

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення

Пояснювальна записка

до магістерської роботи
на ступінь вищої освіти магістр

на тему: **«РОЗРОБКА СИСТЕМИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ
ОПТИМІЗАЦІЇ БАЗ ДАНИХ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ»**

Виконав: студент 6 курсу, групи ПДМ-61
спеціальності:

121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

Кобаєв Д.О

(прізвище та ініціали)

Керівник Негоденко О.В

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Нормоконтроль _____

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення _____

Ступінь вищої освіти - «Магістр» _____

Напрямок підготовки - 121 «Інженерія програмного забезпечення» _____

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

О.В. Негоденко

— _____ || _____ 2022 року

ЗАВДАННЯ НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Копасєв Данііл Олександрович

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка системи прийняття рішень для оптимізації баз даних на основі нейронних мереж»

Керівник роботи Негоденко О.В., кандидат технічних наук, доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від — 11 жовтня 2021 року №170

2. Строк подання студентом роботи 24 грудня 2021 _____
р.

3. Вихідні дані до роботи:

3.1 Вимоги до кваліфікаційної роботи магістра з актуальних завдань спеціальності;

3.2 Нормативні матеріали (стандарти, Гости);

3.3 Технічні вимоги;

3.4 Науково-технічна література з питань, пов'язаних з темою роботи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

4.1 Порівняльний аналіз результатів, отриманих іншими авторами;

4.2 Методика дослідження;

4.3 Результати дослідження;

4.4 Висновки

5. Перелік графічного матеріалу.

6. Дата видачі завдання 29 жовтня 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	29.10-07.11	Виконано
2	Аналіз математичного апарату нечіткої логіки	08.11-10.11	Виконано
3	Аналіз застосування парадигм нечіткої логіки в нейронних мережах	11.11-18.11	Виконано
4	Аналіз архітектури адаптивної нейро-нечіткої системи висновку (ANFIS)	19.11-28.11	Виконано
5	Розробка системи прийняття рішень для кредитного скорінгу застосуючи базу даних клієнтів	29.11-04.12	Виконано
6	Вступ, висновки, реферат	05.12-12.12	Виконано
7	Розробка обов'язкових матеріалів	29.11-15.12	Виконано
8	Попередній захист роботи	21.12	Виконано
9	Пред'явлення роботи в деканат	24.12	Виконано

Студент Копаєв Д.О _____
(прізвище та ініціали) (підпис)

Керівник роботи Негоденко О.В _____
(підпис)

РЕФЕРАТ

Текстова частина магістерської роботи : 80 с., 46 рис., 5 табл., 20 джерел.

НЕЙРОННІ МЕРЕЖІ, MATLAB, ANFIS, ГІБРИДНІ МОДЕЛІ, FUZZY CLUSTERING, НЕЙРО-НЕЧІТКІ МЕРЕЖІ, ECOS, НЕЙРО-НЕЧІТКЕ МОДЕЛЮВАННЯ, GARIC, NEFCON

Об`єкт дослідження – оптимізація баз даних на основі нейронних мереж.

Предмет дослідження- методи та алгоритми для розробки системи прийняття рішень на основі нейронних мереж

Мета роботи– підвищення ефективності слабоструктурованих процесів.

Методи дослідження — засоби моделювання на основі нейро-нечіткої моделі до слабоструктурованих процесів за допомогою системи прийняття рішень на основі нейронних мереж.

Проведено дослідження застосування нейронечіткого моделювання до слабоструктурованих процесів, моделлю ANFIS у середовищі MATLAB.

На основі результатів виконаних досліджень запропоновано до реалізації універсальний модель ANFIS для роботи з інформацією, де краще застосована нечітку логіку.

Отримані результати на основі моделювання кредитних ризиків показують, що застосування нейро-нечіткого моделювання за допомогою ANFIS, надає точне моделювання слабоструктурованого процесу.

Сфера застосування- прогнозування, прийняття рішень, розпізнавання образів та аналіз даних в промислових та розважальних сферах людської діяльності.

ЗМІСТ

ВСТУП.....	8
1.ІНФОРМАЦІЙНІ ПРОЦЕСИ І СПОСОБИ ЇХ МОДЕЛЮВАННЯ.....	10
1.1 Інформаційне суспільство.....	10
1.2 Інформаційні процеси	14
1.3 Методологія моделювання.	17
1.4 Нейро-нечітка класифікація об'єктів та їх станів.....	26
2. НЕЙРО-НЕЧІТКІ МЕТОДИ ФОРМАЛІЗАЦІЇ ІНФОРМАЦІЙНИХ ПРОЦЕСІВ.....	28
2.1 Нечітка логіка.....	28
2.2. Механізм нечіткого висновку.....	36
2.3 Штучний інтелект.....	43
2.4 Нейро-мережеве моделювання інформаційних процесів.....	52
3. НЕЙРО-НЕЧІТКІ МЕТОДИ І ЗАСОБИ МОДЕЛЮВАННЯ СЛАБОСТРУКТУРОВАНИХ ІНФОРМАЦІЙНИХ ПРОЦЕСІВ.....	58
3.1 Нейро-нечітке моделювання.....	58
3.2. Моделювання слабоструктурованих процесів.....	68
3.3. Опис роботи системи.....	79
ВИСНОВКИ.....	86
ПЕРЕЛІК ПОСИЛАНЬ.....	87
ДОДАТОК.....	89

ВСТУП

Обґрунтування вибору теми та її актуальність: З метою оптимізації та спрощення проведення операцій оптимізації баз даних з допомогою нейронних мереж було прийнято рішення розробити комплексну методику роботи з слабоструктурованими процесами. За допомогою розроблюваної методики можна було виконувати роботу по кредитного скорінгу, який допомагає банкам генерувати більше прибутків та зменшує ризики у разі дефолту.

Слабоструктурованими інформаційними процесами, називаються процеси, які перебувають у проміжку між структурованими та неструктурованими процесами. Слабоструктуровані процеси мають більшу гнучкість, ніж структуровані, тому що вони виконуються більш невизначено і їм доводиться мати справу із змінами та винятками. А, на відміну від неструктурованих, вони можуть описуватись досить формалізовано.

Тому розробка комплексного рішення проблеми є доцільною і залишається актуальною науковою роботою, адже від оптимізації баз даних залежить дуже багато ніж просто втрата грошей.

Наукова новизна роботи полягає в розробці методики, яка в комплексі буде використовувати нейро-нечіткі моделі такі як ANFIS та NEFCON до слабоструктурованих процесів, та їх оптимізації в базах даних використовуючи програмне забезпечення Matlab та Fuzzy Logic Toolbox.

Метою роботи є підвищення ефективності слабоструктурованих процесів.

Об'єкт дослідження – оптимізація баз даних на основі нейронних мереж.

Предмет дослідження – методи та алгоритми для розробки системи прийняття рішень на основі нейронних мереж.

Метод дослідження – засоби моделювання на основі нейро-нечіткої моделі до слабоструктурованих процесів за допомогою системи прийняття рішень на основі нейронних мереж.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- проаналізувати діючі алгоритми нечіткої логіки такі як модель Мамдані та Такагі-Сугено;
- дослідити сучасні проблеми слабоструктурованих процесів;
- проаналізувати існуючі архітектури нейронечітких мереж;
- розробити методику прийняття рішень для оптимізації баз даних на основі нейронних мереж.

Обсяг та структура дослідження: Магістерська робота складається з вступу, 3 розділів у сумі, що містять 11 розділів, виведення та списку використаної літератури.

Перший розділ говорить про найпростіші поняття інформації, інформаційного процесу та про узагальнене поняття методів моделювання, які можна застосувати до інформаційних процесів.

Другий розділ визначає нечітку логіку та її механізм виведення, нейронні мережі та їх моделі для моделювання.

Третій розділ описує застосування гібридного методу нечіткої логіки та нейронних мереж для моделювання слабоструктурованих процесів, розгляд нейро-нечітких моделей, застосування цих моделей у реальних процесах.

РОЗДІЛ 1. ІНФОРМАЦІЙНІ ПРОЦЕСИ І СПОСОБИ ЇХ МОДЕЛЮВАННЯ

1.1 Інформаційне суспільство

У нашому сьогоднішньому світі інформація відіграє важливу роль у будь-якій сфері діяльності. Інформація (переклад з лат. informatio уявлення, опис) є ключем у нинішній науці, і знаходиться поряд з наступними поняттями подібно до "енергія" і "матерія". Попри те що є багато понять "інформації", можна назвати кілька основних інтерпретацій поняття "інформація": наукова інтерпретація, абстрактна інтерпретація, конкретна інтерпретація.

У науковому трактуванні інформація є вихідна загальнонаукова категорія, яка відбиває як структурність об'єкта, і методи її розуміння, яка зводиться до легших положень. У абстрактної інтерпретації, вона сприймається як певний порядок символічних значень, які мають певним смисловим поняттям для виконув прийняття чи передачу, як узагальнено, і зокрема. У конкретній інтерпретації проглядаються певні виконуючі із застосуванням специфічних системних команд та семантичної мови. Наприклад, для тварин, людей це бачення, звуки; з боку технологій інформація це «0 та 1» і т.д. Є деякі концепції парадигм інформації.

