

Міністерство освіти і науки України
Дніпровський національний університет імені Олеся Гончара
Міністерство освіти і науки України
Дніпровський національний університет імені Олеся Гончара

Кваліфікаційна наукова
праця на правах рукопису

Єгошкін Данило Ігорович

УДК 004.891:004.82:519.7

ДИСЕРТАЦІЯ

**РОЗРОБКА МЕТОДІВ І АЛГОРИТМІВ АВТОМАТИЧНОЇ ГЕНЕРАЦІЇ ТА
ДОСЛІДЖЕННЯ СТРУКТУРИ НЕЧІТКОЇ БАЗИ ЗНАНЬ**

11 – Математика та статистика
113 – Прикладна математика

Подається на здобуття ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей,
результатів і текстів інших авторів мають посилання на відповідне джерело

Єгошкін Д. І. Єгошкін

Науковий керівник

Гук Наталія Анатоліївна

доктор фізико-математичних наук,
професор

Дніпро - 2023

АНОТАЦІЯ

Єгошкін Д. І. Розробка методів і алгоритмів автоматичної генерації та дослідження структури нечіткої бази знань. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 113 «Прикладна математика» (11 – Математика та статистика). – Дніпровський національний університет імені Олеся Гончара, м. Дніпро, 2023.

Дисертаційну роботу присвячено розробці методів та алгоритмів автоматичної генерації бази знань системи логічного виведення для задач класифікації на основі експериментальних даних, вдосконаленню методів логічного виведення, методів налаштування параметрів моделі представлення знань, методів доведення коректності бази знань та дослідженню якості побудованої системи логічного виведення.

В даний час системи, що засновані на знаннях, набули широке застосування в різних прикладних областях науки і техніки для розв’язання задач класифікації, прогнозування, розпізнавання образів, медичної та технічної діагностики, управління, автоматизованого проектування та інших.

Системи логічного виведення дозволяють автоматизувати процес прийняття рішень і в такий спосіб частково замінити фахівця-експерта в певній галузі, крім того, вони можуть передавати знання для навчання нових фахівців. Із застосуванням таких систем можна створювати опис моделей для погано формалізованих об’єктів і процесів та уникнути необхідності побудови складної і коштовної математичної моделі. Найчастіше об’єкти та процеси характеризуються невизначеністю, неповнотою вхідних даних, нелінійним характером внутрішніх зв’язків та закономірностей, наявністю значної кількості параметрів різних типів, невизначеністю структури та її змінюваністю в залежності від ситуації.

Основним компонентом систем логічного виведення є база знань, що складається з набору правил, за якими здійснюється логічне виведення. Правила бази знань описують взаємозв'язки в предметній області, які можна виявити на основі експериментальних даних, отриманих в результаті спостережень за об'єктом або процесом. Правила зображуються у формі чітких або нечітких продукцій та повинні адекватно описувати закономірності в наявних для аналізу даних.

Однак, зазначені особливості об'єктів та процесів призводять до необхідності формулювати надскладні системи правил, що мають враховувати взаємозв'язки між значною кількістю параметрів. Також складним етапом побудови системи логічного виведення є налаштування моделі.

Для формування бази знань у системах логічного виведення існує кілька підходів, до яких належить аналіз знань експертом (групою експертів) на основі досвіду, автоматичне формування бази правил з використанням методів інтелектуального аналізу даних і алгоритмів машинного навчання, або поєднання зазначених підходів. Процес генерування знань потребує значної аналітичної роботи з боку експерта та суттєво ускладнює розробку таких систем, тому саме розвитку другого підходу – автоматичної генерації бази правил та налаштуванню параметрів моделі на основі експериментальних даних – присвячено тему дисертаційного дослідження.

Розробка методів та алгоритмів автоматичної генерації баз знань суттєво спрощує процедури створення та налаштування систем логічного виведення, а також забезпечує контроль їх якості. Оскільки знання є динамічною структурою, що може розвиватися та змінюватися у процесі функціонування, у роботі розроблено механізм додавання нових знань до вже існуючої бази знань.

Слід зазначити, що якість функціонування експертної системи залежить від повноти та несуперечності правил, а також розміру бази правил. Велика кількість правил призводить як до зниження швидкості виконання

логічного виведення, що неприпустимо для систем, які працюють у режимі реального часу, так і до суперечливості накопичених знань та ефекту перенавчання. Тому розробка методів і алгоритмів, що дозволяють структурувати систему правил, визначати порядок їх викликів, визначати необхідну кількість правил, контролювати несуперечність і повноту бази правил є актуальною задачею, що вирішується у дисертаційній роботі. Використання такого підходу значно спростить і прискорить процес розробки системи, база знань якої, будується в процесі навчання за допомогою об'єктів навчальної вибірки.

Метою роботи є розробка методів та алгоритмів автоматичної генерації бази знань системи логічного виведення для задач класифікації на основі експериментальних даних, вдосконалення методів логічного виведення, методів налаштування параметрів моделі представлення знань, методів доведення коректності бази знань та дослідження якості побудованої системи логічного виведення.

Для представлення знань про об'єкти предметної області в роботі запропоновано продукційну модель представлення знань, що поєднує моделі Мамдані та Такагі-Сугено-Канг і використовує у консеквенті правила функцію належності вихідного об'єкта певному класу як ваговий коефіцієнт. Це дозволило визначити рівень впливу кожного правила на результат логічного виведення продукційної системи.

Із застосуванням кореляційного аналізу пропонується виявляти взаємозв'язків між змінними та приховані закономірності, патерни чи аномалії у наборах даних. Особлива увага приділяється обчисленню коефіцієнта кореляції Пірсона та виявленню мультиколінеарності як ключових аспектів аналізу великих обсягів навчальних даних. Додатково запропоновано використання діаграм точкового розсіювання класів для візуалізації розподілу даних, що сприятиме більш глибокому розумінню їхньої суті та виявленню взаємозв'язків.

Розроблено алгоритм автоматичного формування бази правил на основі навчальної вибірки для розв'язання задачі класифікації. Матриця антецедентів створюється за допомогою ознак об'єктів з деякої скінченної множини з використанням декартового добутку. Для формування вектора консеквентів використовується навчальна вибірка, що забезпечує можливість створювати нечітку базу правил та здійснювати налаштування моделі в автоматичному режимі. Удосконалено метод логічного виведення в частині вирішення конфліктів при прийнятті рішення в разі, коли системі не вдається однозначно визначити належність об'єкта певному класу. Для цього реалізовано процедуру порівняння відстані між об'єктами та класами на етапі класифікації. Для оцінки якості побудованої нечіткої системи логічного виведення та отриманих результатів класифікації обрано метрики, що визначають частину правильних відповідей моделі (ассурасу), точність (precision), повноту (recall), f-міру (f1-score), додатково використовується матриця невідповідностей.

Для здійснення процедури редукції бази правил запропоновано інтегральну метрику, що враховує кількість активацій певного правила та ступінь належності об'єктів навчальної вибірки правилу. Із використанням методу дихотомії з динамічним кроком та запропонованої метрики розроблено адаптивний алгоритм редукції бази правил. Одержана в результаті редукції база правил суттєво скорочується, але це не погіршує результат класифікації.

Для налаштування параметрів нечіткої продукційної моделі запропоновано автоматичне визначення границь термів з використанням вибіркового та міжквартильного середніх значень, що дозволило зменшити вплив наявних у навчальній вибірці аномальних об'єктів на результат класифікації та зробити процедуру визначення границь термів робастною.

Здійснено аналіз коректності бази правил шляхом доведення її повноти, несуперечливості, зв'язаності та мінімальності із використанням логіки Хоара та предикату найслабшої передумови $Q \Rightarrow WP(S, R)$. Доведення

проведено з використанням методу резолюцій та програмного додатку Simplify, що забезпечує автоматизацію перевірки коректності бази правил на етапі навчання системи.

Запропонований підхід було застосовано до розв'язання задач мультикласової класифікації на відомих модельних наборах даних Іриси Фішера (Iris Data Set) та видова популяція арктичних пінгвінів. Здійснено налаштування параметрів нечіткої продукційної моделі. Для визначення границь термів виконано аналіз ознак об'єктів навчальної вибірки та аналіз частотного розподілу для кожної ознаки об'єкта, а застосування міжквартильного середнього значення дозволило зменшити вплив наявних аномальних об'єктів в навчальній вибірці на результат класифікації.

Метод логічного виведення вдосконалено в частині вирішення конфлікту, що утворюється, коли об'єкт належить двом класам одночасно. Завдяки запропонованій метриці, що інтегрально враховує кількість активацій певного правила та ступінь належності об'єктів навчальної вибірки правилу, здійснено редукцію бази правил без втрати якості результату класифікації.

Наведено порівняння результатів класифікації з використанням запропонованого в роботі підходу і класичного алгоритму нечіткої класифікації.

Здійснено аналіз впливу розміру навчальної вибірки на результати класифікації та якість системи логічного виведення. Встановлено, що збільшення розміру навчальної вибірки призводить до покращення результатів класифікації та значень метрик якості.

Наукова новизна одержаних результатів полягає у наступному:

- дістав подальшого розвитку метод побудови нечіткої продукційної моделі для представлення знань про об'єкти предметної області на основі навчальної вибірки за допомогою розроблених алгоритмів навчання;
- запропоновано вид нечіткого продукційного правила, що поєднує моделі Мамдані та Такагі-Сугено-Канг та містить у консеквенті правила в

якості вагового коефіцієнту функцію, що визначає ступінь належності вихідної змінної до результуючої терм-множини;

- удосконалено методику побудови сукупності правил бази знань та алгоритм їх автоматичного формування за допомогою попереднього аналізу ознак навчальної вибірки, матричного представлення антецедентів та векторного представлення консеквентів правил;

- дістав подальшого розвитку метод логічного виведення з використанням бази знань нечіткої продукційної моделі для розв’язання задачі класифікації в умовах нечіткості та неповноти інформації про об’єкти предметної області; за наявності конфлікту при прийнятті рішення, що полягає у неспроможності системи однозначно визначити належність об’єкта певному класу, запропоновано процедуру порівняння відстані між об’єктами та класами;

- дістав подальшого розвитку метод ідентифікації та налаштування значень параметрів нечіткої продукційної моделі, зокрема автоматизовано визначення лінгвістичних змінних, терм-множин, границь термів, параметрів функцій належності, вагових коефіцієнтів правил, кількості правил;

- вперше для перевірки коректності автоматично побудованої бази правил за критеріями повноти, мінімальності, зв’язності і несуперечності застосовано логіку Хоара, метод резолюцій та програмне забезпечення Simplify;

- вперше запропоновано адаптивний алгоритм редукції бази правил із використанням методу дихотомії з динамічним кроком та інтегральної метрики, що враховує кількість активацій певного правила та ступінь належності об’єктів навчальної вибірки правилу. Множина правил поділяється на підмножину правил, що виконуються, та підмножину правил, що є надмірними для досягнення критерію якості;

- досліджено якість побудованої нечіткої продукційної моделі з використанням матриці невідповідностей та метрик, що визначають частину

правильних відповідей моделі (accuracy), точність (precision), повноту (recall), f-міру (f1-score);

- створено нові ефективні алгоритми розв’язання вищеназваних задач;
- створено програмний продукт на об’єктно-орієнтованих мовах програмування C++, JavaScript та Python, що реалізує розроблені алгоритми;
- методи і алгоритми застосовано до розв’язання модельних задач класифікації.

Ключові слова: штучний інтелект, нечітка логіка, відображення, класифікація, база знань, експертна система, мова C/C ++, мова JavaScript, Python, нечітка класифікація, квантильні оцінки, логіка Хоара, метод резолюцій, антецедент, консеквент, редукція, доведення знання, нечіткі параметри, система підтримки прийняття рішень, інформаційні технології, машинне навчання, нечітка модель, нечіткі множини, продукційна модель.

ABSTRACT

Yehoshkin D. The development of methods and algorithms for the automatic generation and research of the fuzzy knowledge base structure. – Qualifying scientific work on manuscript rights.

Dissertation for obtaining the degree of Doctor of Philosophy in the specialty 113 «Applied Mathematics» (11 - Mathematics and Statistics). – Oles Honchar Dnipro National University, Dnipro, 2023.

The dissertation is devoted to the development of methods and algorithms for the automatic generation of the knowledge base of a logical inference system for classification tasks based on experimental data, the improvement of logical inference methods, methods for tuning the parameters of the knowledge representation model, methods for proving the correctness of the knowledge base, and the research of the quality of the constructed logical inference system.

Currently, knowledge-based systems have been widely used in various applied fields of science and technology to solve the problems of classification, forecasting, image recognition, medical and technical diagnostics, management, automated design, and others.

Logical inference systems provide the automation of the decision-making process and can partially replace an expert in a particular field. In addition, they can transfer knowledge for the training of new experts. With the use of such systems, it is possible to create a description of models for poorly formalized objects and processes, and avoid the need to build a complex and expensive mathematical model. Most often, objects and processes are characterized by uncertainty, incompleteness of input data, a nonlinear nature of internal relationships and patterns, the presence of a significant number of parameters of various types, uncertainty of structure and its changeability depending on the situation.

The main component of logical inference systems is the knowledge base, which consists of a set of rules that are used for logical inference. The knowledge base rules describe the relationships in the subject domain, which can be identified based on experimental data obtained from observations of an object or process. The rules are represented in the form of crisp or fuzzy productions and must adequately describe the patterns in the data available for analysis.

However, the aforementioned characteristics of objects and processes lead to the need to formulate complex systems of rules that take into account the relationships between a large number of parameters. Also, a difficult stage in the construction of a logical inference system is the setting of the model.

There are several approaches to the formation of a knowledge base in logical inference systems. These include the analysis of knowledge by an expert (group of experts) based on experience, and automatic formation of a rule base using methods of intelligent data analysis and machine learning algorithms, or a combination of these two approaches. The knowledge generation process requires significant analytical work from the expert and significantly complicates the development of such systems. Therefore, the development of the second approach, automatic generation of a rule base and setting of the model parameters based on experimental data, is the subject of the dissertation research.

The development of methods and algorithms for automatic generation of knowledge bases significantly simplifies the procedures for creating and configuring logical inference systems, and also ensures their quality control. Since knowledge is a dynamic structure that can develop and change during operation, the work developed a mechanism for adding new knowledge to an existing knowledge base.

It should be noted that the quality of the functioning of an expert system depends on the completeness and consistency of the rules, as well as the size of the rule base. A large number of rules leads to both a decrease in the speed of logical inference, which is unacceptable for systems that operate in real time, and to the inconsistency of accumulated knowledge and the effect of overfitting. Therefore, the development of methods and algorithms that allow structuring the rule system, determining the order of their calls, determining the required number of rules, and controlling the consistency and completeness of the rule base is an important task that is being solved in the dissertation. The use of this approach will significantly simplify and accelerate the process of developing an expert system, the knowledge base of which is built during the learning process using objects of the training sample from the dataset.

The purpose of the work is to develop methods and algorithms for automatic generation of the knowledge base of a logical inference system for classification tasks based on experimental data. It also proposes methods to improve logical inference, setting the parameters of the knowledge representation model, prove the correctness of the knowledge base, and study the quality of the constructed logical inference system.

To represent knowledge about objects of the subject domain, the work proposes a production knowledge representation model that combines Mamdani and Takagi-Sugeno-Kang models and uses the membership function of the output object to a certain class as a weight coefficient in the consequent of the rule. This made it possible to determine the level of influence of each rule on the result of logical inference of the production system.

Correlation analysis is proposed to identify relationships between variables and hidden patterns or anomalies in data sets. Particular attention is paid to calculating the Pearson correlation coefficient and detecting multicollinearity as key aspects of the analysis of large amounts of training data. Additionally, the use of class scatter diagrams is proposed for visualizing the data distribution, which

will contribute to a deeper understanding of their essence and the identification of relationships.

The algorithm for automatic generation of a rule base based on a training set for solving a classification problem has been developed. The antecedent matrix is created using the features of objects from a finite set using the Cartesian product. The consequent vector is formed using the training set, which allows for the creation of a fuzzy rule base and the automatic setting of the model. The logical inference method has been improved in terms of conflict resolution when making a decision in case the system is unable to unambiguously determine the membership of an object to a certain class. To do this, a procedure for comparing the distance between objects and classes at the classification stage has been implemented. To assess the quality of the constructed fuzzy logical inference system and the obtained classification results, accuracy, precision, recall, f1-score metrics have been selected, and the confusion matrix is also used.

To carry out the rule base reduction procedure, an integral metric is proposed that takes into account the number of activations of a certain rule and the degree of membership of objects of the training set to the rule. Using the dynamic step dichotomy method and the proposed metric, an adaptive rule base reduction algorithm has been developed. The rule base obtained as a result of the reduction is significantly reduced, but this does not worsen the classification result.

For the parameters setting of a fuzzy production model, the automatic definition of term boundaries using the sample and midhinge values is proposed. This made it possible to reduce the impact of anomalous objects present in the training set on the classification result and make the term boundary definition procedure robust.

The correctness of the rule base was analyzed by proving its completeness, minimality, connectedness, and consistency using Hoare logic and the predicate of the weakest precondition $Q \Rightarrow WP(S, R)$. The proof was carried out using the resolution method and the Simplify software application, which provides

automation of the verification of the correctness of the rule base at the stage of system learning.

The proposed approach was applied to solving the problem of multiclass classification on the known model datasets Iris Data Set and species population of Arctic penguins. The setting of the parameters of the fuzzy production model was carried out. To define the term boundaries, the analysis of the features of the objects of the training set and the analysis of the frequency distribution for each object feature were performed, and the use of the midhinge made it possible to reduce the impact of anomalous objects present in the training set on the classification result.

The logical inference method has been improved in terms of conflict resolution when an object belongs to two classes at the same time. The proposed metric, which integrates the number of activations of a certain rule and the degree of membership of objects of the training set to the rule, has been used to reduce the rule base without loss of classification result quality.

A comparison of the classification results using the proposed approach and the classical fuzzy classification algorithm is presented.

The impact of the training set size on the classification results and the quality of the logical inference system was analyzed. It was found that increasing the size of the training set leads to improved classification results and the values of quality metrics.

The scientific novelty of the obtained results is as follows:

- the method of constructing a fuzzy production model for representing knowledge about objects of the subject domain based on a training set using developed learning algorithms has been further developed;
- a type of fuzzy production rule that combines Mamdani and Takagi-Sugeno-Kang models has been proposed. The rule consequent contains a weighting function that determines the degree of membership of the output variable to the resulting term set;

- the methodology for constructing a set of rules in the knowledge base has been improved, as well as the algorithm for their automatic formation using a preliminary analysis of the features of the training set, a matrix representation of antecedents, and a vector representation of the consequents of rules;

- the method of logical inference using the knowledge base of a fuzzy production model for solving the classification problem in the presence of fuzziness and incompleteness of information about objects of the subject domain has been further developed. In the presence of a conflict in decision-making, which consists in the inability of the system to unambiguously determine the membership of an object to a certain class, a procedure for comparing the distance between objects and classes is proposed;

- the method of identification and setting of the parameters of a fuzzy production model has been further developed, including automated determination of linguistic variables, term sets, term boundaries, parameters of membership functions, rule weights, and the number of rules;

- for the first time, Hoare logic, the resolution method, and the Simplify software were used to verify the correctness of an automatically constructed rule base for the criteria of completeness, minimality, connectedness, and consistency;

- for the first time, an adaptive rule base reduction algorithm has been proposed using the dichotomy method with a dynamic step and an integral metric that takes into account the number of activations of a certain rule and the degree of membership of the objects of the training set to the rule. The set of rules is divided into a subset of rules that are executed and a subset of rules that are redundant to achieve the quality criterion;

- the quality of the constructed fuzzy production model was investigated using the confusion matrix and accuracy, precision, recall, and f1-score metrics;

- new efficient algorithms for solving the above-mentioned problems have been created;

- a software product has been created by object-oriented programming languages C++, JavaScript, and Python that implements the developed algorithms;

– the methods and algorithms were applied to solving model classification tasks.

Keywords: artificial intelligence, fuzzy logic, mapping function, classification, knowledge base, expert system, C/C++, JavaScript, Python, fuzzy classification, quantiles, Hoare logic, resolution method, antecedent, consequent, reduction, proof of knowledge, fuzzy parameters, decision support system, information technology, machine learning, fuzzy model, fuzzy sets, production system.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Yehoshkin D., Guk N. Automatic construction of a fuzzy system with a matrix representation of rules and a correct knowledge base. *Eastern-European Journal of Enterprise Technologies*, 2022. 6(4(120)). Р. 14–22. doi: <https://doi.org/10.15587/1729-4061.2022.268908>. Режим доступу до ресурсу: <http://journals.uran.ua/eejet/article/view/268908> (Scopus).

2. Єгошкін Д.І., Гук Н.А. Адаптивний алгоритм редукції нечіткої бази правил на основі даних навчальної вибірки. *Збірник наукових праць «Питання прикладної математики і математичного моделювання»*. Дніпро, 2022. Вип. 22. С. 49-59. doi: <https://doi.org/10.15421/322205>. Режим доступу до ресурсу: <https://pm-mm.dp.ua/index.php/pmmm/article/view/337>.

3. Єгошкін Д.І., Гук Н.А. Налаштування та навчання нечіткої моделі для задачі класифікації. *Вісник Запорізького національного університету. Серія фізико-математичні науки.*, м. Запоріжжя: Видавничий дім «Гельветика», 2021. Вип. 1. С. 33-43. doi: <https://doi.org/10.26661/2413-6549-2021-1-04>. Режим доступу до ресурсу: <http://journalsofznu.zp.ua/index.php/comp-science/article/view/2286>.

4. Єгошкін Д.І., Гук Н.А., Сірик С.Ф. Алгоритм класифікації на базі нечіткої логіки з розширюваною кількістю виводів. *Збірник наукових праць «Питання прикладної математики і математичного моделювання»*, м. Дніпро, 2018. Вип. 18. С. 66-75. doi: <https://doi.org/10.15421/321807>. Режим доступу до ресурсу: <https://pm-mm.dp.ua/index.php/pmmm/article/view/224>.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

5. Єгошкін Д.І., Гук Н.А. Вплив розміру навчальної вибірки на якість класифікації нечіткої експертної системи. *Математичне та програмне забезпечення інтелектуальних систем (МПЗІС-2023): Матеріали XXI*

міжнародна науково-практична конференція до 105-річчя Дніпровського національного університету імені Олеся Гончара, 22-24 листопада 2023., м. Дніпро, 2023. С. 122-123. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2023/11/mpzis-2023.pdf#page=122>.

6. Єгошкін Д.І., Гук Н.А. Аналіз коректності нечіткої бази правил в системах логічного виведення. *Математика та інформаційні технології. Матеріали міжнародної наукової конференції, присвяченої 55-річчю факультету математики та інформатики, 28–30 вересня 2023 р.* – Чернівці: Чернівецький нац. ун-т, 2023. С. 193-194. Режим доступу до ресурсу: <https://fmi.chnu.edu.ua/media/ghufs0d5/materialy-mizhnorodnoi-naukovo-konferentsii-fmi55.pdf#page=193>.

7. Єгошкін Д.І., Гук Н.А. Адаптивна редукція нечітких правил відносно навчальної вибірки. *Математичне та програмне забезпечення інтелектуальних систем (MSSIS-2022): Матеріали XX ювілейної міжнародної науково-практичної конференції, 23-25 листопада 2022., м. Дніпро, 2022. С. 83-84.* Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2022/12/MPZIS-2022-1.pdf#page=84>.

8. Єгошкін Д.І., Гук Н.А. The advantages of using MISO and MIMO models in fuzzy expert systems. *Прикладна математика та інформаційні технології ПМ&ІТ 2022: Тези доп. міжнародної наукової конференції присвяченої 60-річчю кафедри прикладної математики та інформаційних технологій, 22 – 24 вересня 2022., м. Чернівці, 2022. С. 170-172.* Режим доступу до ресурсу: <http://www.amit60.fmi.org.ua/files/AMIT2022-Materials.pdf#page=171>.

9. Єгошкін Д.І., Гук Н.А. Перевірка коректності бази знань та адекватності нечіткої моделі. *Математичне та програмне забезпечення інтелектуальних систем (MSSIS-2021): Матеріали XIX міжнародної науково-практичної конференції, 17-19 листопада 2021., м. Дніпро, 2021. С. 75-77.* Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2021/12/mpzis-2021.pdf#page=75>.

10. Єгошкін Д.І., Гук Н.А. The application of fuzzy logic in classification tasks. *Сучасні інформаційні та комунікаційні технології на транспорті, в промисловості та освіті: Матеріали XIV міжнародної науково-практичної конференції, 15.12.2020 – 16.12.2020., м. Дніпро, 2020. С. 69-70.* Режим доступу до ресурсу: <https://crust.ust.edu.ua/server/api/core/bitstreams/cf8adc16-fd69-42b1-a82f-cbc77dbcfe2f/content#page=70>.

11. Єгошкін Д.І., Гук Н.А. The fuzzy logic classification algorithm with three dimensional membership functions and dynamic knowledge base. *Молодь у світі сучасних технологій за тематикою: Використання інформаційних та комунікаційних технологій в сучасному цифровому суспільстві: матеріали міжнар. наук.-практ. конф. (4-5 червня 2020 р., м. Херсон) / за заг. ред. Г.О. Райко., м. Херсон, 2020. С. 226-227.* Режим доступу до ресурсу: <http://kntu.net.ua/ukr/content/download/76712/444549/file/YPWMT-2020.pdf#page=277>.

12. Єгошкін Д.І., Гук Н.А. The classification algorithm on the basis of the fuzzy logic with dynamic knowledge-based system. *Сучасні науково-технічні дослідження у контексті мовного простору (іноземними мовами) 11-12 квітня 2019 року: матеріали VIII Регіональної науково-практичної конференції молодих учених та студентів., м. Дніпро, 2019. С. 83-84.* Режим доступу до ресурсу: https://www.dnu.dp.ua/docs/ndc/202/such_nauk_teh_dosl.pdf#page=83.

13. Єгошкін Д.І., Гук Н.А. Алгоритм класифікації на базі нечіткої логіки з динамічною базою знань. *Математичне та програмне забезпечення інтелектуальних систем (MPZIS-2018): Матеріали XVI міжнародна науково-практична конференція, 21-23 листопада 2018., м. Дніпро, 2018. С. 67-68.* Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2021/06/mpzis-2018.pdf#page=67>.

ЗМІСТ

ВСТУП	21
Розділ 1. АНАЛІЗ МОДЕЛЕЙ ПРЕДСТАВЛЕННЯ ЗНАНЬ, МЕТОДІВ ЛОГІЧНОГО ВИВЕДЕННЯ ТА НАЛАШТУВАННЯ ПАРАМЕТРІВ ПРИ ПОБУДОВІ ЕКСПЕРТНИХ СИСТЕМ. ВИБІР НАПРЯМКУ ДОСЛІДЖЕННЯ	30
1.1 Поняття бази знань експертної системи	30
1.2 Моделі представлення знань в експертних системах	33
1.3 Способи отримання знань в експертних системах	44
1.4 Редукція бази правил експертної системи	50
1.5 Основні алгоритми нечіткого виведення в експертних системах	56
1.6 Методи налаштування експертної системи	60
1.7 Методи аналізу коректності бази правил	62
1.8 Критерії оцінки коректності побудованих експертних систем	64
1.9 Методи нечіткої класифікації	65
1.10 Висновки до розділу	67
Розділ 2. МАТЕМАТИЧНА МОДЕЛЬ ПРОДУКЦІЙНОЇ НЕЧІТКОЇ ЕКСПЕРТНОЇ СИСТЕМИ, ПОБУДОВАНОЇ З ВИКОРИСТАННЯМ ЕКСПЕРИМЕНТАЛЬНИХ ДАНИХ, ТА МЕТОД ЛОГІЧНОГО ВИВЕДЕННЯ	70
2.1 Постановка задачі	70
2.2 Вибір параметрів об'єктів експериментальної вибірки та встановлення взаємозв'язків	72
2.3 Методика побудови бази правил експертної системи	75
2.4 Алгоритм автоматичного формування бази правил	77
2.5 Метод логічного виведення	78
2.6 Метод вирішення конфліктів при прийнятті рішень з урахуванням відстані між класами.	80
2.7 Метрики оцінки якості класифікації	84
2.8 Адаптивний алгоритм редукції бази правил	85
2.9 Висновки до розділу	87
Розділ 3. НАЛАШТУВАННЯ ПАРАМЕТРІВ НЕЧІТКОЇ ПРОДУКЦІЙНОЇ МОДЕЛІ	90
3.1 Попередня обробка експериментальних даних для побудови експертної системи	91

3.1.1	Застосування точкової діаграми розсіювання для попереднього аналізу даних	91
3.1.2	Налаштування параметрів функцій належності	93
3.1.3	Налаштування границь термів для функції належності за допомогою вибіркового середнього значення	94
3.1.4	Налаштування границь термів для функції належності за допомогою міжквартильного середнього	95
3.2	Аналіз коректності бази правил експертної системи	100
3.3	Програмна реалізація розробленого підходу доведення повноти бази правил	104
3.4	Аналіз якості роботи експертної системи	109
3.5	Висновки до розділу	110
Розділ 4. ЗАСТОСУВАННЯ ЕКСПЕРТНОЇ СИСТЕМИ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ КЛАСИФІКАЦІЇ		112
4.1	Налаштування та навчання нечіткої моделі для задачі класифікації	112
4.1.1	Налаштування нечіткої моделі для класифікації Ірисів Фішера	112
4.1.2	Налаштування нечіткої моделі для класифікації видової популяції арктичних пінгвінів	116
4.2	Приклади та аналіз результатів	120
4.2.1	Класифікація Ірисів Фішера	120
4.2.2	Редукція бази правил для задачі класифікації Ірисів Фішера	128
4.2.3	Класифікація видової популяції арктичних пінгвінів	133
4.3	Висновки до розділу	138
ВИСНОВКИ		140
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ		142
ДОДАТОК А. АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ РОБОТИ		161
ДОДАТОК Б. ВИБІР МОВИ ПРОГРАМУВАННЯ		164
ДОДАТОК В. СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА		169

ВСТУП

Актуальність теми.

В останні роки штучний інтелект привертає дедалі більше уваги та стає одним із ключових напрямків у дослідженнях. Однією з перших успішних областей застосування штучного інтелекту, що використовується у різних сферах людської діяльності, є системи, що базуються на знаннях. Термін "система, що заснована на знаннях" означає інформаційну систему, в якій використовується представлення знань та логічне виведення, наближене до людського способу міркування. Експертні системи є одним із найбільш успішних прикладів систем, які здатні пропонувати рішення та давати поради в певних галузях знань на рівні експертів.

В даний час системи, що засновані на знаннях, набули широке застосування в різних прикладних областях науки і техніки для розв'язання задач класифікації, прогнозування, розпізнавання образів, медичної та технічної діагностики, управління, автоматизованого проектування та інших.

Системи логічного виведення дозволяють автоматизувати процес прийняття рішень і в такий спосіб частково замінити фахівця-експерта в певній галузі, крім того, вони можуть передавати знання для навчання нових фахівців. Із застосуванням таких систем можна створювати опис моделей для погано формалізованих об'єктів і процесів та уникнути необхідності побудови складної і кошовної математичної моделі. Найчастіше об'єкти та процеси характеризуються невизначеністю, неповнотою вхідних даних, нелінійним характером внутрішніх зв'язків та закономірностей, наявністю значної кількості параметрів різних типів, невизначеністю структури та її змінюваністю в залежності від ситуації.

Основним компонентом систем логічного виведення є база знань, що складається з набору правил, за якими здійснюється логічне виведення. Правила бази знань описують взаємозв'язки в предметній області, які можна виявити на основі експериментальних даних, отриманих в результаті

спостережень за об'єктом або процесом. Правила зображуються у формі чітких або нечітких продукцій та повинні адекватно описувати закономірності в наявних для аналізу даних.

Однак, зазначені особливості об'єктів та процесів призводять до необхідності формулювати надскладні системи правил, що мають враховувати взаємозв'язки між значною кількістю параметрів. Також складним етапом побудови системи логічного виведення є налаштування моделі.

Для формування бази знань у системах логічного виведення існує кілька підходів, до яких належить аналіз знань експертом (групою експертів) на основі досвіду, автоматичне формування бази правил з використанням методів інтелектуального аналізу даних і алгоритмів машинного навчання, або поєднання зазначених підходів. Процес генерування знань потребує значної аналітичної роботи з боку експерта та суттєво ускладнює розробку таких систем, тому саме розвитку другого підходу – автоматичної генерації бази правил та налаштуванню параметрів моделі на основі експериментальних даних – присвячено тему дисертаційного дослідження.

Розробка методів та алгоритмів автоматичної генерації баз знань суттєво спрощує процедури створення та налаштування систем логічного виведення, а також забезпечує контроль їх якості. Оскільки знання є динамічною структурою, що може розвиватися та змінюватися у процесі функціонування, у роботі розроблено механізм додавання нових знань до вже існуючої бази знань. Слід зазначити, що якість функціонування експертної системи залежить від повноти та несуперечності правил, а також розміру бази правил. Велика кількість правил призводить як до зниження швидкості виконання логічного виведення, що неприпустимо для систем, які працюють у режимі реального часу, так і до суперечливості накопичених знань та ефекту перенавчання.

Аналіз літературних джерел за темою дисертаційної роботи свідчить про те, що багато аспектів побудови експертних систем залишаються

недостатньо дослідженими. Відомі моделі і алгоритми, які призначені для автоматичної генерації бази знань, при практичному використанні виявляються неефективними, вимагають значних обчислювальних ресурсів та стають складними для застосування.

Тому тема дисертаційної роботи, спрямована на розробку методів і алгоритмів, що дозволяють структурувати систему правил, визначати порядок їх викликів, визначати необхідну кількість правил, контролювати несуперечність і повноту бази правил є актуальною задачею, що вирішується у дисертаційній роботі. Використання такого підходу значно спростить і прискорить процес розробки системи, база знань якої будується в процесі навчання за допомогою об'єктів навчальної вибірки. Досягнення зазначеної мети сприятиме підвищенню ефективності створення експертних систем та забезпечить отримання адекватних і точних розв'язків задач класифікації.

Зв'язок роботи з науковими програмами, планами, темами.

Дисертаційні дослідження здійснювалися у відповідності до планів наукових досліджень кафедри комп'ютерних технологій Дніпровського національного університету імені Олеся Гончара в рамках наукових тем «Дослідження математичних моделей фізичних процесів методами ідентифікації та рекурентного аналізу із застосуванням інформаційних технологій» (№ держреєстрації 0119U101053, 2019-2021 р.р.), «Детерміновані та стохастичні алгоритми комп'ютерного моделювання об'єктів та процесів різної природи» (№ держреєстрації 0122U001467, 2022-2024 р.р.) при кафедрі комп'ютерних технологій у відповідності до тематичних планів науково-дослідних робіт Дніпровського національного університету імені Олеся Гончара.

Мета і задачі дослідження.

Метою дисертаційної роботи є розробка методів та алгоритмів автоматичної генерації бази знань системи логічного виведення для задач

класифікації на основі експериментальних даних, вдосконалення методів логічного виведення, методів налаштування параметрів моделі представлення знань, методів доведення коректності бази знань та дослідження якості побудованої системи логічного виведення.

Досягнення зазначеної мети сприятиме підвищенню ефективності створення експертних систем та забезпечить отримання адекватних і точних розв'язків задач класифікації.

Для досягнення зазначеної мети було поставлено такі завдання:

- створити нечітку продукційну модель, що об'єднує моделі Мамдані та Такагі-Сугено-Канг, для представлення знань про об'єкти предметної області з використанням навчальної вибірки та алгоритмів навчання;
- обрати спосіб представлення правил, розробити методику побудови сукупності правил бази знань та алгоритм їх автоматичного формування;
- розробити алгоритми ідентифікації значень параметрів нечіткої продукційної моделі;
- розробити підхід для перевірки коректності побудованої бази правил за критеріями повноти, мінімальності, зв'язності і несуперечності;
- вдосконалити метод логічного виведення із урахуванням того, що інформація про об'єкти предметної області може бути нечіткою та неповною;
- обрати метрики для оцінки якості автоматично побудованої нечіткої бази правил;
- розробити на основі побудованих алгоритмів програмне забезпечення для формування систем нечітких правил та реалізації механізму нечіткого логічного виведення;
- застосувати розроблені методи та алгоритми до розв'язання модельних задач.

Об'єктом дослідження є нечіткі бази знань експертних систем.

Предметом дослідження є моделі, методи та алгоритми автоматичного формування бази знань системи логічного виведення для

розв'язання задачі класифікації з використанням експериментальної інформації.

Методи дослідження: для розв'язання зазначених завдань застосовано методи математичного моделювання, дискретної математики, теорію нечітких множин та нечіткої логіки, логіку Хоара та метод резолюцій, методи кореляційного аналізу, об'єктно-орієнтованого програмування.

Достовірність отриманих результатів забезпечується використанням добре апробованих моделей представлення знань, коректністю математичних постановок задач, використанням методів та алгоритмів, що є теоретично обґрунтованими та не суперечливими з відомими положеннями інших авторів, контрольованою точністю обчислень, доброю узгодженістю між собою теоретичних і експериментальних результатів, несуперечністю отриманих результатів відповідним опублікованим результатам інших авторів.

Наукова новизна одержаних результатів полягає у наступному:

- дістав подальшого розвитку метод побудови нечіткої продукційної моделі для представлення знань про об'єкти предметної області на основі навчальної вибірки за допомогою розроблених алгоритмів навчання;
- запропоновано вид нечіткого продукційного правила, що поєднує моделі Мамдані та Такагі-Сугено-Канг та містить у консеквенті правила в якості вагового коефіцієнту функцію, що визначає ступінь належності вихідної змінної до результуючої терм-множини;
- удосконалено методику побудови сукупності правил бази знань та алгоритм їх автоматичного формування за допомогою попереднього аналізу ознак навчальної вибірки, матричного представлення антецедентів та векторного представлення консеквентів правил;
- дістав подальшого розвитку метод логічного виведення з використанням бази знань нечіткої продукційної моделі для розв'язання задачі класифікації в умовах нечіткості та неповноти інформації про об'єкти предметної області; за наявності конфлікту при прийнятті рішення, що

полягає у неспроможності системи однозначно визначити належність об'єкта певному класу, запропоновано процедуру порівняння відстані між об'єктами та класами;

- дістав подальшого розвитку метод ідентифікації та налаштування значень параметрів нечіткої продукційної моделі, зокрема автоматизовано визначення лінгвістичних змінних, терм-множин, границь термів, параметрів функцій належності, вагових коефіцієнтів правил, кількості правил;

- вперше для перевірки коректності автоматично побудованої бази правил за критеріями повноти, мінімальності, зв'язності і несуперечності застосовано логіку Хоара, метод резолюцій та програмне забезпечення Simplify;

- вперше запропоновано адаптивний алгоритм редукції бази правил із використанням методу дихотомії з динамічним кроком та інтегральної метрики, що враховує кількість активацій певного правила та ступінь належності об'єктів навчальної вибірки правилу. Множина правил поділяється на підмножину правил, що виконуються, та підмножину правил, що є надмірними для досягнення критерію якості;

- досліджено якість побудованої нечіткої продукційної моделі з використанням матриці невідповідностей та метрик, що визначають частину правильних відповідей моделі (ассурасу), точність (precision), повноту (recall), f-міру (f1-score);

- створено нові ефективні алгоритми розв'язання вищеназваних задач;
- створено програмний продукт на об'єктно-орієнтованих мовах програмування C++, JavaScript та Python, що реалізує розроблені алгоритми;
- методи і алгоритми застосовано до розв'язання модельних задач класифікації.

Практичне значення одержаних результатів. В дисертаційній роботі досліджено застосування нечітких продукційних моделей для розв'язання задач класифікації в умовах нечіткості та неповноти інформації про об'єкти предметної області .

Розроблений підхід та відповідне програмне забезпечення для автоматизації побудови системи логічного виведення, що базується на нечіткої продукційної моделі представлення знань, вдосконалює алгоритми логічного виведення та налаштування параметрів моделі задля отримання адекватних і точних розв'язків задач класифікації, сприяє покращенню точності та ефективності процесів обробки та аналізу даних.

Розроблені в роботі методи й алгоритми можуть бути застосовані до розв'язання прикладних задач із різних предметних областей, що зводяться в математичній постановці до задач класифікації. Результати цього дослідження можуть значно підвищити ефективність та застосовність систем логічного виведення у різних галузях, забезпечуючи більш точні та надійні результати класифікації на основі експериментальних даних та досвіду. Це має важливе значення для розвитку сучасних технологій та розв'язання складних задач у науці, промисловості, медицині та інших галузях людської діяльності.

Окремі теоретичні результати дослідження включено до змісту дисципліни «Методи Machine Learning», що викладається на кафедрі комп'ютерних технологій факультету прикладної математики Дніпровського національного університету імені Олеся Гончара для студентів магістратури за спеціальністю 113 «Прикладна математика», освітня програма «Інформатика». Результати дисертації також використовуються при виконанні курсових та дипломних робіт студентами, які навчаються за спеціальністю 113 «Прикладна математика».

Особистий внесок здобувача. Результати дисертаційної роботи відображено в 13 наукових працях. Усі результати дисертаційної роботи, що виносяться на захист, отримані автором особисто. У працях, що опубліковані у співавторстві, здобувачеві належить:

[17, 20, 134] – запропоновано вид нечітких продукційних правил на основі моделей Мамдані та Такагі-Сугено-Канг для формалізації чітких та нечітких залежностей в даних, що підлягають аналізу, застосовано логіку

Хоара, метод резолюцій та програмне забезпечення Simplify для перевірки коректності автоматично побудованої бази правил експертної системи за критеріями повноти, мінімальності, зв'язності і несуперечності; [13, 14] – із застосуванням методу дихотомії з автоматичним кроком розроблено адаптивний алгоритм редукції бази правил, що поділяє множину правил на підмножину правил, які застосовуються під час логічного виведення, та підмножину правил, що є надмірними для досягнення критерію якості; [10, 19] – запропоновано алгоритм визначення границь термів з використанням міжквартильного середнього, розроблено модифікацію класичного алгоритму нечіткої класифікації шляхом виконання двох етапів – проміжної та остаточної класифікації у разі, коли класи розташовані поблизу один від одного; [11, 16] – удосконалено метод логічного виведення шляхом додавання блоку порівняння вихідного значення для нового об'єкту з вихідними значеннями для наявних у базі знань об'єктів; [18] - досліджено вплив розміру навчальної вибірки на якість класифікації нечіткої системи логічного виведення; [9] - здійснено аналіз застосування типів моделей MISO та МІМО в нечітких експертних системах; [12] - розглянуто введення тривимірних функцій належності для зменшення недетермінізму при розв'язанні задач класифікації; [15] - запропоновано алгоритм класифікації з динамічною базою знань.

