprecyzowanie jego wymagań wobec poszukiwanych reguł [239].

Przedstawiony sposób współpracy z algorytmem EXPLORE był ukierunkowany na wybrane ilościowe miary oceny, tj.  $AS(Q|P)$ ,  $AS(P|Q)$ ,  $DI(P)$ ,  $N(R)$  oraz  $\eta_{ov}$ . Możliwe jest także sterowanie tym algorytmem dodatkowo za pomocą informacji od użytkownika odnośnie do wymaganej składni reguł. Omawiane powyżej rozszerzenie metody pozwala na uwzględnienie dodatkowo następujących parametrów:

- wybór atrybutu decyzyjnego i jego klas decyzyjnych,
- określenie zbioru preferowanych i zakazanych atrybutów warunkowych,
- określenie zbioru preferowanych i zakazanych warunków elementarnych,
- wybór preferowanego warunku złożonego (koniunkcji warunków elementarnych),
- określenie zbioru zakazanych warunków złożonych.

Wybór klasy decyzyjnej pozwala na wygenerowanie reguł dla podzbioru przykładów. Zbiór preferowanych i zakazanych atrybutów warunkowych określa wymogi co do składni części warunkowej reguł. Przyjmuje się, że każdy z atrybutów może przyjąć jeden z 3 stanów:

- atrybut neutralny,
- atrybut preferowany,
- atrybut zakazany.

Dla zbioru atrybutów preferowanych przynajmniej jeden atrybut musi zostać użyty w części warunkowej reguły. Zbiór atrybutów zakazanych określa te atrybuty, które nie mogą należeć do części warunkowej żadnej ze znalezionych reguł. Domyślnie wszystkie atrybuty są neutralne i mogą (lecz nie muszą) być wykorzystane do tworzenia części warunkowej poszukiwanych reguł. Podobne preferencje użytkownik może deklarować względem poszczególnych warunków elementarnych. Każdy warunek elementarny może przyjmować jeden ze stanów: neutralny, preferowany lub zakazany.

Preferowany warunek złożony, oznaczony przez *IP*, traktuje się jako kandydata na część warunkową reguły i próbuje się znaleźć tylko te reguły, których części warunkowe będą zawierały taki warunek złożony.

Niektóre parametry dotyczące składni części warunkowej reguły wykluczają się wzajemnie. Przykładowo można przyjąć, że jeśli użytkownik poda preferowany warunek złożony, to nie może określać preferencji względem warunków elementarnych. Natomiast jeśli się nie wyspecyfikuje warunku złożonego, to można ustalać właściwości na poziomie pojedynczych warunków. Zmodyfikowany algorytm o nazwie *Interactive Explore* ma na celu odnalezienie  $k$  najsilniejszych (pokrywających najwięcej przykładów pozytywnych) reguł decyzyjnych spełniających ograniczenia wobec powyższych parametrów, gdzie  $k$  jest parametrem definiowanym dla pojedynczej klasy decyzyjnej [239]. Omówmy krótko najważniejsze modyfikacje.

Uwzględnianie tylko podzbioru klas decyzyjnych jest naturalne, gdyż algorytm *Explore* jest uruchamiany dla kolejnych klas decyzyjnych. Zbiór atrybutów zakazanych jest wykorzystany do wstępnej redukcji tablicy decyzyjnej. Informacja o wyborze atrybutu jako preferowanego jest zamieniana na dołączenie warunków elementarnych z nim związanych do zbioru warunków preferowanych. Zbiory pre-

ferowanych i neutralnych warunków elementarnych umieszcza się posortowane według malejącego pokrycia odpowiednio na listach  $IS$  oraz  $NS$ . Następnie, w zależności od tego, czy zdefiniowano preferowany warunek złożony  $IP$ , czy nie, są one odpowiednio umieszczane na listach  $KI$  oraz  $LS$ . Zbiór warunków zakazanych umieszcza się na specjalnej liście  $FC$ . Lista ta jest dodatkowo wykorzystywana w funkcji *GoodCandidate* do oceny kandydata na część warunkową reguły. Poniżej prezentujemy uproszczony schemat zmodyfikowanej procedury.

**procedure** *Interactive\_EXPLORE* ( $IP, IS, NS, FC$ ;  $SC$  parametry podane przez użytkownika;  
**var**  $R$ : zbiór reguł);

**begin**

**if**  $IP \neq \emptyset$  **then begin**

$KI := IP$ ; {preferowany warunek złożony umieść na  $KI$ }

$LS := NS$  **end** {neutralne warunki umieść na liście  $LS$ }

**else if**  $IS \neq \emptyset$  **then** {jeśli zdefiniowano listę warunków preferowanych}

**begin**

$KI := IS$ ; {posortowaną listę preferowanych warunków umieść na  $KI$ }

$LS := IS \cup NS$

**end**

**else begin**

$KI := NS$ ; {posortowaną listę neutralnych warunków umieść na  $KI$ }

$LS := NS$

**end;**

posortuj listę  $KI$  malejąco względem pokrycia, a rosnąco względem jego długości;

*Explore*( $LS, SC, R$ ); {procedura zawiera modyfikacje}

**end.**

W zmodyfikowanej procedurze *EXPLORE* nie tworzy się kolejki  $KI$ , lecz korzysta z listy przygotowanej przed wywołaniem procedury. Ponadto w procedurze *GoodCandidate* na liście  $R$  zapamiętuje się wyłącznie  $k$  najsilniejszych reguł. Część warunkowa każdej znalezionej reguły umieszczana jest na liście  $FC$ .

Powyższa procedura może być użyta w iteracyjny sposób, umożliwiając użytkownikowi stopniowe modyfikowanie swoich wymogów wobec parametrów.

## 4. Indukcja reguł decyzyjnych z danych numerycznych

### 4.1. Wprowadzenie

Wiele algorytmów indukcji reguł decyzyjnych stosowanych w odkrywaniu wiedzy wymaga zdefiniowania tablicy decyzyjnej na podstawie atrybutów nominalnych. Z drugiej strony rzeczywiste bazy danych często zawierają dane numeryczne, tj. zdefiniowane za pomocą atrybutów o wartościach będącymi liczbami rzeczywistymi lub całkowitymi. W dalszych rozważaniach będziemy nazywać te atrybuty liczbowymi (ang. *numerical attributes*). Zauważmy, że stosowanie bezpośrednio do danych numerycznych typowych algorytmów, zwłaszcza wykorzystujących elementy teorii zbiorów przybliżonych, jest niedogodne ze względu na stosowane w algorytmie strategie tworzenia warunków elementarnych lub prowadzi do niezadowolających rezultatów. W studium eksperymentalnym [80] pokazano, że stosując w takim przypadku algorytm LEM2, otrzymuje się zbyt dużą liczbę reguł wspieranych małą liczbą obiektów. Otrzymane zbiory reguł charakteryzowały się także niską trafnością klasyfikacyjną.

Rzeczony problematyki odkrywania wiedzy zwiększył zainteresowanie analizą danych numerycznych. W praktyce proponuje się użycie jednego z dwóch podejść:

- zastosowania metod wstępnej dyskretyzacji przed dokonaniem indukcji reguł decyzyjnych,
- wyboru specjalizowanego algorytmu dostosowanego do indukcji reguł bezpośrednio z danych numerycznych.

Pierwsze podejście wykorzystuje dyskretyzację. **Dyskretyzacja** jest procesem zamiany atrybutów liczbowych na atrybuty symboliczne typu porządkowego. Polega ona na podziale oryginalnej dziedziny atrybutu liczbowego na pewną liczbę podprzedziałów i przypisaniu tym przedziałom kodów symbolicznych. Zaproponowano dotychczas wiele metod dyskretyzacji, które są stosowane w kroku **przetwarzania wstępnego** przed użyciem algorytmu indukcji reguł. W takim postępowaniu algorytm analizuje dane przetworzone do postaci symbolicznej, choć często nie wykorzystuje się informacji o porządku wartości atrybutów. Dotyczy to w szczególności podstawowych algorytmów teorii zbiorów przybliżonych opartych na relacji nierozróżnialności.

Drugie podejście wykorzystuje fakt, iż niektóre z algorytmów indukcji drzew czy reguł decyzyjnych mogą być użyte bezpośrednio na danych numerycznych bez wykonania wstępnej dyskretyzacji. Przykładowo algorytm C4.5 [185] pozwala na tworzenie uogólnionych warunków odpowiadających podziałowi binarnemu dziedziny takiego atrybutu w trakcie przeszukiwania przestrzeni kandydatów na węzły

drzewa. Część autorów nazywa taką operację rodzajem dyskretyzacji wewnętrznej algorytmu wykonywanej lokalnie w trakcie procesu uczenia się (patrz [35, 41, 75]). Podkreślmy, że w pierwszym podejściu dyskretyzacja jest w ogólności niezależna od stosowanego później algorytmu indukcji, podczas gdy w drugim podejściu jest częścią samego algorytmu i stosowane rozwiązania przetwarzania wartości atrybutów liczbowych są silnie związane ze strategiami samego algorytmu.

Obecnie większość prac badawczych dotyczy metod wstępnej dyskretyzacji. Właściwie dobrana dyskretyzacja może prowadzić przede wszystkim do poprawy miary trafności klasyfikowania, ale także innych miar opisowych stosowanych w odkrywaniu wiedzy [41, 75].

W tym rozdziale prezentuje się oryginalną propozycję autora w zakresie drugiej grupy podejść. Jest to algorytm MODLEM, który może być stosowany bezpośrednio do danych z atrybutami liczbowymi [235]. Tworzenie uogólnionych warunków elementarnych obejmujących przedziały wartości jest wykonywane lokalnie w trakcie indukcji reguł. Część wykorzystanych strategii przeszukiwania przestrzeni możliwych koniunkcji warunków elementarnych jest inspirowana rozwiązaniami algorytmu LEM2 [73], który jest jednym z popularniejszych algorytmów indukcji minimalnego zbioru reguł. W związku z tym wykonano także eksperymentalne studium porównujące użycie algorytmu MODLEM oraz podejście tradycyjne, w którym dokonuje się wstępnej dyskretyzacji globalną wersją metody opartej na entropii warunkowej i stosuje się algorytm LEM2 do indukcji reguł. Podstawowe informacje o metodzie dyskretyzacji podane są w następnych podrozdziałach.

## 4.2. Dyskretyzacja atrybutów liczbowych

Niech  $DT = (U, A \cup \{d\})$  będzie tablicą decyzyjną. Załóżmy, że dziedziną atrybutu liczbowego  $a \in A$  jest postaci  $V_a = [l_a, r_a] \subset \mathbb{R}$ , gdzie  $\mathbb{R}$  oznacza zbiór liczb rzeczywistych. Niech  $\mathbf{P}_a$  będzie podziałem  $V_a$  na przedziały, tzn.

$$\mathbf{P}_a = \{[c_0^a, c_1^a], [c_1^a, c_2^a], \dots, [c_{k_a-1}^a, c_{k_a}^a]\} \quad (4.1)$$

gdzie  $l_a = c_0^a < c_1^a < c_2^a < \dots < c_{k_a-1}^a < c_{k_a}^a = r_a$ , a  $k_a$  jest liczbą przedziałów. Dziedzina atrybutu  $a$  może być określona jako  $V_a = [c_0^a, c_1^a) \cup [c_1^a, c_2^a) \cup \dots \cup [c_{k_a-1}^a, c_{k_a}^a]$ . Podział  $\mathbf{P}_a$  jest jednoznacznie zdefiniowany za pomocą zbioru  $C_a = \{c_1^a, c_2^a, \dots, c_{k_a-1}^a\}$  nazywanego zbiorem punktów granicznych lub cięć (ang. *cut points*). Nowe przedziały z  $\mathbf{P}_a$  mogą być zastąpione przez kody (np. kolejne liczby naturalne), nazwy symboliczne czy oznaczenia samych przedziałów – różnica dotyczy wyłącznie aspektu syntaktycznego [75]. Możemy przyjąć, że kolejnym przedziałom przypisuje się uporządkowany ciąg wartości  $v_1^a < v_2^a < \dots < v_{k_a-1}^a$ .

Transformację dziedziny atrybutu liczbowego  $V_a$  na podstawie podziału  $\mathbf{P}_a$  do uporządkowanego ciągu wartości  $v_i^a$  ( $i = 1, \dots, k_a - 1$ ) nazywamy **dyskretyzacją** atrybutu  $a$ .

### 4.3. Metody dyskretyzacji wstępnej

Dotychczas zaproponowano wiele metod dyskretyzacji wstępnej wykonywanej dla atrybutów opisujących obiekty reprezentowane w tablicach informacyjnych lub decyzyjnych. Studia przeglądowe omawiające i systematyzujące różne podejścia dostępne są w [33, 35, 41, 75, 154]. Zgodnie z nimi można dokonać klasyfikacji dostępnych metod na:

- nienadzorowane i nadzorowane,
- lokalne i globalne,
- statyczne i dynamiczne.

Metody dyskretyzacji, które nie wykorzystują informacji o przydziale obiektów do klas decyzyjnych, nazywane są nienadzorowanymi (ang. *unsupervised* lub *class-blind*). Metody, w których ta informacja jest wykorzystywana, nazywane są nadzorowanymi (ang. *supervised*).

Niektóre metody przetwarzają informację o wszystkich atrybutach liczbowych równocześnie i w trakcie obliczeń starają się znaleźć punkty graniczne równocześnie dla wielu atrybutów. Są one nazywane dynamicznymi [33, 41]. Inne metody, nazywane statycznymi, ograniczają w danym momencie analizę do pojedynczego atrybutu, starając się znaleźć dla niego właściwą liczbę przedziałów niezależnie od wartości innych atrybutów. Niektórzy autorzy [251] wyróżniają wśród metod statycznych także takie, które wymagają podania jako parametru stałej liczby przedziałów.

Metody lokalne tworzą podziały, które odnoszą się do lokalnych obszarów przestrzeni dziedzin atrybutów opisujących przykłady uczące. Metody globalne budują „siatkę” (ang. *mesh*) na  $m$ -wymiarowej przestrzeni liczb rzeczywistych, gdzie zbiór wartości każdego z  $m$  atrybutów jest podzielony na przedziały niezależnie od innych.

Informacje o niektórych metodach w ramach powyższej klasyfikacji przedstawione są w tabeli 4.1. Celem niniejszego podrozdziału nie jest szczegółowe omawianie powyższych metod; czytelnik może zapoznać się z takimi omówieniami np. w [33, 35, 41, 75, 154]. W dalszej części krótko scharakteryzujemy wyłącznie metodę globalną opartą na minimalizacji entropii, gdyż będzie ona stosowana w eksperymentach porównawczych.

Złożoność obliczeniowa problemów dyskretyzacji była omawiana przez Nguyen [154, 157]. Pokazał on, że problem poszukiwania optymalnej dyskretyzacji, tj. dyskretyzacji z minimalną liczbą punktów granicznych, która zapewnia rozróżnialność obiektów z różnych klas decyzyjnych, jest NP-zupełny. Praktycznie stosowane algorytmy są heurystykami. W pracy [33] podaje się, że algorytmy globalne i nadzo-

rowane charakteryzują się najczęściej wyższymi kosztami obliczeniowymi niż algorytmy lokalne.

**Tabela 4.1.** Wybrane metody wstępnej dyskretyzacji

Kategorie podziału	Globalne	Lokalne
Nadzorowane	Algorytm ChiMerge [104] Globalne wersje metod wykorzystujących entropię [33, 255] Kwantowanie adaptacyjne [29] Algorytmy oparte na zasadzie minimalizacji długości opisu MDL [48] Algorytm 1RD tworzenia przedziałów z klasami większościowymi [92] Nadzorowana aglomeracyjna analiza skupień [33] Metody wykorzystujące funkcje boolowskie – heurystyka MD [154, 157] Podział przestrzeni na podstawie minimalizacji globalnego kosztu błędnego zaklasyfikowania [128, 129]	Lokalna metoda oparta na minimalizacji miary entropii [47] Hierarchiczna metoda maksymalizacji entropii [32] Kwantyzacja wektorowa Metody łączenia sąsiednich przedziałów [33, 78]
Nienadzorowane	Metoda równych przedziałów Metoda przedziałów o równej częstości obiektów Nienadzorowana analiza skupień oparta na kryterium MCC [268]	Grupowanie lokalne oparte na algorytmie „ <i>k</i> -means”

Podsumowując, należy zauważyć, że wielu autorów pisze, iż żadna metoda nie jest ogólnie najlepsza (ze względu na trafność klasyfikowania; rozważa się także koszty obliczeniowe) dla szerokiego zakresu problemów. Pisze się, że dobór właściwej metody jest raczej trudną „sztuką” [75]. Z drugiej strony część autorów pisze, że metody dyskretyzacji oparte na różnych wersjach entropii warunkowej działają zadowalająco w odniesieniu do wielu problemów testowych [41, 75, 107].

#### 4.4. Bezpośrednia indukcja z danych numerycznych

Metody te nie wymagają stosowania wstępnej dyskretyzacji danych numerycznych. Przykładem jest algorytm indukcji drzew decyzyjnych C4.5 [185]. W przypadku przetwarzania danych numerycznych konstruowane jest drzewo binarne. Kandydaci na węzły takiego drzewa są rozważani na podstawie warunków o postaci ( $a < r_a$ ) lub ( $a \geq r_a$ ), to jest warunku sprawdzającego, czy wartość atrybutu liczbowego  $a$  jest poniżej lub powyżej pewnego progu  $r_a$ . Wybór wartości progu  $r_a$  dokonywany jest na podstawie analizy szeregu wartości będących średnimi arytmetycznymi pomiędzy uporządkowanymi wartościami atrybutu liczbowego  $a$  dla podzbioru

przykładów uczących rozważanego w danym węźle. Wybrane wartości progów są użyte do tworzenia binarnych podziałów zakresu wartości atrybutu  $a$  dla tych obiektów. Jakość podziałów jest następnie oceniana za pomocą kryterium stosowanego w indukcji drzewa (w C4.5 miary entropii [185]).

W propozycjach kilku algorytmów indukcji reguł znajdują się także elementy „ukrytej dyskretyzacji lokalnej” dla pojedynczych atrybutów. Algorytm PVM [263] dokonuje nadzorowanej dyskretyzacji przed zbudowaniem warunku elementarnego, kierując się maksymalizacją zdolności predykcyjnej. Globalna i dynamiczna dyskretyzacja nadzorowana jest także pierwszym i najważniejszym etapem algorytmu ProbRough Lenarcika i Piasty [129]. Tworzenie reguł w tym algorytmie polega na odpowiednim przekształcaniu i upraszczaniu opisu podziału przestrzeni atrybutów wynikającego z dyskretyzacji.

Należy także wspomnieć o uogólnieniach teorii zbiorów przybliżonych. Oryginalna postać tej teorii jest oparta na relacji nierozróżnialności (2.8). Praktyczne jej zastosowania wymagają więc stosowania wstępnej dyskretyzacji. Analiza rzeczywistych wartości atrybutów ilościowych możliwa jest natomiast w uogólnieniach teorii zbiorów przybliżonych opartych na **relacji tolerancji** [152, 208, 248] lub **relacji podobieństwa** (ang. *similarity*) [226, 227]. W tych uogólnieniach dwa obiekty  $x$  i  $y$  mogą być traktowane jako podobne do siebie ze względu na atrybut  $a$ , jeśli różnica wartości tego atrybutu  $f(a,x) - f(a,y)$  jest akceptowalnie mała. Zaproponowano już metody uczenia się relacji tolerancji lub podobieństwa z przykładów uczących [115, 152, 248, 249]. Rezultatem końcowym jest możliwość tworzenia reguł, w których warunki elementarne mają postać  $(f(a,x) \in [\varepsilon_l, \varepsilon_r])$ , gdzie  $\varepsilon_l$ ,  $\varepsilon_r$  są parametrami wyuczonymi z danych.

#### 4.5. Metoda globalnej wstępnej dyskretyzacji oparta na minimalizacji entropii warunkowej

Jest to metoda nadzorowana i dynamiczna wykorzystująca miarę entropii. Dla rozważanego zbioru przykładów  $S$  i atrybutu  $a$  entropia jest zdefiniowana jako  $\text{Ent}(S) = -\sum_{i=1}^r p_i \cdot \lg_2 p_i$ , gdzie  $p_i$  jest prawdopodobieństwem wystąpienia klasy  $K_i$ , a  $r$  jest liczbą klas. Prawdopodobieństwo  $p_i$  estymuje się jako stosunek liczby przykładów z  $S$  należących do klasy  $K_i$  do liczby wszystkich przykładów z  $S$ .

Dowolny punkt graniczny  $c^a$  dokonuje podziału binarnego zbioru  $S$  na dwa rozłączne podzbiory  $S_1$  oraz  $S_2$  ( $S = S_1 \cup S_2$ ). Dla takiego podziału entropia warunkowa zdefiniowana jest jako:

$$\frac{|S_1|}{|S|} \cdot \text{Ent}(S_1) + \frac{|S_2|}{|S|} \cdot \text{Ent}(S_2) \quad (4.2)$$

Dla danego atrybutu  $a$  wybiera się ten punkt graniczny, który minimalizuje wartość entropii warunkowej. Wartości atrybutu  $a$  powinny być posortowane przed analizą. Liczba kandydatów do sprawdzenia może być ograniczona na podstawie



wyników podanych przez Fayyada i Iraniego [47]. Dowiedli oni, że nie jest konieczne badanie punktów granicznych pomiędzy dwoma wartościami atrybutu charakteryzującego obiekty należące do tej samej klasy decyzyjnej.

Metodę opartą na entropii oryginalnie podano w wersji lokalnej dla pojedynczego atrybutu [41, 47]. Możliwe jest także użycie jej rekurencyjnie dla obu podzbiorów  $S_1$  oraz  $S_2$  utworzonych przez najlepszy punkt graniczny. Prowadzi to do wyboru potencjalnie kolejnych punktów granicznych dla tego samego atrybutu, dopóki nie zostanie spełniony pewien warunek zatrzymania. Fayyad i Irani proponują w [47] warunek wynikający z zasady minimalnej długości opisu.

W niniejszej pracy stosujemy inną wersję tej metody, tzw. globalną metodę [33]. W metodzie tej dla początkowego zbioru obiektów  $S$  i dla każdego z atrybutów poszukuje się najlepszego punktu granicznego. Wybiera się najkorzystniejszy z punktów na podstawie najniższej wartości wyrażenia (4.2). Dalej, na tej podstawie dokonuje się podziału zbioru obiektów na podzbiory  $S_1$  oraz  $S_2$ . Następnie procedurę powyższą stosuje się rekurencyjnie dla każdego z podzbiorów – poszukując kolejnego najlepszego punktu granicznego (rozpatrując znowu wszystkie atrybuty). Warunkiem zatrzymania algorytmu jest osiągnięcie jakości przybliżenia klasyfikacji (wzór 2.12) za pomocą zdyskretyzowanych danych powyżej pewnego progu. W pracy [33] podano, że domyślną wartością tego progu jest 1,0, czyli stworzenie spójnej tablicy decyzyjnej.

Zauważmy, że w rezultacie zastosowania tej metody dla niektórych atrybutów można znaleźć większą liczbę punktów granicznych, podczas gdy dla innych atrybutów liczba ta będzie mniejsza lub atrybuty te nie zostaną użyte do dyskretyzacji. Porównanie metod wstępnej dyskretyzacji opartej na entropii warunkowej z innymi metodami pokazane jest w [33, 41, 107].

#### 4.6. Algorytm MODLEM

Algorytm MODLEM został zaproponowany przez autora w [235]. Motywacją do jego powstania było przetwarzanie bezpośrednio danych zawierających atrybuty liczbowe bez ich wcześniejszej dyskretyzacji. Algorytm MODLEM uwzględnia wartości tych atrybutów podczas indukcji reguł w momencie poszukiwania najlepszych warunków elementarnych. Propozycja sposobu ich uwzględniania jest w pewnym stopniu inspirowana techniką stosowaną w algorytmie indukcji drzew C4.5. Ogólny schemat poszukiwania reguł jest zbliżony do zasady generowania lokalnych pokryć stosowanej w algorytmie LEM2.

Algorytm MODLEM, podobnie jak jego poprzednik, LEM2, jest algorytmem heurystycznym generującym minimalny zbiór reguł (porównaj dyskusję w rozdz. 2.2). Aktualna postać algorytmu MODLEM może wykorzystywać elementy teorii zbiorów przybliżonych przez generację reguł z przybliżeń klas decyzyjnych. Jest on uruchamiany iteracyjnie dla kolejnych przybliżeń klas decyzyjnych.

Warunki elementarne są zdefiniowane i poszukiwane w inny sposób niż w algorytmie LEM2. Niech  $DT = \{U, A \cup \{d\}\}$  będzie tablicą decyzyjną zawierającą przy-

klasy uczące. W przypadku algorytmu LEM2 warunek elementarny  $w$  dla atrybutu  $a \in A$  był zdefiniowany jako wyrażenie  $(f(a,x) = v_a)$ , gdzie  $v_a$  jest wartością z dziedziny  $V_a$ . W algorytmie MODLEM stosowana jest następująca reprezentacja  $w$ :

$$\begin{aligned} (f(a,x) = v_a), \text{ gdy } a \text{ jest atrybutem typu nominalnego,} \\ (f(a,x) < v_a) \text{ lub } (f(a,x) \geq v_a), \text{ gdy atrybut jest typu liczbowego.} \end{aligned} \quad (4.3)$$

Wybór reprezentacji  $(f(a,x) < v_a)$  lub  $(f(a,x) \geq v_a)$  jest dokonywany w trakcie poszukiwania warunku elementarnego na podstawie maksymalnego pokrycia  $[w]$  klasy decyzyjnej (lub jej przybliżenia), dla której indukuje się reguły. W dalszych rozważaniach przez  $K$  będziemy oznaczali zbiór obiektów będących odpowiednim przybliżeniem klasy decyzyjnej  $X_j$  w  $DT$ .

Koniunkcja warunków elementarnych  $P = w_1 \wedge w_2 \wedge \dots \wedge w_m$  może być zaakceptowana jako część warunkowa reguły wskazującej  $K$ , jeżeli:

$$\emptyset \neq [P] = \bigcap_{i=1}^m [w_i] \subseteq K \quad (4.4)$$

Ponadto  $P$  powinno być minimalną koniunkcją spełniającą warunek (4.4).

Przyjmujemy, że jeżeli ten sam atrybut liczbowy zostanie wybrany przynajmniej dwukrotnie podczas tworzenia koniunkcji  $P$ , to możliwe jest skonstruowanie warunku  $(f(a,x) \in [v_{a1}, v_{a2}])$ . Wynika to z nałożenia dwóch warunków elementarnych  $(f(a,x) \geq v_{a1})$  i  $(f(a,x) < v_{a2})$ , gdzie  $v_{a1} < v_{a2}$ . Poszukiwanie warunków elementarnych przedstawione jest w procedurze *Find\_best\_condition*.

Zapis algorytmu MODLEM dla pojedynczej iteracji wykonywanej dla zbioru obiektów  $K$  przedstawiony jest poniżej. W stosowanej notacji  $\mathbf{P}$  jest zbiorem koniunkcji  $P$  stanowiącym lokalne pokrycie zbioru  $K$ . Na podstawie tych koniunkcji są tworzone reguły decyzyjne wskazujące właściwe klasy decyzyjne.

#### Procedure MODLEM

(input:  $K$  zbiór przykładów;  $C$  zbiór atrybutów;

output: lokalne pokrycie  $\mathbf{P}$  zbioru  $K$ );

**begin**

$G := K$ ; {zbiór obiektów nie pokrytych dotychczas przez elementy z  $\mathbf{P}$ }

$\mathbf{P} := \emptyset$ ;

**while**  $G \neq \emptyset$  **do**

**begin**

$P := \emptyset$ ; {kandydat na część warunkową reguły}

$S := G$ ; {zbiór obiektów pokrywany przez  $P$ }

**while**  $(P = \emptyset)$  **or not**  $([P] \subseteq K)$  **do**

**begin**

$w := \emptyset$ ; {najlepszy kandydat na warunek elementarny}

$w\_eval := \infty$ ; {ocena warunku elementarnego  $w$  według wybranej miary}

**for** każdy atrybut  $a \in C$  **do**

**begin**

*Find\_best\_condition*( $a, S, new\_p, eval\_new\_p$ ); {poszukuj najlepszego warunku

```

                                elementarnego  $new\_p$  dla atrybutu  $a$ 
if  $eval\_new\_p < w\_eval$  then {sprawdź, czy warunek  $new\_p$  jest lepszy niż  $w$ }
begin
     $w := new\_p$ ;
     $w\_eval := new\_eval\_p$ ;
end;
end; {for}
 $P := P \cup \{w\}$ ; {dołącz najlepszy warunek do koniunkcji  $P$ }
 $S := S \cap [w]$ ; {ogranicz zbiór obiektów}
end; {while not ( $[P] \subseteq K$ )}
for każdy warunek elementarny  $w \in P$  do
if  $[P - \{w\}] \subseteq K$  then  $P := P - \{w\}$ ;
 $\mathbf{P} := \mathbf{P} \cup \{P\}$ ;
 $G := K - \bigcup_{P \in \mathbf{P}} [P]$ ;
end; {while  $G \neq \emptyset$ }
for każdy  $P \in \mathbf{P}$  do
    if  $\bigcup_{T \in \mathbf{P} - \{P\}} [T] = K$  then  $\mathbf{P} := \mathbf{P} - \{P\}$ ;
end {procedure}.

procedure Find_best_condition
(input    $a$  atrybut;  $S$  zbiór obiektów;
output   $best\_t$  aktualnie najlepszy warunek elementarny
           $eval\_best\_t$  ocena najlepszego warunku);

begin
     $best\_t := \emptyset$ ;
     $eval\_best\_t := \infty$ ;
    sort  $H$ ; {stwórz listę wartości atrybutu  $a$  dla obiektów z  $S$  uporządkowaną
              zgodnie z rosnącymi wartościami;  $H(i)$  jest  $i$ -tą wartością na liście}
    for  $i := 1$  to length( $H$ ) - 1 do
    begin
         $v := (H(i) + H(i + 1)) / 2$ ; {rozpatruj jako punkty graniczne tylko te punkty, które
                                     oddzielają przykłady należące do różnych klas}
         $S_1 := \{x \in S \mid f(a,x) < v\}$ ; {dokonaj podziału obiektów z  $S$  według progu  $v$ }
         $S_2 := \{x \in S \mid f(a,x) \geq v\}$ ;
         $eval\_t := (|S_1|/|S_1 \cup S_2|) \cdot Ent(S_1) + (|S_2|/|S_1 \cup S_2|) \cdot Ent(S_2)$ ; {oblicz entropię warunkową}
        if  $eval\_t < eval\_best\_t$  then
            begin
                if  $|G \cap S_2| \geq |G \cap S_1|$  then
                     $best\_t := (f(a,x) \geq v)$  else  $best\_t := (f(a,x) < v)$ ;
                     $eval\_best\_t := eval\_t$ 
                end {if}
            end {if}
        end
    end {procedure }.

```

Skomentujmy przebieg funkcji *Find\_best\_condition*. Wykonywana jest oddzielnie dla każdego atrybutu  $a \in C$ . Na początku dla rozpatrywanego podzbioru

obiektów  $x \in S$  ich wartości atrybutu  $f(x, a)$  są sortowane według wzrastającego porządku. Następnie poszukuje się najlepszego punktu granicznego definiującego podział binarny obiektów  $S$  oceniany za pomocą entropii warunkowej (wzór 4.2). Jako warunek elementarny jest wybierany ten z dwóch możliwych, który definiuje przedział pokrywający więcej przykładów pozytywnych.

Algorytm MODLEM, podobnie jak jego poprzednik, LEM2, jest algorytmem heurystycznym ukierunkowanym na indukcję minimalnego zbioru reguł. Powyżej przedstawiony sposób wyboru kolejnych warunków elementarnych w ogólności nie gwarantuje znalezienia optymalnego minimalnego zbioru reguł. Z drugiej strony, jak wskazują dalej przedstawione wyniki oceny eksperymentalnej tego algorytmu, zarówno liczba reguł, jak i liczba użytych warunków jest satysfakcjonująca.

Podczas prezentacji algorytmu użyto miary entropii warunkowej do oceny warunków elementarnych. W [81] przedstawiono alternatywną propozycję użycia estymaty Laplace'a (patrz wzór 2.7). Zauważmy, że odróżnieniu od entropii, preferowane są większe wartości tej miary. Propozycja użycia miary Laplace'a wynika z obserwacji, że dla niektórych danych miara entropii może faworyzować wybór warunków „czystych” ze względu na wskazanie klasy decyzyjnych, lecz pokrywających małą liczbę przykładów. Jak dyskutowano w rozdziale 2, estymata Laplace'a w większym stopniu preferuje wybór warunków pokrywających dużą liczbę przykładów. Właściwości obu miar dyskutowano już w przypadku różnych rozszerzeń algorytmu CN2 [36, 37, 45]. W dalszej części oceny eksperymentalnej algorytm MODLEM będzie użyty w dwóch wersjach, MODLEM-Entropii oraz MODLEM-Laplace, w zależności od wykorzystywanej miary.

W powyższym opisie przedstawiono dla prostoty zapis algorytmu MODLEM tylko dla atrybutów liczbowych. W ogólności algorytm uwzględnia również atrybuty nominalne [81]. Dla tych atrybutów warunki są prezentowane w postaci  $(f(a, x) = v_a)$ . W funkcji *Find\_best\_condition* rozpatruje się wszystkie wartości  $v_a \in V_a$ . Każdy kandydat na warunek generuje podział zbioru  $S$  na podzbiór  $S_1 = \{x \in S \mid f(a, x) = v_a\}$  i jego dopełnienie ( $S_2 = S - S_1$ ), które są wykorzystywane do oceny warunku w podobny sposób jak dla atrybutów liczbowych. Algorytm MODLEM można stosować do danych zawierających atrybuty typu liczbowego i atrybuty typu nominalnego. Więcej szczegółów samego algorytmu przedstawiono w [81, 235].

Przedyskutujmy krótko złożoność obliczeniową algorytmu MODLEM. Załóżmy, że podstawową operacją jest ocena pojedynczego warunku elementarnego (zbudowanego z wykorzystaniem progów  $v$ ). W najgorszym przypadku pojedyncza reguła pokrywa jeden obiekt i wykorzystuje do zbudowania warunków elementarnych wszystkie atrybuty. Jeśli  $n$  jest liczbą przykładów,  $W$  – średnią liczbą wartości w dziedzinie atrybutu,  $m$  – liczbą atrybutów, to potrzebne jest w najgorszym przypadku  $nWm(m + 1)/2$  operacji. Funkcja złożoności algorytmu MODLEM jest wielomianowa, rzędu  $O(nWm^2)$ .

#### 4.7. Eksperymenty obliczeniowe

W celu oceny przydatności nowej propozycji algorytmu wykonano eksperyment obliczeniowy. Porównano w nim trzy podejścia do przetwarzania danych numerycznych:

- zastosowanie bezpośrednio algorytmu MODLEM-Entropy,
- zastosowanie bezpośrednio algorytmu MODLEM-Laplace,
- dwuetapowe „podejście tradycyjne”, tzn. wykonanie wstępnej dyskretyzacji metodą globalną opartą na entropii warunkowej i następnie indukcja reguł oryginalnym algorytmem LEM2.

Eksperymenty wykonano na standardowych zbiorach testowych pochodzących z University of California w Irvine [14] lub ze znanych zastosowań teorii zbiorów przybliżonych. Podstawowa charakterystyka tych danych przedstawiona jest w tabeli 4.2. Dane te są zdefiniowane wyłącznie lub w bardzo dużym stopniu za pomocą atrybutów liczbowych.

**Tabela 4.2.** Charakterystyka danych

Zbiór danych	Liczba przykładów	Liczba atrybutów	Liczba klas decyzyjnych
<i>Bank</i>	66	5	2
<i>Bupa</i>	345	6	2
<i>Buses</i>	76	8	2
<i>Glass</i>	214	9	6
<i>HSV</i>	122	11	2
<i>Iris</i>	150	4	3
<i>Pima</i>	768	8	2
<i>Bricks</i>	216	10	2
<i>Segmentation</i>	210	19	7
<i>German (numeric)</i>	1000	24	2

**Tabela 4.3.** Algorytm MODLEM-Laplace

Zbiór danych	Liczba reguł	Liczba warunków	Średnie wsparcie	Trafność [%]
<i>Bank</i>	6	7	22	94
<i>Bupa</i>	101	228	5,6	68
<i>Buses</i>	5	5	35	97
<i>Glass</i>	80	139	3,6	58
<i>HSV</i>	54	96	2,9	63
<i>Iris</i>	12	24	19,8	91
<i>Pima</i>	188	400	9,1	74
<i>Bricks</i>	22	38	17,7	91
<i>Segementation</i>	47	80	5,3	72
<i>German (numeric)</i>	253	774	8,4	73

**Tabela 4.4.** Algorytm MODLEM-Entropy

Zbiór danych	Liczba reguł	Liczba warunków	Średnie wsparcie	Trafność [%]
<i>Bank</i>	3	16	28	94
<i>Bupa</i>	79	219	6,5	66
<i>Buses</i>	4	5	35,3	97
<i>Glass</i>	43	111	6,8	72
<i>HSV</i>	35	92	4,7	57
<i>Iris</i>	10	20	20,3	94
<i>Pima</i>	125	426	13	74
<i>Bricks</i>	16	33	25,2	91
<i>Segementation</i>	22	45	10,5	85
<i>German (numeric)</i>	182	751	9,3	73

**Tabela 4.5.** Dyskretyzacja wstępna oparta na entropii oraz LEM2

Zbiór danych	Liczba reguł	Liczba warunków	Średnie wsparcie	Trafność [%]
<i>Bank</i>	10	13	7,5	97
<i>Bupa</i>	169	501	2,4	66
<i>Buses</i>	3	4	33,3	99
<i>Glass</i>	111	262	2,2	67
<i>HSV</i>	62	206	2,3	56
<i>Iris</i>	14	33	12,6	97
<i>Pima</i>	252	895	4,6	74
<i>Bricks</i>	25	61	9,2	92
<i>Segementation</i>	108	322	2	64
<i>German (numeric)</i>	290	1226	5,5	74

Trzy następujące tabele (4.3–4.5) prezentują wyniki eksperymentów porównawczych. Zbiory reguł oceniane są w kategoriach: liczby reguł (im mniej, tym lepiej), średniego wsparcia reguły (tj. średniej liczby przykładów wspierających regułę; im więcej, tym lepiej), łącznej liczby warunków elementarnych użytych w regułach (im mniej, tym lepiej) oraz oceny trafności klasyfikowania (im więcej, tym lepiej). W przypadku klasyfikowania obiektów dla wszystkich porównywanych podejść użyto tej samej techniki stosowanej w systemie LERS [74]; została ona przedstawiona w rozdziale 2.3. Ocenę trafności klasyfikowania wykonano za pomocą techniki *10-fold cross validation*.

#### 4.8. Dyskusja otrzymanych wyników

Rezultaty eksperymentu obliczeniowego oceniono globalnie (z punktu widzenia wszystkich zbiorów danych) za pomocą nieparametrycznego testu Wilcozona z poziomem istotności 5%. Wnioski są następujące:

1. Całkowita liczba reguł: najlepsze jest podejście MODLEM-Laplace, na drugim miejscu jest MODLEM-Entropy, najgorsze jest podejście wykorzystujące dyskretyzację wstępną.
2. Całkowita liczba warunków: najlepsze jest podejście MODLEM-Laplace, w drugiej kolejności MODLEM-Entropy, najgorsze jest podejście wykorzystujące dyskretyzację wstępną.
3. Średnie wsparcie reguły: najlepszy jest algorytm MODLEM-Entropy, w drugiej kolejności MODLEM-Laplace, najgorsze jest podejście oparte na dyskretyzacji wstępnej.
4. Trafność klasyfikowania: różnice pomiędzy podejściami nie są statystycznie znaczące.

Można więc stwierdzić, że algorytm MODLEM pozwala dla badanych danych na indukcję prostszych reguł, wspieranych przez większą liczbę przykładów uczących. W przypadku trafności klasyfikowania wniosek dotyczy globalnego porównania na podstawie wszystkich zbiorów danych. Zauważmy jednak, że dla niektórych zbiorów danych, w szczególności *Glass*, *HSV* i *Segmentation*, występują przyrosty trafności klasyfikowania na rzecz algorytmu MODLEM. Należy także podkreślić, że do porównania wybrano literaturowo najskuteczniejszą metodę globalną opartą na entropii [41]; jeśli użyje się wersji lokalnej tej metody, wyniki są korzystniejsze na rzecz algorytmu MODLEM.

Podsumowując dyskusję problemów odkrywania reguł decyzyjnych z danych zawierających atrybuty ilościowe, należy zauważyć, że użytkownik może dokonać wyboru: stosować dyskretyzację wstępną lub specjalizowany algorytm indukcji.