1-е поняття це концепція Шеннона, яка відбиває кількісне напрям, що визначає інформацію як межу неточності – ентропію явища.

У другій її розглядають як властивість речовини (її атрибут). Така концепція виникла завдяки еволюції кібернетики і полягає в тому, що інформація містить будь-які повідомлення, які сприймаються і людьми, і приладами.

І нарешті третя формується на логіко-семантичному напрямі, у цій концепції інформація викладається як знання, причому така частина знання, яка застосовується з метою орієнтації, з метою активних дій, з метою контролю та самоконтролю. Іншими виявленнями інформація потрібна, важлива, дієва, "використовується" частина знань. (Концепція В.Г. Афанасьєва.)

У сучасні дні слово "інформація" має поглиблений зміст і численні погляди. Інформація також має властивості, і її можна розділяти на види.

Інформація має такі властивості:

- корисність;
- повнота та зрозумілість;
- достовірність;
- актуальність;
- доступність;
- захищеність;
- об'єктивність.

Корисність інформації - це властивість, яка говорить, наскільки вона є цінною в тій чи іншій сфері діяльності.

Повнота, зрозумілість якості, визначальне, представляється інформація повної, тобто. закінченою за змістом, і чи ясна вона одержує її.

Достовірність властивість інформації, що визначає її істинність, наскільки вона є достовірною для тієї сфери, в якій вона буде застосовуватися.

Актуальність і об'єктивність властивість, що визначає наскільки отримана інформація, буде придатною для теперішнього часу, чи вона втратила свою придатність.

Доступність і захищеність інформації визначають наскільки інформація секретна і доступна, і чи має вона захист від несанкціонованого доступу.

Інформацію можна також розділити на такі види, від способу прийняття цієї інформації:

- Візуальна інформація, яка досягається завдяки зору.
- Аудіальна інформація, яка сприймається завдяки слуху.
- Тактильна інформація досягається завдяки дотику об'єктів, предметів тощо.
- Смакова інформація досягається завдяки смаковим рецепторам.
- Нюхова інформація сприймається як запахів.

Всі ці види інформації виходять із способу прийняття інформації людьми. Дослідження цього наукою довело, що з допомогою зору слуху людина досягає близько 90% інформації у навколишньому середовищі, інші 10% інформації сприймається завдяки тактильним, смаковим і нюховим якостям. Людина, щоб отримати інформацію, яка є недоступною їй, завдяки якостям викладеним вище, винаходить різні пристрої та прилади: пристрої вимірювання температури і тиску, прилади спостереження, моніторингу і т.д. Людина приймає інформаційні сигнали завдяки органам почуттів, а приладах цю роль грають датчики.

Тому інформація щодо способу прийняття ділиться також наступні види: аналогова і цифрова (рису 1.1).

При надходженні інформації у вигляді органів чуття, вона приймається безперервно, і це вид називається аналогової. На пристосуваннях, роль прийняття інформації грають мікро-датчики, на відміну органів почуттів вони приймають її дискретно.

Але як і скрізь, і серед пристроїв є винятки, деякі з яких також мають справу з безперервною інформацією (ТВ, телефон і т.д.).

Також безперервну інформацію є можливість перетворити на цифрову, і цифрову на безперервну, для цього існують пристрої двох типів «Цифро-аналоговий перетворювач» та «Аналого-цифровий перетворювач».

Значення інформації в інформаційних процесах є основою. У будь-якій сфері діяльності людина використовує інформацію, проводячи її пошук, зберігання, передачу, обробку та виведення цієї інформації для виконання різних видів завдань у тій чи іншій сфері діяльності. Ми сказали, що інформація надходить до людини шляхом органів чуття та приладів.

З метою виконання зберігання та обробки інформації людина використовує мозок, у якому виконуються операції завдяки нейронам мозку. Людина виконуючи такі операції, наприклад написання чогось, розмовляючи з будь-ким, читаючи статті тощо, працює з інформацією. Операції з нею виконуються у людській сфері, а й у навколишньому середовищі. Все, що оточує нас, має зв'язок з інформацією, з її переробкою.

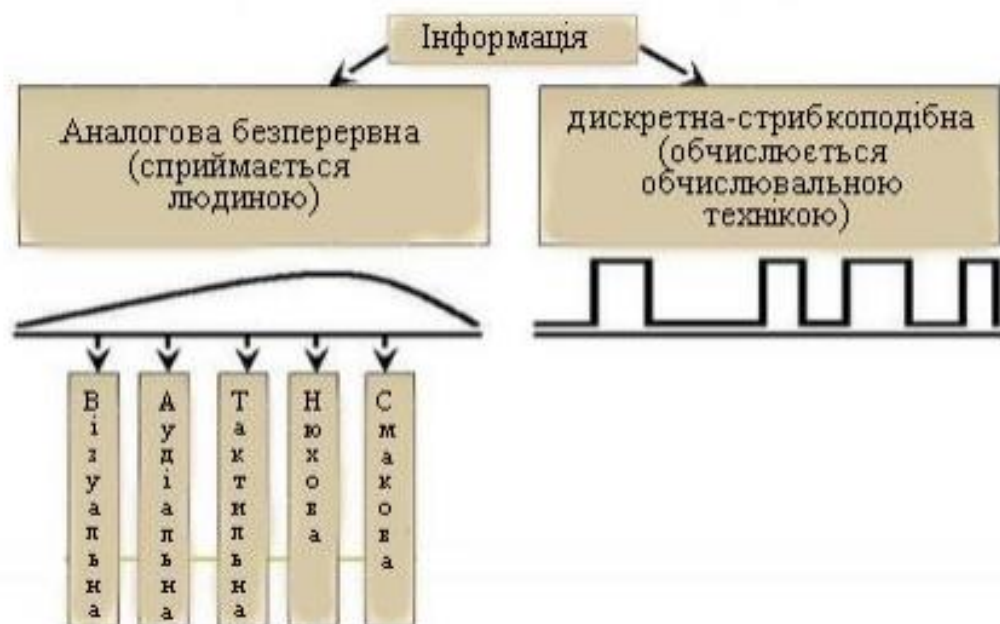


Рисунок 1.1. Аналогова та дискретна інформація.

1.2. Інформаційні процеси

Ми сказали, що ІІ – це процес пошуку, введення, обробки, передачі та подання інформації для виконання завдань за допомогою цієї інформації (її також можна назвати знаннями). Інформація – важливі факти, отримані із даних. Знання це інформація, яка була протестована та представлена у вигляді законів, теорій. ІІ це цикл створення інформації та її збереження у вигляді нових даних.

В інформаційний процес входять такі дії:

- Збір даних - це накопичення даних, щоб забезпечити його повноту для прийняття подальших рішень;
- Формування даних дані, отримані з різних джерел, перетворюються на однорідність;
- Фільтрування даних видаляє «зайві» дані, які не потрібні для ухвалення рішення;
- Сортування даних упорядковує дані за певною ознакою;
- Групування даних – це об'єднання даних за певними характеристиками;
- Архівування даних організує зберігання даних у зручній і доступній формі, зазвичай у більш економічному форматі;
- Захист даних – це набір заходів, спрямованих на запобігання втраті, модифікації або відтворенню даних.

Інформаційний процес можна розділити на кілька груп:

- пошук;
- зберігання;
- обробка;

- передача;
- уявлення.

Пошук процес знаходження інформації з джерел зберігання (книги, статті, журнали тощо), для виконання операцій та її застосування в майбутньому.

Зберігання процес, який потрібен, щоб зберегти інформацію, потім застосувати її в майбутньому. Засобами зберігання інформації можуть бути: книги, журнали, електронні носії, статті в інтернеті і т.д.

Обробляє процес виконання алгоритмічних дій з інформацією, для знаходження шуканої або нової інформації.

Передача інформації – це обмін інформацією між двома або декількома джерелами. При передачі обов'язково існує той, хто віддає і приймає інформацію, незалежно від того чи це людина чи щось.

Подання процес результату вирішення інформації, у тому вигляді в якому необхідно, для використання. Подання можливо у вигляді картинок, таблиць, графів, рівнянь, формул тощо.

ІІ із структуризації поділяються на:

- структуровані інформаційні процеси;
- слабоструктуровані інформаційні процеси;
- неструктуровані інформаційні процеси.

Структуровані ІІ це такі процеси, які формалізовані за допомогою кількісних даних, кількісної інформації. Структуровані ІІ легко піддаються програмуванню та математико- алгоритмічним діям.

Неструктуровані ІІ це повна протилежність до структурованих, будуються ці ІІ на основі якісних даних і дуже складні з метою вживання в сучасний час.

Слабоструктуровані ІІ мають і кількісні, так і якісні дані, але більшою мірою процеси протікають з якісними даними, і мають справу з погано формалізованими даними. Ці інформаційні процеси погано піддаються програмуванню, і мають справу з винятками та змінами даних у часі.