Апробація результатів дисертації. Результати дисертаційної роботи доповідались і обговорювались на семінарі «Сучасні питання оптимізації та дискретної математики» при Науковій раді НАН України з проблеми «Кібернетика», який функціонує при Дніпровському національному університеті імені Олеся Гончара; на міжнародних конференціях «Математичне та програмне забезпечення інтелектуальних систем» (м. Дніпро, 2018, 2019, 2020, 2021, 2022, 2023 рр.); «Сучасні науково-технічні дослідження у контексті мовного простору (іноземними мовами)» (м. Дніпро, 2019 р.); «Використання інформаційних та комунікаційних технологій в сучасному цифровому суспільстві» (м. Херсон 2020 р.);

«Сучасні інформаційні та комунікаційні технології на транспорті, в промисловості та освіті» (м. Дніпро, 2020 р.); «Прикладна математика та інформаційні технології ПМ&ІТ» (м. Чернівці, 2022, 2023 рр.).

Публікації. Основні результати дисертаційної роботи опубліковано в 13 наукових працях: одна стаття [134] у виданні, що індексується науково-метричною базою Scopus; 3 статті ([14, 16, 19]) у наукових фахових виданнях України категорії Б з фізико-математичних наук, 9 тез доповідей у збірниках матеріалів міжнародних наукових конференцій ([9-13, 15, 17, 18, 20]).

Структура та обсяг дисертації. Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, переліку використаних джерел, що містить 137 найменувань на 19 сторінках та додатків на 9 сторінках. Загальний обсяг дисертації – 171 сторінки, обсяг основного тексту – 121 сторінка. Робота містить 29 рисунків та 17 таблиць.

Подяки. Автор висловлює щиру подяку науковому керівнику, доктору фізико-математичних наук, професору Гук Наталії Анатоліївні за постійну увагу, доброзичливе ставлення та допомогу в роботі.

Розділ 1. АНАЛІЗ МОДЕЛЕЙ ПРЕДСТАВЛЕННЯ ЗНАНЬ, МЕТОДІВ ЛОГІЧНОГО ВИВЕДЕННЯ ТА НАЛАШТУВАННЯ ПАРАМЕТРІВ ПРИ ПОБУДОВІ ЕКСПЕРТНИХ СИСТЕМ. ВИБІР НАПРЯМКУ ДОСЛІДЖЕННЯ

1.1 Поняття бази знань експертної системи

Для представлення досвіду експерта та знань з певної предметної галузі в експертних системах (ЕС) використовується база знань. Ця база застосовується для керування знаннями та метаданими, зокрема для виконання таких операцій над знаннями як зберігання та відновлення (Backup and Restore), пошук (Search), вилучення даних (Data Querying), додавання (Data Insertion), редагування (Data Update), видалення (Data Deletion), індексування даних (Data Indexing), виконання транзакцій (Transactions), вирішення колізій (Collision Detection).

В базі знань зберігаються правила логічного виведення, що отримані з досвіду експерта, або із систем з самонавчанням. База знань може зберігати інформацію, що є результатом попереднього розв'язання задач. Область штучного інтелекту, що пов'язана з вивченням баз даних, баз знань та методів роботи із знаннями, називається інженерією знань. Ця галузь штучного інтелекту займається вивченням та розробкою методів для аналізу, представлення, зберігання і використання знань. Вона зосереджується на створенні комп'ютерних систем, що можуть засвоювати, розуміти і застосовувати знання в різних областях, таких як медицина, фінанси, експертні системи, ігри, розпізнавання образів тощо. Інженерія знань також досліджує методи і засоби для автоматичного виявлення патернів, структуризації даних та використання знань для прийняття рішень.

Інженерія знань була винайдена Едвард Фейгенбаумом та МакКордаком як [72]: «Інженерія знань – розділ (дисципліна) інженерії, направлений на впровадження знань в комп'ютерні системи для вирішення комплексів завдань, що зазвичай вимагають участі людського досвіду.»

Інженерія знань та база знань – є ключовими поняттями при створенні експертних систем. Розглянемо стандартну архітектуру компонентів експертної системи нечіткого виведення [93]:

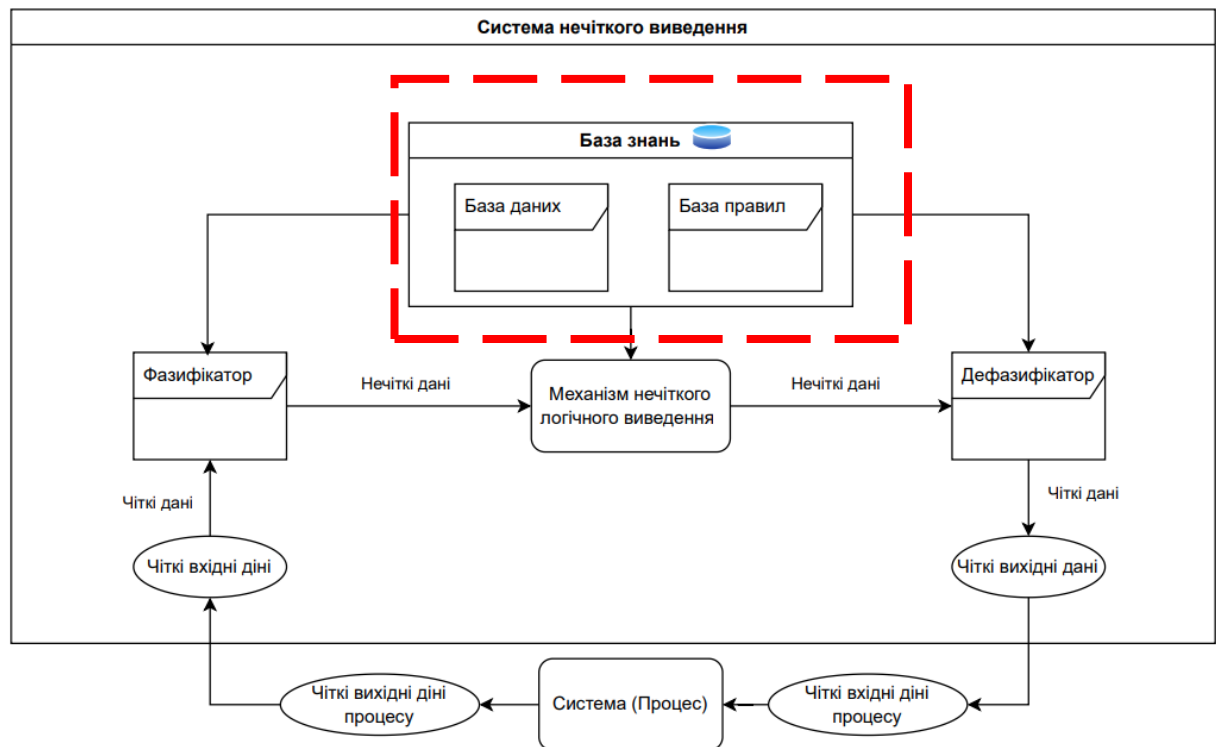


Рис. 1.1. Стандартна архітектура компонентів системи нечіткого виведення

На рис 1.1. пунктиром наведено розташування бази знань в архітектурі систем нечіткого виведення. Можна побачити, що три частини системи пов'язані з базою знань. Сама ж база знань у класичному випадку складається з двох частин – бази даних та бази правил.

База знань містить як декларативні (фактичні), так і процедурні знання (процедурні правила для прийняття рішень) з певної предметної області.

База даних формується в процесі роботи ЕС, описує контекст проблемної області і, як правило, вважається набором корисних фактів та даних. Це факти, що задовольняють активній частині логічної імплікації правила – антецеденту.

База правил є основною структурою для представлення знань в системах логічного виведення, що формулює зв'язки між вхідними змінними та відповідними вихідними діями чи висновками. Вона складається з набору

продукційних правил «ЯКЩО-ТО», кожне з яких інкапсулює певну частину експертних знань. Правила формулюються в формі логічних тверджень, що робить їх зрозумілими як для експертів, так і для системи штучного інтелекту. Це дозволяє експертній системі міркувати та ухвалювати рішення на основі вхідних даних і правил. Кожне правило складається з частини «ЯКЩО» (антецедент) і частини «ТО» (консеквент).

Таким чином, база знань зберігає довгострокові дані, що описують певну предметну область, та набори правил для прийняття рішень на основі досвіду експерта в даній області.

Розглянемо більш детальну структуру бази знань:

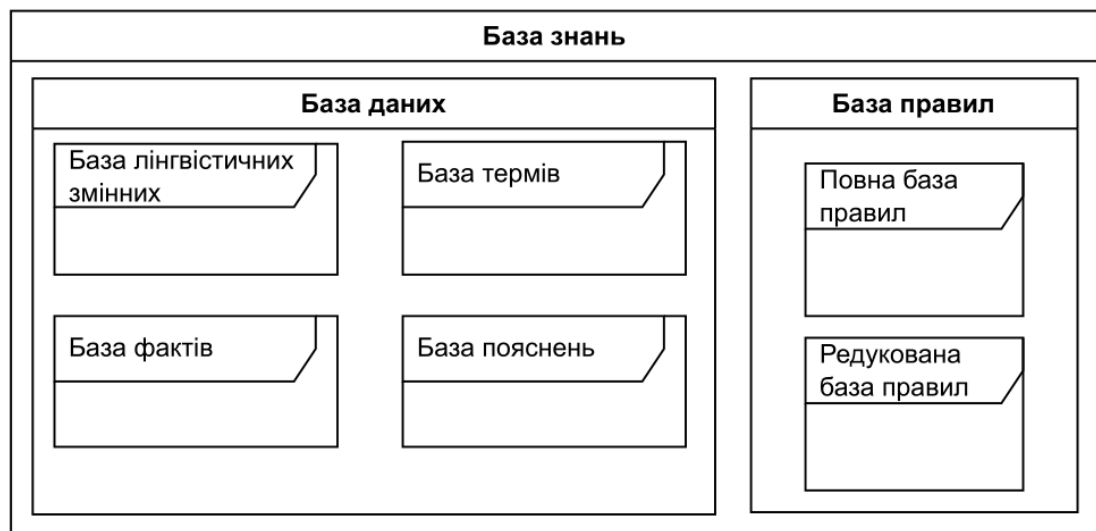


Рис. 1.2. Структура компонентів бази знань

База даних може зберігати в собі:

- базу лінгвістичних змінних – назви змінних, тип даних лінгвістичної змінної, універсум – границі, діапазон, терм множини тощо;
- базу термів для лінгвістичних змінних, зокрема, назви термів, границі термів, функції належності;
- базу фактів з певної предметної області, в який застосовується база знань, може зберігати пояснення щодо використання лінгвістичних змінних та терм множини;

- базу пояснень для опису результатів, що отримано у процесі прийняття рішення, та надання висновків у зрозумілій для користувача формі;
- інші компоненти.

База правил зберігає правила, за якими здійснюється процес міркування та прийняття рішень. Вона бути представлена у вигляді повної бази правил, що має в собі всі можливі правила, яким змогла навчитися система, або у вигляді редукованої бази правил. Редукована база правил є спрощеною та може містити меншу кількість правил, але така база правил має бути ефективною за Парето, тобто забезпечувати оптимальне співвідношення між критеріями складності (кількістю термів для опису нечітких змінних, кількістю правил) і точністю моделі. Таким чином, в середині бази правил може зберігатися декілька редукованих баз правил кожна зі своїм рівнем точності моделі прийняття рішень. Застосування редукованої бази правил в процесі експлуатації експертної системи дозволить скоротити обсяг пам'яті, що необхідно для виконання процедури виведення, і час на отримання логічного висновку.

База знань може містити і інші додаткові компоненти, це залежить від реалізації та архітектури експертної системи.

1.2 Моделі представлення знань в експертних системах

У розвиток теорії експертних систем великий внесок зробили дослідники та фахівці з усього світу як вітчизняні, так і зарубіжні вчені у цій галузі знань – Башликов А. А., Венделін А. Г., Геловані В. А., Євланов Л. Г., Заде Л. А., Ларічев О. І., Любарський Ю. Я., Лорьер Ж. Л., Люггер Дж. Ф., Поспелов Г. С., Поспелов Д. А., Саати Т., Самойлов В. Д., Таунсенд К., Терелянський П. В., Тиугу Е. Х., Ульман Дж., Уотермен Д., Фішберн П. К., Форсайт Ф., Чавчанідзе В. В. та багато інших. Зокрема, значний внесок в розвиток теорії прийняття рішень, розробки та застосування нечітких

експертних систем для розв'язання прикладних задач внесли Заде Л. А., Башмаков І. А., Берштейн Л. С., Борисов А. Н., Гарт Л.Л., Ісідзука М., Карелін В. П., Кісельова О.М., Коровін С. Я., Притоманова О.М., Ротштейн А.П., Турчина В.А., Штовба С. Д., Яковлев С.В.

В процесі побудови експертної системи виникає питання, як саме зберігати знання експерта, та у якому вигляді. Вивчення того, як люди зберігають та обробляють інформацію, дозволяє створювати більш реалістичні та ефективні експертні системи, які можуть імітувати людську здатність приймати рішення на основі знань та досвіду. Розуміння цих процесів сприяє розвитку та вдосконаленню експертних систем, здатних успішно вирішувати складні завдання та підтримувати користувача у процесі прийнятті рішень.

Термін “представлення знань” відноситься до галузі штучного інтелекту та містить відомості про відображення та зберігання інформації про всесвіт у формі, що є зрозумілою для комп'ютерних систем. Основною метою представлення знань є формалізація способу подання знань та залежностей між ними задля розуміння їх комп'ютерними системами та подальшого маніпулювання ними для розв'язання задач теорії прийняття рішень, мовленнєвого аналізу, діагностики, планування тощо. Представлення знань передбачає використання різних форм їх зображення, зокрема, семантичних мереж, графів, логічних висловлювань, фреймів, онтологій та інших структур, які дозволяють описати факти предметної області та зв'язки між ними [118].

Знання в експертних системах представляють собою опис відомостей про властивості об'єктів (процесів) предметної області та про взаємозв'язки з іншими об'єктами (процесами), а також правила, за якими можна отримувати нові знання. Проте самі по собі знання, навіть ті, які є перевіреними практикою та досвідом, насправді малопродуктивні, оскільки породжують новий тип хаотичної інформації, яку складно формалізувати. У зв'язку з цим важливо формалізувати знання, що утворюють базу знань експертної системи

шляхом зображення їх за допомогою формальних моделей. Організація бази знань спрямована на те, щоб якомога ширше використовувати семантику і специфіку предметної області з метою підвищення ефективності роботи всієї експертної системи. [6]

В основу знань експертної системи покладено інформацію у вигляді фактів, які є твердженнями про поточний стан предметної області, аксіом, що описують загальні закони, і правил логічного висновку.

Початковою точкою в побудові моделей для представлення знань є гіпотеза про знання, яку вперше формалізовано Брайаном С. Смітом у 1985 році [55, 56]. Відповідно до гіпотези Брайана С. Сміта: «будь-який інтелектуальний процес, який має механічне втілення, складається зі структурних компонентів. Ці компоненти є природними для зовнішніх спостерігачів (експертів) засобами представлення пропозиційних оцінок знань, що описують поведінку процесу, та незалежно від зовнішніх семантичних атрибутів відіграють формальну, причинну та суттєву роль у породженні поведінки, що відображає ці знання».

Для ефективного функціонування експертних систем необхідно обрати відповідний спосіб представлення знань та адаптувати його до вимог конкретної задачі. У сучасній літературі застосовують такі моделі представлення знань [38]: логічні моделі, семантичні мережі, фреймові моделі, сценарії, продукційні моделі, дерева рішень, асоціативні правила. Розглянемо переваги та недоліки зазначених моделей та підходи, за якими здійснюється вибір моделі для певних типів задач.

Дерева рішень. Дерева рішень розглянуто в багатьох наукових роботах [32, 54, 74, 106, 121]. Основна ідея дерев рішень полягає в тому, що вони використовують структуру дерева для подання різних варіантів вибору та послідовного прийняття рішень. Кожен вузол дерева представляє рішення або тест на певну характеристику даних. Результат виконання тесту визначає, який шлях в дереві буде продовжений в залежності від значення цієї характеристики. Кінцеві листя дерева представляють рішення або прогнози. З

дерев рішень можна автоматично виводити правила, які засновані на структурі дерева, що дозволяє легше зрозуміти, яким чином було прийнято певне рішення. Проте дерева рішень мають схильність до перенавчання, у деяких випадках дерева рішень можуть стати досить великими, що призведе до збільшення часу його обходу для прийняття рішення. Дерев рішень зазвичай не можуть ефективно працювати з нечіткими або неявними знаннями, оскільки вони засновані на чітких правилах та тестах.

Фреймові моделі. У роботах [30, 69, 97, 120] розглядаються Фреймові моделі представлення знань. Це ієрархічна структура з атрибутами та слотами, що ідеально підходить для опису комплексних знань про сутності предметної області. Основною перевагою застосування фреймів як моделі подання знань є те, що вони відображають концептуальну основу організації пам'яті людини, а також дозволяють відтворити її гнучкість і наочність. Подання знань у вигляді фреймової моделі є найбільш ефективним для структурного опису складних понять і розв'язання задач, у яких відповідно до ситуації бажано застосовувати різні способи логічного виведення. Водночас у разі застосування фреймів ускладнюється керування завершеністю і сталістю цілісного образу, що є недоліком зазначеної моделі представлення знань. Зокрема, з цієї причини існує велика небезпека порушення приєднаної процедури. Слід зазначити, що фреймову систему без механізму приєднаних процедур часто використовують як базу даних системи продукцій [38].

Семантичні мережі. В роботах [6, 27, 100, 112] розглядаються семантичні мережі як моделі представлення знань. У семантичних мережах знання подаються у вигляді вузлів (або концептів) та зв'язків між ними., кожен вузол представляє поняття або об'єкт з області знань, а зв'язки описують взаємозв'язки та взаємодію між цими поняттями. Така структура дозволяє моделювати знання у вигляді семантичної мережі, що зближує розуміння та обробку інформації комп'ютером зі способом, у який це робить мозок людини.

Такий підхід до представлення знань дозволяє зробити системи більш гнучкими, здатними до самостійного навчання та адаптації до змін у предметної області, що робить їх більш ефективними в різних ситуаціях. Однак, побудова та підтримка семантичних мереж може вимагати значних зусиль та експертного досвіду для правильного моделювання знань і забезпечення точності та коректності їх опису. Тому дослідники продовжують працювати над удосконаленням методів побудови та використання семантичних мереж для розв'язання різноманітних задач у галузі штучного інтелекту та інших дисциплінах.

Лінгвістичні та логіко-лінгвістичні моделі представлені у роботах [5, 34, 39]. Цей підхід базується на використанні природної мови для формалізації та представлення знань. Лінгвістичні моделі дозволяють використовувати мовні структури, такі як слова, фрази, речення та правила граматики для опису знань та зв'язків між ними. Такий підхід є зрозумілим для людей, оскільки використовує природну мову, але його формалізація та переведення в комп'ютерний код можуть бути складними і трудомісткими завданнями. Відтак, виникає ризик недостатньої точності або проблем з інтерпретацією знань комп'ютерною системою. Крім того, лінгвістичні моделі можуть мати обмежені можливості для вміщення великих обсягів знань через те, що вони можуть стати неефективними та складними при обробці значного обсягу інформації.

Серед сучасних підходів до моделювання знань можна виділити об'єктно-орієнтовану, UML, квантову та SQL моделі представлення знань, нечіткі когнітивні карти.

Квантова модель. З розвитком квантових комп'ютерів та квантових обчислень дослідники звертають увагу на нові моделі представлення знань, враховуючи властивості та можливості зазначених засобів обчислень, зокрема кодування інформації за допомогою кубітів [28, 51]. В роботі пропонується квантова модель кодування символів багатозначного алфавіту

у вигляді кубіту стану замість класичного бінарного кодування, де кожен символ представлений одиничним бітом (0 або 1).

Квантове кодування багатозначного алфавіту ґрунтується на використанні суперпозиції станів кубітів, що дозволяє представити кілька символів одночасно. Наприклад, для кодування символів із 4-х елементів алфавіту A, B, C та D, використовується стан кубіту, який є лінійною комбінацією станів $|A\rangle$, $|B\rangle$, $|C\rangle$ та $|D\rangle$. Таким чином, кубіт може представляти не одне значення, а відразу кілька, що суттєво збільшує ємність та паралельність кодування інформації.

Основною перевагою квантової моделі кодування символів багатозначного алфавіту є можливість ефективного зберігання та обробки значних обсягів інформації. Квантові системи здатні обробляти складні взаємозв'язки та структури даних з великою швидкістю та із застосуванням паралелізму, що робить їх ідеальним вибором для побудови потужних експертних систем. Однак, слід зазначити, що квантові технології все ще знаходяться у стадії розвитку, і їх застосування в експертних системах потребує подальших досліджень.

Об'єктно-орієнтована модель. Об'єктно-орієнтовану модель представлення знань почали використовувати з розвитком об'єктно-орієнтованих мов програмування. Так в роботах [48, 98, 103, 115] застосовано об'єктно-орієнтовану модель представлення знань. Ця методологія організації та структурування інформації заснована на концепціях об'єктів та класів з об'єктно-орієнтованого програмування. Із застосуванням цієї моделі знання представляються у вигляді об'єктів, які мають характеристики (властивості) і поведінку (методи), а також об'єднуються в класи, що мають узагальнюючі характеристики та визначають відношення між об'єктами.

Модель дозволяє організувати знання у вигляді об'єктів та класів, що робить їх структурованими та легко зрозумілими. Об'єкти та класи дозволяють висловити знання у більш інтуїтивній та природній формі,

близької до реального світу. Спадкування та створення нових класів на основі існуючих дозволяє ефективно перевикористовувати знання та код. Знання можуть бути організовані у вигляді окремих модулів (класів), що спрощує розробку, тестування та їх супровід. Додавання нових об'єктів та класів не потребує зміни всієї системи, що забезпечує її розширюваність та адаптивність. За допомогою абстракцій модель дозволяє виділяти загальні характеристики та ігнорувати деталі, що спрощує аналіз та розуміння складних концепцій.

Недоліками об'єктно-орієнтованої моделі є складність створення адекватних об'єктних моделей для складних і абстрактних задач. При застосуванні такої моделі спостерігається перевантаження абстракціями, у деяких випадках надмірна абстракція може ускладнити розуміння знань та унеможливити їх застосування.

UML. UML (Unified Modeling Language) — це добре відомий стандарт для моделювання систем програмного забезпечення та інших складних систем. Хоча UML в основному зосереджений на розробці програмного забезпечення, його можна адаптувати для представлення різних аспектів знань в експертних системах. Так у роботах [76, 104, 109] використовуються UML модель представлення знань. Діаграми класів в UML можна використовувати для представлення знань у термінах класів, об'єктів та їхніх зв'язків. Кожен клас представляє концепцію або сутність у домені, а атрибути та методи класу представляють знання, пов'язані з цією концепцією. Відношення між класами можуть відображати різні асоціації, агрегації або залежності, здійснюють допомогу при моделюванні структури знань у системі.

Use Case діаграми ілюструють взаємодію між користувачами (експертами домену) та експертною системою. Вони можуть представляти різні сценарії або випадки використання, в яких експертна система застосовує свої знання для вирішення певних проблем. Діаграми варіантів використання допомагають зрозуміти, як знання використовуються в

реальних ситуаціях. Діаграми послідовностей корисні для ілюстрації взаємодії та зв'язку між різними компонентами експертної системи під час міркування та процесу прийняття рішень. Вони можуть представляти, як знання передаються між модулями або компонентами експертної системи. Діаграми компонентів можна використовувати для представлення різних модулів або компонентів експертної системи та того, як вони взаємодіють для обробки та використання знань.

Загалом UML надає універсальну та широко зрозумілу мову моделювання, яку можна адаптувати для зображення різних аспектів знань в експертних системах, полегшуючи проектування систем та спілкування між розробниками та зацікавленими сторонами.

SQL. Розробка та застосування SQL (Structured Query Language) моделі представлення знань наведена у роботах [45, 71, 78, 79, 94, 99, 136]. Модель ґрунтується на використанні реляційних баз даних та мови SQL для організації та зберігання інформації. Така модель надає структурований спосіб опису та управління знаннями, визначати відношення між ними та застосовує структуровані запити для вилучення, зміни та аналізу даних. Знання подаються у вигляді таблиць з певними стовпцями, таблиці зв'язуються між собою через ключові поля, що дозволяє встановити відношення між знаннями. Знання про об'єкти предметної області описуються сутностями та їх атрибутами. За допомогою SQL можна виконувати різноманітні запити до даних, зокрема, вибірку даних (SELECT), додавання нових даних (INSERT), зміну даних (UPDATE) та видалення даних (DELETE). Операції JOIN дозволяють поєднувати дані з різних таблиць. Для ідентифікації конкретних знань використовують унікальні ключі або ідентифікатори, які забезпечують унікальність записів у таблиці. SQL модель надає механізми забезпечення цілісності даних і зв'язок між таблицями.

До переваг SQL моделі представлення знань можна віднести: зображення знань у структурованому вигляді; можливість відтворювати

складні зв'язки між даними та таблицями; наявність механізмів для виконання запитів до даних, що дозволяє здійснювати їх обробку та аналіз.

Недоліками SQL моделі є: побудова складних ієрархічних зв'язків між даними, що утруднює використання таких знань користувачем та системою; складність зберігання та обробки текстової інформації чи неструктурованих даних; вимога визначення структури даних заздалегідь, що може обмежити здатність такої моделі адаптуватися до змін.

Однак, слід зазначити, що SQL модель представлення знань знаходить своє застосування у багатьох галузях, включаючи управління базами даних, системами обліку, аналізу даних та інших, де структуровані дані грають визначну роль.

Нечіткі когнітивні карти. Ще один підхід для представлення знань описується у роботах [85]. Нечіткі когнітивні карти – це когнітивні нечіткі графи висновків, у яких вузли позначають поняття, які використовуються для опису поведінки системи, а причинно-наслідкові зв'язки між поняттями представлені дугами зі знаком і вагою [89]. Оскільки нечіткі когнітивні карти моделюють динамічну еволюцію на основі початкової моделі, їх можна використовувати для аналізу та тестування впливу параметрів і прогнозування поведінки системи. Структура графів дозволяє систематичне причинно-наслідкове розповсюдження, зокрема пряме та зворотне, що дозволяє розвивати бази знань шляхом з'єднання різних когнітивних карт.

У низці досліджень нечіткі когнітивні карти застосовуються для моделювання експертних систем [60, 83, 96, 113, 119, 124, 125, 130]. Даний підхід дозволяє відображати неоднозначні та складні зв'язки між концепціями, що допомагає розв'язувати задачі, в яких розв'язок залежить від багатьох факторів. Такі моделі можуть використовуватися для моделювання різних типів знань, зокрема, технічні, наукові, соціальні та інші. Однак створення та налаштування нечітких когнітивних карт є часозатратним і вимагає експертних знань. Деякі зв'язки важко візуалізувати, особливо якщо між поняттями існують складні або суб'єктивні взаємозв'язки. Обробка

нечітких когнітивних карт вимагає значних обчислювальних ресурсів, особливо при великій кількості понять та зв'язків між ними. Інтерпретація карт може бути суб'єктивною та залежати від індивідуального розуміння та досвіду експерта.

Продукційні моделі. Найбільш застосованою при побудові експертних систем є продукційна модель представлення знань, у якій основним елементом є правило вигляду "якщо-то", що ефективно формалізує логічні зв'язки між причиною та наслідком. Продукційні моделі можуть бути реалізовані процедурно і декларативно. У процедурних системах неодмінно повинні бути: база даних, набір продукційних правил та інтерпретатор, за допомогою якого визначається послідовність активізації продукцій [41].

Продукційні моделі застосовуються в тих предметних галузях, де немає чітких правил і задачі розв'язуються на основі незалежних правил (евристик). Правила продукції несуть інформацію про послідовність цілеспрямованих дій. Вони добре відображають прагматичну складову знань і використовуються для невеликих завдань [80].

Так в роботі [64] розроблено ефективну експертну систему на основі правил, для вирішення проблем з напругою, що виникають у споживачів фотоелектричних систем. Основними перевагами цієї експертної системи є "швидке сканування" вхідних даних, виявлення проблем з напругою та генерація рішення щодо положення перемикача РПН під навантаженням, яке не порушує умов для інших користувачів мережі з великою кількістю споживачів. Стійкість мережі, що визначається експертною системою характеризується реакцією на небезпечні події, що представлені раптовими змінами напруги через інтермітентний режим джерел відновлювальної енергії (фотоелектричних систем). Проведено тестування системи та аналіз ефективності.

В роботі [21] розглянуто продукційну нечітку систему, яка використовується для аналізу економічної безпеки підприємства. Ця система складається з узгодженої множини окремих нечітких продукцій, важливим

компонентом системи нечітких продукцій є метод виведення на основі нечітких умов, які містяться у базі нечітких продукційних правил. У процесі роботи методу зайві продукційні правила виключаються.

У [8] пропонується побудова і аналіз нечіткої продукційної моделі оцінки параметра методу випадково-спрямованого пошуку збалансованого стану ротора зі змінним дисбалансом. Нечітка продукційна модель дає можливість всебічно вивчати роботу складної системи без проведення численних фізичних експериментів.

Розглянуто проблеми побудови моделей представлення знань у статті [41]. Описані основні існуючі моделі представлення знань та запропонована гібридна модель на базі технології представлення знань адаптивної системи дистанційного навчання та контролю знань «EduPRO», що поєднує в собі властивості двох моделей - фреймової та продукційної.

В роботі [22] досліджуються можливості використання середовища MATLAB для аналізу моделі загроз інформаційної безпеки та оцінки ризиків інформаційної безпеки об'єкта. Крім цього, у статті розробляється загальна схема інформаційної системи та нечіткої продукційної моделі інформаційної безпеки, підкреслюється доцільність використання нечіткої логіки для оцінки ризику. Реалізація процесу нечіткого моделювання базується на використанні спеціалізованого пакету Fuzzy Logic Toolbox у середовищі MATLAB та використовує алгоритм нечіткого логічного виведення Mamdani.

У статті [43] запропоновано формалізацію представлення продукційної моделі знань у функціональній мові програмування Erlang, включаючи відповідне подання продукційних правил, де умовна частина відповідає логіці першого порядку. Основною метою цієї роботи є розробка функції в Erlang, яка не лише представляє правила в базі знань, але й активує їх при виклику. Наведено відомості про існуючі реалізації логічного виведення із застосуванням Erlang за двома підходами до формування міркувань: логічним — Erlog та продукційним: ERESYE, SERESYE та RUNES II.

Обґрунтовано доцільність розроблення для Erlang власного механізму міркувань, що базується саме на властивостях цієї мови програмування.

Запропонована формалізація продукційної моделі для Erlang передбачає, що кожна одиниця представлення має два визначення, відповідно до синтаксисів логіки та Erlang. Поділ формалізації відповідно до об'єктів представлення на три частини: визначення базових логічних елементів, представлення умов логічного висновування (шаблонів і фактів) та представлення компонентів продукційної моделі (правила, робоча пам'ять, конфліктний набір).

На основі запропонованої формалізації продукційної моделі для Erlang розроблено функцію активації правил бази знань для ефективного логічного виведення. При її виклику, у випадку успішної активації, ця функція повертає кортеж зі списком екземплярів поточного правила та показника його пріоритетності. Кожен елемент списку відповідає комбінації фактів робочої пам'яті, які були успішно узгоджені з шаблонами умовної частини поточного правила. У випадку відсутності будь-якого зіставлення, функція повертає кортеж із порожнім списком. В роботі також наведено приклад створення та використання функції активації для поточного правила.

З огляду на сучасні моделі представлення знань, впливає, що правильний вибір моделі представлення є ключовим фактором для досягнення точності, адекватності та ефективності експертних систем, сприяючи їх успішному застосуванню в розробці експертних систем. Вдосконалення існуючих способів представлення знань та розробка нових підходів є цінним внеском у покращення існуючих експертних систем.

1.3 Способи отримання знань в експертних системах

Сьогодні задачі, що пов'язані з отриманням знань з наборів даних, відносяться до задач аналізу даних, Big Data та Data Mining. Такі задачі виникають під час дослідження об'єктів та процесів, що характеризуються даними з багатофакторними зв'язками, а для їх розв'язання застосовуються

методи штучного інтелекту, що дозволяють знаходити приховані взаємозв'язки між наборами вхідних та вихідних параметрів.

Однією із задач аналізу даних є задача класифікації. Її розв'язок дозволяє розбити множину об'єктів на заздалегідь створені групи (класи) на основі аналізу їх формального опису, що дозволяє прискорити процес подальшої обробки даних. При класифікації кожен об'єкт спостереження відноситься до певної групи або номінальної категорії на основі деякої якісної властивості або сукупності властивостей. В задачах класифікації кожен об'єкт представляється у вигляді певного набору ознак, які описують його характеристики, та класу, до якого цей об'єкт належить. Метою є точне визначення належності об'єкта до певного класу на основі аналізу ознак та виявлення багатofакторних взаємозв'язків між цими ознаками та класами.

На даний момент у різноманітних галузях людської діяльності широко застосовуються інформаційні системи, що акумулювали значні обсяги даних які необхідно обробити та класифікувати. Для ефективного використання цього накопиченого досвіду все більшого значення набувають технології отримання знань та інтелектуальні методи їх обробки при створенні інтелектуальних систем. Основна функція даних систем полягає у підтримці процесу прийняття рішень, що здійснюється на основі акумульованої бази знань та механізму логічного висновку. Важливо, що процес накопичення і формалізації знань характеризується неоднозначністю і, часто, має нетривіальний характер.

У сучасній літературі виділяють такі види методів набуття знань [38]: активні та пасивні комунікативні методи, текстологічні методи та методи отримання знань на основі методів інтелектуального аналізу даних.

Набуття знань з даних на основі використання методів інтелектуального аналізу даних (нейромережеві і нечіткі моделі, дерева рішень, асоціативні правила тощо) дозволяє виконувати процедури автоматично із використанням навчальних вибірок.

Зазначені методи дозволяють отримати знання, проте при формуванні баз знань для експертних систем існує два основні підходи: аналіз знань експертом (групою експертів) на основі досвіду, або автоматичне формування бази знань з використанням методів інтелектуального аналізу даних і алгоритмів машинного навчання. Кожен із цих підходів має свої особливості, переваги, недоліки та може застосовувати дедуктивний, або індуктивний підходи.

Дедуктивний підхід зосереджений на формалізації експертних знань та розробці баз даних, що включають правила і логіку, яку використовують експерти. Індуктивне навчання, або навчання за прецедентами, базується на виявленні загальних властивостей об'єктів на основі неповної інформації, здобутої через досвід. Індуктивний підхід до машинного навчання базується на ідеї розвинення алгоритмів, які можуть навчатися за наявними даними шляхом виявлення загальних закономірностей. Використовуючи навчальний набір даних, алгоритми ідентифікують патерни та залежності, які можуть бути використані для прийняття рішень в нових ситуаціях.

Цей підхід ґрунтується на використанні комп'ютерних алгоритмів для отримання знань із великих обсягів даних, що дозволяє автоматизувати процес отримання знань та надає переваги в обробці масивів даних та виявленні прихованих взаємозв'язків між наборами вхідних та вихідних параметрів.

Обидва підходи мають свої переваги та обмеження, а їх використання залежить від конкретних завдань та контексту. Проте, у ряді випадків комбінація обох підходів може дати найкращі результати, дозволяючи ефективно використовувати експертні знання та збагачувати їх автоматичним аналізом даних.

Низка досліджень застосовувала дедуктивні методи для моделювання експертних систем [83, 96, 119, 125, 130]. Хоча методи дедуктивного моделювання добре зарекомендували себе, вони мають недоліки в тому, що вимагають від експерта повного знання предметної області. Тож модель

може бути обмежена знаннями експерта та його досвідом, що призведе до того, що база знань експертної системи може бути неповною, суб'єктивною, упередженою, суперечливою. Щоб усунути зазначені недоліки, було запропоновано та досліджено кілька методів індуктивного моделювання для генерації моделей з тренувальних вхідних даних без втручання людини [60, 113, 124]. В роботі [113] надає алгоритм навчання на основі Хебба для створення вагових матриць, які наближують адекватність моделі до прийнятного рівня точності. Також в роботах описують алгоритми навчання на основі популяцій, наприклад: алгоритм рою, розділяй і володарюй, та інші для обчислення вагових коефіцієнтів матриць та правил на основі даних, які найкраще відповідають послідовності станів вузлів.

В [52] з використанням теорії графів продукційна база знань представляється у вигляді мультиграфа, в якому кожному продукційному правилу відповідає власний підграф. Побудована таким чином база знань структурується, для збільшення швидкодії механізму логічного виводу умови правил зв'язуються зі значеннями атрибутів в робочій пам'яті, що дозволяє врахувати вплив результатів виконання одних продукційних правил на умови реалізації інших.

В роботі [137] розроблено математичну модель, яка представляє собою гіперграф. Всі сутності і залежності, що представлені в базі знань, об'єднуються у гіперграфі. Модифіковані алгоритми прямого та зворотного виведення здійснюють пошук на отриманому підграфі. Після чого на основі отриманого підграфа будується база знань за допомогою марківської логічної мережі та методу Апріорі.

Представлено підхід до автоматизованого інтелектуального аналізу даних із правилами асоціації на основі знань предметної області в роботі [116]. Правила асоціації представлені як пари загальних булевих атрибутів. Для формулювання обґрунтованих аналітичних запитань використовуються елементи знань предметної області, що відповідають різним відношенням небулевих атрибутів. Окремі елементи знань відображаються в наборах

правил асоціації, які можна вважати їх наслідками. Набори наслідків потім використовуються для інтерпретації наборів правил, отриманих у результаті процедури аналізу даних.

Методи інтелектуального аналізу даних і нейронечітких технології останнім часом знаходять застосування при створенні та аналізі баз знань. Так в [133] з метою вирішення проблеми керування та оптимізації системи доменного газу у металургійній промисловості запропоновано заснований на знаннях оптимальний алгоритм керування, який поєднує вилучення нечітких правил із прогнозуванням на основі нейронних мереж. Нечітка модель розроблена для вилучення знань експортного контролю з даних промислового процесу. Запропонована математична модель нейронної мережі, допускає пряме перетворення бази знань в мережу. З використанням байєсовської інтерпретації регуляризації ядра виконується параметрична оптимізація структури бази знань.

Для налаштування параметрів системи штучного інтелекту застосовуються теорія оптимального розбиття та нечіткі діаграми Вороного. Так в роботі [86] наведено постановку задачі розпізнавання образів в умовах невизначеності. Для розв'язання задачі використовуються методи, засновані на теорії оптимального розбиття множин, яка розроблена О.М. Кісельовою. Особливу увагу приділено застосуванню методів для побудови нечітких діаграм Вороного. Наведено приклади побудови нечітких діаграм Вороного з оптимальним розміщенням точок-генераторів, що підтверджує високу ефективність застосування цих методів у розв'язанні задач для систем штучного інтелекту.

В роботі [88] розглядається задача автоматичної генерації системи нечітких продукційних правил для здійснення нечіткої класифікації об'єктів. Задача генерації нечіткої бази правил за умов відсутності чи недостатньо повного обсягу знань експертів про реальні процеси управління є складною. У роботі представлено дві модифікації методу синтезу та оптимізації баз

правил нечітких систем для прийняття рішень та керування складними технічними об'єктами в умовах невизначеності.

Однак спроби індуктивного моделювання також мають фундаментальні обмеження. По-перше, зазначені методи зосереджені на ідентифікації вагових коефіцієнтів, в той час як набір концептуальних вузлів дається експертами [113]. По-друге, індуктивний підхід використовує лише навчальні дані та перелік явищ, що стосується цільової системи, і припускає, що ті самі тенденції переважатимуть у майбутньому, тобто не може адаптуватися до змін, та передбачити оновлення вагових коефіцієнтів у матриці.

Тому в роботах розробляються системи, що здатні використовувати комбінацію кількох систем штучного інтелекту. Так в роботах [49, 61] розглядаються складові інтелектуальної експертної системи прийняття рішень для епідеміологічної діагностики та вивчається взаємодія системи з користувачем. Система включає банк моделей та методів машинного навчання, засоби візуалізації та звітності, а також блок системи управління для прийняття рішень. Пропонується модель компонентів інформаційних технологій та способів їх розгортання на сервері. Дослідження присвячено розробці та вивченню моделей епідемічного процесу COVID-19 на основі статистичного машинного навчання та оцінці результатів прогнозування. Проведено оцінку результатів експериментального дослідження та аналіз розроблених моделей з точки зору точності та обчислювальної складності.

Це підкреслює важливість вдосконалення індуктивного підходу у вивченні систем штучного інтелекту, де акцент робиться на здатності алгоритмів до самостійного набуття знань і визначення патернів з навчальної вибірки даних, що дозволяє їм ефективно працювати в різних задачах без явного внесення правил в систему. Використання другого підходу дозволяє виконувати процедури створення і контролю баз знань в автоматичному режимі.

Вивчення другого підходу до формування бази знань призводить до висновку, що він є кращим варіантом, виходячи з наступних факторів: він дозволяє значно скоротити часові витрати, зменшити залежність від експертів і врахувати повний обсяг вихідних даних. У рамках цього підходу експерт може брати участь у процесі оцінки сформованої бази знань та її коригування.

У більшості задач моделювання об'єкти мають багатовимірні характеристики, а кількість даних, що потрібно обробити, є потенційно значною. У цьому випадку другий підхід є більш продуктивним і, отже, актуальним. Однак у разі автоматичної генерації отримана база правил буде повною та може містити надлишкові правила. Для скорочення кількості правил використовуються методи редукції бази правил.