W pierwszym przypadku metoda dyskretyzacji jest w ogólności niezależna od stosowanego później algorytmu indukcji reguł. Trudnością jest jednak dobór właściwej metody dyskretyzacji. Obserwacja praktycznych zastosowań wskazuje, że często stosuje się studia porównawcze kilku metod dyskretyzacji. Z drugiej strony,

co omówiono w [41, 86], właściwa dyskretyzacja może przyczynić się również do przyspieszenia indukcji reguł dla wielkich baz danych zawierających atrybuty o bardzo dużej liczbie wartości. Przykładowo Catlett [24] pokazał w analizie rzeczywistych danych pochodzących z NASA, że czas bezpośredniej indukcji drzewa decyzyjnego algorytmem ID3 Quinlana z bazy danych zawierających ponad milion rekordów jest nieakceptowany w rozważanym zastosowaniu.

W drugim przypadku sposób analizy przetwarzania danych numerycznych jest związany ze strategiami charakterystycznymi dla danego algorytmu. Autor zaproponował w tym zakresie oryginalną propozycję algorytmu MODLEM.

Należy także zauważyć, że w niektórych problemach przydatne może być rozważanie **dyskretyzacji rozmytej**, tzn. takiej, w której przedziały nie są definiowane na podstawie „ostrych” wartości punktów granicznych, lecz modelowane za pomocą liczb rozmytych o nakładających się wzajemnie zboczach. Podejście do analizy informacji tego typu w ramach teorii zbiorów przybliżonych dyskutowano w [220, 232]. Rozszerzono je w pracy [223], a następnie wraz z propozycją tworzenia rozmytej dyskretyzacji opartej na rozmytej wersji miary entropii zostało zastosowane do analizy rzeczywistego przypadku diagnostyki technicznej w [225]. Inne podejście do analizy dyskretyzacji rozmytej pokazano także w [155].

Interesujące jest także porównanie algorytmu MODLEM z innymi skutecznymi technikami budowy klasyfikatorów. W ostatnim czasie Doumpos i Zopounidis przedstawili eksperymentalne porównanie algorytmu MODLEM z trzema znanymi technikami wielowymiarowymi analizy dyskryminacyjnej (funkcje liniowe, kwadratowe i logistyczne) [42]. Eksperymenty przeprowadzono na wielu symulowanych danych numerycznych, gdzie w systematyczny sposób zmieniano rozkłady prawdopodobieństwa, liczbę klas decyzyjnych, liczbę przykładów, skorelowanie atrybutów itp. Przedstawione wyniki wskazują, że trafność klasyfikowania osiągnięta przez algorytm MODLEM jest lepsza od wyników analizy dyskryminacyjnej dla rozkładów danych odbiegających od normalnego (im większa asymetria rozkładów, tym większa przewaga MODLEM). Tylko w przypadku rozkładów normalnych niektóre z funkcji dyskryminacyjnych osiągały przewagę, co jest zgodne z intuicją i założeniami analizy dyskryminacyjnej. Podane wyniki należy uznać za zachęcające, gdyż wskazują, że algorytm MODLEM może być skutecznym narzędziem dla zadań klasyfikacyjnych.



## 5. Zastosowanie teorii zbiorów przybliżonych oraz indukcji reguł decyzyjnych do danych zawierających nieznane wartości atrybutów

### 5.1. Brakujące i niedostępne wartości atrybutów

Rzeczywiste dane, które podlegają procesowi odkrywania wiedzy, są często niespójne, niekompletne, mogą zawierać nieprecyzyjne opisy lub wymagać równoczesnej analizy różnego typu atrybutów. Część z tych problemów była już dyskutowana w poprzednich rozdziałach rozprawy. W niniejszym rozdziale rozważać będziemy tablice decyzyjne, w których występują nieznane (niezdefiniowane) wartości części atrybutów (ang. *unknown values of attributes*) dla niektórych obiektów. Jak wskazuje wielu autorów [50, 76, 86, 184, 262], jest to częsta sytuacja w zastosowaniach praktycznych. Z drugiej strony wiele metod odkrywania wiedzy, w szczególności metod indukcji reguł decyzyjnych, nie było przygotowanych w podstawowych wersjach do uwzględniania niezdefiniowanych wartości (porównaj [262, s. 61]). Problem ma znaczenie zarówno badawcze, jak i praktyczne. Proponowane dotychczas rozwiązania można podzielić na dwie grupy:

- 1) stosowane w **przetwarzaniu wstępnym danych** przed użyciem właściwego algorytmu indukcji,
- 2) stosowane jako część **wewnętrzna** samego algorytmu indukcji reguł.

Pierwsza grupa metod wykorzystuje założenie, że rozmiary danych są wystarczające do oszacowania prawdopodobieństwa wystąpienia wartości atrybutów. Większość z tych metod jest zorientowana na **zastąpienie** nieznanej wartości atrybutu dla określonego obiektu wartością z dziedziny tego atrybutu. Przykłady często stosowanych metod (patrz przeglądy w [76, 79, 86, 262]) to:

- – Zastąpienie za pomocą najczęściej występującej wartości atrybutu, określonej na podstawie przykładów opisanych zdefiniowaną wartością tego atrybutu. Przykładowo metody tej używa się w algorytmie CN2 [37]. Inną wersją tej metody jest stosowanie wartości średniej tego atrybutu w przypadku rozważania atrybutów liczbowych [86].
- – Użycie najczęstszej wartości atrybutu znajdującej na podstawie rozkładu wartości wśród przykładów należących tylko do tej samej klasy decyzyjnej co analizowany przykład.
- – Użycie zbioru wszystkich możliwych wartości tego atrybutu określonych dla wszystkich opisanych obiektów lub tylko obiektów z danej klasy [72].
- – Użycie podzbioru wartości atrybutu wraz z informacją o stopniach możliwości ich realizacji [223].

Rozważa się też inne bardzo proste podejścia:

- – Zignorowanie wszystkich przykładów opisanych przez przynajmniej jedną nieznaną wartość (należy tu zakładać, że liczba pozostałych przykładów jest wystarczająco reprezentatywna do dalszej indukcji wiedzy).
- – Użycie globalnej stałej wartości do zastąpienia brakujących danych; najczęściej stosowana w wersji traktowania nieznannej wartości atrybutu jako dodatkowej wartości w dziedzinie.

Inne propozycje opierają się na predykcji najbardziej prawdopodobnej wartości atrybutu dla danego obiektu na podstawie analizy zależności wielowymiarowej od innych atrybutów. Można spotkać w literaturze użycie do tego celu analizy regresji, formuł wywodzących się z twierdzenia Bayesa, indukcji drzew lub reguł decyzyjnych [86, 184].

W drugiej grupie metod zakłada się, że nieznanne wartości atrybutów nie będą zastępowane wartościami w trakcie przetwarzania wstępnego, lecz muszą być bezpośrednio uwzględniane w podejściu do indukcji wiedzy. Takie metody są więc częściami składowymi samego algorytmu indukcji i są z nim silnie powiązane. Przykładem jest rozwiązanie użyte w algorytmie C4.5 [185]. W przypadku rozważania atrybutu, dla którego występują obiekty z niezdefiniowanymi wartościami, stosuje się zmodyfikowaną miarę oceny (opartą na entropii). Jest ona wyliczana na podstawie zdefiniowanej części danych wraz z uwzględnieniem funkcji kary zależnej od względnej częstości nieznannej wartości dla tego atrybutu. Znana jest też modyfikacja algorytmu LEM2, która w prostszy sposób pomija niezdefiniowane wartości atrybutów w trakcie poszukiwania warunków elementarnych [83].

Prace porównujące eksperymentalnie niektóre powyższe metody z punktu widzenia skuteczności klasyfikowania z użyciem określonego algorytmu indukcji wiedzy dostępne są w: [184] – drzewa decyzyjne, [76] – reguły decyzyjne indukowane algorytmem LEM2. Wnioski tam przedstawione oraz dyskusje w [262, s. 61] sugerują, że „średnio” metody z drugiej grupy pozwalają na osiągnięcie lepszej trafności klasyfikacji.

Zainteresowania badawcze autora prezentowane w niniejszej rozprawie dotyczą użycia teorii zbiorów przybliżonych oraz indukcji reguł decyzyjnych z tablic decyzyjnych zawierających nieznanne wartości atrybutów. Należy tutaj zauważyć, że oryginalne sformułowanie teorii zbiorów przybliżonych [164, 165] wykorzystuje **relację nierozróżnialności** pomiędzy opisem obiektów (porównaj definicję (2.8)). Definicja ta odnosi się do sytuacji, gdy wartości atrybutu  $a_i$  dla porównywanych obiektów  $x$  i  $y$  są zdefiniowane. Natomiast definicja w oryginalnej postaci nie jest użyteczna, gdy przynajmniej jedna z wartości jest niezdefiniowana.

Oczywiście można zastosować metody z pierwszej grupy w celu transformacji tablicy danych zawierającej nieznanne wartości atrybutów do w pełni zdefiniowanej tablicy. W tym zakresie mieszczą się też propozycje podane przez Grzymałę [72, 76] lub przez Słowińskiego i Stefanowskiego [219]. Inne propozycje można zaliczyć do drugiej grupy metod. Obejmują uogólnienia teorii zbiorów przybliżonych oparte na

innych relacjach niż oryginalna postać relacji nierozróżnialności. Kryszkiewicz [118] zaproponowała podejście wykorzystujące relację tolerancji. Pewne uogólnienia były też rozważane w pracy [260]. Ostatnio Greco, Matarazzo i Słowiński zaproponowali jeszcze inne uogólnienie wykorzystujące specyficzną postać nierozróżnialności [65]. W ich interpretacji relacja nierozróżnialności jest porównaniem skierowanym od obiektu będącego przedmiotem porównania do obiektu będącego referentem – czyli odniesieniem tego porównania. Referent nie może zawierać żadnej niezdefiniowanej wartości. Uogólnienie to ma ciekawe właściwości formalne i odniesienie do zastosowania w wielokryterialnych problemach decyzyjnych [66].

Na tym tle autor zamierza zaprezentować oryginalne propozycje w zakresie uogólnienia teorii zbiorów przybliżonych i indukcji reguł decyzyjnych [242, 243, 244]. Rozszerzają one dotychczasowe uogólnienia teorii zbiorów przybliżonych poprzez rozróżnienie dwóch możliwych różnych semantyk nieznanych wartości atrybutu, nazywanych odpowiednio:

- semantyką **brakujących** wartości (ang. *missing values*),
- semantyką **niedostępnych** wartości.

Pierwsza interpretacja oznacza, że wartość atrybutu dla pewnego obiektu jest w danej chwili nieznaną, ale w ogólności istnieje. Z różnych przyczyn w chwili tworzenia zapisu danych nie została zarejestrowana lub została zagubiona. W ogólności obiekt może otrzymać pełen opis, lecz w chwili analizy jest on niekompletny. Wskazane powyżej uogólnienia teorii zbiorów przybliżonych są przeznaczone do analizy właśnie takich sytuacji.

Druga interpretacja odnosi się do sytuacji, w których niekompletność opisu wynika z ostatecznej niedostępności informacji o wartościach pewnych atrybutów dla niektórych obiektów. Przykładowo w analizie danych medycznych może to oznaczać fizyczną niewykonalność pewnych testów u pacjentów cierpiących na określoną alergię. Innymi słowy, opis takiego obiektu nigdy nie będzie kompletny dla takiej wartości.

Z tego powodu proponuje się dwa uogólnienia teorii zbiorów przybliżonych: pierwsze oparte na **wartościowanej relacji tolerancji** dla przypadku brakujących wartości oraz drugie oparte na **niesymetrycznej relacji podobieństwa** dla przypadku niedostępnych wartości.

Punktem odniesienia dla pierwszego podejścia, a także w pewnym stopniu dla drugiego, jest propozycja Kryszkiewicz oparta na „prostej” relacji tolerancji [118]. Dlatego w kolejnym podrozdziale przedstawimy krótko jej podstawowe definicje. Motywacją wprowadzenia nowej propozycji opartej na wartościowanej relacji tolerancji jest przekonanie, że propozycja prostej relacji tolerancji jest zbyt „zachowawcza” w analizie nieznanych wartości atrybutów i jej użycie może prowadzić do przybliżeń klas decyzyjnych o niskiej dokładności i zbyt ograniczonych zbiorów reguł. Motywacją do wprowadzenia propozycji opartej na relacji podobieństwa wynika z braku narzędzi do analizy semantyki niedostępnych wartości.

Przyjmijmy, że przedmiotem rozważań będą reprezentacje danych w postaci tablicy decyzyjnej  $DT = (U, A \cup \{d\})$ . Dla obiektu  $x \in U$  oraz atrybutu  $a \in A$  symbol  $*$  oznaczać będzie **nieznaną wartość atrybutu**  $f(a, x)$ . We wszystkich przedstawianych propozycjach zakładamy, że dla każdego obiektu  $x \in U$  wartość atrybutu decyzyjnego  $d$  jest zdefiniowana, tzn.  $f(d, x) \neq *$ .

Tablicę decyzyjną będziemy nazywać **niekompletną**, jeśli  $\exists x \in U$  taki, że przynajmniej dla jednego  $a \in A$  zachodzi  $f(a, x) = *$ .

W celu ilustracji oraz porównania dyskutowanych dalej podejść wprowadzimy prosty przykład tablicy decyzyjnej.

### Przykład 5.1

Dana jest następująca tablica decyzyjna  $DT$ , gdzie zbiór atrybutów  $A$  zawiera cztery atrybuty  $a_1, a_2, a_3, a_4$  zdefiniowane na czterowartościowej dziedzinie  $\{0, 1, 2, 3\}$ , a zbiór  $U$  składa się z 12 obiektów  $x_1, \dots, x_{12}$ , które mogą być przydzielone do dwóch klas decyzyjnych  $\Phi$  lub  $\Psi$ .

**Tabela 5.1.** Przykład tablicy decyzyjnej z niezdefiniowanymi wartościami atrybutów

$U$	$a_1$	$a_2$	$a_3$	$a_4$	$d$
$x_1$	3	2	1	0	$\Phi$
$x_2$	2	3	2	0	$\Phi$
$x_3$	2	3	2	0	$\Psi$
$x_4$	*	2	*	1	$\Phi$
$x_5$	*	2	*	1	$\Psi$
$x_6$	2	3	2	1	$\Psi$
$x_7$	3	*	*	3	$\Phi$
$x_8$	*	0	0	*	$\Psi$
$x_9$	3	2	1	3	$\Psi$
$x_{10}$	1	*	*	*	$\Phi$
$x_{11}$	*	2	*	*	$\Psi$
$x_{12}$	3	2	1	*	$\Phi$

## 5.2. Uwzględnianie brakujących wartości oparte na relacji tolerancji

Poniżej przedstawiamy krótko informacje o głównej idei podejścia zaproponowanego przez Kryszkiewicz [118, 119, 120] opartego na podanej przez nią definicji relacji tolerancji (ang. *tolerance relation*) dla niekompletnych tablic decyzyjnych. Należy zauważyć, że w tym podejściu dopuszcza się porównywanie pomiędzy sobą obiektów w przypadku występowania nieznanymi wartości atrybutów w ich opisach. Dla podzbioru atrybutów  $B \subseteq A$  relacja tolerancji jest zdefiniowana jako:

$$T_B = \{(x,y) \in U \times U : \forall a \in B, f(a,x) = f(a,y) \text{ lub } f(a,x) = * \text{ lub } f(a,y) = *\} \quad (5.1)$$

Relacja jest zwrotna, symetryczna, lecz nieprzechodnia. Niech  $T_B(x)$  oznacza zbiór obiektów  $y$ , dla których zachodzi relacja  $T_B$ . Innymi słowy jest to zbiór obiektów „podobnych” do  $x$  z uwagi na atrybuty z  $B$ . Definicje przybliżeń zbiorów są następujące:

$$\underline{B}_T(X) = \{x \in U : T_B(x) \subseteq X\} \quad (5.2)$$

$$\overline{B}_T(X) = \{x \in U : T_B(x) \cap X \neq \emptyset\} = \bigcup_{x \in X} T_B(x) \quad (5.3)$$

Podstawowe właściwości przybliżeń są dyskutowane w [118]. Definiuje się także pojęcie reduktu i pokazuje sposoby jego poszukiwania. Z punktu widzenia problemów omawianych w tym rozdziale należy zauważyć, że kluczowym elementem tego podejścia jest przyjęcie założenia, że za wartością nieznaną „\*” może się „ukrywać” każda dopuszczalna wartość z dziedziny tego atrybutu. Jeśli choć jeden z porównywanych obiektów,  $x$  lub  $y$ , przyjmuje dla dowolnego  $a \in B$  wartość „\*”, to przyjmuje się, że jest on podobny do drugiego, gdyż może przyjąć taką samą wartość atrybutu jak drugi obiekt.

Kryszkiewicz w pracach [118, 120] wprowadza składnie tzw. uogólnionych reguł decyzyjnych „jeżeli  $P$  to  $Q$ ”, gdzie  $P$  jest koniunkcją warunków ( $f(a_i, x) = v_{ai}$ ), a  $Q$  może być dysjunkcją ( $f(d, x) = v_d^j$ ). Jeżeli  $Q$  zawiera tylko jeden element, to reguła odpowiada postaci reguły pewnej omawianej w rozdziale 2.2. Jeśli  $C$  jest podzbiorem atrybutów użytych w części warunkowej  $P$ , to  $\forall x \in [P]$  zachodzi  $T_C(x) \subseteq [Q]$ . Więcej informacji na temat tego podejścia i reguł w nim używanych przedstawiono w [118, 119].

### Kontynuacja przykładu 5.1

Używając zbioru wszystkich atrybutów  $A$ , można znaleźć następujące klasy relacji tolerancji:  $T_A(x_1) = \{x_1, x_{11}, x_{12}\}$ ,  $T_A(x_2) = \{x_2, x_3\}$ ,  $T_A(x_3) = \{x_2, x_3\}$ ,  $T_A(x_4) = \{x_4, x_5, x_{10}, x_{11}, x_{12}\}$ ,  $T_A(x_5) = \{x_4, x_5, x_{10}, x_{11}, x_{12}\}$ ,  $T_A(x_6) = \{x_6\}$ ,  $T_A(x_7) = \{x_7, x_8, x_9, x_{11}, x_{12}\}$ ,  $T_A(x_8) = \{x_7, x_8, x_{10}\}$ ,  $T_A(x_9) = \{x_7, x_9, x_{11}, x_{12}\}$ ,  $T_A(x_{10}) = \{x_4, x_5, x_8, x_{10}, x_{11}\}$ ,  $T_A(x_{11}) = \{x_1, x_4, x_5, x_7, x_9, x_{10}, x_{11}, x_{12}\}$ ,  $T_A(x_{12}) = \{x_1, x_4, x_5, x_7, x_9, x_{11}, x_{12}\}$ .

Klasy decyzyjne  $\Phi$  i  $\Psi$  mogą być przybliżone jako:  $\underline{A}_T(\Phi) = \emptyset$ ,  $\overline{A}_T(\Phi) = \{x_1, x_2, x_3, x_4, x_5, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}\}$ ,  $\underline{A}_T(\Psi) = \{x_6\}$ ,  $\overline{A}_T(\Psi) = U$ .

W tabeli 5.1 można odkryć tylko jedną regułę pewną:

$$\text{jeżeli } (f(a_1, x) = 2) \wedge (f(a_2, x) = 3) \wedge (f(a_4, x) = 1) \text{ to } (f(d, x) = \Psi)$$

### 5.3. Uwzględnianie niedostępnych wartości atrybutów za pomocą relacji podobieństwa

W tej semantyce nieznaną wartość atrybutu interpretujemy jako niedostępną dla danego obiektu. W odróżnieniu od poprzedniego podejścia nie można porównywać ze sobą zdefiniowanej i niezdefiniowanej wartości atrybutu. Obiekty mogą posiadać mniej lub bardziej kompletny opis. Rozważanie podobieństwa pomiędzy obiektem  $x$  oraz obiektem  $y$  dotyczy tylko w pełni zdefiniowanych atrybutów. Konieczne jest więc użycie innej definicji relacji pomiędzy obiektami, która nie będzie w ogólności symetryczna, gdyż jeden z porównywanych obiektów może mieć bardziej kompletny opis niż drugi. Koncepcja wyłącznie zwrotnej relacji podobieństwa (ang. *similarity relation*) została wprowadzona do uogólnienia teorii zbiorów przybliżonych przez Słowińskiego i Vanderpootena [226, 227]. Podajemy własną definicję tej relacji inspirowaną definicją podaną w [226].

Dla tablicy decyzyjnej  $DT = (U, A \cup \{d\})$ , podzbioru atrybutów  $B \subseteq A$  oraz obiektów  $x, y \in U$  mówimy, że obiekt  $y$  jest podobny do obiektu  $x$  (co oznaczamy  $yS_Bx$ ) wtedy i tylko wtedy, gdy:

$$\forall a \in B \text{ takiego że } f(a, y) \neq * \text{ zachodzi } f(a, x) = f(a, y) \quad (5.4)$$

Relacja ta nie jest symetryczna ani przechodnia. Można ją także interpretować jako reprezentację relacji zawierania, gdyż podobieństwo  $y$  do  $x$  odpowiada stwierdzeniu, że opis obiektu  $y$  jest zawarty w opisie obiektu  $x$ . W związku z tym, że relacja  $S_B$  nie indukuje podziału na zbiorze  $U$ , klasy relacji  $S_B(x)$  są definiowane dla każdego obiektu  $x \in U$ . Formalnie wprowadzamy definicje dwóch typów klas jako:

$$S_B(x) = \{y \in U : yS_Bx\} \quad (5.5)$$

$$S_B^{-1}(x) = \{y \in U : xS_By\} \quad (5.6)$$

Są to dwa różne zbiory. Klasa  $S_B(x)$  jest zbiorem obiektów podobnych do  $x$ . Klasa  $S_B^{-1}(x)$  jest zbiorem obiektów, do których  $x$  jest podobny. Na podstawie tych zbiorów dolne i górne przybliżenia zbioru obiektów  $X \subseteq U$  definiuje się jako:

$$\underline{B}_S(X) = \{x \in U : S_B^{-1}(x) \subseteq X\} \quad (5.7)$$

$$\overline{B}_S(X) = \bigcup \{S_B(x) : x \in X\} \quad (5.8)$$

Innymi słowy, do dolnego przybliżenia  $X$  zalicza się tylko te obiekty  $x$ , które są podobne wyłącznie do obiektów należących do  $X$ . W podejściu definiuje się także pojęcia jakości przybliżenia klasyfikacji oraz reduktu w podobny sposób jak dla podejścia klasycznego (szczegóły podano w [242, 244]).

Pomimo różnicy w semantyce nieznanych wartości może być interesujące porównanie przybliżeń zbiorów otrzymanych za pomocą podejścia opartego na relacji

tolerancji z przybliżeniami otrzymanymi w wyniku podejścia opartego na relacji podobieństwa. Można sformułować następujące twierdzenie.

**Twierdzenie 5.1**

Dla tablicy decyzyjnej  $DT = (U, A \cup \{d\})$  oraz podzbioru atrybutów  $B \subseteq A$  zachodzi następujący związek pomiędzy przybliżeniami zbioru obiektów  $X \subseteq U$  otrzymanymi za pomocą relacji tolerancji oraz relacji podobieństwa:

$$\underline{B}_T(X) \subseteq \underline{B}_S(X) \text{ oraz } \overline{B}_S(X) \subseteq \overline{B}_T(X) \quad (5.9)$$

■

Dowód tego twierdzenia zamieszczono w [242]. Nieformalnie mówiąc, stosowanie semantyki niedostępnych wartości atrybutów prowadzi do precyzyjniejszych przybliżeń niż w przypadku semantyki brakujących wartości atrybutów.

Reguły decyzyjne są definiowane z wykorzystaniem klas relacji podobieństwa typu  $S_B(x)$ . Przyjmujemy zapis reguły o postaci zgodnej z (2.15), tj. *jeżeli P to Q*, gdzie  $Q = (f(d,x) = v_d^j)$ , natomiast część warunkowa może być zbudowana wyłącznie na podstawie warunków elementarnych odnoszących się do zdefiniowanych wartości atrybutów  $(f(a_i,x) = v_{ai})$ , gdzie  $v_{ai} \neq *$ .

Reguła pewna jest prawdziwa wtedy i tylko wtedy, gdy dla każdego obiektu  $x$  dopasowanego do części warunkowej (w sensie definicji 2.16) zachodzi:

$$S_B(x) \subseteq [Q] \quad (5.10)$$

Wymaga się, aby część warunkowa reguły była minimalna. Powyższy sposób definiowania reguły jest zgodny z właściwościami reguł podanymi w [115, 226]. Uważamy, podobnie jak autorzy pracy [115], że w tym uogólnieniu reguły pewne są bardziej interesujące niż reguły możliwe. Uogólniony opis reprezentowany przez pewne reguły jest oparty na przykładach uczących (obiektach), dla których nie ma podobnych przykładów negatywnych.

Jeśli następnie zadać pytanie o sposoby indukcji takich reguł, to należy stwierdzić, że poszukiwanie minimalnego, satysfakcjonującego lub wyczerpującego zbioru reguł można wykonać w podobny sposób jak omówiono w poprzednich rozdziałach. Zasadnicza modyfikacja wymaga stosowania relacji podobieństwa pomiędzy obiektem a częścią warunkową reguły oraz przestrzegania warunku (5.10) przy ocenie koniunkcji warunków elementarnych.

Przykładowo, w algorytmie poszukiwania minimalnego zbioru reguł (np. w algorytmie LEM2) przy ocenie warunku elementarnego podczas dołączenia go do części warunkowej reguły należy zamiast relacji nierozróżnialności stosować relacje podobieństwa pomiędzy obiektami w  $DT$  a wartościami atrybutów  $(f(a_i,x) = v_{ai})$  w warunkach elementarnych nowo tworzonej koniunkcji. Podkreślimy, że podobieństwo jest badane wyłącznie dla atrybutów istniejących w warunkach rozpatrywanej koniunkcji. W podobny sposób można zmodyfikować strategię poszukiwania zbioru

reguł w algorytmie EXPLORE lub poszukiwania reguł przez redukcję tablicy decyzyjnej za pomocą reduktów względnych związanych z obiektami.

### Kontynuacja przykładu 5.1

Używając zbioru wszystkich atrybutów  $A$ , można znaleźć następujące klasy relacji podobieństwa dla przykładów w tabeli 5.1:

$$S_A^{-1}(x_1) = \{x_1\}, S_A^{-1}(x_2) = \{x_2, x_3\}, S_A^{-1}(x_3) = \{x_2, x_3\}, S_A^{-1}(x_4) = \{x_4, x_5\}, S_A^{-1}(x_5) = \{x_4, x_5\}, S_A^{-1}(x_6) = \{x_6\}, S_A^{-1}(x_7) = \{x_7, x_9\}, S_A^{-1}(x_8) = \{x_8\}, S_A^{-1}(x_9) = \{x_9\},$$

$$S_A^{-1}(x_{10}) = \{x_{10}\}, S_A^{-1}(x_{11}) = \{x_1, x_4, x_5, x_9, x_{11}, x_{12}\}, S_A^{-1}(x_{12}) = \{x_1, x_9, x_{12}\}.$$

$$S_A(x_1) = \{x_1, x_{11}, x_{12}\}, S_A(x_2) = \{x_2, x_3\}, S_A(x_3) = \{x_2, x_3\}, S_A(x_4) = \{x_4, x_5, x_{11}\}, S_A(x_5) = \{x_4, x_5, x_{11}\}, S_A(x_6) = \{x_6\}, S_A(x_7) = \{x_7\}, S_A(x_8) = \{x_8\}, S_A(x_9) = \{x_7, x_9, x_{11}, x_{12}\}, S_A(x_{10}) = \{x_{10}\}, S_A(x_{11}) = \{x_{11}\}, S_A(x_{12}) = \{x_{11}, x_{12}\}.$$

Na tej podstawie klasy decyzyjne  $\Phi$  i  $\Psi$  mogą być przybliżone jako:

$$\underline{A}_S(\Phi) = \{x_1, x_{10}\}, \overline{A}_S(\Phi) = \{x_1, x_2, x_3, x_4, x_5, x_7, x_{10}, x_{11}, x_{12}\}, \underline{A}_S(\Psi) = \{x_6, x_8, x_9\},$$

$$\overline{A}_S(\Psi) = \{x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{11}, x_{12}\}.$$

Z tabeli można wygenerować następujący zbiór reguł decyzyjnych:

jeżeli  $(f(a_1, x) = 1)$  to  $(f(d, x) = \Phi)$

jeżeli  $(f(a_3, x) = 1) \wedge (f(a_4, x) = 0)$  to  $(f(d, x) = \Phi)$

jeżeli  $(f(a_1, x) = 3) \wedge (f(a_4, x) = 0)$  to  $(f(d, x) = \Phi)$

jeżeli  $(f(a_2, x) = 3) \wedge (f(a_4, x) = 1)$  to  $(f(d, x) = \Psi)$

jeżeli  $(f(a_2, x) = 0)$  to  $(f(d, x) = \Psi)$

jeżeli  $(f(a_3, x) = 0)$  to  $(f(d, x) = \Psi)$

## 5.4. Wartościowana relacja tolerancji

### 5.4.1. Motywacje

Rozważmy opis trzech obiektów  $x_1$ ,  $x_{11}$  oraz  $x_{12}$  z przykładu 5.1, stosując ponownie semantykę brakujących wartości atrybutów. Dla ilustracji przedstawiamy go w tabeli 5.2.

**Tabela 5.2.** Przykład ilustrujący różne stopnie podobieństwa obiektów

$U$	$a_1$	$a_2$	$a_3$	$a_4$	$d$
$x_1$	3	2	1	0	$\Phi$
$x_{11}$	*	2	*	*	$\Psi$
$x_{12}$	3	2	1	*	$\Phi$

Użycie dotychczasowego podejścia opartego na relacji tolerancji pozwala stwierdzić, że obiekty  $x_1$  i  $x_{11}$  są do siebie podobne. Podobnie powinniśmy stwierdzić, że



zachodzi podobieństwo pomiędzy obiektami  $x_1$  oraz  $x_{12}$ . Z drugiej strony intuicja wskazuje na różnice w stopniu podobieństwa w tych parach. Moglibyśmy stwierdzić, że podobieństwo obiektu  $x_{12}$  do obiektu  $x_1$  jest większe niż podobieństwo obiektu  $x_{11}$  do obiektu  $x_1$ . Wynika to z faktu, że w przypadku opisu obiektu  $x_{12}$  tylko jedna z wartości atrybutu jest nieznana, a reszta jest równa wartościom atrybutów w opisie  $x_1$ , podczas gdy dla obiektu  $x_{11}$  aż trzy wartości atrybutów są nieznane. W związku z tym, że prosta postać relacji tolerancji nie obejmuje tego zróżnicowania, wydaje się nam naturalne wprowadzenie możliwości wartościowania stopnia podobieństwa pomiędzy obiektami. Zamierzamy tego dokonać za pomocą **wartościowanej relacji tolerancji** (ang. *valued tolerance relation*).

Zauważmy, że można skonstruować różnego typu wartościowane relacje tolerancji (czy podobieństwa). Możliwe jest także budowanie takiej relacji dla w pełni zdefiniowanej tablicy decyzyjnej (porównaj np. propozycje [61, 67]). Jednak dla rozważanego tutaj przypadku niekompletnych tablic decyzyjnych z semantyką brakujących wartości atrybutów przedstawimy własne podejście konstruowania tej relacji wprowadzone w [242]. Uogólnienie tego podejścia pokazano w [243].

#### 5.4.2. Definiowanie przybliżeń zbiorów

Dla każdego obiektu  $x \in U$  można zdefiniować klasę wartościowanej relacji tolerancji  $\theta(x)$ , która jest zbiorem rozmytym zawierającym obiekty wraz funkcjami przynależności odpowiadającymi stopniowi ich podobieństwa do obiektu  $x$ .

Przekładając do rozdziału 5.4.3 przedstawienie definicji tego stopnia, należy zwrócić uwagę że podstawowym problemem jest zaproponowanie sposobu definiowania dolnych i górnych przybliżeń zbioru  $X \subset U$  na podstawie takich klas tolerancji. W omawianej koncepcji proponuje się ciągłe wartościowanie przybliżeń, tzn. dla podzbioru obiektów  $Z \subset U$  będziemy się starali określić **stopień**, w jakim może on być dolnym lub górnym przybliżeniem zbioru  $X$ .

Niektórzy autorzy, np. [43, 61, 67], mieli podobne inspiracje i badali różne sposoby połączenia zbiorów przybliżonych i rozmytych. W ich podejściach dolne i górne przybliżenia były rozpatrywane jako zbiory rozmyte, do których pojedyncze elementy mogły należeć w różnym stopniu. Propozycją najbliższą do rozważanej w tym rozdziale jest rozmyta relacja podobieństwa sformułowana przez Greco i Słowińskiego [61, 67]. Zwróćmy jednak uwagę, że wprowadzana tutaj propozycja, w odróżnieniu od [61], dotyczy niekompletnych tablic decyzyjnych.

Zbiór potęgowy  $2^U$  jest rozważany jako zbiór rozmyty z punktu widzenia dolnego i górnego przybliżenia. Każdy z podzbiorów  $U$  może być dolnym lub górnym przybliżeniem  $X$ , lecz w różnym stopniu. W celu zdefiniowania, jaki jest stopień możliwości, by dowolny podzbiór  $Z \subset U$  został dolnym (górnym) przybliżeniem  $X$ , konieczne jest określenie rozmytych odpowiedników typowych operatorów logicznych negacji, koniunkcji, dysjunkcji i implikacji. Niech w stosowanej poniżej notacji  $m, n$  oznaczają stopnie przynależności.

Negacja jest funkcją  $N : [0,1] \rightarrow [0,1]$  taką, że  $N(0) = 1$  oraz  $N(1) = 0$ . Typową reprezentacją negacji jest  $N(m) = 1 - m$ .

Odpowiednikiem koniunkcji jest  $T$ -norma. Funkcja ta jest ciągła, niemalejąca  $T : [0,1]^2 \rightarrow [0,1]$  taka, że  $T(m,1) = m$ . Typowe realizacje to  $T(m, n) = \min(m, n)$ ,  $T(m, n) = m \cdot n$ ; lub  $T$ -norma Łukasiewicza  $\max(m + n - 1, 0)$ .

Reprezentacją dysjunkcji jest  $T$ -konorma będąca funkcją ciągłą, niemalejącą  $S : [0,1]^2 \rightarrow [0,1]$  taką, że  $S(0, m) = m$ . Typowe realizacje to  $S(m, n) = \max(m, n)$ ;  $S(m, n) = m + n - m \cdot n$ ;  $T$ -konorma Łukasiewicza  $S(m, n) = \min(m + n, 1)$ .

Jeżeli zachodzi  $S(m, n) = N(T(N(m), N(n)))$ , to mamy odpowiednik prawa De Morgana. Niech  $I(m, n)$  oznacza stopień, w jakim  $m$  implikuje  $n$ . Jest to funkcja  $I : [0,1]^2 \rightarrow [0,1]$ . Dyskusja właściwości, które powinny charakteryzować funkcję  $I$ , przedstawiona jest w [44]. W dalszych rozważaniach będziemy stosować postać  $I(m, n) = S(N(m), n)$ .

Odwołując się do podstawowych definicji dolnego i górnego przybliżenia zbioru  $X$ , zapisujemy je w postaci:

$$Z = \underline{B}(X) \Leftrightarrow \forall z \in Z, \theta_B(z) \subseteq X \quad (5.11)$$

$$Z = \overline{B}(X) \Leftrightarrow \forall z \in Z, \theta_B(z) \cap X \neq \emptyset \quad (5.12)$$

gdzie  $\theta_B(z)$  jest klasą wartościowanej relacji tolerancji obiektu  $z$  dla podzbioru atrybutów  $B \subseteq A$ .

Rozważając funkcyjne odpowiedniki logicznych elementów tych definicji:

$$\forall x \varphi(x) = T_x \varphi(x); \quad \exists x \varphi(x) = S_x \varphi(x); \quad \Phi \subseteq \Psi = T_x(I(\mu_\Phi(x), \mu_\Psi(x)));$$

$$\Phi \cap \Psi \neq \emptyset = \exists x \varphi(x) \wedge \psi(x) = S_x(T(\mu_\Phi(x), \mu_\Psi(x)))$$

otrzymujemy następujące definicje:

$$\mu_{\underline{B}(X)}(Z) = T_{z \in Z} (T_{x \in \theta_B(z)} (I(R_B(z, x), \mu_X(x)))) \quad (5.13)$$

$$\mu_{\overline{B}(X)}(Z) = T_{z \in Z} (S_{x \in \theta_B(z)} (T(R_B(z, x), \mu_X(x)))) \quad (5.14)$$

gdzie:

$\mu_{\underline{B}(X)}(Z)$  jest stopniem, w jakim zbiór  $Z$  stanowi dolne przybliżenie zbioru  $X$ ;

$\mu_{\overline{B}(X)}(Z)$  jest stopniem, w jakim zbiór  $Z$  stanowi górne przybliżenie zbioru  $X$ ;

$\theta_B(z)$  jest klasą wartościowanej relacji tolerancji obiektu  $z$ ;

$T, S, I$  są funkcjami uprzednio zdefiniowanymi;

$R_B(z, x)$  jest stopniem przynależności obiektu  $x$  do klasy wartościowanej relacji tolerancji obiektu  $z$  – w przypadku tutaj rozważanym  $R_B(z, x) = T_{ai \in B} R_{ai}(z, x)$  (przykład formuły definiującej  $R_{ai}(z, x)$  jest podany w rozdziale 5.4.3);

$\mu_X(x)$  jest stopniem przynależności obiektu  $x$  do zbioru  $X$ ; przy przyjętym założeniu co do jednoznacznego zdefiniowania wartości atrybutu decyzyjnego  $d$  w tablicy decyzyjnej  $DT$  zachodzi  $\mu_X(x) \in \{0, 1\}$ .

W pracach [243, 244] sformułowano twierdzenia dotyczące podstawowych właściwości stopni (5.13) oraz (5.14).

### 5.4.3. Przykład użycia wartościowanej relacji tolerancji

Wybór funkcji do realizacji operatorów  $N$ ,  $T$ ,  $S$  lub  $I$  zależy od znaczenia, które przypisuje się użyciu relacji tolerancji. Poniżej przedstawiamy przykład wartościowania wynikający z przypisania rozkładów prawdopodobieństwa możliwym wartościom atrybutów [242]. W konsekwencji stosowane operatory będą generowane przez iloczynową wersję  $T$ -normy. Przykład użycia podejścia „posybilistycznego” dla innej semantyki i operatora min w  $T$ -normie przedstawiono w [243].

Założmy, że zbiór możliwych wartości każdego atrybutu jest dyskretny, skończony oraz istnieje równomierny rozkład prawdopodobieństwa przyjęcia przez atrybut poszczególnych wartości. Niech  $a_i$  będzie atrybutem w tablicy decyzyjnej  $DT$ , a zbiór  $V_i = \{v_{ai}^1, \dots, v_{ai}^m\}$  niech będzie zbiorem wszystkich możliwych wartości tego atrybutu. Przyjmujemy, że dla każdego obiektu  $x$  prawdopodobieństwo, że  $f(a_i, x) = v_{ai}^i$  jest równe  $1/|V_i|$ .

Dla dwóch obiektów  $x, y \in U$  i atrybutu  $a_i$ , jeżeli  $f(a_i, y) = v_{ai}^i$  oraz  $f(a_i, x) = *$ , to prawdopodobieństwo, że obiekt  $x$  jest podobny do obiektu  $y$  jest równe  $1/|V_i|$ . Jest to stopień podobieństwa obiektu  $x$  do  $y$ , oznaczony przez  $R_{ai}(x, y)$ . Jednocześnie, jeżeli obie wartości są nieznane, to  $R_{ai}(x, y)$  stopień podobieństwa  $x$  do  $y$  na atrybucie  $a_j$  jest równy  $1/|V_i|^2$ . Jeżeli  $f(a_i, x) = f(a_i, y) = v_{ai}^i$ , to  $R_{ai}(x, y) = 1$ . W ostatnim możliwym przypadku, czyli gdy obie wartości tego atrybutu są znane lecz różne,  $R_{ai}(x, y)$  wynosi 0.