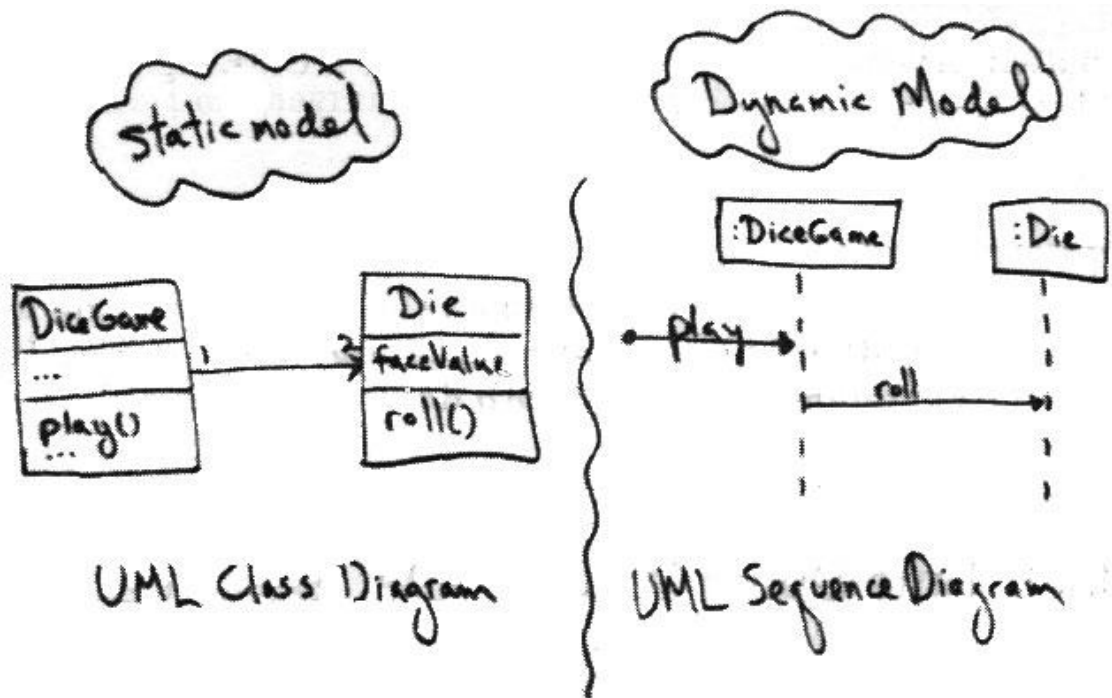


Рисунок 1.2. Статичне та динамічне моделювання.

Слабоструктуровані ІІ знаходяться між структурованими та неструктурованими, є дуже гнучкими по відношенню до структурованих.

Слабоструктуровані процеси сьогодні широко застосовуються у банківській сфері, де потоки даних, які втікають у банк, є якісними. Моделювання цих слабоструктурованих процесів є трудомістким процесом, що має справу з невизначеністю через уявлення саме цих якісних даних.

1.3 Методологія моделювання

Насамперед не приступаючи до методів і засобів моделювання, треба розуміти, що собою в першу чергу представляють терміни "model" і "modelling". Модель представлення подоби будь-якого предмета в простому вигляді, який має властивості та характеристики оригінального предмета.

Моделювання процес, із застосуванням подібності будь-якого предмета, з дослідженнями властивостей, зв'язків, якостей та параметрів оригінального предмета та застосуванням їх при прийнятті рішень та прогнозу станів у майбутньому.

Існує багато методів та засобів моделювання, які широко застосовуються у різних сферах діяльності. Modelling оригінальних предметів, подій відбувається для того, щоб прогнозувати їх дії в майбутньому. Поговоримо про класифікації моделювання.

Детерміноване моделювання.

Детерміновані моделі будуються з урахуванням математичних закономірностей, і з допомогою цих закономірностей результат виявляється точнішим. Детерміноване побудова (modeling) висвічує ті події, які мають випадкових впливів із боку.

При виведенні результату дослідження процесів за допомогою цього моделювання виводиться єдино точний результат.

Стохастичне моделювання

Стохастична модель складна при конструюванні, що складаються з декількох кроків. Прийоми будуються з урахуванням математичної статистики, і висуває можливості операцій. Завдяки стохастичній моделі виводяться кілька результатів досліджуваного предмета і відбувається оцінка середнього значення показників, тобто безліч однорідних результатів.

Стохастичне моделювання можна вважати, як доповнення до детермінованого моделювання, використовується в тих випадках, де є невизначеності, і не можна вдатися до детермінованого моделювання.

Статичне та динамічне моделювання.

Статична побудова (modeling) відображає показники подій у певний момент t або не враховує тимчасових факторів. Відображення показників у певний момент t називається статистичним зрізом, яке показано на рисунку 1.2.

Динамічна побудова (modeling) відображає показники в повний час (t), тобто весь цикл досліджуваної події.

Дискретне, безперервне та дискретно-безперервне моделювання.

Дискретне побудова (modeling) відображає показники процесу дискретний момент t , наприклад у проміжку $t_0 \dots t_1$.

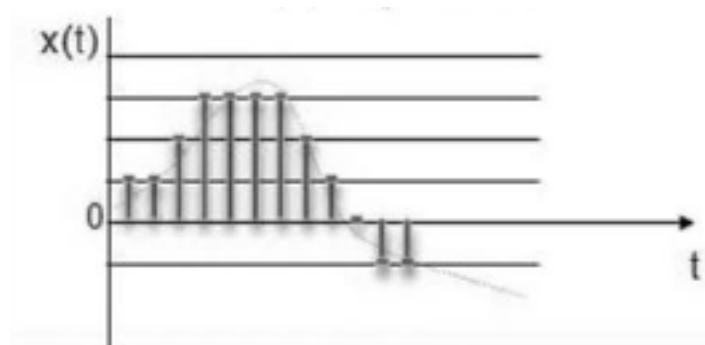


Рисунок 1.3. Дискретне моделювання.

Безперервна побудова (modeling) відображає показники процесу, що змінюються безперервно з плином t . Наприклад, стан процесу в момент часу $t_1, t_2 \dots t_n$.

Щоб представити показники процесу будь-якого предмета (процесу), як дискретно-подійно, і безперервно, застосовується дискретно-безперервне побудова (modeling).

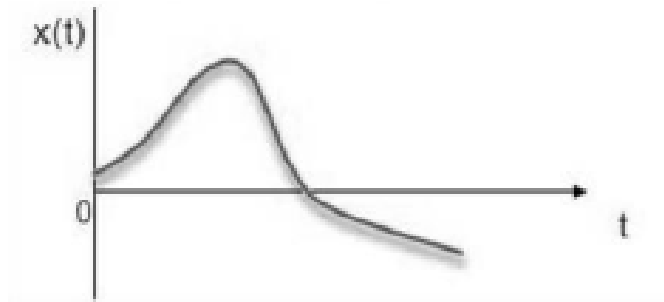


Рисунок 1.4. Безперервне моделювання.

Уявна побудова (modeling) називається таким моделюванням предметів і подій, які неможливо реалізувати на практиці в природних умовах із перебігом часу. Завдяки цьому моделюванню можна досліджувати процеси на мікрорівні, які не піддаються реалізації у фізично-експериментальних умовах.

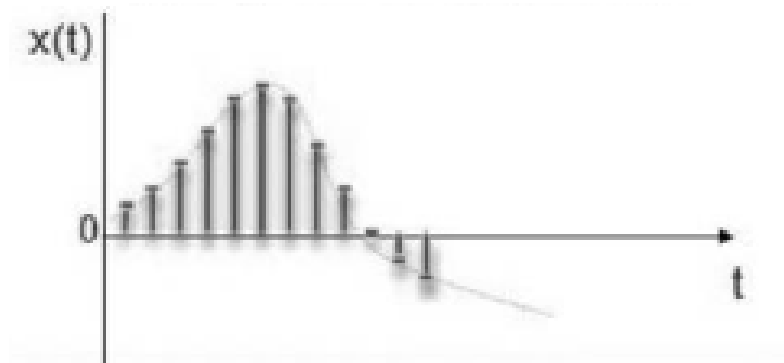


Рисунок 1.5. Дискретно-безперервне моделювання

Думкова побудова поділяється на:

- наочне;
- символічний;
- математичне.

Наочне моделювання ілюструє процеси, які усередині процесів, предметів, станів. Прикладами наочної побудови є діаграми, схематичні картини, макети, діаграми і т.д.

Символічна побудова (modeling) створює процеси предмета які у штучному вигляді, дії та властивості якого визначаються знаками і символами, і схожі на реальний процес. До прикладів можна віднести мови програмування, рівняння тощо.

Для відображення будь-якого явища, предмета як математичних співвідношень чи алгоритму, застосовується математичне побудова (modeling). Головною метою цієї побудови (modeling) є використання математичних методології для дослідження подій, що виникають реально, як і в оригіналі.