1.4 Редукція бази правил експертної системи

У процесі побудови та налаштуванні бази знань експертних систем виникає проблема, пов'язана зі значним збільшенням числа правил у базі знань. Використання експертних систем з такими базами правил вимагатиме значні обчислювальні ресурси і час для обробки та інтерпретації всіх правил, що зрештою призводить до зниження ефективності функціонування системи.

Одним з важливих підходів до вирішення даної проблеми є редукція – скорочення бази правил. Цей процес включає аналіз і оптимізацію правил, щоб зменшити їх кількість, не втративши при цьому набутих знань і не зменшивши функціональності системи. Редукція бази правил дозволяє знизити обчислювальні витрати та підвищити продуктивність експертної системи.

Відповідно до теореми FAT (Fuzzy Approximation Theorem) Б. Коско про нечітку апроксимацію [90] будь-яка, скільки завгодно складна математична залежність може бути апроксимована системою правил на основі нечіткої логіки.

При цьому отримана нечітка система правил має бути ефективною за Парето, тобто необхідно забезпечити оптимальне співвідношення між критеріями складності (кількістю термів для опису нечітких змінних, кількістю правил) і точністю моделі. Такий підхід також покращить інтерпретованість бази правил. База правил, що побудована для змінних, визначених на невеликій кількості термів, і складена з такої кількості правил, що забезпечує необхідну точність моделі, дозволить легко налаштувати модель, а в процесі експлуатації дозволить скоротити обсяг необхідної пам'яті і час виконання логічного виводу.

В експертних системах застосовуються різні методи редукції бази правил з метою оптимізації процесу прийняття рішень та підвищення ефективності системи. Важливість цих методів полягає в необхідності управління зростаючим обсягом правил, що дозволяє забезпечити більш швидку та точну роботу системи.

Декілька поширених методів редукції бази правил [24]:

- 1) виключення правил передбачає видалення надлишкових, нерелевантних чи менш впливових правил з бази знань. Цей метод спрямований на зменшення розміру бази правил, збереження точності та інтерпретованості системи. Оцінюючи внесок кожного правила у результат системи, можна видалити менш значущі правила, що призведе до підвищення ефективності експертної системи;
- 2) виключення суперечливих правил передбачає видалення правил, які взаємно компенсуються;
- 3) виключення збігів передбачає виключення одного з двох правил, які збігаються, як таких, що не несуть нової інформації;
- 4) агрегація правил дозволяє об'єднувати правила із подібними антецедентами/консеквентами в більш узагальнені правила. Цей процес консолідації сприяє скороченню загальної кількості правил у системі, при цьому зберігає точність системи. Агрегація значно зменшує базу правил, полегшуючи управління та оцінку системи експертом;

- 5) апроксимація правил замінює складні правила з великою кількістю антецедентів/консеквентів, меншим набором більш простих правил, які виконують ті ж самі дії при прийнятті рішень. Цей метод знижує складність набору правил, що дозволяє знизити обчислювальні витрати при збереженні прийнятного рівня точності;
- 6) ієрархічні структури правил організовують правила в групи або рівні на основі пріоритету або специфічності. Таке ієрархічне розташування дозволяє системі ефективно орієнтуватися в правилах під час процесу логічного висновку. Використовуючи ієрархічні структури, експертні системи можуть оптимізувати процес прийняття рішень і зменшити витрати на обчислення.
- 7) доменно-спеціальні фільтри знань класифікують правила на основі їх відповідності різним сценаріям. Активуючи лише необхідні підмножини правил для конкретних ситуацій, експертна система може ефективно зменшити кількість правил, покращуючи продуктивність системи та використання ресурсів [81].
- 8) ортогоналізація передбачає видалення тих продукційних правил, вплив яких на точність виявляється мінімальним після оцінки індивідуального внеску кожного продукційного правила у вихідний сигнал мережі, одержуваної шляхом використання ортогонального методу найменших квадратів.

Задача редукції бази правил є актуальною і розглядається у низці робіт. Відомо, що велика кількість правил призводить до ускладнення перевірки бази знань на несуперечність, а також суттєво збільшує час виконання логічного виводу. Тому авторами низки робіт пропонуються підходи, що дозволяють скоротити кількість правил у системі.

Так у роботі [1] розглядається проблема підвищення ефективності використання інтелектуальних систем за рахунок редукції нечітких правил у базах знань. Пропонується два методи редукції: на основі алгоритму кластеризації та на основі генетичного алгоритму. На прикладі редукції бази

знань задачі класифікації ірисів виконано порівняння запропонованих підходів. Із застосуванням генетичного алгоритму задача редукції зводиться до пошуку хромосоми з мінімальним числом правил, що забезпечує якість класифікації. Найкращою буде та хромосома, яка дозволяє досягти максимуму оцінки класифікуючої здатності (не менше від початкової) бази знань за мінімальної кількості правил. Генетичний алгоритм виконується до тих пір, поки в результаті проведення обчислень не з'являтимуться хромосоми з кращою функцією пристосованості протягом певної кількості поколінь. Обчислювальна складність алгоритму безпосередньо залежить від числа лінгвістичних змінних та кількості терм-множин.

Редукції кількості правил для великих нечітких систем на основі знань розглядається у статті [46]. Зберігаючи лінгвістичну інформацію вихідної системи, здійснюється зменшення шляхом вибору найважливіших правил, які несуть значний вклад у зниження кількості помилок. Використовуючи чисельно ефективну модифіковану ортогоналізацію Грама-Шмідта, було отримано показник, відомий як внесок правила у коефіцієнт зменшення помилок (RCERR), що дає змогу користувачеві змінювати послідовність застосування правил залежно від їхньої важливості. Цей алгоритм забезпечує істотну економію витрат на експлуатацію та впровадження. Крім того, він може знайти потенційне застосування, особливо в ситуаціях, де знання послідовності правил є корисним, наприклад у сферах контролю, прийняття рішень і соціально-економічних галузях. Ефективність запропонованої методології ілюструється розглядом деяких типових задач та порівнянням результатів з наявними дослідженнями.

Для досягнення більшої компактності бази правил використовуються різноманітні методи інтерполяції. Так у роботі [87] методи інтерполяції були використані для ущільнення бази правил. Ущільнення бази правил є важливим кроком у вдосконаленні експертних систем і підвищенні їхньої ефективності. Методи інтерполяції базуються на принципі комбінування існуючих правил з метою створення нових, більш компактних та зручних для

системи. Процес інтерполяції може включати у себе аналіз контекстів, у яких використовуються правила, оцінку їхньої значущості та відповідне врахування при об'єднанні або заміні правил. Такий підхід дозволяє побудувати більш узагальнені та універсальні правила, які можуть ефективно застосовуватися в різних ситуаціях. Одним з важливих аспектів використання методів інтерполяції є можливість адаптувати систему до змінних умов та нових джерел даних. Це дозволяє системі бути більш гнучкою та адаптивною, забезпечуючи високу якість результатів у різних сценаріях використання.

За допомогою використання булевих алгоритмів синтезу, було надано метод мінімізації бази правил у роботі [117]. Процес мінімізації бази правил полягає у виявленні та видаленні зайвих, дубльованих або малоефективних правил. Це дозволяє знизити складність системи, покращити швидкість обробки та знизити кількість можливих помилок. Булеві алгоритми синтезу, які застосовуються в цьому методі, відображають взаємозв'язки між різними правилами та сприяють їх оптимальному поєднанню.

Під час розробки методу мінімізації бази правил можуть використовуватися різні стратегії, наприклад, виключення правил, що мають незначний внесок у роботу системи, об'єднання схожих правил або заміна послідовностей правил більш оптимальними рішеннями. Остаточна конфігурація бази правил після мінімізації дозволяє досягти балансу між точністю та продуктивністю системи.

Застосування генетичного алгоритму розглядається також у роботі [44], де здійснюється відбір правил нечіткої бази знань збалансований за критеріями точності та компактності. В якості критерію точності нечіткої моделі обрано середньоквадратичну помилку (RMSE), формулюється задача багатофакторної оптимізації на основі правил і будується область припустимих рішень на границях парето-фронту. Баланс між точністю моделі та її компактністю задається лінійним обмеженням. Для розв'язання задачі оптимізації використовується генетичний алгоритм, в якому кодування

варіантів здійснюється на основі Піттсбурзького підходу. Варіант розв'язку задається хромосомою, кожен ген якої визначає належність відповідного правила до бази знань. Початкова популяція генерується випадково з використанням субоптимальних розв'язків, що відшукуються жадібним алгоритмом. Експерименти проводилися для сингтонних нечітких баз знань, у яких антецеденти правил задаються нечіткими множинами, а консеквенти – числовими значеннями.

Задача формування бази знань нейро-нечітких систем розв'язується у роботі [23]. Використовуються основні принципи редукції правил: виключаються правила, для яких результуюча функція належності менша за певний поріг; виключаються суперечливі правила, що взаємно компенсуються; виключається одне з двох правил, що збігаються; після оцінки індивідуального вкладу кожного правила виконується видалення тих правил, вплив яких на точність є мінімальним.

В роботі [57] застосовується система нечітких правил Такагі-Сугено. Підхід ґрунтується на контролі параметра «coverage of a data», який обчислює кількість активованих правил на основі навчальної вибірки, після чого редукції підлягають лише ті правила, які не збільшують значення coverage. У роботі не розглядалися питання точності та адекватності бази правил після редукції, оскільки видаляються лише правила, що не використовуються. У процесі редукції база правил розбивається на підмножини правил із однаковими консеквентами.

Застосування підходів розрідженого кодування розглядається в роботі [82] разом із теорією розріджених представлень. Завдяки розрідженому кодуванню нечітких базисних функцій правила підлягають скороченню. Для вибору правил, які найбільше впливають на результат, застосовується алгоритм регресії за найменшим кутом LARS (least-angle regression), який є алгоритм підбору моделей лінійної регресії до багатовимірних даних. Правила, що мають найменший вплив на результат, підлягають редукції. У порівнянні з іншими алгоритмами розрідженого кодування, алгоритм регресії

з найменшим кутом має меншу обчислювальну складність і кращу точність, особливо в задачах з багатовимірним даними.

Ще одним перспективним підходом є побудова прогнозуючої нейро-нечіткої мережі. Так у роботі [33] нейро-нечітка мережа використовується для синтезу адаптивного нечіткого регулятора. Нечіткі нейронні мережі поєднують у собі переваги нейронних мереж та систем нечіткого виведення, що дозволяє формувати модель управління багатфакторним процесом у вигляді правил нечітких продукцій. Редукція бази правил здійснюється шляхом вибору правил за допомогою нейронної мережі. У процесі налаштування нейронна мережа оцінює якість поточного набору нечітких правил щодо отриманих результатів та прогнозує коефіцієнт якості управління. З використанням отриманого прогнозу нечіткий регулятор дозволяє спрацьовувати на випередження до виникнення аварійної ситуації. За допомогою такого прогнозу експерт може провести заміну правил у базі знань нечіткого регулятора, або блок адаптації самостійно може обрати необхідні правила та стабілізувати процес регулювання без участі оператора технологічного процесу.

Таким чином, редукція бази правил відіграє важливу роль в поліпшенні ефективності роботи експертних систем, дозволяючи справлятися зі складністю і обсягом знань, що збільшуються. Завдяки цьому процесу система здатна ефективніше опрацьовувати інформацію та швидко приймати рішення.

1.5 Основні алгоритми нечіткого виведення в експертних системах

Основним завданням алгоритмів нечіткого виведення є перетворення нечітких входних даних, виражених у вигляді лінгвістичних термів, у чіткі вихідні значення чи рішення. Ці алгоритми дозволяють системі приймати

рішення з урахуванням нечітких правил, враховуючи невизначеність і нечіткість даних.

До основних алгоритмів нечіткого виведення відносяться: алгоритм Мамдані, що застосовує агрегацію нечітких правил і дефазифікацію для отримання чітких значень вихідних змінних, зокрема, методом визначення середньозваженого центроїдного значення [102]; алгоритм Ларсена, відмінністю якого є застосування операції мінімуму та оператора множення для більш точного моделювання нечітких відношень [95]; алгоритм Сугено, що зображує нечіткі правила у вигляді лінійних функцій, де вихідні значення залежать від вхідних змінних; алгоритм Цукамото, що використовує лінгвістичні терми та вагові коефіцієнти для оцінки ступеня сумісності вхідних значень та вихідних лінгвістичних термів [131].

У роботах [40, 28] в якості алгоритма виведення використовується алгоритм Мамдані, в роботі [2] модель побудована за алгоритмом Мамдані та має чотири вхідних та один вихідний параметр, кожен з яких зображується відповідною функцією належності. Побудована база нечітких знань налічує 81 правило нечіткого логічного виведення. За допомогою побудованої моделі розв'язано задачу оцінювання загального забруднення атмосферного повітря на основі результатів вимірювання концентрації чотирьох чинників: твердих часток PM_{2.5}, чадного газу, діоксиду сірки та діоксиду азоту.

Застосування трьох різних методів нечіткого логічного виведення, а саме, методу Мамдані [102], Ларсена [95] і Цукамото [131] для оцінювання успішності студентів наведено у роботі [59]. Запропоновано нечітку модель для оцінювання успішності студентів на основі двох різних іспитів. Порівняно успішність студентів із застосуванням зазначених методів нечіткого логічного виводу.

В роботі [110] досліджується моделювання нелінійного нечіткого ПІД-регулятора за допомогою алгебраїчного добутку AND-Maximum OR- Larsen та оператора множення.

Методику побудови бібліотек для систем нечіткої логіки розроблено в [4] для FLS-бібліотек. Математичний апарат передбачає застосування правила Мамдані та коефіцієнтів Такагі-Сугено для нечіткої множини першого порядку, нечіткої множини другого порядку та інтервальної нечіткої множини другого порядку. Розроблену комплексну схему використання бібліотек нечіткої логіки може бути використано у мікроконтролерах для роботизованих промислових систем.

В роботі [34] реалізовано програмний комплекс системи підтримки прийняття рішень для автоматизації диспетчерського оперативного керування нормальними і аварійними режимами енергосистеми в умовах невизначеності. У базі знань використовуються правила нечіткої імплікації Ларсена, а блок дефазифікації побудовано на основі Centre Average Defuzzification (CAD). Розроблена експертна система передбачає її використання в якості інтелектуальної комп'ютерної системи підтримки прийняття рішень в складі автоматизованої системи контролю і управління електроенергетичних комплексів.

В рамках дослідження [26] розробляються методи структурно-алгоритмічного синтезу нечітких ПД-регуляторів типу Сугено та Мамдані, що дозволяють автоматичне управління температурою в суднових приміщеннях. Увага приділяється моделюванню систем управління, в яких використовуються синтезовані нечіткі регулятори, з урахуванням наявності збурень. Після цього проводиться порівняльний аналіз властивостей та показників якості автоматичних систем управління, що побудовано на основі традиційних та нечітких ПД-регуляторів.

В роботі [91] розглядається узагальнена нечітка система Такагі - Сугено типу MISO. Запропонований метод дозволяє проводити ефективну оптимізацію різних параметрів нечітких систем, знаходячи оптимальний розв'язок задачі за допомогою удосконаленого алгоритма Grey Wolf. Дослідження ефективності запропонованої інформаційної технології

проведено на імітаційному експерименті з параметричної оптимізації системи нечіткого керування квадаторним безпілотним літальним апаратом.

Використання нечіткої моделі TSK заснованої на системі нечіткого виведення Такагі-Сугено-Канга розглядається у роботі [63]. Пропонується нечітка модель для прогнозування швидкості вітру та кількості виробленої електроенергії на вітровій електростанції. Метою є розробка нечіткої моделі TSK, яка забезпечує прогнозування швидкості вітру на контрольній ділянці на основі коливань швидкості вітру на сусідніх ділянках. Вхідними та вихідними значеннями моделі є середні значення швидкості вітру в минулому, що визначено у певні інтервали часу. Навчальна вибірка вміщує дані про швидкість і напрямок вітру, що вимірюються на сусідніх ділянках на відстані до 30 км від групи вітрових турбін. Розширені результати моделювання надаються для двох сценаріїв застосування, які дозволяють здійснювати прогнози щодо швидкості вітру на період від 30 хвилин до двох годин. Було доведено, що запропонована модель забезпечує адекватний розв'язок поставленої задачі та дозволяє досягти значного покращення енергоефективності вітряної електростанції.

Можна відзначити, що в більшості робіт використовуються класичні алгоритми нечіткого логічного виведення. Однак, з розвитком досліджень у галузі нечіткої логіки та штучного інтелекту з'являються нові підходи та модифікації цих алгоритмів. Крім того, активно досліджуються комбінації різних алгоритмів нечіткого логічного виведення з метою отримання найкращих результатів. Також актуальним є застосування гібридних систем, що поєднують нечітку логіку з іншими методами машинного навчання, такими як нейронні мережі та генетичні алгоритми.

Так в роботі [3] розглядається використання інтелектуального аналізу даних та нейронечітких технології для розв'язання логістичних задач розміщення та розбиття в умовах нечітких початкових даних та невизначеності. В роботі застосовуються нейронечітких технології для розв'язання задачі з нечіткими вхідними параметрами.

Вибір конкретної моделі та алгоритму нечіткого логічного виведення часто залежить від типів даних та конкретної задачі, тому актуальним залишається дослідження та обґрунтування принципів такого вибору в умовах зростання кількості даних, які потрібно аналізувати за допомогою нечітких моделей. Нечітка логіка продовжує залишатися потужним інструментом для моделювання та управління в умовах невизначеності та нечіткості, що робить цю область актуальною і перспективною для досліджень та застосувань.

1.6 Методи налаштування експертної системи

З використанням моделей представлення знань та методів логічного виведення можна побудувати експертну систему, однак для подальшого використання будь-яка система потребує налаштування. Коли експерт має достатній досвід та обсяг знань у певній предметній області, в процесі створення експертної системи він може помітити певну залежність між знаннями та скорегувати систему так, щоб вона залишилась коректною.

Так у роботі [122] представлено ієрархічну нечітку базу правил для інтелектуальної системи підтримки прийняття рішень стосовно причин структурних тріщин кам'яної будівлі. У будівельній практиці причини виникнення тріщин у конструкціях класифікують за такими діагнозами: статичне перевантаження; динамічне перевантаження; особливе перевантаження; дефекти основи та фундаменту; температурний вплив; порушення технологічного процесу під час будівництва. Вхідною інформацією, необхідною для прийняття рішення, є дані візуального дослідження будівлі, що включають вимірювання. Для прийняття рішення враховуються 42 вхідні ознаки. Ієрархічна система пов'язує 9 нечітких баз знань, які містять 151 правило. Виявлення причини тріщини здійснюється методом *max-min* нечіткого висновку з ієрархічною базою знань.

Однак, потребують розвитку методи, що дозволяють налаштовувати експертні системи автоматично на основі даних про об'єкт дослідження. В якості параметрів, що можуть бути налаштовані, в літературних джерелах розглядаються параметри функцій належності, границі термів, кількість правил та порядок їх викликів, вагові коефіцієнти правил та їх компонентів тощо [127, 84].

В статті [127] з метою підвищення ефективності методів пошуку нечітких асоціативних правил з бази даних SIEM запропоновано використовувати вагові коефіцієнти атрибутів, які характеризують ступінь їх важливості в нечіткому правилі. Представлено метод формування асоціативних правил з бази даних SIEM-системи для виявлення кіберінцидентів, який базується на теорії нечітких множин та методах Data Mining. На основі проведеного аналізу зроблено висновок про доцільність виявлення кіберінцидентів у спеціальних інформаційно-комунікаційних системах (SICS) за допомогою правило-орієнтованих методів. Обґрунтовано необхідність застосування технологій інтелектуального аналізу даних, зокрема методів формування асоціативних правил для доповнення бази знань SIEM системи з метою покращення її характеристик у процесі виявлення кіберінцидентів. Розглянуто математичний апарат для формування нечітких асоціативних правил та наведено приклади його застосування. Наведено метод формування нечітких асоціативних правил зі зваженими атрибутами з бази даних SIEM. Сформульовано задачу визначення вагових коефіцієнтів відносної важливості атрибутів правил бази знань SIEM системи та запропоновано метод її розв'язання.

В роботі [84] пропонуються зважені Байєсові асоціативні правила з використанням теорії нечітких множин для проектування та розробки системи підтримки прийняття рішень на основі мережі Байєса. Оскільки область задачі має вищу ступінь непередбачуваності. Кожне правило в базі в процесі обробки навчальних даних отримує певний ваговий коефіцієнт.

Визначення границь термів, зазвичай є задачею експерта. Експерту необхідно обрати кількість термів та обрати границі термів відносно області задачі чи навчальних даних. Тому розробка методів для автоматичного знаходження границь термів та налаштування моделі, дозволить зменшити вплив експерта та поліпшити ефективність процесу моделювання експертної системи.

1.7 Методи аналізу коректності бази правил

Автоматичне формування продукційної бази правил в експертних системах та доведення її коректності є актуальною задачею, оскільки дозволяє забезпечити простоту розробки якісних систем прийняття рішень для користувачів і експертів, та високу якість логічного виведення такою системою. В літературі для вирішення подібних проблем розробляються різні підходи.

В ряді робіт розглядаються різні моделі представлення продукційних баз правил та досліджується коректність експертної системи після її формування [25, 28, 40, 65, 128, 129]. Однак, запропоновані підходи до перевірки коректності бази правил не дозволяють здійснювати перевірку в автоматичному режимі з урахуванням інформації про об'єкт дослідження, що визначає необхідність теоретичних та практичних досліджень у цій галузі знань.

Для аналізу достовірності бази знань нечіткого логічного виведення у роботі [40] розглядається представлення системи правил як метаграфа. Як алгоритм виведення використовується алгоритм Мамдані. Після процедури верифікації база правил має задовольняти наступним умовам: надмірності, лінгвістичної несуперечності, відсутності зациклювання та лінгвістичної повноти. Запропоновано проводити статичну верифікацію баз нечітких правил на основі структури метаграфу шляхом знаходження зациклювання у структурі графа. У роботі розглядається база знань, побудована експертом,

має сенс перевірити даний підхід на базах знань, що генеруються автоматично з великою кількістю правил.

У роботі [28] розглядається питання автоматизації перевірки коректності бази знань продукційних правил. Пропонується квантова модель кодування символів багатозначного алфавіту за допомогою кубічного обчислення. Як алгоритм виведення використовується алгоритм Мамдані. Розробляється комп'ютерна програма, що дозволяє виконати формальну перевірку бази продукційних правил на коректність за допомогою методу резолюцій. Теоретико-множинні операції над символами алфавіту зводяться до відповідних порозрядних логічних операцій над їх кубітами. До недоліків даного підходу можна віднести наступне: вхідні лінгвістичні змінні повинні мати однакову кількість термів, а терми різних вхідних лінгвістичних змінних повинні мати однакові діапазони значень. Крім того, продукційні правила повинні бути повністю визначені, тобто будь-якому поєднанню значень термів вхідних лінгвістичних змінних відповідає певне значення термів вихідної лінгвістичної змінної.

Також для представлення бази знань використовуються ациклічні графи. У роботі [65] для акумуляції великих наборів знань у реальному часі застосовується алгоритм побудови орієнтованого ациклічного графу. Наступна трансляція графа здійснюється за допомогою спеціалізованої мови TLC (Target Language Compiler). У роботі пропонується проводити перевірку бази знань на коректність на етапі трансляції. Застосовуються як класичні методи з використанням диз'юнктивних та кон'юнктивних нормальних форм, так і методи з використанням нормальної форми заперечення NNF (Negation Normal Form). Це дозволяє виводити користувачеві контрприклад, якщо база знань є суперечливою чи неповною. В роботі розглядається побудова бази знань в режимі реального часу, але негативним наслідком стає ефект перенавчання.

У випадку, якщо необхідно провести віддалену приватну верифікацію коректності бази знань, тобто при реалізації розподілених обчислювальних

систем, коли експертна система віддалено навчається на даних, перевірку можна провести за допомогою побудови NIZKP неінтерактивного доведення з нульовим розголошенням (Non-Interactive Zero-Knowledge Proof) [123]. Коли експертні системи працюють з важливими даними, NIZKP може забезпечити безпеку, доводячи коректність експертної системи, не розкриваючи внутрішні параметри та базу правил.

1.8 Критерії оцінки коректності побудованих експертних систем

Однієї з проблем, які виникають при побудові нечіткої моделі, є обґрунтування коректності бази правил. Аналіз продукційних правил бази знань на коректність виконується з урахуванням розробленої вербальної моделі, тому формалізована модель у вигляді бази правил, має задовольняти певним формальним вимогам, які стосуються смислового аспекту правил [80]. База правил є коректною, якщо виконуються такі умови: база правил є повною, несуперечливою, зв'язаною та мінімальною.

Питання, пов'язані з розробкою методів для оцінки якості експертних систем та відлагодження їх баз знань, докладно досліджені в працях вчених, таких, як Ліпаєв В.В, Наріньяні А., Міллер, Нгуєн, Суви, Перкінс, Крагун [31, 35, 107, 111, 131] та інших дослідників. З аналізу цих публікацій, можна зрозуміти, що найбільший успіх досягнуто в розвитку методів статичного відлагодження, тобто тих методів, які дозволяють виявляти помилки без запуску експертної системи.

Варто зазначити, що на сьогоднішній день не існує загальноприйнятого підходу до формалізації так званих структурних помилок, які можуть бути виявлені методами статичного відлагодження. Такі помилки можуть включати надмірність, неповноту та суперечливість. І саме тому, навіть якщо база правил є статично коректною, це не гарантує високої якості прийнятих рішень через можливі помилки в самому знанні, які часто пов'язані з

складністю певної предметної області. Наприклад, це може включати дублювання логіки міркувань.

Так у роботі [36] запропоновано процес синтезу комплексного плану дій з метою підвищення якості системи прийняття рішень. План розробляється з урахуванням різних керуючих впливів і станів зовнішнього середовища. Для цього будуються моделі і алгоритми, які використовують дерево подій, що базується на елементах булевої алгебри і ранговому підході.

Перевірка коректності бази правил, дозволяє уникнути хибних висновків моделі ще на етапі її проектування, а подальша перевірка адекватності нечіткої моделі дозволяє перевірити відповідність моделі реальному об'єкту або процесу.

1.9 Методи нечіткої класифікації

Однією із задач аналізу даних є задача класифікації. Її розв'язок дозволяє розбити множину об'єктів на заздалегідь створені групи (класи) на основі аналізу їх формального опису, що дозволяє прискорити процес подальшої обробки даних. При класифікації кожен об'єкт спостереження відноситься до певної групи або номінальної категорії на основі деякої якісної властивості або сукупності властивостей [70].

Всього існує декілька видів задач класифікації, а саме: бінарна класифікація (binary classification), коли відбувається класифікація об'єкту в один з двох класів; мультикласова класифікація (multiclass classification), коли необхідно класифікувати об'єкт в один з класів; багатоетикетна класифікація (multi-label classification), коли об'єкт може належати одразу кільком класам.

Підходи до розв'язання задач класифікації розглядаються у багатьох роботах. До числа поширених методів розв'язання задачі класифікації відносяться: нейронні мережі; логістична і пробіт-регресія; дерева рішень;

метод найближчого сусіда; машини опорних векторів; дискримінантний аналіз.

Традиційні підходи, засновані на апараті математичної статистики або імітаційному моделюванні, не дозволяють будувати адекватні моделі в умовах обмеженості часових, обчислювальних і матеріальних ресурсів. Тому при розв'язанні багатьох практичних задач, пов'язаних з класифікацією об'єктів, широко застосовуються моделі і методи штучного інтелекту з використанням технологій інтелектуального аналізу даних.

Останнім часом дедалі популярнішими стають нечіткі класифікатори, тобто. класифікатори, у процесі функціонування чи навчання яких використовуються нечіткі множини [92]. Ідея застосування нечітких множин для задач класифікації вперше викладена у статті [50]. Сьогодні найчастіше застосовуються класифікатори на основі логічного висновку за базою продукційних правил, антецеденти яких містять нечіткі терми.

В роботі [101] досліджується застосування нечітких моделей в задачах класифікації на основі представлення даних у вигляді нечітких градацій. Введено класи об'єктів, еталонні зразки в кожному класі, а кожен клас характеризується розподіленою областю значень нечітких критеріїв. Початкові дані для класифікації є суперечливими, а об'єкти, еталони і класи перетинаються, тому побудова розв'язку задачі в класичній постановці викликає значні труднощі. Класифікація розглядається як різновид задачі прийняття рішень, в якій за допомогою узагальнення нечітких фактів, що характеризують властивості, стан або зміну станів об'єктів, здійснюється вибір найкращого класу для кожного об'єкту. Проведено порівняння різних мір узгодженості об'єктів з класами, досліджено їх вплив на результати класифікації.

В роботі [29] розвивається метод гірської кластеризації Ягера-Фільова на випадок нечітких представлень простору станів і ознак, який заснований на обчисленні щільності розподілу інтегральних характеристик об'єктів в нечіткому просторі станів. З використанням нечіткої відстані Хеммінга

визначено приналежність об'єкту при перетині областей нечіткого розподілу ознак.

В роботі [37] розглядається актуальна на даний момент проблема генерації набору нечітких правил для системи нечіткого виводу Мамдані на основі числових даних, отриманих в процесі навчання системи керування. Запропонований підхід базується на алгоритмах чіткої і нечіткої кластеризації - алгоритму гірської кластеризації і алгоритму Густафсона-Кесселя.

1.10 Висновки до розділу

У розділі детально розглянуто архітектуру компонентів системи нечіткого логічного виведення, стандартну та розширену структури компонентів бази знань. Особлива увага приділялася базі знань, яка є основним елементом в роботі експертних систем.

Наведено огляд різних моделей представлення знань в експертних системах. Не зважаючи на велику кількість розглянутих моделей, продукційна модель є більш прийнятною для розв'язання задач класифікації, оскільки легко інтерпретується, дозволяє враховувати складні логічні зв'язки та умови, легко налаштовується шляхом навчання на наборах даних та не вимагає значного їх обсягу. А розширення продукційної моделі нечіткою логікою дозволяє врахувати нелінійні зв'язки між ознаками та нечітко визначені значення параметрів, що корисно, коли границі між класами є "розмитими".

Розглянуто сучасні методи отримання знань в експертних системах. Зазначені підходи істотно поліпшують роботу експертних систем, однак здебільшого дозволяють працювати тільки зі статичними базами знань, накладають обмеження на кількість логічних висновків і не можуть бути застосовані для випадків, коли в існуючу систему необхідно додавати нові логічні правила та динамічно оновлювати під час тренувань. Тому розробка

систем, здатних автоматично навчатися на експериментальних даних та оновлювати існуючу базу правил, дозволяє створювати гнучкі експертні системи, які зберігають свою здатність робити висновки в умовах, що змінюються. Крім цього, такі системи дозволяють акумулювати значні обсяги знань про об'єкти, що мають багатовимірні характеристики, з наборів даних автоматично.

В роботах, що описуються в розділі 1.4, розглядаються різні підходи до розв'язання задач редукції бази правил в експертних системах. Однак, актуальним залишається питання вдосконалення методів редукції правил у базах знань з метою отримання стійких результатів при розв'язанні задач класифікації.

Аналіз літературних джерел та сучасних тенденцій в галузі нечіткої логіки та нечіткого логічного виведення свідчить про широке застосування класичних алгоритмів нечіткого логічного виведення в більшості досліджень. Однак, з розвитком галузі спостерігається зростання інтересу до нових підходів та модифікації цих алгоритмів. В процесі аналізу літератури, було виявлено, що в задачах класифікації границі між класами об'єктів визначені нечітко, тому врахування цього факту потребує вдосконалення методів нечіткого логічного виведення.

Подальше вдосконалення методів нечіткого логічного виведення та їх алгоритмів з урахуванням постановки задачі, характеру та особливостей даних, є необхідним для забезпечення балансу між точністю, швидкістю та інтерпретованістю результатів, що одержуються із застосуванням систем логічного виведення.

При побудові систем класифікації на основі нечіткої логіки важливим є правильна побудова системи правил та налаштування параметрів моделі. Зазвичай для виконання таких робіт залучають групи експертів, однак, використання результатів спостережень та автоматична побудова системи логічного виведення на їх основі дозволить мінімізувати людський фактор, скоротити час та витрати на налаштування моделі.

Точність та якість системи логічного виведення залежить від коректності побудованої бази правил, база правил повинна бути повною, несуперечливою, зв'язаною та мінімальною. Тому потребують розробки нові підходи щодо аналізу та доведенню правильності побудованої бази правил.

Зазначені фактори визначають актуальність розробки та подальшого вдосконалення моделей представлення знань, методів логічного виведення, методів налаштування параметрів моделі нечіткого логічного виведення, методів доведення коректності побудованої бази правил для розв'язання задачі класифікації.

Основні результати розділу опубліковані в [10, 11, 13-16, 19, 20, 134].

Розділ 2. МАТЕМАТИЧНА МОДЕЛЬ ПРОДУКЦІЙНОЇ НЕЧІТКОЇ ЕКСПЕРТНОЇ СИСТЕМИ, ПОБУДОВАНОЇ З ВИКОРИСТАННЯМ ЕКСПЕРИМЕНТАЛЬНИХ ДАНИХ, ТА МЕТОД ЛОГІЧНОГО ВИВЕДЕННЯ

2.1 Постановка задачі

За допомогою штучного інтелекту створюються системи для імітації діяльності експертів у різних галузях. Задачі управління, ідентифікації, моделювання складних фізичних явищ, класифікації, розпізнавання образів успішно розв'язуються з використанням елементів нечіткої логіки [135].

Метою дослідження є розробка підходу до автоматичної генерації нечітких продукційних правил на основі навчальної вибірки для розв'язання задачі класифікації з подальшою перевіркою коректності побудованої моделі.

Це дозволить створити нечіткі експертні системи, здатні навчатися на тестових наборах даних. Продукційні бази правил таких систем дозволять експертам знаходити приховані залежності між ознаками об'єктів та їх класами.

Розглядається множина об'єктів X деякої предметної області. Об'єкти предметної області в задачі класифікації описуються системою ознак k_1, k_2, \dots, k_L , для кожного об'єкта $x_i \in X$ ознакам відповідають лінгвістичні змінні $\hat{k}_1, \hat{k}_2, \dots, \hat{k}_L$, при цьому опис лінгвістичної змінної містить терм-множину $A_l = (A_{l1}, A_{l2}, A_{l3}, \dots, A_{lt})$, де l номер ознаки $l = \overline{1, L}$, а $t = \overline{1, T_l}$; T_l – кількість термів для ознаки k_l ; $\mu_{A_{lt}}(k_l)$ – функція належності чіткого значення вхідної змінної k_l терму.

Вихідна змінна \mathcal{Y} відповідає класу, до якого належить об'єкт предметної області. Існує скінченна множина класів C , серед яких необхідно розподілити об'єкти $x_i \in X$, $i = \overline{1, N}$, N – кількість об'єктів в X .

Для побудови системи класифікації використовуються результати спостережень за об'єктами, ознаки вимірюються та подаються числовими

значеннями, для кожного з об'єктів визначено відповідний клас, до якого він належить. Такий набір даних формує навчальну вибірку X^{Train} виду:

$$X^{Train} = \{(x_{i_{Train}}^{Train}, C_m)\}, i_{Train} = \overline{1, I_{Train}}, m = \overline{1, M},$$

де $(x_{i_{Train}}^{Train}, C_m)$ – пара об'єкт-клас; C_m – мітка класу, до якого належить об'єкт

$x_{i_{Train}}^{Train}$; $C_m \in C$; $m = \overline{1, M}$; M – кількість класів C ; I_{Train} –

розмір навчальної вибірки; $x_{i_{Train}}^{Train}$ – об'єкт навчальної вибірки $x_{i_{Train}}^{Train} \in X^{Train}$;

X^{Train} – навчальна вибірка; $X^{Train} \subset X$.

Навчальна вибірка для скінченного числа об'єктів описує відображення $F^*: x_i \rightarrow C_m$, за допомогою якого можна визначити приналежність певного об'єкту x_i до певного класу C_m .

Необхідно побудувати відображення $F: x_i \rightarrow C_m$, за допомогою якого можливо класифікувати довільний об'єкт з множини X .

У класичній постановці задача класифікації передбачає, що в результаті розбиття будуть отримані детерміновані класи об'єктів, і кожен об'єкт належить тільки одному класу. Однак на практиці такий підхід у деяких випадках призводить до аналітичної невизначеності, тому в роботі розвивається підхід, заснований на представленні об'єктів і класів об'єктів у вигляді нечітких даних, який передбачає розробку системи нечіткого логічного виведення для розв'язання задачі класифікації та налаштування параметрів моделі в процесі виконання процедури класифікації.

Для досягнення мети було поставлено такі завдання:

- провести попередню обробку та аналіз експериментальних даних для дослідження взаємозв'язків між змінними та для виявлення прихованих закономірностей, патернів чи аномалій у наборах даних.

- створити нечітку продукційну модель, що об'єднує моделі Мамдані та Такагі-Сугено-Канг, для представлення знань про об'єкти предметної області з використанням навчальної вибірки та алгоритмів навчання;

- обрати спосіб представлення правил, розробити методику побудови сукупності правил бази знань та алгоритм їх автоматичного формування;
- розробити алгоритми ідентифікації значень параметрів нечіткої продукційної моделі;
- розробити підхід для перевірки коректності побудованої бази правил за критеріями повноти, мінімальності, зв'язності і несуперечності;
- вдосконалити метод логічного виведення із урахуванням того, що інформація про об'єкти предметної області може бути нечіткою та неповною;
- обрати метрики для оцінки якості автоматично побудованої нечіткої бази правил;
- розробити на основі побудованих алгоритмів програмне забезпечення для формування систем нечітких правил та реалізації механізму нечіткого логічного виведення;
- застосувати розроблені методи та алгоритми до розв'язання модельних задач.

2.2 Вибір параметрів об'єктів експериментальної вибірки та встановлення взаємозв'язків

Ознаки об'єкта дослідження бувають суттєві та несуттєві, прямі та непрямі, атрибутивні та кількісні, первинні та вторинні, факторні та результативні. Класифікація статистичних ознак має важливе значення для побудови статистичних моделей та здійснення прогнозу. Так, при моделюванні у ряді випадків важливо правильно виділити факторні та результативні ознаки. Серед факторних ознак необхідно відбирати лише суттєві, що визначають основний зміст явищ.

В аналізі даних широко застосовуються методи кореляційного аналізу, за допомогою яких можна визначати, які змінні найбільше впливають на цільову змінну, і враховувати це при виборі найбільш інформативних ознак моделі, виявити аномалії або непередбачувані зв'язки між ознаками, що може

вказувати на помилки в даних або наявні закономірності, які необхідно врахувати. Процедури кореляційного аналізу передбачають обчислення коефіцієнта кореляції Пірсона та виявлення мультиколінеарності між даними, що є важливими складовими аналітичного аналізу вхідної інформації та дозволяють отримувати глибоке розуміння суті даних і взаємозв'язків між ними. Розглянемо ці аспекти більш детально.

У разі, якщо значення коефіцієнту кореляції наближається до 1, то це може означати, що обидві ознаки сильно впливають на цільову змінну. У цьому випадку включення обох ознак у модель може покращити її здатність до класифікації.

Якщо значення коефіцієнту кореляції наближається до -1, тоді обидві ознаки сильно впливають на цільову змінну, але у протилежних напрямках. Їх включення до моделі також може покращити результати класифікації.

В той же час, висока позитивна чи негативна кореляція між ознаками може викликати проблеми мультиколінеарності, що ускладнює інтерпретацію вкладу кожної з ознак і може погіршити здатність моделі до класифікації. У таких випадках може знадобитися відбір певних ознак, які будуть використовуватися у моделі представлення знань, або застосування методів скорочення вимірності множини ознак.

Якщо значення коефіцієнту кореляції наближається до 0, то це може вказувати на те, що ознаки слабо пов'язані одна з одною і, можливо, не впливатимуть на цільову зміну. Якщо ж коефіцієнт кореляції дорівнює 0, це означає повну незалежність між ознаками. При цьому включення таких ознак у модель класифікації може бути надлишковим та не призведе до покращення результатів класифікації. Скоріш за все, такі ознаки не будуть викликати проблем мультиколінеарності.

Визначення кореляції між ознаками слід розглядати в контексті конкретної задачі машинного навчання. Важливо проводити аналіз даних, щоб визначитися з тим, які ознаки включати в модель, і звернути увагу на можливі проблеми мультиколінеарності.

Однак, кореляція не завжди вказує на причинно-наслідкові зв'язки між ознаками і тому необхідно враховувати інші фактори при побудові моделі.

Також для аналітичного аналізу даних використовують точкову діаграму розсіювання класів. Така діаграма представляє собою інформативний засіб для візуалізації розподілу класів або категорій в деякому наборі даних. Діаграма дозволяє аналізувати кількісне співвідношення різних класів або категорій у досліджуваних даних.

На діаграмі точкового розсіювання класів можна візуально спостерігати, як різні класи розподілені в просторі даних. Це корисний інструмент для задач класифікації, де важливо розуміти, як розділити об'єкти на різні класи на основі їх характеристик.

Діаграма точкового розсіювання може виглядати як графік, де різні класи позначені різними кольорами або символами. Цей графік допомагає відзначити області, де класи перетинаються або знаходяться близько один до одного. Також ця діаграма дозволяє наочно виявити, які класи добре розділяються та їх буде легко класифікувати, та які можуть бути складні для класифікації.

Діаграми точкового розсіювання класів є важливим інструментом для роботи з класифікаційними моделями і допомагають розробляти та вдосконалювати алгоритми класифікації для вирішення різних завдань.

Для перевірки даних на наявність проблем мультиколінеарності в роботі пропонується застосовувати обидва підходи, а саме обчислювати коефіцієнт кореляції Пірсона для наборів тренувальних даних, а також будувати діаграми точкового розсіювання класів.

2.3 Методика побудови бази правил експертної системи

В роботі для опису знань про об'єкти предметної області застосовується продукційна модель представлення знань. Побудова продукційних правил базується на двох основних логічних елементах – антецеденті та консеквенті. Антецедент – це активна частина логічної імплікації, що знаходиться ліворуч від імплікації. Використовується у логіці для опису певної умови, причини, передумови. Консеквент у логіці та правилах це поняття, що означає слідство, висновок щодо причини – антецеденту. Знаходиться праворуч від імплікації.