Dla tablicy decyzyjnej  $DT$ , podzbioru atrybutów  $B \subseteq A$  stopień podobieństwa obiektów  $x$  i  $y$  wartościowanej relacji tolerancji  $R_B$  wyrażany jest łącznym prawdopodobieństwem, że obiekty  $x$  i  $y$  mają takie same wartości atrybutów, tj.:

$$R_B(x, y) = \prod_{ai \in B} R_{ai}(x, y) \quad (5.15)$$

#### Kontynuacja przykładu 5.1

Rozważmy ponownie zbiór obiektów z tabeli 5.1, pamiętając, że dziedzina każdego atrybutu zawiera zbiór czterech wartości  $\{0, 1, 2, 3\}$ . Prawdopodobieństwo tego, że obiekt  $x_{12}$  jest podobny do  $x_1$ , jest równe  $1/4$ , co wynika z  $R_{a1}(x_{12}, x_1) = 1$ ,  $R_{a2}(x_{12}, x_1) = 1$ ,  $R_{a3}(x_{12}, x_1) = 1$ ,  $R_{a4}(x_{12}, x_1) = 1/4$ . Podobnie można obliczyć  $R_A(x_{11}, x_1) = 1/64$ , co wynika z  $R_{a1}(x_{11}, x_1) = 1/4$ ,  $R_{a2}(x_{11}, x_1) = 1$ ,  $R_{a3}(x_{11}, x_1) = 1/4$ ,  $R_{a4}(x_{11}, x_1) = 1/4$ . Oznacza to, że obiekt  $x_{12}$  jest w większym stopniu podobny do  $x_1$  niż obiekt  $x_{11}$ , co jest zgodne z motywacją przedstawioną w rozdziale 5.4.1. Zastosowanie formuły (5.15) do wszystkich par obiektów prowadzi do określenia relacji tolerancji przedstawionej w tabeli 5.3.

**Tabela 5.3.** Wartościowana relacja tolerancji dla obiektów z przykładu 5.1

Ob.	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$
$x_1$	1	0	0	0	0	0	0	0	0	0	1/64	1/4
$x_2$	0	1	1	0	0	0	0	0	0	0	0	0
$x_3$	0	1	1	0	0	0	0	0	0	0	0	0
$x_4$	0	0	0	1	1/256	0	0	0	0	1/1024	1/1024	1/64
$x_5$	0	0	0	1/256	1	0	0	0	0	1/1024	1/1024	1/64
$x_6$	0	0	0	0	0	1	0	0	0	0	0	0
$x_7$	0	0	0	0	0	0	1	1/256	1/16	0	1/1024	1/64
$x_8$	0	0	0	0	0	0	1/256	1	0	1/1024	0	0
$x_9$	0	0	0	0	0	0	1/16	0	1	0	1/64	1/4
$x_{10}$	0	0	0	1/1024	1/1024	0	0	1/1024	0	1	1/4096	0
$x_{11}$	1/64	0	0	1/1024	1/1024	0	1/1024	0	1/64	1/4096	1	1/256
$x_{12}$	1/4	0	0	1/64	1/64	0	1/64	0	1/4	0	1/256	1

Wiersze w powyższej tabeli można traktować jako wektory wartości funkcji przynależności do klas wartościowanej relacji tolerancji  $\theta_B(x)$  poszczególnych obiektów. Wektor  $[1,0,0,0,0,0,0,0,0,0,1/64,1/4]$  odpowiada klasie dla obiektu  $x_1$ . Jedynie obiekty  $x_{11}$  i  $x_{12}$  są podobne do  $x_1$  w stopniu większym od zera. Stosując podejście „probabilistyczne”, stosujemy następujące reprezentacje operatorów:

$$T(m, n) = mn;$$

$$S(m, n) = m + n - mn;$$

$$I(m, n) = 1 - m + mn.$$

$$N(m) = 1 - m$$

W konsekwencji stopnie, w jakich zbiór  $Z \subset U$  może być dolnym lub górnym  $B$ -przybliżeniem zbioru  $X \subset U$ , są zdefiniowane następująco:

$$\mu_{\underline{B}(X)}(Z) = \prod_{z \in Z} \prod_{x \in \theta_B(z)} (1 - R_B(z, x) + R_B(z, x) \cdot \mu_X(x)) \quad (5.16)$$

$$\mu_{\overline{B}(X)}(Z) = \prod_{z \in Z} (1 - \prod_{x \in \theta_B(z)} (1 - R_B(z, x) \cdot \mu_X(x))) \quad (5.17)$$

Stopnie, w jakich pojedynczy obiekt  $z \in U$  może stanowić dolne lub górne przybliżenie zbioru  $X$ , jest zdefiniowany następująco:

$$\mu_{\underline{B}(X)}(z) = \prod_{x \in \theta(z)} (1 - R_B(z, x) + R_B(z, x) \cdot \mu_B(x)) \quad (5.18)$$

$$\mu_{\overline{B}(X)}(z) = (1 - \prod_{x \in \theta_B(z)} (1 - R_B(z, x) \cdot \mu_B(x))) \quad (5.19)$$

Przykładowo dla obiektu  $x_1$  mamy  $\mu_{\underline{A}(\Phi)}(x_1) = (1 - 1 - 1 \cdot 1)(1 - 1/64 + (1/64) \cdot 0) \cdot (1 - 1/4 + (1/4) \cdot 1) = 0,984$ . Podobnie  $\mu_{\overline{A}(\Phi)}(x_1) = (1 - (1 - 1 \cdot 1) \cdot (1 - (1/64) \cdot 0)) \cdot (1 - (1/4))$ .

$\cdot 1) = 1$ . W podobny sposób można obliczyć dla reszty obiektów stopnie, w jakich są dolnym lub górnym przybliżeniem. Wyniki zawiera tabela 5.4.

**Tabela 5.4.** Stopnie, w jakich pojedyncze obiekty mogą stanowić dolne i górne przybliżenia klas  $\Psi$  i  $\Phi$

Stopień	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$
$\mu_{\underline{A}(\Phi)}(z)$	0,984	0	0	0,995	0	0	0,933	0	0	0,998	0	0,735
$\mu_{\overline{A}(\Phi)}(z)$	1	1	1	1	0,02	0	1	0,005	0,297	1	0,022	1
$\mu_{\underline{A}(\Psi)}(z)$	0	0	0	0	0,98	1	0	0,995	0,703	0	0,978	0
$\mu_{\overline{A}(\Psi)}(z)$	0,016	1	1	0,005	1	1	0,067	1	1	0,002	0	0,265

Przykładem obliczenia zdolności podzbioru obiektów  $Z$  do przybliżenia zbioru  $\Phi$  są  $\mu_{\underline{A}(\Phi)}(\{x_1, x_4\}) = 0,984 \cdot 0,995 = 0,979$ ,  $\mu_{\underline{A}(\Phi)}(\{x_1, x_7\}) = 0,984 \cdot 0,933 = 0,918$ ,  $\mu_{\underline{A}(\Phi)}(\{x_1, x_{12}\}) = 0,984 \cdot 0,735 = 0,72$ ,  $\mu_{\underline{A}(\Phi)}(\{x_1, x_4, x_{10}\}) = 0,984 \cdot 0,995 \cdot 0,998 = 0,974$  itd.

Zauważmy, że w odróżnieniu od poprzednich podejść, każdy podzbiór obiektów może być dolnym lub górnym przybliżeniem, lecz w różnym stopniu. Naturalne jest umożliwienie użytkownikowi akceptacji tylko niektórych podzbiorów poprzez zdefiniowanie progu  $\lambda$ . Próg ten wyraża minimalny akceptowany przez użytkownika stopień dla dolnego (górnego) przybliżenia. Technicznie tworzenie rodziny zbiorów mogących być dolnym przybliżeniem może przebiegać według poniższego schematu (postępowanie dla górnych przybliżeń jest analogiczne).

1. Wybierz wszystkie obiekty  $z$ , dla których  $\mu_{\underline{B}(X)}(z) = 1$ .
2. Dodawaj kolejne obiekty  $z$  wartościami stopni  $\mu_{\underline{B}(X)}(z) > k$ , dla malejących wartości  $k$ , np. 0,99, 0,98, ... Otrzymasz w ten sposób rodzinę podzbiorów  $Z$  wraz z malejącymi wartościami  $\mu_{\underline{B}(X)}(Z)$ .
3. Dla ustalonego progu  $\lambda$  zaakceptuj jako  $B$ -dolne przybliżenia tylko te zbiory  $Z$ , dla których  $\mu_{\underline{B}(X)}(Z) \geq \lambda$ .

Dalsza dyskusja na temat budowy przybliżeń przedstawiona jest w [244]. W powyższym schemacie postępowania otrzymuje się rodzinę zbiorów obiektów, które mogą być dolnymi przybliżeniami w akceptowalnym stopniu. W części zastosowań użytkownik może być zainteresowany wyborem tylko jednego zbioru. Wtedy rozwiązaniem jest wybór jednego, najliczniejszego (w sensie liczby obiektów) zbioru, dla którego  $\mu_{\underline{B}(X)}(Z) \geq \lambda$ .

#### 5.4.4. Reguły decyzyjne

W omawianym podejściu należy przyjąć, że z każdą regułą decyzyjną jest związany współczynnik wiarygodności wynikający z faktu, że obiekty mogą być podobne do części warunkowej reguły tylko w pewnym stopniu, oraz że sama implikacja może być niepewna. Poniżej przedstawimy funkcyjny odpowiednik reprezentacji reguły oparty na wyrażeniu (2.14).

Niech  $r^C$  oznacza regułę „jeżeli  $Q$  to  $P$ ”,  $C$  – podzbiór atrybutów występujących w części warunkowej reguły;  $\rho$  jest „fikcyjnym obiektem”, którego opis odpowiada warunkom elementarnym z części warunkowej reguły. Użyjemy wartościowanej relacji tolerancji  $R_C(x, \rho)$ , aby określić stopień, w jakim obiekt  $x$  jest podobny do części warunkowej reguły (obiektu  $\rho$ ) i w konsekwencji może ją spełniać. Przez  $S(r) = \{x : R_C(x, \rho) > 0\}$  oznaczamy będziemy zbiór obiektów podobnych do części warunkowej reguły. Dla reguły pewnej w przypadku nierozmytym spełniony jest warunek:

$$\forall x \in S(r) \quad \theta_C(x) \subseteq [Q] \quad (5.20)$$

Jest to odpowiednik wyrażenia (2.18). Poszukując transformacji wyrażenia (5.20) dla przypadku wartościowanego, dochodzi się do wyrażenia [242]:

$$\mu(r^C) = T_{x \in S(r)}(I(R_C(x, \rho), T_{y \in \theta_C(x)}(I(\mu_{\theta_C(x)}(y), \mu_D(y)))))) \quad (5.21)$$

gdzie  $D$  oznacza klasę decyzyjną wskazywaną przez regułę,  $\mu_D(y)$  stopień przynależności obiektu  $y$  do zbioru  $D$  przyjmujący tylko dwie wartości  $\{0, 1\}$ , a  $\mu_{\theta_C(x)}(y) = R_C(x, y)$ . Stopień  $\mu(r^C)$  nazywać będziemy współczynnikiem wiarygodności reguły decyzyjnej  $r^C$ .

Wymaga się także, aby część warunkowa reguły nie była nadmiarowa, tzn. nie może istnieć podzbiór atrybutów  $\hat{C} \subset C$  spełniający warunek  $\mu(r^{\hat{C}}) \geq \mu(r^C)$ .

W pracy [243] przedstawiono dowód następującego twierdzenia:

#### Twierdzenie 5.2

Dana jest reguła  $r^C$  przydzielająca obiekty do klasy decyzyjnej  $D$  i wykorzystująca w części warunkowej podzbiór atrybutów  $C$ . Jeżeli  $T, S, I$  spełniają prawo De-Morgana i  $R_C$  jest wartościowaną relacją tolerancji, to współczynnik wiarygodności  $\mu(r^C)$  jest z góry ograniczony najniższą wartością stopnia  $\mu_{\underline{C}(D)}(x)$  definiowania dolnego przybliżenia zbioru  $D$  przez obiekt  $x$ , którego opis (z uwagi na atrybuty  $C$ ) jest identyczny z częścią warunkową reguły. ■

Wynik tego twierdzenia jest przydatny do indukcji reguł decyzyjnych. Można założyć, że użytkownik jest zainteresowany regułami o odpowiednio wysokim współczynnikiem wiarygodności. Oznacza to, że może on zdefiniować pewien próg  $\tau$

i nie akceptować reguł z  $\mu(r^C) < \tau$ . Oczywiście możliwe jest wykonanie analizy wrażliwości wynikowego zbioru reguł wokół pewnej wartości progu  $\tau$ .

W odniesieniu do indukcji reguł, dla których część warunkowa odpowiada redukcji opisu pewnych obiektów w tablicy decyzyjnej  $DT$ , można podać kilka uwag ułatwiających projektowanie algorytmu indukcji zbioru wszystkich reguł:

1. Załóżmy, że użytkownik zdefiniował wartość progu akceptacji współczynnika wiarygodności reguł  $\tau$ .
2. Wybierając obiekty, na podstawie których będą tworzone części warunkowe reguł, wystarczy rozważać tylko obiekty ze stopniem  $\mu_{C(D)}(x) \geq \tau$  (wynika to z twierdzenia 5.2). Inne obiekty można pominąć. Dla wybranych obiektów podczas tworzenia kandydatów na warunki elementarne należy rozważać tylko w pełni zdefiniowane atrybuty.
3. Problem poszukiwania nienadmiarowych części warunkowych odpowiada poszukiwaniu reduktu (lub reduktów) względnych związanych z obiektem.

### Kontynuacja przykładu 5.1

Używając wybranych realizacji operatorów wartościowanej relacji tolerancji, otrzymujemy następującą definicję współczynnika wiarygodności reguły  $r^C$ :

$$\mu(r^C) = \prod_{x \in S(r)} (1 - R_C(x, \rho) + R_C(x, \rho) \prod_{y \in \theta_C(x)} (1 - \mu_{\theta_C(x)}(y) + \mu_{\theta_C(x)}(y) \cdot \mu_D(y)))$$

Założmy, że próg akceptacji reguł  $\tau$  ma wartość 0,9. Zgodnie z powyższymi uwagami i rezultatami zamieszczonymi w tabeli 5.4 dla klasy  $\Phi$  wystarcza rozważać jako podstawę do tworzenia warunków elementarnych wyłącznie opisy obiektów  $x_1, x_4, x_7, x_{10}$ . Wartościowana relacja tolerancji określona wyłącznie dla zdefiniowanych atrybutów opisujących obiekty prowadzi do wyboru tylko obiektu  $x_1$ , gdyż dla pozostałych obiektów wartości stopnia realizacji dolnego przybliżenia obniżą się poniżej progu  $\tau$ . Podobne rozważanie można przeprowadzić dla klasy  $\Psi$ , gdzie opisy obiektów  $x_5, x_6, x_8, x_{11}$  mogą być podstawą tworzenia warunków elementarnych.

Stosując podejście generowania wszystkich reguł, można wygenerować następujący zbiór reguł:

*jeżeli*  $(f(a_{1,x}) = 3) \wedge (f(a_{3,x}) = 1) \wedge (f(a_{4,x}) = 0)$  *to*  $(f(d,x) = \Phi)$  *z*  $\mu(r) = 0,905$

*jeżeli*  $(f(a_{1,x}) = 2) \wedge (f(a_{2,x}) = 3) \wedge (f(a_{4,x}) = 1)$  *to*  $(f(d,x) = \Psi)$  *z*  $\mu(r) = 1,0$

*jeżeli*  $(f(a_{1,x}) = 2) \wedge (f(a_{4,x}) = 1)$  *to*  $(f(d,x) = \Psi)$  *z*  $\mu(r) = 0,931$

*jeżeli*  $(f(a_{2,x}) = 3) \wedge (f(a_{3,x}) = 2) \wedge (f(a_{4,x}) = 1)$  *to*  $(f(d,x) = \Psi)$  *z*  $\mu(r) = 0,969$

Szczegóły otrzymania tego zbioru przedstawione są w pracy [244]. Jak łatwo zauważyć, jest to liczniejszy zbiór reguł niż w przypadku użycia prostej relacji tolerancji.

Reguły decyzyjne mogą być użyte zarówno w perspektywie opisu, jak i klasyfikowania w odkrywaniu wiedzy z niekompletnych tablic decyzyjnych.

W drugim przypadku konieczne jest użycie reguł do przewidywania klasyfikacji nowego obiektu. Podjęcie decyzji klasyfikacyjnej jest związane z dwoma źródłami niepewności. Po pierwsze, klasyfikowany obiekt może być podobny do części warunkowej  $\rho$  tylko w pewnym stopniu. Po drugie, reguła  $r^C$  może posiadać wiarygodność o współczynniku mniejszym niż jeden. Ponadto nowy obiekt może być w różnym stopniu podobny do wielu reguł wskazujących różne decyzje – odpowiada to problemowi wielokrotnego dopasowania omówionemu w rozdziale 2.3. Poniżej przedstawiamy jeden z możliwych sposobów rozwiązania tego problemu w warunkach użycia wartościowanej relacji tolerancji, który opiera się na wyborze najbardziej wiarygodnej reguły decyzyjnej:

1. Dla każdej reguły decyzyjnej  $r^C$  oblicz stopień podobieństwa nowego obiektu  $e$  do jej części warunkowej  $R_C(e, \rho)$
2. Oblicz stopień przynależności obiektu  $e$  do każdej z klas decyzyjnych  $D$  na podstawie podobieństwa do reguły  $r^C$  jako  $\mu_D(e) = T(R_C(e, \rho), \mu(r^C))$  – wybierz dla każdej klasy regułę o największej wartości tego wyrażenia.
3. Przydziel obiekt  $e$  do klasy decyzyjnej  $D$  o największej wartości  $\mu_D(e)$ .
4. Jeśli wystąpi niejednoznaczność w przydziale (więcej niż jedna klasa posiada regułę o najwyższej wartości  $\mu_D(e)$ ), to wybierz regułę o największej liczbie wspierających obiektów z  $S(r)$ .

#### 5.4.5. Dyskusja użycia wartościowanej relacji tolerancji

W rozdziale 5 przedstawiono podejście do uwzględniania nieznanymi wartości atrybutów za pomocą wartościowanej relacji tolerancji. Uważamy, że zaproponowane podejście jest bardziej „elastyczne” z punktu widzenia użytkownika oraz dostarcza więcej informacji niż podejście oparte na prostej relacji tolerancji. Przykład ilustracyjny 5.1 wskazuje, że stosując to podejście, można było zbudować precyzyjniejsze przybliżenia oraz liczniejszy zbiór reguł decyzyjnych. Z drugiej strony użycie wartościowanej relacji tolerancji wymaga większych kosztów obliczeniowych. W pracy [244] przeprowadzono eksperymentalne studium porównawcze użycia obu podejść do analizy danych o różnych rozmiarach i o różnej liczbie nieznanymi wartości atrybutów. Wykorzystywano jeden zbiór danych, w którym symulowano 7 różnych rozkładów nieznanymi wartości oraz 5 rzeczywistych zbiorów danych pochodzących z [14]. W przypadku podejścia opartego na wartościowanej relacji tolerancji systematycznie badano szereg różnych progów akceptacji dolnych (górnym) przybliżeń  $\lambda$  oraz wiarygodności reguł  $\tau$ . Otrzymane eksperymentalne wyniki porównania obu podejść wskazują, że dla obu progów mniejszych od 1 proponowane podejście prowadziło także do precyzyjniejszych przybliżeń (liczniejsze dolne oraz mniej liczne górne przybliżenia) oraz do liczniejszych zbiorów reguł. W szczególności



możliwe było odkrycie reguł decyzyjnych w tych zbiorach, dla których zastosowanie relacji tolerancji nie umożliwiło znalezienia żadnej reguły pewnej.

W przykładzie ilustracyjnym i w eksperymentach wykorzystano poszukiwanie wszystkich reguł decyzyjnych poprzez redukcję opisów wybranych obiektów w tablicy decyzyjnej. Jest to kosztowna obliczeniowo opcja indukcji reguł, choć twierdzenie oraz uwagi podane w rozdziale 5.4.4 zmniejszają liczbę obiektów do sprawdzenia. W dalszych kierunkach badań pożądane jest zaproponowanie algorytmu indukcji minimalnego zbioru reguł.

## 6. Reguły decyzyjne dla wielokryterialnego problemu sortowania

### 6.1. Teoria zbiorów przybliżonych i reguły decyzyjne w wielokryterialnych problemach decyzyjnych

Nowym kierunkiem badawczym jest zastosowanie teorii zbiorów przybliżonych oraz metod indukcji reguł decyzyjnych do analizy tablic decyzyjnych zawierających informacje dotyczące wielokryterialnych problemów decyzyjnych.

W ogólności celem analizy decyzyjnej jest wyjaśnienie sytuacji decyzyjnej lub przygotowanie zaleceń, jak podejmować decyzje w określonych warunkach [192, 193]. Pierwsze z zagadnień dotyczy wyjaśnienia tych elementów podjętych poprzednio decyzji, które nie są oczywiste dla uczestników procesu decyzyjnego. Natomiast zalecenia mogą być formułowane na podstawie różnych **modeli preferencji**, w szczególności reguł decyzyjnych. Zgodnie z klasyfikacją przedstawioną przez Roya [192] trzy podstawowe problemy decyzyjne to: **wybór** najlepszego wariantu decyzyjnego, **ranking** wariantów oraz **sortowanie** wariantów do wcześniej znanych kategorii decyzyjnych.

Decyzje oparte są na znajomości ocen wariantów decyzyjnych (obiektów). Oceny wariantów mogą być interpretowane jako wartości atrybutów. Atrybuty, których dziedziny są uporządkowane według preferencji, nazywają się **kryteriami**. Przykładowo decyzje dotyczące samochodów mogą być oparte na cenie, prędkości maksymalnej, zużyciu paliwa, kolorze, kraju produkcji itp. Takie cechy jak cena, prędkość czy zużycie paliwa mogą być interpretowane jako kryteria, gdyż decydenci (użytkownicy samochodów) zwykle preferują niższą cenę od wyższej, większą prędkość od niższej czy wolą mniejsze zużycie paliwa. W sytuacjach, kiedy na podjęcie decyzji wpływa wiele kryteriów mówimy o **wielokryterialnych problemach decyzyjnych** [192]. Cechą charakterystyczną i trudnością problemów wielokryterialnych jest to, że kryteria mogą wyrażać konfliktowe punkty widzenia. Zauważmy ponadto, że decyzje mogą mieć również charakter porządkowy w zależności od tego, czy wyrażają pewne preferencje.

W wielokryterialnej analizie decyzyjnej stosowano dotychczas dwie główne klasy modeli preferencji: funkcyjne i relacyjne. Modele funkcyjne są używane na przykład w kontekście funkcji użyteczności [103]. Przykładem modelu relacyjnego jest relacja przewyższania [192]. Modele te wymagają jednak wyspecyfikowania przez użytkownika bardzo wielu informacji preferencyjnych. Alternatywnym modelem zyskującym rosnące zainteresowanie w ostatnich latach jest model wyrażania preferencji za pomocą reguł decyzyjnych. Zgodnie z [228] ludzie często poszukują reguł, które uzasadniają podejmowane przez nich wybory. Eksploatacja modelu regułowego może prowadzić do wypracowania przekonujących zaleceń dla decydentów [63, 68].

Formalne badanie zależności pomiędzy modelami funkcyjnym, relacyjnym a regułowym wskazuje na dużą przydatność modelu regułowego w wielokryterialnej analizie decyzyjnej [69, 70]. Reguły mogą być pozyskiwane od ekspertów poprzez odpowiednie procedury dialogu [9, 127] lub w wyniku analizy przykładów podejmowania decyzji [63]. W drugim przypadku przykłady są reprezentowane w tablicy decyzyjnej *DT* lub tablicy przetworzonej do postaci porównań wariantów parami – dotyczy to problematyki rankingu czy wyboru [59]. W dalszych rozważaniach będziemy się zajmować pierwszą z reprezentacji. W takiej tablicy decyzyjnej obiekty mogą być opisane zarówno kryteriami, jak i atrybutami; klasy decyzyjne mogą być także uporządkowane według preferencji.

Jedną z trudności w analizie decyzyjnej jest występowanie niespójnej informacji o sytuacjach decyzyjnych [59, 63, 224]. W tablicy decyzyjnej oznacza to występowanie niespójności w opisie przykładów. Należy podkreślić, że niespójności nie mogą być rozpatrywane jako „szum” czy błąd, co, jak zwracano uwagę w rozdziale 2, jest typowym podejściem w niektórych metodach uczenia maszynowego. W kontekście analizy decyzyjnej takie niespójności mogą odzwierciedlać nieprecyzyjność stosowanego modelu, wahanie decydenta czy niestabilności w wyrażaniu przez niego modelu preferencji. W związku z tym niespójności nie powinny być agregowane lub pomijane, lecz celowa jest raczej ich separacja w celu wyprowadzenia pewnych i możliwych konkluzji z dostępnej informacji. Powyższe motywacje prowadzą do rozważania użycia teorii zbiorów przybliżonych do indukcji reguł decyzyjnych na potrzeby wielokryterialnej analizy decyzyjnej [172].

Jak jednak wskazano w [59, 63], oryginalna teoria zbiorów przybliżonych [110, 165] nie może być zastosowana do wielokryterialnych problemów decyzyjnych, gdyż nie uwzględnia kryteriów, których dziedziny są uporządkowane według preferencji. Jest więc ograniczona tylko do problemów wieloatrybutowej klasyfikacji [172]. W przypadku analizy przykładów reprezentujących wielokryterialne problemy teoria nie pozwala odkryć drugiego typu niespójności informacji wynikających z naruszenia fundamentalnej **zasady dominacji** w zbiorze wariantów. Zasada dominacji mówi, że obiekty posiadające lepsze (lub przynajmniej nie gorsze) oceny nie mogą być przydzielone do gorszej kategorii.

Greco, Matarazzo i Słowiński zaproponowali [59] uogólnienie teorii zbiorów przybliżonych (ang. *Dominance-based Rough Sets Approach* – DRSA) na potrzeby wielokryterialnych problemów decyzyjnych. Główną ideą jest zastąpienie relacji nierozróżnialności za pomocą **relacji dominacji**. Uogólnienie to doprowadziło do powstania licznych prac nad różnymi zagadnieniami analizy decyzyjnej [60–71],

W dalszych rozważaniach skoncentrujemy się nad jednym z podstawowych problemów decyzyjnych, tj. nad **wielokryterialnym problemem sortowania**. Polega on na przydziale obiektów opisanych za pomocą zbioru kryteriów do jednej z klas decyzyjnych. Klasy decyzyjne są uporządkowane preferencyjnie. Problem sortowania jest odpowiednikiem wieloatrybutowego problemu klasyfikacji.

W uogólnieniu teorii zbiorów przybliżonych (DRSA) opartym na relacji dominacji rozważa się reguły decyzyjne o uogólnionej składni. Warunki elementarne w części warunkowej reguł reprezentują stwierdzenie „ocena obiektu na kryterium  $a$  jest co najmniej (lub co najwyżej) tak dobra jak pewna wartość  $r_a$ ”. Ponadto części decyzyjne wskazują, że obiekt należy co najmniej (lub co najwyżej) do danej klasy decyzyjnej.

W zakresie tego uogólnienia autor rozprawy, współpracując z Greco i Słowińskim, zaproponował algorytmy indukcji uogólnionych reguł, tj.:

- algorytm DOMLEM przeznaczony do indukcji minimalnego zbioru reguł [71],
- algorytm DOMApriori przeznaczony do indukcji satysfakcjonującego zbioru reguł [238].

Rozważano także problematykę indukcji wyczerpującego zbioru reguł.

Analiza dużych zbiorów danych wskazała także, że dla niektórych problemów wielokryterialnego sortowania niespójności z zasadą dominacji obiektów mogą silnie obniżać liczebność dolnych przybliżeń i w konsekwencji uniemożliwiać odkrycie silnych reguł decyzyjnych wspartych dużą liczbą przykładów. Dlatego autor wspólnie z Greco i Słowińskim zaproponował kolejną modyfikację stosowanego uogólnienia, która akceptuje występowanie w dolnych przybliżeniach klas decyzyjnych ograniczonej liczby przykładów negatywnych, co jest kontrolowane za pomocą **współczynnika spójności** (ang. *consistency level*). Nowe uogólnienie nazywane jest opartym na relacji dominacji modelem zbiorów przybliżonych o zmiennej spójności (ang. *Variable consistency model of dominance-based rough sets approach* – VC-DRSA) [70].

W niniejszym rozdziale omówiono problematykę reguł decyzyjnych w uogólnieniu teorii zbiorów przybliżonych opartym na dominacji, a także jego rozszerzenie z modelem o zmiennym współczynniku spójności. Przedstawiono algorytmy indukcji reguł DOMLEM i DOMApriori oraz pokazano ich modyfikacje dla modelu VC-DRSA.

## 6.2. Teoria zbiorów przybliżonych oparta na relacji dominacji

Przykłady decyzji o sortowaniu wariantów decyzyjnych reprezentowane są w tablicy decyzyjnej  $DT$  (definicja  $DT$  w rozdziale 2.1.1). Zakładamy, że wszystkie atrybuty  $a \in A$  są kryteriami; wersja podejścia zawierająca także atrybuty będzie krótko omówiona w rozdziale 6.5 (porównaj także [62]).

Niech  $S_a$  będzie **relacją przewyższania** (ang. *outranking relation*) [192] określoną na zbiorze obiektów  $U$  ze względu na kryterium  $a$  taką, że  $xS_a y$  oznacza: „obiekt  $x$  jest przynajmniej tak dobry jak obiekt  $y$  ze względu na kryterium  $a$ ”.

Niech atrybut decyzyjny  $d$  określa podział  $U$  na skończoną liczbę klas decyzyjnych  $CI = \{Cl_t, t \in T\}$ ,  $T = \{1, \dots, n\}$ . Zakładamy, że klasy są uporządkowane, tj. dla wszystkich  $r, s \in T$ , takich że  $r > s$ , obiekty z  $Cl_r$  są bardziej preferowane od obiektów z  $Cl_s$ . Formalnie  $\forall (x, y) \in U: [x \in Cl_r, y \in Cl_s, r > s] \Rightarrow [xSy \text{ oraz } \neg ySx]$ .

Zbiory, które się przybliża, nazywane są **skumulowanymi klasami decyzyjnymi** lub inaczej złoženiami. Wyróżniamy skumulowaną klasę decyzyjną skierowaną ku górze (ang. *upward union*):

$$Cl_t^{\geq} = \bigcup_{s \geq t} Cl_s, \quad t = 1, \dots, n \quad (6.1)$$

oraz skumulowaną klasę decyzyjną skierowaną ku dołowi (ang. *downward union*)

$$Cl_t^{\leq} = \bigcup_{s \leq t} Cl_s, \quad t = 1, \dots, n \quad (6.2)$$

Zauważmy, że  $Cl_1^{\geq} = Cl_n^{\leq} = U$ ,  $Cl_n^{\geq} = Cl_n$ ,  $Cl_1^{\leq} = Cl_1$ . Ponadto dla  $t = 2, \dots, n$ , zachodzi:  $Cl_{t-1}^{\leq} = U - Cl_t^{\geq}$  oraz  $Cl_t^{\geq} = U - Cl_{t-1}^{\leq}$ .

Mówimy, że obiekt  $x$  **dominuje** obiekt  $y$  ze względu na podzbiór kryteriów  $B \subseteq A$ , co oznacza się przez  $x D_B y$ , jeżeli  $\forall a \in B \quad x S_a y$ . Relacja dominacji jest zwrotna i przechodnia. Dla podzbioru kryteriów  $B \subseteq A$  oraz obiektu  $x \in U$  „zbiory elementarne” używane w podejściu DRSA do budowy przybliżeń są następujące:

– zbiór obiektów dominujących  $x$ , nazywany zbiorem  $B$ -dominującym:

$$D_B^+(x) = \{y \in U : y D_B x\}, \quad (6.3)$$

– zbiór obiektów zdominowanych przez  $x$ , nazywany zbiorem  $B$ -dominowanym:

$$D_B^-(x) = \{y \in U : x D_B y\}. \quad (6.4)$$

Geometrycznie, zbiory  $D_B^+(x)$  i  $D_B^-(x)$  przedstawiają stożki wielościenne w  $B$ -wymiarowej przestrzeni ocen obiektów, których początki osadzone są w punkcie  $x$ , przy czym stożek odpowiadający  $D_B^+(x)$  jest otwarty w stronę lepszych wartości kryteriów niż  $x$ , a  $D_B^-(x)$  jest otwarty w stronę gorszych wartości kryteriów niż  $x$ .

$B$ -dolnym przybliżeniem skumulowanej klasy  $Cl_t^{\geq}$ , oznaczonym przez  $\underline{B}(Cl_t^{\geq})$ , i  $B$ -górnym przybliżeniem  $Cl_t^{\geq}$ , oznaczonym przez  $\overline{B}(Cl_t^{\geq})$  nazywamy zbiory:

$$\underline{B}(Cl_t^{\geq}) = \{x \in U : D_B^+(x) \subseteq Cl_t^{\geq}\}, \quad \overline{B}(Cl_t^{\geq}) = \bigcup_{x \in Cl_t^{\geq}} D_B^+(x), \quad \text{dla } t = 1, \dots, n \quad (6.5)$$

Podobnie, używając  $D_B^-(x)$ , można zdefiniować  $B$ -dolne przybliżenie oraz  $B$ -górne przybliżenie skumulowanej klasy  $Cl_t^{\leq}$ :

$$\underline{B}(Cl_t^{\leq}) = \{x \in U : D_B^-(x) \subseteq Cl_t^{\leq}\}, \quad \overline{B}(Cl_t^{\leq}) = \bigcup_{x \in Cl_t^{\leq}} D_B^-(x), \quad \text{dla } t = 1, \dots, n \quad (6.6)$$

$B$ -granice  $Cl_t^{\geq}$  oraz  $Cl_t^{\leq}$  są zdefiniowane jako:

$$Bn_B(Cl_t^{\geq}) = \overline{B}(Cl_t^{\geq}) - \underline{B}(Cl_t^{\geq}), \quad Bn_B(Cl_t^{\leq}) = \overline{B}(Cl_t^{\leq}) - \underline{B}(Cl_t^{\leq}), \quad t = 1, \dots, n \quad (6.7)$$

Inne pojęcia teorii zbiorów przybliżonych, jak jakość przybliżenia klasyfikacji, redukty, uogólnia się także w podejściu opartym na relacji dominacji [58, 61, 67].

Przybliżenia skierowanych ku górze i ku dołowi skumulowanych klas decyzyjnych służą do generacji reguł decyzyjnych. Dla skumulowanej klasy  $Cl_t^{\geq}$  (lub  $Cl_s^{\leq}$ ),  $s, t \in T$ , reguły pewne są indukowane na podstawie obiektów z  $\underline{B}(Cl_t^{\geq})$  (lub  $\underline{B}(Cl_s^{\leq})$ ) traktowanych jako przykłady pozytywne, podczas gdy reszta obiektów jest traktowana jako przykłady negatywne dla algorytmu indukcji reguł. Te reguły nazywane są **pewnymi**  $D_{\geq}$ - (lub  $D_{\leq}$ )-regułami decyzyjnymi, gdyż wskazują jednoznacznie przydział obiektu do „co najmniej klasy  $Cl_t$ ” (lub „co najwyżej klasy  $Cl_s$ ”). **Przybliżone**  $D_{\geq\leq}$ -reguły decyzyjne są indukowane przy założeniu, że obiekty należące do  $\overline{B}(Cl_s^{\leq}) \cap \overline{B}(Cl_t^{\geq})$  ( $s < t$ ) są przykładami pozytywnymi, a reszta obiektów to przykłady negatywne. Alternatywnie można dokonać indukcji  $D_{\geq}$ - (lub  $D_{\leq}$ )-**możliwych** reguł decyzyjnych przy założeniu, że obiekty należące do  $\underline{B}(Cl_t^{\leq})$  (lub do  $\overline{B}(Cl_s^{\leq})$ ) są przykładami pozytywnymi, a reszta obiektów to przykłady negatywne. Takie reguły wskazują, że obiekt może należeć do „co najmniej klasy  $Cl_t$ ” (lub „co najwyżej klasy  $Cl_s$ ”).

Zakładając, że dla każdego kryterium  $a \in A$ ,  $V_a \subseteq \mathbb{R}$  oraz  $\forall x, y \in U$ ,  $f(a, x) \geq f(a, y)$  implikuje  $x S_a y$  (tj.  $V_q$  uporządkowana według rosnących preferencji), wyróżniamy pięć typów reguł decyzyjnych:

1) pewne  $D_{\geq}$ -reguły decyzyjne o składni:

$$\text{jeżeli } (f(a_1, x) \geq r_{a_1}) \wedge (f(a_2, x) \geq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \geq r_{a_p}) \text{ to } x \in Cl_t^{\geq},$$

2) możliwe  $D_{\geq}$ -reguły decyzyjne o składni:

$$\text{jeżeli } (f(a_1, x) \geq r_{a_1}) \wedge (f(a_2, x) \geq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \geq r_{a_p}) \text{ to } x \text{ może należeć do } Cl_t^{\geq},$$

3) pewne  $D_{\leq}$ -reguły decyzyjne o składni:

$$\text{jeżeli } (f(a_1, x) \leq r_{a_1}) \wedge (f(a_2, x) \leq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \leq r_{a_p}) \text{ to } x \in Cl_t^{\leq},$$

4) możliwe  $D_{\leq}$ -reguły decyzyjne o składni:

$$\text{jeżeli } f(a_1, x) \leq r_{a_1} \wedge f(a_2, x) \leq r_{a_2} \wedge \dots \wedge f(a_p, x) \leq r_{a_p} \text{ to } x \text{ może należeć do } Cl_t^{\leq},$$

gdzie  $C = \{a_1, \dots, a_p\} \subseteq A$ ,  $(r_{a_1}, \dots, r_{a_p}) \in V_{a_1} \times V_{a_2} \times \dots \times V_{a_p}$  oraz  $t \in T$ ;

5) przybliżone  $D_{\geq\leq}$ -reguły decyzyjne o składni:

$$\text{jeżeli } (f(a_1, x) \geq r_{a_1}) \wedge (f(a_2, x) \geq r_{a_2}) \wedge \dots \wedge (f(a_k, x) \geq r_{a_k}) \wedge (f(a_{k+1}, x) \leq r_{a_{k+1}}) \wedge \dots \wedge (f(a_p, x) \leq r_{a_p}) \\ \text{to } x \in Cl_s \cup Cl_{s+1} \cup \dots \cup Cl_t,$$

gdzie  $O' = \{a_1, \dots, a_k\} \subseteq A$ ,  $O'' = \{a_{k+1}, \dots, a_p\} \subseteq A$ ,  $C = O' \cup O''$ ,  $O'$  i  $O''$  nie muszą być rozłączne,  $(r_{a_1}, \dots, r_{a_p}) \in V_{a_1} \times V_{a_2} \times \dots \times V_{a_p}$ ,  $s, t \in T$  takie, że  $s < t$ . Jeżeli  $\{a_1, \dots, a_k\} \cap$

$\{a_{k+1}, \dots, a_p\} \neq \emptyset$ , to w części warunkowej  $D_{\geq}$ -reguły wystąpią oba warunki  $(f(a, x) \geq r_a)$  i  $(f(a, x) \leq r'_a)$ , gdzie  $r_a \leq r'_a$ , dla pewnego  $a \in C$ . Jeśli  $r_q = r'_q$ , to oba warunki sprowadzają się do warunku  $(f(a, x) = r_a)$ .

Ponadto wymaga się, aby każda reguła decyzyjna była **minimalna**. Ponieważ reguła jest implikacją, przez minimalną regułę decyzyjną rozumiemy taką implikację, dla której nie istnieje żadna inna implikacja zawierająca część warunkową przynajmniej tak samo ogólną (innymi słowy zawierająca podzbiór warunków elementarnych i/lub warunki ogólniejsze) i część decyzyjną przypisującą obiekt do tej samej skumulowanej klasy lub podzbioru skumulowanych klas decyzyjnych.

Rozważmy  $D_{\geq}$ -regułę decyzyjną o składni: *jeśli*  $(f(a_1, x) \geq r_{a_1}) \wedge (f(a_2, x) \geq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \geq r_{a_p})$  *to*  $x \in Cl_t^{\geq}$ . Jeżeli istnieje obiekt  $x \in \underline{B}(Cl_t^{\geq})$  taki, że  $(f(a_1, x) = r_{a_1})$ ,  $(f(a_2, x) = r_{a_2})$ , ...,  $(f(a_p, x) = r_{a_p})$ , to mówimy, że jest to **obiekt bazowy** danej reguły. Regułę nazywamy opartą na obiekcie bazowym. Podobne rozważania przeprowadza się dla innych typów reguł.

Definicja obiektu wspierającego regułę oraz definicja obiektu pokrytego przez regułę są zgodne z definicjami (2.17) oraz (2.16).

Zbiór reguł pewnych i przybliżonych jest kompletny, jeżeli są spełnione następujące warunki:

1. Każdy  $y \in \underline{B}(Cl_t^{\geq})$  wspiera przynajmniej jedną pewną  $D_{\geq}$ -regułę decyzyjną z konkluzją  $x \in Cl_r^{\geq}$ , gdzie  $r, t \in \{2, \dots, n\}$  i  $r \geq t$ .
2. Każdy  $y \in \underline{B}(Cl_u^{\leq})$  wspiera przynajmniej jedną pewną  $D_{\leq}$ -regułę decyzyjną z konkluzją  $x \in Cl_u^{\leq}$ , gdzie  $u, t \in \{1, \dots, n-1\}$  i  $u \leq t$ .
3. Każdy  $y \in \overline{B}(Cl_s^{\leq}) \cap \overline{B}(Cl_t^{\geq})$  wspiera przynajmniej jedną przybliżoną  $D_{\geq}$ -regułę decyzyjną z konkluzją  $x \in Cl_v \cup Cl_{v+1} \cup \dots \cup Cl_z$ , gdzie  $s, t, v, z \in T$  oraz  $s \leq v < z \leq t$ .