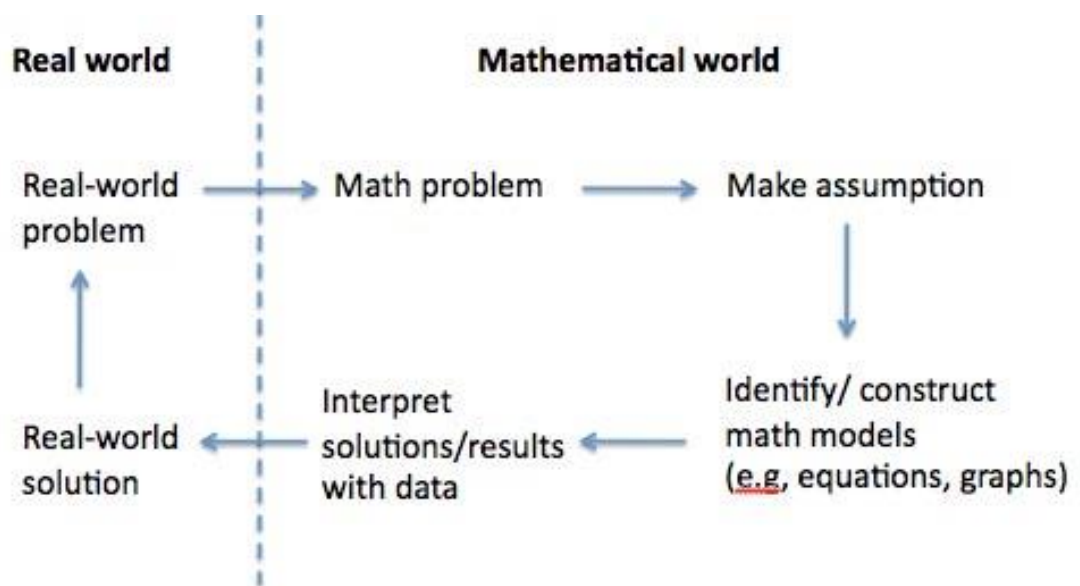


Рисунок 1.6. Створення математичної моделі.

Модель представляє процеси, наближені до точніших процесів оригінального (істинного) предмета. Ця модель може складатися з

математичних рівнянь, що включають змінні, події якого схожі, з діями реального процесу.

Математичне моделювання процесів поділяється на:

- аналітичне;
- імітаційне;
- комбіноване.

Аналітичне моделювання.

Перебіг процесів у системах, в аналітичному моделюванні фіксуються на кшталт функціональних залежностей та логічних виразів.

$$MM = \begin{cases} L_1(x_1, x_2, \dots, x_n, y_1) \leq b_1 \\ L_2(x_1, x_2, \dots, x_n, y_2) \geq b_2 \\ L_3(x_1, x_2, \dots, x_n, y_3) > b_3 \\ \dots \\ G(x_1, x_2, \dots, y_1, y_2, \dots) \rightarrow opt \end{cases}$$

Рисунок 1.7. Аналітична математична модель.

Де, x_1, x_2, \dots, x_n і y_1, y_2, \dots, y_m - Параметри модельованого об'єкта;

b_1, b_2, \dots, b_k – обмеження, що накладаються функціонування об'єкта довкіллям;

$G(x_1, x_2, \dots, y_1, y_2, \dots)$ – цільова функція моделювання.

Аналіз в аналітичній моделі протікає з такими методами, як:

- Аналітичний, який використовується для знаходження залежностей у загальному вигляді, між введеними властивостями та характеристиками.

- Чисельний метод, застосовується, а то й виходить вирішити рівняння у вигляді, зі знаходженням числових значень і введенням вихідних значень.
- Якісний метод, застосовується, якщо не можна вирішити рівняння у певному вигляді, щоб виявити особливості обчислень.

Імітаційне моделювання.

Імітаційне моделювання є способом побудови прототипу, досліджуваного процесу, який виконує процеси з точністю як реальні. Завдяки цій моделі можна стежити за діями системи у часовому факторі. Плюсом цієї моделі є його застосування до більш складних завдань, на відміну від аналітичної моделі.

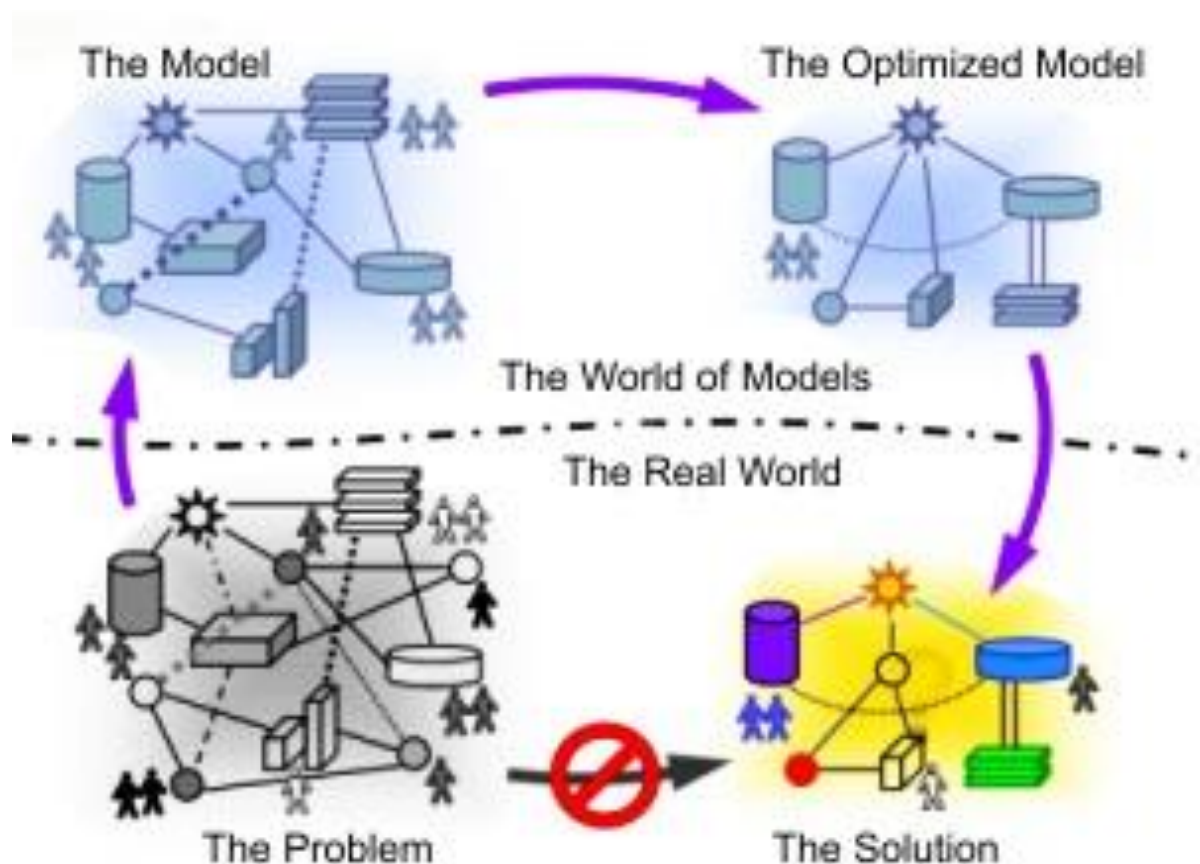


Рисунок 1.8. Приклад імітаційного моделювання.

У час імітаційне побудова можна назвати однією з ефективних методів експериментального спостереження реальних об'єктів і багато використовується при пізнання складних систем.

Комбіноване моделювання.

Комбінована побудова (modeling) включає дві вище перераховані моделі. При проектуванні цієї моделі проводиться розщеплення процесів на підпроцеси, у місцях розщеплення застосовується аналітичне, а де немає розщеплення імітаційне.

Така модель допомагає досліджувати системи, які неможливо досліджувати, лише імітаційним чи аналітичним моделюванням.

Реальне моделювання є таким методом побудови, який застосовується для спостереження процесів на самому предметі (original). Через спостереження на реальному предметі можливості цього моделювання є обмеженими.

Реальне моделювання поділяється на:

- натурне;
- фізичне.

Натурне моделювання здійснює обробку значень, знайдених після вивчення реального предмета (явлення), на основі теорій подібності. Фізична побудова (modeling) відтворює зовнішнє середовище та вивчає дії предмета (оригіналу) або його прототипу, незалежно від тимчасового фактора.

Кібернетичне моделювання.

Кібернетична побудова (modeling) грає значної ролі у моделюванні, й у ньому немає схожості процесів до оригіналу, які у моделі.