Продукційні правила подаються у вигляді наступних конструкцій:

Rule_n: IF «Antecedent_1» AND/OR «Antecedent_2» ... AND/OR «Antecedent_k» THEN «Consequent_q»;

де Rule - продукційне правило, n – номер правила у продукційній системі; антецедент (antecedent) - перелік умов, що з'єднуються за допомогою логічних операцій; консеквент (consequent) - логічний висновок, результат класифікації.

Існує два основних типи нечітких правил, а саме нечіткі правила Мамдані та нечіткі правила Такагі-Сугено-Канг (TSK).

Розглянемо загальну схему побудови правил Мамдані:

$P_p: \text{IF } k_1 \text{ is } \tilde{a}_{p1} \text{ AND } k_2 \text{ is } \tilde{a}_{p2} \text{ AND } \dots \text{ AND } k_L \text{ is } \tilde{a}_{pL}$

$\text{THEN } y_1 \text{ is } c_{p1}, \dots, y_m \text{ is } c_{pm},$

де k_1, k_2, \dots, k_L – ознаки об'єкту $x_i \in X$, що описуються лінгвістичними змінними $\hat{k}_1, \hat{k}_2, \dots, \hat{k}_L$; \tilde{a}_{pl} – лінгвістичний терм, яким описується змінна у правилі p; y_1, y_2, \dots, y_m – вихідні змінні; $p = \overline{1, P}$; P – загальна кількість правил П; c_{pm} – приймає значення $C_m \in C$ для правила p; C_m – мітка класу, до якого належить об'єкт $x_i \in X$.

На відміну від нечітких правил Мамдані правила Такагі-Сугено-Канг (TSK) використовують у якості консеквенту правила функції. Аргументами таких функцій є вхідні змінні.

$$P_p: \text{IF } k_1 \text{ is } \tilde{a}_{p1} \text{ AND } k_2 \text{ is } \tilde{a}_{p2} \text{ AND } \dots \text{ AND } k_L \text{ is } \tilde{a}_{pL}$$

$$\text{THEN } y_1 = f_1(k_1, \dots, k_L), \dots, y_M = f_M(k_1, \dots, k_L),$$

де функція $f_m(k_1, \dots, k_L)$ – дійсна, невід'ємна, нормована, неперервна на відрізьку $[a, b]$.

В роботі для розв'язання задачі класифікації застосовується продукційна модель правил, що поєднує моделі Мамдані та Такагі-Сугено-Канг (TSK).

Для розв'язання задач класифікації використовуються системи з декількома вхідними та одним вихідним параметром для класифікації об'єктів до одного класу (MISO), та декількома вхідними та декількома вихідними параметрами (MIMO) для класифікації об'єктів у декілька класів. Сам термін MISO означає "Multiple-Input and Single-Output", а термін MIMO – "Multiple-Input and Multiple-Output" [47, 53, 58, 105].

Запропонована модель представлення правил для систем типу MISO та MIMO зображується у такий спосіб:

правила P_p для MISO систем мають вигляд:

$$P_p: \text{IF } k_1 \text{ is } \tilde{a}_{p1} \text{ AND } k_2 \text{ is } \tilde{a}_{p2} \text{ AND } \dots \text{ AND } k_L \text{ is } \tilde{a}_{pL} \quad (2.1)$$

$$\text{THEN } y \text{ is } f(k_1, \dots, k_L, p) \cdot c_p,$$

де c_p – змінна, що приймає значення $C_m \in C$ для правила p ;

правила P_p для MIMO систем мають вигляд:

$$P_p: \text{IF } k_1 \text{ is } \tilde{a}_{p1} \text{ AND } k_2 \text{ is } \tilde{a}_{p2} \text{ AND } \dots \text{ AND } k_L \text{ is } \tilde{a}_{pL} \quad (2.2)$$

$$\text{THEN } y_1 \text{ is } f(k_1, \dots, k_L, p) \cdot c_{p1}, \dots, y_m \text{ is } f(k_1, \dots, k_L, p) \cdot c_{pm}.$$

Дійсна, невід'ємна, нормована функція $f(k_1, \dots, k_L, p)$ є неперервною на відрізьку $[a, b]$ і використовується як ваговий коефіцієнт для консеквенту

правила. Вона визначає ступінь належності вихідної змінної y до терм-множини C_m та обчислюється в такий спосіб:

$$f(k_1, \dots, k_L, p) = \frac{\sum_{l=1}^L \mu_{\tilde{a}_{pl}}(k_l)}{L}.$$

де $\mu_{\tilde{a}_{pl}}(k_l)$ – функція належності чіткого значення вхідної змінної k_l нечіткому терму \tilde{a}_{pl} .

2.4 Алгоритм автоматичного формування бази правил

Для автоматизації процесу побудови системи правил в роботі пропонується зображення бази правил у матричному вигляді із застосування матриці антецедентів A та вектору консеквентів \vec{C} .

Для кожної лінгвістичної змінної k_l , існує скінченна (квазіскінченна, якщо допустимо доповнення моделі) нечітка терм-множина $A_l = (A_{l1}, A_{l2}, A_{l3}, \dots, A_{lt})$, що визначає скінченний алфавіт для опису станів або множини станів лінгвістичної змінної.

Кількість нечітких класів C_m , до яких належить об'єкт $x_i \in X$, також обмежена. Таким чином, елементи A_{lt} та C_m визначають алфавіт продукційної моделі.

Формально матриця антецедентів A задається впорядкованим набором $\langle A_{l1}, \dots, A_{lt} \rangle$, де кожен вектор-рядок матриці A представляє собою множину умов продукційного правила. Вектор консеквентів \vec{C} складається з класів C_m і має вимірність, що дорівнює кількості правил, та встановлює відповідність між продукційним правилом у матриці A та нечітким класом C_m .

Множину продукційних правил Π можна представити у вигляді: $\Pi_p = \langle a_p, c_p \rangle$, де a_p – вектор-рядок матриці A . Терм-множина $A_l = (A_{l1}, A_{l2}, A_{l3}, \dots, A_{lt})$ є впорядкованою відносно семантичного порядку та границь термів. Для побудови матриці A пропонується використовувати декартовий добуток:

$$A_1 \times A_2 \times A_3 \times \dots \times A_l = A^{xl}.$$

Після чого, множину A^{xl} , можна представити у вигляді матриці A :

$$A = \begin{pmatrix} A_{11} & A_{21} & A_{31} & \dots & A_{l1} \\ A_{12} & A_{22} & A_{32} & \dots & A_{l2} \\ A_{13} & A_{23} & A_{33} & \dots & A_{l3} \\ \dots & \dots & \dots & \dots & \dots \\ A_{1l} & A_{2l} & A_{3l} & \dots & A_{ll} \end{pmatrix}.$$

Кількість правил P визначається з властивостей декартового добутку та дорівнює добутку кількості елементів множин-співмножників:

$$P = \prod_{l=1}^L |A_l| = \prod_{l=1}^L T_l. \quad (2.3)$$

Формування вектора консеквентів здійснюється за допомогою процедури навчання бази правил. Кожен об'єкт з навчальної вибірки X^{Train} проходить процедуру фазифікації $fuzz(x_{i_{Train}}^{Train})$, після чого вектор $\vec{C} = (c_1, c_2, \dots, c_p)$ визначається наступним чином:

$$c_p = C_m : \max_i \left(card_i \left(fuzz(x_{i_{Train}}^{Train}) \in a_p, C_m \right) \right),$$

де $card()$ – функція потужності множини.

Вектор \vec{C} повинен містити в собі всі класи C_m , інакше на вхід подано неповну або неправильну навчальну вибірку і необхідно перенавчити систему на іншому наборі даних.

2.5 Метод логічного виведення

Для здійснення процедури логічного виведення застосовуються операції фазифікації та дефазифікації.

Процедура фазифікації визначається в такий спосіб:

$$\tilde{a}_{pl} = \int_{\bar{k}_l}^{k_l} \mu_{\tilde{a}_{pl}}(k_l) |k_l| dk,$$

де $\mu_{\tilde{a}_{pl}}(k_l)$ – функція належності чіткого значення вхідної змінної k_l нечіткому терму \tilde{a}_{pl} .

Для автоматизації процесу побудови системи правил, зручно представити антецеденти правил у вигляді матриці A :

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1l} \\ a_{21} & a_{22} & \dots & a_{2l} \\ \dots & \dots & \dots & \dots \\ a_{p1} & a_{p2} & \dots & a_{pL} \end{pmatrix},$$

де a_{pl} – елемент матриці, що приймає значення \tilde{a}_{pl} змінної із системи правил і дорівнює нечіткому значенню A_{lt} для правила p ; P – кількість правил; L – вимірність вектору ознак $\bar{K} = (k_1, k_2, \dots, k_L)$ об'єкту $x_i \in X$.

Консеквенти правил представлятимемо у вигляді вектору стовпця \vec{C} :

$$\vec{C} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \dots \\ c_p \end{pmatrix}.$$

Далі, нечітка множина класів \tilde{C} , що відповідають вихідній змінній Y , визначається у вигляді:

$$\tilde{C}_m = \left\langle \sum_{p=1}^P f(k_1, \dots, k_L, p) \mid C_m \in \Pi_p, C_m \right\rangle.$$

Таким чином, множина \tilde{C} складається з упорядкованих пар $\langle f_m | C_m \rangle$, таких що:

$$f_m = \sum_{p=1}^P f(k_1, \dots, k_L, p) \mid C_m \in \Pi_p,$$

де пара $\langle f_m \mid C_m \rangle$ – визначає ступінь належності f_m об'єкта x_i класу C_m .

Для дефазифікації \tilde{C} застосовується алгоритм Мамдані із центроїдним методом [102]. Нечітка множина \tilde{C} класів, відповідає вхідному об'єкту X визначається у вигляді:

$$\tilde{C} = \underset{m=1, M}{agg} \left(\int_{\underline{C}}^{\bar{C}} imp(f_m \cdot \mu_{C_m}(C), \mu_{C_m}(C)/C) dC \right),$$

де операції імплікації imp та агрегування agg реалізуються шляхом знаходження min та max відповідно.

$$C^* = \int_{\underline{C}}^{\bar{C}} C \cdot \mu_{\tilde{C}}(C) dC / \int_{\underline{C}}^{\bar{C}} \mu_{\tilde{C}}(C) dC.$$

Чітке значення логічного висновку C^* визначається в результаті дефазифікації нечіткої множини \tilde{C} за методом центру тяжіння.

2.6 Метод вирішення конфліктів при прийнятті рішень з урахуванням відстані між класами.

У разі, коли не вдається визначити клас, до якого належить об'єкт, або, якщо в результаті класифікації об'єкт належить двом класам одночасно, необхідно здійснити процедуру вирішення конфлікту для прийняття рішення експертною системою.

В роботі розроблено алгоритм міжкласової класифікації на основі відстані між об'єктами з навчальної вибірки.

Для цього кожен об'єкт навчальної вибірки $X^{Train} = \{(x_{i_{Train}}^{Train}, C_m)\}$ в процесі навчання проходить через процедуру фазифікації та дефазифікації.

На етапі дефазифікації отримуємо центр тяжіння для кожного об'єкта з навчальної вибірки. Збережемо отримані центри до множини $Y_{i_{Train}}^*$.

Після закінчення тренування системи розпочинається процес класифікації об'єкта заданого вектором ознак. На цьому етапі кожна ознака об'єкта $x_i \in X$ проходить через стандартний алгоритм фазифікації, після чого на етапі дефазифікації отримуємо центр тяжіння об'єкта X^* . Після чого відбувається пошук найближчих об'єктів з навчальної вибірки, що відрізняються один від одного на заздалегідь задану величину $\varepsilon > 0$. Пошук найближчих об'єктів системи здійснюється за допомогою введеної метрики – евклідової відстані. У разі, якщо в базі знань не існує об'єкта, з яким можна ототожнити вхідний об'єкт, то для нього формулюється нове логічне правило, створюється новий клас об'єктів, динамічно перераховуються границі термів для всіх об'єктів бази знань.

Для організації процедури порівняння визначимо відстань між об'єктом X^* та об'єктами навчальної вибірки $X^{Train} = \{(x_{i_{Train}}^{Train}, C_m)\}$. Відстань визначається на основі обраної метрики в просторі параметрів:

$$d((X^*, M^*), (Y_{i_{Train}}^*, M_i^*)) = \sqrt{(X^* - Y_{i_{Train}}^*)^2 + (M^* - M_i^*)^2} \quad (2.4)$$

$$d((X^*, M^*), (Y_{i_{Train}}^*, M_i^*)) < \varepsilon, \quad (2.5)$$

де $M^* = \mu_{X^*}(x_i)$, $M_i^* = \mu_{Y_{i_{Train}}^*}(x_{i_{Train}}^{Train})$,

Після виконання процедури дефазифікації обчислюється $d((X^*, M^*), (Y_{i_{Train}}^*, M_i^*))$ та перевіряється умова (2.5).

Далі будуємо множину $\overset{d}{Y}$, до якої відносяться об'єкти навчальної вибірки, що схожі з об'єктом X^* , якщо $d((X^*, M^*), (Y_{i_{Train}}^*, M_i^*)) < \varepsilon$, то

$$(x_{i_{Train}}^{Train}, C_m) \in \overset{d}{Y}.$$

Клас, до якого належить вхідний об'єкт X^* , визначається у такий спосіб:

$$\max_{m=1}^M \left(\text{card}_{C_m}^d(Y) \right), \quad (2.6)$$

де $\text{card}_{C_m}^d(Y)$ - потужність множини Y .

Результат класифікації за допомогою запропонованого підходу наведено на рис. 2.1.

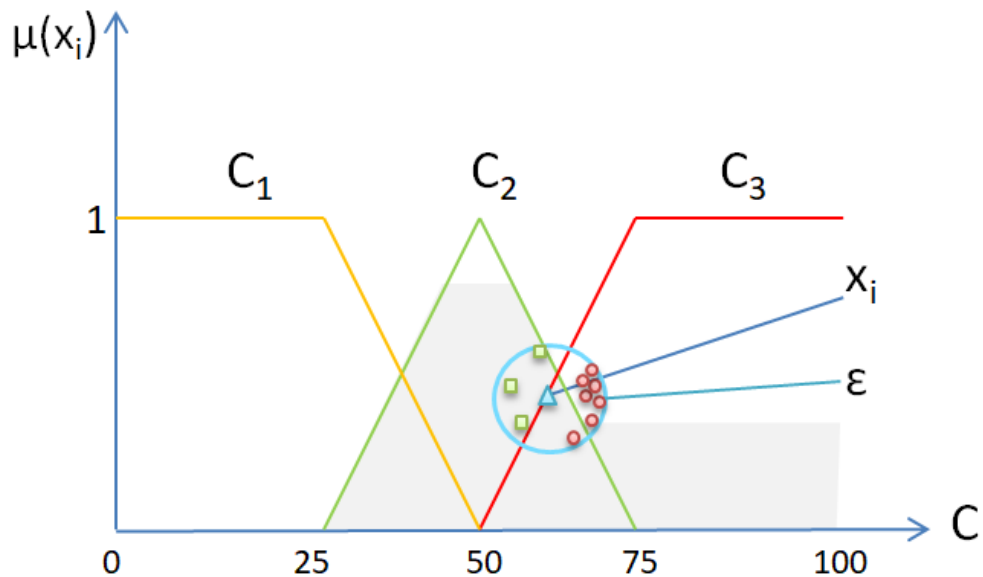


Рис. 2.1. Результат класифікації за допомогою запропонованого підходу

З аналізу рис. 2.1. можна бачити, що навколо об'єкту x_i (на рис. 2.1. позначений трикутником) розташовано три об'єкти класу C_2 (на рис. 2.1. позначені квадратами) та сім об'єктів класу C_3 (на рис. 2.1. позначені кругами), що з урахуванням (2.6) свідчить про належність об'єкта X до класу C_3 з більшою ймовірністю, ніж до класу C_2 .

Вказану послідовність дій було описано таким алгоритмом.

Алгоритм:

Крок 0. Ініціалізація. Задати навчальну вибірку $X^{Train} = \{(x_{i_{Train}}^{Train}, C_m)\}$; значення ε – похибка системи.

Крок 1. За допомогою навчальної вибірки побудувати функції належності та розрахувати границі термів лінгвістичних змінних.

Крок 2. Побудувати систему правил.

Крок 3. За допомогою механізму нечіткого логічного виведення розрахувати ступені приналежності об'єктів навчальної вибірки нечітким множинам, визначити чітке значення $Y_{i_{Train}}^*$ та M_i^* .

Крок 4. Для вхідного об'єкту $x_i \in X$ на базі сформованих продукційних правил також провести процедуру фазифікації та дефазифікації та отримати чітке значення X^* та M^* .

Крок 5. Обчислити відстані між X^* та об'єктами з навчальної вибірки за формулою (2.4), перевірити виконання умови (2.5). Якщо нерівність (2.5) є вірною, побудувати множину $\overset{d}{Y}$ та перейти до кроку 7, інакше, якщо, не вдалось побудувати множину $\overset{d}{Y}$, перейти до кроку 6.

Крок 6. Об'єкт неможливо класифікувати – необхідно створити новий клас.

Крок 7. Для побудованої множини $\overset{d}{Y}$, знайти потужність для кожного з класів C_m . Отримати результат ідентифікації з умови (2.6).

У разі, коли неможливо однозначно визначити клас об'єкта, або коли об'єкт може належати кільком класам одночасно, розроблений алгоритм міжкласової класифікації на основі відстані між об'єктами з навчальної вибірки дозволяє системі приймати обґрунтовані рішення в умовах невизначеності.

2.7 Метрики оцінки якості класифікації

Оскільки для розв'язання задачі класифікації використовується експертна система з нечіткою базою правил, отриманий результат класифікації є наближеним, тому необхідно мати можливість оцінити якість отриманого результату. Для оцінки якості моделі та результату класифікації, отриманого з використанням нечіткої бази правил, необхідно порівняти висновки, що отримано за допомогою моделі, з тренувальними даними або з результатами для реальних об'єктів (процесів). Для цього в роботі застосовуються метрики, що визначають частину правильних відповідей моделі (accuracy), точність (precision), повноту (recall), f-міру (f1-score) [126]:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} ;$$

$$\text{precision} = \frac{TP}{TP + FP} ;$$

$$\text{recall} = \frac{TP}{TP + FN} ;$$

$$\text{f1-score} = \frac{2 \times (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}} ,$$

де TP (True Positive) - кількість об'єктів, які були правильно класифіковані в системі як такі, що належать до певного класу C; TN (True Negative) - кількість об'єктів, які були правильно класифіковані як такі, що не належать до певного класу C; FP (False Positive) - кількість об'єктів, які були неправильно класифіковані в системі як такі, що належать до певного класу C, хоча насправді не належать до зазначеного класу; FN (False Negative) – кількість об'єктів, які були неправильно класифіковані в системі як такі, що не належать до певного класу C, хоча насправді належать зазначеному класу.

За допомогою метрики Accuracy обчислюється частина правильних відповідей моделі. З використанням метрики Precision можливо знайти

відношення дійсно правильно класифікованих об'єктів до загальної кількості об'єктів, що класифіковано як такі, що належать класу. Метрика Recall описує наскільки взагалі модель може виявляти правильні відповіді та обчислюється як відношення вірно класифікованих об'єктів до загальної кількості об'єктів класу. Узагальнююча метрика f1-score приймає значення 1 лише тоді, коли значення метрик Precision та Recall є рівними 1. F-міра зростає лише тоді, коли Precision та Recall зростають.

Подальша перевірка адекватності нечіткої моделі здійснюється за допомогою метрик, що дозволяє визначити якість результату нечіткої класифікації.

2.8 Адаптивний алгоритм редукції бази правил

Враховуючи, що автоматично побудована база правил має значний розмір і може бути надлишковою, необхідно розробити підхід, який дозволяє виконати скорочення кількості правил у базі знань не знижуючи при цьому якість класифікації.

Для зменшення надлишкових правил прийнято застосовувати процедури редукції [1], коли база правил поділяється на правила, що будуть використовуватися, та надлишкові правила відповідно до визначеного критерію. В роботі для здійснення процедури редукції бази правил введемо критерій, за яким будемо поділяти правила з бази правил, у такий спосіб:

$$\Delta < \min(\text{accuracy}, \text{avg}(\text{precision}, \text{recall}, \text{f1-score})), \quad (2.7)$$

де accuracy, precision, recall, f1-score – метрики якості та повноти класифікації; avg - операція, що усереднює значення метрик якості.

Критерій Δ приймає значення від 0 – 100% і буде мірою якості, яку необхідно досягти. Досягнення критерієм заданого значення використовується як умова зупинення процедури редукції вхідної бази правил.

Для організації процедури редукції у кожне правило вводяться додаткові логічні умови:

$$\Pi_p : \text{IF} (\alpha_p > 0 \text{ AND } \beta_p > 0) \text{ AND } k_1 \text{ is } a_{p1} \text{ AND } k_2 \text{ is } a_{p2} \text{ AND } \dots \text{ AND } k_L \text{ is } a_{pL} \quad (2.8)$$

THEN y is $f(k_1, \dots, k_L, p) \cdot c_p$;

$$\alpha_p = \sum_{i_{Train}=1}^{I_{Train}} 1, x_{i_{Train}}^{Train}(k_1, k_2, \dots, k_L) : (k_1 \text{ is } a_{p1} \text{ AND } k_2 \text{ is } a_{p2} \text{ AND } \dots \text{ AND } k_L \text{ is } a_{pL}) \text{ EQ}(\text{TRUE}) \quad ; \quad (2.9)$$

$$\beta_p = \frac{\sum_{i_{Train}=1}^{I_{Train}} f(k_1, \dots, k_L, p), x_{i_{Train}}^{Train}(k_1, k_2, \dots, k_L) : (k_1 \text{ is } a_{p1} \text{ AND } \dots \text{ AND } k_L \text{ is } a_{pL}) \text{ EQ}(\text{TRUE})}{\alpha_p}, \quad (2.10)$$

де коефіцієнт активації α є лічильником кількості активацій правила Π_p для

об'єктів $x_{i_{Train}}^{Train}$ навчальної вибірки X^{Train} ; коефіцієнт β визначає ступінь

належності об'єктів навчальної вибірки правилу Π_p ; I_{Train} – розмір

навчальної вибірки, $x_{i_{Train}}^{Train}$ – об'єкт навчальної вибірки $x_{i_{Train}}^{Train} \in X^{Train}$,

X^{Train} – навчальна вибірка.

Для організації алгоритму редукції застосовується метод дихотомії з динамічним кроком. Даний метод дозволяє розділити множину правил на результуючу множину правил і множину правил, що є надмірними для досягнення критерію якості Δ . Таким чином, результуюча множина правил задовольнятиме критерію якості (2.7).

Адаптивний алгоритм редукції бази правил. Алгоритм процедури може бути представлений такою послідовністю дій:

Крок 0. На вхід системи подати базу правил. Задати критерій точності Δ . Задати навчальну та тестову вибірки. Виконати процедуру нечіткого логічного виводу Мамдані з центроїдним методом. Обчислити коефіцієнти α і β за формулою (2.9) і (2.10) для навчальної вибірки. Скоротити правила, для яких значення коефіцієнта α дорівнює 0.

Крок 1. Обчислити метрики якості моделі accuracy, precision, recall, f1-score для всіх правил на тестовій вибірці та перевірити умову (2.7). Якщо (2.7) є істиною, то набір правил Π підлягає редукції. Перейти до кроку 2.

Крок 2. Відсортувати усі правила Π відносно коефіцієнта β , отримати впорядкований набір правил $\vec{\Pi}$. Створити порожню множину $\dot{\Pi} = \emptyset$ для збереження в неї нової бази правил, яка формуватиметься внаслідок ітеративного процесу.

Крок 3. Обчислити потужність множини $\vec{\Pi}$, якщо $card(\vec{\Pi})=1$, то $\dot{\Pi} = \dot{\Pi} \cup \vec{\Pi}$ і перейти до кроку 7, інакше, перейти до кроку 4.

Крок 4. Розділити набір правил $\vec{\Pi}$ на 2 множини: $\vec{\Pi}_1 = \{\vec{\Pi}_1, \dots, \vec{\Pi}_{\bar{p}/2-1}\}$ і $\vec{\Pi}_2 = \{\vec{\Pi}_{\bar{p}/2}, \dots, \vec{\Pi}_{\bar{p}}\}$, при цьому $\forall (\alpha, \beta) \leq \forall (\alpha, \beta)$.

Крок 5. Обчислити accuracy, precision, recall, f1-score для правил $\dot{\Pi} \cup \vec{\Pi}_2$ на тестовій вибірці.

Крок 6. Перевірити умову (2.7), якщо вона є істинною, то набір правил $\vec{\Pi} = \vec{\Pi}_2$, перейти до кроку 3, інакше, $\dot{\Pi} = \dot{\Pi} \cup \vec{\Pi}_2$, $\vec{\Pi} = \vec{\Pi}_1$, перейти до кроку 3.

Крок 7. Завершити процес редукції бази правил, отримати множину $\dot{\Pi}$.

В результаті роботи алгоритму з бази правил видаляються правила, антецеденти яких не активовані, серед правил, що залишилися, вибираються правила, які мають найбільший вплив на результат класифікації.

2.9 Висновки до розділу

В розділі надано постановку задачі автоматичної генерації нечітких продукційних правил на основі навчальної вибірки з метою розв'язання задачі класифікації. Цей підхід розширює можливості нечітких експертних систем, дозволяючи їм навчатися на основі наборів даних та будувати продукційні бази правил, які визначають приховані залежності між ознаками

об'єктів та класами, до яких вони належать. У традиційних задачах класифікації передбачається визначення детермінованих класів для об'єктів, і кожен об'єкт належить лише одному класу. Проте на практиці, особливо в складних задачах, такий підхід може бути недостатнім, а аналітична невизначеність може виникати внаслідок непередбачуваних залежностей між ознаками. У цьому контексті розглянуто можливість використання нечітких даних, побудовано систему нечіткого логічного виведення для задач класифікації.

Для проведення попередньої обробки та аналізу експериментальних даних в роботі запропоновано застосування кореляційного аналізу для виявлення взаємозв'язків між змінними та прихованих закономірностей, патернів чи аномалій у наборах даних. Особлива увага приділяється обчисленню коефіцієнта кореляції Пірсона та виявленню мультиколінеарності, як ключових аспектів аналізу великих обсягів навчальних даних. Додатково, запропоновано використання діаграм точкового розсіювання класів для візуалізації розподілу даних, що сприяє більш глибокому розумінню їхньої суті та взаємозв'язків.

Створено продукційну модель представлення знань, яка об'єднує моделі Мамдані та Такагі-Сугено-Канг і використовує функцію як ваговий коефіцієнт для консеквенту правила. Це дозволило визначити рівень впливу кожного правила на результат висновку продукційної системи.

Розроблено алгоритм автоматичного формування бази правил з урахуванням даних моделі предметної області. Матриця антецедентів створюється за допомогою ознак об'єктів з деякої кінцевої множини з використанням декартового добутку. Для формування вектора консеквентів використовується навчальна вибірка, завдяки цьому забезпечується можливість створювати базу правил та здійснювати налаштування моделі в автоматичному режимі.

Удосконалено метод логічного виведення в частині вирішення конфліктів при прийнятті рішень у разі, коли системі не вдається однозначно

визначити клас, якому належить об'єкт. Для цього реалізовано процедуру порівняння відстані між об'єктами та класами на етапі класифікації.

Для оцінки якості отриманої нечіткої експертної системи та результатів класифікації обрано метрики accuracy, precision, recall, f1-score.

Важливим етапом цього дослідження було також розроблення алгоритму адаптивної редукції правил бази знань на основі запропонованої метрики, що поділяє систему правил на ті, що застосовуються, і ті, що є зайвими. Для реалізації алгоритму редукції застосовується метод дихотомії з динамічним кроком.

Основні результати розділу опубліковані в [9-16, 19, 134].

Розділ 3. НАЛАШТУВАННЯ ПАРАМЕТРІВ НЕЧІТКОЇ ПРОДУКЦІЙНОЇ МОДЕЛІ

Для побудови якісної експертної системи класифікації необхідно провести налаштування нечіткої продукційної моделі. У розділі 2.2 було розглянуто методи та інструменти для вибору ознак об'єктів експериментальної вибірки та встановлення взаємозв'язків між об'єктами. За допомогою наведених методів та інструменти необхідно провести попередню обробку експериментальних даних. Це дозволить виявити взаємозв'язки між змінними, виявляти приховані закономірності, патерни чи аномалії у наборах даних та отримати розуміння суті даних.

Також побудована модель нечіткого логічного виведення потребує налаштування задля досягнення якості класифікації. Налаштування моделі передбачає налаштування функцій належності шляхом визначення границь термів. Налаштування функцій належності є важливим етапом, оскільки функція належності визначає ступінь належності ознаки об'єкту до певного терму. Це впливає не лише на якість експертної системи але і на те, як система інтерпретує та використовує знання експерта в процесі прийняття рішень.

Налаштування додаткових параметрів, таких як ваги правил, що визначають важливість кожного правила, чи параметри для оптимізації штрафів за нечіткість або надмірне навчання сприятимуть покращенню побудованої моделі. Після налаштування моделі необхідно провести аналіз коректності отриманої експертної системи та аналіз якості класифікації, що здійснюється із застосуванням системи.

3.1 Попередня обробка експериментальних даних для побудови експертної системи

3.1.1 Застосування точкової діаграми розсіювання для попереднього аналізу даних

Для наочного встановлення взаємозв'язків між змінними, застосовується точкова діаграма розсіювання, за допомогою якої можна встановити попарні взаємозв'язки між різними змінними в наборі даних. Діаграма розсіювання зображується у вигляді графіку, на якому кожна точка відповідає одному спостереженню. Дані зображуються у вигляді набору точок, кожна з яких має значення однієї з ознак, та визначає значення ознак на горизонтальній та вертикальній осях.

Діаграми розсіювання часто використовуються для пошуку кореляцій між змінними, оцінки розподілу даних, виявлення викідів та аномалій у наборі даних, виявлення та подальшого моніторингу трендів. За допомогою діаграм розсіювання можна подолати складності та шум у даних, візуалізувати багатовимірні взаємозв'язки та ухвалювати обґрунтовані рішення на основі даних.

На рис. 3.1 можна побачити приклад побудованої діаграми розсіювання для трьох класів.

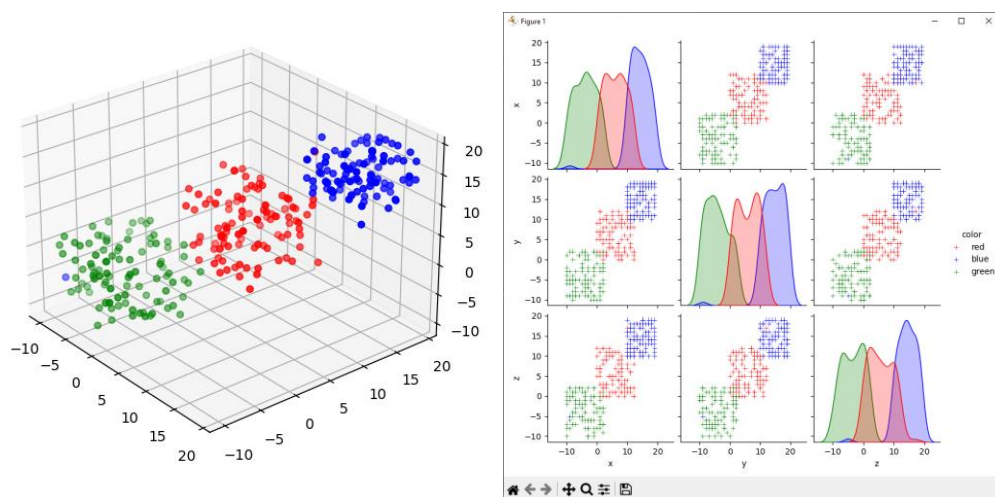


Рис. 3.1. Точкова діаграма розсіювання

На рис. 3.2 побудовано точкову діаграму розсіювання для набору Iris Data Set - Іриси Фішера [73]. Задача класифікації передбачає визначення належності ірису до одного з трьох типів рослин: Setosa, Versicolor, Virginica. Усього в наборі міститься 150 об'єктів, кожному із трьох класів відповідає по 50 об'єктів. В якості ознак об'єкту використовуються довжина чашолистка – sepalLength, ширина чашолистка – sepalWidth, довжина пелюстки – petalLength, ширина пелюстки – petalWidth.

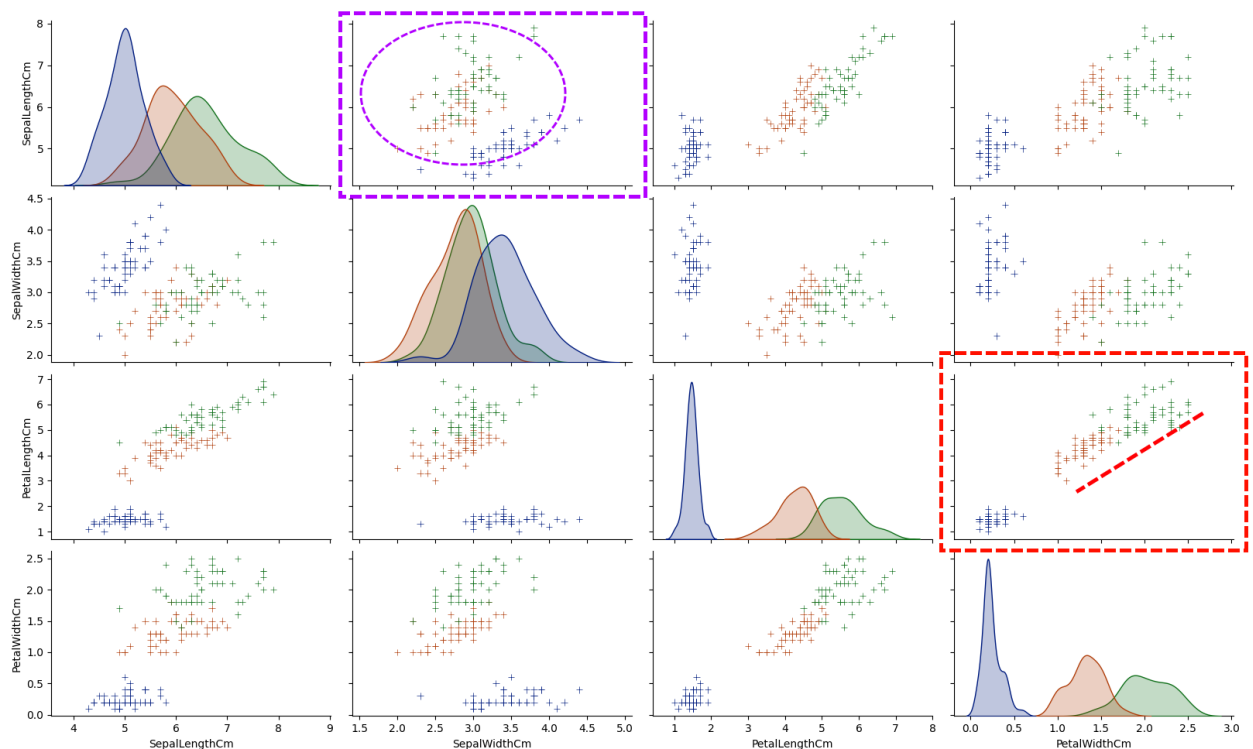


Рис. 3.2. Точкова діаграма розсіювання класів: ● – Iris-setosa, ● – Iris-versicolor, ● – Iris-virginica

З урахуванням інтерпретації точкової діаграми розсіювання встановлено, що між змінними «petal length (cm)» та «petal width (cm)» існує сильна залежність (на рис. 3.2 позначено червоною лінією). Множина точок витягнута вздовж однієї лінії, а точки згруповані, отже за цими ознаками можна будувати класифікацію. Натомість, між змінними «sepal width (cm)» та «sepal length (cm)» спостерігається слабка залежність (позначено на рис. 3.2 фіолетовим колом), об'єкти, що відносяться до Iris-versicolor та Iris-virginica,

перемішані між собою. Для більш детального аналізу необхідно застосовувати розрахунок коефіцієнтів кореляції Пірсона.

3.1.2 Налаштування параметрів функцій належності

Функції належності описують, наскільки об'єкт належить певній нечіткій множині. Зазвичай функції належності використовуються в нечітких експертних системах, контролерах та інтелектуальних системах для обробки нечіткої інформації.

У реальних даних при моделюванні часто існує невизначеність, яку можна краще описати з використанням нечітких множин. У різних галузях, зокрема в медицині, фінансах, інженерії та управлінні, функції належності використовуються для обробки невизначеності та прийняття рішень за умови неповних або нечітких даних. Функції належності дозволяють моделювати цю невизначеність і враховувати її при прийнятті рішень. Це особливо важливо в ситуаціях, коли точні обчислення неможливі або надто коштовні.

Кожна ознака об'єкту описується лінгвістичною змінною, для якої формується терм-множина та задаються границі термів.

Для відображення чітких вхідних значень ознак k_l – в нечіткі множини вводяться функції належності M_{it} виду:

$$M_{it}(k_l, a_{it}, b_{it}, c_{it}, d_{it}) = \begin{cases} 0, & k_l \leq a_{it} \\ \frac{k_l - a_{it}}{b_{it} - a_{it}}, & a_{it} \leq k_l \leq b_{it} \\ 1, & b_{it} \leq k_l \leq c_{it} \\ \frac{d_{it} - k_l}{d_{it} - c_{it}}, & c_{it} \leq k_l \leq d_{it} \\ 0, & d_{it} \leq k_l \end{cases}, \quad (3.1)$$

де M_{lt} - функції належності ознаки k_l терму t ; $a_{lt}, b_{lt}, c_{lt}, d_{lt}$ - числові параметри, які визначають границі термів, набувають дійсних значень та впорядковані відношенням $a_{lt} \leq b_{lt} \leq c_{lt} \leq d_{lt}$.

Для визначення границь термів будемо використовувати дані навчальної вибірки. Аналіз великих обсягів спостережень демонструє, що в результатах можуть бути присутніми нетипові спостереження, значення яких не можуть бути описані загальними закономірностями. Наявність таких даних може бути пов'язано з похибкою у вимірюваннях, аномаліями в розподілі даних або факторами, які не були враховані при побудові моделі. Значення, що суттєво відхиляються, можуть істотно погіршити налаштування процедури класифікації, оскільки класифікатор буде намагатися пояснити нетипові спостереження.

Для визначення границь термів із використанням даних навчальної вибірки у роботі запропоновано два варіанти, а саме за допомогою вибіркового середнього значення та міжквартильного середнього.

3.1.3 Налаштування границь термів для функції належності за допомогою вибіркового середнього значення

Вибіркове (емпіричне) середнє значення — характеристика положення для вибіркового розподілу. Вибірковим середнім називається середнє арифметичне елементів даної вибірки (avg - average):

$$avg(\bar{x}) = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n}.$$

Границі термів визначаються за формулами відносно бази даних навчальної вибірки в такий спосіб:

$$\bar{k}_t = \begin{cases} \left(\min_{i=1} (k_i), \max_{i=1} (k_i) \right), t=1, T=2 \\ \left(\min_{i=1} (k_i), \max_{i=1} (k_i) \right), t=T, T=2 \\ \left(\min_{i=1} (k_i), \min_{i=1} (k_i) + \frac{\left(\text{avg}_{i=1} (k_i) \right)}{T-2} \right), j=1, T>2 \\ \left(\min_{i=1} (k_i) + \frac{(t-1) \cdot \text{avg}_{i=1} (k_i)}{T-1}, \max_{i=1} (k_i) - \frac{(T-t) \cdot \text{avg}_{i=1} (k_i)}{T-1} \right), 1 < t < T, T>2 \\ \left(\max_{i=1} (k_i) - \frac{\left(\text{avg}_{i=1} (k_i) \right)}{T-2}, \max_{i=1} (k_i) \right), t=T, T>2 \end{cases}, \quad (3.2)$$

де $t = \overline{1, T}$, $T \geq 2$, T - кількість елементів терм-множини лінгвістичної змінної, якщо $T < 2$ - немає сенсу створювати таку лінгвістичну змінну.

3.1.4 Налаштування границь термів для функції належності за допомогою міжквартильного середнього

В роботі для обчислення границь термів пропонується використовувати квартильні оцінки, а саме міжквартильне середнє (МН - Midhinge), значення якого обчислюється за навчальною вибіркою. Застосування такого підходу робить процедуру визначення границь термів робастною.

Квартилі ділять розподіл даних на чотири рівні частини. Значення, які ділять ряд розподілу, називаються першим (Q1), другим (Q2) і третім квартилям (Q3). Значення другого квартиля відповідає значенню медіани – оскільки рівно 50% значень знаходяться до другого квартиля і 50% значень –

після другого квартиля. Перший квартиль вміщує значення, що знаходяться посередині між найменшим значенням і другим квартилем.

Квартилі є корисним інструментом для вимірювання розкиду, оскільки вони набагато менше залежать від екстремальних значень чи перекосу в наборі даних, аніж середнє та стандартне відхилення. Тому, інформацію про квартилі часто подають разом із медіаною (як міри розкиду та центральної тенденції) – у випадку роботи з масивами даних, де є суттєві перекоси або екстремальні значення. Часто квартилі подають у вигляді міжквартильного розмаху – як різницю між третім і першим квартилями.

Для визначення міжквартильного середнього, набір даних з навчальної вибірки впорядковується та ділиться на чотири частини, потім обчислюється середнє значення між першим і третім квартилями [132]:

$$\begin{aligned}
 MH &= \frac{Q1 + Q3}{2}, \\
 Q1 &= x_{\min Q1} + \Delta x_{Q1} \frac{\frac{N}{4} - S_{Q1-1}}{n_{Q1}}, \\
 Q3 &= x_{\min Q3} + \Delta x_{Q3} \frac{\frac{3N}{4} - S_{Q3-1}}{n_{Q3}},
 \end{aligned} \tag{3.3}$$

де $Q1, Q3$ – медіани h найменших і h найбільших значень відповідно; $x_{\min Q1}$, $x_{\min Q3}$ – нижні границі інтервалів, що містять перший і третій квартиль; $\Delta x_{Q1}, \Delta x_{Q3}$ – ширина інтервалів; S_{Q1-1}, S_{Q3-1} – накопичена частота інтервалу, що передує даному – може бути визначена, як сума всіх попередніх частот до поточної; n_{Q1}, n_{Q3} – частота попадання значень навчальної вибірки в інтервали, що містять перший і третій квартилі відповідно.