Powyższe warunki oznaczają, że zbiór reguł pozwala na pokrycie wszystkich obiektów w tablicy decyzyjnej w taki sposób, że spójne przykłady zostaną zaklasyfikowane do ich oryginalnych klas decyzyjnych, a niespójne przykłady do grup klas związanych z niespójnościami. Analogiczne definicje można podać dla reguł możliwych.

Zbiór reguł decyzyjnych jest minimalny, jeżeli jest kompletny i nienadmiarowy, tzn. usunięcie dowolnej reguły powoduje utratę kompletności.

### Przykład 6.1

Załóżmy, że rozważamy tablicę decyzyjną, gdzie 17 obiektów opisanych jest zbiorem trzech kryteriów  $C = \{a_1, a_2, a_3\}$  i przydzielonych do trzech kategorii decyzyjnych. Na wartościach wszystkich kryteriów są zdefiniowane rosnące preferencje (tzn.  $f(a, x) \geq f(a, y)$  implikuje  $x S_a y$ ). Ponadto klasy decyzyjne są także uporządkowane według rosnących preferencji;  $Cl_3$  jest najbardziej, a  $Cl_1$  jest najmniej prefe-

rowaną klasą decyzyjną. Analizowany zbiór przykładów przedstawiono w tabeli 6.1.

Porządek klas decyzyjnych prowadzi do zdefiniowania następujących skumulowanych klas:  $Cl_1^{\leq} = \{x_3, x_4, x_7, x_9, x_{14}\}$ ,  $Cl_2^{\leq} = \{x_1, x_2, x_3, x_4, x_6, x_7, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\}$ ,  $Cl_2^{\geq} = \{x_1, x_2, x_5, x_6, x_8, x_{10}, x_{11}, x_{12}, x_{13}, x_{15}, x_{16}, x_{17}\}$ ,  $Cl_3^{\geq} = \{x_5, x_8, x_{16}, x_{17}\}$ .

**Tabela 6.1.** Tablica decyzyjna zawierająca przykłady wielokryterialnego problemu sortowania

$U$	$a_1$	$a_2$	$a_3$	$d$
$x_1$	1,5	3	12	$Cl_2$
$x_2$	1,7	5	9,5	$Cl_2$
$x_3$	0,5	2	2,5	$Cl_1$
$x_4$	0,7	0,5	1,5	$Cl_1$
$x_5$	3	4,3	9	$Cl_3$
$x_6$	1	2	4,5	$Cl_2$
$x_7$	1	1,2	8	$Cl_1$
$x_8$	2,3	3,3	9	$Cl_3$
$x_9$	1	3	5	$Cl_1$
$x_{10}$	1,7	2,8	3,5	$Cl_2$
$x_{11}$	2,5	4	11	$Cl_2$
$x_{12}$	0,5	3	6	$Cl_2$
$x_{13}$	1,2	1	7	$Cl_2$
$x_{14}$	2	2,4	6	$Cl_1$
$x_{15}$	1,9	4,3	14	$Cl_2$
$x_{16}$	2,3	4	13	$Cl_3$
$x_{17}$	2,7	5,5	15	$Cl_3$

Rozważmy przykłady obliczania relacji dominacji. Obiekt  $x_1$  dominuje obiekt  $x_3$ , gdyż  $f(a_1, x_1) = 1,5 > f(a_1, x_3) = 0,5$  i  $f(a_2, x_1) = 3 > f(a_2, x_3) = 2$  i  $f(a_3, x_1) = 12 > f(a_3, x_3) = 2,5$ . Nie występuje sprzeczność z zasadą dominacji, gdyż  $f(d, x_1) = Cl_2 > f(d, x_3) = Cl_1$ . Postępując w podobny sposób, można skonstruować następujące zbiory dominujące i dominowane  $D_C^-(x_1) = \{x_1, x_3, x_4, x_6, x_7, x_9, x_{12}, x_{13}\}$ ,  $D_C^+(x_1) = \{x_1, x_{15}, x_{16}, x_{17}\}$ .

W zbiorze obiektów występują niespójności z zasadą dominacji. Przykładowo,  $x_9$  dominuje  $x_6$  na kryteriach  $a_1, a_2, a_3$ . Jest on jednak przypisany do klasy  $Cl_1$ , która jest gorsza niż  $Cl_2$ , do której należy  $x_6$ . Tabela zawiera 5 obiektów naruszających zasadę dominacji, tj.  $x_6, x_8, x_9, x_{11}, x_{14}$ . Przybliżenia skumulowanych klas są następujące:

$$\underline{C}(Cl_1^{\leq}) = \{x_3, x_4, x_7\}, \quad \bar{C}(Cl_1^{\leq}) = \{x_3, x_4, x_6, x_7, x_9, x_{14}\}, \quad Bn_C(Cl_1^{\leq}) = \{x_6, x_9, x_{14}\},$$

$$\underline{C}(Cl_2^{\leq}) = \{x_1, x_2, x_3, x_4, x_6, x_7, x_9, x_{10}, x_{12}, x_{13}, x_{14}, x_{15}\},$$

$$\bar{C}(Cl_2^{\leq}) = \{x_1, x_2, x_3, x_4, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\}, \quad Bn_C(Cl_2^{\leq}) = \{x_8, x_{11}\},$$



$$\begin{aligned} \underline{C}(Cl_2^{\geq}) &= \{x_1, x_2, x_5, x_8, x_{10}, x_{11}, x_{12}, x_{13}, x_{15}, x_{16}, x_{17}\}, \\ \overline{C}(Cl_2^{\geq}) &= \{x_1, x_2, x_5, x_6, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}\}, \quad Bn_C(Cl_2^{\geq}) = \{x_6, x_9, x_{14}\}, \\ \underline{C}(Cl_3^{\geq}) &= \{x_5, x_{16}, x_{17}\}, \quad \overline{C}(Cl_3^{\geq}) = \{x_5, x_8, x_{11}, x_{16}, x_{17}\}, \quad Bn_C(Cl_3^{\geq}) = \{x_8, x_{11}\}. \end{aligned}$$

### 6.3. Model ze zmienną spójnością w teorii zbiorów przybliżonych opartej na relacji dominacji

Analiza rzeczywistych zbiorów danych wskazuje, że dla niektórych problemów zastosowanie uogólnienia teorii zbiorów przybliżonych opartego na relacji dominacji tworzy liczne obszary brzegowe pomiędzy dolnymi i górnymi przybliżeniami, a także prowadzi do indukcji dużej liczby reguł decyzyjnych wspieranych przez małą liczbę przykładów z dolnych przybliżeń [70]. Jedną w przyczyn jest silniejsze niż w przypadku oryginalnego sformułowania teorii zbiorów przybliżonych oddziaływanie niespójności z zasadą dominacji pomiędzy przykładami także na inne przykłady. Przykładowo, tego typu niespójność pomiędzy obiektami  $x$  oraz  $y$  przypisanymi do odległych klas decyzyjnych odpowiednio  $h$  i  $t$  ( $x$  dominuje  $y$ , podczas gdy klasa  $h$  jest gorsza niż  $t$ ) może powodować także niespójność z obiektami należącymi do pośrednich klas decyzyjnych (od  $h$  do  $t$ ) i zdominowanymi przez obiekt  $x$ . W takich przypadkach naturalne wydaje się osłabienie wymagań co do przynależności obiektów do dolnych przybliżeń. Zgodnie z definicjami (6.5) i (6.6) tylko „jednoznaczne” w sensie relacji dominacji obiekty mogły być zaliczone do dolnego przybliżenia. Rozluźnienie tego wymagania pozwoliłoby na zaliczenie do dolnego przybliżenia pewnej liczby niejednoznacznych obiektów w sensie przynależności. Zakres uwzględnienia tych obiektów byłby kontrolowany i ograniczony za pomocą współczynnika **poziomu spójności** (ang. *consistency level*). Powyższe motywacje doprowadziły do sformułowania kolejnego uogólnienia nazywanego modelem ze zmienną spójnością w teorii zbiorów przybliżonych opartej na relacji dominacji (ang. *variable consistency model of dominance-based rough sets approach – VC-DRSA*) [70].

Początkowe definicje reprezentacji przykładów, relacji dominacji oraz zbiorów dominujących i dominowanych są identyczne jak przedstawione w rozdziale 6.2.

Dla podzbioru kryteriów  $B \subseteq A$  mówimy, że  $x \in U$  należy do  $Cl_t^{\geq}$  ze spójnością nie mniejszą od poziomu  $l \in (0, 1]$ , jeżeli  $x \in Cl_t^{\geq}$  i przynajmniej  $l \cdot 100\%$  obiektów  $y \in U$  dominujących  $x$  ze względu na podzbiór kryteriów  $B$  także należy do  $Cl_t^{\geq}$ , tzn. spełnione jest:

$$\frac{|D_B^+(x) \cap Cl_t^{\geq}|}{|D_B^+(x)|} \geq l \quad (6.8)$$

Współczynnik  $l$  nazywany jest poziomem spójności, gdyż kontroluje on stopień spójności pomiędzy obiektami przydzielonymi do  $Cl_t^{\geq}$  bez wątpliwości. Jeżeli  $l < 1$ , to  $(1-l)*100\%$  obiektów  $y \in U$  dominujących  $x$  nie należy do  $Cl_t^{\geq}$  i przeciwstawia się przynależności obiektu  $x$  do  $Cl_t^{\geq}$ .

Podobnie dla  $B \subseteq A$  mówimy, że  $x \in U$  należy do  $Cl_t^{\leq}$  ze spójnością nie mniejszą od poziomu  $l \in (0, 1]$ , jeżeli  $x \in Cl_t^{\leq}$  i przynajmniej  $l*100\%$  obiektów  $y \in U$  zdominowanych przez  $x$  ze względu na podzbiór kryteriów  $B$  także należy do  $Cl_t^{\leq}$ , tzn.:

$$\frac{|D_B^-(x) \cap Cl_t^{\leq}|}{|D_B^-(x)|} \geq l \quad (6.9)$$

Dla danego podzbioru kryteriów  $B \subseteq A$  oraz poziomu spójności  $l$   $B$ -dolne przybliżenia skumulowanych klas decyzyjnych  $Cl_t^{\geq}$  i  $Cl_t^{\leq}$  są zdefiniowane następująco:

$$\underline{B}^l(Cl_t^{\geq}) = \{x \in Cl_t^{\geq} : \frac{|D_B^+(x) \cap Cl_t^{\geq}|}{|D_B^+(x)|} \geq l\} \quad (6.10)$$

$$\underline{B}^l(Cl_t^{\leq}) = \{x \in Cl_t^{\leq} : \frac{|D_B^-(x) \cap Cl_t^{\leq}|}{|D_B^-(x)|} \geq l\} \quad (6.11)$$

$B$ -górne przybliżenia skumulowanych klas decyzyjnych  $Cl_t^{\geq}$  i  $Cl_t^{\leq}$ , oznaczane  $\bar{B}^l(Cl_t^{\geq})$  i  $\bar{B}^l(Cl_t^{\leq})$ , są zdefiniowane jako uzupełnienia  $\underline{B}^l(Cl_{t-1}^{\leq})$  i  $\underline{B}^l(Cl_{t+1}^{\geq})$  ze względu na  $U$ , tzn.:

$$\bar{B}^l(Cl_t^{\geq}) = U - \underline{B}^l(Cl_{t-1}^{\leq}), \quad \bar{B}^l(Cl_t^{\leq}) = U - \underline{B}^l(Cl_{t+1}^{\geq}) \quad (6.12)$$

$B$ -brzegi  $Cl_t^{\geq}$  i  $Cl_t^{\leq}$  są zdefiniowane jako:

$$Bn_P(Cl_t^{\geq}) = \bar{B}^l(Cl_t^{\geq}) - \underline{B}^l(Cl_t^{\leq}), \quad Bn_P(Cl_t^{\leq}) = \bar{B}^l(Cl_t^{\leq}) - \underline{B}^l(Cl_t^{\geq}), \quad \text{dla } t = 1, \dots, n.$$

Model z poziomem spójności umożliwia elastyczność w określaniu przydziału obiektów do przybliżeń skumulowanych klas decyzyjnych. Można pokazać [70], że dla  $0 < l' < l \leq 1$  oraz  $t = 2, \dots, n$ :

$$\underline{B}^l(Cl_t^{\leq}) \subseteq \underline{B}^{l'}(Cl_t^{\leq}) \quad \text{i} \quad \bar{B}^{l'}(Cl_t^{\geq}) \subseteq \bar{B}^l(Cl_t^{\geq}) \quad (6.13)$$

Omawiane uogólnienie jest inspirowane modelem zmiennej precyzji w teorii zbiorów przybliżonych wprowadzonym przez Ziarkę [274]. Istnieją jednak znaczne różnice w konstruowaniu przybliżeń. W oryginalnym modelu Ziarki dolne przybliżenia są konstruowane na podstawie zbiorów elementarnych i w konsekwencji mogą

zawierać ograniczoną liczbę przykładów spoza rozważanej skumulowanej klasy, podczas gdy w przedstawionym uogólnieniu dolne przybliżenia są skonstruowane wyłącznie na podstawie obiektów z właściwej skumulowanej klasy. Szersza dyskusja różnic przedstawiona jest w [70]. Praca ta zawiera także omówienie innych właściwości przybliżeń, definicje jakości przybliżenia klasyfikacji oraz reduktu.

Reguły decyzyjne generowane są z dolnych przybliżeń skumulowanych klas decyzyjnych. Dla przybliżenia skumulowanej klasy  $CI_t^{\geq}$  lub  $CI_s^{\leq}$  indukcja reguł odbywa się przy założeniu, że obiekty należące do  $\underline{B}^l(CI_t^{\geq})$  (lub  $\underline{B}^l(CI_s^{\leq})$ ) są przykładami pozytywnymi, a wszystkie inne obiekty są przykładami negatywnymi. Ze względu na stosowanie modelu VC-DRSA z każdą regułą związany jest parametr  $\alpha$ , nazywany dokładnością reguły.

Zakładając, że dla każdego kryterium  $a \in A$ ,  $V_a \subseteq \mathbf{R}$  oraz  $\forall x, y \in U$ ,  $f(a, x) \geq f(a, y)$  implikuje  $x S_{ay}$ , wyróżniamy dwa typy reguł decyzyjnych:

1)  $D_{\geq}$ -reguły decyzyjne o składni:

*jeżeli*  $(f(a_1, x) \geq r_{a_1}) \wedge (f(a_2, x) \geq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \geq r_{a_p})$  *to*  $x \in CI_t^{\geq}$  z dokładnością  $\alpha$

2)  $D_{\leq}$ -reguły decyzyjne o składni:

*jeżeli*  $(f(a_1, x) \leq r_{a_1}) \wedge (f(a_2, x) \leq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \leq r_{a_p})$  *to*  $x \in CI_t^{\leq}$ , z dokładnością  $\alpha$  gdzie  $C = \{a_1, \dots, a_p\} \subseteq A$ ,  $(r_{a_1}, \dots, r_{a_p}) \in V_{a_1} \times V_{a_2} \times \dots \times V_{a_p}$  oraz  $t \in T$ .

Podobnie jak w podejściu DRSA, można zdefiniować regułę decyzyjną opartą na obiekcie bazowym. Definicja obiektu pokrytego przez regułę oraz definicja obiektu wspierającego regułę pozostają także bez zmian. Jeżeli  $P$  oznacza część warunkową reguły typu (1), tj.  $(f(a_1, x) \geq r_{a_1}) \wedge (f(a_2, x) \geq r_{a_2}) \wedge \dots \wedge (f(a_p, x) \geq r_{a_p})$ , a  $[P]$  – zbiór obiektów spełniających warunki z  $P$ , to dokładność reguły jest zdefiniowana jako:

$$\alpha = \frac{|[P] \cap CI_t^{\geq}|}{|[P]|} \quad (6.14)$$

Analogiczną definicję można sformułować dla reguł typu (2). Zauważmy, że reguły indukowane są z dolnych przybliżeń skumulowanych klas zdefiniowanych z pewnym poziomem spójności  $l$ , który może być zmieniany w zależności od specyfiki zbioru oraz oczekiwań użytkownika. Wartości dokładności reguły powinny być także ograniczone. W skrajnym przypadku, kiedy części warunkowe reguł odpowiadają opisom niektórych obiektów z dolnego przybliżenia (reguły są oparte na takich obiektach), dokładność reguł  $\alpha$  będzie ograniczona z dołu przez stopień spójności  $l$ . Można wprowadzić wymóg w stosunku do algorytmu indukcji reguł typu (1) oraz (2), aby zachodził warunek  $\alpha \geq l$ . Użytkownik może podnieść próg

minimalnej wartości  $\alpha$  powyżej  $l$ , lecz wygenerowane reguły mogą wtedy nie pokryć wszystkich obiektów z przybliżeń skumulowanych klas decyzyjnych.

Ponadto wymaga się, aby każda reguła decyzyjna była minimalna. Ponieważ reguła jest implikacją, przez minimalną regułę decyzyjną rozumiemy taką implikację, dla której nie istnieje żadna inna implikacja zawierająca część warunkową przynajmniej tak samo ogólną (innymi słowy zawierająca podzbiór warunków elementarnych i/lub warunki ogólniejsze) i część decyzyjną przypisującą obiekt do tej samej skumulowanej klasy lub podzbioru skumulowanych klas decyzyjnych z dokładnością reguły  $\alpha \geq l$ .

### Kontynuacja przykładu 6.1

Rozważmy ponownie tablicę decyzyjną 6.1, ograniczając podzbiór kryteriów do  $B = \{a_1, a_2\}$ . Załóżmy, że użytkownik jest zainteresowany skumulowaną klasą decyzyjną  $Cl_2^{\geq}$ . Używając podejścia DRSA, można znaleźć następujące dolne przybliżenie  $\underline{B}(Cl_2^{\geq}) = \{x_1, x_2, x_5, x_8, x_{10}, x_{11}, x_{15}, x_{16}, x_{17}\}$  oraz brzeg  $Bn_B(Cl_2^{\geq}) = \{x_6, x_9, x_{12}, x_{13}, x_{14}\}$ . Załóżmy, że użytkownik zdecydował się wykorzystać model VC-DRSA ze współczynnikiem spójności  $l$  równym 0,8. Wtedy zbiory  $B$ -dominujące dla obiektów  $x_6, x_{12}, x_{13}$  zawierają się w  $Cl_2^{\geq}$  ze stopniem większym niż  $l$  (odpowiednio 0,83, 0,9 oraz 0,91). Mogą być więc dodane do dolnego przybliżenia  $\underline{B}^{0,75}(Cl_2^{\geq})$ . Obszar brzegu zostaje ograniczony do obiektów  $x_9$  i  $x_{14}$ . Oznacza to zwiększenie jakości przybliżenia  $Cl_2^{\geq}$ .

## 6.4. Indukcja reguł decyzyjnych w teorii zbiorów przybliżonych opartej na relacji dominacji

Zaproponowane dotychczas algorytmy indukcji reguł decyzyjnych wykorzystują oryginalną teorię zbiorów przybliżonych opartą na relacji nierozróżnialności. Żaden z tych algorytmów nie może być jednak bezpośrednio zastosowany do indukcji reguł w omawianych uogólnieniach. Wynika to z konieczności uwzględnienia zasady dominacji, wykorzystywania innego rodzaju przybliżeń oraz uogólnionej składni reguł. W związku z tym autor wprowadzał nowe algorytmy lub rozważał modyfikacje znanych algorytmów z uczenia maszynowego czy eksploracji danych [71, 238]. W dalszej części prezentujemy krótki opis dwóch z zaproponowanych rozwiązań, tj. algorytmów DOMLEM i DOMApriori. Oba algorytmy są iteracyjnie uruchamiane dla odpowiednich dolnych i/lub górnych przybliżeń skierowanych ku górze i ku dołowi skumulowanych klas decyzyjnych. W zależności od typu przybliżenia otrzymuje się reguły decyzyjne odpowiedniego typu. W podejściu DRSA indukuje się reguły:

- typu 1 z dolnych przybliżeń skierowanych ku górze skumulowanych klas decyzyjnych,

- typu 2 z górnych przybliżeń skierowanych ku górze skumulowanych klas decyzyjnych,
- typu 3 z dolnych przybliżeń skierowanych ku dołowi skumulowanych klas decyzyjnych,
- typu 4 z górnych przybliżeń skierowanych ku dołowi skumulowanych klas decyzyjnych,
- typu 5 z brzegów odpowiednich skumulowanych klas decyzyjnych.

W przypadku modelu VC-DRSA indukcję reguł typu 1 prowadzi się z rozszerzonych dolnych przybliżeń skierowanych ku górze skumulowanych klas decyzyjnych, a indukcję reguł typu 2 z rozszerzonych dolnych przybliżeń skierowanych ku dołowi skumulowanych klas decyzyjnych.

Ze względu na minimalność reguł dogodnie jest rozpoczynać indukcję reguł od najsilniejszych skumulowanych klas decyzyjnych, np. dla reguł typu 1 kolejność iteracji jest następująca  $\underline{B}(Cl_t^{\geq}), \underline{B}(Cl_{t-1}^{\geq}), \dots, \underline{B}(Cl_2^{\geq})$ .

#### 6.4.1. Algorytm DOMLEM

Algorytm DOMLEM jest ukierunkowany na poszukiwanie minimalnego zbioru reguł. Jego konstrukcja jest inspirowana algorytmem MODLEM [235]. Poniżej przedstawiamy zapis algorytmu dla podejścia DRSA i reguł typu 1, w którym obiekty opisane są za pomocą zbioru kryteriów  $B$ . Więcej szczegółów podano w [71]. W stosowanej notacji  $\mathbf{P}$  jest zbiorem koniunkcji  $P$  stanowiącym lokalne pokrycie zbioru  $K$ , będącego odpowiednim dolnym przybliżeniem  $\underline{B}(Cl_s^{\geq})$ , gdzie  $s = t, \dots, 2$ . Koniunkcje te są użyte do tworzenia reguł decyzyjnych. Koniunkcja warunków elementarnych  $P = w_1 \wedge w_2 \wedge \dots \wedge w_m$ , gdzie  $w_i = (f(a_i, x) \geq r_{ai})$ , może być zaakceptowana jako część warunkowa reguły wskazującej  $x \in Cl_s^{\geq}$ , jeżeli:

$$\emptyset \neq [P] = \bigcap_{i=1}^m [w_i] \subseteq K \quad (6.15)$$

##### Procedure DOMLEM

(input:  $L_{upp}$  – rodzina dolnych przybliżeń  $\{\underline{B}(Cl_t^{\geq}), \underline{B}(Cl_{t-1}^{\geq}), \dots, \underline{B}(Cl_2^{\geq})\}$ ;

output:  $R_{\geq}$  zbiór  $D_{\geq}$ -reguł decyzyjnych);

**begin**

$R_{\geq} := \emptyset$ ;

**for** każdego przybliżenia  $K \in L_{upp}$  **do begin**

$\mathbf{E} := \text{Find\_rules}(K, B)$ ;

**for** każdej reguły  $r \in \mathbf{E}$  **do**

**if** reguła  $r$  jest minimalna **then**  $R_{\geq} := R_{\geq} \cup r$ ;

**end**

**end** {procedure}.

**Function** *Find\_rules***(input:**  $K$  zbiór przykładów pozytywnych;  $C$  zbiór kryteriów;**output:**  $E$  – zbiór reguł pokrywający przykłady z  $K$ );**begin** $G := K$ ; {zbiór obiektów nie pokrytych dotychczas poprzez elementy z  $P$ } $P := \emptyset$ ; {lokalne pokrycie zbioru  $K$ }**while**  $G \neq \emptyset$  **do****begin** $P := \emptyset$ ; {kandydat na część warunkową reguły} $S := G$ ; {zbiór obiektów pokrywany przez  $P$ }**while**  $(P = \emptyset)$  **or not**  $([P] \subseteq K)$  **do****begin** $w := \emptyset$ ; {najlepszy kandydat na warunek elementarny}**for** każdego kryterium  $a_i \in C$  **do****begin** $Cond := \{(f(a_i, x) \geq r_{ai}) : \exists x \in S \text{ taki, że } (f(a_i, x) = r_{ai})\}$ ;{dla każdego przykładu pozytywnego z  $S$  rozważ utworzenie warunku elementarnego}**for** każdy warunek elementarny  $new\_p \in Cond$  **do****begin****if** *evaluate* $(P \cup new\_p, P \cup w)$  **then**{sprawdź, czy warunek  $new\_p$  jest lepszy niż  $w$ } $w := new\_p$ ; $w\_eval := new\_eval\_p$ ;**end**;**end**; {for} $P := P \cup \{w\}$ ; {dołącz najlepszy warunek do koniunkcji  $P$ } $S := S \cap [w]$ ; {ogranicz zbiór obiektów}**end**; {while not  $([P] \subseteq K)$ }**for** każdy warunek elementarny  $w \in P$  **do****if**  $[P - \{w\}] \subseteq K$  **then**  $P := P - \{w\}$ ; $P := P \cup \{P\}$ ; $G := G - \cup_{P \in \mathcal{P}} [P]$ ;**end**; {while  $G \neq \emptyset$ }utwórz reguły  $E$  na podstawie lokalnego pokrycia  $P$ **end** {function}

Poszukiwany „kandydat” na warunek elementarny oceniany jest za pomocą funkcji *evaluate*. Kandydująca koniunkcja warunków może być oceniana za pomocą różnych miar [71] (porównaj dyskusje w rozdziałach 2.1 oraz 4). W przykładzie omawianym dalej stosujemy następujący porządek leksykograficzny kryteriów: za lepszą uznaje się tę koniunkcję  $P$ , dla której współczynnik  $\frac{|[P] \cap G|}{|[P]|}$  jest większy; jeśli obie koniunkcje mają taką samą wartość, za lepszą uznaje się tę, dla której wyrażenie  $[P] \cap G$  osiąga większą wartość; jeśli i tutaj koniunkcje mają identyczną, wartość wybierz arbitralnie pierwszą z listy. Taki porządek inspirowany jest rozwiązaniami z algorytmu LEM2 [73].

Należy zauważyć, że wymóg indukcji reguł opartych na obiektach silnie ogranicza przestrzeń przeszukiwania, gdyż po wyborze pierwszego warunku należy dalej rozpatrywać wyłącznie warunki z progiem  $r_a$  odnoszącym się do opisu obiektu bazowego użytego przy pierwszym wybranym warunku elementarnym.

Powyższy schemat indukcji reguł obowiązuje także dla reguł typu 2–4. Zmiany dotyczą rodzaju przybliżeń podawanych na wejście algorytmu oraz sposobu definiowania warunku elementarnego. W przypadku poszukiwania reguł przybliżonych typu 5 należy jako parametrów wejściowych do algorytmu użyć odpowiednich brzegów pomiędzy uniami klas oraz dopuszczać tworzenie obu postaci warunków ( $f(a_i, x) \geq r_{ai}$ ) i ( $f(a_i, x) \leq r_{ai}$ ) dla tego samego kryterium  $a_i$ . Dyskusja tego przypadku podana jest w [70].

Algorytm DOMLEM może być w prosty sposób zmodyfikowany dla indukcji reguł w modelu ze współczynnikiem spójności VC-DRSA [70]. Reguły są indukowane z przykładów pozytywnych należących do rozszerzonych  $B^l$ -dolnych przybliżeń odpowiednich skumulowanych klas decyzyjnych. Zmianie ulega też warunek akceptacji reguły. Dotychczas był on zdefiniowany jako  $[P] \subseteq K$ . W przypadku VC-DRSA warunek akceptacji polega na sprawdzeniu, czy dokładność  $\alpha$  reguły zbudowanej na podstawie koniunkcji  $P$  jest nie gorsza niż założony przez użytkownika próg  $l$ . Reszta schematu algorytmu DOMLEM nie ulega zmianie.

Przedyskutujmy krótko złożoność obliczeniową algorytmu DOMLEM. Podobnie jak w przypadku algorytmu MODLEM funkcja złożoności jest wielomianowa rzędu  $O(nWm^2)$ , gdzie  $n$  jest liczbą przykładów,  $W$  średnią liczbą wartości w dziedzinie kryterium,  $m$  – liczbą kryteriów. Należy tutaj zauważyć, że w algorytmie DOMLEM realizacja sprawdzenia minimalności wymaga przeprowadzenia większej liczby operacji ze względu na konieczność sprawdzenia minimalności dla reguł wygenerowanych dla skumulowanej klasy, np.  $Cl_s^{\geq}$ , względem poprzednio wygenerowanych reguł dla silniejszych skumulowanych klas  $Cl_t^{\geq}$  ( $t > s$ ). Indukcja reguł opartych na obiektach bazowych zmniejsza koszty obliczeń.

### Kontynuacja przykładu 6.1

Zastosowanie algorytmu DOMLEM do przykładów z tabeli 6.1 prowadzi do wygenerowania następującego minimalnego zbioru 12 reguł pewnych i przybliżonych (w nawiasach podano obiekty pokryte przez regułę):

jeżeli ( $f(a_3, x) \leq 2,5$ ) to  $x \in Cl_1^{\leq}$   $\{x_3, x_4\}$

jeżeli ( $f(a_2, x) \leq 1,2$ )  $\wedge$  ( $f(a_1, x) \leq 1,0$ ) to  $x \in Cl_1^{\leq}$   $\{x_4, x_7\}$

jeżeli ( $f(a_1, x) \leq 2,0$ ) to  $x \in Cl_2^{\leq}$   $\{x_1, x_2, x_3, x_4, x_6, x_7, x_9, x_{10}, x_{12}, x_{13}, x_{14}, x_{15}\}$

jeżeli ( $f(a_1, x) \geq 2,7$ ) to  $x \in Cl_3^{\geq}$   $\{x_5, x_{17}\}$

jeżeli ( $f(a_3, x) \geq 13$ )  $\wedge$  ( $f(a_1, x) \geq 2,3$ ) to  $x \in Cl_3^{\geq}$   $\{x_{16}, x_{17}\}$

jeżeli ( $f(a_1, x) \geq 1,2$ )  $\wedge$  ( $f(a_3, x) \geq 7,0$ ) to  $x \in Cl_2^{\geq}$   $\{x_1, x_2, x_5, x_8, x_{11}, x_{13}, x_{15}, x_{16}, x_{17}\}$

$$\begin{aligned}
& \text{jeżeli } (f(a_2, x) \geq 2,8) \wedge (f(a_3, x) \geq 6,0) \text{ to } x \in Cl_2^{\geq} \quad \{x_1, x_2, x_5, x_8, x_{11}, x_{12}, x_{15}, x_{16}, x_{17}\} \\
& \text{jeżeli } (f(a_2, x) \geq 2,8) \wedge (f(a_1, x) \geq 1,7) \text{ to } x \in Cl_2^{\geq} \quad \{x_2, x_5, x_8, x_{10}, x_{11}, x_{15}, x_{16}, x_{17}\} \\
& \text{jeżeli } (f(a_1, x) \geq 2,3) \wedge (f(a_2, x) \leq 3,3) \text{ to } x \in Cl_2 \cup Cl_3 \quad \{x_8\} \\
& \text{jeżeli } (f(a_1, x) \geq 2,5) \wedge (f(a_2, x) \leq 4,0) \text{ to } x \in Cl_2 \cup Cl_3 \quad \{x_{11}\} \\
& \text{jeżeli } (f(a_3, x) \leq 6,0) \wedge (f(a_1, x) \geq 2,0) \text{ to } x \in Cl_1 \cup Cl_2 \quad \{x_{14}\} \\
& \text{jeżeli } (f(a_3, x) \geq 4,5) \wedge (f(a_3, x) \leq 5,0) \text{ to } x \in Cl_1 \cup Cl_2 \quad \{x_6, x_9\}
\end{aligned}$$

Zwróćmy uwagę, że niektóre reguły oparte są na obiektach bazowych, podczas gdy inne z reguł nie są. Przykładowo, reguła *jeżeli*  $(f(a_2, x) \geq 2,8) \wedge (f(a_3, x) \geq 6,0)$  to  $x \in Cl_2^{\geq}$  nie jest oparta na żadnym z obiektów bazowych, podczas gdy reguła *jeżeli*  $(f(a_2, x) \geq 2,8) \wedge (f(a_1, x) \geq 1,7)$  to  $x \in Cl_2^{\geq}$  oparta jest na obiekcie bazowym  $x_{10}$ .

Algorytm DOMLEM dla modelu VC-DRSA ze współczynnikiem spójności  $l = 0,8$  i wymogiem, aby dokładność reguły  $\alpha \geq l$ , prowadzi do zbioru 5 reguł:

$$\begin{aligned}
& \text{jeżeli } (f(a_3, x) \leq 2,5) \text{ to } x \in Cl_1^{\leq} \quad z \alpha = 1,0 \quad \{x_3, x_4\} \\
& \text{jeżeli } (f(a_2, x) \leq 1,2) \wedge (f(a_1, x) \leq 1,0) \text{ to } x \in Cl_1^{\leq} \quad z \alpha = 1,0 \quad \{x_4, x_7\} \\
& \text{jeżeli } (f(a_3, x) \leq 14) \text{ to } x \in Cl_2^{\leq} \quad z \alpha = 0,81 \quad \{x_1, x_2, x_3, x_4, x_6, x_7, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_5, x_8, x_{16}\} \\
& \text{jeżeli } (f(a_1, x) \geq 2,3) \text{ to } x \in Cl_3^{\geq} \quad z \alpha = 0,8 \quad \{x_5, x_8, x_{16}, x_{17}, x_{11}\} \\
& \text{jeżeli } (f(a_3, x) \geq 3,5) \text{ to } x \in Cl_2^{\geq} \quad z \alpha = 0,8 \quad \{x_1, x_2, x_5, x_6, x_8, x_{10}, x_{11}, x_{12}, x_{13}, x_{15}, x_{16}, x_{17}, x_7, x_9, x_{14}\}
\end{aligned}$$

Wśród obiektów pokrywanych przez regułę za pomocą pogrubionej czcionki zaznaczono obiekty spoza skumulowanej klasy wskazywanej przez regułę. Jak łatwo sprawdzić, powyższy zbiór reguł jest kompletny. Eksperymenty obliczeniowe z implementacją algorytmu DOMLEM w wersji VC-DRSA prowadziły do zmniejszenia (lub utrzymania tej samej) liczby reguł w stosunku do wersji DRSA.

#### 6.4.2. Algorytm DOMApriori

Algorytm DOMApriori umożliwia generację zbioru wszystkich reguł spełniających narzucone przez użytkownika ograniczenia co do minimalnej wartości wsparcia reguły  $G(P \wedge Q)$ <sup>9</sup> oraz maksymalnej dopuszczalnej długości reguły wyrażonej przez liczbę warunków elementarnych  $Dl(P)$ . W przypadku modelu ze zmienną spójnością użytkownik może dodatkowo wyrazić ograniczenie do minimalnej dopuszczalnej wartości dokładności reguły  $\alpha$ . Tak ograniczony zbiór reguł odpowiada pojęciu satysfakcjonującego zbioru reguł przedstawionego w rozdziale 2.2.

Dla podejścia DRSA indukuje się reguły pewne, np. typu 1 na podstawie przykładów pozytywnych z dolnych przybliżeń skumulowanych klas skierowanych ku górze. Reguły możliwe lub przybliżone są indukowane analogicznie z odpowiednich górnych przybliżeń lub brzegów. Ze względu na warunek akceptacji reguły

<sup>9</sup> W samym algorytmie wartość względnego wsparcia (patrz wzór 3.3) przeliczana jest na bezwzględną wartość liczby obiektów  $n_{PQ}$ .



w podejściu DRSA, wewnątrz algorytmu wsparcie koniunkcji reguły obliczane jest wyłącznie dla przykładów pozytywnych. Dla modelu VC-DRSA od początku wylicza się wsparcie na podstawie przykładów z danej skumulowanej klasy.

Proponowany algorytm wynika z inspiracji wykonaną poprzednio przez autora i Bobina [16] modyfikacją algorytmu generacji reguł asocjacyjnych [1] dla poszukiwania reguł w zadaniach klasyfikacji. Należy zwrócić uwagę, że ta modyfikacja była zaproponowana do przetwarzania atrybutów nominalnych. W przypadku problemów wielokryterialnych rozważa się kryteria, które są definiowane na skalach przynajmniej porządkowych.

W przedstawionym poniżej algorytmie zasadniczą częścią jest generacja tzw. silnych zbiorów warunków (ang. *large* lub *frequent itemsets*). Są to koniunkcje warunków elementarnych, które wspierane są przez liczbę obiektów nie mniejszą niż założony próg *minsupport*. Podobnie jak w oryginalnym algorytmie APRIORI [1] przyjmuje się, że warunki elementarne są uporządkowane leksykograficznie. W tworzonej koniunkcji warunki o wyższym indeksie muszą występować po warunkach o niższym indeksie. Wprowadzamy ponadto wymóg, aby kryterium  $a_i$  mogło być użyte w warunkach elementarnych silnego zbioru najwyżej raz. Stosuje się podobną notację jak w oryginalnym algorytmie APRIORI:

- $L_k$  – rodzina zaakceptowanych silnych zbiorów warunków,
- $C_k$  – rodzina kandydujących zbiorów warunków.

Indeks  $k$  oznacza wielkość zbioru, to jest liczbę użytych warunków elementarnych. Jeżeli kandydujący zbiór warunków z  $C_k$  spełnia warunek minimalnego wsparcia *minsupport*, to jest zaliczany do  $L_k$ . Z każdym zbiorem  $c$  z  $C_k$  są związane dwa parametry:

- $c.postive\_support$  – liczba pozytywnych przykładów (DRSA) lub obiektów z właściwej skumulowanej klasy pokrytych przez zbiór warunków,
- $c.negative\_support$  – liczba negatywnych przykładów pokrytych przez zbiór warunków.

W przypadku składni reguł omawianej w rozdziałach 6.2 oraz 6.3 elementarne warunki konstruowane są na podstawie opisu przykładów z danego przybliżenia skumulowanej klasy. Przykładowo, dla  $D_{\geq}$ -reguł, kryterium  $a_i$  i obiektów  $x$  z przybliżenia  $K$  tworzy się następującą listę warunków:

$$Cond := \{(f(a_i, x) \geq r_{ai}) : \exists x \in K (f(a_i, x) = r_{ai})\} \quad (6.16)$$

Dla  $D_{\leq}$ -reguł należy rozważać warunki postaci  $(f(a_i, x) \leq r_{ai})$ .

Podkreślmy, że pomiędzy proponowanym algorytmem DOMApriori a algorytmem APRIORI istnieje zasadnicza różnica w tworzeniu reguł decyzyjnych na podstawie znalezionych silnych zbiorów warunków  $L_k$ . Reguły asocjacyjne reprezentują zależności pomiędzy atrybutami, gdzie nie występują żadne klasy decyzyjne [1], podczas gdy w tablicach decyzyjnych  $DT$  definicje klas decyzyjnych

są znane i jawnie podane. Z tego powodu faza samej indukcji reguł w omawianym algorytmie DOMApriori jest zrealizowana w inny sposób. Warunki elementarne są tworzone wyłącznie na podstawie przykładów z przybliżenia danej skumulowanej klasy. Faza generacji reguł z silnych zbiorów  $L_k$  sprowadza się więc do sprawdzania warunku akceptacji reguły (DRSA –  $c.negative\_support = 0$ ) lub dokładności reguły utworzonej na jego podstawie (VC-DRSA –  $\alpha \geq$  próg akceptacji).

Realizacja testów minimalności reguł wykonana jest dwuetapowo. Najpierw wewnątrz danego przybliżenia skumulowanej klasy sprawdzane jest istnienie reguł o bardziej ogólnej części warunkowej. Następnie drugi test jest wykonywany przeciwnie regułom z silniejszej skumulowanej klasy decyzyjnej.

Poniżej prezentujemy schemat algorytmu DOMApriori dla  $D_{\geq}$ -reguł indukowanych kolejno z  $\underline{B}(Cl_s^{\geq})$ , gdzie  $s = t, \dots, 2$ . Dla innych typów reguł oraz modelu ze zmienną spójnością przebieg algorytmu jest podobny. Stosowane warunki zatrzymania to:  $minsupport$ ,  $maxlength$  oraz warunek progu  $\alpha l$  dla VC-DRSA.