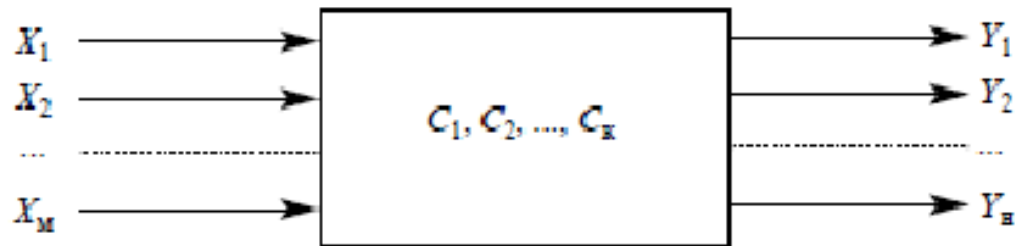


Рисунок 1.9. Кібернетичне моделювання.

Вона створюється на основі формальних відображень процесів, із зв'язком із зовнішнім середовищем. Зв'язок із зовнішнім середовищем позначається поняттями "входу" та "виходу".

X_1, X_2, \dots, X_M - входи;

Y_1, Y_2, \dots, Y_N - виходи;

C_1, C_2, \dots, C_K - Стан.

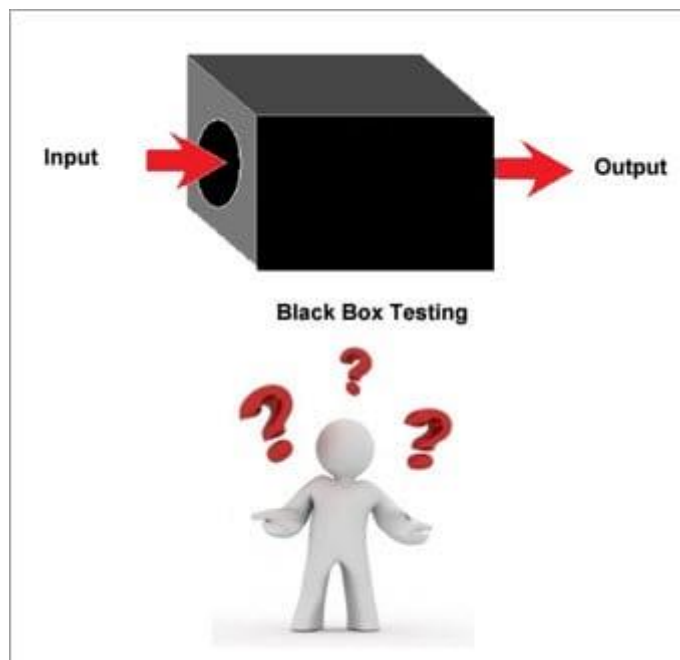


Рисунок 1.10. Метод "чорного ящика".

На основі кібернетичного моделювання лежить метод "чорного ящика", який розглядає поведінки реального об'єкта (вихідної інформації), із впливами зовнішнього середовища (вхідної інформації), незалежно від процесів, що протікають усередині самого об'єкта.

Когнітивне моделювання .

В даний час одна з галузей когнітивних наук, що розвиваються, є когнітивне моделювання, яке є міждисциплінарним науковим напрямом, що поєднує теорію пізнання, когнітивної психології, нейрофізіології, теорії когнітивної лінгвістики та штучного інтелекту. Когнітивне моделювання, допомагає вирішити низку завдань прогнозування. Традиційно, когнітивна модель є когнітивними картами у вигляді орієнтованого графа (орграфа) $G = \langle V, E \rangle$, де:

V являє собою набір понять (концепцій),

$V_i \in V, i = 1, 2, \dots, k$ – елементи системи дослідження;

E є безліч дуг, а дуги $e_{ij} \in E, i, j = 1, 2, \dots, N$ відображають співвідношення між поняттями.

Когнітивна карта дозволяє аналізувати як якісні, так і кількісні дані, а загальна мета когнітивного моделювання полягає у створенні та тестуванні гіпотези про функціональну структуру спостережуваної ситуації. Ітерації проводяться до певного стану причинно-наслідкових відносин, структурна схема якого дозволить нам зрозуміти та аналізувати поведінку системи.

Розглянувши основу методології моделювання, можна сказати що всі перераховані вище методи моделювання можна застосовувати до інформаційних процесів. Але сьогодні існує багато методів і гібридів для моделювання, які будуть розглянуті в наступних розділах.

1.4. Нейро-нечітка класифікація об'єктів та їх станів.

Нейронна мережа являє собою складними відносинами між входами і виходами. На основі поєднання результатів теорії нейронних мереж та теорії нечітких множин можливе створення оригінальних інтелектуальних систем. Нечіткі системи, як правило, не вчаться і не пристосовуються, у той час як нейронні мережі мають здатність до самонавчання та самоорганізації. З цих причин цілком логічно створення нейро-нечітких систем, які є комбінацію систем нечіткого виведення та нейронної мережі. Нейро-нечітка система являє собою нечітку нейронну мережу, яка включає

себе систему нечітких висновків, долає деякі обмеження нейронних мереж і розширює межі нечітких систем.

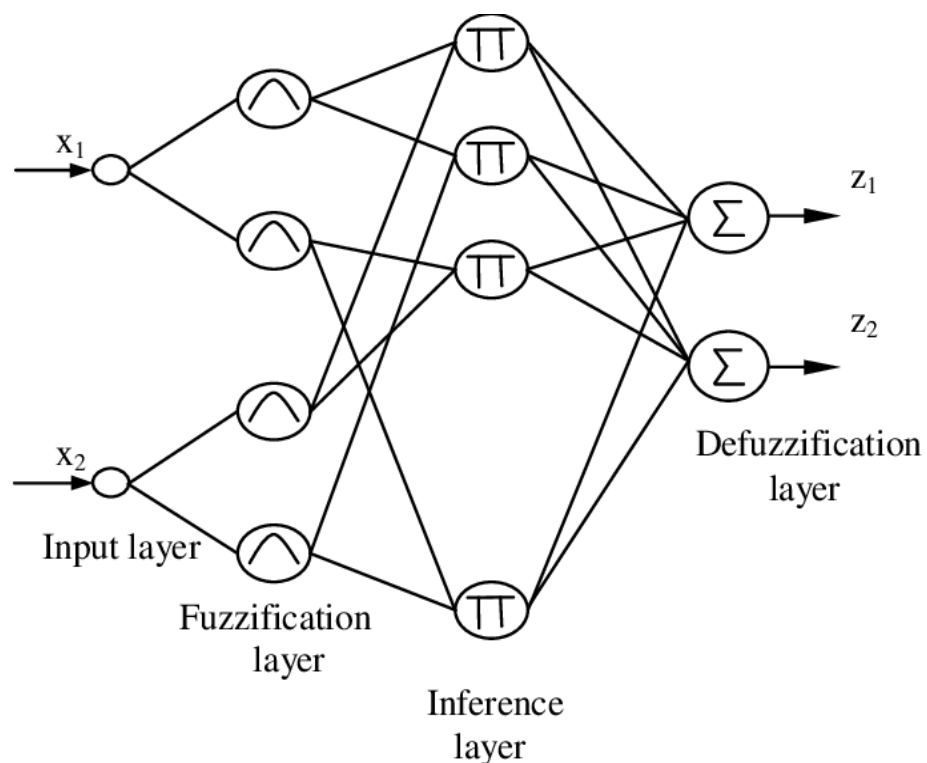


Рисунок 1.11. Структура нейро-нечіткого класифікатора

У відомих нечітких нейронних системах - нейро-нечітких класифікаторах (ННК) - кожен вузол мережі описується параметризованою дзвоноподібною

функцією активації (ФА) або гаусової ФА. На даний момент пропонуються нові моделі нечітких НС (ННС), що використовують імпульсні нечіткі ФА нейронів ННС, навчання ННК алгоритму масштабованого сполученого градієнта (SCG) та дослідження можливості ННК для вирішення завдань класифікації об'єктів та його станів. Для тестування ННК зазвичай використовуються класичні набори даних: іриса Фішера та шкали балансу.

Іриса Фішера – це набір даних для завдання класифікації, з прикладу якого Рональд Фішер в 1936 продемонстрував ефективність розробленого ним методу дискримінантного аналізу. Цей набір даних став вже класичним та часто використовується для ілюстрації ефективності роботи різних статистичних алгоритмів Шкали балансу також є класичним набором даних завдання класифікації. Цей набір даних запропонований Зіглером та описує процес зважування з використанням шкали балансу.

Завдання нечіткої класифікації – це завдання розбиття простору класифікаційних ознак на нечіткі класи. Можна описати простір класифікаційних ознак з нечіткими областями та керувати кожною областю за допомогою нечітких правил.

Для досягнення цієї мети нечіткий параметр функції і нечіткий кластер повинні бути оптимізовані. Коли правила нечіткої класифікації будуються як мережу, їх параметри можуть бути адаптовані до нейронної мережі. В результаті системи з нечіткою класифікацією та нейронні мережі можуть бути поєднані зі збереженням їх позитивних властивостей. Комбінована система називається нейро-нечітким класифікатором, який є адаптивною мережею з декількома входами та кількома виходами.