Границі термів визначаються відносно елементів навчальної вибірки у такий спосіб:

$$\bar{k}_t = \begin{cases} \left(\min_{i=1} (k_i), \max_{i=1} (k_i) \right), t=1, T=2 \\ \left(\min_{i=1} (k_i), \max_{i=1} (k_i) \right), t=T, T=2 \\ \left(\min_{i=1} (k_i), \min_{i=1} (k_i) + 2 \cdot \frac{\left| \text{МН}(k_i) - \min_{i=1} (k_i) \right|}{T+1} \right), t=1, T>2 \\ \left(\min_{i=1} (k_i) + \frac{(t-1) \cdot 2 \cdot \left| \text{МН}(k_i) - \min_{i=1} (k_i) \right|}{T+1}, \max_{i=1} (k_i) - \frac{(T-t) \cdot 2 \cdot \left| \text{МН}(k_i) - \max_{i=1} (k_i) \right|}{T+1} \right), 1 < t < T, T > 2 \\ \left(\max_{i=1} (k_i) - 2 \cdot \frac{\left| \text{МН}(k_i) - \max_{i=1} (k_i) \right|}{T+1}, \max_{i=1} (k_i) \right), t=T, T > 2 \end{cases}, \quad (3.4)$$

де МН - міжквартильне середнє; $t = \overline{1, T}$, $T \geq 2$, T – кількість елементів терм-множини лінгвістичної змінної.

Розглянемо результат побудови границь термів для набору Іриси Фішера (рис. 3.3). Для кожної ознаки об'єкту `sepalLength`, `sepalWidth`, `petalLength`, `petalWidth` вводяться лінгвістичні змінні та визначаються по три терм-множини {"Low", "Mid", "High"}. В результаті роботи алгоритму автоматично визначено границі термів на основі значень ознак об'єктів з навчальної вибірки.

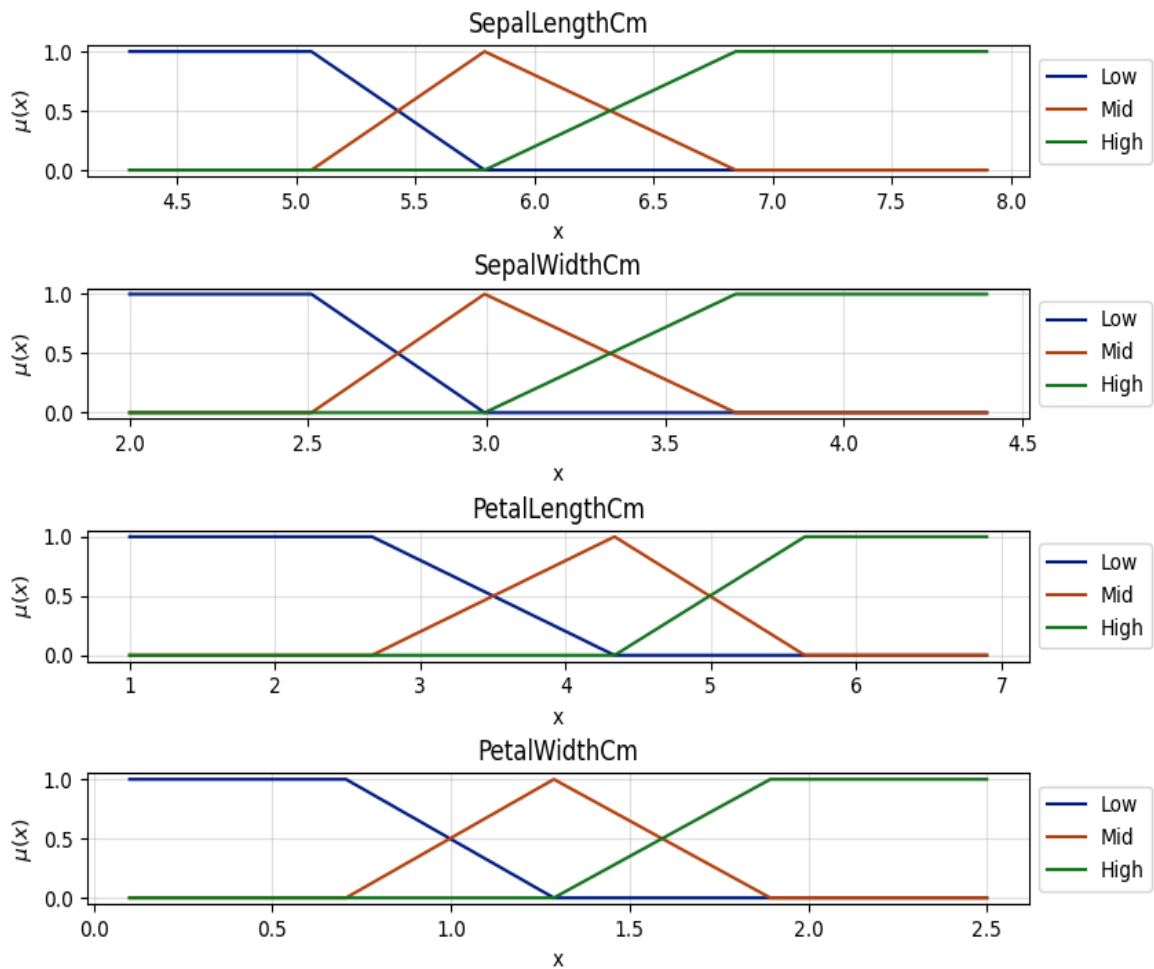


Рис. 3.3. Границі термів для набору Іриси Фішера при $T = 3$.

Далі на рис. 3.4 та рис. 3.5 наведено результати розрахунку границь, коли кількість термів для визначення ознаки дорівнює п'яти { "LLow", "Low", "Mid", "High", "HHigh" } та шістьом термам { "Very Very Low", "Very Low", "Low", "Mid", "Very High", "Very Very High" }.

Незалежно від кількості заданих користувачем термів, що обираються для опису лінгвістичної змінної, запропонований алгоритм дозволяє налаштувати границі термів таким чином, що терми рівномірно розподіляють весь діапазон припустимих значень кожної лінгвістичної змінної, терми йдуть послідовно відносно лінгвістичного порядку назв термів.

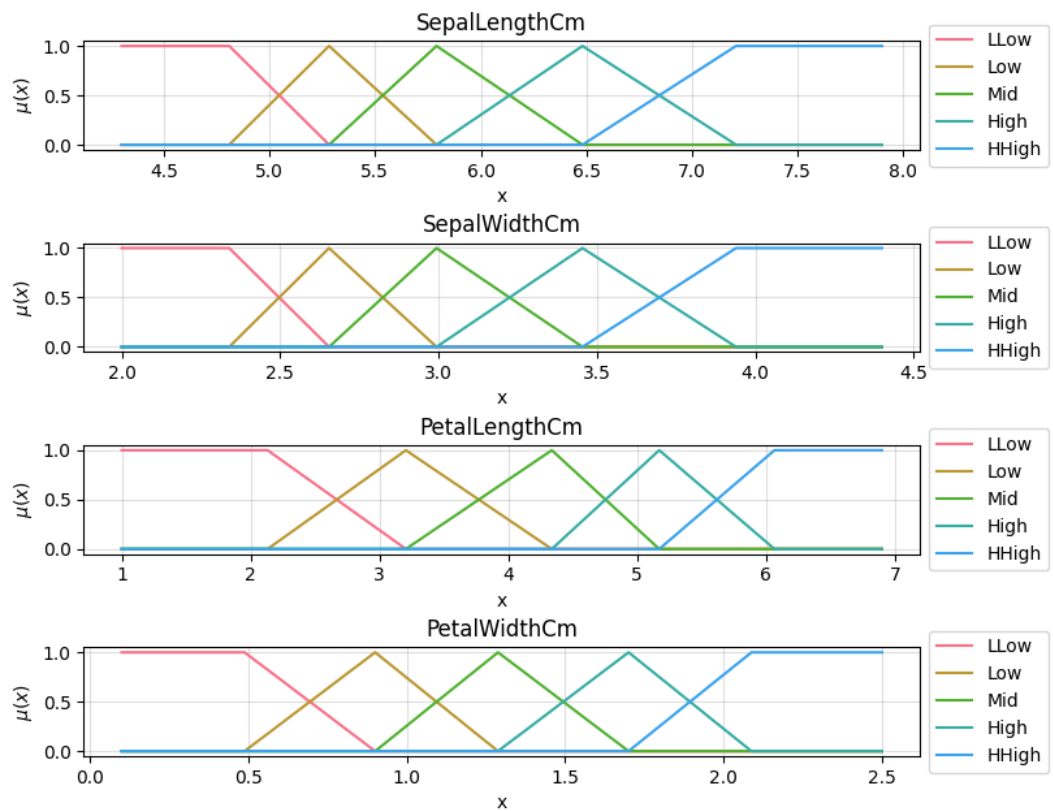


Рис. 3.4. Границі термів для набору Іриси Фішера при $T = 5$.

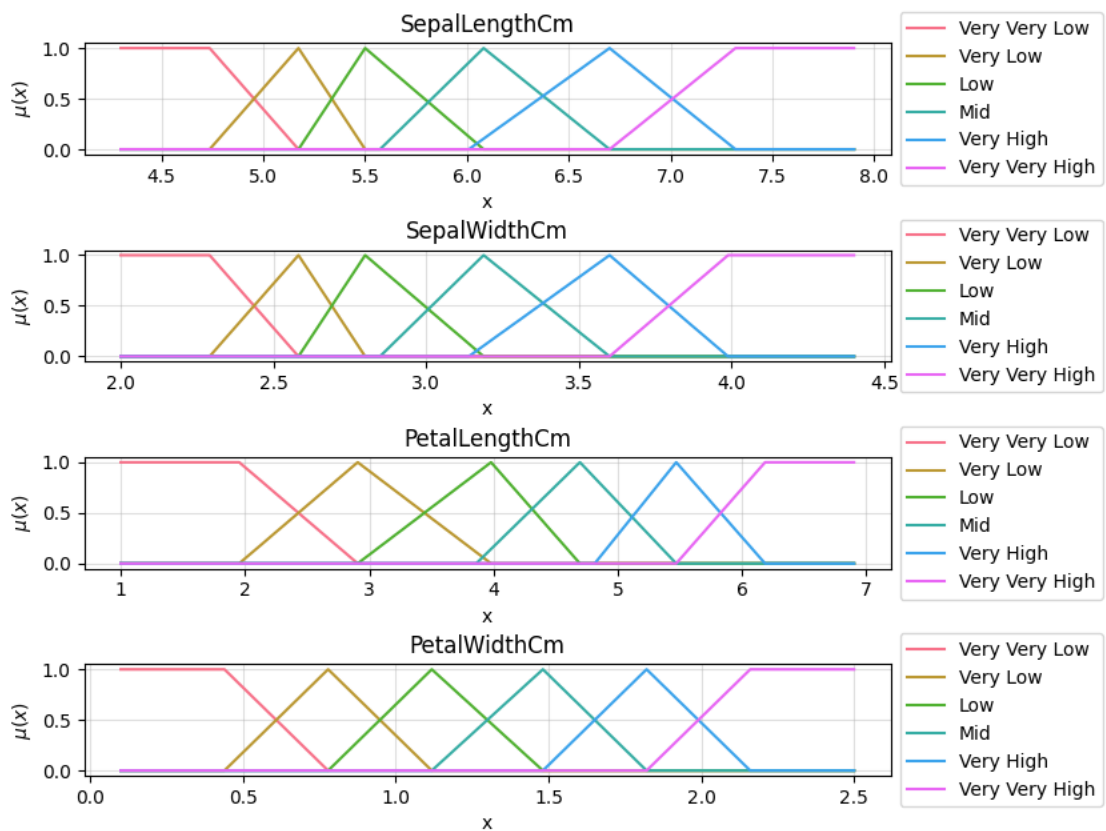


Рис. 3.5. Границі термів для набору Іриси Фішера при $T = 6$.

Отже, налаштування границь термів є важливою процедурою, що дозволяє адаптувати нечітку експертну систему до конкретних умов та вимог задачі. Правильно налаштовані границі термів покращують продуктивність, інтерпретованість та точність системи.

Реалізовані вище методи налаштування границь термів за допомогою вибіркового середнього та міжквартильного середнього дозволяють обчислити границі термів автоматично на етапі навчання за допомогою навчальної вибірки та зазначеної кількості термів, що задається користувачем для лінгвістичної змінної.

3.2 Аналіз коректності бази правил експертної системи

Однієї з проблем, які виникають при побудові нечіткої моделі, є обґрунтування коректності бази правил. Аналіз продукційних правил бази знань на коректність виконується з урахуванням розробленої вербальної моделі, тому формалізована модель у вигляді бази знань, має задовольняти певним формальним вимогам, які стосуються смислового аспекту правил [7].

База знань повинна задовольняти єдиним формальним вимогам коректності, що не стосуються смислового аспекту продукційних правил, а саме, бути повною, мінімальною, зв'язною та несуперечливою [28]. Під повнотою розуміється, що будь-якому вхідному поєднанню значень термів вхідних лінгвістичних змінних відповідає певне правило в базі знань, тобто немає такої ситуації, коли система логічного виведення не може знайти правила для вхідного набору лінгвістичних змінних.

Мінімальною вважається база знань, з якої неможливо видалити жодного з продукційних правил, не порушивши цим самим її повноту.

База знань є несуперечливою (узгодженою), якщо вона не містить несумісних правил, тобто не містить правил з однаковими лінгвістичними умовами, але різними висновками.

База знань є зв'язною, якщо для будь-якої пари суміжних правил значення вихідної лінгвістичної змінної теж суміжні, тобто, якщо умови цих правил відрізняються лише однією з n підумов, то терми для вихідної змінної мають бути суміжними.

Під час автоматичного формування бази правил з використанням декартового добутку, враховуючи властивості декартового добутку, утворюється база правил, що є мінімальною та несуперечною.

Зв'язність бази правил забезпечується відносно вектора консеквентів \vec{C} , побудованого у процесі навчання на основі навчальної вибірки. Оскільки на початку формування матриці A терм-множина $A_l = (A_{l1}, A_{l2}, A_{l3}, \dots, A_{ln})$ впорядкована відносно семантичного порядку та границь термів і кожен рядок антецеденту матриці A відрізняється лише однією з l підумов, то терм-множини висновків таких правил є суміжними.

Для автоматичного доведення повноти бази правил в роботі застосуємо логіку Хоара [77].

Логіка Хоара, як формальна система з набором логічних правил, призначалася саме для доведення коректності програм. Зазначена логіка намагається формалізувати та довести коректність програм шляхом визначення передумов та післяумов для кожної команди. Доведення будується на представленні предикатів у вигляді станів програми і команд та дозволяє проводити верифікацію програм для попередження помилок.

Основною складовою логіки Хоара є трійка Хоара, що визначає, як саме виконання фрагмента коду змінює стан результату обчислення.

Експертна система має дуже схожу реалізацію, оскільки база правил в процесі логічного виведення виконує команди та змінює результат класифікації. Тому в роботі для автоматичного доведення повноти використовується логіка Хоара, а предикат передумови та предикат післяумови визначаються у вигляді кон'юнктивних та диз'юнктивних нормальних форм.

Визначимо трійку Хоара:

$$\{Q\} S \{R\},$$

де Q – предикат передумови; S – набір команд; R – предикат післяумови

Множину продукційних правил $\Pi_p = \langle a_p, c_p \rangle$ зобразимо набором команд S в такий спосіб:

$$S: \langle A, \vec{C} \rangle.$$

Передумовою Q описуються всі можливі початкові значення, яких можуть набувати лінгвістичні змінні для замикання множини всіх вхідних значень:

$$Q: \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right) \wedge \bigvee_{m=1}^M result = C_m$$

Предикат Q є істинним, оскільки кожна з ознак k_l об'єкту належить хоча б одному з термів A_{lt} .

Постумова R описує очікуваний результат виконання набору команд S у такий спосіб:

$$R: \bigvee_{p=1}^P \left(\left(\bigwedge_{l=1}^L k_l \in a_{pl} \right) \wedge result = c_p \right).$$

Якщо передумова Q виконується, то набір команд S робить вірною постумову R [66].

Це твердження доводиться з використанням предикату найслабшої передумови $WP(S, R)$ [68].

$$Q \Rightarrow WP(S, R)$$

Предикат WP виконує підстановку $\langle A, \vec{C} \rangle$ з набору команд S в постумову R .

$$Q \Rightarrow R_s \tag{3.5}$$

Для доведення істинності предикату (3.5) скористаємося методом резолюцій:

$$Q \models R_s,$$

$$\begin{aligned}
& \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right) \wedge \bigvee_{m=1}^M result = C_m \models \bigvee_{p=1}^P \left(\left(\bigwedge_{l=1}^L k_l \in a_{pl} \right) \wedge result = c_p \right) \Big|_S, \\
& \left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigvee_{m=1}^M result = C_m, \neg \bigvee_{p=1}^P \left(\left(\bigwedge_{l=1}^L k_l \in a_{pl} \right) \wedge result = c_p \right) \Big|_S \right\}, \\
& \left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigvee_{m=1}^M result = C_m, \neg \left(\bigvee_{p=1}^P \left(\bigwedge_{l=1}^L k_l \in a_{pl} \right) \Big|_S \wedge \bigvee_{p=1}^P (result = c_p) \Big|_S \right) \right\}, \\
& \left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigvee_{m=1}^M result = C_m, \bigwedge_{p=1}^P \left(\bigvee_{l=1}^L \neg(k_l \in a_{pl}) \right) \Big|_S \vee \neg \left(\bigvee_{p=1}^P result = c_p \right) \Big|_S \right\},
\end{aligned}$$

Оскільки елементи вектору консеквентів c_p приймають значення з множини класів C_m , після підстановки з набору команд S, отримуємо:

$$\left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigvee_{m=1}^M result = C_m, \bigwedge_{p=1}^P \left(\bigvee_{l=1}^L \neg(k_l \in a_{pl}) \right) \Big|_S \vee \neg \left(\bigvee_{m=1}^M result = C_m \right) \right\},$$

Далі правило резолюції застосовується до другого та третього диз'юнктивів:

$$\left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigwedge_{p=1}^P \left(\bigvee_{l=1}^L \neg(k_l \in a_{pl}) \right) \Big|_S \right\},$$

Враховуючи (2.3), отримаємо:

$$\left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigwedge_{p=1}^P \left(\bigvee_{l=1}^L \neg(k_l \in a_{pl}) \right) \Big|_S \right\},$$

Застосуємо закон дистрибутивності до другого диз'юнкту та здійснимо підстановку з набору команд S. Оскільки елементи a_{pl} матриці антецедентів A приймають значення з терм-множини A_{lt} , після підстановки з набору команд S, отримуємо:

$$\left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \bigvee_{l=1}^L \left(\bigwedge_{t=1}^{T_l} \neg(k_l \in A_{lt}) \right) \right\},$$

$$\left\{ \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} k_l \in A_{lt} \right), \neg \left(\bigwedge_{l=1}^L \bigvee_{t=1}^{T_l} (k_l \in A_{lt}) \right) \right\}.$$

В результаті застосування правила резолюцій до першого та другого диз'юнктивів отримаємо:

$$\{ _ \}$$

Ланцюг здійснених перетворень доводить повноту передумов за допомогою предикату WP.

В такий спосіб було доведено, що відносно специфікації програми, тобто за відносно предикату передумови, набору команд та предикату післяумови, база правил експертної системи задовольняє формальним вимогам повноти. Таким чином повноту бази правил доведено.

3.3 Програмна реалізація розробленого підходу доведення повноти бази правил

Для застосування апарату автоматичного доведення повноти бази правил будемо використовувати програмне забезпечення **Simplify** [67], у якому для доведення істинності заданих предикатів виходячи з логіки першого порядку реалізований метод резолюцій.

Формальне обчислення допускає висловлювання відносно змінних, фіксованих функцій та предикатів, що у свою чергу розширює логіку висловлювань. На відміну від інших подібних засобів Simplify ніколи не утворює нескінченний цикл у процесі доведення і вважає, що предикати, доведення істинності яких є неможливим, є хибними.

Таким чином, істотна частина розв'язку задачі полягає в тому, щоб переконатися, що Simplify має всі необхідні «знання» для доведення повноти бази правил [42].

Для доведення повноти сформулюємо предикати Q, S, R з використанням директив BG_PUSH, FORALL, та логічних операторів AND, OR, EQ, IMPLIES мови Simplify проведемо автоматичне доведення.

Аксиома Simplify для першого кон'юнкту передумови Q, матиме вигляд:

```
(BG_PUSH
  (FORALL (k_i ai1 ai2 ai3 ... ait)
    (OR
      (EQ k_i ai1)
      (EQ k_i ai2)
      (EQ k_i ai3)
      ;...
      (EQ k_i ait)
    )
  )
)
```

Аксиому для другого кон'юнкту передумови Q представимо у вигляді:

```
(BG_PUSH
  (FORALL (result c1 c2 c3 ... cm)
    (OR
      (EQ result c1)
      (EQ result c2)
      (EQ result c3)
      ;...
      (EQ result cm)
    )
  )
)
```

)

Передумова Q матиме вигляд:

(AND

(AND

(OR (EQ k_1 a_11) (EQ k_1 a_12) (EQ k_1 a_13) ... (EQ k_1 a_1t))

(OR (EQ k_2 a_21) (EQ k_2 a_22) (EQ k_2 a_23) ... (EQ k_2 a_2t))

(OR (EQ k_3 a_31) (EQ k_3 a_32) (EQ k_3 a_33) ... (EQ k_3 a_3t))

...

(OR

(EQ k_9 a_91) (EQ k_9 a_92) (EQ k_9 a_93)

(EQ k_9 a_94) (EQ k_9 a_95) (EQ k_9 a_96)

(EQ k_9 a_97) (EQ k_9 a_98) (EQ k_9 a_99)

...

(EQ a_9 a_9t)

)

...

(OR

(EQ k_L a_L1) (EQ k_L a_L2) (EQ k_L a_L3)

(EQ k_L a_L4) (EQ k_L a_L5) (EQ k_L a_L6)

(EQ k_L a_L7) (EQ k_L a_L8) (EQ k_L a_L9)

...

(EQ k_L a_Lt)

)

)

(OR

(EQ result c1)

(EQ result c2)

(EQ result c3)

...

(EQ result cm)

)

)

)

Післяумову R визначимо у вигляді:

```
(OR
  (AND (EQ k_1 a_11)(EQ k_2 a_21)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
  (AND (EQ k_1 a_12)(EQ k_2 a_21)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
  (AND (EQ k_1 a_11)(EQ k_2 a_22)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
  ...
  (AND (EQ k_1 a_1t)(EQ k_2 a_2t)(EQ k_3 a_3t)...(EQ k_L a_Lt) (EQ result cm))
)
```

$Q \Rightarrow WP(S, R)$.

```
(IMPLIES
  ;Q
  (AND
    (AND
      (OR (EQ k_1 a_11) (EQ k_1 a_12) (EQ k_1 a_13) ... (EQ k_1 a_1t))
      (OR (EQ k_2 a_21) (EQ k_2 a_22) (EQ k_2 a_23) ... (EQ k_2 a_2t))
      (OR (EQ k_3 a_31) (EQ k_3 a_32) (EQ k_3 a_33) ... (EQ k_3 a_3t))
      ...
      (OR
        (EQ k_L a_L1) (EQ k_L a_L2) (EQ k_L a_L3)
        (EQ k_L a_L4) (EQ k_L a_L5) (EQ k_L a_L6)
        (EQ k_L a_L7) (EQ k_L a_L8) (EQ k_L a_L9)
        ...
        (EQ k_L a_Lt)
      )
    )
  )
  (OR (EQ result c1) (EQ result c2) (EQ result c3) ... (EQ result cm))
)
;R_S
(OR
  (AND (EQ k_1 a_11)(EQ k_2 a_21)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
  (AND (EQ k_1 a_12)(EQ k_2 a_21)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
  (AND (EQ k_1 a_11)(EQ k_2 a_22)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
  ...
  (AND (EQ k_1 a_1t)(EQ k_2 a_2t)(EQ k_3 a_3t)...(EQ k_L a_Lt) (EQ result cm))
)
)
```

З використанням введених аксіом Simplify та предикатів Q, S, R – організовано автоматичне доведення повноти бази знань. Після чого

поєднуємо аксіоми, предикати та предикат найслабшої постумови WP в один текстовий файл, та передаємо цей файл на вхід до аргументів програмного забезпечення Simplify: *Simplify-1.5.4.exe DatabaseWP.simp*

На рис. 3.6 наведено результат перевірки бази знань на повноту за допомогою предикату найслабшої постумови WP, та програми Simplify.

```

...
(OR
(EQ k_L a_L1) (EQ k_L a_L2) (EQ k_L a_L3)
(EQ k_L a_L4) (EQ k_L a_L5) (EQ k_L a_L6)
(EQ k_L a_L7) (EQ k_L a_L8) (EQ k_L a_L9)
...
(EQ k_L a_Lt)
)
)
(OR
(EQ result c1)
(EQ result c2)
(EQ result c3)
...
(EQ result cm)
)
)
;R_5
(OR
(AND (EQ k_1 a_11)(EQ k_2 a_21)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
(AND (EQ k_1 a_12)(EQ k_2 a_21)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
(AND (EQ k_1 a_11)(EQ k_2 a_22)(EQ k_3 a_31)...(EQ k_L a_Lt) (EQ result c1))
...
(AND (EQ k_1 a_1t)(EQ k_2 a_2t)(EQ k_3 a_3t)...(EQ k_L a_Lt) (EQ result cm))
)
)
1: Valid.

```

Рис. 3.6. Результат перевірки повноти, «Valid» – істина

Отримано результат «Valid» (істина), що підтверджує повноту бази знань відносно предикату найслабшої постумови WP. Програмне забезпечення Simplify поступово проводить аналіз аксіом та предикатів після чого починає доводити істинність предикату найслабшої постумови WP, якщо доведення істинності предикату можливо, то Simplify поверне значення «Valid», в іншому випадку «Invalid» та контрприклад, за допомогою якого можна проаналізувати доведення та знайти помилку у базі знань.

Таким чином, за допомогою Simplify було автоматично доведено повноту бази знань. Використовуючи програмне забезпечення Simplify можна проводити перевірку баз правил експертних систем на повноту, а генерацію аксіом, предикатів та предикату найслабшої постумови WP реалізувати автоматично в процесі побудови бази знань, використовуючи директив BG_PUSH, FORALL та логічні оператори AND, OR, EQ, IMPLIES мови Simplify.

3.4 Аналіз якості роботи експертної системи

Для подальшої оцінки якості моделі та результатів класифікації, важливо провести порівняння висновків, отриманих за допомогою розробленої експертної системи, з тренувальними даними або фактичними результатами спостережень за реальними процесами. Цей порівняльний аналіз дозволить визначити, наскільки успішно модель може адаптуватися до нових даних та наскільки ефективно вона розрізняє класи об'єктів.

Окрім метрик accuracy, precision, recall та f1-score, що було введено у розділі 2.7, в роботі для оцінки якості результату класифікації пропонується використання матриць невідповідностей.

Матриця невідповідностей (або матриця помилок) дозволяє визначити кількість правильно класифікованих об'єктів та кількість помилково класифікованих об'єктів для кожного класу. Це надає деталізовану інформацію щодо ефективності моделі для кожного окремого класу та загальної ефективності класифікації в цілому.

Для проведення класифікації формуються навчальна та тестова вибірки. Зазвичай розмір навчальної вибірки повинен бути більший ніж розмір тестової вибірки для якісного навчання моделі. Після навчання системи за допомогою навчальної вибірки, проводиться порівняльний аналіз кількості правильно класифікованих об'єктів та кількості помилково класифікованих об'єктів з тестової вибірки. Для цього необхідно побудувати матрицю невідповідностей.

На рис. 3.7 зображено загальне представлення матриці невідповідностей. Що дозволяє наочно визначити кількість правильно та помилково класифікованих об'єктів. На рис. 3.7 зображено 2 класи Positive та Negative та прогнозовані системою класи, після чого у зелених клітинках рахується кількість правильно класифікованих об'єктів Positive та Negative відповідно, а у червоних клітинках кількість тих, що помилково було віднесено до класів Positive та Negative.

		Прогнозований клас	
Загальна кількість $= P + N$		Predicted Positive (PP)	Predicted Negative (PN)
Справжній клас	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

Рис. 3.7. Матриця невідповідностей

Матриця невідповідностей допомагає ідентифікувати, в яких випадках модель схильна до помилок, та які саме помилкові класи було обрано в процесі класифікації. Вона є корисним інструментом для подальшого вдосконалення моделі та роботи над виправленням помилок, що можуть виникати під час класифікації.

3.5 Висновки до розділу

В роботі запропоновано здійснювати попередню обробку експериментальних даних шляхом побудови діаграми розсіювання. Аналіз точкової діаграми розсіювання дозволяє виявити залежності між ознаками, якщо існує сильна залежність, то дані на діаграмі згруповані, і за такими ознаками можна будувати класифікацію, якщо ж залежність слабка, дані перемішані між собою, то будувати класифікацію за цими ознаками не бажано, при цьому включення таких ознак у модель класифікації бути надлишковим та не призведе до покращення результатів класифікації.

В роботі запропоновано процедуру автоматичного визначення границь термів за допомогою статистичних методів з використанням вибіркового та міжквартильного середніх значень, що зменшує вплив наявних у навчальній вибірці аномальних об'єктів на результат класифікації та робить процедуру визначення границь термів робастною. Автоматичне налаштування границь

термів дозволяє користувачеві вказати кількість термів для лінгвістичної змінної після чого, система автоматично розраховує границі термів виходячи зі значень ознак об'єктів з навчальної вибірки. Налаштування границь термів підвищує точність висновків, враховує контекст та зміни у вхідних даних, а також знижує ймовірність помилок.

Здійснено аналіз коректності бази правил враховуючи її повноту, несуперечливість, зв'язаність та мінімальність. Забезпечення цих аспектів перевірки бази правил є обов'язковим для забезпечення надійності та коректності експертної системи на основі нечітких знань.

В роботі запропоновано автоматичне проведення перевірки повноти бази правил з використанням логіки Хоара та предикату найслабшої передумови $Q \Rightarrow WP(S, R)$. Доведення проведено з використанням методу резолюцій та додатку Simplify, що забезпечує автоматизацію перевірки коректності бази правил на етапі навчання.

Для оцінки якості результатів класифікації розробленої моделі, запропоновано використання матриць невідповідностей, за допомогою яких здійснюється порівняльний аналіз кількості правильно класифікованих об'єктів та кількості помилково класифікованих об'єктів з тестової вибірки. Цей аналіз дозволяє визначити, наскільки ефективно модель адаптується до нових даних та розрізняє класи об'єктів. Додатково до стандартних метрик, таких як accuracy, precision, recall та f1-score, використання матриць невідповідностей дозволяє отримати детальну інформацію щодо ефективності класифікації для кожного окремого класу, а також результат загальної ефективності всієї системи.

Основні результати розділу опубліковані в [14, 17, 19, 20, 134].

Розділ 4. ЗАСТОСУВАННЯ ЕКСПЕРТНОЇ СИСТЕМИ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ КЛАСИФІКАЦІЇ

4.1 Налаштування та навчання нечіткої моделі для задачі класифікації

Запропонований підхід був застосований для налаштування бази знань для розв'язання задач мультикласової класифікації на відомих модельних наборах даних Іриси Фішера (Iris Data Set) та видова популяція арктичних пінгвінів.

4.1.1 Налаштування нечіткої моделі для класифікації Ірисів Фішера

Запропонований підхід до побудови нечіткої бази правил для класифікації об'єктів за допомогою нечіткої логіки та елементів навчальної вибірки, був застосований для розв'язання відомої задачі мультикласової класифікації Іриси Фішера (Iris Data Set) [73]. Задача класифікації передбачає визначення належності ірису до одного з трьох типів рослин: Setosa, Versicolor, Virginica. У таблиці 4.1 наведено загальну інформацію про дані з набору.

Усього в наборі $X = \{(x_i, C_m)\}$ міститься 150 об'єктів x_i , кожному з трьох класів $C_m \in C$ відповідає по 50 об'єктів з набору даних. В якості ознак об'єкту використовуються: довжина чашолистка – sepalLength, ширина чашолистка – sepalWidth, довжина пелюстки – petalLength, ширина пелюстки – petalWidth. Для проведення класифікації були сформовані навчальна та тестова вибірки у розмірі 130/20 (навчальна/тестова).

Таблиця 4.1 - Data Set - Іриси Фішера

Id	SepalLength (cm)	SepalWidth (cm)	PetalLength (cm)	PetalWidth (cm)	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3.0	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
...
51	7.0	3.2	4.7	1.4	Iris-versicolor
52	6.4	3.2	4.5	1.5	Iris-versicolor
53	6.9	3.1	4.9	1.5	Iris-versicolor
...
101	6.3	3.3	6.0	2.5	Iris-virginica
102	5.8	2.7	5.1	1.9	Iris-virginica
103	7.1	3.0	5.9	2.1	Iris-virginica

Після чого розраховано частотний розподіл для кожної ознаки об'єкту: довжина чашолистка – sepalLength, ширина чашолистка – sepalWidth, довжина пелюстки – petalLength, ширина пелюстки – petalWidth на Рис. 4.1(а,б,в,г).

Length of the sepal (in cm)

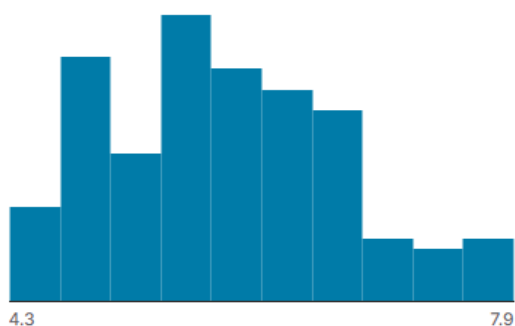


Рис. 4.1а. Розподіл довжини чашолистка – sepalLength

Width of the sepal (in cm)

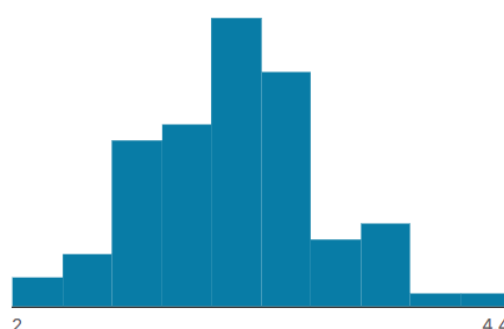


Рис. 4.1б. Розподіл ширини чашолистка – sepalWidth

Length of the petal (in cm)

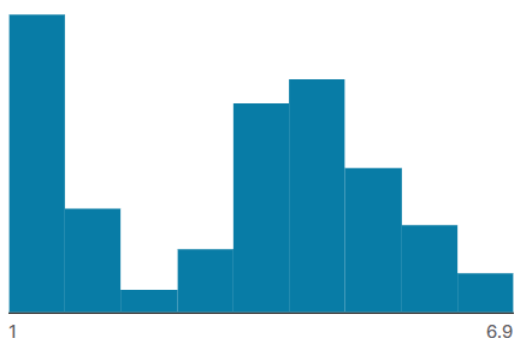


Рис. 4.1в. Розподіл довжини пелюстки – petalLength

Width of the petal (in cm)

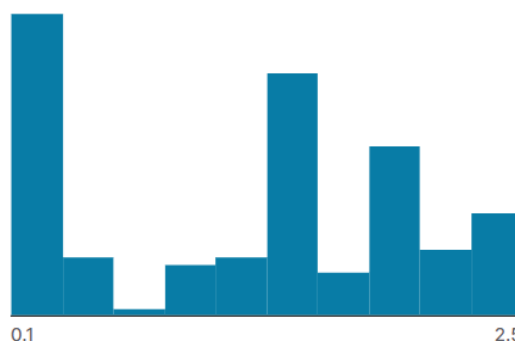


Рис. 4.1г. Розподіл ширини пелюстки – petalWidth

Далі обраховано характеристики набору для побудови границь термів у таблиці 4.2.

Таблиця 4.2 – Характеристики Data Set - Іриси Фішера

Ознака	Середнє	Стандартне відхилення	Квантилі
sepalLength	5.84	0.83	4.3 - Min 5.1 - 25% 5.8 - 50% 6.4 - 75% 7.9 – Max
sepalWidth	3.05	0.43	2 - Min 2.8 - 25% 3 - 50% 3.3 - 75% 4.4 – Max
petalLength	3.76	1.76	1 - Min 1.6 - 25% 4.4 - 50% 5.1 - 75% 6.9 – Max
petalWidth	1.2	0.76	0.1 - Min 0.3 - 25% 1.3 - 50% 1.8 - 75% 2.5 - Max

Для лінгвістичних змінних, які описують кожну з ознак, вводяться терми {"Low", "Mid", "High"}. Для кожної вхідної та вихідної змінної вводяться функції належності виду (3.1), обчислюються границі термів за формулами (3.2) та (3.4). Розглянемо результат побудови границі термів для набору Іриси Фішера на Рис. 4.2. за формулою (3.4) за допомогою міжквартильного середнього.

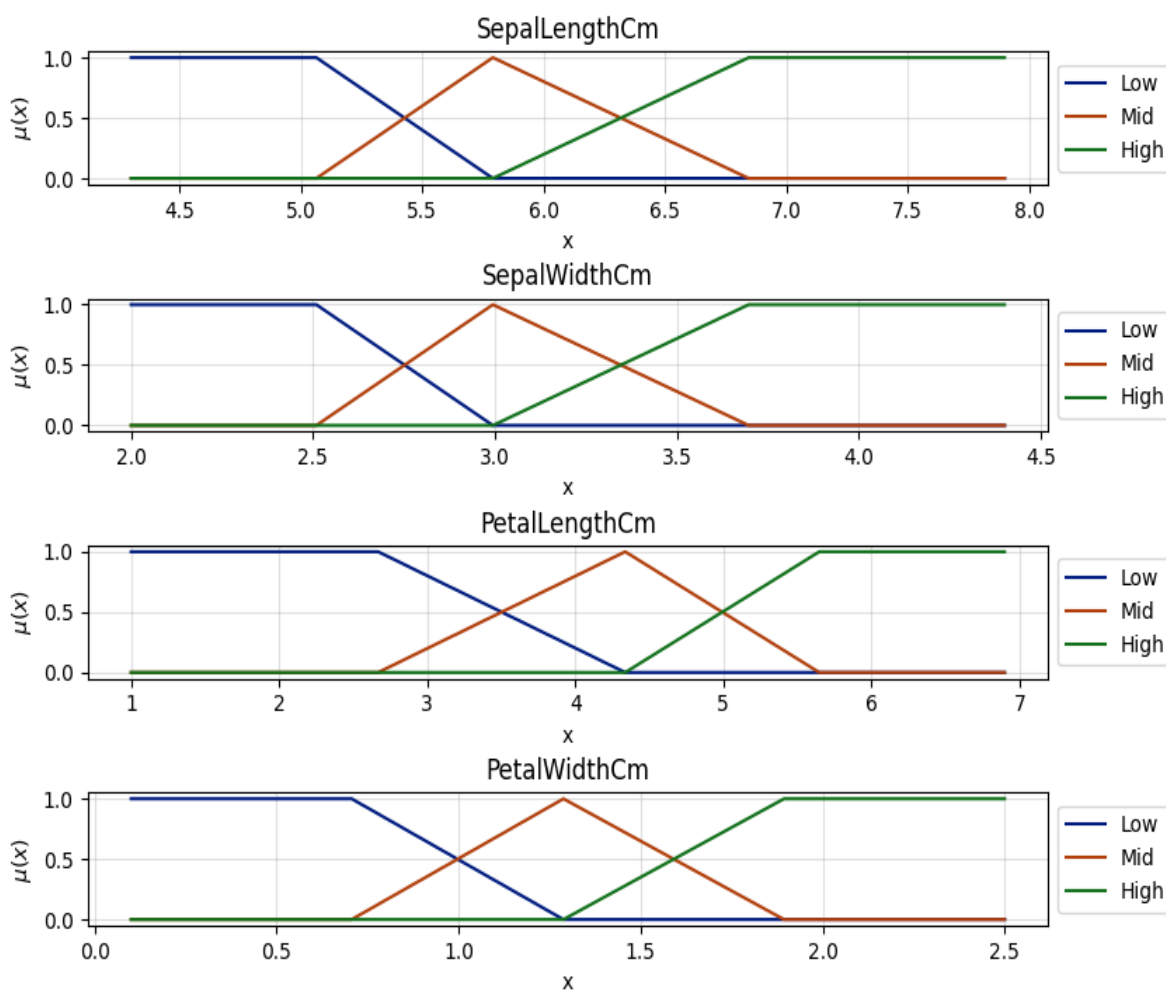


Рис. 4.2. Границі термів для набору Іриси Фішера.

Далі формується система правил (2.1) за допомогою матричного представлення, передумови яких складені з усіх можливих комбінацій значень нечітких вхідних змінних (всього 81 правило).

4.1.2 Налаштування нечіткої моделі для класифікації видової популяції арктичних пінгвінів

Для аналізу результату роботи алгоритму також використовується набір видової популяції арктичних пінгвінів, даний набір отримано на основі реальних даних Palmer Station Antarctica LTER – описаних Крістін Горман [75]. Також варто зауважити, що даний набір даних є відкритим та випущений під ліцензією Creative Commons CC0 1.0.

Набір даних $X = \{(x_i, C_m)\}$ містить дані про 344 особини трьох видів Chinstrap (68), Adelie(152), Gentoo (124). Реальні дані зібраних з трьох островів архіпелагу Палмера в Антарктиді. Даний набір є складнішою альтернативою набору Iris Data Set – Іриси Фішера [73], оскільки містить у собі більше даних.

Ознаки особин для класифікації: bill_length_mm – довжина дзьоба; bill_depth_mm – глибина дзьоба; flipper_length_mm - довжина плавця; body_mass_g – маса тіла. У таблиці 4.3 наведено загальну інформацію про дані з набору.