**Procedure** DOMApriori

(**input**  $\underline{B}(Cl_s^{\geq})$  – dolne przybliżenie  $Cl_s^{\geq}$ ,

**output**  $R_{\geq}$  –  $D_{\geq}$ -reguły decyzyjne dla  $Cl_s^{\geq}$ );

**function** *createconditions* ( $\underline{B}(Cl_s^{\geq})$ ):  $L_1$ : rodzina silnych zbiorów; {stwórz  $L_1$ }

**begin**

**for** każdego  $x \in \underline{B}(Cl_s^{\geq})$  **do begin**

**for** każdego kryterium  $a \in B$  **do begin**

$r_{ai} := f(a_i, x)$ ;

**if** warunek  $(f(a_i, x) \geq r_{ai}) \notin C_1$  **then** dodaj  $(f(a_i, x) \geq r_{ai})$  do  $C_1$ ;

**end**

**end;**

**for** każdego  $x \in U$  **do**

*Updatesupport*( $C_1, x$ ); {uaktualnij liczniki  $c.positive\_support$  or  $c.negative\_support$  dla wszystkich  $c \in C_1$ , które pokrywają  $x$ }

$L_1 := \{ c \in C_1 : c.positive\_support \geq minsupport \}$  {odrzuć za „słabe” warunki}

**end**

**function** *apriori2\_gen* { $L_{k-1}$ : rodzina silnych zbiorów}: rodzina zbiorów kandydujących  $C_k$ ;

**begin**

**insert into**  $C_k$

**select**  $p.item_1, p.item_2, \dots, p.item_{k-2}, p.item_{k-1}, q.item_{k-1}$  {wykorzystaj porządek

    leksykograficzny warunków przy połączeniu dwóch silnych zbiorów  $L_{k-1}p, L_{k-1}q$ }

**from**  $L_{k-1}p, L_{k-1}q$

**where**

$p.negative\_support > 0$ ,           {nie rozszerzaj pewnych reguł}

$q.negative\_support > 0$ ,

$p.item_1 = q.item_1$ ,

$p.item_2 = q.item_2, \dots, p.item_{k-2} = q.item_{k-2}$

```

     $p.item_{k-1} < q.item_{k-1}$  {zbiory  $L_{k-1}p, L_{k-1}q$  mogą się różnić tylko na ostatniej pozycji}
    and oba  $p.item_{k-1}, q.item_{k-1}$  nie dotyczą tego samego kryterium
for wszystkich  $(k-1)$  podzbiorów  $s \in C_k$  do {podzbiory silnych warunków muszą}
    if  $(s \notin L_{k-1})$  then usuń  $s$  z  $C_k$  {także być silne}
end {apriori2_gen}

begin {główna część algorytmu}
     $C_1 := \emptyset$ ;
     $L_1 := createconditions ( \underline{B}(CI_s^{\geq}) );$ 
     $k := 2$ ;
    while  $(L_{k-1} \neq \emptyset)$  and  $(k \leq maxlen)$  do
    begin
         $C_k := apriori2\_gen(L_{k-1})$ ; {znajdź nowych kandydatów}
        for wszystkich  $x \in U$  do
            Updatesupport( $C_k, x$ );
             $L_k := \{ c \in C_k : c.positive\_support \geq minsupport \}$ ;
             $k := k+1$ ;
        end; {while} {powtarzaj dopóki można znaleźć silne zbiory warunków}
    {generuj reguły}
    for all  $L_k$  do {rozważaj  $k$  od 1 do  $maxlength(L_k)$ }
    begin
        for each itemset  $c \in L_k$  do
            if  $c.negative\_support > 0$  {sprawdź warunek akceptacji reguły dla DRSA}
            {lub  $c.positive\_support / (c.positive\_support + c.negative\_support) < \alpha$  dla VC-DRSA}
            then usuń  $c$  z  $L_k$ 
        end;
        for  $k = 2$  to  $maxlength(L_k)$  do begin {test minimalności wewnątrz skumulowanej klasy}
            for każdy  $c \in L_k$  do
                for  $j = k$  downto 2 do
                    for każdy  $c' \in L_j$  do
                        if  $c'$  posiada przynajmniej tak ogólną część warunkową jak  $c$ 
                        and dokładność  $\alpha$  koniunkcji  $c \leq \alpha$  koniunkcji  $c'$  then usuń  $c$  z  $L_k$ 
                    end;
                end;
            end;
        end;
        Utwórz reguły  $R_{\geq}$  na podstawie  $\bigcup_k L_k$ ;
    end {Apriori_domrules}

```

Powyższa procedura jest iteracyjnie uruchamiana dla kolejnych przybliżeń  $\underline{B}(CI_s^{\geq})$ , gdzie  $s = t, \dots, 2$ . Po wygenerowaniu każdego zbioru reguł  $R_{\geq}$  dla przybliżeń  $\underline{B}(CI_s^{\geq})$ ,  $s < t$ , należy sprawdzić minimalność wygenerowanych reguł w stosunku do otrzymanych dla poprzednich przybliżeń. Szczegółowy zapis algorytmu i uwagi o jego implementacji zamieszczono w pracy [238].

Ocena złożoności obliczeniowej algorytmu jest podobna jak w przypadku algorytmu EXPLORE (porównaj rozdział 3). Funkcja złożoności jest rzędu  $O(NF)$ , gdzie  $N = n \cdot m$  jest rozmiarem danych wejściowych,  $F$  jest sumą rozmiarów sil-

nych zbiorów warunków i zbiorów kandydujących (porównaj dyskusję nad kosztami obliczeniowymi oryginalnego algorytmu APRIORI [138]). Jeżeli parametry *minsupport*, *maxlength* silnie ograniczają wielkość  $F$  (czyli liczbę silnych zbiorów kandydujących), to czas obliczania zbioru reguł narasta prawie liniowo ze wzrostem liczby przykładów. Natomiast jeśli użytkownik nie dobierze parametrów ograniczających do danego zbioru przykładów lub zrezygnuje z ich stosowania (szukanie wszystkich reguł), to wielkość  $F$  może być zbyt duża. Zauważmy, że w najgorszym przypadku szukania wszystkich reguł, gdy reguły pokrywają pojedyncze obiekty i wykorzystują wszystkie atrybuty w budowie warunków elementarnych, liczba potencjalnych kandydatów na zbiory warunków  $C_k$  rośnie wykładniczo. Tak jak dyskutowano przy opisie algorytmu EXPLORE, jest to wielkość rzędu  $O(W^k)$ , gdzie  $W$  jest średnią liczbą wartości w dziedzinie kryterium,  $k < m$  – liczbą kryteriów. Rozpatrywanie w proponowanym podejściu w składni reguł warunków ( $f(a_i, x) \geq r_{ai}$ ) lub ( $f(a_i, x) \leq r_{ai}$ ) definiowanych na oryginalnych skalach kryteriów  $a_i$  (np. na dziedzinie liczb rzeczywistych) zwiększa liczbę możliwych operacji połączeń warunków w stosunku do rozpatrywania możliwości połączeń warunków dla atrybutów nominalnych ( $f(a_i, x) = r_{ai}$ ) w oryginalnym algorytmie APRIORI. Podsumowując, stosowanie algorytmu DOMApriori dla poszukiwania wszystkich reguł nie jest sensowne dla dużych tablic decyzyjnych. Ponadto dla danych, w których kryteria liczbowe przyjmują bardzo dużo różnych wartości, dogodnie może być wykonanie pewnego rodzaju wstępnej dyskretyzacji i staranny dobór parametrów *minsupport* oraz *minlength*.

### Kontynuacja przykładu 6.1

Zastosowanie algorytmu DOMApriori w wersji DRSA do indukcji reguł z przykładów przedstawionych w tabeli 6.1, przy parametrze *minsupport* = (2 obiekty), prowadzi do zbioru 15 reguł pewnych oraz 12 reguł przybliżonych. Dla przykładu skumulowana klasa  $Cl_2^{\leq}$  jest teraz opisana za pomocą trzech reguł

jeżeli ( $f(a_1, x) \leq 2, 0$ ) to  $x \in Cl_2^{\leq}$   $\{x_1, x_2, x_3, x_4, x_6, x_7, x_9, x_{10}, x_{12}, x_{13}, x_{14}, x_{15}\}$

jeżeli ( $f(a_2, x) \leq 3, 0$ ) to  $x \in Cl_2^{\leq}$   $\{x_1, x_3, x_4, x_6, x_7, x_9, x_{10}, x_{12}, x_{13}, x_{14}\}$

jeżeli ( $f(a_3, x) \leq 8, 0$ ) to  $x \in Cl_2^{\leq}$   $\{x_3, x_4, x_6, x_7, x_9, x_{10}, x_{12}, x_{13}, x_{14}\}$

Algorytm DOMLEM pozwolił na otrzymanie tylko pierwszej reguły.

Użycie algorytmu DOMApriori w modelu VC-DRSA z  $l = 0,8$  prowadzi do indukcji 30 reguł. Doświadczenia eksperymentalne wskazują, że zastosowanie modelu ze zmienną spójnością prowadzi najczęściej do wzrostu liczby reguł, lecz część z nich pokrywa większą liczbą przykładów niż w modelu DRSA.

### 6.4.3. Inne aspekty indukcji reguł dla wielokryterialnego problemu sortowania

W niektórych zastosowaniach użytkownik może być zainteresowany poszukiwaniem wszystkich reguł. Koszty poszukiwania takiego zbioru reguł mogą być

jednak bardzo wysokie, gdyż funkcja złożoności obliczeniowej dla najgorszego przypadku jest wykładnicza. Algorytm DOMApriori został zaprojektowany w celu poszukiwania satysfakcjonującego zbioru reguł i może być nieskuteczny przy poszukiwaniu wszystkich reguł. Dlatego rozważano alternatywne algorytmy poszukiwania zbioru wszystkich reguł. Zdecydowanie się na indukcję reguł opartych na obiektach bazowych pozwala na skorzystanie z pewnej właściwości tych reguł mówiącej, że gdy reguła o składni *jeśli*  $(f(a_1,x) \geq r_{a_1}) \wedge (f(a_2,x) \geq r_{a_2}) \wedge \dots \wedge (f(a_p,x) \geq r_{a_p})$  *to*  $x \in Cl_t^{\geq}$ , gdzie  $C = \{a_1, \dots, a_p\} \subseteq A$ , jest oparta na obiekcie bazowym  $y \in \underline{C}(Cl_t^{\geq})$  i jest regułą minimalną, wówczas nie istnieje żaden inny obiekt  $z \in \underline{C}(Cl_t^{\geq})$  taki, że  $y D_C z$ . Przez wyrażenie „obiekt  $z$  inny od obiektu  $y$ ” rozumiemy, że w opisie obu obiektów istnieje przynajmniej jedno kryterium  $a_i \in C$  takie, że  $f(a_i,y) \neq f(a_i,z)$ . Powyższa właściwość wynika bezpośrednio z definicji minimalności reguły. Podobne właściwości można sformułować dla reguł typu 2–4. Innymi słowy, reguły oparte na obiektach bazowych oraz wykorzystujące podzbiór kryteriów  $C$  do budowy części warunkowych powinny być budowane na podstawie opisu tych obiektów  $x \in \underline{C}(Cl_t^{\geq})$ , które nie są względem siebie w relacji dominacji. W interpretacji geometrycznej takie obiekty znajdują się w przestrzeni  $V_{a_1} \times V_{a_2} \times \dots \times V_{a_p}$  „na granicy” pomiędzy  $\underline{C}(Cl_t^{\geq})$  a  $Bn_C(Cl_t^{\geq})$ .

Rozważano algorytm poszukiwania wszystkich reguł opartych na obiektach bazowych wykorzystujący powyższą właściwość. W algorytmie systematycznie przeszukuje się przestrzeni podzbiorów kryteriów. Ze względu na wymóg minimalności reguł poszukiwanie obiektów bazowych rozpoczyna się od jednoelementowych zbiorów kryteriów, następnie rozpatruje się podzbiory dwuelementowe itd. aż do momentu, kiedy zwiększanie liczby rozpatrywanych kryteriów nie powoduje generowania nowych minimalnych reguł. Dla podzbioru kryteriów  $C \subseteq A$  oraz przybliżenia  $\underline{C}(Cl_t^{\geq})$  zasadniczy przebieg algorytmu jest następujący:

skonstruuj przybliżenie  $\underline{C}(Cl_t^{\geq})$ ;

**for**  $\forall x \in \underline{C}(Cl_t^{\geq})$  **do**

**begin**

**if** nie istnieje  $z \neq x$  taki, że  $x D_C z$  **then**

wygeneruj regułę  $r$  na podstawie opisu  $x$  za pomocą zbioru kryteriów  $C$ ;

**if** nie istnieje reguła  $r' \in R_{\geq}$  zbudowana na podstawie podzbioru kryteriów  $C' \subset C$  posiadająca przynajmniej tak ogólną część warunkową jak  $r$  **then**  $R_{\geq} := R_{\geq} \cup r$ ;

**end**

Zastosowanie takiego algorytmu do tabeli 6.1 umożliwia odkrycie 30 reguł pewnych i przybliżonych.

Interesującym kierunkiem dalszych badań mogłoby być sprawdzenie możliwości modyfikacji metod wywodzących się z wnioskowania boolowskiego na potrzeby indukcji reguł opartych na obiektach bazowych. W szczególności dotyczyłoby

to algorytmów poszukiwania przybliżonych reduktów lokalnych związanych z obiektami lub tzw.  $\alpha$ -reduktów [153, 158].

Odmiernym problemem jest analiza wielokryterialnych i wieloatrybutowych problemów decyzyjnych. Oznacza to, że reprezentacja problemu sortowania w tablicy  $DT$  wykorzystuje do opisu przykładów zarówno kryteria, jak i atrybuty. Analiza takich problemów jest możliwa poprzez modyfikacje dotychczasowych uogólnień teorii zbiorów przybliżonych opartych na relacji dominacji [62].

Zakładamy, że przykłady opisane są za pomocą zbioru  $B = B^{\succ} \cup B^{\bar{}} \subseteq A$ , gdzie  $B^{\succ}$  jest zbiorem kryteriów, a  $B^{\bar{}}$  jest zbiorem atrybutów.

Z każdym kryterium  $a \in B^{\succ}$  związana jest relacja przewyższania  $S_a$  zgodna z definicją przedstawioną w rozdziale 6.2. Z każdym atrybutem  $a \in B^{\bar{}}$  związana jest relacja nierozróżnialności  $I_a$  zgodna z definicją (2.8) przedstawioną w rozdziale 2.2.

Na podstawie powyższych relacji definiuje się nową relację tzw. in-dominacji (ang. *in-dominance relation*).

Dla każdego podzbioru kryteriów i atrybutów  $B = B^{\succ} \cup B^{\bar{}} \subseteq A$  oraz obiektów  $x, y \in U$  obiekt  $x$  jest w relacji in-dominacji z obiektem  $y$ , co oznaczamy  $xID_{By}$ , wtedy i tylko wtedy, gdy:

$$\forall a \in B^{\succ} xS_a y \text{ oraz } \forall a \in B^{\bar{}} xI_a y. \quad (6.17)$$

Relacja ta pozwala na zdefiniowanie zbiorów  $B$ -dominujących  $D_B^+(x) = \{y \in U : yID_B x\}$  oraz  $B$ -dominowanych  $D_B^-(x) = \{y \in U : xID_B y\}$ . Definicje przybliżeń w podejściu DRSA oraz uogólnieniu VC-DRSA są analogiczne jak w przypadku wyłącznie kryterialnym [238].

Składnia reguł decyzyjnych jest rozszerzeniem poprzednich definicji składni reguł; przykładowo pewne  $D_{\geq}$ -reguły decyzyjne typu 1 mają następującą składnię:

jeśli  $(f(a_1, x) \geq r_{a_1}) \wedge \dots \wedge (f(a_p, x) \geq r_{a_p}) \wedge (f(a_{k+1}, x) = r_{a_{k+1}}) \wedge \dots \wedge (f(a_p, x) = r_{a_p})$  to  $x \in Cl_I^{\geq}$ ,  
gdzie  $B = \{a_1, \dots, a_p\} \subseteq A$ ,  $B^{\succ} = \{a_1, \dots, a_k\}$ ,  $B^{\bar{}} = \{a_{k+1}, \dots, a_p\}$ ;  $(r_{a_1}, \dots, r_{a_p}) \in V_{a_1} \times V_{a_2} \times \dots \times V_{a_p}$ .

Dla innych typów reguł 4–5 w podejściu DRSA oraz reguł w podejściu VC-DRSA składnie rozszerza się w ten sam sposób. Pozostałe definicje związane z regułami są również identyczne jak dla przypadku używania samych kryteriów.

Zaprezentowane dotychczas algorytmy można w prosty sposób zmodyfikować, aby uwzględniać także obecności atrybutów  $B^{\bar{}}$  w opisie przykładów. Modyfikacje algorytmów przedstawiono w [238]. Ogólna zasada polega na rozpatrywaniu jako kandydatów na warunki elementarne oprócz warunków  $Cond := \{(f(a_i, x) \geq r_{a_i}) : \exists x \in K (f(a_i, x) = r_{a_i})\}$  (lub  $Cond := \{(f(a_i, x) \leq r_{a_i}) : \exists x \in K (f(a_i, x) = r_{a_i})\}$ ) dla kryteriów, także warunków o postaci  $Cond := \{(f(a_i, x) = r_{a_i}) : \exists x \in K (f(a_i, x) = r_{a_i})\}$  dla atrybutów. W algorytmie indukcji reguł DOMLEM obie postaci warunków są równouprawnione do „współzawodnictwa” jako najlepsze warunki do dodania do koniunkcji  $P$ . W rezultacie zostanie wybrany lepszy z nich ze względu na funkcję

*evaluate*. Podobnie w przypadku algorytmu DOMApriori obie postaci warunków są wykorzystywane w trakcie budowy kandydatów  $C_k$ . Szczegóły indukcji reguł dla wielokryterialnych i wieloatrybutowych problemów sortowania zostały przedstawione w pracy [238].

Omawiane w tym rozdziale podejścia analizy wielokryterialnych, a także wielokryterialnych i wieloatrybutowych problemów sortowania oraz algorytmy indukcji reguł decyzyjnych zostały zaimplementowane w systemie 4-eMka.

## 7. Złożone systemy klasyfikujące

### 7.1. Motywacje

Większość dotychczasowych badań w zakresie uczenia maszynowego i odkrywania wiedzy w perspektywie klasyfikacyjnej dotyczy tworzenia i eksperymentalnej oceny pojedynczych algorytmów uczących się. W ostatnich latach obserwuje się rosnące zainteresowanie tworzeniem złożonych systemów klasyfikujących. Idea ta polega na integracji wielu pojedynczych algorytmów uczenia się w jeden system klasyfikujący. Celem takiej integracji jest przede wszystkim osiągnięcie lepszej trafności klasyfikacji od otrzymanej w rezultacie użycia oddzielnie pojedynczych klasyfikatorów wchodzących w skład systemu.

Motywacją dla tworzenia takich systemów jest także obserwacja ograniczeń pojedynczych algorytmów. Wiele z tych algorytmów używa różnych reprezentacji przykładów uczących, hipotez oraz strategii przeszukiwania przestrzeni hipotez. Takie algorytmy są więc wyspecjalizowane w rozwiązywaniu pewnej klasy problemów. Wyniki eksperymentalnych badań (np. [148]) wskazują, że nie ma jednego typu algorytmów równie skutecznego dla szerokiego zakresu rozważanych problemów uczenia maszynowego. Odpowiednio przeprowadzona integracja pojedynczych algorytmów może nie tylko zwiększyć skuteczność klasyfikowania, ale także pozwolić na rozwiązywanie większej liczby problemów.

W latach dziewięćdziesiątych powstało wiele prac dotyczących złożonych systemów klasyfikujących, np. [3, 10, 20, 30, 38, 88, 89]. Wyniki eksperymentalne potwierdzają wzrost trafności klasyfikowania dla proponowanych systemów. Integracja jednorodnych lub różnorodnych klasyfikatorów (lub algorytmów uczenia się) może być przeprowadzona różnymi sposobami. Przedyskutujemy to krótko w rozdziale 7.2. Na potrzeby dalszych rozważań wyróżniamy dwie główne kategorie złożonych systemów klasyfikujących:

- systemy wielokrotne,
- systemy hybrydowe.

Pojęcie **systemu wielokrotnego** (odpowiednik terminu angielskiego *multiple model*, *multiple classifier* lub *ensemble of classifiers*)<sup>10</sup> odnosi się do zbioru pojedynczych klasyfikatorów, których odpowiedzi są zagregowane do jednej odpowiedzi całego systemu. Klasyfikatory składowe mogą być jednorodne (homogeniczne) lub

---

<sup>10</sup> W pracy [96] można się spotkać z użyciem terminu „klasyfikator złożony” dla tej kategorii systemów. W niniejszej rozprawie będziemy jednak stosować konsekwentnie termin „systemy wielokrotne”, gdyż dokładniej odzwierciedla to sposób integracji pojedynczych klasyfikatorów. Ponadto uważamy, że termin „złożone systemy” odnosi się do szerszego zakresu różnych sposobów integracji.



różnorodne (heterogeniczne). W pierwszym przypadku klasyfikatory uczone są na zróżnicowanych danych.

Pojęcie **systemu hybrydowego** odnosi się do systemu integrującego w fazie uczenia się przynajmniej dwa różne modele. Różnorodność modeli dotyczy takich aspektów jak inne reprezentacje hipotez oraz strategii ich poszukiwania. System hybrydowy „dopasowuje się” do danych za pomocą wielu różnych reprezentacji wiedzy. Zauważmy, że system hybrydowy jest pojęciem innym niż wielokrotny system heterogeniczny. W systemie wielokrotnym występuje zbiór „gotowych” klasyfikatorów nauczonych z danych najczęściej niezależnie od siebie. Integracja dotyczy agregacji ich odpowiedzi. W przypadku systemu hybrydowego integracja występuje przede wszystkim w fazie uczenia się. Dotyczy ona różnych algorytmów uczenia się, które są silnie ze sobą połączone w jedną strategię.

Autor rozprawy wraz ze współpracownikami prowadził badania nad złożonymi systemami klasyfikującymi [97, 99, 100, 237, 247]. W niniejszym rozdziale syntetycznie omówimy dwie propozycje, w których wykorzystuje się regułową reprezentację wiedzy. Dotyczą one:

- hybrydowego systemu integrującego uczenie się reguł decyzyjnych i przypadków; reguły odnoszą się do typowych sytuacji decyzyjnych, podczas gdy pojedyncze przypadki reprezentują sytuacje trudne i wyjątkowe;
- klasyfikatora wielokrotnego  $n^2$ , który jest systemem specjalizowanym do rozwiązywania wieloklasowych problemów uczących.

W konstruowaniu części regułowej powyższych systemów wykorzystuje się algorytmy indukcji reguł EXPLORE i MODLEM omawiane w ramach realizacji innych celów szczegółowych rozprawy (patrz rozdziały 3 i 4).

## 7.2. Przegląd metod tworzenia złożonych systemów klasyfikujących

Zasadnicze pytanie dotyczy sposobów budowania złożonych systemów, tak aby pozwalały na osiągnięcie wyższej trafności klasyfikacji niż klasyfikatory składowe działające oddzielnie. Zgodnie z opinią niektórych badaczy klasyfikatory powinny działać z pewnym stopniem niezgodności, tzn. popełniane przez nie błędy klasyfikowania powinny być wzajemnie nie skorelowane [38, 88]. Rozważania teoretyczne nad wielokrotnymi systemami klasyfikującymi dla problemów dwuklasowych wskazują, że jeśli błędy klasyfikatorów składowych nie są ze sobą skorelowane i każdy klasyfikator popełnia błąd poniżej 50%, to całkowity błąd systemu maleje proporcjonalnie do liczby użytych klasyfikatorów [3].

Agregacja odpowiedzi pojedynczych klasyfikatorów w globalną odpowiedź systemu dokonywana jest w sposób grupowy albo specjalizowany. W pierwszym sposobie wszystkie klasyfikatory biorą udział w wypracowaniu decyzji końcowej dla nowego obiektu. W drugim sposobie wyznacza się te klasyfikatory, których specjalizacja obejmuje charakterystykę klasyfikowanego obiektu. Sama agregacja

realizowana jest najczęściej poprzez głosowanie proste (głos każdego klasyfikatora ma równy priorytet) lub ważone.

Istnieje wiele sposobów tworzenia złożonych systemów klasyfikujących. Część z nich polega na modyfikowaniu danych uczących dla poszczególnych jednorodnych klasyfikatorów, inne zakładają różnicowanie samych modeli klasyfikujących. Niewiele jest dotychczas prac przeglądowych dotyczących różnych propozycji [58, 96]. Poniżej zamieszczamy skrótowo klasyfikację podaną przez autora w [237]:

### **Systemy wielokrotne oparte na zróżnicowanych klasyfikatorach**

#### ***Generalizacja stosowa***

Jest to model hierarchicznego, wielowarstwowego klasyfikatora złożonego zaproponowany przez Wolperta [265]. Klasyfikatory składowe pierwszej warstwy uczone są na oryginalnych danych, a ich odpowiedzi są danymi wejściowymi do kolejnej warstwy. W ostatniej warstwie umieszczony jest pojedynczy klasyfikator wypracowujący ostateczną decyzję. Klasyfikatory wyższych warstw starają się minimalizować błędy popełniane przez klasyfikatory z niższych warstw.

#### ***Metauczenie się***

Modele metauczenia się zostały wprowadzone przez Chana i Stolfo [30] w dwóch architekturach: (ang. *combiner* oraz *arbiter*). W obu wykorzystuje się tzw. metaklasyfikator, który uczony jest z metadanych powstałych na podstawie odpowiedzi klasyfikatorów bazowych, tj. różnorodnych klasyfikatorów uczonych bezpośrednio na przykładach. Pierwsza architektura jest zbliżona do modelu generalizacji stosowej, przy czym dane uczące mogą zawierać więcej informacji niż tylko odpowiedzi klasyfikatorów bazowych. W architekturze arbitra metaklasyfikator jest uczony tzw. reguły arbitrażu na podzbiorze trudnych przykładów, tak aby rozstrzygać niejednoznaczne decyzje klasyfikatorów bazowych. Istnieją też architektury drzew wielu arbitrów lub „combinerów”.

### **Systemy wielokrotne oparte na jednorodnych klasyfikatorach**

#### ***Technika „bagging”***

Model wprowadzony przez Breimana [20] wykorzystuje ideę powielania zbioru uczącego za pomocą wielokrotnego próbkowania techniką „bootstrapping”. Każdy powielony zbiór ma tę samą wielkość co oryginalny, lecz w wyniku próbkowania niektóre przykłady uczące są kopiowane, podczas gdy inne nie mogą się wielokrotnie powtórzyć. Z każdego z tych zbiorów generuje się klasyfikator tym samym algorytmem uczącym. Odpowiedzi klasyfikatorów są agregowane poprzez proste głosowanie.

#### ***Technika „boosting”***

Podana została poprzez Freunda i Schapirego [54, 196]. Idea metody polega na skojarzeniu z każdym obiektem uczącym wagi, której wartość odzwierciedla jego

istotność. Algorytm uczenia się uruchamiany jest iteracyjnie nad modyfikowanym zbiorem uczącym. Modyfikacje dotyczą zmiany wag w kolejnych kopiach zbioru uczącego, które skupiają działanie algorytmu uczącego wokół tzw. trudnych przykładów, to jest tych, dla których klasyfikatory popełniały błędy w poprzednich iteracjach. Odpowiedź końcowa systemu opiera się na ważonym głosowaniu (tutaj wagi głosujących klasyfikatorów uzależnione są od błędów popełnianych przez odpowiednie klasyfikatory składowe). Porównanie eksperymentalne złożonych klasyfikatorów zbudowanych za pomocą technik „bagging” oraz „boosting” przedstawiono w [10, 38, 186]. Według autorów tych prac techniki te są skuteczne, jeśli używa się tzw. niestabilnych algorytmów uczących się, tzn. takich, które tworzą różne klasyfikatory w odpowiedzi na małe zmiany w zbiorze uczącym.

### ***Systemy z podziałem danych***

Metody te uczą się klasyfikatorów na podzielonych podzbiorach danych. Podział ten może być losowy lub wynikać z rozproszenia danych. Same podzbiory danych mogą być rozłączne lub częściowo powielone. Wynik końcowy jest „uśrednieniem” wyników częściowych [31, 95, 269]. Część podanych propozycji uwzględnia aspekt eksploracji wielkich baz danych (porównaj np. [86, 262]).

### ***Metoda ECOC***

Jest to specjalizowany model służący do rozwiązywania wieloklasowych problemów uczenia [39] (tzn. takich, w których liczba klas jest większa niż dwa). Główna idea polega na kodowaniu binarnym etykiet poszczególnych klas decyzyjnych. Klasyfikator ECOC (ang. *error-correcting output codes*) uczy się funkcji odpowiadających określonym bitom zakodowanej klasy.

Rozważa się też inne sposoby grupowania różnorodnych klasyfikatorów dla problemów wieloklasowych, np. „jeden-z-wielu” [30], *pairwise polychotomous classification* [55] czy *pairwise coupling* [89]. Więcej informacji na ich temat można znaleźć w [96].

### ***Systemy hybrydowe***

Motywacje dla tworzenia systemów hybrydowych wynikają z obserwacji, że dla każdego algorytmu uczącego istnieje pewna klasa problemów, dla których jest skuteczny. Dla pozostałych problemów algorytm może być mniej skuteczny od innych algorytmów. Ponadto niektóre dane uczące mogą mieć złożoną strukturę wymagającą modelowania więcej niż jedną formą reprezentacji wiedzy. W tych sytuacjach interesujące jest tworzenie systemu hybrydowego, który przybliży różne obszary danych za pomocą różnorodnych reprezentacji. W rezultacie synergii połączonych modeli cały system ma działać skuteczniej.

Badania nad tworzeniem różnych systemów hybrydowych zapoczątkowano już w latach osiemdziesiątych. Istnieje wiele propozycji łączących np. paradygmaty uczenia się drzew decyzyjnych czy reguł decyzyjnych z uczeniem się na wcześniej

zapamiętanych przypadkach lub z uczeniem się sieci neuronowych. Przegląd niektórych propozycji dostępny jest w [58, 146].

### 7.3. System hybrydowy integrujący uczenie się reguł decyzyjnych oraz wcześniej zapamiętanych przypadków

#### 7.3.1. Architektura systemu

W niniejszym rozdziale omówiono propozycje systemu hybrydowego zaproponowanego w [247]. Motywacje do jego stworzenia wynikają z przesłanek omówionych poprzednio oraz z rozważań nad psychologicznymi aspektami podejmowania decyzji podanych przez Risbecka i Shanka w [190]. Stwierdzili oni między innymi, że:

„jeśli pewne działanie jest powtarzane dostatecznie często, to staje się regułą. Ludzie nie wnioskujeją z wcześniejszych przypadków, kiedy dostępne są dobrze uwiarygodnione reguły [...]. Kiedy reguła zawiedzie, alternatywą dla człowieka jest odszukanie przypadku, który obejmuje tę sytuację”.

Prowadzi to sformułowania następującego schematu działania przy klasyfikowaniu nowych przypadków:

**jeśli** nowy przypadek spełnia choć jedną regułę ze zbioru reguł  $R$   
**to** zastosuj rozwiązanie wskazywane przez regułę o największym priorytecie  
**w przeciwnym razie** zaadaptuj rozwiązanie wynikające z najbardziej podobnego przypadku ze zbioru wyjątków  $Ex$ ,

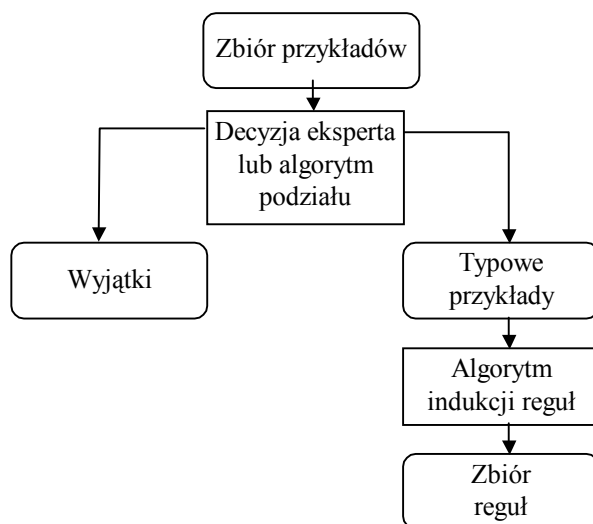
gdzie zbiór ogólnych reguł  $R$  opisuje typowe sytuacje decyzyjne, a zbiór wyjątków  $Ex$  zawiera pojedyncze przykłady, które są odstępstwem od tych reguł. Powyższy schemat jest zaadaptowany do zbudowania systemu hybrydowego zawierającego dwie reprezentacje wiedzy: reguły decyzyjne i pojedyncze przypadki.

Zasadnicze pytanie dotyczy sposobu pozyskania powyższych reprezentacji wiedzy. Przykładowo można założyć, że za pomocą pewnych metod można dokonać podziału zbioru uczącego na dwie rozłączne części:

- zbiór typowych przykładów  $UG$ ,
- zbiór wyjątków  $Ex$ .

Następnie, stosując jeden z typowych algorytmów, można dokonać indukcji reguł decyzyjnych ze zbioru  $UG$ . Przedstawiono to na rysunku 7.1.

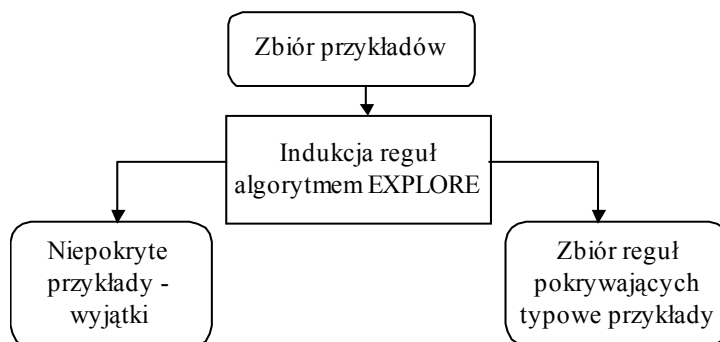
Podział zbioru przykładów nie jest łatwym zagadnieniem. Może wynikać z sugestii eksperta, analizy statystycznej (np. być rezultatem analizy skupień) lub rozwiązania heurystycznego. Pierwsze podejście zastosowano na przykład w [250]. Istnieją też nietypowe sposoby indukcji reguł, np. Domingos w systemie RISE wprowadza jednolitą reprezentację reguł i przypadków [40]. Przypadki są maksymalnie szczegółowymi regułami, które w trakcie procesu uczenia się mogą być uogólniane do reguł. Specyficzne podejście indukcji reguł i przypadków stosuje się także w systemie FCLS [272].



Rysunek 7.1. Technika podziału w tworzeniu systemu hybrydowego

#### 4.3.2. Proces uczenia się w systemie hybrydowym

W propozycji systemu hybrydowego użyto specyficznego algorytmu indukcji reguł decyzyjnych EXPLORE, który omówiono w rozdziale 3. Przypomnijmy, że pozwala on na indukcję z danych zbioru reguł ogólnych i dostatecznie silnych w sensie miary wsparcia (lub pokrycia). Takie reguły reprezentują więc ogólne przypadki lub, innymi słowy, typowe sytuacje decyzyjne. Przykłady nie pokryte przez wygenerowane reguły mogą być traktowane jako wyjątki i tworzyć bazę wyjątków *Ex*. Proces ten zilustrowano na rysunku 7.2.



Rysunek 7.2. Tworzenie reprezentacji wiedzy za pomocą algorytmu EXPLORE

Zauważmy, że wybór algorytmu EXPLORE i właściwy dobór jego parametrów pozwala rozwiązać problem podziału zbioru uczącego. Użycie standardowego algorytmu indukcji, np. minimalnego zbioru reguł, nie byłoby właściwe dla tej sytu-

acji. Opis algorytmu EXPLORE podano w rozdziale 3.3.1. Przypomnijmy, że algorytm może być sterowany za pomocą parametrów związanych ze wsparciem lub pokryciem reguły, z jej dokładnością czy długością części warunkowej. W systemie hybrydowym wybrano tylko jeden parametr  $l$  minimalnego względnego pokrycia reguły związanego z warunkiem zatrzymania algorytmu  $SC$  (wyrażenie 3.9). Tak jak diskutowano w rozdziałach 3.3.2, właściwy dobór progu  $l$  dla tego warunku może doprowadzić do zbudowania w miarę prostego i ogólnego zbioru silnych reguł decyzyjnych pozwalających równocześnie na osiągnięcie akceptowalnej trafności klasyfikacji.

Należy zauważyć, że w przypadku systemu hybrydowego schemat klasyfikowania nowych przykładów jest odmienny od omawianego w rozdziale 2.3. Nowo klasyfikowany obiekt  $e$  jest dopasowywany do części warunkowych ogólnych reguł  $R$ . Możliwe są trzy rezultaty dopasowania:

- Obiekt  $e$  jest dopasowany do dokładnie jednej reguły z  $R$ ; decyzja klasyfikacyjna jest podejmowana na jej podstawie.
- Obiekt  $e$  jest dopasowany do więcej niż jednej reguły z  $R$ ; stosuje się jedną z strategii rozwiązywania problemu wielokrotnego dopasowania (porównaj rozdział 2.3 oraz [247]).
- Obiekt  $e$  nie jest dopasowany do żadnej z reguł.

W trzecim przypadku decyzja klasyfikacyjna jest określana na podstawie podobieństwa obiektu  $e$  do  $k$  najbliższych przypadków z bazy wyjątków  $Ex$ . Finalna decyzja jest klasą większościową pośród  $k$  najbliższych przypadków.

Najważniejszym problemem jest właściwy dobór parametru algorytmu EXPLORE, tj. progu minimalnego względnego pokrycia reguły  $l$ . Przyjęto rozwiązanie automatycznej procedury przedstawionej w rozdziale 3.3.1 pod nazwą *Tunethresholds*, przy czym miara oceny dotyczy obecnie całego systemu hybrydowego, tj. reguły  $R$  i wyjątków  $Ex$ , a nie tylko samej części regułowej, jak poprzednio. W stosowanej funkcji *Classification* rozważa się jako podstawowe kryterium trafność klasyfikacji, a jako drugie minimalizację liczby reguł. Warunek *stop* związany jest ze spadkiem miary wrażliwości (wzór 2.30) w klasie, dla której indukowane są reguły. Początkowe wartości progów z  $\mathcal{L}$  są ustalane na podstawie wyniku działania standardowego algorytmu indukcji minimalnego zbioru reguł. Więcej szczegółów przedstawiono w pracy [247].

### 7.3.3. Ocena eksperymentalna

W celu oceny skuteczności proponowanej koncepcji systemu hybrydowego dokonano analizy kilku realnych zbiorów danych [247, 264]. Omówimy dokładniej wyniki jego zastosowania do analizy danych finansowych. Problem dotyczył oceny wniosków firm ubiegających się o kredyt komercyjny. Przykład uczący odpowiadający wnioskowi kredytowemu był opisany 35 atrybutami i mógł być przydzielony do jednej z 5 grup tzw. ryzyka bankowego.

Początkowe wartości progów  $\mathcal{L}$  zostały ustalone na podstawie wyniku działania algorytmu LEM2. Następnie zastosowano procedurę *Tunethresholds*. Końcowa architektura systemu składała się z 70 reguł i 15 przykładów interpretowanych jako wyjątki. Średnie względne pokrycie reguły było wysokie – 56% (oznacza to, że każda z reguł opisywała ponad połowę przykładów ze swojej klasy), a średnia długość reguły wynosiła 2,6 warunku elementarnego. Trafność klasyfikowania wynosiła 81%. Dla badanego zastosowania jest to najlepszy osiągnięty dotychczas wynik. Algorytm EXPLORE i systemy uczenia się na wcześniej zapamiętanych przypadkach IBL1-3 użyte niezależnie od siebie prowadzą do niższych trafności klasyfikacji. Podobnie studium porównawcze z innymi pojedynczymi algorytmami indukcji reguł (C4.5, LEM2, RISE) wykazuje przewagę systemu hybrydowego. Zbiorcze wyniki zawarte są w tabeli 7.1. Należy też zauważyć, że proponowany system pozwolił osiągnąć najlepszą wartość miary wrażliwości w klasach dużego ryzyka bankowego, co jest ważne w tym zastosowaniu. Szczegółowa dyskusja osiągniętych wyników przedstawiona jest w pracy [247].

**Tabela 7.1.** Ocena działania systemu hybrydowego oraz innych pojedynczych algorytmów

System	Trafność klasyfikowania [%]	Wielkość: liczba reguł i/lub przypadków
System hybrydowy	81,1	70 reg./15 przyp.
Algorytm EXPLORE	76,67	64 reguł
IBL1	72,22	89 przypadków
IBL2	60	38 przypadków
IBL3	74,44	22 przypadków
LEM2	73,33	16 reguł
C4.5tree	73,33	23 węzłów i liści
C4.5rules	66,7	9 reguł
RISE	72,22	72 reguł i przyp.

W pracy [264] rozważano też inne zbiory testowe. Otrzymane wyniki wskazują na przyrost trafności klasyfikowania w przypadku analizy kilku medycznych zbiorów danych charakteryzujących się silnym niezrównoważeniem liczości klas decyzyjnych. Z drugiej strony, system nie prowadził do przyrostów trafności klasyfikacji w przypadku „dobrze zdefiniowanych” danych, takich jak np. *Iris*, *Bank* czy *Buses*, dla których możliwe jest osiągnięcie bardzo dobrych wyników z zastosowaniem pojedynczych algorytmów. Jest to w pewnym stopniu zgodne z przesłankami stojącymi za systemem hybrydowym. Wydaje się, że może on być skutecznym rozwiązaniem w przypadku analizy danych o złożonej strukturze, które wymagają modelowania za pomocą różnych reprezentacji hipotez.