Нейро-нечіткий класифікатор є багатошаровою односпрямованою мережею, що складається з наступних шарів: вхідний шар, шар нечітких ФА, шар фазифікації, шар дефазифікації, шар нормалізації та вихідний шар. Класифікатор має кілька входів та кілька виходів.

РОЗДІЛ 2. НЕЙРО-НЕЧІТКІ МЕТОДИ ФОРМАЛІЗАЦІЇ ІНФОРМАЦІЙНИХ ПРОЦЕСІВ

2.1. Нечітка логіка

Системи нечіткої логіки, розроблені Лотфі Заде в 1965 р. Його робота була мотивована тому, що існує широке розмаїття неточної інформації, що створює людина. Де деякі проблеми не можна вирішити тільки за допомогою класичної булевої логіки. У деяких ситуаціях лише двох значень недостатньо щоб розв'язати задачу. Нечіткі методи вирішують невизначеність вирішення складних проблем, пов'язаних з можливістю представлення складних проблем з більш значними мінливістю уявлень.

Нечітка логіка дозволяє використовувати математику для представлення в формування в області невизначеності для вирішення різноманітних задач. За допомогою нечіткої логіки можна створювати системи, які можуть виконувати наближення функції, переважно за допомогою універсального теорема наближень, що адитивна система $F: X \rightarrow Y$ апроксимує функцію $f: X \rightarrow Y$, якщо X — компактна область і f неперервна. Це тому, що в нечіткій системі кожне правило представляє локальну модель, яка агрегується з іншими правилами, щоб надати остаточний вихід моделі.

Мовні структури і чіткі (числові) дані можна використовувати як інформацію для нечітких системи. Якщо використовуються чіткі дані, то інформація передається в фазифікацію, що дозволяє створити відповідний нечіткий набір на нечіткий вхід. Вміст введених змінних відображаються в лінгвістичні значення вихідних змінних, які треба вміти керувати відповідним методом наближених міркувань (машина висновку) з використанням експертних знань, які інтерпретуються як сукупність нечітких умовних правил (база знань). В

розширеннях лінгвістичних значень, числові дані можуть бути необхідними як вихід нечіткої системи. У вище згаданих обставинах, використовуються методи дефазифікації, що дозволяють чіткі репрезентативні дані для результуючого вихідного нечіткого набору.

Корисний рішенням є використання нечітких систем для задач в яких повне математичне представлення недоступне, або де використання точної (нечіткої) моделі є економічним або дуже зручним. Використання нечітких систем є необхідним у випадках, коли класичний підхід стає нездійсненним для вирішення проблеми через характер її складності. Найвідоміші методи здатні вносити різкі зміни розв'язувати задачі за рахунок спрощення моделі, однак нечіткі системи мають ресурси (функції належності, правила та оператори агрегації), які дозволяють точнішу апроксимацію до фактичної моделі, уникаючи рішення, створеного за допомогою нечіткої системи відрізняється від реальної моделі. Іншим актуальним поняттям у нечітких системах є нечітка множина. Це тип елемента може бути представлений відношенням, яке оцінює діапазон цільової змінної задачі, наприклад висота рослини, яка може варіюватися від нуля до 15 м.

Область обчислювального інтелекту демонструє значний прогрес у розробці методів і моделей, які моделюють процеси і системи, з акцентом на штучні нейронні мережі, нечіткі системи та їх гібридні моделі, при цьому пропонується значна кількість нових застосувань. в літературі. Однією з головних цілей досліджень обчислювального інтелекту є створення та моделювання обчислювальних систем, які імітують специфічні людські характеристики, такі як навчання, інтуїція, логічне міркування, класифікація та регресія.

Нечіткі нейронні мережі — це нейронні мережі з нечітких нейронів, ці мережі мають як головну характеристику синергічної співпраці між нечіткою теорією та нейронними мережами, генеруючи моделі, які інтегрують лікування невизначеності та інтерпретації, що надаються нечіткими системами, та здатність до навчання, що забезпечується нейронними мережами

З іншого боку, нечіткі нейронні мережі можна визначити як нечітку систему, яка навчається за алгоритмом, наданим інтелектуальною моделлю. Враховуючи цю аналогію, об'єднання нейронної мережі з нечіткою логікою має на меті пом'якшити недолік кожної з цих систем, створюючи більш ефективну, надійну та легко зрозумілу систему. Нечіткі нейронні мережі можна класифікувати за тим, як їх нейрони пов'язані. Ця форма з'єднання визначає, як сигнали будуть передаватися в мережі. Загалом, існує пряма передача, коли нечіткі нейрони групуються в шари, і сигнал проходить всю мережу в одному напрямку, зазвичай від входу моделі до її виходу, генеруючи очікуваний результат. Нечіткі нейрони в одному шарі, які не мають зв'язку, а їхні мережі також відомі як мережі без них зворотний зв'язок.

Архітектура гібридних моделей нейронних мереж і нечітких систем має кілька характеристик. Загалом, кожен рівень відповідає за завдання, яке має бути виконано, що дозволяє моделям мати більш динамічний спосіб вирішення проблем. Загалом, архітектурні моделі гібридних мереж дотримуються подібної організації моделей нейронних мереж, щоб їх можна було організувати з прямим зв'язком, де є шаруваті нейрони. У ньому сигнал проходить по мережі в одному напрямку від входу до виходу.

Нейрони в одному шарі не з'єднані. Подібно до того, як існують рекурентні мережі, які мають вихід деяких нейронів, які живлять нейрони того ж шару (включаючи себе) або попередніх шарів, сигнал проходить мережу в двох напрямках, має динамічну пам'ять і здатність представляти стани в динамічних системах

Тип і кількість шарів в архітектурі гібридних моделей може змінюватися залежно від складності задачі. Вони мають кілька функцій для виконання завдань. Більш традиційні моделі мають один або два шари, приховані в їх структурі. Вони можуть бути відповідальними за процес нечіткого висновку, формування правил, нечіткості або дефазифікації даних. Кожен шар має точні функції щодо його

продуктивності в моделі, але загалом перший відповідає за процес фазифікації, а останній — за остаточні відповіді моделі. Деякі з цих шарів можуть представляти нечіткі системи висновку або нейронну мережу агрегації.

В останнє десятиліття було запропоновано моделі з п'ятьма шарами для вирішення проблем регресії, провісників часових рядів і для класифікації моделей у відповідних задачах. Були розроблені моделі, запропоновані в моделях, які застосовується до динамічної обробки системи. Інші приклади, такі як ідентифікація динамічних систем, розроблена, мають ефективність вирішення проблем у галузях. Модель, яка використовує нечіткі правила корельований контурів для апроксимації функцій, та адаптивну нейро-нечітку модель, створену Інші моделі з п'ятьма шарами або більше

За останні кілька років спостерігалось швидке зростання числа та різноманітності додатків нечіткої логіки (FL). Методи FL використовувалися у додатках для розуміння зображень, таких як виявлення країв, витяг ознак, класифікація та кластеризація. Нечітка логіка створює здатність імітувати людський розум, ефективно використовувати методи міркування, які є приблизними, а не точними. У традиційних жорстких обчисленнях, рішення чи дії засновані на точності та упевненості.

Нечітка логіка починається з поняття нечіткої множини. Нечітка множина - це набір без чіткої, чітко визначеної межі. Він може містити елементи із частковим ступенем приналежності. Класична множина - це контейнер, який повністю включає або повністю виключає будь-який елемент. Інша версія цього закону говорить: «З будь-якого предмета потрібно або затверджувати, або заперечувати». Тобто перехід від «приналежності до множини» до «не множини» є поступовим, і цей плавний перехід характеризується функцією приналежності, яка дає гнучкість безлічі нечітких множин при моделюванні лінгвістичних виразів, що зазвичай використовуються .

Функція приладдя (MF) - це крива, яка визначає, як кожна точка у просторі введення відображається на значенні приналежності (або ступеня приналежності) між 0 та 1. Вхідний простір іноді називають універсумом дискурсу (простір міркувань). Вихідна вісь є числом, відомим як значення приналежності між 0 і 1. Крива відома як функція приналежності і часто дається позначення μ . Найпростіші MF формуються з використанням прямих ліній. З них найпростішим є функція трикутної приналежності, і вона має ім'я функції `trimf`. Ця функція є ні що інше, як сукупність трьох точок, що утворюють трикутник. Функція трапецієподібної приналежності, `trapmf`, має плоску вершину і насправді є лише усіченою трикутною кривою. Ці функції лінійної власності мають перевагу простоти.