Таблиця 4.3 - Data Set - видова популяція арктичних пінгвінів

Id	Bill Length (mm)	Bill Depth (mm)	Flipper Length (mm)	Body Mass (g)	Species
N1A1	39.1	18.7	181	3750	Adelie Penguin (Pygoscelis adeliae)
N1A2	39.5	17.4	186	3800	Adelie Penguin (Pygoscelis adeliae)
N2A1	40.3	18	195	3250	Adelie Penguin (Pygoscelis adeliae)
...
N61A1	46.5	17.9	192	3500	Chinstrap penguin (Pygoscelis antarctica)
N61A2	50	19.5	196	3900	Chinstrap penguin (Pygoscelis antarctica)
N62A1	51.3	19.2	193	3650	Chinstrap penguin (Pygoscelis antarctica)
...
N31A1	46.1	13.2	211	4500	Gentoo penguin (Pygoscelis

					papua)
N31A2	50	16.3	230	5700	Gentoo penguin (Pygoscelis papua)
N32A1	48.7	14.1	210	4450	Gentoo penguin (Pygoscelis papua)

Після чого розраховано частотний розподіл для кожної ознаки об'єкту:
 bill_length_mm – довжина дзьоба, bill_depth_mm – глибина дзьоба,
 flipper_length_mm – довжина плавця, body_mass_g – маса тіла на
 Рис. 4.3(а,б,в,г).

Bill Length (mm)

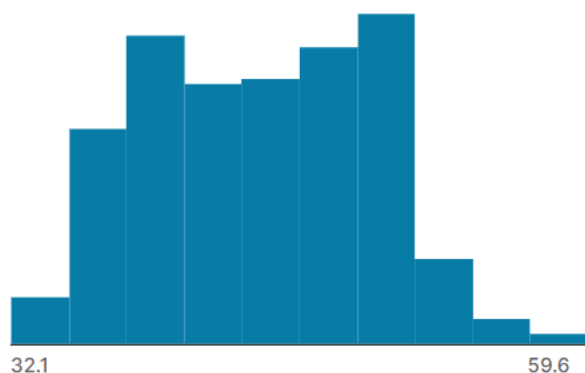


Рис. 4.3а. Розподіл довжини дзьоба – bill_length_mm

Bill Depth (mm)

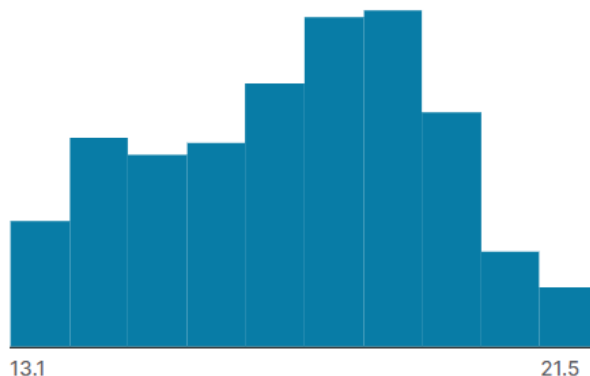


Рис. 4.3б. Розподіл глибини дзьоба – bill_depth_mm

Flipper Length (mm)

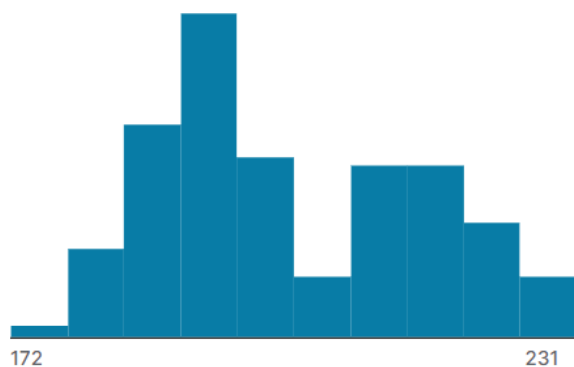


Рис. 4.3в. Розподіл довжини плавця – flipper_length_mm

Body Mass (g)

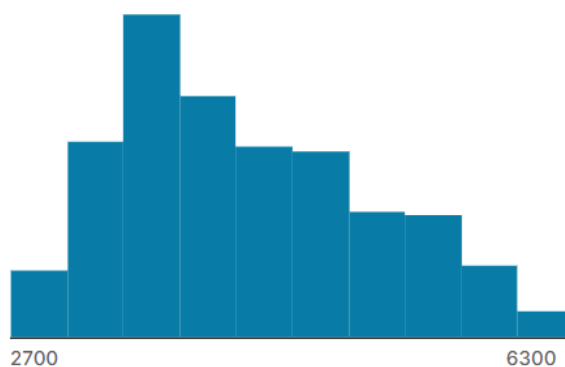


Рис. 4.3г. Розподіл маси тіла – body_mass_g

Далі обраховано характеристики набору для побудови границь термів у таблиці 4.4.

Таблиця 4.4 – Характеристики Data Set - видова популяція арктичних пінгвінів.

Ознака	Середнє	Стандартне відхилення	Квантилі
bill_length_mm	43.9	5.45	32.1 - Min 39.2 - 25% 44.5 - 50% 48.5 - 75% 59.6 - Max
bill_depth_mm	17.2	1.97	13.1 - Min 15.6 - 25% 17.3 - 50% 18.7 - 75% 21.5 - Max
flipper_length_mm	201	14	172 - Min 190 - 25% 197 - 50% 213 - 75% 231 - Max
body_mass_g	4200	801	2700 - Min 3550 - 25% 4050 - 50% 4750 - 75% 6300 - Max

Для лінгвістичних змінних, що описують кожну з ознак, вводяться терми {"Low", "Mid", "High"}. Для кожної вхідної та вихідної змінної вводяться функції належності виду (3.1), обчислюються границі термів за формулами (3.2) та (3.4). Розглянемо результат побудови границь термів за допомогою міжквартильного середнього для набору даних видової популяції арктичних пінгвінів (Рис. 4.4).

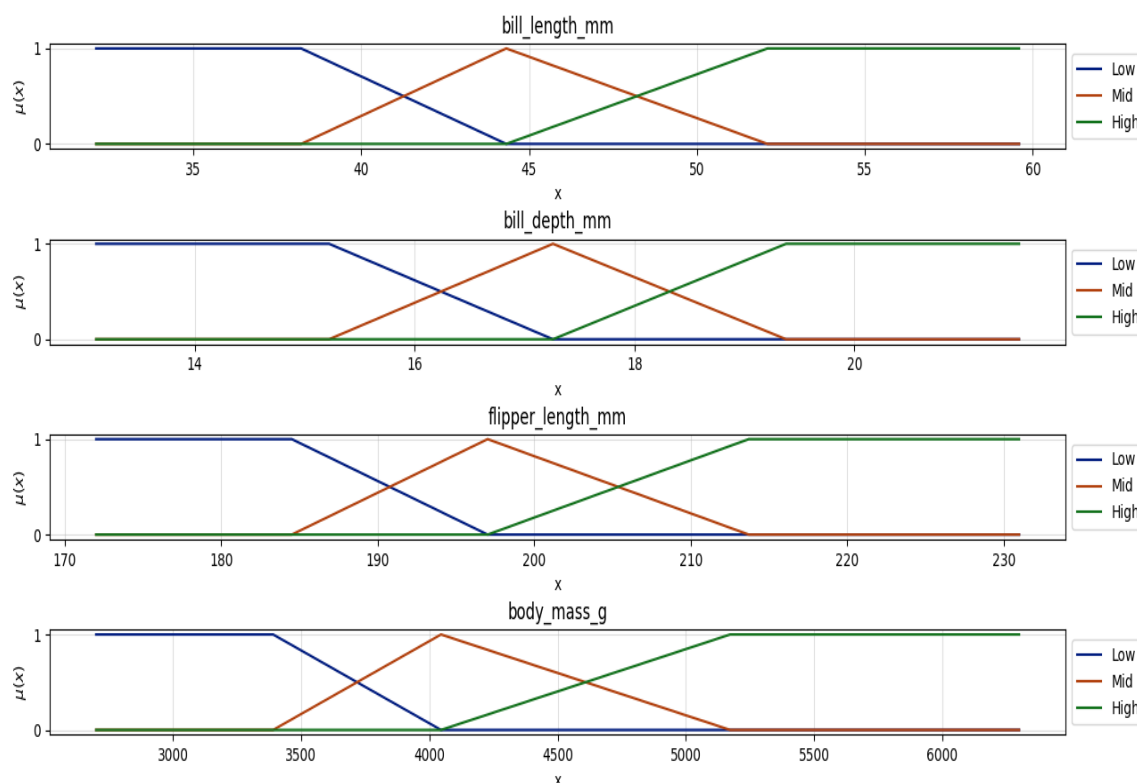


Рис. 4.4. Границі термів для набору видової популяції арктичних пінгвінів.

Далі формується система правил (2.1) за допомогою матричного представлення, передумови яких складені з усіх можливих комбінацій значень нечітких вхідних змінних.

4.2 Приклади та аналіз результатів

4.2.1 Класифікація Ірисів Фішера

Для апробації запропонованого підходу до класифікації об'єктів вибірку $X = \{(x_i, C_m)\}$ було поділено на дві частини: навчальну Q – 130 об'єктів та тестову V – 20 об'єктів.

На рис. 4.5 наведено результат роботи програми для розрахунку діаграми точкового розсіювання. З урахуванням інтерпретації точкової діаграми розсіювання встановлено, що між змінними «petal length (cm)» та «petal width (cm)» існує сильна залежність, а між змінними «sepal width (cm)» та «sepal length (cm)» слабка залежність, але для більш детального аналізу, потрібно буде розрахувати коефіцієнти кореляції.

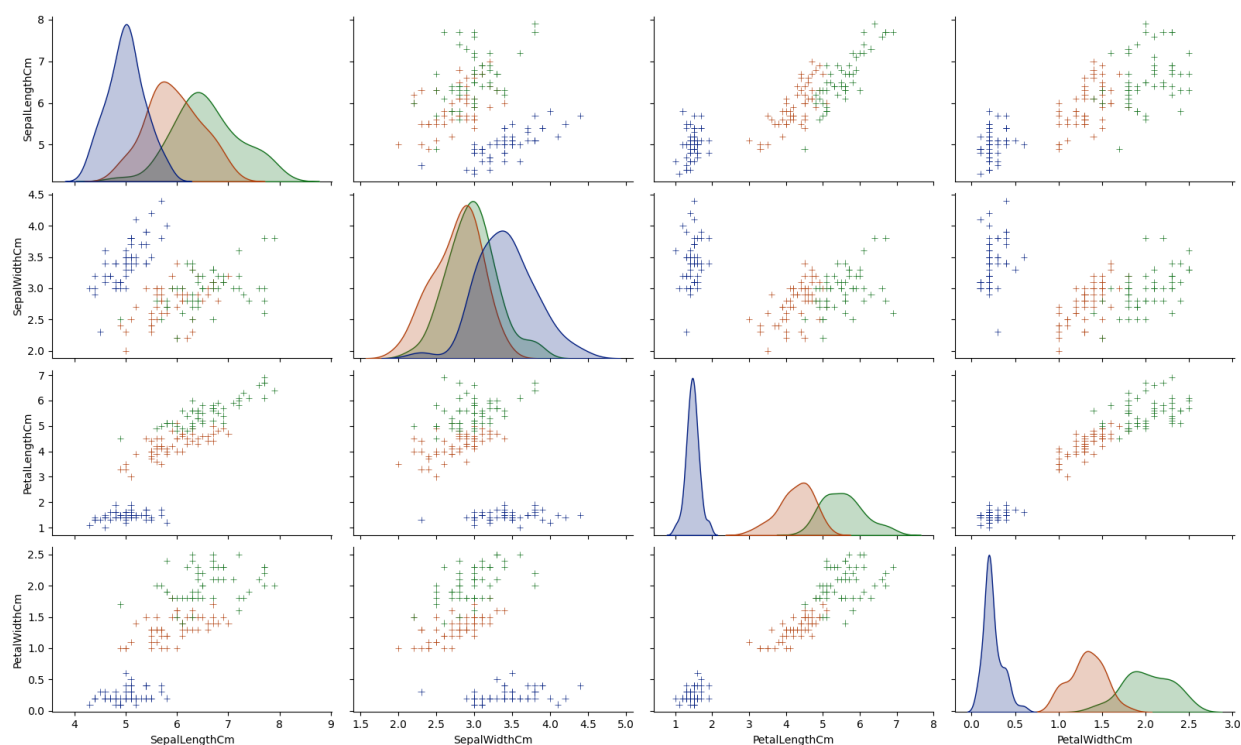


Рис. 4.5. Точкова діаграма розсіювання класів: ● – Iris-setosa, ● – Iris-versicolor, ● – Iris-virginica

Для початкових даних задачі було обчислено коефіцієнти кореляції ознак (Рис. 4.6):

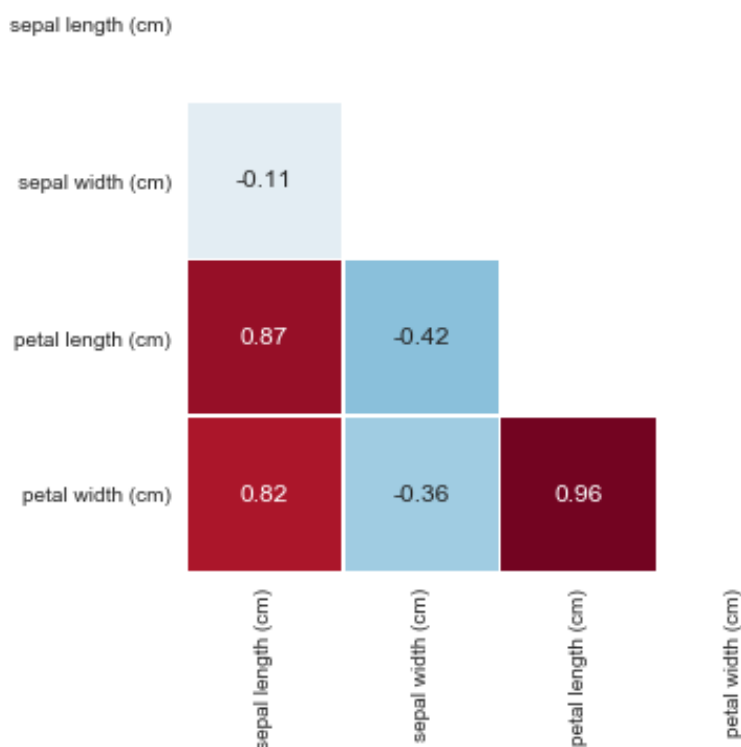


Рис. 4.6. Значення коефіцієнтів кореляції

З урахуванням інтерпретації значень коефіцієнтів було встановлено, що між змінними «petal length (cm)» та «petal width (cm)» існує сильна залежність, значення коефіцієнта кореляції – 0.96, між змінними «sepal width (cm)» та «sepal length (cm)» слабка залежність, значення відповідного коефіцієнта кореляції дорівнює -0.11.

Розглянемо етапи виконання процедури міжкласової класифікації на основі відстані між об'єктами з навчальної вибірки для об'єкту $x_i \in X$, $x_i = (6.3, 2.7, 4.9, 1.8)$, $x_i \in V$. На першому етапі класифікації з використанням сформованих продукційних правил та шляхом виконання процедури фазифікації було розраховано ступені приналежності об'єкта нечітким множинам \tilde{X}^* . За допомогою процедури дефазифікації

розраховано значення $X^* = 60.35$, $M^* = 0.46$. Визначено, що за значенням $X^* = 60.35$, об'єкт x_i належить до двох класів Versicolor та Virginica. Далі за допомогою (2.4), (2.5) побудовано множину ${}^d Y$ і визначено клас, до якого належить об'єкт, за допомогою (2.6):

$$\max \left(\underset{\text{setosa}}{card}({}^d Y), \underset{\text{versicolor}}{card}({}^d Y), \underset{\text{virginica}}{card}({}^d Y) \right) = \max(0, 3, 7) = 7 \Rightarrow \underset{\text{virginica}}{card}({}^d Y)$$

За результатами виконаних розрахунків отримано, що об'єкт належить класу Virginica, що відповідає навчальній вибірці.

Порівняння результатів класифікації з використанням запропонованого в роботі двоетапного підходу і класичного алгоритму нечіткої класифікації наведено на рис. 4.7.

У класичному алгоритмі використовувалась модель нечіткого логічного виведення типу Сугено з чотирма вхідними і однією вихідною змінною, функції належності побудовано за середнім значеннями. При використанні класичного підходу об'єкт x_i було віднесено до класу Versicolor, результат фазифікації вихідної змінної, отриманий методом центру тяжіння, наведено на рис. 4.7а.

Результат класифікації за допомогою запропонованого підходу наведено на рис. 4.7b. Можна бачити, що навколо об'єкту x_i (на рис. 4.7b позначений трикутником) є 3 об'єкти класу Versicolor (на рис. 4.7b позначені квадратами) та 7 об'єктів класу Virginica (на рис. 4.7b позначені кругами), що за (2.6) свідчить про належність об'єкту x_i до класу Virginica з більшою ймовірністю, ніж до класу Versicolor. Проведений аналіз дозволяє визначити належність об'єкту x_i до класу Virginica, що співпадає з результатом класифікації за вибіркою [73].

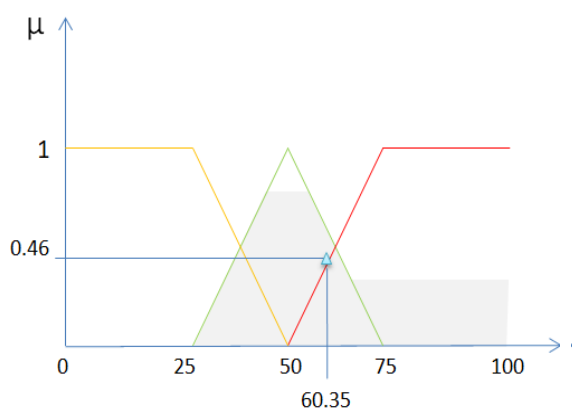


Рис. 4.7а. Класичний алгоритм нечіткої класифікації

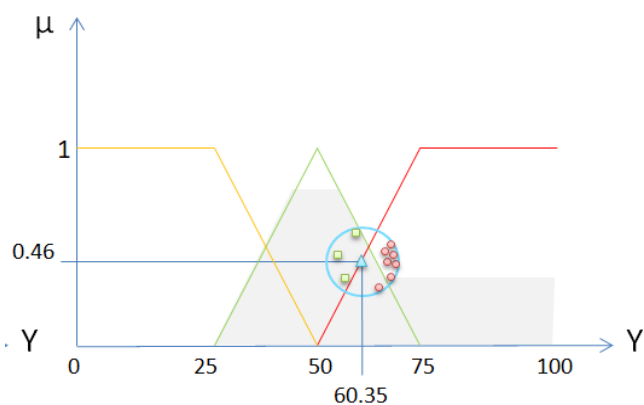


Рис. 4.7b. Розроблений метод

У таблиці 4.5 подано деякі результати класифікації елементів тестової вибірки V.

Таблиця 4.5 – Результати класифікації

X	Вхідні параметри				Дефазифі- цировані значення X^* та M^*	Результат 1-го етапу класифікації (проміжна класифікація)	d $card(Y)$ C_m			Результат класифіка- ції	Дійсне значення
	sepalLength	sepalWidth	petalLength	petalWidth			$C_j = \text{Setosa}$	$C_j = \text{Versicolor}$	$C_j = \text{Virginica}$		
1	4.9	3.1	1.5	0.2	31.25, 0.45	Setosa Versicolor	8	0	0	Setosa	Setosa
10	5.1	3.3	1.7	0.5	33.81, 0.46	Setosa Versicolor	7	0	0	Setosa	Setosa
14	5.8	2.7	3.9	1.2	48.97, 0.38	Setosa Versicolor	0	5	0	Versicolor	Versicolor
20	5.6	3	4.5	1.5	55.49, 0.4	Versicolor Virginica	0	5	0	Versicolor	Versicolor
21	7.7	2.6	6.9	2.3	63.06, 0.49	Versicolor Virginica	0	0	5	Virginica	Virginica
26	6.3	2.5	5	1.9	59.10, 0.47	Versicolor Virginica	0	0	3	Virginica	Virginica
27	6	2.7	5.1	1.6	57.81, 0.44	Versicolor Virginica	0	4	6	Virginica	Versicolor
30	6.3	2.5	4.9	1.5	56.28, 0.45	Versicolor Virginica	0	6	3	Virginica	Versicolor

З використанням введених метрик accuracy, precision (точність), recall (повнота), f1-score (f-міра) для результату класифікації, наведеного в табл. 4.5, отримано такі значення: accuracy=0,95; precision= 0,91; recall = 0,91; f1-score=0,94.

На рис. 4.8 наведено розподіл об'єктів навчальної вибірки в двовимірному просторі на три проміжні класи, можна зазначити, що проміжні класи об'єктів навчальної вибірки добре розрізняються.

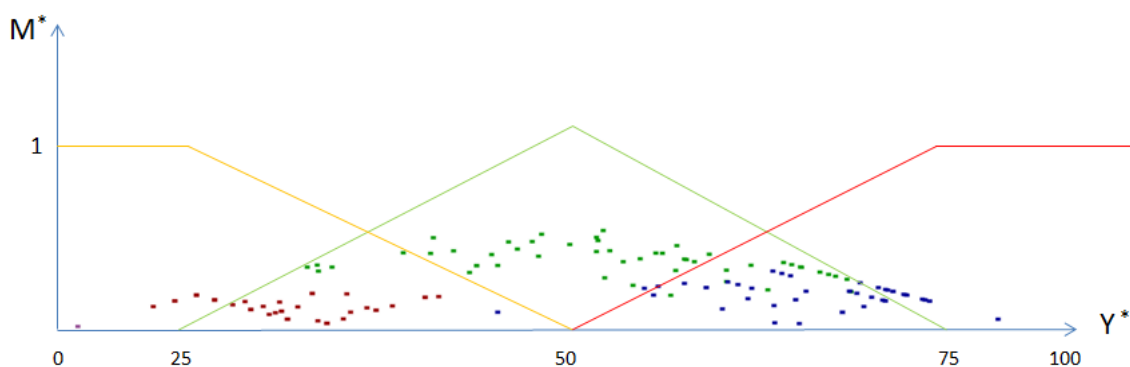


Рис. 4.8. Загальний розподіл об'єктів

Розглянемо вплив запропонованого способу робастного визначення границь термів у випадку появи в наборі $X = \{(x_i, C_m)\}$ об'єктів з аномальними значеннями, ознаки об'єктів, доданих до вибірки, наведені в табл. 4.6. аномальні значення виділені сірим кольором.

Таблиця 4.6 - Додаткові об'єкти з аномальними значеннями

sepalLength	sepalWidth	petalLength	petalWidth	Клас
0.3	0.1	0.1	0.8	Setosa
0.9	0.3	510.1	0.8	Virginica

У таблиці 4.7 представлені результати визначення границь термів, які обчислені з використанням МН (3.4) і з застосування АМ (Arithmetic mean) – середнього арифметичного значення для вибірок, які не містять і містять аномальні об'єкти.

Таблиця 4.7 - Результати визначення границь термів

Лінгвістичні змінні	Лінгвістичні значення	Границі термів (МН)	Границі термів (АМ)	Границі термів (МН) при додаванні аномальних об'єктів	Границі термів (АМ) при додаванні аномальних об'єктів
sepalLength	low	[4.3, 5.75]	[4.3, 5.85]	[0.3, 5.75]	[0.3, 5.7]
	mid	[5.025, 6.025]	[5.07, 6.88]	[3.025, 6.825]	[3.01, 6.80]
	high	[5.75, 7.9]	[5.85, 7.9]	[5.75, 7.9]	[5.7, 7.9]
sepalWidth	low	[2.0, 3.099]	[2.0, 3.06]	[0.1, 3.099]	[0.1, 2.98]
	mid	[2.55, 3.75]	[2.53, 3.73]	[1.59, 3.75]	[1.54, 3.69]
	high	[3.099, 4.4]	[3.06, 4.4]	[3.099, 4.4]	[2.98, 4.4]
petalLength	low	[1.0, 3.35]	[1.0, 3.76]	[0.1, 3.30]	[0.1, 6.97]
	mid	[2.175, 5.125]	[2.38, 5.33]	[1.7, 256.7]	[3.54, 258.54]
	high	[3.35, 6.9]	[3.76, 6.9]	[3.30, 510.1]	[6.97, 510.1]
petalWidth	low	[0.1, 1.05]	[0.1, 1.2]	[0.1, 1.05]	[0.1, 1.19]
	mid	[0.575, 1.775]	[0.65, 1.85]	[0.575, 1.775]	[0.65, 1.85]
	high	[1.05, 2.5]	[1.2, 2.5]	[1.05, 2.5]	[1.19, 2.5]

Можна відзначити, що при додаванні в навчальну вибірку аномальних об'єктів, різниця між границями термів обчислених за допомогою МН і АМ змінюється в діапазоні від 0.015 до 3.67. Найбільші відмінності спостерігаються при визначенні границь термів лінгвістичної змінної petalLength (в табл. 4.7 виділено сірим кольором).

Покажемо вплив робастного визначення границь термів МН на результат класифікації. На рис. 4.9, рис. 4.10 наведено розподіл об'єктів між проміжними класами, який отримано для термів, границі яких визначалися з використанням АМ (рис. 4.9) і МН (рис. 4.10).

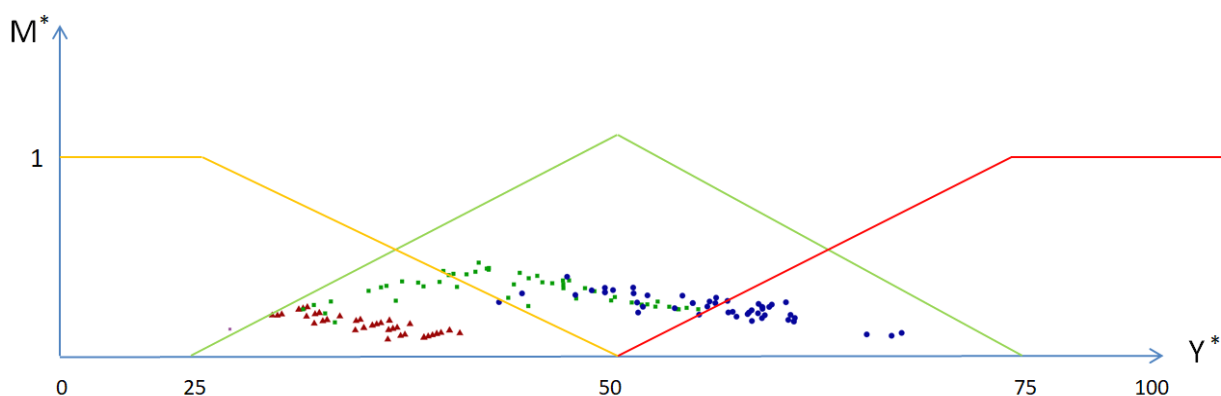


Рис. 4.9. Розподіл об'єктів між проміжними класами, обчислений з використанням АМ

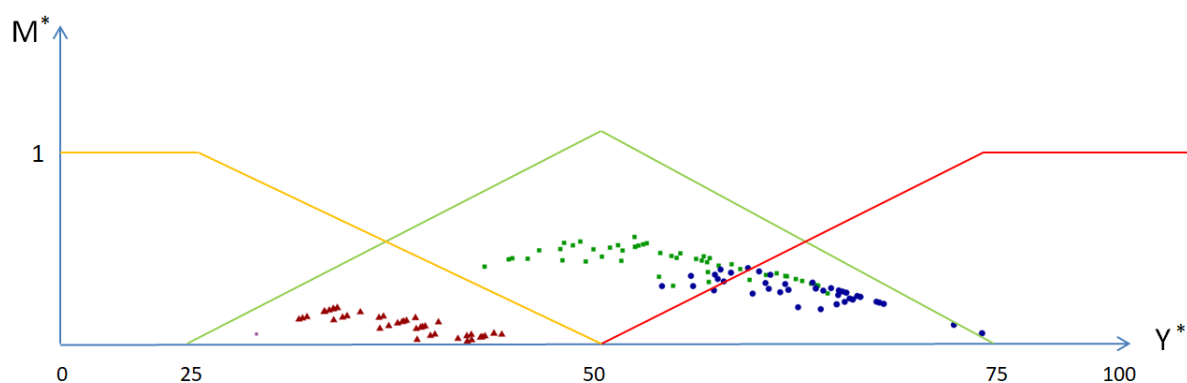


Рис. 4.10. Розподіл об'єктів між проміжними класами, обчислений з використанням МН

У випадку, коли границі термів обчислювались з використання АМ (рис. 4.9), отримані класи об'єктів Setosa (об'єкти позначено трикутником), Versicolor (об'єкти позначено квадратами), Virginica (об'єкти позначено колом) погано розрізняються. У випадку застосування МН для обчислення границь термів (рис. 4.10) класи добре розрізняються, незначні перетинання присутні для об'єктів Versicolor (позначено квадратами), Virginica (позначено колом).

У таблиці 4.8 наведені значення метрик якості результатів класифікації з використанням різних способів обчислення границь термів. З аналізу таблиці видно, що використання міжквартильного середнього для визначення границь термів дозволяє отримати прийнятну точність класифікації для вибірок, які містять об'єкти з аномальними характеристиками.

Таблиця 4.8 - Результатів класифікації з використанням різних способів обчислення границь термів

	Метод визначення границь термів	accuracy	precision	recall	f1-score
Звичайна вибірка	MH	0,95	0,91	0,91	0,94
	AM	0,93	0,90	0,90	0,92
Вибірка з аномальними об'єктами	MH	0,94	0,90	0,90	0,93
	AM	0,60	0,58	0,55	0,56

В роботі досліджувалося питання впливу способу розбиття вибірки на навчальну і тестову, а також розміру навчальної вибірки на якість класифікації. Результати порівняння наведені в табл. 4.9.

Таблиця 4.9 - Вплив розміру навчальної вибірки на результат класифікації

Розмір навчальної вибірки (навчальна/ тестова)	accuracy	precision	recall	f1-score
130/20	0,95	0,91	0,91	0,94
120/30	0,94	0,88	0,89	0,92
100/50	0,92	0,85	0,83	0,90
75/75	0,85	0,83	0,81	0,85

З аналізу результатів випливає, що розмір навчальної вибірки впливає на результат класифікації, зі зменшенням обсягу навчальної вибірки якість класифікації погіршується.

Отримана експертна система має точність класифікації 95% після тренування на 130 об'єктах з 150, для розрахунку точності було проведено 100 експериментів у циклі, де система постійно наново перенавчалася на випадкових вибірках 130/20 (навчальна/тестова). Порівнюючи з сучасними алгоритмами, нечіткої класифікації, штучного інтелекту, дерев рішень, та алгоритмами класифікації, загальний розподіл точності для вирішення задачі класифікації ірисів 88-97% розроблений алгоритм має прийнятний рівень точності [62, 114].

4.2.2 Редукція бази правил для задачі класифікації Ірисів

Фішера

Відповідно до запропонованого підходу було побудовано повну базу правил, що складається з 81 правила і має точність класифікації 95%. З використанням розробленого адаптивного алгоритму було проведено редукцію бази правил, оцінено вплив процедури редукції на якість отриманої бази правил, отримано результати класифікації з використанням скороченої бази правил.

В таблиці 4.10 представлені антецеденти правил та значення коефіцієнтів α , що відповідають кожному з правил. Перші 19 правил, антецеденти яких не були активовані жодним об'єктом навчальної вибірки, мають значення коефіцієнту $\alpha=0$ та можуть бути видалені з бази знань без втрати якості класифікації. Для решти 62 правил значення коефіцієнту α відрізняється від нуля.

Таблиця 4.10 – Антецеденти правил та значення коефіцієнтів α , що відповідають кожному з правил

№	α	Антецедент
1	0.0	{'SepalLengthCm': 'Low', 'SepalWidthCm': 'High', 'PetalLengthCm': 'Low', 'PetalWidthCm': 'Mid'}
2	0.0	{'SepalLengthCm': 'Low', 'SepalWidthCm': 'High', 'PetalLengthCm': 'Mid', 'PetalWidthCm': 'Low'}
3	0.0	{'SepalLengthCm': 'High', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'Low', 'PetalWidthCm': 'Low'}
...
19	0.0	{'SepalLengthCm': 'High', 'SepalWidthCm': 'High', 'PetalLengthCm': 'High', 'PetalWidthCm': 'Low'}
20	10.0	{'SepalLengthCm': 'Low', 'SepalWidthCm': 'Low', 'PetalLengthCm': 'High', 'PetalWidthCm': 'Low'}
...
33	63.0	{'SepalLengthCm': 'Low', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'High', 'PetalWidthCm': 'High'}
...
48	93.0	{'SepalLengthCm': 'High', 'SepalWidthCm': 'Low', 'PetalLengthCm': 'High', 'PetalWidthCm': 'Mid'}
...
81	100	{'SepalLengthCm': 'High', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'High', 'PetalWidthCm': 'High'}

Для правил, що залишилися, було обчислено значення коефіцієнта β , в таблиці 4.11 представлені правила, що розташовані в порядку зменшення значень коефіцієнта β .

Таблиця 4.11 - Правила та значення коефіцієнтів β , що відповідають кожному з правил

№	β	Правила
1	0.4 31	IF {'SepalLengthCm': 'Low', 'SepalWidthCm': 'High', 'PetalLengthCm': 'Low', 'PetalWidthCm': 'Low'} THEN 'Iris-setosa'
2	0.3 03	IF {'SepalLengthCm': 'Low', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'Low', 'PetalWidthCm': 'Low'} THEN 'Iris-setosa'
3	0.2 19	IF {'SepalLengthCm': 'High', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'High', 'PetalWidthCm': 'High'} THEN 'Iris-virginica'
...
10	0.0 89	IF {'SepalLengthCm': 'Mid', 'SepalWidthCm': 'Low', 'PetalLengthCm': 'Mid', 'PetalWidthCm': 'Low'} THEN 'Iris-versicolor'
...
23	0.0 44	IF {'SepalLengthCm': 'High', 'SepalWidthCm': 'Low', 'PetalLengthCm': 'Mid', 'PetalWidthCm': 'Mid'} THEN 'Iris-versicolor'
24	0.0 43	IF {'SepalLengthCm': 'Mid', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'Mid', 'PetalWidthCm': 'Low'} THEN 'Iris-versicolor'
25	0.0 39	IF {'SepalLengthCm': 'High', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'High', 'PetalWidthCm': 'Mid'} THEN 'Iris-versicolor'
...
43	0.0 12	IF{'SepalLengthCm': 'High', 'SepalWidthCm': 'Low', 'PetalLengthCm': 'Mid', 'PetalWidthCm': 'Low'} THEN 'Iris-versicolor'
...
62	0.0 01	IF {'SepalLengthCm': 'Low', 'SepalWidthCm': 'Mid', 'PetalLengthCm': 'High', 'PetalWidthCm': 'Low'} THEN 'Iris-versicolor'

Після застосування алгоритму редукції для заданого значення точності $\Delta = 90\%$, базу правил було скорочено до 15 правил. Слід зазначити, що у результуючій базі перебувають правила, які описують усі вихідні класи об'єктів.

Для проведення класифікації з використанням редукованої бази правил були сформовані навчальна та тестова вибірки у розмірі 130/20, відповідно. На рис. 4.11 наведено матрицю протиріч, отриману для результатів класифікації на тестовій вибірці за допомогою редукованої бази правил. У тестовій вибірці містилося 20 об'єктів (3 – *Iris-setosa*, 9 – *Iris-versicolor*, 8 – *Iris-virginica*), серед яких правильно ідентифіковані всі об'єкти класів *Iris-setosa* та *Iris-virginica*. З 9 об'єктів класу *Iris-versicolor* вірно ідентифіковані 7 об'єктів, один об'єкт був неправильно класифікований як *Iris-setosa*, а другий – як *Iris-virginica*.

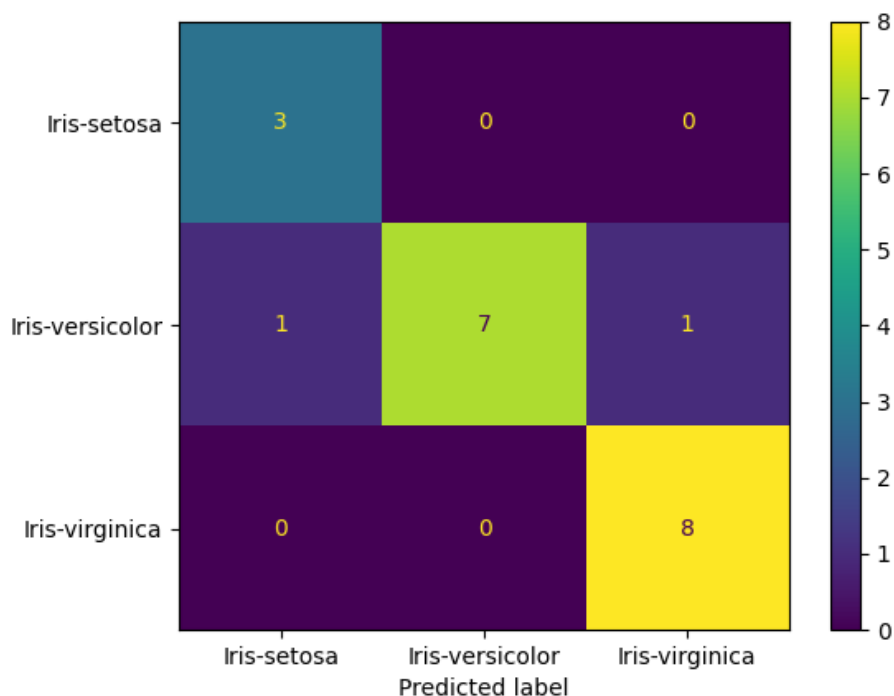


Рис. 4.11. Матриця протиріч – 130/20 (навчальна/тестова), точність 90% при використанні 1–15 правил з таблиці 4.11.

Таблиця 4.12 - Значення метрик якості класифікації

Клас ірисів	precisio n	recall	f1- score	кількість елементів тестової вибірки
<i>Iris-setosa</i>	0.75	1.00	0.86	3
<i>Iris-versicolor</i>	1.00	0.78	0.88	9
<i>Iris-virginica</i>	0.89	1.00	0.94	8
avg accuracy	0.9			

В таблиці 4.12 наведено значення метрик якості класифікації. Зменшення кількості правил у базі знань дозволило суттєво скоротити час, необхідний для проведення класифікації.

Далі було досліджено вплив розмірів навчальної та тестової вибірок на якість класифікації. Об'єкти було розподілено на навчальну та тестову вибірки у відношенні 100/50 відповідно. При заданому критерії якості $\Delta = 90\%$ антецеденти 25 правил не були активовані жодним об'єктом навчальної вибірки та підлягають видаленню з бази правил. Після виконання етапів алгоритму результуюча база правил складається з 17 правил.

На рис. 4.12. наведено матрицю протиріч, отриману для результатів класифікації на тестовій вибірці за допомогою редукованої бази правил. У тестовій вибірці містилося 50 об'єктів (17 – *Iris-setosa*, 16 – *Iris-versicolor*, 17 – *Iris-virginica*), серед яких правильно ідентифіковані всі об'єкти класу *Iris-setosa*. З 16 об'єктів класу *Iris-versicolor* вірно ідентифіковані 14 об'єктів, 2 об'єкта було неправильно класифіковано як *Iris-virginica*. З 17 об'єктів класу *Iris-virginica* вірно ідентифіковані 14 об'єктів, 3 об'єкта було неправильно класифіковано як *Iris-setosa*. Можна зазначити, що при зменшенні розмірів навчальної вибірки розмір редукованої бази правил збільшився незначно, при цьому якість класифікації відповідає заданому критерію.

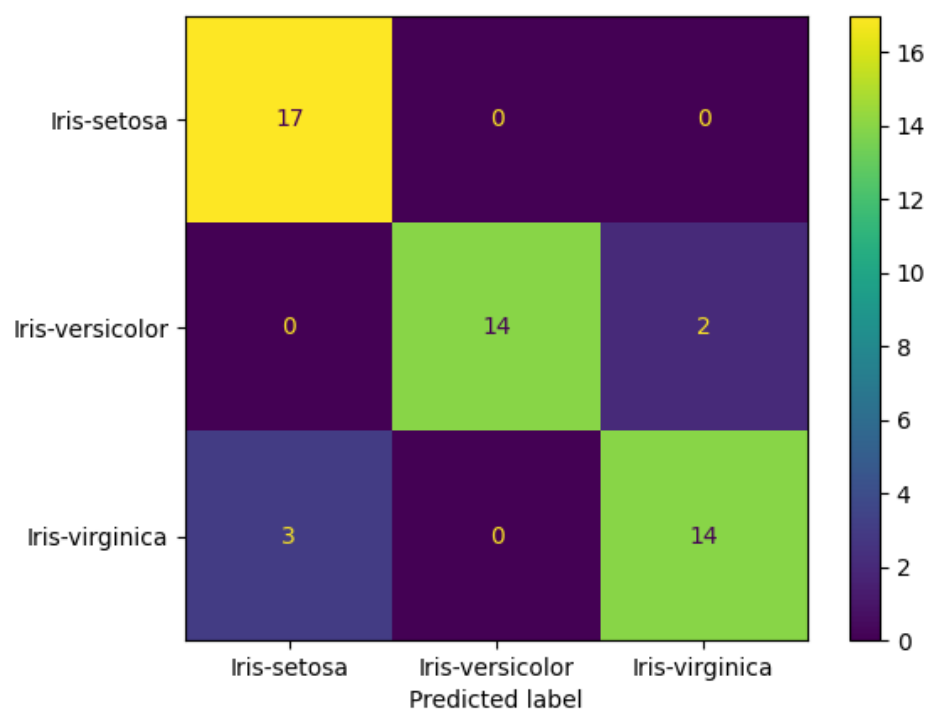


Рис. 4.12. Матриця протиріч – 100/50 (навчальна/тестова), точність 90% при використанні 1–17 правил.

У таблиці 4.13 наведено значення метрик якості класифікації.