Zauważmy ponadto, że konstrukcja proponowanego systemu jest zgodna z ludzkim sposobem wnioskowania. Oznacza to także, że pozyskana reprezentacja

wiedzy jest potencjalnie łatwiejsza do analizy przez użytkownika, a także umożliwia przedstawienie wyjaśnienia proponowanych decyzji klasyfikacyjnych.

Ograniczenie zastosowania powyższego systemu hybrydowego może się wiązać z kosztami obliczeniowymi. Wynika to zarówno z kosztów samego algorytmu indukcji reguł, które, jak dyskutowano w rozdziale 3.3.3, mogą być jednak ograniczone przez dobór dodatkowych warunków zatrzymania, jak i z konieczności wielokrotnego powtarzania obliczeń w celu doboru wartości parametru minimalnego względnego pokrycia reguł.

## 7.4. Klasyfikator wielokrotny $n^2$

### 7.4.1. Architektura i zasada działania

Omawiany w tym rozdziale klasyfikator  $n^2$  został przedstawiony przez Jelonka i autora rozprawy w [97] (patrz także [96, 99, 100]). Jest to klasyfikator wielokrotny przeznaczony do rozwiązywania  $n$ -klasowych problemów uczenia, gdzie  $n > 2$  jest liczbą klas decyzyjnych.

Zauważmy, że pomimo możliwości stosowania algorytmów standardowych (np. C4.5, CN2, LEM2) w odniesieniu do problemów wieloklasowych, zaproponowano dotychczas wiele rozwiązań specjalizowanych, takich jak ECOC [39], metaklasyfikator „jeden z wielu” [30], *pairwise polychotomous classification* [55], *pairwise coupling* [89] (przeglądu różnych rozwiązań dokonano w [96]). Jak wskazują wyniki eksperymentów obliczeniowych, metody specjalizowane mogą prowadzić do lepszej trafności klasyfikowania niż algorytmy standardowe.

Motywacją, która doprowadziła do stworzenia klasyfikatora  $n^2$ , jest hipoteza, zgodnie z którą uczenie się rozróżniania przykładów wyłącznie z par klas decyzyjnych może być łatwiejsze i skuteczniejsze w sensie predykcji klasyfikacji nowych obiektów niż uczenie się rozróżniania obiektów z wszystkich klas decyzyjnych.

Główna idea klasyfikatora polega na podziale jednego  $n$ -klasowego problemu uczenia maszynowego na  $n \cdot (n-1)/2$  problemów uczenia się binarnych klasyfikacji. Dla każdego podproblemu rozróżniania obiektów należących do odpowiedniej pary klas  $(i, j)$ ,  $i, j = 1, \dots, n$ ,  $i \neq j$ , konstruowany jest niezależny klasyfikator binarny  $C_{ij}$ . Z uwagi na interpretację graficzną klasyfikatorów binarnych w postaci macierzy kwadratowej  $n \times n$  (patrz rysunek 7.3) nazwano tę strukturę klasyfikatorem  $n^2$ . Każdy binarny klasyfikator  $C_{ij}$ , odpowiadając tylko za rozróżnienie obiektów należących do związanej z nim pary klas, uczony jest wyłącznie na przykładach należących do klasy  $i$  lub klasy  $j$ . Klasyfikator podejmuje decyzję binarną, wskazując, czy nowo klasyfikowany obiekt  $e$  należy do klasy  $i$  lub klasy  $j$ . W dalszych notacjach przyjmujemy, że  $C_{ij}(e)$  oznacza decyzję binarnego klasyfikatora  $C_{ij}$  co do klasyfikacji obiektu  $e$ .

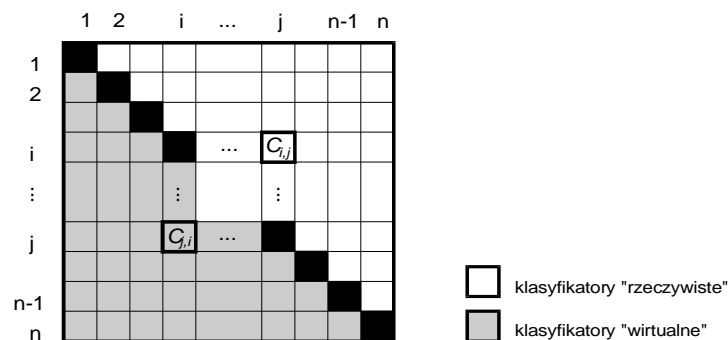
Niech  $C_{ij}(e) = 1$  oznacza, że obiekt  $e$  jest zaklasyfikowany przez  $C_{ij}$  do klasy  $i$ -tej; w przeciwnym razie (tzn.  $C_{ij}(e) = 0$ )  $e$  jest sklasyfikowany do  $j$ -tej klasy. Zauważmy, że klasyfikatory  $C_{ij}$  oraz  $C_{ji}$  (gdzie  $i, j = 1, \dots, n$ ,  $i \neq j$ ) rozwiązują ten sam



problem dyskryminacji obiektów należących do klasy  $i$ -tej bądź klasy  $j$ -tej. Różnią się tylko odpowiedziami dla nowo klasyfikowanych obiektów  $e$ , tzn. jeśli  $C_{ij}(e) = 1$ , to  $C_{ji}(e) = 0$  oraz jeśli  $C_{ij}(e) = 0$ , to  $C_{ji}(e) = 1$ . Przyjmuje się, że oba klasyfikatory pełnią funkcję komplementarną i na podstawie odpowiedzi jednego z nich można określić klasyfikację drugiego, co wyrażamy wzorem:

$$C_{ij}(e) = 1 - C_{ji}(e) \quad (7.1)$$

W konstruowaniu klasyfikatora  $n^2$  wystarczy implementować wyłącznie po jednym klasyfikatorze dla każdej pary klas. Nie rozważa się także konstrukcji klasyfikatorów  $C_{ij}$  dla  $i = j$ . Oznacza to, że wystarczy rozważyć  $(n^2 - n)/2$  klasyfikatorów binarnych odnoszących się do każdej kombinacji par klas decyzyjnych. Architektura klasyfikatora złożonego przedstawiono w postaci macierzy na rysunku 7.3, gdzie wyróżniono rzeczywiste klasyfikatory  $C_{ij}$  oraz „wirtualne” (tj. nieimplementowane) klasyfikatory  $C_{ji}$ , których odpowiedzi wylicza się według wzoru 7.1.



Rysunek 7.3. Architektura złożonego klasyfikatora  $n^2$  przedstawiona w postaci macierzy jednorodnych klasyfikatorów binarnych

W dotychczasowych pracach [96, 97, 99, 100] rozważano użycie tego samego typu algorytmu uczenia do budowy jednorodnych klasyfikatorów binarnych.

Klasyfikacja nowego obiektu  $e$  przez system polega na niezależnej klasyfikacji tego obiektu przez wszystkie klasyfikatory binarne (rzeczywiste i „wirtualne”). Końcowa decyzja systemu jest wynikiem agregacji wszystkich binarnych odpowiedzi. Najprostszą regułą agregacji jest zasada „zwycięzca bierze wszystko”, tzn. wybiera się tę klasę decyzyjną, która wygrywa większość porównań parami [55]. Analizę tej sytuacji w przypadku idealnie działających klasyfikatorów binarnych przeprowadzono w [96]. Skuteczność klasyfikacyjna klasyfikatorów składowych może być jednak zróżnicowana, gdyż każdy z nich był uczony na przykładach z innej pary klas decyzyjnych. Należy oceniać wiarygodność odpowiedzi poszczególnych klasyfikatorów. W dotychczasowych pracach [97] przyjęto, że z każdym

klasyfikatorem binarnym  $C_{ij}$  skojarzony jest współczynnik wiarygodności  $P_{ij}$  zdefiniowany następująco:

$$P_{ij} = \frac{ncor_i}{nerror_{ij} + ncor_i} \quad (7.2)$$

gdzie  $ncor_i$  jest liczbą poprawnie sklasyfikowanych obiektów z  $i$ -tej klasy decyzyjnej, a  $nerror_{ij}$  jest liczbą błędnie zaklasyfikowanych obiektów z  $j$ -tej klasy do klasy  $i$ -tej. Obliczanie współczynników wiarygodności jest przeprowadzane w fazie uczenia klasyfikatorów, tzn. na zbiorze uczącym.

Proces klasyfikacji nowego obiektu  $e$  przez klasyfikator złożony  $n^2$  ma następujący przebieg:

1. Sklasyfikuj nowy obiekt  $e$  przez wszystkie klasyfikatory binarne  $C_{ij}$  ( $i < j$ ,  $i = 1, \dots, n$ ), i oblicz ich odpowiedzi  $C_{ij}(e)$ .
2. Wyznacz odpowiedzi „wirtualnych” klasyfikatorów  $C_{ij}$  ( $i > j$ ) na podstawie wyrażenia (7.1).
3. Dla każdej klasy  $i$  ( $i = 1, \dots, n$ ), oblicz wartość wyrażenia:

$$S_i = \sum_{\substack{j=1 \\ j \neq i}}^n P_{ij} \cdot C_{ij}(e) \quad (7.3)$$

4. Znajdź największą wartość wyrażenia  $S_i$  i zwróć jej indeks  $i$  jako końcową decyzję klasyfikacyjną (w przypadku więcej niż jednej klasy o maksymalnej wartości  $S_i$  wybierz arbitralnie indeks  $i$ ).

W pracach [96, 97] przeprowadzono analizę złożoności pamięciowej i czasowej tego systemu. Ze względu na konieczność replikowania obiektów z klasy  $i$ -tej dla klasyfikatorów  $C_{ij}$  liczba obiektów wzrasta  $(n-1)$  razy w stosunku do pojedynczego algorytmu ( $n$  jest liczbą klas). Jelonek [96] podaje, że funkcja złożoności jest rzędu  $O(n^2 \cdot BC)$ , gdzie  $BC$  jest złożonością klasyfikatora binarnego.

Podobne inspiracje do tworzenia złożonego klasyfikatora opartego na rozwiązaniu problemów dwuklasowych (ang. *pairwise classification*) zaproponował także Friedman [55]. Pokazał on, że stosowana przez niego reguła głosowania jest równoważna z zastosowaniem reguły Bayesa. Podane przez niego wyniki eksperymentalne (w przypadku użycia analizy dyskryminacyjnej lub drzew CART do budowy klasyfikatorów składowych) wskazywały na przewagę złożonego klasyfikatora nad pojedynczym użyciem odpowiedniego algorytmu w wersji wiele

klasowej. Odmiennej schemat klasyfikacji w podobnej architekturze zaproponowali Hastie i Tibshirani w tzw. koncepcji *pairwise coupling* [89], gdzie na podstawie prawdopodobieństw wynikających z porównań parami poszukuje się estymat łącznego rozkładu prawdopodobieństw przynależności klasyfikowanego obiektu do

wszystkich klas. Interesujące wydają się także dyskusje przeprowadzone w [55, 97, 237] w celu odpowiedzi, dlaczego złożone klasyfikatory tego typu mogą dla danych o określonej charakterystyce być skuteczniejsze niż pojedyncze użycie standardowego algorytmu. Poniżej skomentujemy niektóre z przytaczanych tam argumentacji.

Potencjalną wadą podejścia złożonego jest rozwiązywanie dwuklasowych problemów uczenia się na mniejszej liczbie przykładów, niż ma do dyspozycji pojedynczy algorytm wieloklasowy. Jest to jednak rekompensowane faktem, że najczęściej łatwiej jest znaleźć prostsze funkcje dyskryminujące (często liniowe) pomiędzy dwoma klasami niż bardziej złożoną funkcję (często nieliniową) pozwalającą rozróżniać obiekty dla wielu klas. Friedman stwierdza [55], że dla wielu zadań uczenia się rzeczywiste funkcje opisujące klasy decyzyjne mogą mieć bardzo złożoną postać. Standardowe, wieloklasowe algorytmy uczenia się mogą mieć znaczne trudności z ich poprawną aproksymacją. W tych sytuacjach rozróżnianie klas parami może być korzystniejsze. Ponadto zauważa się, że dla takich złożonych zadań mogą występować niestabilności rozwiązań generowanych przez standardowy algorytm, tzn. algorytm uczący generuje zróżnicowane wyniki końcowe nad zmieniającymi się losowo zbiorami uczącymi o tych samych rozmiarach. W rezultacie uśredniony wynik może cechować duża wariancja. W stosunku do klasyfikatora złożonego przypuszcza się, że mniej skomplikowanych funkcji dyskryminujących problemów dwuklasowych można się nauczyć z mniejszą wariancją niż dla jednego problemu wieloklasowego. Zjawisko to jest także zależne od doboru algorytmu uczącego dla klasyfikatorów bazowych, a zwłaszcza od jego możliwości dostrojenia się (poprzez modyfikacje parametrów algorytmu lub odpowiednie strategie wewnętrzne związane np. z wyborem atrybutów) do charakterystyki konkretnych problemów dwuklasowych.

Podsumowując, skuteczność klasyfikacyjna proponowanego klasyfikatora  $n^2$  zależy od specyfiki danych i problemu wieloklasowego, rozmiarów zbioru uczącego oraz od samego algorytmu uczącego użytego do budowy klasyfikatorów binarnych.

#### **7.4.2. Ocena stosowania różnych algorytmów uczących w budowie klasyfikatorów binarnych**

Istotnym elementem konstrukcji klasyfikatora  $n^2$  jest wybór algorytmu uczącego do tworzenia jednorodnych klasyfikatorów binarnych. Z punktu widzenia niniejszej rozprawy jesteśmy zainteresowani przede wszystkim sprawdzeniem przydatności do tego celu algorytmów indukcji reguł decyzyjnych. Z uwagi na ograniczenia związane z większymi kosztami obliczeń niż w przypadku standardowych algorytmów, konieczne jest rozważanie algorytmów o wielomianowej funkcji złożoności. Z tych powodów zdecydowano się wykorzystać algorytm MODLEM omówiony szczegółowo w rozdziale 4. Innym czynnikiem tego wyboru jest też zdolność tego algorytmu do przetwarzania

zdolność tego algorytmu do przetwarzania atrybutów liczbowych w danych użytych w studium eksperymentalnym.

Ponadto w innych badaniach eksperymentalnych [97, 99] sprawdzano przydatność trzech innych typów algorytmów uczenia się z przykładów, tj.:

- algorytmów indukcji drzew decyzyjnych,
- sztucznych sieci neuronowych,
- uczenia się z wcześniej zapamiętanych przypadków na podstawie  $k$  najbliższych sąsiadów ( $k$ -NN).

W pierwszym przypadku użyto algorytmu indukcji binarnych drzew zgodnego z systemem ASSISTANT86 [28]. W zakresie parametryzacji opcji redukcji rozmiarów drzewa użyto najlepszych wartości omówionych w [28]. W przypadku sieci neuronowych badano klasyczną architekturę wielowarstwową z neuronami nieliniowymi uczonymi algorytmem wstecznej propagacji błędów. Testowano różne topologie sieci, aby znaleźć najlepszą topologię dla danego zbioru danych. Więcej szczegółów omówiono w [99, 100]. Dobór właściwej topologii sieci oraz parametrów był najbardziej kosztowny obliczeniowo ze wszystkich porównywanych metod budowy klasyfikatorów binarnych. Dla trzeciego algorytmu użyto algorytmu uczenia się na wcześniej zapamiętanych przypadkach. Stosowana miara odległości była zgodna z podaną w [2], a rozważana liczba sąsiadów  $k$  była równa 1. Wszystkie podejścia porównano w studium eksperymentalnym na typowych zbiorach danych pochodzących przede wszystkim z Univeristy of California w Irvine [14] oraz z zastosowań zbiorów przybliżonych. Charakterystyka danych podana jest w tabeli 7.2.

**Tabela 7.2.** Charakterystyka danych użytych w eksperymentalnej ocenie użycia różnych algorytmów w budowie klasyfikatorów binarnych

Zbiór danych	Liczba przykładów	Liczba klas	Liczba atrybutów
<i>Automobile</i>	159*	6	25
<i>Cooc</i>	700	14	22
<i>Ecoli</i>	336	8	7
<i>Glass</i>	214	6	9
<i>Hist</i>	700	14	17
<i>Meta-data</i>	528	5*	20
<i>Primary Tumor</i>	339	21	17
<i>Soybean-large</i>	542*	14*	35
<i>Vowel</i>	990	11	10
<i>Yeast</i>	1484	10	8

Symbol \* oznacza wykonanie drobnych modyfikacji danych w stosunku do danych oryginalnych – patrz [99].

Kryterium oceny przydatności algorytmu uczącego była trafność klasyfikowania. Estymowano ją na podstawie techniki 10-krotnej walidacji krzyżowej z opcją losowania warstwowego (ang. *stratified 10-fold cross-validation*).

Dla każdego ze zbiorów danych porównywano klasyfikator  $n^2$  z pojedynczym użyciem algorytmu uczącego w wersji wieloklasowej. Znaczenie różnicy w trafności klasyfikowania badano za pomocą testu *t*-Studenta dla wersji zmiennych zależnych. Przyjęto poziom istotności  $\alpha = 0,05$ .

**Tabela 7.3.** Porównanie trafności klasyfikowania systemu  $n^2$  opartego na klasyfikatorach regułowych oraz pojedynczego użycia algorytmu MODLEM

Zbiór danych	Trafność klasyfikowania (%)		Poprawa (%) $n^2$ vs MODLEM
	MODLEM	$n^2_{\text{MODLEM}}$	
<i>Automobile</i>	89,62 ± 1,3	90,69 ± 1,5	1,57*
<i>Cooc</i>	55,57 ± 0,7	59,51 ± 1,1	3,94
<i>Ecoli</i>	78,63 ± 0,8	81,7 ± 1,8	1,07*
<i>Glass</i>	72,62 ± 2,2	74,02 ± 1,5	1,4
<i>Hist</i>	70,96 ± 2,2	74,1 ± 1,3	3,5
<i>Meta-data</i>	45,27 ± 0,9	49,96 ± 1,1	4,6
<i>Primary Tumor</i>	40,9 ± 1,5	45,6 ± 1,9	4,9
<i>Soybean-large</i>	90,61 ± 0,9	91,75 ± 0,4	1,14
<i>Vowel</i>	78,91 ± 0,3	83,61 ± 0,8	4,7
<i>Yeast</i>	54,72 ± 1,2	55,74 ± 0,5	1,02*

Symbol \* oznacza, że różnica trafności nie jest statystycznie znacząca.

**Tabela 7.4.** Porównanie trafności klasyfikowania systemu  $n^2$  opartego na klasyfikatorach drzew decyzyjnych oraz pojedynczego użycia drzewa decyzyjnego DT

Zbiór danych	Trafność klasyfikowania (%)		Poprawa (%) $n^2$ vs DT
	DT	$n^2_{\text{DT}}$	
<i>Automobile</i>	85,5 ± 1,9	87,0 ± 1,9	1,5*
<i>Cooc</i>	54,0 ± 2,0	59,0 ± 1,7	5,0
<i>Ecoli</i>	79,7 ± 0,8	81,0 ± 1,7	1,3
<i>Glass</i>	70,7 ± 2,1	74,0 ± 1,1	3,3
<i>Hist</i>	71,3 ± 2,3	73,0 ± 1,8	1,7
<i>Meta-data</i>	47,2 ± 1,4	49,8 ± 1,4	2,6
<i>Primary Tumor</i>	40,2 ± 1,5	45,1 ± 1,2	4,9
<i>Soybean-large</i>	91,9 ± 0,7	92,4 ± 0,5	0,5*
<i>Vowel</i>	81,1 ± 1,1	83,7 ± 0,5	2,6
<i>Yeast</i>	49,1 ± 2,1	52,8 ± 1,8	3,7

Symbol \* oznacza, że różnica trafności nie jest statystycznie znacząca.

Wyniki porównania struktury klasyfikatora  $n^2$  zbudowanego z użyciem algorytmu MODLEM oraz jednokrotnej indukcji reguł w wersji wieloklasowej przedstawione są w tabeli 7.3. W trakcie klasyfikowania obiektów testowych wykorzy-

stano strategię opartą na mierze odległościowej (wzory (2.25) i (2.26)). Wynik trafności klasyfikacji podano jako wartość średnią wraz z odchyleniem standardowym. Podobnie wyniki porównawcze dla drzew decyzyjnych zamieszczono w tabeli 7.4. Dla dwóch pozostałych typów algorytmów podano tylko zbiorcze zestawienie w tabeli 7.5. Więcej szczegółowych wyników podano w pracy [99].

**Tabela 7.5.** Porównanie trafności klasyfikowania systemu  $n^2$  opartego na klasyfikatorach sieci neuronowych (ANN) i algorytmie  $k$ -NN

Zbiór danych	Trafność klasyfikowania (%)			
	ANN	$n^2_{ANN}$	$k$ -NN	$n^2_{k-NN}$
<i>Automobile</i>	52,6 ± 2,0	58,1 ± 2,3	77,7 ± 0,9	76,7 ± 1,0
<i>Cooc</i>	56,0 ± 1,9	65,3 ± 0,7	68,4 ± 0,6	68,3 ± 0,6
<i>Ecoli</i>	81,7 ± 1,7	83,0 ± 1,6	81,3 ± 0,5	81,3 ± 0,4
<i>Glass</i>	62,7 ± 2,0	62,8 ± 0,8	68,8 ± 0,8	68,5 ± 1,0
<i>Hist</i>	65,7 ± 3,0	83,3 ± 1,4	89,3 ± 0,5	89,3 ± 0,5
<i>Meta-data</i>	50,5 ± 1,6	47,2 ± 1,5	40,6 ± 1,6	42,1 ± 1,6
<i>Primary Tumor</i>	38,2 ± 1,5	43,4 ± 1,2	33,4 ± 1,2	36,2 ± 0,5
<i>Soybean-large</i>	90,1 ± 0,8	92,9 ± 0,7	89,9 ± 0,4	89,9 ± 0,4
<i>Vowel</i>	59,7 ± 2,4	86,1 ± 1,0	98,9 ± 0,2	98,9 ± 0,2
<i>Yeast</i>	53,1 ± 1,4	59,0 ± 0,9	52,8 ± 0,7	53,3 ± 0,7

Omawiając uzyskane wyniki, należy zauważyć, że w przypadku użycia algorytmu indukcji reguł decyzyjnych nigdy nie zaobserwowano spadku trafności klasyfikowania. W 7 na 10 badanych zbiorów danych zaobserwowano statystycznie znaczący przyrost trafności. Największe przyrosty wystąpiły dla zbiorów *Cooc* (3,9%) oraz *Meta-Data* (4,69%). Należy więc uznać, że algorytmy regułowe są skuteczne dla konstrukcji klasyfikatora  $n^2$ .

Podobne interpretacje wyników można przeprowadzić dla użycia drzew decyzyjnych. W 8 na 10 zbiorów danych zaobserwowano znaczące statystycznie przyrosty trafności klasyfikowania, największe – dla zbiorów *Cooc* i *Primary Tumor*. Poprawa trafności klasyfikowania wydaje się trochę większa niż dla reguł decyzyjnych, ale wynika to z faktu, że algorytm MODLEM w wersji wieloklasowej pozwalał na osiągnięcie lepszej trafności klasyfikowania niż drzewa.

W przypadku użycia sieci neuronowych można zaobserwować także przyrosty trafności klasyfikowania, szczególnie dla zbioru *Hist*, lecz wystąpił także spadek trafności dla zbioru *Meta-Data*.

Algorytm  $k$ -NN nie okazał się skutecznym narzędziem do tworzenia klasyfikatorów binarnych. Szukając przyczyny, postawiono hipotezę, że istotne dla poprawy trafności klasyfikowania może być stosowanie dedykowanych zbiorów atrybutów do rozróżniania obiektów z odpowiednich par klas. Algorytm  $k$ -NN uwzględnia wszystkie atrybuty w trakcie obliczania odległości pomiędzy obiektami, więc jest wrażliwy na obecność „nierelwantnych” atrybutów. Algorytmy indukcji reguł czy drzew decyzyjnych mają „wewnętrzny mechanizm” wyboru silniej dyskryminują-

cych atrybutów i redukcji zbędnych atrybutów podczas indukcji wiedzy z danych. Stąd mogą się lepiej „dostroić” do trudności problemu uczącego (porównaj dyskusję na temat wyboru algorytmu uczącego przeprowadzoną w [55] oraz [96]; patrz także ostatnie akapity w rozdziale 7.4.1).

Rozwiązaniem jest wprowadzenie mechanizmu selekcji atrybutów dla algorytmu  $k$ - $NN$ . Kierując się tym, w pracy [100] zaproponowano rozszerzenie architektury klasyfikatora  $n^2$  przez włączenie w fazę uczenia się klasyfikatorów składowych mechanizmu selekcji cech zgodnego z tzw. modelem *wrapper*. Model ten został podany oryginalnie przez Kohaviego dla pojedynczych klasyfikatorów [101, 108]. W modelu wrapper zaimplementowano strategię sekwencyjnego doboru cech FBFS (ang. *Forward Beam Feature Selection*), zgodną z propozycją Jelonka oraz autora [98]. Rozszerzoną architekturę klasyfikatora  $n^2$  dla algorytmu  $k$ - $NN$  oceniono eksperymentalnie dla tych samych danych. Wyniki przedstawione w [96 oraz 100] jednoznacznie wskazały na przyrost trafności klasyfikacji po zastosowaniu mechanizmu wyboru podzbiorów cech. Szczegółowe wyniki oraz ich dyskusję zaprezentowano w pracach [96, 100].

Podsumowując wyniki oceny eksperymentalnej, należy stwierdzić, że tworzenie złożonego klasyfikatora  $n^2$  prowadzi do przyrostu trafności klasyfikacji w przypadku stosowania zarówno algorytmu indukcji reguł decyzyjnych MODLEM, jak i algorytmu indukcji drzew decyzyjnych ASSISTANT86. Wyniki uzyskane dla tych podejść są porównywalne.

Wzrost trafności klasyfikowania można otrzymać także w przypadku użycia algorytmu uczenia wielowarstwowej sieci neuronowej metodą wstecznej propagacji błędu. Należy jednak podkreślić, że liczba konfiguracji sieci do sprawdzenia, a także dobór parametrów algorytmu wymaga wielu eksperymentów i zwiększa koszty obliczeniowe w porównaniu z poprzednimi algorytmami. Algorytm  $k$ - $NN$  w wersji podstawowej nie jest skuteczny w konstrukcji klasyfikatora  $n^2$  i wymaga zastosowania mechanizmu selekcji cech.

Z punktu widzenia celów niniejszej rozprawy ważnym wynikiem jest potwierdzenie możliwości zbudowania na podstawie reguł decyzyjnych klasyfikatora złożonego pozwalającego na skuteczniejsze rozwiązywanie zadania klasyfikowania w przypadku problemów wieloklasowych.

Należy jednak również pamiętać o ograniczeniach tego typu podejścia. Struktura takiego systemu klasyfikującego jest dość skomplikowana i raczej trudna do analizy dla eksperta czy użytkownika. Ponadto koszty obliczeniowe są wyższe niż w przypadku standardowego pojedynczego algorytmu. Wprowadzenie mechanizmu selekcji cech dodatkowo je zwiększa. Pomimo przyrostu trafności klasyfikowania może to stanowić przeszkodę w analizie dużych zbiorów danych.

## **8. Przykłady zastosowania metod indukcji reguł decyzyjnych w analizie danych medycznych i technicznych**

### **8.1. Znaczenie zastosowań praktycznych**

W pracy omawiano metody indukcji reguł decyzyjnych. Niektóre propozycje oceniano za pomocą eksperymentów obliczeniowych na danych testowych. Przy najmniej część przedstawionych metod była jednak tworzona także z zamiarem, aby były one przydatne do rozwiązywania praktycznych problemów. Należy przypomnieć, że stosowanie algorytmu indukcji wiedzy jest częścią całości procesu odkrywania wiedzy, a jedną z motywacji powstawania i rozwoju systemów odkrywania wiedzy jest właśnie rozwiązywanie praktycznych problemów w analizie różnego rodzaju danych naukowych, ekonomicznych czy przemysłowych. Warto zapoznać się z dyskusjami aspektów zastosowań praktycznych przedstawionymi w [50, 86, 126]. Zacytujmy myśl podaną przez Langleya i Simona [126] mówiącą, że „jednym z głównych celów uczenia maszynowego jest nadzieja na automatyzację procesu inżynierii wiedzy i zastępowanie go za pomocą technik polepszających trafność i efektywność przez odkrywanie regularności z danych uczących. Ostatecznym testem tych technik jest zdolność do stworzenia systemów, które będą używane w przemyśle, edukacji czy innych dziedzinach”. Podkreślmy także, że nie jest to proste zagadnienie. Jak pisze Han [86], „pomimo szerokich i różnorodnych zastosowań [...] istnieje ciągle nietrywialna luka pomiędzy ogólnymi podstawami metod a efektywnymi narzędziami dla specyficznego zastosowania”. Ciekawe problemy praktyczne są także inspiracją nowych problemów badawczych. Ponadto doświadczenia płynące zarówno z udanych zastosowań, jak i z niepowodzeń, są przydatne w dyskusji zasad wyboru przyszłych problemów do analizy oraz doboru właściwych dla nich metod. Autor rozprawy wspólnie ze współpracownikami pracował nad rozwiązaniem różnych rzeczywistych problemów eksploracji danych, starając się wykorzystać część metod omawianych w niniejszej rozprawie. Doświadczenia obejmują głównie problemy medyczne i techniczne. Podkreślmy, że przedstawione w tym rozdziale wybrane problemy nie wykorzystują wyłącznie jednego schematu metodologicznego, lecz różnią się zastosowanymi aspektami metod indukcji reguł decyzyjnych.

Ze względu na to, że szczegółowe omawianie zagadnień użytych w analizie tych problemów czy osiągniętych rezultatów może być bardziej interesujące dla specjalistów w danych dziedzinach, a także biorąc pod uwagę konieczność ograniczenia objętości rozprawy, zdecydowano się na ich krótką i syntetyczną prezentację według następującego schematu:



- **Sformułowanie problemu:** omówienie celu eksploracji danych z punktu widzenia dziedziny zastosowania.
- **Dane i ich charakterystyka:** informacje o analizowanych danych, sposobie ich zebrania i ich cechach charakterystycznych.
- **Aspekty metodologiczne:** informacja o zastosowanych metodach, charakterystyka szczególnie interesujących aspektów przeprowadzonej analizy.
- **Osiągnięte rezultaty:** krótka charakterystyka wyników oraz informacja o ewentualnych próbach szerszego zastosowania praktycznego w danej dziedzinie.

## 8.2. Przegląd wybranych zastosowań

### 8.2.1. Analiza danych medycznych

Przedstawiane problemy dotyczą eksploracji danych wykonanej w celu wspomaganie decyzji w zakresie diagnozy i doboru terapii. W ogólności podejmowanie decyzji diagnostycznych w medycynie jest skomplikowanym procesem, który obejmuje zbieranie różnych danych i ich interpretację przez lekarza. Dane mogą pochodzić z różnych źródeł, np. z wywiadu chorobowego, badań przeprowadzonych przez lekarza czy z wyników testów laboratoryjnych lub procedur diagnostycznych (obrazowych, sygnałowych itp.). Zarówno analiza tych danych, jak i doświadczenie kliniczne lekarza powinny prowadzić do postawienia diagnozy (rozpoznania wstępnego i później końcowego) oraz do określenia właściwej terapii. Zadanie to jest trudne, jeśli aktualny stan wiedzy na temat danej choroby nie jest jeszcze dostatecznie rozwinięty lub nie ma jednomyślności pomiędzy specjalistami. Ponadto postęp technologiczny w zakresie urządzeń diagnostycznych znacząco zwiększył rozmiar dostępnych danych pomiarowych [200]. W obu przypadkach zebrane dane mogą więc mieć różne znaczenie. Podlegają one dalszej analizie w celu odnalezienia i wybrania najważniejszych elementów do interpretacji medycznej. Typowe zadania z medycznego punktu widzenia to [213]:

1. Identyfikacja najważniejszych atrybutów służących do klasyfikacji pacjentów.
2. Odkrywanie zależności pomiędzy wartościami atrybutów a przydziałem pacjenta do określonych klas.

W celu rozwiązania tych zadań autor wspólnie z Krzysztofem i Romanem Słowińskimi stosowali metody wykorzystujące między innymi indukcję reguł decyzyjnych oraz elementy teorii zbiorów przybliżonych. Poniżej syntetycznie charakteryzujemy wybrane problemy.

#### **Klasyfikacja pacjentów z chorobą wrzodową dwunastnicy leczonych metodą wysoce wybiórczej wagotomii**

**Sformułowanie problemu.** Celem było określenie wskazań do przeprowadzania zabiegu chirurgicznego wysoce wybiórczej wagotomii (*HSV*) u pacjentów cier-

piących na chorobę wrzodową dwunastnicy. Opis medyczny problemu znajduje się w [171].

**Dane i ich charakterystyka.** Dane zebrał K. Słowiński w jednym z poznańskich szpitali na podstawie historii chorób pacjentów [171]. Pacjenci opisani są za pomocą 11 atrybutów i przydzieleni do 4 klas wyrażających skuteczność zabiegu chirurgicznego. Klasy w sensie liczebności są nierównoważone (silna przewaga grupy wyleczonych pacjentów). Atrybuty są różnego typu (nominalnego, porządkowego i liczbowego). Dostępne są także normy medyczne interpretacji wyników testów, które były podstawą eksperckiej dyskretyzacji atrybutów liczbowych.

**Aspekty metodologiczne.** Postawiono następujące zadania: ocena znaczenia atrybutów i ich podzbiorów dla przybliżenia klasyfikacji pacjentów; poszukiwanie tzw. modeli pacjentów charakterystycznych dla klas decyzyjnych; poszukiwanie reprezentacji zależności pomiędzy wartościami wybranych atrybutów warunkowych a atrybutem decyzyjnym w postaci reguł decyzyjnych. W analizie stosowano głównie elementy teorii zbiorów przybliżonych.

**Osiągnięte rezultaty.** Wyniki osiągnięte przez K. i R. Słowińskich [171] są jednym z pierwszych zastosowań teorii zbiorów przybliżonych do analizy rzeczywistych danych. Wprowadzono metody oceny znaczenia atrybutów dla przybliżenia klasyfikacji obiektów oraz podejście do analizy wrażliwości jakości przybliżenia klasyfikacji na zmiany w dyskretyzacji atrybutów liczbowych [209]. Autor rozprawy brał udział w drugim etapie analizy problemu związanym z zastosowaniem algorytmu LEM2 do indukcji reguł decyzyjnych, z oceną ich zdolności klasyfikacyjnej oraz z obszernymi studiami porównawczymi z użyciem analizy dyskryminacyjnej oraz algorytmów indukcji drzew decyzyjnych (ASSISTANT86, CART) [116, 117, 230]. Wyniki (zwłaszcza modele pacjentów oraz wybrane reguły decyzyjne) były dyskutowane w środowisku medycznym i wykorzystywane w kwalifikacji następnych pacjentów do zabiegu, dopóki metoda wysoce wybiórczej wagotomii nie została zastąpiona przez nowe terapie farmakologiczne.

### **Wieloetapowe leczenie ostrego zapalenia trzustki płukaniem otrzewnej**

**Sformułowanie problemu.** Celem była analiza wieloetapowego problemu decyzyjnego dotyczącego leczenia pacjentów cierpiących na ostre zapalenie trzustki metodą płukania otrzewnej. Z medycznego punktu widzenia dążono do identyfikacji najważniejszych atrybutów w celu określenia stanu chorego oraz prognozowania wyników i liczby potrzebnych płukań [210, 212].

**Dane i ich charakterystyka.** Dane zebrał K. Słowiński w jednym z poznańskich szpitali. Dysponowano różnego rodzaju tablicami decyzyjnymi zawierającymi informacje o stanie pacjentów przed rozpoczęciem terapii oraz dane opisujące pacjentów na każdym etapie płukania. Liczba pacjentów zmniejsza się pomiędzy etapami ze względu na poprawę stanu zdrowia niektórych z nich.

**Aspekty metodologiczne.** Do oceny ważności atrybutów wykorzystano doświadczenia wypracowane w problemie wysoce wybiórczej wagotomii. Istotnym elementem odróżniającym to zastosowanie jest wieloetapowość procesu oraz analiza doświadczenia terapeutycznego. Do indukcji reguł stosowano algorytm LEM2. Wykorzystywano także różne miary oceny pojedynczych reguł.

**Osiągnięte rezultaty.** Zidentyfikowano główne symptomy i wyniki laboratoryjne, na które lekarz powinien zwracać uwagę przy ocenie stanu pacjenta z ostrym zapaleniem trzustki leczonego płukaniem otrzewnej. Dokonano analizy powtarzających się wzorców w regułach decyzyjnych w kolejnych etapach (podzbiorów warunków elementarnych), stosując różne miary oceny związane z pokryciem obiektów. Wyniki te mają istotną interpretację medyczną [212].

### **Leczenie kamicy moczowej za pomocą ESWL**

**Sformułowanie problemu.** Celem była analiza doświadczenia klinicznego w zakresie leczenia kamicy moczowej metodą kruszenia kamieni za pomocą techniki fali uderzeniowej generowanej poza organizmem (ang. ESWL). Poszukiwano modeli charakteryzujących stan pacjentów po zabiegu (bez komplikacji lub z komplikacjami) oraz wspomagających długoterminowe prognozowanie wyniku leczenia. Opis medyczny problemu znajduje się w [7].

**Dane i ich charakterystyka.** Dane zebrał A. Antczak w jednym z poznańskich szpitali na podstawie ankiet związanych z wykonywanymi zabiegami. Dotyczą one 500 pacjentów opisanych za pomocą 33 atrybutów (nominalnych lub porządkowych). Dostępne są dwie możliwości zdefiniowania klasyfikacji pacjentów. Klasy decyzyjne są względnie zrównoważone.

**Aspekty metodologiczne.** Badano przydatność reduktów oraz różnych procedur oceny zmiany jakości przybliżenia klasyfikacji do zadania selekcji atrybutów. Indukcję reguł wykonano algorytmami indukcji: minimalnego zbioru reguł oraz satysfakcjonującego zbioru reguł (EXPLORE). Wykorzystywano także model zmiennej precyzji [274] do indukcji reguł częściowo dyskryminujących. Badano eksperymentalnie trafność klasyfikowania w celu wyboru najlepszego zbioru reguł.

**Osiągnięte rezultaty.** Zbiór danych okazał się bardzo trudny do analizy. Metody wyboru atrybutów oparte na poszukiwaniu reduktów były nieskuteczne. Wprowadzono więc oryginalną metodę poszukiwania mniejszych podzbiorów niezależnych atrybutów i badania znaczenia atrybutów wewnątrz tych podzbiorów [240]. Najskuteczniejszy zbiór reguł decyzyjnych w sensie trafności klasyfikowania otrzymano, stosując algorytm poszukiwania minimalnego zbioru reguł w modelu zmiennej precyzji. Prowadzono także wspólnie z ekspertami dziedzinowymi analizę zbioru reguł za pomocą różnych miar oceny. W wyniku tej analizy wyselekcjonowano podzbiór reguł mogący stanowić podstawę do wspomagania działania lekarza lub przynajmniej użyteczny w celach szkoleniowych [8]. W trakcie oceny reguł wykorzystywano narzędzia stworzone na potrzeby interakcyjnego odkrywa-

nia reguł (porównaj rozdz. 3.5). Otrzymane wyniki spotkały się z szerokim zainteresowaniem środowiska medycznego [6–8].

### **Analiza leczenia pacjentów z obrażeniami wielonarządowymi**

**Sformułowanie problemu.** Celem była analiza doświadczenia dotyczącego postępowania wobec pacjentów z obrażeniami wielonarządowymi przyjmowanych w trybie nagłym (były to ofiary różnego typu wypadków przyjmowane na izbie przyjęć do oddziału chirurgii urazowej). Z uwagi na duże zagrożenie życia (część pacjentów zmarła) badano znaczenie poszczególnych atrybutów dotyczących wykonywanych badań lub procedur diagnostycznych dla klasyfikacji pacjentów [241].

**Dane i ich charakterystyka.** Dane zabrano na podstawie analizy dokumentacji izby przyjęć jednego z poznańskich szpitali. Pacjenci są opisani 22 atrybutami różnego typu, w większości nominalnymi, i podzieleni na dwie klasy decyzyjne o nierównoważonej liczności.

**Aspekty metodologiczne.** Problem został sformułowany w kategoriach wyboru podzbioru atrybutów oraz poszukiwania zbioru reguł decyzyjnych o dobrych właściwościach klasyfikujących.