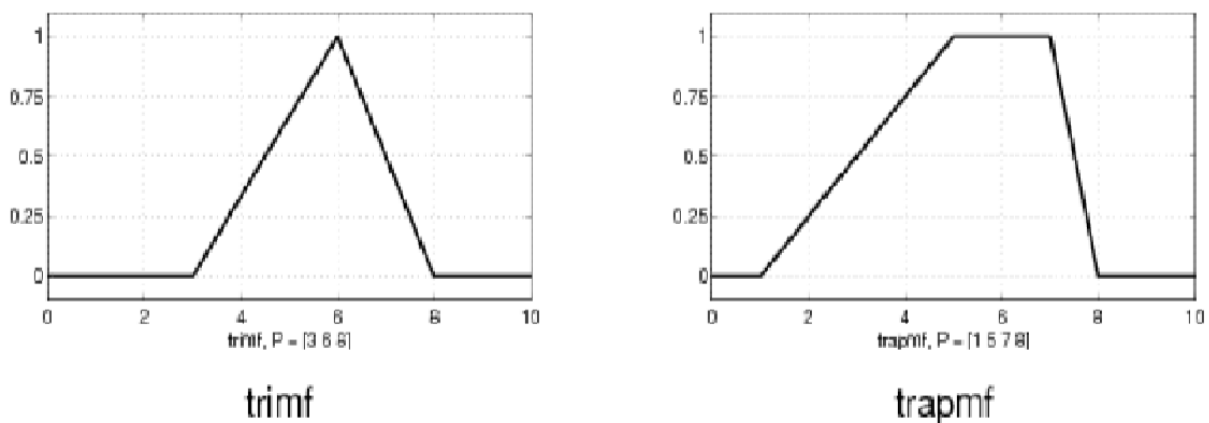


Рисунок 2.1. Трикутної та Трапецієподібної MF.

На кривій розподілу Гауса побудовано дві функції приналежності: проста гаусова крива та двосторонній композит двох різних гаусових кривих. Дві функції є `gaussmf` та `gauss2mf`. MF “узагальненого бані” визначається трьома параметрами та має ім'я функції `gbellmf`.

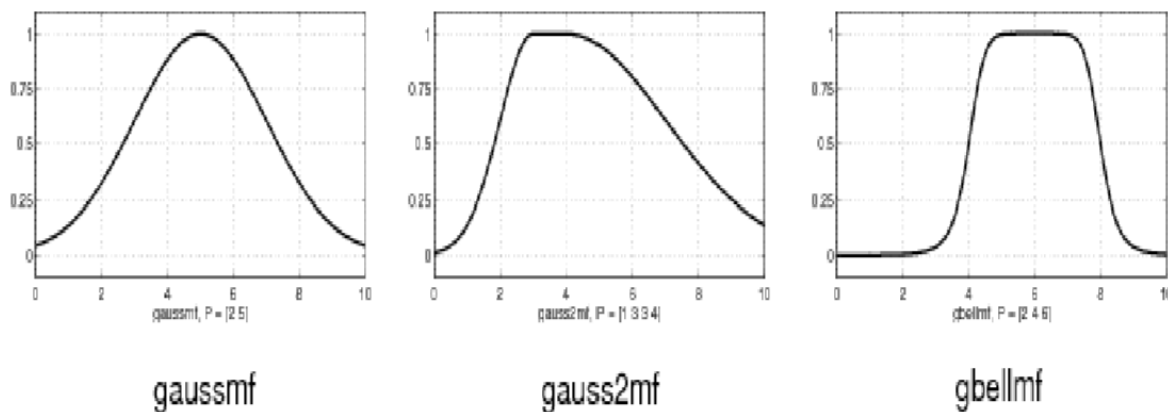


Рисунок 2.2. MF Гауса та “узагальненого купола”.

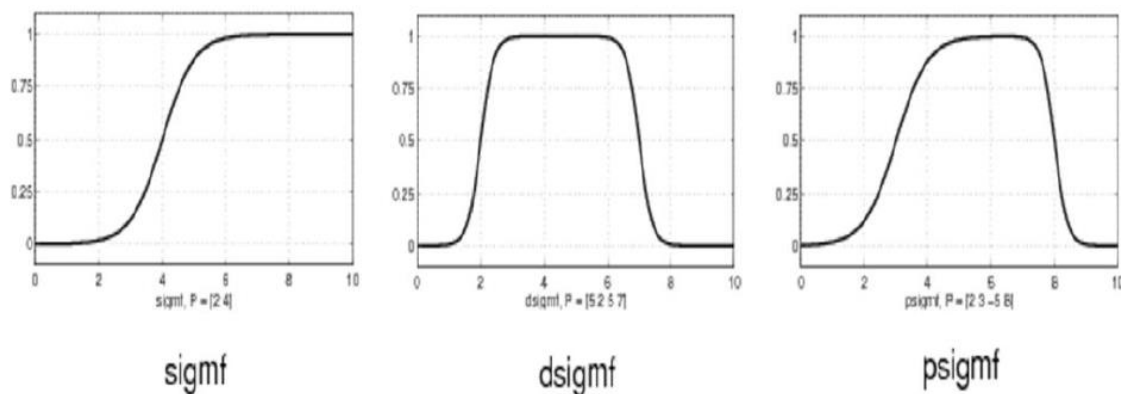


Рисунок 2.3. Сигмоїдні MF.

Також ми можемо визначити сигмоїдальну MF, яка або відкрита вліво або вправо. Асиметричні та закриті (тобто не відкриті ліворуч або праворуч) MF можуть бути синтезовані з використанням двох сигмоїдальних функцій, тому на додаток до основного `sigmf`, ми також маємо різницю між двома сигмоїдальними функціями – `dsigmf` та добуток двох сигмоїдальних функцій `psigmf`.

Нечітке правило *If-Then*.

Нечітке правило *If-Then* набуває вигляду:

$$\text{If } x \text{ is } A, \text{ then } y \text{ is } B, \quad (2.1)$$

де A і B - лінгвістичні значення, що визначаються fuzzy множинами в просторі міркувань X і Y відповідно. Часто « $x \in A$ » називають антецедентом або передумовою, тоді як « $y \in B$ » називають наслідком або висновком. Приклади нечітких правил *if-then* у наших щоденних лінгвістичних виразах такі: якщо дорога слизька, водіння небезпечне. Перш ніж ми зможемо використовувати правила *if-then* для моделювання та аналізу системи, ми спочатку повинні формалізувати те, що розуміється під виразом *if x is A , then y is B* , що іноді скорочується як $A \rightarrow B$. по суті, вираз описує зв'язок між двома змінними x та y ; це передбачає, що нечітке правило *if-then* визначається як двійкове нечітке відношення R на просторі творів $X \times Y$.

Бінарне відношення R є розширенням класичного декартового твору, причому кожен елемент $(x, y) \in X \times Y$ пов'язаний з класом приналежності, що позначається $\mu_R(x, y)$. Інтерпретація правила *if-then* включає різні частини: спочатку оцінює антецедент (який включає fuzzifying вхід і застосування будь-яких нечітких операторів), а другий застосовує цей результат до наступного (відомого як імплікація). У разі двозначної чи двійкової логіки правила *if-then* не становлять великої складності.

У більшості додатків рішення FL є перекладом людського рішення. Потретьє, FL може моделювати нелінійні функції довільної складності з необхідним ступенем точності. FL - зручний спосіб відобразити вхідний простір у вихідний простір. FL є одним з інструментів, що використовуються для моделювання багаторівневої системи.

Залежно від структури правил *if-then* можна виділити два основні типи нечітких моделей: модель Мамдані (або лінгвістична) та модель Такагі-Сугено.

Модель Мамдані.

У цій моделі антецедент (if – частина правила) та наступна (then – частина правила) є нечіткими пропозиціями:

$$\mathcal{R}_i: \text{If } \mathbf{x} \text{ is } A_i \text{ then } y_i = \mathbf{a}_i^T \mathbf{x} + b_i, \quad i = 1, 2, \dots, K, \quad (2.2)$$

Тут A_i і B_i – антецедентні та наступні лінгвістичні терміни (такі як «малі», «великі» тощо), представлені нечіткими множинами, а K – кількість правил у моделі. Лінгвістична нечітка модель корисна уявлення якісних знань.

Модель Такагі-Сугено (TS).

Модель Мамдані зазвичай використовують у системах знань (експертних). В ідентифікації на основі даних, модель Такагі і Сугено стала популярною. У цій моделі антецедент визначається так само, як і вище, тоді як наступна є лінійною функцією вхідних змінних:

$$\mathcal{R}_i: \text{If } \mathbf{x} \text{ is } A_i \text{ then } y_i = \mathbf{a}_i^T \mathbf{x} + b_i, \quad i = 1, 2, \dots, K, \quad (2.3)$$

де a_i – послідовний параметр вектора, а b_i – скалярне зміщення. Ця модель поєднує лінгвістичне опис зі стандартною функціональною регресією: антецеденти описують нечіткі області у вхідному просторі, у якому діють відповідні функції. Результат у обчислюється шляхом обчислення зваженого середнього внеску індивідуальних правил:

$$y = \frac{\sum_{i=1}^K \beta_i(\mathbf{x}) y_i}{\sum_{i=1}^K \beta_i(\mathbf{x})} = \frac{\sum_{i=1}^K \beta_i(\mathbf{x}) (\mathbf{a}_i^T \mathbf{x} + b_i)}{\sum_{i=1}^K \beta_i(\mathbf{x})} \quad (2.4)$$

де $\beta_i(x)$ – ступінь виконання i -го правила. Для правила зазначеного зверху, $\beta_i(x) = \mu_{A_i}(x)$, але воно також може бути складнішим виразом, як показано нижче. Антецедентні нечіткі множини зазвичай визначаються для опису окремих частково перекриваються областей у вхідному просторі. Параметри a_i (наближені) локальні лінійні моделі аналізованої нелінійної системи.