Таблиця 4.13 - Значення метрик якості класифікації

Клас ірисів	precision	recall	f1-score	кількість елементів тестової вибірки
Iris-setosa	0.85	1.00	0.92	17
Iris-versicolor	1.00	0.88	0.93	16
Iris-virginica	0.88	0.82	0.85	17
avg accuracy	0.9			

4.2.3 Класифікація видової популяції арктичних пінгвінів

Для аналізу результату роботи алгоритму також використовується для задачі класифікації видової популяції арктичних пінгвінів. Для проведення класифікації рис. 4.14., 4.15. були сформовані навчальна та тестова вибірки у розмірі 300/44 (навчальна/тестова).

На рис. 4.13 наведено результат програми для розрахунку діаграми точкового розсіювання для ознак: `bill_length_mm` – довжина дзьоба; `bill_depth_mm` – глибина дзьоба; `flipper_length_mm` – довжина плавця; `body_mass_g` – маса тіла.

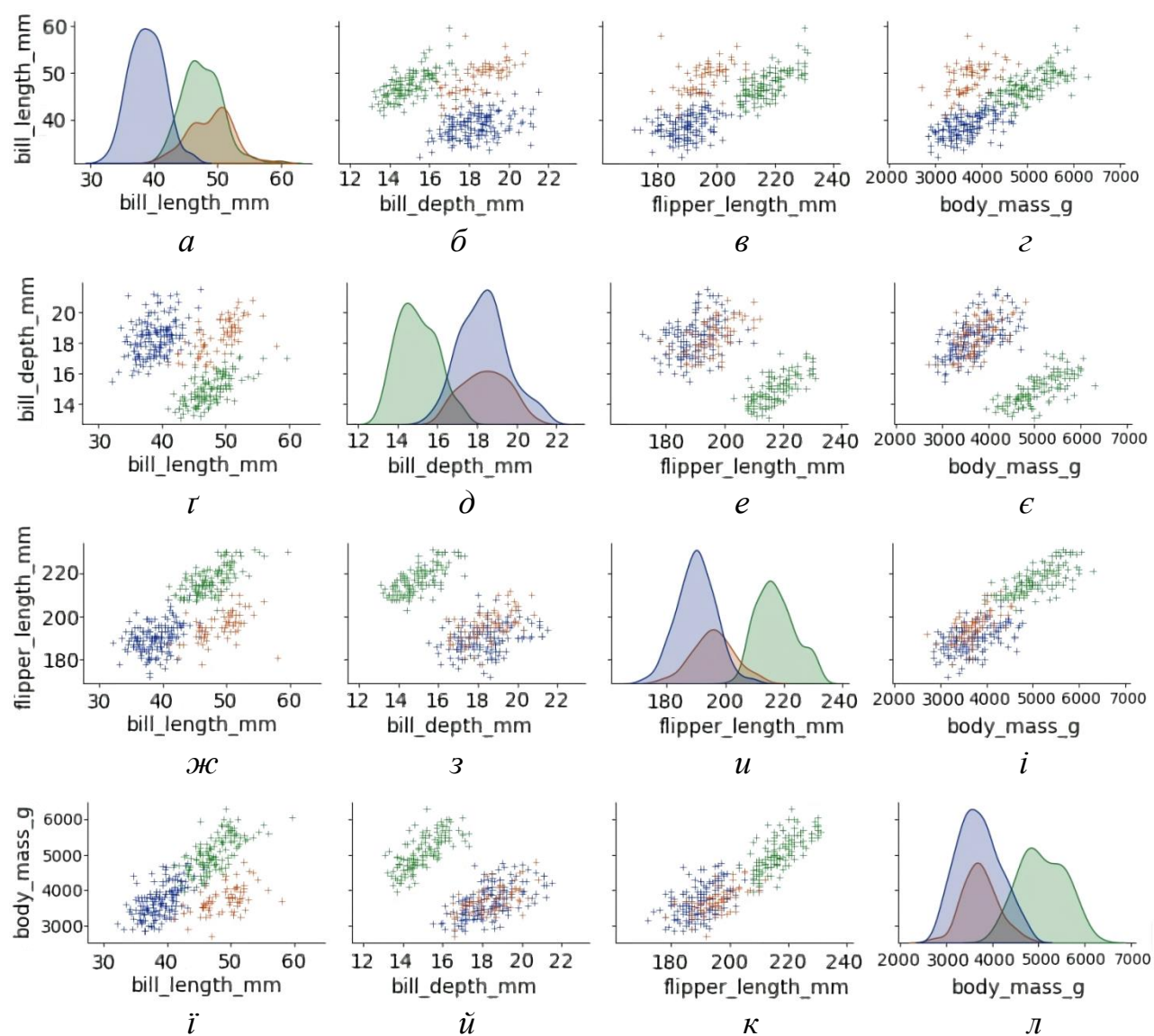


Рис. 4.13. Точкова діаграма розсіювання класів: ● – Chinstrap, ● – Adelie, ● – Gentoo

З аналізу рис. 4.13. можна помітити, що класи Adelie і Chinstrap досить сильно схожі, і їх класифікація може бути ускладненою.

З урахуванням інтерпретації значень коефіцієнтів та точкової діаграми розсіювання було встановлено, що для класів Adelie і Chinstrap:

- між змінними «bill_depth_mm» та «flipper_length_mm» існує сильна залежність – Рис. 4.13є,з;
- між змінними «bill_depth_mm» та «body_mass_g» існує сильна залежність – Рис. 4.13є,й.

На рис. 4.14, 4.15 наведено результати класифікації до та після навчання нечіткої бази правил.

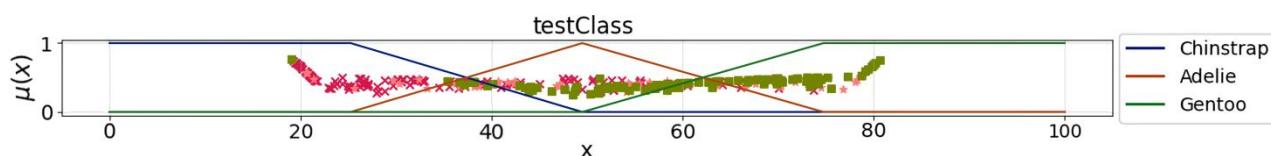


Рис. 4.14. До навчання

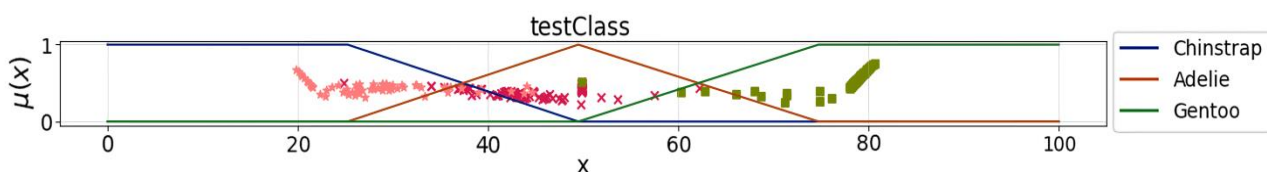


Рис. 4.15. Після навчання

На Рис. 4.14. можна побачити, що класи не визначені та значення у результатах перемішані.

Після навчання моделі (Рис. 4.15) можна побачити, що дані згрупувалися, і кожен клас знаходиться у межах своєї функції належності. Слід зазначити, що на графіках точками зображено одночасно всі 344 об'єкти з навчального та тестового наборів.

Розглянемо, як змінилося стандартне відхилення до навчання моделі та після навчання:

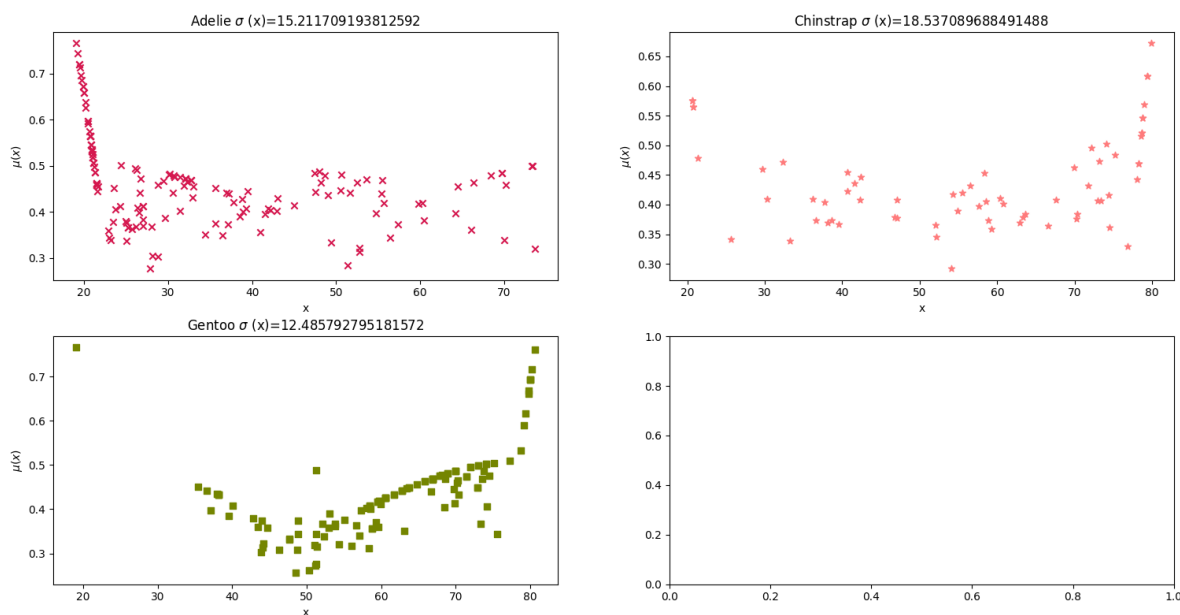


Рис. 4.16. До навчання: Графік стандартного відхилення

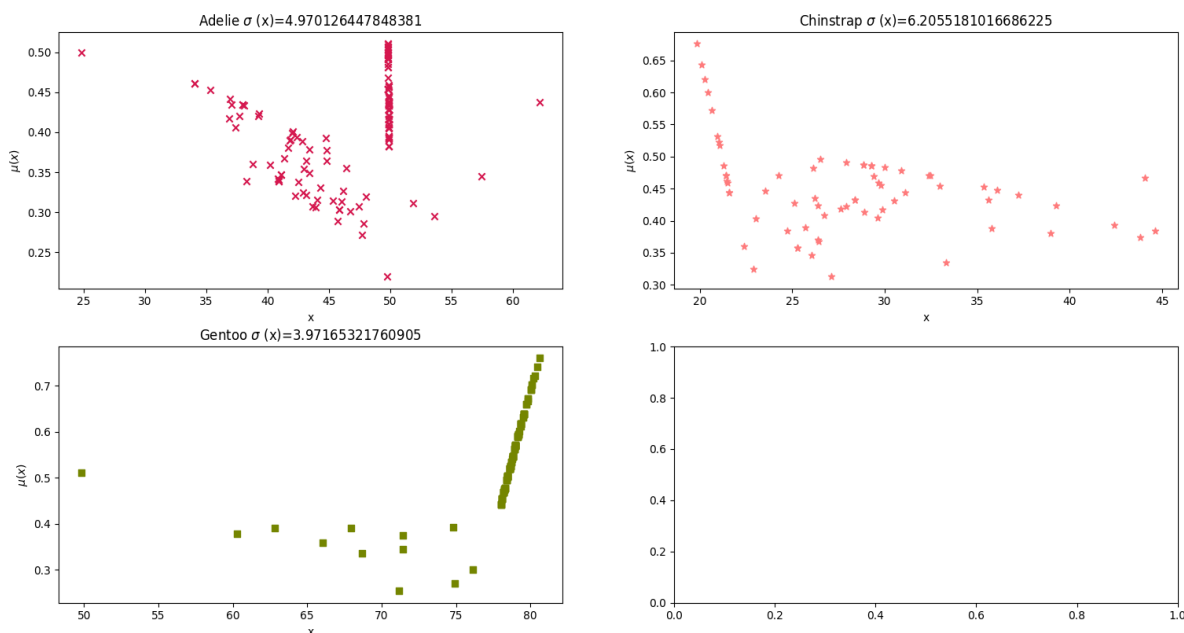


Рис. 4.17. Після навчання: Графік стандартного відхилення

До навчання:

Adelie $\sigma(x) = 15.2117$; Chinstrap $\sigma(x) = 18.53708$; Gentoo $\sigma(x) = 12.48579$.

Після навчання:

Adelie $\sigma(x) = 4.97012$; Chinstrap $\sigma(x) = 6.20551$; Gentoo $\sigma(x) = 3.97165$.

Можна помітити, що після навчання стандартне відхилення зменшилося в рази, а це означає, що множина стала більш згурпованою і більш щільною, що у свою чергу говорить про покращення результатів класифікації. Також, з аналізу рис. 4.16. та рис. 4.17. можна побачити, що

після навчання границі термів стають чіткішими, це можна спостерігати по осі x .

Розглянуто залежність результату класифікації від розміру навчальної вибірки до та після навчання. Для цього були сформовані навчальна та тестова вибірки у розмірі 200/144 (навчальна/тестова). На рис. 4.18а наведено матрицю протиріч до здійснення процедури навчання моделі, на рис. 4.18б - після навчання моделі. В табл. 4.14, 4.15 наведено значення метрик якості до та після навчання відповідно. Матриці протиріч на рис. 4.18, 4.19 побудовано з використанням розробленого власноруч програмного забезпечення на мові Python із застосуванням бібліотек.

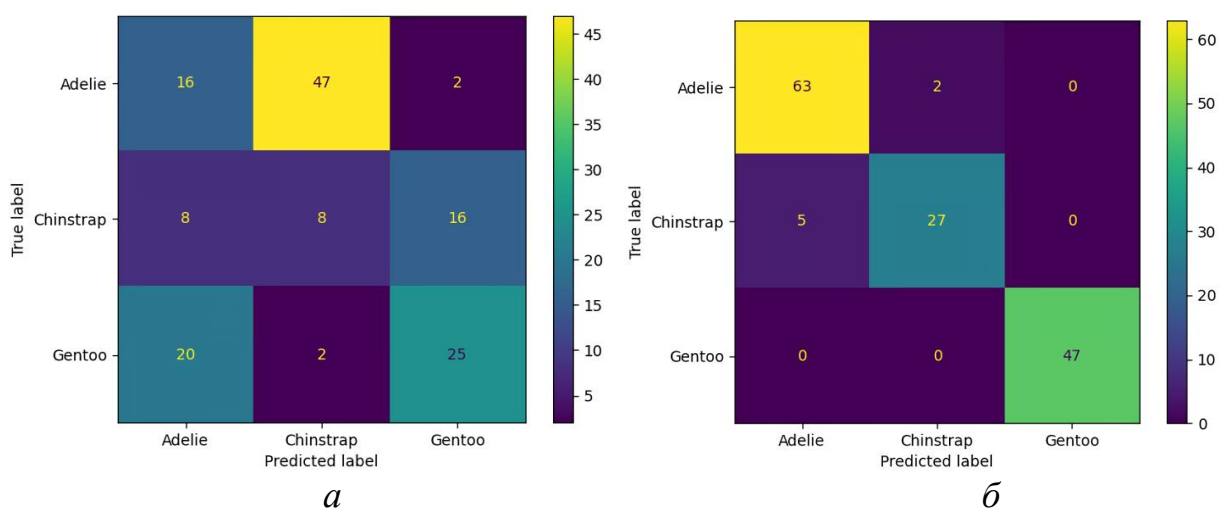


Рис. 4.18. Матриці протиріч – 200/144 (навчальна/тестова): *а* – до навчання; *б* – після навчання

Таблиця 4.14 – До навчання: 200/144 (навчальна/тестова)

Вид	precision	recall	f1-score	support
Adelie	0.36	0.25	0.29	65
Chinstrap	0.14	0.25	0.18	32
Gentoo	0.58	0.53	0.56	47
avg accuracy	0.34			

Таблиця 4.15 – Після навчання: 200/144 (навчальна/тестова)

Вид	precision	recall	f1-score	support
Adelie	0.93	0.97	0.95	65
Chinstrap	0.93	0.84	0.89	32
Gentoo	1.00	1.00	1.00	47
avg accuracy	0.90			

Для аналізу впливу процесу навчання на результат класифікації було змінено розміри вибірок 250/94 (навчальна/тестова). На рис. 4.19а представлена матриця протиріч до навчання, на рис. 4.19б - після навчання системи. У табл. 4.16, 4.17 наведено значення метрик якості відповідно до та після навчання.

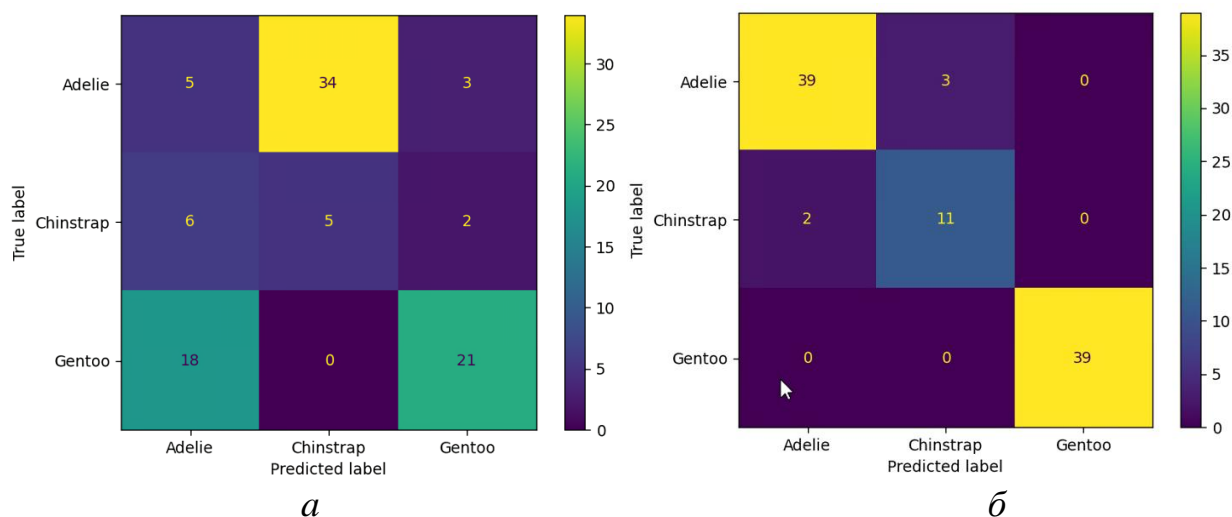


Рис. 4.19. Матриці протиріч – 250/94 (навчальна/тестова): *a* – до навчання; *б* – після навчання

Таблиця 4.16 – До навчання: 250/94 (навчальна / тестова)

Вид	precision	recall	f1-score	support
Adelie	0.17	0.12	0.14	42
Chinstrap	0.13	0.38	0.19	13
Gentoo	0.81	0.54	0.65	39
avg accuracy	0.33			

Таблиця 4.17 – Після навчання: 250/94 (навчальна / тестова)

Вид	precision	recall	f1-score	support
Adelie	0.95	0.93	0.94	42
Chinstrap	0.79	0.85	0.81	13
Gentoo	1.00	1.00	1.00	39
avg accuracy	0.93			

Розглянуто залежність результату класифікації від розміру навчальної вибірки до та після навчання. Використані вибірки 200/144 та 250/94 (навчальна/тестова). З аналізу табл. 4.14–4.17 видно, що збільшення обсягу

навчальної вибірки дозволяє покращити точність класифікації для вибірок, а також налаштувати вектор консеквентів для розв'язання задачі класифікації.

Для задачі класифікації видової популяції арктичних пінгвінів була сформована навчальна та тестова вибірки у розмірі 200/144 (навчальна/тестова). Застосовуючи отриману базу знань до етапу навчання на рис. 4.14 можна побачити, що класи не визначені та значення у результатах перемішані. Відсутнє групування об'єктів за класами. Але, після навчання на рис. 4.15 можна побачити, що дані згрупувалися і кожен клас знаходиться у межах своєї функції належності. До навчання на рис. 4.18а представлена матриця протиріч та аналіз метрик в табл. 4.14. За результатами можна побачити, що точність досить низька 34 %. Після навчання на рис. 4.18б та аналізу метрик у табл. 4.15 можна помітити, що точність системи зросла до 90 %, а кількість невірних відповідей у матриці протиріч значно зменшилася.

Далі була сформована навчальна та тестова вибірки у розмірі 250/94 (навчальна/тестова). З аналізу рис. 4.19а та табл. 4.16 також можна помітити низьку точність 33 %. Але, після навчання (рис. 4.19б) точність зросла до 93 %, як і інші показники в табл. 4.17.

4.3 Висновки до розділу

Проведено обчислювальний експеримент для розв'язання модельних задач мультикласової класифікації на наборах даних Іриси Фішера (Iris Data Set) та видова популяція арктичних пінгвінів.

Проведено налаштування нечіткої моделі для розв'язання задачі класифікації шляхом визначення границь термів для функції належності. Налаштування границь термів проведено за допомогою аналізу ознак об'єктів навчальної вибірки, аналізу частотного розподілу для кожної ознаки та розробленого методу визначення границь термів. Для зменшення впливу наявних аномальних об'єктів в навчальній вибірці на результат класифікації для визначення границь термів застосовано міжквартильне середнє.

З використанням власноруч розробленого програмного забезпечення для наборів даних, що розглядаються, налаштовано моделі представлення знань, автоматично побудовано бази правил в процесі навчання з використанням навчальної вибірки. Запропонований підхід дозволяє мінімізувати участь експерта при формуванні системи правил та налаштуванні моделі нечіткої класифікації. Процедуру класифікації удосконалено у частині вирішення конфліктів під час прийняття рішень у разі, коли об'єкт одночасно належить кільком класам.

Наведено порівняння результатів класифікації з використанням запропонованого в роботі підходу і класичного алгоритму нечіткої класифікації.

Здійснено аналіз впливу розміру навчальної вибірки на результати класифікації та якість експертної системи. Виявлено, що збільшення розміру навчальної вибірки призводить до покращення результатів класифікації та метрик якості.

Здійснено аналіз результатів виконання процедури редукції нечіткої бази правил для розв'язання задачі мультикласової класифікації з використанням запропонованого критерію якості. В результаті редукції з повної бази правил видаляються правила, антецеденти яких не активовані, серед правил, що залишилися, обираються правила, які мають найбільший вплив на результат класифікації.

Аналіз результатів класифікації з використанням редукованої бази правил виконано для об'єктів тестової вибірки. З аналізу одержаних результатів випливає, що запропонована процедура редукції дозволяє суттєво скоротити базу правил без втрат у якості класифікації.

Основні результати розділу опубліковані в [13, 14, 18, 19, 134].

ВИСНОВКИ

У дисертаційній роботі вирішено актуальну наукову проблему, що полягає у розробці методів та алгоритмів автоматичної генерації бази знань системи логічного виведення для задач класифікації на основі експериментальних даних, вдосконаленню методів логічного виведення, методів налаштування параметрів моделі представлення знань, методів доведення коректності бази знань та дослідженню якості побудованої системи логічного виведення.

Основні наукові результати дисертації полягають у наступному:

1. Дістав подальшого розвитку метод побудови нечіткої продукційної моделі для представлення знань про об'єкти предметної області на основі навчальної вибірки за допомогою розроблених алгоритмів навчання.
2. Для формалізації чітких та нечітких залежностей в даних запропоновано вид нечіткого продукційного правила, що поєднує моделі Мамдані та Такагі-Сугено-Канг та містить у консеквенті правила функцію. Зазначена функція використовується як ваговий коефіцієнт та визначає ступінь належності вихідної змінної до результуючої терм-множини.
3. Удосконалено методику побудови сукупності правил бази знань та алгоритм їх автоматичного формування за допомогою попереднього аналізу ознак навчальної вибірки, матричного представлення антецедентів та векторного представлення консеквентів правил.
4. Дістав подальшого розвитку метод логічного виведення для розв'язання задач класифікації в умовах нечіткості та неповноти інформації про об'єкти предметної області. За наявності конфлікту при прийнятті рішення, що полягає у неспроможності системи однозначно визначити належність об'єкта певному класу, запропоновано процедуру порівняння відстані між об'єктами та класами.
5. Дістав подальшого розвитку метод ідентифікації та налаштування значень параметрів нечіткої продукційної моделі, зокрема, автоматизовано

визначення лінгвістичних змінних, терм-множин, границь термів, параметрів функцій належності, вагових коефіцієнтів правил, кількості правил.

6. Вперше для перевірки коректності автоматично побудованої бази правил за критеріями повноти, мінімальності, зв'язності і несуперечності застосовано логіку Хоара, метод резолюцій та програмне забезпечення Simplify.

7. Вперше запропоновано адаптивний алгоритм редукції бази правил із використанням методу дихотомії з динамічним кроком та інтегральної метрики, що враховує кількість активацій певного правила та ступінь належності об'єктів навчальної вибірки правилу. Множина правил поділяється на підмножину правил, що виконуються, та підмножину правил, що є надмірними для досягнення критерію якості

8. Досліджено якість побудованої нечіткої продукційної моделі з використанням матриці невідповідностей та метрик, що визначають частину правильних відповідей моделі (accuracy), точність (precision), повноту (recall), f-міру (f1-score).

9. Створено нові ефективні алгоритми розв'язання вищеназваних задач.

10. Створено програмний продукт на об'єктно-орієнтованих мовах програмування C++, JavaScript та Python, що реалізує розроблені алгоритми.

11. Методи і алгоритми застосовано до розв'язання модельних задач класифікації.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Абдулхаков А. Р., Катасёв А. С., Кирпичников А. П. Методы редукции нечетких правил в базах знаний интеллектуальных систем. *Вестник Казанского технологического университета*, 2014. 17(23), С. 389-392.
2. Божко К., Морозова І. Нечітка модель для оцінювання якості атмосферного повітря. *Measuring and Computing Devices in Technological Processes*, 2022. Vol. 72(4), С. 134-141. DOI: 10.31891/2219-9365-2022-72-4-19.
3. Булат А.Ф., Кісельова О.М., Гарт Л.Л., Притоманова О.М. Математичні моделі двоетапних задач оптимального розміщення-розбиття в умовах невизначеності. *Питання прикладної математики і математичного моделювання*, 2022. Вип. 22. С. 11-23. DOI: <https://doi.org/10.15421/322202>
4. Буценко Ю., Лабжинський В. Машинний аналіз неконтрольованих параметрів промислових систем в режимі реального часу на основі нечіткої логіки. *Information Technology: Computer Science, Software Engineering and Cyber Security*, 2023. С. 12-23. DOI: 10.32782/IT/2022-2-2.
5. Вавіленкова А.І. Автоматизація процесу побудови простого речення. *Проблеми інформатизації та управління*. Зб. наук. пр. Київ, 2018, Вип. 2(24). С. 28–31.
6. Вавіленкова А.І. Аналіз моделей представлення знань в експертних системах управління. *Проблеми інформатизації та управління: зб. наук. праць*. – К.: НАУ, 2007. Вип. 4(22). С. 14–17. URL: <https://jrnl.nau.edu.ua/index.php/PIU/article/view/9140>.
7. Джексон П. Введение в экспертные системы. *Introduction to Expert Systems*. 3-е изд. М.: Вильямс, 2001. 624 с.
8. Драч І.В., Ткачук В.П., Пакліна Ю.С. Нечітка продукційна модель оцінки параметра методу паводковий-спрямованого пошуку збалансованого стану ротора. *Наука і освіта: збірник праць XIII*

- Міжнародної наукової конференції, 4–13 січ. 2019 р., м. Хайдусобосло (Венгрія). Хмельницький: ХНУ, 2019. С. 83-88.*
9. Єгошкін Д.І., Гук Н.А. The advantages of using MISO and MIMO models in fuzzy expert systems. *Прикладна математика та інформаційні технології ПМ&ІТ 2022: Тези доп. міжнародної наукової конференції присвяченої 60-річчю кафедри прикладної математики та інформаційних технологій, 22 – 24 вересня 2022., м. Чернівці, 2022. С. 170-172.* Режим доступу до ресурсу: <http://www.amit60.fmi.org.ua/files/AMIT2022-Materials.pdf#page=171>.
 10. Єгошкін Д.І., Гук Н.А. The application of fuzzy logic in classification tasks. *Сучасні інформаційні та комунікаційні технології на транспорті, в промисловості та освіті: Матеріали XIV міжнародної науково-практичної конференції, 15.12.2020 – 16.12.2020., м. Дніпро, 2020. С. 69-70.* Режим доступу до ресурсу: <https://crust.ust.edu.ua/server/api/core/bitstreams/cf8adc16-fd69-42b1-a82f-cbc77dbcfe2f/content#page=70>.
 11. Єгошкін Д.І., Гук Н.А. The classification algorithm on the basis of the fuzzy logic with dynamic knowledge-based system. *Сучасні науково-технічні дослідження у контексті мовного простору (іноземними мовами) 11-12 квітня 2019 року: матеріали VIII Регіональної науково-практичної конференції молодих учених та студентів., м. Дніпро, 2019. С. 83-84.* Режим доступу до ресурсу: https://www.dnu.dp.ua/docs/ndc/202/such_nauk_teh_dosl.pdf#page=83.
 12. Єгошкін Д.І., Гук Н.А. The fuzzy logic classification algorithm with three dimensional membership functions and dynamic knowledge base. *Молодь у світі сучасних технологій за тематикою: Використання інформаційних та комунікаційних технологій в сучасному цифровому суспільстві: матеріали міжнар. наук.-практ. конф. (4-5 червня 2020 р., м. Херсон) / за заг. ред. Г.О. Райко., м. Херсон, 2020. С. 226-227.* Режим доступу до ресурсу:

<http://kntu.net.ua/ukr/content/download/76712/444549/file/YPWMT-2020.pdf#page=277>.

13. Єгошкін Д.І., Гук Н.А. Адаптивна редукція нечітких правил відносно навчальної вибірки. *Математичне та програмне забезпечення інтелектуальних систем (MSSIS-2022): Матеріали XX ювілейної міжнародної науково-практичної конференції, 23-25 листопада 2022.*, м. Дніпро, 2022. С. 83-84. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2022/12/MPZIS-2022-1.pdf#page=84>.
14. Єгошкін Д.І., Гук Н.А. Адаптивний алгоритм редукції нечіткої бази правил на основі даних навчальної вибірки. *Збірник наукових праць «Питання прикладної математики і математичного моделювання»*. Дніпро, 2022. Випуск 22. С. 49-59. doi: <https://doi.org/10.15421/322205>. Режим доступу до ресурсу: <https://pmm.dn.dp.ua/index.php/pmmm/article/view/337>.
15. Єгошкін Д.І., Гук Н.А. Алгоритм класифікації на базі нечіткої логіки з динамічною базою знань. *Математичне та програмне забезпечення інтелектуальних систем (MPZIS-2018): Матеріали XVI міжнародна науково-практична конференція, 21-23 листопада 2018.*, м. Дніпро, 2018. С. 67-68. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2021/06/mpzis-2018.pdf#page=67>.
16. Єгошкін Д.І., Гук Н.А. Алгоритм класифікації на базі нечіткої логіки з розширюваною кількістю виводів. *Збірник наукових праць «Питання прикладної математики і математичного моделювання» Випуск 18.*, м. Дніпро, 2018. С. 66-75. doi: <https://doi.org/10.15421/321807>. Режим доступу до ресурсу: <https://pmm.dn.dp.ua/index.php/pmmm/article/view/224>.
17. Єгошкін Д.І., Гук Н.А. Аналіз коректності нечіткої бази правил в системах логічного виведення. *Математика та інформаційні технології. Матеріали міжнародної наукової конференції, присвяченої*

- 55-річчю факультету математики та інформатики, 28–30 вересня 2023 р. – Чернівці: Чернівецький нац. ун-т, 2023. С. 193-194. Режим доступу до ресурсу: <https://fmi.chnu.edu.ua/media/ghufs0d5/materialy-mizhnorodnoi-naukovoi-konferentsii-fmi55.pdf#page=193>.
18. Єгошкін Д.І., Гук Н.А. Вплив розміру навчальної вибірки на якість класифікації нечіткої експертної системи. *Математичне та програмне забезпечення інтелектуальних систем (МПЗІС-2023): Матеріали XXI міжнародна науково-практична конференція до 105-річчя Дніпровського національного університету імені Олеся Гончара, 22-24 листопада 2023., м. Дніпро, 2023. С. 122-123. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2023/11/mpzis-2023.pdf#page=122>.*
19. Єгошкін Д.І., Гук Н.А. Налаштування та навчання нечіткої моделі для задачі класифікації. *Вісник Запорізького національного університету. Серія фізико-математичні науки.*, м. Запоріжжя: Видавничий дім «Гельветика», 2021. Вип. 1. С. 33-43. doi: <https://doi.org/10.26661/2413-6549-2021-1-04>. Режим доступу до ресурсу: <http://journalsofznu.zp.ua/index.php/comp-science/article/view/2286>.
20. Єгошкін Д.І., Гук Н.А. Перевірка коректності бази знань та адекватності нечіткої моделі. *Математичне та програмне забезпечення інтелектуальних систем (MSSIS-2021): Матеріали XIX міжнародної науково-практичної конференції, 17-19 листопада 2021., м. Дніпро, 2021. С. 75-77. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2021/12/mpzis-2021.pdf#page=75>.*
21. Іванченко Н.О. Представлення системи нечіткого логічного висновку та активації нечітких правил бази знань доменного простору економічної безпеки підприємства. *Науковий вісник Херсонського державного університету. Серія: економічні науки*, Херсон, 2014. Вип.

- 9(6). С. 223-227. URL: http://www.ej.kherson.ua/journal/economic_09/economic_09_6.pdf.
- 22.Кавин О., Кавин С., Кавин Б., Кавин Я. Застосування інтерактивного програмного середовища для оцінки ризиків інформаційної безпеки. *Scientific Collection "InterConf"*, 2022. Vol. 123. С. 312–319. URL: <https://archive.interconf.center/index.php/conference-proceeding/article/view/1280>.
- 23.Клебанова Т.С., Рудаченко О.О. Особливості використання нейронечітких моделей у фінансовій діяльності комунальних підприємств. *Економічний вісник Запорізької державної інженерної академії*. 2016. Вип. 5(1). С. 108-112.
- 24.Клебанова Т.С., Чаговец Л.О., Панасенко О.В. Нечітка логіка та нейронні мережі в управлінні підприємством: монографія. Харків: ІНЖЕК. 2011. 240 с.
- 25.Кондратенко Ю. П., Козлов А. В. Генерація баз правил нечітких систем на основі модифікованих мурашиних алгоритмів. *Міжнародний наук.-техн. журнал «Проблеми керування та інформатики»*, 2019. № 2. С. 59-79.
- 26.Кондратенко Ю.П., Кондратенко Г.В., Мельник Т.Л. Синтез та аналіз властивостей нечітких регуляторів для стабілізації температури в судових приміщеннях. *Автоматизація судових технічних засобів*, Одеса: ОНМА, 2007. Вып. 12. С. 53-67.
- 27.Котов І. А. Семантичні мережі як компонент уніфікованої моделі подання знань в інтелектуальних системах. 2012. Vol. 5(2). С. 21-22. URL: http://nbuv.gov.ua/UJRN/Tatrv_2012_5%282%29_11.
- 28.Кривуля Г. Ф., Шкиль А. С., Кучеренко Д. Е. Анализ корректности продукционных правил в системах нечеткого логического вывода с использованием квантовых моделей. *АСУ и приборы автоматизации : всеукр. межвед. науч.-техн. сб.*, 2013. Вып. 165, С. 42–53. URL: <http://openarchive.nure.ua/handle/document/11830>.

29. Кучеренко Е. И., Глушенкова И. С., Глушков С. А. Нечеткое разбиение объектов на основе критериев плотности. *Радіоелектроніка, інформатика, управління*. 2016. № 1 (36). С. 32-39. DOI:10.15588/1607-3274-2016-1-4.
30. Лещинський О. Л. Фрейм-підхід до викладання математичних дисциплін для студентів комп'ютерно-орієнтованих спеціальностей ВНЗ I-II рівнів акредитації. *Науковий часопис Національного педагогічного університету імені М. П. Драгоманова. Серія 3 : Фізика і математика у вищій і середній школі*. 2014. Вип. 13. С. 74-89. URL: http://nbuv.gov.ua/UJRN/Nchnpu_3_2014_13_12.
31. Липаев В.В. Надежность и функциональная безопасность комплексов программ реального времени, М., 2013. 207 с.
32. Мельникова Н.І., Стебліна К.В. Особливості проектування систем підтримки лікувальних рішень. *Математичні машини і системи*, 2014. № 1. С. 92-100.
33. Михайленко В.С., Харченко Р.Ю. Синтез адаптивного нечіткого регулятора з прогнозувальною нейронечіткою мережею. *Одеська державна академія холоду*, м. Одеса; Вісник СумДУ. Серія "Технічні науки". 2012. №3. С. 30-37.
34. Моркун В.С, Котов І.А, Сердюк О.Ю, Гапоненко І.А. Подання знань в інтелектуальних системах автоматизації керування енергосистемами гірничо-металургійного комплексу в умовах невизначеності. *Вісник Східноукраїнського національного університету імені Володимира Даля*, 2021. С. 40-48. DOI: 10.33216/1998-7927-2021-268-4-40-48.
35. Нариньяни А., Яхно Т. Продукционные системы. Представление знаний в человекомашинных и робототехнических системах. М.: ВЦ АН СССР, ВИНТИ, 1984. С. 136-177.
36. Осієвський С., Третьак В., Кулагін К., Власов А., Закіров З., Кривчун В. Метод підвищення ефективності функціонування людино-машинної системи за рахунок підвищення якості програмного забезпечення

- систем підтримки прийняття рішень. *Грааль науки*. 2021. С. 170-181. DOI: 10.36074/grail-of-science.25.06.2021.029.
37. Пташко Е. А., Ухоботов В. И. Автоматическая генерация нечетких правил для управления мобильным роботом с гусеничным шасси на основе числовых данных. *Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика*. 2017. Vol. 6(3). С. 60-72.
38. Субботін С.О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень. *Навчальний посібник*. Запоріжжя: ЗНТУ, 2008. 341 с. ISBN: 978–966–7809–87–4.
39. Судаков Б.М., Михайлова Ю.М. Лінгвістичний препроцесор обробки природно-мовних текстів для експертних систем. *Вісник НТУ "ХПІ". Серія: Інформатика та моделювання*. Харків: НТУ "ХПІ". 2012. Vol. № 38. С. 178 – 183.
40. Терновой М. Ю., Штогрина Е. С. Формальная спецификация свойств баз нечетких знаний Мамдани на основе метаграфа. *Вісник ХНУ імені В. Н. Каразіна. Серія: Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*, 2015. Вип. 27, С. 157-171.
41. Федорук П.І., Масловський С.М. Модель представлення знань в адаптивній системі дистанційного навчання та контролю знань «EduPRO». *Штучний інтелект*, 2011. 3. С. 463-472. URL: <http://dspace.nbu.gov.ua/handle/123456789/60073>.
42. Хижа А.Л., Высокопоясний И.Г. Автоматическая проверка семантической правильности решений задач по программированию. *Питання прикладної математики і математичного моделювання*. 2017. Вип. 17. С. 234-246. ISSN 2074-5893. doi: 10.15421/321727. URL: http://nbuv.gov.ua/UJRN/Ppmmm_2017_17_29.
43. Шаповалова С.І., Формалізація представлення продукційних правил в Erlang. *Математичне та комп'ютерне моделювання*. Серія: Технічні

- науки: зб. наук. пр. Кам'янець-Подільський: Кам'янець-Подільськ. нац. ун-т, 2020. Вип. 21. С. 125-139. DOI: 10.32626/2308-5916.2020-21.125-139.
- 44.Штовба С.Д., Мазуренко В.В., Савчук Д.А. Генетичний алгоритм вибору правил нечіткої бази знань, збалансованої за критеріями точності та компактності. *Наукові праці Вінницького національного технічного університету*, 2012. № 3. С. 1-9.
 - 45.Agarwal A., Aggarwal A., Agarwal A. An Approach for Augmenting Selection Operators of SQL Queries using Skyline and Fuzzy-Logic Operators. *Procedia Computer Science*. 2017, Vol. 115, P. 14-21, DOI: 10.1016/j.procs.2017.09.071.
 - 46.Alturki F. A. A rule reduction methodology for fuzzy logic systems using modified orthogonal techniques. *Department of Electrical Engineering, King Saud University, P.O.Box 800, Riyadh 11421, Saudi Arabia*. 2007, 19 p.
 - 47.Askari S. A novel and fast MIMO fuzzy inference system based on a class of fuzzy clustering algorithms with interpretability and complexity analysis. *Expert Systems with Applications: An International Journal*. 2017, Vol. 84(C), P. 301–322, DOI: <https://doi.org/10.1016/j.eswa.2017.04.045>.
 - 48.Ayman M., Al A. A Prototype Student Advising Expert System Supported with an Object-Oriented Database. *International Journal of Advanced Computer Science and Applications*. 2011, P. 100-105, DOI: 10.14569/SpecialIssue.2011.010316.
 - 49.Bazilevych, K.O., Chumachenko, D.I., Hulianytskyi, L.F. , Meniailov I. S. , Yakovlev S. V. Intelligent Decision-Support System for Epidemiological Diagnostics. II. Information Technologies Development, *Cybern Syst Anal*, 2022, Vol. 58(3), P. 499–509. DOI: <https://doi.org/10.1007/s10559-022-00484-9>

50. Bellman R., Kalaba R., Zadeh L. Abstraction and pattern classification. *Journal of Mathematical Analysis and Applications*. 1966. Vol. 13(1). P. 1–7.
51. Bickley S., Chan H. F., Schmidt S., Torgler B. Quantum-sapiens: the quantum bases for human expertise, knowledge, and problem-solving. *Technology Analysis and Strategic Management*, 2021. 33. P. 1290-1302. DOI: 10.1080/09537325.2021.1921137.
52. Biswas S. S. I-v Fuzzy Shortest Path in a Multigraph. *Oriental Journal of Computer Science and Technology*, 10, 2017. P. 364-370. DOI: 10.13005/ojcs/10.02.16.
53. Bojan-Dragoş C.A., Hedrea E.L., Precup R.E., Szedlak-Stinean A.I., Roman R.C. MIMO Fuzzy Control Solutions for the Level Control of Vertical Two Tank Systems. Department of Automation and Applied Informatics, Politehnica University Timisoara, Bd. V. Parvan 2, Timisoara, Romania, 2019. P. 810–817.
54. Borys, R.M., Martsenyuk. V.P. Classification Algorithm Polytrauma by Induction of Decision Trees. *Medical Informatics and Engineering*, no. 2 (November), 2013, P. 12-17. DOI: <https://doi.org/10.11603/mie.1996-1960.2013.2.1693>.
55. Brachman R.J., Levesque H.J., Knowledge Representation and Reasoning. *Morgan Kaufmann*, 2004. 604 p. ISBN-10: 1558609326.
56. Brachman R.J., Levesque H.J. Readings in Knowledge Representation. *Morgan Kaufmann Pub*, 1985. 571 p. ISBN-10: 978-0934613019.
57. Burda M., Stepnicka M. Reduction of Fuzzy Rule Bases Driven by the Coverage of Training Data. *Proceedings of the 2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology, Advances in Intelligent Systems Research*, Atlantis Press, 2015. Vol. 67. P. 463-470. DOI: <https://doi.org/10.2991/ifsa-eusflat-15.2015.67>.