**Osiągnięte rezultaty.** Zbiór danych nie jest łatwy do analizy. Żadna z pojedynczo stosowanych metod nie doprowadziła do akceptowalnych wyników (niejednoznaczność redukcji liczby atrybutów, zbyt liczne zbiory reguł wspieranych przez zbyt małą liczbę przykładów uczących). Wprowadzono oryginalną propozycję zastosowania mechanizmu indukcji konstruktywnej do redefiniowania przestrzeni atrybutów. Propozycja obejmuje: elementy indukcji sterowanej danymi (ocenę znaczenia statystycznego atrybutów za pomocą testów nieparametrycznych), analizę hipotez (tj. składni wygenerowanych reguł w celu wyszukiwania „silnych” warunków elementarnych), mechanizm wielostrategicznej indukcji konstruktywnej sterowanej wiedzą dziedzinową (ekspert medyczny zdefiniował akceptowalne możliwości przekształcania atrybutów) oraz strategię iteracyjnego poszukiwania lepszej przestrzeni reprezentacji [241]. Zastosowanie tej propozycji doprowadziło do zdefiniowania nowych 12 atrybutów i zbudowania zbioru 18 reguł decyzyjnych o dobrych właściwościach klasyfikujących.

### **Weryfikacja diagnozowania uszkodzeń stawu kolanowego na podstawie MRI**

**Sformułowanie problemu.** Problem dotyczył weryfikacji klinicznej diagnoz pewnej formy uszkodzenia stawu kolanowego (*anterior cruciate ligament* – ACL) postawionych na podstawie symptomów zaobserwowanych na zdjęciach wykonanych techniką magnetycznego rezonansu jądrowego (MRI). Procedura chirurgiczna artroskopii umożliwia bezpośrednią weryfikację uszkodzenia stawu, lecz jest techniką inwazyjną i wykonana niepoprawnie może prowadzić do komplikacji [214].

**Dane i ich charakterystyka.** Wybrano 140 pacjentów, których objęto zarówno standardowymi procedurami diagnostycznymi i badaniem za pomocą MRI, jak

i badaniem metodą artroskopii. U 100 pacjentów wykluczono, a u 40 potwierdzono występowanie uszkodzenia ACL. We współpracy z ekspertami medycznymi oraz na podstawie wstępnej analizy statystycznej wyselekcjonowano zbiór atrybutów opisujących pacjentów. Większość atrybutów wynikała z pomiarów przeprowadzonych na podstawie zdjęcia MRI.

**Aspekty metodologiczne.** Problem został sformułowany w kategoriach wyboru podzbioru atrybutów, poszukiwania zbioru reguł decyzyjnych o dobrych właściwościach klasyfikujących oraz próby interpretacji reguł w celu zbudowania modeli pacjentów charakterystycznych dla klasy bez uszkodzeń oraz dla klasy z uszkodzeniami stawu kolanowego.

**Osiągnięte rezultaty.** Wykonano studium porównawcze użycia różnych technik wstępnej dyskretyzacji. Wybrana automatyczna dyskretyzacja okazała się skuteczniejsza od eksperckiej. W przypadku poszukiwania zbiorów reguł decyzyjnych stosowano algorytmy LEM2 oraz EXPLORE. Nowym elementem jest ocena zdolności klasyfikacyjnych za pomocą miar wrażliwości i specyficzności. Minimalny zbiór reguł okazał się skuteczniejszy z uwagi na miarę wrażliwości dla klasy pacjentów z ACL. Satisfakcjonujący zbiór reguł zapewniał nieznacznie gorszą skuteczność klasyfikacji, lecz dostarczał dodatkowych informacji niezbędnych do zbudowania modeli charakterystycznych pacjentów [214]. Z medycznego punktu widzenia najważniejszym wynikiem było potwierdzenie wysokiej skuteczności diagnostycznej wskaźnika PCL określanego na podstawie zdjęć MRI.

### 8.2.2. Problemy diagnostyki technicznej

Przedstawimy kilka uwag na temat celów diagnostyki technicznej i zadań, w których rozwiązaniu mogą być przydatne metody odkrywania wiedzy. Efektywna eksploatacja obiektów mechanicznych czy maszyn w procesach przemysłowych wymaga wiarygodnej informacji o ich stanie technicznym. Informacja ta jest często rozszerzona o predykcję zmiany ich stanu technicznego. Ma to szczególne znaczenie w przypadku maszyn o krytycznym znaczeniu w procesie, wymagających specjalnych skomputeryzowanych systemów nadzoru (monitorowania) – np. turbiny w elektrowniach. W przypadku maszyn o mniejszym znaczeniu dokonuje się okresowych pomiarów za pomocą urządzeń przenośnych. Ocena stanu technicznego wykonywana jest na podstawie tzw. symptomów, czyli wielkości, które zmieniają się wraz z pogarszaniem się stanu technicznego maszyny. Przykładami symptomów są poziom drgań reprezentatywnych punktów obiektu, poziom hałasu w otoczeniu pracującej maszyny, temperatura, ciśnienie czy wzajemna pozycja części składowych maszyny. Z diagnostycznego punktu widzenia poszczególne symptomy mają różną przydatność dla konkretnego problemu. Dla pojedynczych symptomów mierzonych na skalach liczbowych definiuje się czasami tzw. wartości graniczne. Dzielią one dziedzinę symptomu na pewne podprzedziały, które mogą być interpretowane w kategoriach normalnych warunków pracy i ich stopniowego pogarszania się. Są one

definiowane na podstawie zaleceń producentów i wskazań literaturowych, lecz praktyczne badania nie zawsze potwierdzają ich użyteczność. Ponadto w postępowaniu diagnostycznym mamy zwykle dostęp do dużych zbiorów danych. Dane gromadzone podczas obserwacji obiektów diagnozowania mogą być także niekompletne, sprzeczne, niedokładne czy obciążone niepewnością. Także sama wiedza diagnostyczna ma często charakter przybliżony. Dyskusja tych zagadnień przedstawiona jest w [150]. Z punktu widzenia zastosowań rozważanych w tym rozdziale istotnymi zadaniami badań diagnostycznych są [159, 160, 279]:

1. ocena zdolności diagnostycznej poszczególnych symptomów,
2. ocena różnych metod definiowania wartości granicznych dla tych symptomów,
3. selekcja podzbioru symptomów zapewniających satysfakcjonującą ocenę stanu technicznego,
4. stworzenie klasyfikatora stanu technicznego.

Autor wraz ze współpracownikami zajmował się rozwiązaniem tych zadań dla kilku problemów praktycznych. Poniżej krótko charakteryzujemy dwa z nich.

#### **Analiza stanu technicznego łożysk**

**Sformułowanie problemu.** Dotyczy oceny stopnia uszkodzenia łożyska tocznego podczas jego eksploatacji. Odpowiada przedstawionym powyżej zadaniom 1–4.

**Dane i ich charakterystyka.** Rozważano dwa problemy analizy podobnego typu łożysk związane z różnymi zastosowaniami. Pierwszy zbiór danych pochodził z badań laboratoryjnych [159], w których w sposób celowy generowano różnego typu uszkodzenia. Drugi zbiór danych został zebrany podczas badań diagnostycznych na tzw. transporterach taśmowych w kopalni węgla brunatnego [160]. W obu przypadkach mierzono symptomy wibroakustyczne. Dane zostały przygotowane i wstępnie przetworzone przez R. Nowickiego.

**Aspekty metodologiczne.** Problem został sformułowany w kategoriach (1) badania wpływu dyskretyzacji na jakość klasyfikacji łożysk, (2) oceny zdolności diagnostycznych pojedynczych symptomów oraz ich podzbiorów, (3) poszukiwania zbioru reguł decyzyjnych o jak najlepszych właściwościach klasyfikujących.

**Osiągnięte rezultaty.** Zastosowanie teorii zbiorów przybliżonych pozwoliło na ocenę różnych metod definiowania wartości granicznych dla tego typu problemu i wybór najlepszej z nich. Możliwa okazała się także redukcja zbioru mierzonych symptomów. Zastosowanie algorytmu LEM2 doprowadziło do wygenerowania zbioru kilku reguł o wysokiej skuteczności klasyfikacyjnej. Rezultaty oraz ich interpretację przedstawiono w pracach [159, 160].

#### **Ocena stanu technicznego pojazdów**

**Sformułowanie problemu.** Problem dotyczył oceny stanu technicznego autobusów określonego typu. Na podstawie wyników okresowych przeglądów oraz

informacji na temat bieżącej eksploatacji pojazdu podejmuje się decyzję, czy mają być dalej eksploatowane czy skierowane do remontu [279].

**Dane i ich charakterystyka.** Dane zostały zebrane w jednym z polskich przedsiębiorstw i dotyczą podstawowego typu autobusu wykorzystywanego w komunikacji podmiejskiej. Większość atrybutów dotyczy oceny stanu technicznego silnika wykonanej podczas przeglądu na stanowisku laboratoryjnym. Reszta atrybutów zawiera informacje o parametrach eksploatacyjnych, takich jak np. średnie zużycie paliwa. Wszystkie symptomy zdefiniowane są na skalach liczbowych. Dane zostały przygotowane przez J. Żaka.

**Aspekty metodologiczne.** Problem sformułowano jako zadania 1, 3 oraz 4.

**Osiągnięte rezultaty.** Wykonano studium porównawcze użycia wielu metod automatycznej dyskretyzacji. Wybrana metoda okazała się skuteczniejsza (z uwagi na końcową trafność klasyfikacji) niż dyskretyzacja ekspercka oparta na wartościach granicznych. Przeprowadzono studium porównawcze użycia wielu technik indukcji reguł decyzyjnych. Zastosowanie algorytmu EXPLORE doprowadziło do wygenerowania zbioru reguł o takiej samej skuteczności klasyfikacyjnej (bardzo wysoka trafność klasyfikowania – 98%) jak minimalny zbiór reguł, lecz o znacznie lepszych wartościach wybranych miar oceny (np. średnie względne wsparcie powyżej 50% obiektów w klasie decyzyjnej). W pracy [225] przedstawiono studium użycia w odniesieniu do tych danych uogólnienia teorii zbiorów przybliżonych wykorzystującego elementy teorii zbiorów rozmytych do analizy nieprecyzyjnych danych.

### 8.2.3. Analiza danych finansowych

Problem dotyczył oceny wniosków firm ubiegających o się o kredyt komercyjny w jednym z polskich banków [247]. Jeśli wnioski spełniają wymogi prawne, są oceniane na podstawie parametrów ekonomicznych oceniających firmę. Z punktu widzenia ryzyka bankowego udzielenie „złego” kredytu związane jest z możliwością opóźnień w spłacie lub ryzykiem jego niespłacenia. Zebrano zbiór danych historycznych o różnych przykładach udzielonych kredytów. Przykład uczący odpowiadający wnioskowi kredytowemu był opisany 35 atrybutami i mógł być przydzielony do jednej z 5 grup ryzyka bankowego. Interesujące z punktu widzenia zastosowania jest poszukiwanie wiarygodnych wzorców w danych opisujących klientów z różnych grup ryzyka bankowego. Do analizy tego problemu wykorzystano hybrydowy system integrujący uczenie się reguł decyzyjnych oraz wcześniej zapamiętanych przypadków. Ideę systemu oraz zbiorcze zestawienie jego skuteczności klasyfikacyjnej omówiono w rozdziale 7.3 niniejszej rozprawy. Szczegółowe informacje przedstawione są w [247].

Podsumowując prezentację wybranych problemów, do których rozwiązania zastosowano niektóre metody proponowane w rozprawie, należy dodać, że autor rozprawy jest współautorem różnego rodzaju oprogramowania; przykładowo:

- podstawowe elementy teorii zbiorów przybliżonych są częścią systemów RoughDAS, RoughClass oraz ROSE [182],
- omawiane algorytmy indukcji reguł i ich oceny klasyfikacyjnej początkowo były samodzielnymi programami; obecnie algorytmy MODLEM oraz EXPLORE są częścią systemu Rose,
- podejście do wielokryterialnych problemów sortowania oparte na regułach decyzyjnych zostało zaimplementowane w systemie 4-eMka.

Narzędzia te były i są wykorzystywane przez innych badaczy także do rozwiązywania wielu innych problemów zastosowań (np. [42, 60, 194]).

### 8.3. Uwagi na temat realizacji zastosowań praktycznych

W niniejszym rozdziale omawiano przykładowe problemy praktyczne, w których autor stosował metody indukcji reguł decyzyjnych. W ogólności można stwierdzić, że pozwoliły one w pewnym stopniu zweryfikować praktyczną przydatność proponowanych metod. Szczegółowa dyskusja wniosków płynących z tych zastosowań wykracza poza zakres tej rozprawy. Z drugiej strony, niektóre z nabytych doświadczeń można uznać za ważne dla stosowania metod odkrywania wiedzy. Dlatego poniżej zamieszczono ich krótką dyskusję, w której uwzględniono także wnioski z prac [50, 126], oraz (w przypadku zastosowań medycznych) z [113]. W zastosowaniach praktycznych metod indukcji wiedzy należy zwrócić uwagę na następujące elementy:

**Iteracyjny charakter procesu.** Proces związany z realizacją zastosowania składa się z wielu etapów i jest iteracyjny. W ramach rozwiązywania pewnego problemu przechodzi się przez kilka etapów i napotyka pewne trudności, co powoduje często konieczność powrotu do wcześniejszych etapów w celu zrewidowania poprzednio podjętych decyzji. Rozwój danego zastosowania może wymagać wielokrotnych cykli działań tego rodzaju. Doświadczenie takie jest także zgodne z dyskutowaną w literaturze wieloetapowością i iteracyjną „naturą” procesu odkrywania wiedzy [18, 50].

**Formułowanie problemu** jest pierwszym z etapów i wymaga zrozumienia realnego problemu zastosowania w celu sformułowania go w kategoriach używanych w metodach indukcji wiedzy. W większości doświadczeń autora krok ten obejmował transformację do zadania klasyfikowania, w którym decyzja klasyfikacyjna podejmowana jest w jednym etapie. Jak zauważa się w literaturze [35, 126, 149] wiele nawet złożonych problemów jest najczęściej także przeformułowanych na zadanie jednoetapowej klasyfikacji. W odniesieniu do niektórych problemów stosuje się czasami zasadę strukturalnej indukcji, która polega na podziale złożonego zadania na podzadania. Dla każdego z podzadań pozyskuje się odpowiedni zbiór przykładów uczących i rozwiązuje się je jak pojedyncze zadanie. Jest to dyskutowane w [126]. Należy także pamiętać, że prowadzenie dialogu z ekspertami w danej dziedzinie w celu pozyskania niezbędnych informacji nie jest na ogół prostym zadaniem.



**Wybór reprezentacji.** Wiąże się ze zdefiniowaniem reprezentacji dla przykładów uczących i formy poszukiwanej wiedzy. W rozprawie oraz w omawianych w tym rozdziale problemach praktycznych rozważano wyłącznie reprezentację przykładów w postaci tablicy informacyjnej (2.1) lub tablicy decyzyjnej (2.2) oraz reprezentację wiedzy w postaci reguł decyzyjnych. Pomimo że są to bardzo popularne formy reprezentacji, należy pamiętać, że dla niektórych problemów dogodniejsze mogą być inne reprezentacje (np. w logice pierwszego rzędu). Czytelnik może zapoznać się z użyciem alternatywnych form reprezentacji przykładów lub wiedzy na przykład w [49, 144]. Następnym i najczęściej niełatwym zadaniem jest wybór właściwych atrybutów do opisu przykładów. Jest to jeden z najważniejszych elementów wpływający na jakość wyników. W niektórych sytuacjach możliwe jest skorzystanie ze współpracy z ekspertami dziedzinowymi. Jednak większość zastosowań wymaga mniej lub bardziej żmudnych prac przeszukiwania przestrzeni możliwych zbiorów atrybutów za pomocą różnych metod. W niektórych problemach przydatna jest także redefinicja początkowych atrybutów, np. za pomocą algorytmów indukcji konstruktywnej [15]. W doświadczeniach autora zastosowania techniczne były lepiej zdefiniowane, podczas gdy w zastosowaniach medycznych sytuacja była trudniejsza. Przeglądy różnych technik przydatnych do wstępnego przetwarzania danych zawarte są w [35, 86, 183, 262].

**Sposoby pozyskiwania danych.** Zebranie danych niezbędnych do rozwiązania danego problemu często nie jest bezpośrednio i łatwe. O ile w przypadku zastosowań technicznych dysponowano elektronicznymi bazami pomiarów, o tyle w zastosowaniach medycznych sytuacja taka nie występowała. Jeżeli istnieją już pewne systemy rejestracji w izbach przyjęć, to tworzone bazy danych poza informacjami osobowymi nie zawierają atrybutów potrzebnych do analizy rozważanych problemów. Konieczne jest więc bardzo żmudne i pracochłonne pozyskiwanie danych z różnego rodzaju źródeł (często dokumentów papierowych) czy prowadzenie badań ankietowych. Nawet w przypadku istnienia baz danych integracja różnych baz, transformacja danych i ich oczyszczenie nie jest łatwe, a ponadto jest etapem bardzo czasochłonnym (patrz [86, rozdz. 3]).

**Ocena wiedzy odkrytej z danych.** Reguły odkryte z danych nie zawsze są wysokiej jakości. Ocena odkrytej wiedzy może być przeprowadzona na różne sposoby, co dyskutowano w rozdziałach 2 i 3 rozprawy. W literaturze podkreśla się, że „uczenie maszynowe jest nauką eksperymentalną” [126] i dla zadań klasyfikowania sugeruje się stosowanie technik podziału, np. *k-fold cross validation*. Inne miary charakteryzujące reguły (porównaj rozdział 3.2) także mają znaczenie. W zastosowaniach medycznych podkreśla się [113], że prezentowana wiedza powinna być przejrzysta, dostarczać nowego spojrzenia na problem przez odkrycie współzależności i regularności, których lekarz poprzednio nie dostrzegał w jawnej formie, powinna także dostarczać wyjaśnień podejmowanych decyzji. Jeżeli jest to możliwe, to pożądana jest współpraca z ekspertami w danej dziedzinie w celu uzyskania opinii o odkrytej wiedzy. Regułowa forma wiedzy wydaje się dogodna do takiej

interpretacji, lecz w przypadku większej liczby wygenerowanych reguł konieczne jest stosowanie specjalnych narzędzi lub form jej przekształcenia. Do przeglądania dużych zbiorów reguł można stosować specjalne przeglądarki [106] lub języki zapytań do regułowych baz wiedzy [26, 151]. Autor rozwijał pewne narzędzia z tym związane w ramach systemów DataEXPLORE oraz Rose2. Zaleca się także stosowanie różnego rodzaju form wizualizacji graficznej, czy to w postaci wielowymiarowych diagramów [143] czy grafów [23]. W pracy [57] Gains proponuje także przekształcanie zbiorów reguł na specjalne grafy acykliczne. Rola graficznych modeli w ocenie wiedzy jest omawiana w [23].

Ostatnim etapem jest użycie lub wdrożenie praktyczne odkrytej wiedzy. Może to być rozumiane w szerokim sensie. W podejściu literaturowym mogłoby to być tworzenie implementacji systemu z bazą wiedzy w celu wspomaganie decyzji. W doświadczeniach praktycznych autora rozprawy nie wystąpiła jednak taka rzeczywista potrzeba. W wielu przypadkach prosty (i odpowiednio zweryfikowany) zbiór reguł zapisany na papierze był wystarczający dla użytkowników. Jest to w pewnym stopniu związane ze specyfiką badanych problemów. Jednak, jak piszą Langley i Simon [126], także w części znanych zastosowań indukcji reguł decyzyjnych taka forma była wystarczająca do późniejszego użycia w podejmowaniu decyzji. Oczywiście niektóre problemy praktyczne wymagają tworzenia pełnej implementacji komputerowej, i to często wymagającej oprócz części związanej z odkrytą wiedzą innych funkcjonalności nie związanych z technikami uczenia maszynowego – dla przykładu porównaj opis spektakularnego systemu analizy i klasyfikowania zdjęć astronomicznych SKICAT [51]. Podsumowując, należy stwierdzić, że sukces rozwiązywania wybranych problemów zależy nie tylko od wyboru specyficznego algorytmu, ale także w dużym stopniu od innych etapów całego procesu, począwszy od poprawnego sformułowania problemu i wyboru reprezentacji aż do oceny i wdrożenia odkrytej wiedzy. We wszystkich tych etapach istotnym elementem jest współpraca z ekspertami. Autor rozprawy podziela tutaj poglądy wyrażone w [126] oraz [148, s. 51] mówiące, że nabywanie wiedzy metodami maszynowego uczenia się nie jest (pomimo samej nazwy i opinii formułowanych np. w artykułach prasowych) czynnością w pełni automatyczną, gdyż wymaga pewnej pomocy i nadzoru ze strony człowieka. Odkrywanie wiedzy nie jest więc czynnością automatyczną i z natury swojej jest procesem interakcyjnym.

## 9. Podsumowanie

W pracy przedstawiono propozycję spójnej metodyki odkrywania wiedzy z danych metodami indukcji reguł decyzyjnych w perspektywie predykcji i opisu. Założono, że dane reprezentowane są w postaci tablicy decyzyjnej zawierającej charakterystykę zbioru obiektów za pomocą zbioru atrybutów. Jeżeli dane, które podlegały procesowi odkrywania wiedzy, zawierały niespójności opisu obiektów, to stosowano elementy teorii zbiorów przybliżonych.

Przegląd powyższej problematyki przedstawiony w rozdziale 1 wskazał luki metodologiczne związane z dotychczasowymi propozycjami algorytmów indukcji reguł decyzyjnych i teorii zbiorów przybliżonych oraz ich stosowaniem w perspektywach klasyfikacji i opisu w odkrywaniu wiedzy. W związku z tym przedstawiono oryginalne metody opracowane przez autora. Poniżej przedstawimy krótkie, syntetyczne podsumowanie ważniejszych wyników uzyskanych w rozprawie i wskażemy niektóre pożądane kierunki dalszych badań.

Najważniejsze wyniki szczegółowe rozprawy są następujące:

1. Zaprezentowano krytyczną dyskusję różnych algorytmów indukcji reguł decyzyjnych i dokonano w tych kategoriach przeglądu zarówno algorytmów uczenia maszynowego, jak i algorytmów wykorzystujących elementy teorii zbiorów przybliżonych (rozdział 2).
2. Wprowadzono oryginalną metodę indukcji ograniczonego zbioru ogólnych reguł wspartych satysfakcjonującą liczbą przykładów uczących oraz zapewniającego zadowalającą trafność klasyfikowania nowych obiektów (rozdział 3). Realizacja tego celu wiązała się z wprowadzeniem nowego algorytmu EXPLORE indukcji reguł, uwzględniającego wymogi wobec satysfakcjonujących wartości wybranych ilościowych miar oceny reguł (rozdział 3.3.1). Przeprowadzone badania eksperymentalne pokazały, że stosowanie proponowanej metody w odniesieniu do wszystkich z analizowanych danych umożliwiło zbudowanie w miarę prostego i ogólnego zbioru reguł decyzyjnych zapewniającego równocześnie trafność klasyfikowania porównywalną z innymi algorytmami (rozdział 3.3.2).
3. Przedstawiono ponadto interaktywną wersję metody opartej na algorytmie EXPLORE, która w trakcie poszukiwania reguł pozwala dodatkowo uwzględniać wymogi użytkownika co do składni reguł (rozdział 3.4).
4. Wprowadzono oryginalną propozycję algorytmu MODLEM; jest to algorytm indukcji reguł z danych zdefiniowanych na podstawie zarówno atrybutów liczbowych jak i nominalnych (rozdział 4). W szczególności wartości atrybutów liczbowych są przetwarzane bezpośrednio w trakcie indukcji reguł bez ich wcześniejszej dyskretyzacji.

5. Przeprowadzono eksperymenty obliczeniowe na wielu zbiorach danych, w których porównywano użycie algorytmu MODLEM oraz podejścia tradycyjnego (dyskretyzacja wstępna oraz użycie algorytmu LEM2). Wyniki wskazują, że algorytm MODLEM pozwala na indukcję prostszych reguł wspieranych większą liczbą przykładów uczących (rozdział 4.7).
6. W celu analizy niekompletnych tablic decyzyjnych wprowadzono rozróżnienie dwóch różnych semantyk nieznanych wartości atrybutów: wartości niedostępnych oraz wartości brakujących. Zaproponowano dwa uogólnienia teorii zbiorów przybliżonych. Pierwsze jest oparte na niesymetrycznej relacji podobieństwa i umożliwia analizę przypadku niedostępnych wartości atrybutów (rozdział 5.3). Druga propozycja jest oparta na propozycji wartościowanej relacji tolerancji (rozdział 5.4). Propozycja ta ma ogólniejszy zakres zastosowania niż tylko dla przypadku brakujących wartości. Przeprowadzono analizę różnych właściwości przybliżeń i reguł decyzyjnych w tym uogólnieniu, formułując twierdzenie, które jest przydatne w projektowaniu algorytmów indukcji reguł. Zarówno analiza formalna, jak i badania eksperymentalne wskazują, że proponowane podejście oparte na wartościowanej relacji tolerancji prowadzi do precyzyjniejszych przybliżeń zbiorów i liczniejszych zbiorów reguł decyzyjnych niż podstawowa wersja relacji tolerancji (rozdział 5.4.5).
7. Przedstawiono podstawy metodyczne analizy tablic decyzyjnych zawierających przykłady wielokryterialnych problemów sortowania (rozdział 6). Odkrywanie wiedzy z takich przykładów wymaga uwzględniania kryteriów (atrybutów z dziedzinami uporządkowanymi preferencyjnie) oraz uporządkowanych klas decyzyjnych. Zaproponowano rozszerzenie podejścia teorii zbiorów przybliżonych opartego na relacji dominacji (DRSA) na model o zmiennej spójności (VC-DRSA). Uogólnienie to wydaje się szczególnie przydatne do odkrywania wiedzy w rzeczywistych danych, które zawierają niespójności w opisie obiektów z odległych klas decyzyjnych (rozdział 6.3).
8. W zakresie obu uogólnień teorii zbiorów przybliżonych opartych na relacji dominacji szczególną uwagę poświęcono indukcji reguł o uogólnionej składni (rozdział 6.4). Zaproponowano dwa oryginalne algorytmy indukcji tego typu reguł: algorytm DOMLEM przeznaczony do tworzenia minimalnego zbioru reguł oraz algorytm DOMApriori przeznaczony do indukcji satysfakcjonującego zbioru reguł.
9. Przedstawiono zasady stosowania algorytmów indukcji reguł do budowy złożonych systemów klasyfikujących (rozdział 7). W szczególności wprowadzono dwie nowe propozycje systemów. Pierwsza to hybrydowy system integrujący uczenie się reguł decyzyjnych i przypadków. Reguły odnoszą się do typowych sytuacji decyzyjnych, podczas gdy pojedyncze przypadki reprezentują sytuacje trudne i wyjątkowe. Drugą propozycją jest klasyfikator wielokrotny  $n^2$ , który jest systemem specjalizowanym do rozwiązywania wieloklasowych ( $n > 2$ )

problemów uczących. W obu przypadkach przedstawiono motywacje proponowanych systemów, ich architekturę oraz zasady konstruowania. W obu systemach reguły są generowane algorytmami EXPLORE lub MODLEM wprowadzonymi w rozdziałach 3 i 4 rozprawy. Szczegółowe badania eksperymentalne jednoznacznie wskazują, że proponowane systemy złożone pozwalają na osiągnięcie wyższej trafności klasyfikowania niż użyte oddzielnie pojedyncze klasyfikatory wchodzące w skład systemu.

10. W rozdziale 8 omówiono przykładowe problemy praktyczne z zakresu analizy danych medycznych, finansowych oraz diagnostyki technicznej, w której autor stosował niektóre z metod omawianych w rozprawie. W ogólności można stwierdzić, że pozwoliły one pozytywnie zweryfikować praktyczną przydatność proponowanych metod.

Dyskusja praktycznych aspektów realizacji procesu odkrywania wiedzy przeprowadzona w rozdziale 8.3 wskazuje, że sukces rozwiązania danego problemu zależy nie tylko od wyboru specyficznego algorytmu indukcji reguł, ale także od właściwej realizacji innych etapów całego procesu, począwszy od poprawnego sformułowania problemu, wyboru reprezentacji przykładów i hipotez, pozyskania niezbędnych danych aż do oceny i użycia odkrytej wiedzy. We wszystkich tych etapach ważnym czynnikiem jest współpraca z ekspertami.

Podsumowując prezentację najważniejszych wyników rozprawy, należy dodać, że wiele proponowanych metod zostało już zaimplementowanych w różnego rodzaju oprogramowaniu, np. RoughDAS, RoughClass, ROSE, DataExplore czy 4-eMka. Narzędzia te były i są wykorzystywane przez innych badaczy także do rozwiązywania różnych problemów praktycznych.

Praca ta nie wyczerpuje oczywiście całości problematyki. Kierunki dalszych badań lub nowe kierunki badawcze zostały już wstępnie zasygnalizowane w treści podrozdziałów kończących poszczególne rozdziały rozprawy. Można jednak zaakcentować potrzebę badań dotyczących między innymi takich problemów jak:

1. Wielokryterialna ocena pojedynczych reguł decyzyjnych. Część algorytmów generuje zbyt dużą liczbę reguł decyzyjnych, co utrudnia ich interpretację przez człowieka. Dotychczasowe propozycje selekcji reguł wykorzystują pojedyncze ilościowe miary oceny lub formuły empiryczne łączące najwyżej dwie miary, które jednak nie mają silnych podstaw teoretycznych. Wydaje się interesujące wybranie spójnej rodziny kryteriów opartych na miarach opisowych i zaproponowanie metodyki wywodzącej się z wielokryterialnego wspomaganie decyzji w celu wyboru podzbioru reguł lub stworzenia ich rankingu.
2. Indukcja reguł z wielkich baz danych. Perspektywa eksploracji zbiorów danych o dużych rozmiarach przechowywanych w bazach danych stawia nowe wymagania wobec algorytmów indukcji, z których większość zakłada przechowywanie informacji o przykładach uczących w pamięci operacyjnej. Interesujące jest więc modyfikowanie tych algorytmów lub tworzenie nowych algo-

rytmów, które będą uczyły się z przykładów przechowywanych na dysku. Inną możliwością jest stosowanie różnych technik losowania podzbioru przykładów z pełnej bazy danych. W ostatnich latach wielu badaczy podjęło badania w tym zakresie.

3. Uczenie się przyrostowe reguł. W części zastosowań nie dysponuje się w chwili indukcji pełnym zbiorem danych. Przykłady uczące są dostępne stopniowo. Stawia się więc pytanie, czy i jak modyfikować wygenerowane poprzednio reguły w zależności od informacji o nowo dostępnych przykładach, lub w jakich warunkach konieczne jest poszukiwanie reguł ze zbioru danych rozszerzonego o nowe przykłady. Ten problem badawczy dotyczy także analizy niekompletnych tablic decyzyjnych, gdzie np. po pewnym czasie od wygenerowania reguł uzyskuje się informacje o prawdziwych wartościach niektórych brakujących wartości atrybutów.
4. Dalsze badania nad regułami dla wielokryterialnych problemów decyzyjnych. Mogą one dotyczyć stworzenia algorytmów indukcji uogólnionych reguł z tzw. tablic porównań wariantów decyzyjnych parami dla wielokryterialnych problemów wyboru lub rankingu. Innym problemem jest poszukiwanie reguł asocjacyjnych w bazach danych w kontekście analizy wielokryterialnej. Badania w tym zakresie już rozpoczęto.
5. Dalsze badania nad złożonymi systemami klasyfikującymi są także jednym z ciekawszych wyzwań. Analiza literatury wskazuje, że jest to obecnie bardzo silnie rozwijany kierunek badań.

Zdaniem autora aktualny stan wiedzy i doświadczenie dotyczące odkrywania wiedzy w różnych reprezentacjach (nie tylko reguł) uzasadniają także potrzebę opracowania w przyszłości podręcznika poświęconego dziedzinie odkrywania wiedzy.

## Literatura

- [1] Agrawal R., Mannila H., Srikant R., Toivonen H., Verkamo A.I., Fast discovery of association rules, w: [49], s. 307-328.
- [2] Aha D.W., Kibler E., Albert M.K., Instance-based learning algorithms, *Machine Learning*, 1991, 6, s. 37-66.
- [3] Ali K., Pazzani M., Error reduction through learning multiple descriptions, *Machine Learning*, 1996, 24 (1).
- [4] Andersen T.L., Martinez T.R., Learning and generalization with bounded order rule sets, w: *Proc. of 10th Int. Symp. on Computer and Information Sciences*, 1995, s. 419-426.
- [5] Ann Aijun, Cercone N., An empirical study on rule quality measures, w: [273], s. 482-491.
- [6] Antczak A., Kwias Z., Słowiński K., Stefanowski J., Analysis of experiences with ESWL using tool provided from artificial intelligence, *British Journal of Urology*, September 1997, 80 (2), s. 338.
- [7] Antczak A., Kwias Z., Sawicki W., Słowiński K., Stefanowski J., Poszukiwanie istotnych atrybutów w analizie doświadczeń nad leczeniem kamicy moczowej za pomocą ESWL, *Urologia Polska*, 1997, 50 (3), s. 298-303.
- [8] Antczak A., Kwias Z., Woźniak W., Słowiński K., Stefanowski J., Algorytmy decyzyjne w leczeniu kamicy moczowej metodą ESWL, *Urologia Polska*, 1997, 50, (3), s. 294-297.
- [9] Azibi R., Vanderpooten D., Construction of rule based assignment model, *European Journal of Operational Research*, 2001 (zgłoszone do czasopisma).
- [10] Bauer E., Kohavi R., An empirical comparison of voting classification algorithms: bagging, boosting and variants, *Machine Learning*, 1999, 36, 105-139.
- [11] Bazan J., A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables, w: [175], s. 321-365.
- [12] Bazan J., Skowron A., Synak P., Dynamic reducts as a tool for extracting laws from decision tables, w: *Proc. of the Symp. on Methodologies for Intelligent Systems*, Oct. 16-19, 1994, LNAI no. 869, Berlin, Springer-Verlag, 1994, s. 346-355.
- [13] Berthold M., Hand D. (red.), *Intelligent data analysis. An introduction*, Berlin, Springer-Verlag, 1999.
- [14] Blake C.L., Merz C.J., *UCI Repository of machine learning databases* [<http://www.ics.uci.edu/mllearn/MLRepository.html>] Irvine, CA: University of California, Department of Information and Computer Science, 1998.
- [15] Bloedorn E., Michalski R.S., *Data-driven constructive induction: A methodology and its application*, w: Liu H., Motoda H. (red.), *Feature selection for knowledge discovery and data mining*, Boston, Kluwer Publishers, 1998.
- [16] Bobin P., *Automatyczne odkrywanie zależności w bazach danych*, Praca magisterska. Instytut Informatyki PP, Poznań 1999.
- [17] Bolc L., Zaremba P., *Wprowadzenie do uczenia się maszyn*, Warszawa, Akademicka Oficyna Wydawnicza, 1992.
- [18] Brachmann R.J., Tej Anand, The process of knowledge discovery: a human centered approach, w: [49], s. 37-58.

- [19] Brazdil P., Torgo L., Knowledge acquisition via knowledge integration, w: Current Trends in Knowledge Acquisition, IOS Press, 2000.
- [20] Breiman, L., Bagging predictors, Machine Learning, 1996, 24 (2), s. 123-140.
- [21] Breiman L., Friedman J.H., Olshen R.A., Stone C.J., Classification and Regression Trees, Wadsworth Int. Group, 1984.
- [22] Bruha I., Quality of decision rules: definitions and classification schemes for multiple rules, w: Nakhaeizadeh G., Taylor C. (red.), Machine Learning and Statistics, Wiley and Sons, 1997, s. 107-131.
- [23] Buntine W., Graphical models for discovered knowledge, w: [49], s. 59-82.
- [24] Catlett J. Megainduction: machine learning on very large databases, PhD thesis, University of Sydney, 1991.
- [25] Cendrowska J., PRISM, an algorithm for inducing modular rules, Int. J. Man-Machine Studies, 1987, 27, s. 349-370.
- [26] Ceri S., Meo R., Psaila G., A new SQL-like operator for mining association rules, w: Proc. of 22nd Int. Conf. on Very Large Data Bases, 1996.
- [27] Cestnik B., Estimating probabilities: a crucial task in machine learning, w: Proc. ECAO 90, Stockholm, August, 1990.
- [28] Cestnik B., Kononenko I., Bratko I., Assistant 86. A knowledge elicitation tool for sophisticated users, w: Bratko I., Lavrac N. (red.), Progress in Machine Learning, Sigma Press, Wilmshow, 1987, s. 31-45.
- [29] Chan C.C., Batur V., Srinivasan A., Determination of quantization intervals in rule based model of dynamic systems, w: Proc. IEEE Conf. on Systems, Man and Cybernetics, Charlottesville, 1991, s. 1719-1723.
- [30] Chan P.K., Stolfo S.J., Experiments on multistrategy learning by meta-learning, w: Proceedings of the Second International Conference on Information and Knowledge Management, 1993, s. 314-323.
- [31] Chan P.K., Stolfo S.J., A comparative evaluation of voting and meta-learning on partitioned data, w: Proceedings of the 12th International Conference on Machine Learning, San Francisco, 1995, s. 90-98.
- [32] Chiu D.K.Y., Wong A.K.C., Cheung B., Information discovery through hierarchical maximum entropy discretization and synthesis, w: [173], s. 125-140.
- [33] Chmielewski M.R., Grzymala-Busse J.W., Global discretization of continuous attributes as preprocessing for machine learning, w: [132], s. 294-297.
- [34] Cichosz P., Systemy uczące się, Warszawa, WNT 2000.
- [35] Cios K.J., Pedrycz W., Świnarski R.W., Data mining methods for knowledge discovery, Dordrecht, Kluwer, 1999.
- [36] Clark P., Boswell R., Rule induction with CN2: some recent improvements, w: Kodratoff Y. (red.), Proceedings of 5th European Working Session on Learning – ESWL 91 Porto, Portugal, Springer-Verlag, 1991, s. 151-163.
- [37] Clark P., Niblett T., The CN2 induction algorithm, Machine Learning, 1989, 3, s. 261-283.
- [38] Dietterich T.G., Ensemble methods in machine learning, w: Kittler J., Roli F. (red.), Proc. of 1st Int. Workshop on Multiple Classifier Systems, 2000, s. 1-15.
- [39] Dietterich T.G., Bakiri G., Solving multiclass learning problems via error-correcting output codes, Journal of Artificial Intelligence Research, 1995, 2, s. 263-286.
- [40] Domingos P., Unifying instance-based and rule-based induction, Machine Learning, 1996, 24, s. 141-168.



- [41] Dougherty J., Kohavi R., Sahami M., Supervised and unsupervised discretizations of continuous features, w: Proc. 12th Int. Conf. on Machine Learning, Morgan Kaufmann, 1995, s. 194-202.
- [42] Doumpos M., Zopounidis C., Rough sets based rule induction and multivariate statistical classification: a simulation study, Computational Economics Journal, 2001, (zgłoszone do czasopisma).
- [43] Dubois D., Prade H., Rough fuzzy sets and fuzzy rough sets, International Journal of General Systems, 1990, 17, s.191-209.
- [44] Dubois D., Lang J., Prade H., Fuzzy sets in approximate reasoning, Fuzzy Sets and Systems, 1991, 40, s. 203-244.
- [45] Dzeroski S., Cestnik B., Petrovski I., Using the m-estimate in rule induction, Journal of Computing and Information Technology, 1993, 1, s. 37-46.
- [46] Eijkel G., Rule induction, w: [13], s. 195-216.
- [47] Fayyad U.M., Irani K.B., On the handling of continuous-valued attributes in decision tree generation, Machine Learning, 1992, 8, s. 87-102.
- [48] Fayyad U.M., Irani K.B., Multi-interval discretization of continuous-valued attributes for classification learning, w: Proc. of 13th Int. Joint Conf. on AI, Morgan Kaufmann, 1993, s. 1022-1027.
- [49] Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R. (red.), Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, Cambridge Mass., 1996.
- [50] Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R., From data mining to knowledge discovery, w: [49], s. 1-36.
- [51] Fayyad U.M., Djorgovski S., Weir N., Automating the analysis and cataloging of sky surveys, w: [49], s. 471-495.
- [52] Frawley W.J., Piatetsky-Shapiro G., Matheus C., Knowledge discovery in databases – An overview, w: [173], s.1-30.
- [53] Freita A., An objective measure of rule surprisngness, w: Proc. of the 2nd European Symposium on Principles of Data Mining and Knowledge Discovery, 1998, s. 1-9.
- [54] Freund Y., Boosting a weak learning algorithms by majority, Information and Computation, 1995, 121 (2), s. 256-285.
- [55] Friedman J.H., Another approach to polychotomous classification, Technical Report, Stanford University, 1996.
- [56] Gago P., Bento C., A metric for selection of the most promising rules, w: Proc. of the 2nd European Symposium on Principles of Data Mining and Knowledge Discovery, 1998, s. 19-27.
- [57] Gains B., Transforming rules and trees into comprehensible knowledge structures, w: [49], s. 205-228.
- [58] Gama J., Combining classification algorithms, PhD Thesis, University of Porto, 1999.
- [59] Greco S., Matarazzo B., Słowiński R., Rough approximation of preference relation by dominance relations, ICS Research Report 16/1996, Warsaw University of Technology; także w: European Journal of Operational Research, 1999, 117, s. 63-83.
- [60] Greco S., Matarazzo B., Słowiński R., A new rough set approach to evaluation of bankruptcy risk, w: Zopounidis C. (red.), Operational Tools in the Management of Financial Risk, Dordrecht, Boston: Kluwer Academic Publishers, 1998, s. 121-136.
- [61] Greco S., Matarazzo B., Słowiński R., Fuzzy similarity relation as a basis for rough approximations, w: [177], s. 283-289.

- [62] Greco S., Matarazzo B., Słowiński R., A new rough set approach to multicriteria and multiattribute classification, w: [177], s. 283-289.
- [63] Greco S., Matarazzo B., Słowiński R., The use of rough sets and fuzzy sets in MCDM, w: Gal T., Stewart T., Hanne T. (red.), *Advances in Multiple Criteria Decision Making*, Kluwer Academic Publishers, 1999, rozdział 14, s. 14.1-14.59.
- [64] Greco S., Matarazzo B., Słowiński R., On joint use of indiscernibility, similarity and dominance in rough approximation of decision classes, w: *Proc. of 5th Int. Conf. Decision Sciences Institute*, Vol II, July 4-7, Athens-Greece, 1999, s. 1380-1382.
- [65] Greco S., Matarazzo B., Słowiński R., Zanakis S., Rough set analysis of information tables with missing values, w: *Proc. of 5th Int. Conf. Decision Sciences Institute*, Vol II, July 4-7, Athens-Greece, 1999, s. 1359-1362.
- [66] Greco S., Matarazzo B., Słowiński R., Handling missing values in rough set analysis of multi-attribute and multi-criteria decision problems, w: [273], s. 146-157.
- [67] Greco S., Matarazzo B., Słowiński R., Rough set processing of vague information using fuzzy similarity relations, w: Calude C.S., Paun G. (red.), *Finite vs infinite: contributions to an eternal dilemma*, Berlin, Springer, 2000, s. 149-173.
- [68] Greco S., Matarazzo B., Słowiński R., Extension of the rough set approach to multicriteria decision support, *INFOR*, 2000, 38(3), s. 161-196.
- [69] Greco S., Matarazzo B., Słowiński R., Rough sets theory for multicriteria decision analysis, *European J. of Operational Research*, 2001, 129 (1), s. 1-47.
- [70] Greco S., Matarazzo B., Słowiński R., Stefanowski J., Variable consistency model of dominance-based rough set approach, w: [278], s. 138-149.
- [71] Greco S., Matarazzo B., Słowiński R., Stefanowski J., An algorithm for induction of decision rules consistent with dominance principle, w: [278], s. 266-275.
- [72] Grzymala-Busse J.W., On the unknown attribute values in learning from examples. *Proc. of Int. Symp. on Methodologies for Intelligent Systems*, 1991, s. 368-377.
- [73] Grzymala-Busse J.W., LERS – a system for learning from examples based on rough sets, w: [215], s. 3-18.
- [74] Grzymala-Busse J.W., Managing uncertainty in machine learning from examples, w: *Proc. of 3rd Int. Symp. on Intelligent Systems*, Wigry, IPI PAN Press, Warszawa, 1994, s. 70-84.
- [75] Grzymala-Busse J.W., Discretization of numerical attributes, rozdział C.3.4. w Zyt-kow J., Klosgen W. (red.), *Handbook of Knowledge Discovery*, 2001 (w druku).
- [76] Grzymala-Busse J.W., Grzymala-Busse W.J., Goodwin L.K., A closest fit approach to missing attribute values, w: [273], s. 405-414.
- [77] Grzymala-Busse J.W., Grzymala-Busse W.J., Zhang X., Increasing sensitivity of pre-term birth predication by changing rule strength, w: *Proceedings of the VIIIth Intelligent Information Systems*, Ustron, IPI PAN Press, 1999, s. 127-136.
- [78] Grzymala-Busse J.W., Lakshmanan A., LEM2 with interval extension: an induction algorithm for numerical attributes, w: [258], s. 67-73.
- [79] Grzymala-Busse J.W., Ming Hu, A comparison of several approaches to missing attribute values in data mining, w: [278], s. 340-349.
- [80] Grzymala-Busse J.W., Stefanowski J., Discretization of numerical attributes by direct use of the LEM2 induction algorithm with interval extension, w: *Proc. of the VI Int. Symp. on Intelligent Information Systems*, Zakopane, 1997, s. 149-158.
- [81] Grzymala-Busse J.W., Stefanowski J., Three approaches to numerical attribute discretization for rule induction, *International Journal of Intelligent Systems* 2001, 16

- (1), s. 29-38.
- [82] Grzymala-Busse J.W., Stefanowski J., Ziarko W., Rough sets: facts versus misconceptions, *Int. Journal Informatica*, 1996, 20, s. 455-464.
  - [83] Grzymala-Busse J.W., Wang A.Y., Modified algorithms LEM1 and LEM2 for rule induction from data with missing attribute values, w: *Proc. of 5th Int. Workshop on Rough Sets and Soft Computing (RSSC'97) at JCIS'97*, 1997, s. 69-72.
  - [84] Grzymala-Busse J.W., Zou X., Classification strategies using certain and possible rules, w: [177], s. 37-44.
  - [85] Hadjmichael M., Wasilewska A., Rough sets-based study of voter preferences in 1988 USA presidential election, w: [215], s. 137-152.
  - [86] Han J., Kamber M., *Data mining: Concepts and techniques*, San Francisco, Morgan Kaufmann, 2000.
  - [87] Hanley J.A., McNeil B.J., The meaning and use of the area under a receiver operating characteristic (roc) curve, *Radiology*, 1982, 143, s. 29-36.
  - [88] Hansen L., Salamon P., Neural network ensembles, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990, 12 (10), s. 993-1001.
  - [89] Hastie T., Tibshirani R., Classification by pairwise coupling, *Proc. NIPS97*.
  - [90] Holsheimer M., Siebes A., *Data mining: The search for knowledge in databases*. Technical Report, CS-R9406, CWI, 1994.
  - [91] Holsheimer M., Kersten M.L., Siebes A., *Data Surveyor: Searching the Nuggets in Parallel*, w: [49], s. 447-467.
  - [92] Holte R.C., Very simple classification rules perform well on most commonly used datasets, *Machine Learning*, 1993, 11, s. 63-91.
  - [93] Hyafil L., Rivest R.L., Constructing optimal binary trees is NP-complete, *Information Processing Letters*, 1976, 5 (1), s. 15-17.
  - [94] Iglesia B., Debuse J.C.W., Rayward-Smith V.J., Discovering knowledge from commercial databases using modern heuristic techniques, w: *Proceedings of KDD-96*, 1996, s. 44-49.
  - [95] Jelonek J., Generalization capability of homogeneous voting classifier based on partially replicated data, w: *Proc. of the AAAI-96 Workshop on Integrating Multiple Learned Models for Improving and Scaling Machine Learning Algorithms*, 1996, Portland, s. 47-52.
  - [96] Jelonek J., Zastosowanie złożonego systemu klasyfikacyjnego  $n^2$  z mechanizmem konstruktywnej indukcji cech do wieloklasowych problemów uczenia maszynowego, *Rozprawa doktorska*, Inst. Informatyki Politechniki Poznańskiej 2000.
  - [97] Jelonek J., Stefanowski J., Using  $n^2$ -classifier to solve multiclass learning problems, *Raport badawczy Instytutu Informatyki Politechniki Poznańskiej RB-01/97*, listopad 1997.
  - [98] Jelonek J., Stefanowski J., Features subset selection for classification of histological images, *Artificial Intelligence in Medicine*, 1997, 9, s. 227-239.
  - [99] Jelonek J., Stefanowski J., Experiments on solving multiclass learning problems by  $n^2$ -classifier, w: *Proceedings 10th European Conference on Machine Learning*, Chemnitz April 21-24, 1998, Springer, LNAI no.1398, s. 172-177.
  - [100] Jelonek J., Krawiec K., Stefanowski J., Comparative study of feature subset selection techniques for machine learning tasks, w: *Proc. VII Int. Symp. on Intelligent Information Systems*, Malbork, 1998, IPI PAN Press, s. 68-77.
  - [101] John G., Kohavi R., Pflieger K., Irrelevant features and the subset selection problem,

- w: Proceedings of the Eleventh International Machine Learning Conference, Morgan Kaufmann, 1994, s. 121-129.
- [102] Kamber M., Shingal R., Evaluating the interestingness of characteristic rules, w: Proc. of 2nd Int. Conf. on Knowledge Discovery and Data Mining, 1996, s. 263-266.
- [103] Keeney R.L., Raiffa H., Decision with multiple objectives – preferences and value tradeoffs, New York, Wiley, 1976.
- [104] Kerber R., Chimerge: discretization of numeric attributes, w: Proc. 10th Nat. Conf. on AI, MIT Press, 1992, s. 123-128.
- [105] Kitainik L., Fuzzy decision procedures with binary relations, Dordrecht, Kluwer Academic, 1993.
- [106] Klemettinen M., Mannila H., Ronkainen P., Toivonen H., A. Inkeri Verkamo. Finding interesting rules from large sets of discovered association rules, w: Proceedings of the third int. conference on information and knowledge management (CIKM'94), 1994, Geithersburg, Maryland, ACM Press, s. 401-407.
- [107] Kohavi R., Sahami M., Error-based and entropy-based discretization of continuous features, w: Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, 1996, s. 114-119.
- [108] Kohavi R., Sommerfield D., Feature subset selection using the wrapper method: overfitting and dynamic search space topology, w: Proc. of the First Int. Conference on Knowledge Discovery and Data Mining, Montreal, AAAI Press, 1995, s. 192-197.
- [109] Komorowski J., Agotness T., Ohrn A., Finding small high performance subsets of induced sets, w: Proc. of the 7th Congress on Intell. Techniques and Soft Computings, EUFIT'99, Aachen, 1999.
- [110] Komorowski J., Pawlak Z., Polkowski L., Skowron A., Rough Sets: tutorial, w: [163], s. 3-98.
- [111] Kononenko I., Estimating attributes: analysis and extensions of RELIEF, w: Proc. European Conf. on Machine Learning, Springer Verlag, 1994, s. 171-182.
- [112] Kononenko I., Bratko I., Information based evaluation criterion for classifier performance, Machine Learning, 1991, 6, s. 67-80.
- [113] Kononenko I., Bratko I., Kukar M., Application of machine learning to medical diagnosis, w: [144], s. 389-408.
- [114] Kowalczyk W., Rough data modeling, a new technique for analyzing data, w: [175], s. 139-143.
- [115] Krawiec K., Słowiński R., Vanderpooten D., Learning decision rules from similarity based rough approximation, w: [176], s. 37-54.
- [116] Krusińska E., Słowiński R., Stefanowski J., Discriminant versus rough sets approach to vague data analysis, Journal of Applied Stochastic Models and Data Analysis, 1992, 8, s. 43-56.
- [117] Krusińska E., Stefanowski J., Stromberg J.E., Comparability and usefulness of newer and classical data analysis techniques. Application in medical domain classification, w: Didey E. i in. (red.), New Approaches in Classification and Data Analysis, Springer Verlag, Studies in Classification, Data Analysis and Knowledge Organization, 1994, s. 644-652.
- [118] Kryszkiewicz M., Rough set approach to incomplete information system. ICS Research Report 55/95, Warsaw University of Technology; także w: International Journal of Information Sciences, 1998, 112, s. 39-49.
- [119] Kryszkiewicz M., Generation of rules from incomplete information systems, w: Ko-

- morowski J., Zytkow J. (red.), Proc. of the First European Symposium on Principles of Knowledge Discovery: PKDD'97, Trondheim, Springer LNAI no. 1263, Springer, 1997, s. 156-166.
- [120] Kryszkiewicz M., Properties of incomplete information systems in the framework of rough sets, w: [175], s. 422-450.
- [121] Kryszkiewicz M., Strong rules in large databases, w: Proceedings of Seventh International Conference IPMU98 Information Processing and Management of Uncertainty in Knowledge-based Systems, Paris, 1998, vol. 2, s. 1520-1527.
- [122] Kryszkiewicz M., Comparative study of alternative types of knowledge reduction in inconsistent systems, *Int. Journal of Intelligent Systems*, 2001, 16(1), s. 105-120.
- [123] Kryszkiewicz M., Rybinski H., Computation of reducts of composed information systems, *Fundamenta Informaticae*, 1996, 27, s. 183-195.
- [124] Kubat M., Bratko I., Michalski R.S., Review of machine learning methods, w: [144], s. 3-70.
- [125] Langley P., *Elements of machine learning*, San Francisco, Morgan Kaufmann, 1996.
- [126] Langley, P., Simon, H.A., Fielded applications of machine learning, w: [144], s. 113-129.
- [127] Larichev O., Moshkovich H., An approach to ordinal classification problems, *International Transactions on Operation Research*, 1994, 1(3), s. 375-385.
- [128] Lenarcik A., Piasta Z., Discretization of condition space, w: [215], s. 373-390.
- [129] Lenarcik A., Piasta Z., Probabilistic rough classifiers with mixture of discrete and continuous attributes, w: [131], s. 373-384.
- [130] Lin T.Y., Rough set theory in very large databases, w: Proceedings of CESA'96, vol. 2, Lille, July 1996, s. 936-941.
- [131] Lin T.Y., Cecrone N. (red.), *Rough Sets and Data Mining*, Kluwer Academic Publishers, 1997.
- [132] Lin T.Y., Wildberger A. (red.), *Soft Computing: Rough Sets, Fuzzy Logic, Neural Networks, Uncertainty Management, Knowledge Discovery*, San Diego, Simulation Councils Inc., 1995.
- [133] Liu B., Hsu W., Chen S., Using general impressions to analyse discovered classification rules, w: Proc. of the 3rd Int. Conference on Knowledge Discovery and Data Mining, 1997, s. 31-36.
- [134] Loken T., Komorowski J., Rough modeling – a bottom-up approach to model construction, *International Journal of Applied Mathematics and Computer Science*, 2001 (zgłoszone do czasopisma).
- [135] Łukasiewicz J., *Die logischen Grundlagen der Wahrscheinlichkeitsrechnung*, Kraków, 1913; także w: Borkowski L. (red.), *Jan Łukasiewicz – Selected Works*, Amsterdam, North Holland Publishers, 1970.
- [136] Major J., Mangano J., Selecting among rules induced from hurricane database, *Journal of Intelligent Information Systems*, 4, 1995.
- [137] Mannila H., Methods and problems in data mining, w: Proceedings of Int. Conference on Database Theory ICDT'97, Greece, 1997, s. 41-55.
- [138] Mannila H., Data mining: machine learning, statistics, and databases, w: Proc. of the Eight Int. Conf. on Scientific and Statistical Database Management, Stockholm June 18-20, 1996, s. 1-8.
- [139] Mehta M., Agrawal R., Rissanen J., SLIQ: a fast scalable classifier for data mining, w: Proc. of the 5th Int. Conf. on Extending Database technology (EDBT-96), 1996.

- [140] Michalski R.S., On the quasi-minimal solution of the general covering problem, w: Proc. of 5th Int. Symp. on Information Processing FCIP'69, Bled, 1969, s. 125-128.
- [141] Michalski R.S., A theory and methodology of inductive learning, w: Michalski R.S., Carbonell J.G, Mitchell T.M. (red.), Machine Learning: An Artificial Intelligence Approach, Morgan Kaufman, 1983, s. 83-134.
- [142] Michalski R.S., Machine Learning, Data Mining and Knowledge Discovery. Principles and Applications, w: Tutorials of Intelligent Information Systems, IIS'97, Zakopane, IPI PAN Press, 1997.
- [143] Michalski R.S, Data mining and knowledge discovery, a review of issues and a multistrategy approach, w: [144], s. 71-112.
- [144] Michalski R.S., Bratko I., Kubat M. (red.), Machine learning and data mining, John Wiley & Sons, 1998.
- [145] Michalski R.S., Mozetic I., Hong J., Lavrac N., The multi-purpose incremental learning system AQ15 and its testing application to three medical domains, w: Proc. of the 5th Nat. Conf. on AI, 1986, s. 1041-1045.
- [146] Michalski R.S., Tecuci G. (red.), Machine learning: a multistrategy approach, vol. 4, Morgan Kaufmann, 1994.
- [147] Mienko R., Stefanowski J., Toumi K., Vanderpooten D., Discovery-oriented induction of decision rules, Cahier du Lamsade no. 141, Paris, Universite de Paris Dauphine, septembre 1996.
- [148] Mitche D., Spiegelhalter C.J., Taylor C. (red.), Machine Learning, Neural and Statistical Classification, Elis Horwood, 1994.
- [149] Mitchell T., Machine Learning, Boston, Mac-Graw Hill, 1997.
- [150] Moczulski W., Metody pozyskiwania wiedzy dla potrzeb diagnostyki maszyn. Zeszyty Naukowe Politechniki Śląskiej. Monografie, Mechanika, z. 130, Gliwice 1997.
- [151] Morzy T., Wojciechowski M., Zakrzewicz M., Data mining support in database management systems, w: Proc. 2nd Int. Conf. on Data Warehousing and Knowledge Discovery, DaWaK 2000, Springer LNCS 1874, 2000, s. 382-392.
- [152] Nguyen Hoa S., Data regularity and application in data mining, Rozprawa doktorska, Uniwersytet Warszawski 1999.
- [153] Nguyen Hoa S., Skowron A., Synak P., Discovery of data patterns with applications to decomposition and classification problems, w: [176], s. 55-97.
- [154] Nguyen H. Son, Discretization of real value attributes. Boolean reasoning approach. Rozprawa doktorska. Uniwersytet Warszawski, Warszawa 1997.
- [155] Nguyen H. Son, Rule induction from continuous data: New discretization concepts, w: Proc. of III Joint Conf. on Information Sci., 1997, s. 81-84.
- [156] Nguyen H. Son, Nguyen Hoa S., Discretization methods in data mining, w: [175], s. 451-482.
- [157] Nguyen H. Son, Skowron A., Quantization of real valued attributes: Rough set and boolean reasoning approaches, w: Proceedings of the 2nd Joint Annual Conf. on Information Sciences, Wrightsville Beach, USA, 1995, s. 34-37.
- [158] Nguyen H. Son, Ślęzak D., Approximate reducts and association rules – correspondence and complexity results, w: [273], s. 137-145.
- [159] Nowicki R., Słowiński R., Stefanowski J., Rough sets analysis of diagnostic capacity of vibroacoustic symptoms, Computers and Mathematics with Applications, 1992, 24 (7), s. 109-123.
- [160] Nowicki R., Słowiński R., Stefanowski J., Evaluation of vibroacoustic symptoms by

- means of the rough sets theory, *Computers in Industry*, 1992, 20, s. 141-152.
- [161] Ohm A., Komorowski J., Skowron A., Synak P., The design and implementation of knowledge discovery toolkit based on rough sets – the ROSETTA, w: [176], s. 376-399.
- [162] Owens D.K., Sox H.C., Medical decision making: probabilistic medical reasoning, w: Shorliffe E.H., Perreault L.E., Wiederhold G., Fagan L.M. (red.), *Medical Informatics Computer Applications in Health Care*, Addison Wesley, 1990, 70-116.
- [163] Pal S.K., Skowron A. (red.), *Rough fuzzy hybridization. A new trend in decision-making*, Hongkong, Springer-Verlag, 1999.
- [164] Pawlak Z., *Rough sets*, *Int. J. Computer and Information Sci.*, 1982, 11, s. 341-356.
- [165] Pawlak Z., *Rough sets. Theoretical aspects of reasoning about data*, Dordrecht, Kluwer Academic Publishers, 1991.
- [166] Pawlak Z., Decision rules, Bayes' rule and rough sets, w: [273], s. 1-9.
- [167] Pawlak Z., Rough sets and decision algorithms, w: [278], s. 1-16.
- [168] Pawlak Z., Drawing conclusions from data – the rough set way, *Int. Journal of Intelligent Systems*, 2001, 16(1), s. 3-12.
- [169] Pawlak Z., Grzymala-Busse J., Słowiński R., Ziarko W., *Rough sets. Communications of the ACM*, 1995, 38 (11), s. 89-95.
- [170] Pawlak Z., Skowron A., A rough set approach for decision rules generation, *ICS Research Report 23/93*, Warsaw University of Technology, 1993.
- [171] Pawlak Z., Słowiński K., Słowiński R., Rough classification of patients after highly selected vagotomy for duodenal ulcer, *International J. Man-Machine Studies*, 1986, 24, s. 413-433.
- [172] Pawlak Z., Słowiński R., Rough set approach to multi-attribute decision analysis, *European Journal of Operational Research*, 1994, 72, s. 443-459.
- [173] Piatetsky-Shapiro G., Matheus C. (red.), *Knowledge discovery in databases*, AAAI/MIT Press, 1991.
- [174] Piatetsky-Shapiro G., Discovery, analysis and presentation of strong rules, w: [173], s. 229-247.
- [175] Polkowski L., Skowron A. (red.), *Rough sets in knowledge discovery, vol 1, Methodology and applications*, Physica-Verlag, Heidelberg, 1998.
- [176] Polkowski L., Skowron A. (red.), *Rough sets in knowledge discovery, vol 2, Applications, Case studies and software systems*, Physica-Verlag, Heidelberg, 1998.
- [177] Polkowski L., Skowron A. (red.), *Rough sets and current trends in computing*, *Proceedings of the RSCTC'98 Conference*, Warszawa 1998, *Lecture Notes in Artificial Intelligence 1424*, Springer Verlag, 1998.
- [178] Polkowski L., Skowron A., Żytkow J., Tolerance based rough sets, w: [132], s. 55-58.
- [179] Polkowski L., Lin T.Y., Tsumoto S. (red.), *Rough sets methods and applications: New developments in knowledge discovery in information systems*, Heidelberg, Physica-Verlag, 2000.
- [180] Provost F., Fawcett T., Kohavi R., The case against accuracy estimation for comparing induction algorithms, w: *Proc. of the 15th Int. Conf. on Machine Learning*, 1998.
- [181] Provost F., Kolluri V., A survey methods for scaling up inductive algorithms, *Journal Data Mining and Knowledge Discovery*, 1999, 3(2), s. 131-169.
- [182] Prędko B., Słowiński R., Stefanowski J., Susmaga R., Wilk S., ROSE – Software implementation of the rough set theory, w: [177], s. 605-608.

- [183] Pyle D., Data preparation for data mining, San Francisco, Morgan Kaufmann, 1999.
- [184] Quinlan J.R., Unknown values in induction, w: Proc. 6th Int. Workshop on Machine Learning, Morgan Kaufmann, San Mateo, CA, 1989, s. 31-37.
- [185] Quinlan J.R., C4.5: Programs for Machine Learning, San Francisco, Morgan Kaufmann, 1993.
- [186] Quinlan J.R., Bagging, boosting and C4.5, w: Proceedings of the 13th National Conference on Artificial Intelligence, 1996, s. 725-730.
- [187] Quinlan J.R., Improved use of continuous attributes in C4.5, Journal of AI Research, 1996, 4, s. 77-90.
- [188] Quinlan J.R., Rivest R.L., Inferring decision trees using the minimum description length principle, Information and Computation, 1989, 80, s. 227-248.
- [189] Riddle P., Segal R., Etzioni O., Representation design and brute-force induction in a Boeing manufacturing domain, Journal of Applied Artificial Intelligence, 1994, 8, s. 125-147.
- [190] Riesbeck C.K., Schank R.C., Inside case-based reasoning, Lawrence Erlbaum: Hillsdale, 1989.
- [191] Rissanen J., Minimum-description length principle, w: Kotz S., Johnson N.L. (red.), Encyclopedia of Statistical Sciences, New York, Wiley, 1985, s. 523-526.
- [192] Roy B., Méthodologie Multicritère d'Aide à la Décision, Economica, Paris 1985.
- [193] Roy B., The outranking approach and the foundation of ELECTRE methods, Theory and Decision, 1991, 31, s. 49-73.
- [194] Rubin S., Michalowski W., Słowiński R., Developing an emergency room diagnostic check list using rough sets – a case study of appendicitis, w: Anderson J., Katzper M. (red.), Simulation in the Medical Sciences, Proc. of the 1996 Western Conference of the Society for Computer Simulation, 1996, s. 19-24.
- [195] Salzberg S.L., On comparing classifiers: pitfalls to avoid and a recommended approach, Journal: Data mining and knowledge discovery, 1997, 1(3), s. 317-328.
- [196] Schapire R.E., The strength of weak learnability, Machine Learning, 1990, 5, s. 197-226.
- [197] Segal R., Etzioni O., Learning decision lists using homogenous rules, w: Proceedings of the AAAI-94 Conference, 1994, s. 619-625.
- [198] Shan N., Ziarko W., Data-based acquisition and incremental modification of classification rules, Journal Computational Intelligence, 1995, 11 (2), s. 357-370.
- [199] Shavlik J.W., Dietterich T.G. (red.), Readings in machine learning, Morgan Kaufman, 1990.
- [200] Shortlife E.H., Barnett G.O., Medical data: their acquisition, storage and use, w: Shorliffe E.H., Perreault L.E., Wiederhold G., Fagan L.M. (red.), Medical Informatics. Computer Applications in Health Care, Addison Wesley, 1990, s. 37-69.
- [201] Silberschatz A., Tuzhilin A., What makes patterns interesting in knowledge discovery systems, IEEE Trans. on Knowledge and Data Engineering, 1996, 8 (6), s. 970-974.
- [202] Skowron A., Boolean reasoning for decision rules generation, w: Komorowski J., Ras Z. (red.), Methodologies for Intelligent Systems, LNAI 689, Berlin, Springer-Verlag, 1993, s. 295-305.
- [203] Skowron A., Extracting laws from decision tables, International Journal Computational Intelligence, 1995, 11(2), s. 371-388.
- [204] Skowron A., Synthesis of adaptive decision systems from experimental data, w: Aamodt A., Komorowski J. (red.), Proc 5th Scandinavian Conference on Artificial



- Intelligence, IOS Press, 1995, s. 220-238.
- [205] Skowron A., Polkowski L., Decision algorithms: a survey on rough set theoretic methods, *Fundamenta Informaticae*, 1997, 30, s. 345-358.
- [206] Skowron A., Rauszer C., The discernibility matrices and functions in information systems, w: [215], s. 331-362.
- [207] Skowron A., Stepaniuk J., Decision rules based on discernibility matrices and decision matrices, w: Lin T.Y. (red.), *Proc. of the 3rd Int. Workshop on Rough Sets and Soft Computing*, San Jose, 1994, s. 602-609.
- [208] Skowron A., Stepaniuk J., Tolerance approximation spaces, *Fundamenta Informaticae*, 1996, 27, s. 245-253.
- [209] Słowiński K., Słowiński R., Sensitivity of rough classification to changes in norms of attributes, w: [215], s. 363-373.
- [210] Słowiński K., Słowiński R., Stefanowski J., Rough sets approach to analysis of data from peritoneal lavage in acute pancreatitis, *Medical Informatics*, 1988, 13, s. 143-159.
- [211] Słowiński K., Stefanowski J., On limitations of using rough set approach to analyse non-trivial medical information systems, w: [258], s. 176-184.
- [212] Słowiński K., Stefanowski J., Multistage rough set analysis of therapeutic experience with acute pancreatitis, w: [176], s. 272-295.
- [213] Słowiński K., Stefanowski J., Medical information systems – problems with analysis and way of solution, w: [163], s. 301-315.
- [214] Słowiński K., Stefanowski J., Siwiński D., Application of rule induction and rough sets to verification of magnetic resonance diagnosis, 2000 (wysłane do czasopisma).
- [215] Słowiński R. (red.), *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory*, Dordrecht, Kluwer Academic Publishers, 1992.
- [216] Słowiński R., Rough set learning of preferential attitude in multi-criteria decision making, w: Komorowski J., Ras Z. (red.), *Methodologies for Intelligent Systems*, LNAI 689, Springer-Verlag, 1993, s. 642-651.
- [217] Słowiński R., Rough set approach to decision analysis, *AI Expert*, March 1995, s. 19-25.
- [218] Słowiński R., Rozszerzenie teorii zbiorów przybliżonych na atrybuty ze skalą preferencji, w: *Materiały Konferencji Informatyka Teoretyczna: Metody Analizy Informacji Niekompletnej i Rozproszonej*, Białystok 2000, s. 114-128.
- [219] Słowiński R., Stefanowski J., Rough classification in incomplete information systems, *Mathematical and Comput. Modelling*, 1989, 12 (10/11), s. 1347-1357.
- [220] Słowiński R., Stefanowski J., Handling various types of uncertainty in the rough set approach, w: [275], s. 366-376.
- [221] Słowiński R., Stefanowski J., Rough classification with valued closeness relation, w: Diday E. i in. (red.), *New Approaches in Classification and Data Analysis*, *Studies in Classification, Data Analysis and Knowledge Organisation*, Springer Verlag, 1994, s. 482-489.
- [222] Słowiński R., Stefanowski J., Using expert's knowledge in a rule-based classification of objects, w: *Proceedings of the 4th Int. Symposium on Intelligent Information Systems*, Augustów 5-9 06 1995, IPI PAN Press, 1996, s. 52-61.
- [223] Słowiński R., Stefanowski J., Rough set reasoning about uncertain data, *Fundamenta Informaticae*, 1996, 27 (2-3), s. 229-244.
- [224] Słowiński R., Stefanowski J., Greco S., Matarazzo B., Rough sets processing of in-

- consistent information, *Control and Cybernetics*, 2000, 29 (1), s. 379-404.
- [225] Słowiński R., Stefanowski J., Susmaga R., Rough set analysis of attribute dependencies in technical diagnostics, w: [258], s. 384-291.
- [226] Słowiński R., Vanderpooten D., Similarity relation as a basis for rough approximations, Institute of Computer Science, Warsaw University of Technology, Research Report 53/95 1995; także w: P.P. Wang (red.), *Advances in machine intelligence & Soft-Computing*, vol. IV, Duke University Press, 1997, s. 17-33.
- [227] Słowiński R., Vanderpooten D., A generalized definition of rough approximations based on similarity, *IEEE Transactions on Data and Knowledge Engineering*, 2000, 12, s. 331-336.
- [228] Slovic P., Choice between equally-valued alternatives, *Journal of Experimental Psychology: Human Perception Performance*, 1975, 1, s. 280-287.
- [229] Smyth P., Goodman R.M., Rule induction using information theory, w: [173].
- [230] Stefanowski J., Rough sets theory and discriminant methods in an analysis of information systems, *Foundations of Computing and Decision Sci.*, 1992, 17 (2), s. 81-89.
- [231] Stefanowski J., Classification support based on the rough sets, *Foundations of Computing and Decision Sciences*, 1993, 18 (3-4), s. 371-380.
- [232] Stefanowski J., Dialogowe wspomaganie decyzji na podstawie wiedzy pozyskanej metodą zbiorów przybliżonych Rozprawa doktorska, Wydział Elektryczny Politechniki Poznańskiej, Poznań 1994.
- [233] Stefanowski J., Using valued closeness relation in classification support of new objects, w: [132], s. 324-327.
- [234] Stefanowski J., On rough set based approaches to induction of decision rules, w: [175], s. 500-529.
- [235] Stefanowski J., Rough set based rule induction techniques for classification problems, w: *Proc. 6th European Congress on Intelligent Techniques and Soft Computing*, vol. 1, Aachen Sept. 7-10, 1998, s. 109-113.
- [236] Stefanowski J., Handling continuous attributes in discovery of strong decision rules, w: [177], s. 394-401.
- [237] Stefanowski J., Multiple and hybrid classifiers, w: *Decision Theory, Language, Multimedia and Robotics – Methods and Applications. Proceedings of 2nd International Conference. PJIIT, Warszawa, październik, 2000.*
- [238] Stefanowski J., An algorithm for inducing a satisfactory set of decision rules in multicriteria and multiattribute sorting, *Research Report of the Institute of Computing Science, Poznań University of Technology, RA-007/2000, Poznań, październik 2000.*
- [239] Stefanowski J., Borkiewicz R., Interactive rule discovery of decision rules, w: *Proceedings of VIIIth Intelligent Information Systems, Ustroń 1999, IPI PAN Press, s. 112-116.*
- [240] Stefanowski J., Słowiński K., Rough sets as a tool for studying attribute dependencies in the urinary stones treatment data set, w: [131], s. 177-198.
- [241] Stefanowski J., Słowiński K., Rough set theory and rule induction techniques for discovery of attributes dependencies in medical information systems, w: Komorowski J., Zytkow J. (red.), *Proc. of the First European Symposium on Principles of Knowledge Discovery: PKDD'97, Trondheim 1997, Springer LNAI no. 1263, s. 36-46; Rozszerzona wersja w Bulletin of the Polish Academy of Sciences, Technical Sciences, 1998, 46 (2), s. 247-263.*
- [242] Stefanowski J., Tsoukias A., On the extension of rough sets under incomplete infor-

- mation, w: [273], s. 73-81.
- [243] Stefanowski J., Tsoukias A., Decision rules and valued tolerance, w: [278], s. 180-187.
  - [244] Stefanowski J., Tsoukias A., Incomplete information tables and rough classification, *Int. Journal of Computational Intelligence*, 2001 (przyjęte do druku).
  - [245] Stefanowski J., Vanderpooten D., A general two stage approach to rule induction from examples w: [275], s. 317-325.
  - [246] Stefanowski J., Vanderpooten D., Induction of decision rules in classification and discovery-oriented perspectives, *International Journal of Intelligent Systems* 2001, 16 (1), s. 13-28.
  - [247] Stefanowski J., Wilk S., Minimizing business credit risk by means of approach integrating decision rules and case based learning, Research Report, Instytut Informatyki PP, RA-001/99, luty 1999; przyjęte do druku w *Journal of Intelligent Systems in Accounting, Finance and Management* 2001.
  - [248] Stepaniuk J., Knowledge discovery by application of rough set models, *Prace IPI PAN* no. 887, Warszawa 1999.
  - [249] Stepaniuk J., Krętkowski M., Decision system based on tolerance rough sets, w: *Proc. of IV Int. Workshop on Intelligent Information Systems*, Augustów, 1995, s. 62-73.
  - [250] Surma J., Vanhoof K., Integrating rules and cases for the classification task, w: *Case Based Reasoning Research and Development. Proceedings of the First International Conference – ICCBR'95*; 1995, s. 325-334.
  - [251] Susmaga R., Analyzing discretization of continuous attributes given a monotonic discrimination function, *Journal Intelligent Data Analysis*, Vol. 1, No. 3 (Elsevier online journal <http://www-east.elsevier.com/ida>).
  - [252] Susmaga R., Experiments in incremental computation of reducts, w: [175], s. 530-553.
  - [253] Szczuka M.S., *Metody symboliczne i sieci neuronowe w konstrukcji klasyfikatorów*, Rozprawa doktorska, Instytut Matematyki, Uniwersytet Warszawski, 1999.
  - [254] Ślęzak D., Approximation reducts in decision tables, w: *Proc. of conf. IPMU'96*, vol. 3, Granada, 1996, s. 1159-1164.
  - [255] Ting K.M., Discretization of continuous-valued attributes and instance-based learning, Technical Report 491, University of Sydney, 1994.
  - [256] Torgo L., Controlled redundancy in incremental rule learning, w: *Proc. ECML-93*, 1993, s. 185-195.
  - [257] Tsumoto S., Modelling medical diagnostic rules based on rough sets, w: [177], s. 475-482.
  - [258] Tsumoto S. i. in. (red.), *Proceedings of the Fourth International Workshop on Rough Sets, Fuzzy Sets and Machine Discovery*, Tokyo, Nov. 6-8 1996, Tokyo University Press.
  - [259] Tsumoto S., Tanaka H., PRIMEROSE: Probabilistic rule induction method based on rough set and resampling methods, *International Journal Computational Intelligence*, 1995, 11 (2), s. 389-405.
  - [260] Toumi Kh., *Aide à la décision dans le cadre de la problématique de description: une approche inductive pour décrire et expliquer*, PhD Thesis, Université Paris Dauphine, 1996.
  - [261] Uthurusamy R., From data mining to knowledge discovery: current challenges and future directions, w: [49], s. 561-569.

- [262] Weiss S.M., Indurkha N., Predictive Data Mining, San Francisco, Morgan Kaufmann, 1999.
- [263] Weiss S.M., Kulikowski C.A., Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems, Morgan Kaufmann, 1991.
- [264] Wilk S., System wspomaganie decyzji integrujący wnioskowanie regułowe i wnioskowanie na podstawie podobieństwa do znanych przypadków, Praca magisterska, Instytut Informatyki PP, Poznań 1997.
- [265] Wolpert D., Stacked generalisation, Neural networks, 1992, 5, s. 241-259.
- [266] Wong S.K.M., Ziarko W., Ye Li W., Comparison of rough set and statistical methods in inductive learning, Int. Journal of Man-Machine Studies, 24, 1986, s. 53-72.
- [267] Wróblewski J., Finding minimal reducts using genetic algorithms, w: Proc. of the 2nd Joint Conference on Information Sciences, Wrightsville Beach, 1995, s. 186-189.
- [268] Van der Merckt T., Decision trees in numerical attribute spaces, w: Proc. of 13th Int. Joint Conf. on AI, Chambery, 1993, s. 1016-1021.
- [269] Xu L., Krzyżak A., Suean C.Y., Methods of combining multiple classifiers and their applications to handwriting recognition, IEEE Transactions on Systems, Man and Cybernetics, 1992, 22 (3), s. 418-435.
- [270] Yao Y.Y., Zhong N., An analysis of quantitative measures associated with rules, w: Proceedings of the Third Pacific-Asia Conference on Knowledge Discovery and Data Mining, LNAI 1574, Springer, 1999, s. 479-488.
- [271] Zembowicz R., Żytkow J., From contingency tables to various forms of knowledge in databases, w: [49], s. 329-352.
- [272] Zhang J., Michalski R.S., An integration of rule induction and exemplar-based learning for graded concepts, Machine Learning, 1995, 21, s. 235-267.
- [273] Zhong N., Skowron A., Ohsuga S. (red.), New Directions in Rough Sets, Data Mining and Granular-Soft Computing, LNAI 1711, Springer-Verlag, Berlin, 1999.
- [274] Ziarko W., Variable precision rough sets model, Journal of Computer and Systems Sciences, 1993, 46 (1), s. 39-59.
- [275] Ziarko W. (red.), Rough Sets, Fuzzy Sets and Knowledge Discovery, Springer-Verlag, 1994.
- [276] Ziarko W., Rough sets as a methodology for data mining, w: [175], s. 554-576.
- [277] Ziarko W., Shan N., KDD-R: A comprehensive system for knowledge discovery in databases using rough sets, w: [132], s. 93-96.
- [278] Ziarko W., Yao Y.Y., Proc. 2nd Int. Conference on Rough Sets and Current Trends in Computing, Banff, October 16-19, 2000.
- [279] Źak J., Stefanowski J., Determining maintenance activities of motor vehicles using rough sets approach, w: Proc. of Euromaintenance'94 Conference, Amsterdam 1994, s. 39-42.

Jerzy Stefanowski

## **Algorithms of rule induction for knowledge discovery**

### **Summary**

Rule induction methods for knowledge discovery are considered in this dissertation. Two perspectives of inducing decision rules are distinguished: prediction and description. Data are represented in, so called, decision tables where rows correspond to objects and columns correspond to attributes; each element of this table characterizes an object by an appropriate attribute. Attributes are divided into condition and decision ones. The condition attributes characterize objects independently of partitions induced by the decision attributes. In the problems of object classification, values of the decision attribute correspond to the decision classes and objects to examples of classification (learning examples). Decision rules are induced from learning examples and determine whether an object satisfying conditions (defined on some condition attributes) belongs to an appropriate decision class. If input data contain inconsistencies, rough set theory can be used to handle them. The dissertation discusses the current state of the research on rule induction and presents new algorithms. First, the method of rule induction specific for description perspective is proposed – an algorithm EXPLORE. Then, another algorithm MODLEM is introduced to handle directly numerical data. Two new generalizations of the rough sets theory for a case of incomplete decision tables, i.e. containing unknown attribute values, are proposed. The first generalization introduces the use of non-symmetric similarity relations in order to handle absent attribute values. The second generalization is based on the use of a valued tolerance relation and deals with missing attribute values. Further on, problems of discovering generalized decision rules from examples of multiple criteria sorting problems are considered. The rough sets approach based on the dominance relation – specific for multiple criteria decision problems – is here extended to a variable consistency approach. Two algorithms for inducing decision rules in both these approaches are introduced: DOMLEM and DOMApriori. The last methodological aspect deals with using rule induction inside multiple or hybrid classification systems. Within this context two proposals are presented: multiple classifier  $n^2$  and hybrid system integrating rule and case based learning. Finally, practical applications of proposed methods are discussed.