- 58.Castellano G., Castiello C., Fanelli A.M., Mencar C. Knowledge discovery by a neuro-fuzzy modeling framework. *Computer Science Department, University of Bari, Via Orabona, 4, Bari 70126, Italy.* 2005. No. 149. P. 187-207.
- 59.Chaudhari T., Patel V., Thakkar R., Singh C. Comparative analysis of Mamdani, Larsen, and Tsukamoto methods of fuzzy inference system for students' academic performance evaluation. *International Journal of Science and Research Archive*, 2023. Vol. 9. P. 517-523. DOI: 10.30574/ijrsra.2023.9.1.0443.
- 60.Chen Y., Mazlack L. J., Minai A. A., Lu L. J. Inferring causal networks using fuzzy cognitive maps and evolutionary algorithms with application to gene regulatory network reconstruction. *Applied Soft Computing*, Volume 37, 2015. P. 667-679. ISSN 1568-4946. DOI: <https://doi.org/10.1016/j.asoc.2015.08.039>.
- 61.Chumachenko D., Meniaillov I., Bazilevych K., Chumachenko T., Yakovlev S. Investigation of Statistical Machine Learning Models for COVID-19 Epidemic Process Simulation: Random Forest, K-Nearest Neighbors, Gradient Boosting. *Computation*. 2022, 10(86), 22 p. DOI: <https://doi.org/10.3390/computation10060086>
- 62.Chicho B., Abdulazeez A., Zeebaree D., Zebari D. Machine Learning Classifiers Based Classification For IRIS Recognition. *Qubahan Academic Journal*, 2021. Vol. 1(2), P. 106–118. DOI: 10.48161/qaj.v1n2a48.
- 63.Damousis Y., Alexiadis M., Theocharis J., Dokopoulos P. A Fuzzy Model for Wind Speed Prediction and Power Generation in Wind Parks Using Spatial Correlation. *Energy Conversion, IEEE Transactions on* 19, 2004. P. 352-361. DOI: 10.1109/TEC.2003.821865.
- 64.Dandea V, Gheorghe G. Expert System Integrating Rule-Based Reasoning to Voltage Control in Photovoltaic-Systems-Rich Low Voltage Electric Distribution Networks: A Review and Results of a Case Study. *Applied Sciences*, 2023. Vol. 13. 6158. DOI: 10.3390/app13106158.

- 65.Darwiche A., Marquis P. A Knowledge Compilation Map. *Journal of Artificial Intelligence*, 2002. Vol. 17. P. 229-264. DOI: <https://doi.org/10.1613/jair.989>.
- 66.Gries D. The Science of Programming. *Monographs in Computer Science*, Springer New York Inc, 1987. 403 p.
- 67.Detlefs D., Nelson G., Saxe J. B. Simplify: A Theorem Prover for Program Checking. Hewlett-Packard Company, 2003. 122 p.
- 68.Dijkstra E. W. Guarded commands, nondeterminacy and formal derivation of programs. *Commun. ACM*, 1975. Vol. 18. P. 453–457. ISSN 0001-0782. DOI:10.1145/360933.360975.
- 69.Doneva R., Gaftandzhieva S., Pashev G., Totkov G. A Software Tool For Programming Training Through Accumulative Frame System. *International Journal of Scientific & Technology Research*, 2020. Vol. 9. P. 1389-1393.
- 70.Duda R.O., Hart P.E., Stork D.G. Pattern Classification, 2nd Edition. Wiley-Interscience. 2001. 688 p.
- 71.Dvoretzkyi M., Savchuk T., Fisun M., Dvoretzka S. Using the analytic hierarchy process with fuzzy logic elements to optimize the database structure. *Radio Electronics, Computer Science, Control*. 2022. Vol. 2. P. 101-112. DOI: 10.15588/1607-3274-2022-2-10.
- 72.Feigenbaum E., McCorduk P. The Fifth Generation. 1st ed., Reading, MA: Addison-Wesley, 1983. ISBN 978-0-201-11519-2. OCLC 9324691.
- 73.Fisher R. A. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 1936. Vol. 7. P. 179-188. URL: <http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x>
74. Franchuk V., Myroshnychenko M., Hnatjuk M., Kalyniuk N., Humenna N., Narizhna A., Franchuk U., Hladii O., Franchuk M. Implementation of the Decision Tree Method in Expert Analysis of the Medical Errors in Obstetric Practice. *Polski Merkurusz Lekarski*, 2023. P. 128-134. DOI: 10.36740/Merkur202302104.

- 75.Gorman K. B., Williams T. D., Fraser W. R. Ecological sexual dimorphism and environmental variability within a community of Antarctic penguins (genus *Pygoscelis*). *PLoS ONE*, 2014. 9(3), e90081. DOI: 10.1371/journal.pone.0090081.
- 76.Grissa-Touzi A., Ben Messaoud M. A. New Approach for Conception and Implementation of Object Oriented Expert System Using UML. *Int. Arab J. Inf. Technol.*, 2009. Vol. 6. P. 99-106.
- 77.Hoare C.A.R. An axiomatic basis for computer programming. *Communications of the ACM*, 1969. Vol. 12(10). P. 576–580 and 583.
- 78.Hudec M. An approach to fuzzy database querying, analysis and realization. *Comput. Sci. Inf. Syst.* 2009. Vol. 6. P. 127-140. DOI: 10.2298/CSIS0902127H.
- 79.Hudec M., Vujošević M. Selection and Classification of Statistical Data Using Fuzzy Logic. *Institute of Informatics and Statistics*, Bratislava, Slovakia. 2009. 11 p. URL: <https://api.semanticscholar.org/CorpusID:6130059>.
- 80.Jackson P. Introduction to Expert Systems. Addison-Wesley, 1998, 560 p. ISBN-10: 0201876868, ISBN-13: 978-0201876864.
- 81.Jain N., Sierra-Múnera A., Streit J., Thormeyer S., Schmidt P., Lomaeva M., Krestel R. Generating Domain-Specific Knowledge Graphs: Challenges with Open Information Extraction. *In Text2KG 2022: International Workshop on Knowledge Graph Generation from Text*, Co-located with the ESWC 2022, May 05-30, Crete, Hersonissos, Greece. 2022. 18 p. URL: https://ceur-ws.org/Vol-3184/TEXT2KG_Paper_4.pdf.
- 82.Jiang H., Chen R.C., Liu Q.E., Huang S.W. Fuzzy Rules Reduction Based on Sparse Coding. *International Journal of Applied Science and Engineering*, 2019. Vol. 16(3). P. 215-227.
- 83.Khan M., Quaddus M. Group Decision Support Using Fuzzy Cognitive Maps for Causal Reasoning: on GDN 2002 Papers (Guest Editors:

- Mohammed Quaddus and Des Klass). *Group Decision and Negotiation*. 2004. P. 13. DOI: 10.1023/B:GRUP.0000045748.89201.f3.
84. Kharya S., Soni S., Swarnkar, T. Fuzzy weighted Bayesian belief network: a medical knowledge-driven Bayesian model using fuzzy weighted rules. *International Journal of Information Technology*. 2023. Vol. 15. P. 1117–1125 DOI: 10.1007/s41870-022-01153-y.
85. Kim J., Han M., Lee Y., Park Y. Futuristic data-driven scenario building: Incorporating text mining and fuzzy association rule mining into fuzzy cognitive map. *Expert Systems with Applications*, 2016. 57 p. DOI: 10.1016/j.eswa.2016.03.043.
86. Kiseleva E.M., Prytomanova O.M., Hart L.L. Application of optimal set partitioning theory to solving problems of artificial intelligence and pattern recognition. *System Research and Information Technologies*, 2021, 2021(4), P. 91–101. DOI: https://doi.org/10.1007/978-3-031-37450-0_21
87. Koczy L.T., Hirota K. Size reduction by interpolation in fuzzy rule bases, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol 27 1997. P. 14-25.
88. Kondratenko Y.P., Kozlov O.V., Korobko O.V. Two modifications of the automatic rule base synthesis for fuzzy control and decision making systems. Chapter in a book: “Information processing and management of uncertainty in knowledge-based systems. Theory and foundations”. Medina J., Ojeda-Aciego M., Verdegay J.L., Pelta D.A., Cabrera I.P., Bouchon-Meunier B., Yager R.R. (Eds.). *Book series: Communications in computer and information science*, 854. Berlin. Heidelberg: Springer International Publishing. 2018. P. 570–582. DOI https://doi.org/10.1007/978-3-319-91476-3_47.
89. Kosko B. Fuzzy cognitive maps. *International Journal of Man-Machine Studies*, Volume 24, Issue 1, 1986. P. 65-75. ISSN 0020-7373. URL: [https://doi.org/10.1016/S0020-7373\(86\)80040-2](https://doi.org/10.1016/S0020-7373(86)80040-2).

- 90.Kosko B. Fuzzy systems as universal approximators. *IEEE Trans. on Computers*. 1994. Vol. 43, N 11. P. 1329–1333. DOI: 10.1109/12.324566
- 91.Kozlov O., Kondratenko Y., Skakodub O. Information Technology for Parametric Optimization of Fuzzy Systems Based on Hybrid Grey Wolf Algorithms. *SN Computer Science*, 2022. Vol. 3. 18 p. DOI:10.1007/s42979-022-01333-4 .
- 92.Kuncheva L.I. Fuzzy Classifier Design. Studies in Fuzziness and Soft Computing, Berlin - Heidelberg: Springer-Verlag. 2000. Vol. 49. 325 p.
- 93.Kwang H. L. First Course on Fuzzy Theory and Applications. *Advances in Intelligent and Soft Computing*. Springer Berlin, Heidelberg. 2005. P. 254-255. DOI: <https://doi.org/10.1007/3-540-32366-X>.
- 94.Larbi A., Mimoun M., Boukhalifa K. Modeling the Imprecision of Flexible Queries Using a Fuzzy SQL Language. *In: The 2nd International Conference on Software Engineering and New Technologies*, Hammamet, Tunisia. 2013. 9 p. ISBN: 978-9938-12-094-3.
- 95.Larsen H. L., Yager, R. R. A framework for fuzzy recognition technology. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 2000. 30(1), P. 65-76.
- 96.Lee D.H., Lee H. Construction of holistic Fuzzy Cognitive Maps using ontology matching method. *Expert Systems with Applications*, Volume 42, Issue 14, 2015. P. 5954-5962. ISSN 0957-4174, DOI: <https://doi.org/10.1016/j.eswa.2015.03.020>.
- 97.Liu B, Duan M, Zhao G. An Object Frame Knowledge Representation Approach for Fault Diagnosis Expert System. *Proceedings - 2011 International Conference on Future Computer Sciences and Application*, ICFCSA 2011, 2011. 74-77. DOI: 10.1109/ICFCSA.2011.24.
- 98.Lorenz F., Günther M. Expert Systems with Logic#. *A Novel Modeling Framework for Logic Programming in an Object-Oriented Context of C#*. 2022. P. 1-23. DOI: <https://doi.org/10.48550/arXiv.2205.07985>.

99. Mahini S. For Fuzzy Classification of Databases with Fuzzy Classification Query Language. *Digital Technologies Research and Applications*. 2022. Vol. 1(2). P. 25-34. DOI: 10.54963/dtra.v1i2.34.
100. Maia L., Lima G. Semantic-relations taxonomy for knowledge representation. *Brazilian Journal of Information Science: research trends*, 2021. Vol. 15. e02123. DOI: 10.36311/1981-1640.2021.v15.e02123.
101. Malabag B., Santiago Jr C., Cahapin E., Reyes J., Legaspi G. Fuzzy Logic-Based Size and Ripeness Classification of Banana using Image Processing Technique. *International Journal of Emerging Technology and Advanced Engineering*. 2022. Vol. 12. P. 11-18. DOI: 10.46338/ijetae1022_02.
102. Mamdani E.H, Assilian S. An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller, *Int. J. Man-Machine Studies*, 1975. 7, P. 1-13.
103. Mansyur R., Rahmat R., Ismail A., Kabit M. H. Decision support system for transport demand management: Object-oriented approach using Kappa PC 2.4 expert system shell. *ARPN Journal of Engineering and Applied Sciences*. 2011. P. 73-81.
104. Maylawati D., Darmalaksana W., Ramdhani M. Systematic Design of Expert System Using Unified Modelling Language. *IOP Conference Series: Materials Science and Engineering*, 2018. 012047. 288 p. DOI: 10.1088/1757-899X/288/1/012047.
105. Mesarovic D., Takahara Y. General Systems Theory: Mathematical Foundations. *Mathematics in Science and Engineering*, Academic Press, New York, San Francisco, London, 1975. Vol. 113. 316 p.
106. Mezhuyev V., Sorko S., Mayer, B., Lackner K. Development of an Expert System to Support the Decision-Making Process on the Shop Floor. 2023. DOI: 10.1007/978-3-031-23844-4_14.
107. Miller S.P., Whalen M.W., Cofer D.D. Software model checking takes off. *Commun. ACM*. 2010. № 53(2). P. 58-64.

108. Mordeson J., Josy A., Mathew S. Transactions On Fuzzy Sets And Systems, & Tfss, Systems. Weighted Graphs and Fuzzy Graphs. *Transactions on Fuzzy Sets and Systems*. 2022. Vol. 2. DOI: 10.30495/tfss.2022.1967246.1048.
109. Muliadi M., Budiman I., Sofyan A. The application of UML in the prototype of design expert system for the diagnosis of chili diseases and pests. *Klik - kumpulan jurnal ilmu komputer*, 2018. P. 214. DOI: 10.20527/klik.v5i2.175.
110. Neelimegham A., Bosukonda M., Kuruvilla N. A Nonlinear Fuzzy PID Controller via Algebraic Product AND-Maximum OR-Larsen Product Inference. *IFAC-PapersOnLine*, 2016. Vol. 49. P. 543-548. DOI: 10.1016/j.ifacol.2016.03.111.
111. Nguyen T., Perkins W., Laffey T., Pecora W. Checking Expert System Knowledge Bases for Consistency and Completeness. *In: Proceedings of the 9th International Joint Conference on AI*, Los Angeles. 1985. P. 375-378.
112. Onyshchenko V., Korolova Y., Nosyk A. Formation of a generalized semantic network of concepts. *Advanced Information Systems*, 2021. Vol. 5. P. 22-30. DOI: 10.20998/2522-9052.2021.3.04.
113. Papageorgiou E. I., Kannappan A. Fuzzy Cognitive Map Ensemble Learning Paradigm to Solve Classification Problems: Application to Autism Identification. *Applied Soft Computing*, Volume 12, Issue 12, 2012. P. 3798-3809. DOI: 10.1016/J.ASOC.2012.03.064.
114. Poojitha M. R. S., Malathi K. Decision Tree Over Support Vector Machine for Better Accuracy in Identifying the Problem Based on the Iris Flower. *Advances in Parallel Computing Algorithms, Tools and Paradigms*, 2022. P. 209-217. DOI: 10.3233/APC220028.
115. Rafanovic A.A. Expert systems development methodologies by means of object-oriented approach. *National Institutes of Health*, Bethesda, MD (USA), 2013. Vol. 2 (24). C. 54-58.

116. Rauch J, Šimůnek M. Learning Association Rules from Data through Domain Knowledge and Automation. *In: Bikakis A, Fodor P, Roman D (eds) Rules on the Web. From Theory to Applications. RuleML Lecture Notes in Computer Science*. Springer, Cham. 2014. Vol. 8620 DOI: 10.1007/978-3-319-09870-8_20.
117. Rovatti R., Guerrieri R., Baccarani G. An enhanced two-level Boolean synthesis methodology for fuzzy rules minimization. *IEEE Transactions on Fuzzy Systems*. 1995. Vol. 3(3). P. 288-299.
118. Russell S, Norvig P. Artificial Intelligence: A Modern Approach. 4th ed. Pearson, 2020. 1136 p. ISBN: 978-0134610993.
119. Salmeron J. L., Vidal R., Mena A. Ranking fuzzy cognitive map based scenarios with TOPSIS. *Expert Systems with Applications*, Volume 39, Issue 3, 2012. P. 2443-2450. ISSN 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2011.08.094>.
120. Saputra R., Khodra M. L., Maulidevi N. Frame-based expert system for monitoring diesel engine. 2015. P. 1-5. DOI: 10.1109/ICAICTA.2015.7335378.
121. Sciavicco G., Stan E. Knowledge Extraction with Interval Temporal Logic Decision Trees. 2020. P. 9:1–9:16. DOI: 10.4230/LIPIcs.TIME.2020.9.
122. Shtovba S. D., Pankevych O. D. Fuzzy Technology-Based Cause Detection of Structural Cracks of Stone Buildings. *Proc. of ICTERI-2018 - ICT in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer*, Kyiv. 2018, Vol. 2105. P. 209–218.
123. Siriak A.V., Turchyna V.A. (2020) Zero-knowledge proof of knowledge of multivariate polynomial's zero. *Питання прикладної математики і математичного моделювання*. Випуск 20, 2020, С. 153-162. DOI: <https://doi.org/10.15421/322015>

124. Stach W., Kurgan L., Pedrycz W. Reformat M. Genetic learning of fuzzy cognitive maps. *Fuzzy Sets and Systems*, 2005. 153(3), P. 371-401. DOI: <https://doi.org/10.1016/j.fss.2005.01.009>.
125. Stach W., Kurgan L., Pedrycz W. A divide and conquer method for learning large Fuzzy Cognitive Maps. *Fuzzy Sets and Systems*, Volume 161, Issue 19, 2010. P. 2515-2532. ISSN 0165-0114. DOI: <https://doi.org/10.1016/j.fss.2010.04.008>.
126. Stevens E., Antiga L., Viehmann T. Deep Learning with PyTorch. Manning Publications Co., Shelter Island, NY 11964. 2020. 520 p.
127. Subach I., Mykytiuk A. Methodology of formation of fuzzy associative rules with weighted attributes from SIEM database for detection of cyber incidents in special information and communication systems. *Collection Information technology and security*. 2023. Vol. 11. P. 47-59. DOI: 10.20535/2411-1031.2023.11.1.283575.
128. Sugiuraj A., Kosekij Y. Comprehensibility Improvement of Tabular Knowledge Bases. *Journal of the Japanese Society for Artificial Intelligence*, 1995. Vol. 10. Issue 4. P. 628-635. DOI: https://doi.org/10.11517/jjsai.10.4_628.
129. Sugiuraj A., Riesenhuberj M., Kosekij Y. Comprehensibility Improvement of Tabular Knowledge Bases. *AAAI-93 Proceedings*, 1993. P. 716-721. ISBN 978-0-262-51071-4.
130. Tsadiras, A., Bassiliades, N. RuleML representation and simulation of Fuzzy Cognitive Maps. *Expert Systems with Applications*, 40(5), 2013. P. 1413-1426. DOI: <https://doi.org/10.1016/j.eswa.2012.08.035>
131. Tsukamoto Y. An approach to fuzzy reasoning method, *in Advances in Fuzzy Set Theory and Applications*, Edited by M. M. Gupta, R. K. Ragade, and R. R. Yager, Amsterdam: North-Holland, 1979. P. 137–149.
132. Tufte E. R. The Visual Display of Quantitative Information. Second Edition, Graphics Press, Box 430, Cheshire, Connecticut, 2007. 191 p.

133. Wang H, Sheng C, Lu X. Knowledge-Based Control and Optimization of Blast Furnace Gas System in Steel Industry. *IEEE Access*. 2017. DOI: 10.1109/ACCESS.2017.2763630.
134. Yehoshkin D., Guk N. Automatic construction of a fuzzy system with a matrix representation of rules and a correct knowledge base. *Eastern-European Journal of Enterprise Technologies*, 2022. 6(4(120)). P. 14–22. DOI: <https://doi.org/10.15587/1729-4061.2022.268908>.
135. Zadeh L. A., Abbasov A. M., Yager R. R., Shahbazova S. N., Reformat M. Z. Recent developments and new directions in soft computing. *Eds. Studfuz 317*, Springer, 2014. P. 3-10. DOI: <https://doi.org/10.1007/978-3-319-06323-2>.
136. Zhang P. A Study on Database Fuzzy Query Method in SQL. *Procedia Engineering*. 2011. Vol. 24. P. 340-344. DOI: 10.1016/j.proeng.2011.11.2653.
137. Zhu J. Discussion on Redundant Processing Algorithm of Association Rules Based on Hypergraph in Data Mining. *Journal of Robotics*, 2022. P. 1-9. DOI: 10.1155/2022/1007464.

ДОДАТОК А. АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ РОБОТИ

ПОГОДЖЕНО

Проректор з наукової роботи Дніпровського національного університету імені Олеся Гончара

«__» _____ 2023 р. Олег МАРЕНКОВ

ЗАТВЕРДЖЕНО

В.о. першого проректора Дніпровського національного університету імені Олеся Гончара

«__» _____ 2023 р. Валентина СІЛІЧ-БАЛГАБАЄВА

АКТ

впровадження результатів роботи, поданої на здобуття наукового ступеня доктора філософії Єгошкіна Д. І. «Розробка методів і алгоритмів автоматичної генерації та дослідження структури нечіткої бази знань» в освітній процес Дніпровського національного університету імені Олеся Гончара

1. Вчена рада факультету прикладної математики у складі 17 осіб заслухала повідомлення асистента кафедри комп'ютерних технологій Єгошкіна Данили Ігоровича про результати наукового дослідження та їх використання в освітньому процесі кафедри комп'ютерних технологій.

2. Стисла характеристика дослідження:

В даний час експертні системи, що засновані на знаннях, набули широке застосування в різних прикладних областях науки і техніки для розв'язання таких задач як класифікація, прогнозування, розпізнавання образів, медична та технічна діагностика та інші. Процес генерування знань потребує значної аналітичної роботи з боку експерта та суттєво ускладнює розробку таких систем, тому підхід до автоматичної генерації бази правил та налаштуванню параметрів моделі на основі даних, що розробляється у дисертаційній роботі, є важливою науковою проблемою.

Впродовж дослідження було запропоновано продукційну модель представлення знань, що поєднує моделі Мамдані та Такагі-Сугено-Канг, розроблено автоматичне формування бази знань на основі навчальної вибірки, вдосконалено метод логічного виведення, що дозволяє розв'язувати задачі класифікації в умовах нечіткості та неповноти інформації, автоматизовано визначення лінгвістичних змінних, терм-множин, границь термів, параметрів функцій належності, вагових коефіцієнтів правил та кількості правил за допомогою розроблено методу редукції.

Варто відмітити, що удосконалення методів логічного виведення, налаштування параметрів, доведення коректності, здійснення редукції бази правил та дослідження якості побудованої системи робить системи логічного виведення більш ефективними та надійними для розв'язання задач.

У дослідженні було вивчено ефективність нечіткої продукційної моделі для розв'язання задач класифікації. Було розроблено нові ефективні алгоритми побудови та використання нечіткої продукційної моделі. Алгоритми були реалізовані на об'єктно-орієнтованих мовах програмування і протестовані на декількох наборах даних. Результати дослідження показали, що нечітка продукційна модель є ефективним інструментом для розв'язання задач класифікації.

3. Використання в освітньому процесу:

Результати дисертаційних досліджень впроваджено в освітній процес кафедри комп'ютерних технологій факультету прикладної математики ДНУ за спеціальністю 113 Прикладна математика ОП «Інформатика» під час викладання дисципліни «Методи Machine Learning». Окремі теоретичні результати було використано при виконанні курсових та дипломних робіт студентами факультету прикладної математики для розв'язання задач штучного інтелекту, класифікації, кластеризації та Data Mining.

4. Відомості про впроваджені об'єкти інтелектуальної власності:

1. Yehoshkin D., Guk N. (2022). Automatic construction of a fuzzy system with a matrix representation of rules and a correct knowledge base. *Eastern-European Journal of Enterprise Technologies*, 6(4(120)). P. 14–22. doi: <https://doi.org/10.15587/1729-4061.2022.268908>. (Scopus) (особистий внесок: аналітичний огляд, частковий підбір та опрацювання літератури, розроблені моделі та методи, програмна реалізація, аналіз результатів, формулювання висновків)

2. Єгошкін Д.І., Гук Н.А. (2022). Адаптивний алгоритм редукції нечіткої бази правил на основі даних навчальної вибірки. Збірник наукових праць «Питання прикладної математики і математичного моделювання». Дніпро, Вип. 22. С. 49-59. doi: <https://doi.org/10.15421/322205>. (Фахова, категорії Б) (особистий внесок: аналітичний огляд, частковий підбір та опрацювання літератури, розроблені моделі та методи, програмна реалізація, аналіз результатів, формулювання висновків)

3. Єгошкін Д.І., Гук Н.А. (2021). Налаштування та навчання нечіткої моделі для задачі класифікації. Вісник Запорізького національного університету. Серія фізико-математичні науки., м. Запоріжжя: Видавничий дім «Гельветика», Вип. 1. С. 33-43. doi: <https://doi.org/10.26661/2413-6549-2021-1-04>. (Фахова, категорії Б) (особистий внесок: аналітичний огляд, частковий підбір та опрацювання літератури, розроблені моделі та методи, програмна реалізація, аналіз результатів, формулювання висновків)

4. Єгошкін Д.І., Гук Н.А., Сірик С.Ф. (2018). Алгоритм класифікації на базі нечіткої логіки з розширеною кількістю виводів. Збірник наукових праць «Питання прикладної математики і математичного моделювання», м. Дніпро, Вип. 18. С. 66-75. doi: <https://doi.org/10.15421/321807>. (Фахова,

категорії Б) (особистий внесок: аналітичний огляд, частковий підбір та опрацювання літератури, розроблені моделі та методи, програмна реалізація, аналіз результатів, формулювання висновків)

5. Пропозиції ради:

Запропоновано впровадити результати дисертаційної роботи Єгошкіна Данили Ігоровича «Розробка методів і алгоритмів автоматичної генерації та дослідження структури нечіткої бази знань» в освітній процес Дніпровського національного університету імені Олеся Гончара.

Голова вченої ради

факультету прикладної математики



Олена КІСЕЛЬОВА

Секретарка



Наталія ЛИСИЦЯ

ДОДАТОК Б. ВИБІР МОВИ ПРОГРАМУВАННЯ

Для проведення складних математичних обчислень існує багато мов програмування. Такі спеціалізовані мови як C, C++, Fortran, R, Matlab та Maple – дозволяють отримувати достатньо швидкі та якісні результати.

Основними критеріями вибору мови програмування були наступні:

- 1) Створення складних шаблонних функцій та класів, з можливістю динамічної зміни типів даних;
- 2) Мова з можливістю використання вкладених класів;
- 3) Мова з можливістю реалізації агрегації та композиції всередині класів – це достатньо зручно для реалізації лінгвістичної змінної, та терм множини;
- 4) Наявність графічних бібліотек для побудови графіків та інтерфейсу;
- 5) Вміння зчитувати файли з експериментальними даними з віддаленого серверу та локально з файлів. Підтримка різних форматів навчальних даних;
- 6) Можливість швидкої реалізації готового прототипу коду – це необхідно, бо виходячи з задачі складність коду буде достатньо високою з великою кількістю класів та взаємозв'язків між класами;
- 7) Garbage collection для автоматичного керування оперативною пам'яттю комп'ютера під час виконання програми;
- 8) Наявність сучасних бібліотек для вирішення задач з математики та статистики.
- 9) Відкрита та безкоштовна реалізація бібліотек, компіляторів та інтерпретаторів мови програмування.

Спочатку була обрана мова програмування C++, але через відсутність певних властивостей описаних вище, а також динамічної типізації даних з часом проект було переписано на іншу мову. Наступною було обрано мову JavaScript за допомогою неї було реалізовано декілька досліджень. Проте мова JavaScript в першу чергу створювалася для розробки сайтів та різних

веб додатків, тому мала не таку велику кількість бібліотек для математичних обчислень та статистики. Окрім того процес навчання ускладнювала швидкість інтерпретаторів мови. Після цього розробку проекту було перенесено на мову програмування Python. Python – інтерпретована об'єктно-орієнтована мова програмування високого рівня з динамічною типізацією. Розроблена в 1990 році Гвідо ван Россумом.

Мова Python все більше домінує у світі науково-дослідних обчислень. Мова Python відносно нова мова програмування, яка підкорила світ, особливо інженерний, науковий та математичний. Враховуючи індекс Тіобе посідає перше місце на серпень 2023 року (див. рис. А.1). Індекс ТІОВЕ (TIOBE programming community index) — індекс, що оцінює популярність мов програмування, на основі підрахунку результатів пошукових запитів, що містять назву мови (запит виду + "<language> programming").














Aug 2023	Aug 2022	Change	Programming Language	Ratings	Change
1	1		 Python	13.33%	-2.30%
2	2		 C	11.41%	-3.35%
3	4	▲	 C++	10.63%	+0.49%
4	3	▼	 Java	10.33%	-2.14%
5	5		 C#	7.04%	+1.64%
6	8	▲	 JavaScript	3.29%	+0.89%
7	6	▼	 Visual Basic	2.63%	-2.26%
8	9	▲	 SQL	1.53%	-0.14%
9	7	▼	 Assembly language	1.34%	-1.41%
10	10		 PHP	1.27%	-0.09%
11	21	▲	 Scratch	1.22%	+0.63%
12	15	▲	 Go	1.16%	+0.20%
13	17	▲	 MATLAB	1.05%	+0.17%

Рис. А.1. Індекс ТІОВЕ programming community index – серпень 2023 року.

Мова Python стала однією з найпопулярніших мов програмування, використовується в аналізі даних, машинному навчанні, DevOps та веб-розробці, а також в інших сферах, включаючи розробку ігор. За рахунок читабельності, простого синтаксису та відсутності необхідності в компіляції

мова добре підходить для навчання програмуванню, дозволяючи сконцентруватися на вивченні алгоритмів, концептів та парадигм. Налагодження та експериментування значною мірою полегшуються тим фактом, що мова інтерпретується. Застосовується мова багатьма великими компаніями, такими як Google та Facebook. Станом на серпень 2023 року Python посідає перше місце у рейтингу ТІОБЕ популярності мов програмування з показником 13,33%. "Мовою року" за версією ТІОБЕ Python оголошувався в 2007, 2010, 2018 та 2020 роках (див. рис. А.2).

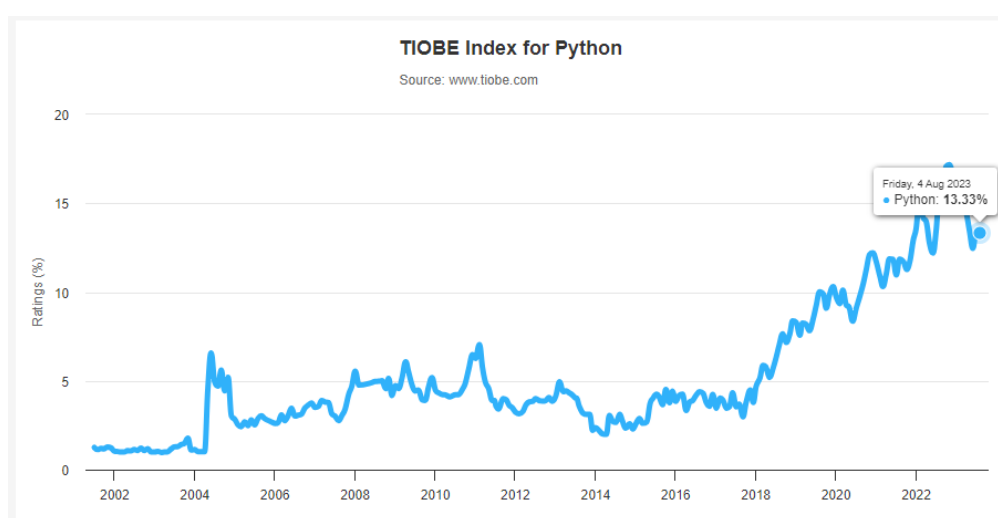


Рис. А.2. Індекс ТІОБЕ для Python – серпень 2023 року.

Також варто розглянути ліцензію мови Python, а саме Python Software Foundation License – BSD-подібна пермісивна ліцензія на вільне ПЗ, сумісна з GNU General Public License (GPL). Тобто, сама мова є вільним ПЗ, що дозволяє розвивати цю мову кожному, хто бажає.

Ще однією перевагою є велика спільнота, а також велика наявність бібліотек – пакетів (packages). Більшість таких пакетів є безкоштовними, що ще більше прискорює розвиток цієї мови та застосування її у різних сферах людської діяльності.

Розглянемо декілька пакетів для роботи з математичними об'єктами, а саме пакети для наукових обчислень: NumPy, SymPy, SciPy, Matplotlib та інші.

Matplotlib – це бібліотека для створення статичних, анімованих та інтерактивних візуалізацій на Python. Дозволяє будувати графіки, робити інтерактивні фігури, які можна масштабувати, панорамувати, оновлювати. Дозволяє налаштовувати візуальний стиль і макет. Може експортувати отримані результати в різні формати файлів для подальшого опублікування.

NumPy – є основним пакетом для наукових обчислень на Python. Це бібліотека Python, яка надає багатовимірний об'єкт масиву, пропонує комплексні математичні функції, генератори випадкових чисел, підпрограми лінійної алгебри, перетворення Фур'є тощо. NumPy підтримує широкий спектр апаратних і обчислювальних платформ і добре працює з розподіленими, графічними процесорами та бібліотеками розріджених масивів.

SciPy – пакет мови Python, що надає доступ до алгоритмів задач оптимізації, інтегрування, інтерполяції, задач на власні значення, алгебраїчних рівнянь, диференціальних рівнянь, статистики та багатьох інших класів задач.

SymPy призначений для символічної математики, тобто роботи з алгеброю та формулами у символічному вигляді. NumPy та SciPy призначені для роботи з фактичними даними. Якщо ви проводите аналіз даних, слід використовувати NumPy та SciPy. Якщо ви займаєтеся теоретичною математикою та фізикою слід використовувати SymPy.

Фактично дані пакети розв'язують задачі одного роду, але по-різному NumPy та SciPy вирішують задачі чисельно, а SymPy символічною математикою, шляхом апарату математичного висновку.

Мова програмування Python відповідає вищенаведеним вимогам через свою розширювану та динамічну природу, а також широкий вибір бібліотек, які допомагають розробникам вирішувати різноманітні задачі. Python дозволяє реалізовувати шаблони за допомогою концепції метакласів і динамічного створення класів і об'єктів, що надає гнучкість для розробки складних структур даних і класів. Python підтримує вкладені класи, що

дозволяє створювати складні структури даних з внутрішніми класами, покращуючи організацію коду. Python має багато бібліотек для графіки, таких як Matplotlib, Seaborn, PyQt, та інші, що дозволяють розробникам будувати графіки та інтерфейси. Python є інтерпретованою мовою, що дозволяє розробникам швидко створювати та тестувати код без необхідності компіляції. Python автоматично керує пам'яттю, що спрощує процес розробки та унеможлиблює витoki пам'яті. Python має багато бібліотек для вирішення математичних і статистичних задач, включаючи NumPy, SciPy, і багато інших. Мова Python та її інтерпретатори є вільним та відкритим програмним забезпеченням, доступним для використання безкоштовно, що робить її доступною для широкого кола розробників та вчених.

ДОДАТОК В. СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Yehoshkin D., Guk N. Automatic construction of a fuzzy system with a matrix representation of rules and a correct knowledge base. *Eastern-European Journal of Enterprise Technologies*, 2022. 6(4(120)). Р. 14–22. doi: <https://doi.org/10.15587/1729-4061.2022.268908>. Режим доступу до ресурсу: <http://journals.uran.ua/eejet/article/view/268908> (Scopus).

2. Єгошкін Д.І., Гук Н.А. Адаптивний алгоритм редукції нечіткої бази правил на основі даних навчальної вибірки. *Збірник наукових праць «Питання прикладної математики і математичного моделювання»*. Дніпро, 2022. Вип. 22. С. 49-59. doi: <https://doi.org/10.15421/322205>. Режим доступу до ресурсу: <https://pm-mm.dp.ua/index.php/pmmm/article/view/337>.

3. Єгошкін Д.І., Гук Н.А. Налаштування та навчання нечіткої моделі для задачі класифікації. *Вісник Запорізького національного університету. Серія фізико-математичні науки.*, м. Запоріжжя: Видавничий дім «Гельветика», 2021. Вип. 1. С. 33-43. doi: <https://doi.org/10.26661/2413-6549-2021-1-04>. Режим доступу до ресурсу: <http://journalsofznu.zp.ua/index.php/comp-science/article/view/2286>.

4. Єгошкін Д.І., Гук Н.А., Сірик С.Ф. Алгоритм класифікації на базі нечіткої логіки з розширюваною кількістю виводів. *Збірник наукових праць «Питання прикладної математики і математичного моделювання»*, м. Дніпро, 2018. Вип. 18. С. 66-75. doi: <https://doi.org/10.15421/321807>. Режим доступу до ресурсу: <https://pm-mm.dp.ua/index.php/pmmm/article/view/224>.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

5. Єгошкін Д.І., Гук Н.А. Вплив розміру навчальної вибірки на якість класифікації нечіткої експертної системи. *Математичне та програмне забезпечення інтелектуальних систем (МПЗІС-2023): Матеріали XXI міжнародна науково-практична конференція до 105-річчя Дніпровського національного університету імені Олеся Гончара, 22-24 листопада 2023.*, м.

Дніпро, 2023. С. 122-123. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2023/11/mpzis-2023.pdf#page=122>.

6. Єгошкін Д.І., Гук Н.А. Аналіз коректності нечіткої бази правил в системах логічного виведення. *Математика та інформаційні технології. Матеріали міжнародної наукової конференції, присвяченої 55-річчю факультету математики та інформатики, 28–30 вересня 2023 р.* – Чернівці: Чернівецький нац. ун-т, 2023. С. 193-194. Режим доступу до ресурсу: <https://fmi.chnu.edu.ua/media/ghufs0d5/materialy-mizhnorodnoi-naukovoi-konferentsii-fmi55.pdf#page=193>.

7. Єгошкін Д.І., Гук Н.А. Адаптивна редукція нечітких правил відносно навчальної вибірки. *Математичне та програмне забезпечення інтелектуальних систем (MSSIS-2022): Матеріали XX ювілейної міжнародної науково-практичної конференції, 23-25 листопада 2022., м. Дніпро, 2022. С. 83-84.* Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2022/12/MPZIS-2022-1.pdf#page=84>.

8. Єгошкін Д.І., Гук Н.А. The advantages of using MISO and MIMO models in fuzzy expert systems. *Прикладна математика та інформаційні технології ПМ&ІТ 2022: Тези доп. міжнародної наукової конференції присвяченої 60-річчю кафедри прикладної математики та інформаційних технологій, 22 – 24 вересня 2022., м. Чернівці, 2022. С. 170-172.* Режим доступу до ресурсу: <http://www.amit60.fmi.org.ua/files/AMIT2022-Materials.pdf#page=171>.

9. Єгошкін Д.І., Гук Н.А. Перевірка коректності бази знань та адекватності нечіткої моделі. *Математичне та програмне забезпечення інтелектуальних систем (MSSIS-2021): Матеріали XIX міжнародної науково-практичної конференції, 17-19 листопада 2021., м. Дніпро, 2021. С. 75-77.* Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2021/12/mpzis-2021.pdf#page=75>.

10. Єгошкін Д.І., Гук Н.А. The application of fuzzy logic in classification tasks. *Сучасні інформаційні та комунікаційні технології на транспорті, в промисловості та освіті: Матеріали XIV міжнародної*

науково-практичної конференції, 15.12.2020 – 16.12.2020., м. Дніпро, 2020. С. 69-70. Режим доступу до ресурсу:

<https://crust.ust.edu.ua/server/api/core/bitstreams/cf8adc16-fd69-42b1-a82f-cbc77dbcf2f/content#page=70>.

11. Єгошкін Д.І., Гук Н.А. The fuzzy logic classification algorithm with three dimensional membership functions and dynamic knowledge base. *Молодь у світі сучасних технологій за тематикою: Використання інформаційних та комунікаційних технологій в сучасному цифровому суспільстві: матеріали міжнар. наук.-практ. конф. (4-5 червня 2020 р., м. Херсон) / за заг. ред. Г.О. Райко., м. Херсон, 2020. С. 226-227. Режим доступу до ресурсу: <http://kntu.net.ua/ukr/content/download/76712/444549/file/YPWMT-2020.pdf#page=277>.*

12. Єгошкін Д.І., Гук Н.А. The classification algorithm on the basis of the fuzzy logic with dynamic knowledge-based system. *Сучасні науково-технічні дослідження у контексті мовного простору (іноземними мовами) 11-12 квітня 2019 року: матеріали VIII Регіональної науково-практичної конференції молодих учених та студентів., м. Дніпро, 2019. С. 83-84. Режим доступу до ресурсу: https://www.dnu.dp.ua/docs/ndc/202/such_nauk_teh_dosl.pdf#page=83.*

13. Єгошкін Д.І., Гук Н.А. Алгоритм класифікації на базі нечіткої логіки з динамічною базою знань. *Математичне та програмне забезпечення інтелектуальних систем (MPZIS-2018): Матеріали XVI міжнародна науково-практична конференція, 21-23 листопада 2018., м. Дніпро, 2018. С. 67-68. Режим доступу до ресурсу: <http://mpzis.dnu.dp.ua/wp-content/uploads/2021/06/mpzis-2018.pdf#page=67>.*