Таким чином, модель TS можна розглядати як гладку шматково-лінійну апроксимацію нелінійної функції або моделі планування параметрів. Звернемо увагу, що антецедент та наступні змінні можуть бути різними.

Оператори нечіткої логіки (AND, OR, NOT).

У нечітких системах з кількома входами антецедентна пропозиція зазвичай подається як комбінація термінів з одновимірними функціями приналежності, використовуючи логічні оператори І (кон'юнкція), АБО (диз'юнкція) і НЕ (доповнення).

Таблиця 2.1. Зазвичай використовувані функції операторів нечіткої логіки.

A and B	A or B	not A
$\min(\mu_A, \mu_B)$	$\max(\mu_A, \mu_B)$	$1 - \mu_A$
$\mu_A \cdot \mu_B$	$\mu_A + \mu_B - \mu_A \cdot \mu_B$	$1 - \mu_A$

Теоретично нечітких множин для цих логічних зв'язок введено кілька сімейств операторів. У таблиці 2.1 показані найпоширеніші.

2.2 Механізм нечіткого висновку

Система нечіткого виведення (FIS) сутнісно визначає нелінійне відображення вектора вхідних даних у скалярний висновок з використанням нечітких правил. Процес включає порівняння собою функцію введення - виведення, оператори FL нечіткі правила якщо щось, встановлює вихід агрегації

і дефазифікації. FIS з кількома виходами можна як набір незалежних багатоканальних систем з одним виходом.



Рисунок 2.4. Загальна модель системи нечіткого виводу (FIS).

FLS відображає чіткі входи у чіткі виходи. З малюнка видно, що FIS містить чотири компоненти: fuzzifier, двигун виведення, основу правил і defuzzifier. База правил містить лінгвістичні правила, що надаються експертами. Також можна витягувати правила із числових даних. Як тільки правила будуть встановлені, FIS можна розглядати як систему, яка відображає вхідний вектор у вихідний вектор. Fuzzifier відображає вхідні числа у відповідні нечіткі членства. Це необхідно для того, щоб активувати правила, що стосуються лінгвістичних змінних. Fuzzifier набуває вхідних значень і визначає ступінь, в якому вони належать кожному з нечітких множин через функції приналежності.

Механізм виведення визначає відображення вхідних нечітких множин у вихідні нечіткі множини. Він визначає ступінь, у якій антецедент виконується кожному за правила. Якщо антецедент цього правила має більше однієї пропозиції, застосовуються нечіткі оператори, щоб отримати одне число, яке є результатом антецедента для цього правила. Можливо, одне або кілька правил можуть спрацювати одночасно. Потім виходи всім правил агрегуються.

Під час агрегації нечіткі множини, що дають висновок кожного правила, об'єднуються в один нечіткий набір. Нечіткі правила запускаються паралельно, що є одним із важливих аспектів FIS. У FIS порядок, у якому запускаються правила, впливає вихід. Defuzzifier відображає вихідні нечіткі множини у чітке число.

Враховуючи нечітку множину, яка охоплює діапазон вихідних значень, defuzzifier повертає одне число, тим самим переходячи від нечіткої множини до чіткого числа.

На практиці використовуються кілька методів дефазифікації, у тому числі центроїд, максимум, середнє значення максимумів, висоти та зміненого деформуючого пристрою.

Найпопулярніший метод дефазифікації - це центроїд, який обчислює та повертає центр тяжіння агрегованої нечіткої множини. FIS використовують правила. Однак, на відміну від правил у звичайних експертних системах, нечіте правило локалізує область простору вздовж функціональної поверхні, а не ізолює точку на поверхні.

Для цього введення може спрацьовувати більше одного правила. Крім того, FIS безліч областей об'єднуються у вихідному просторі для створення складової області.

Процес нечіткого висновку:

Крок 1. Нечіткі входи.

Перший крок - прийняти вхідні дані та визначити ступінь, в якому вони належать кожному з відповідних нечітких множин через функції приналежності.

Крок 2. Застосування нечітких операторів.

Як тільки входи були фазифіковані, ми знаємо ступінь, у якому кожна частина антецедента була задоволена для кожного правила. Якщо задане

правило має більше частини, нечіткі логічні оператори застосовуються з метою оцінки складової сили запуску правила.

Крок 3: Застосування методу імплікації.

Метод імплікації окреслюється формування вихідних функцій членства з урахуванням сили порушення правила. Вхід для процесу імплікації – це одне число, задане антецедентом, а вихід – нечітка множина. Двома зазвичай використовуваними методами імплікації є мінімум та продукт.

Крок 4: Агрегування всіх виходів.

Агрегація - це процес, у якому результати кожного правила уніфіковані. Агрегація відбувається лише один раз для кожної вихідної змінної. Введення в процес агрегації - це усічені вихідні нечіткі множини, що повертаються процесом імплікації для кожного правила. Результатом процесу агрегації є комбінований вихідний нечіткий набір.

Крок 5: Дефазифікація.

Вхід для процесу дефазифікації – це нечітка множина (агрегований набір нечітких вихідних даних), а вихід процесу дефазифікації – це чітке значення, отримане з використанням деякого методу дефазифікації, такого як центроїд, висота або максимум.

Система нечіткого виводу відображає вхідний вектор чітке вихідне значення. Щоб отримати чіткий вихід, нам потрібний процес дефазифікації. Входом у процес дефазифікації є нечітка множина (агрегований набір нечітких вихідних даних), а вихід процесу дефазифікації - це один номер. Пропонується багато методів дефазифікації. Найчастіше використовуваний метод - це центроїд. Інші методи включають максимум, засіб максимумів, висоти і метод зміненої висоти. П'ять методів можуть бути описані таким чином:

Метод дефазифікації Centroid: у цьому методі defuzzifier визначає центр тяжіння (центроїд) B і використовує це значення як результат FLS. Для безперервної агрегованої нечіткої множини центроїд задається формулою:

$$y' = \frac{\int y_i \mu_B(y) dy}{\int \mu_B(y) dy} \quad (2.5)$$

Де, S означає носія $\mu_B(y)$.

Часто використовуються дискретизовані змінні, так що y' можна апроксимувати, як показано нижче, в якому замість інтегрування використовуються суми.

$$y' = \frac{\sum_{i=1}^n y_i \mu_B(y_i)}{\sum_{i=1}^n \mu_B(y_i)} \quad (2.6)$$

Метод центроїдного дефазифікування знаходить «точку рівноваги» нечіткої області рішення, обчислюючи зважене середнє вихідне нечіткої області. Це найбільш широко використовуваний метод, тому що, коли він використовується, дефазифіковані значення, як правило, плавно переміщуються навколо вихідної нечіткої області. Однак ця технологія є унікальною і нелегко реалізувати обчислювально.

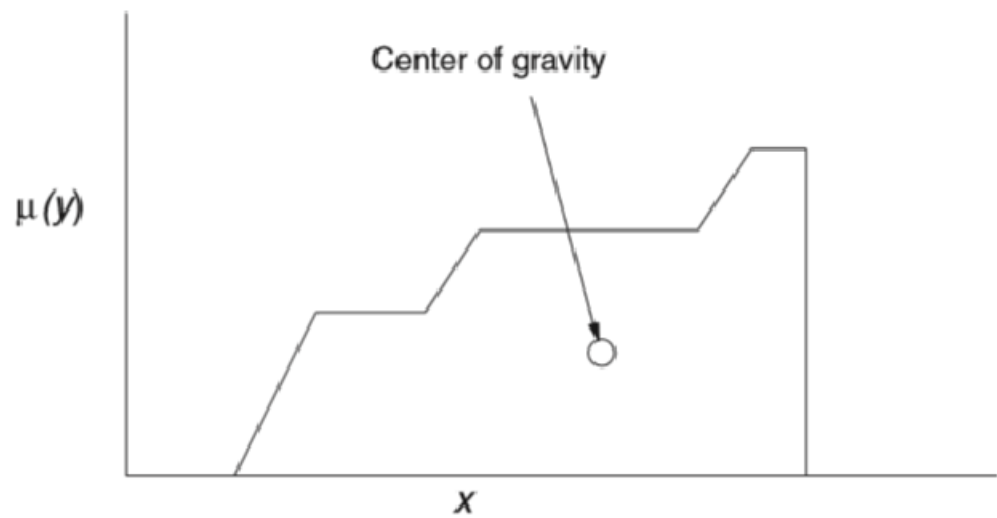


Рисунок 2.6. Метод центроїдної дефазифікації.

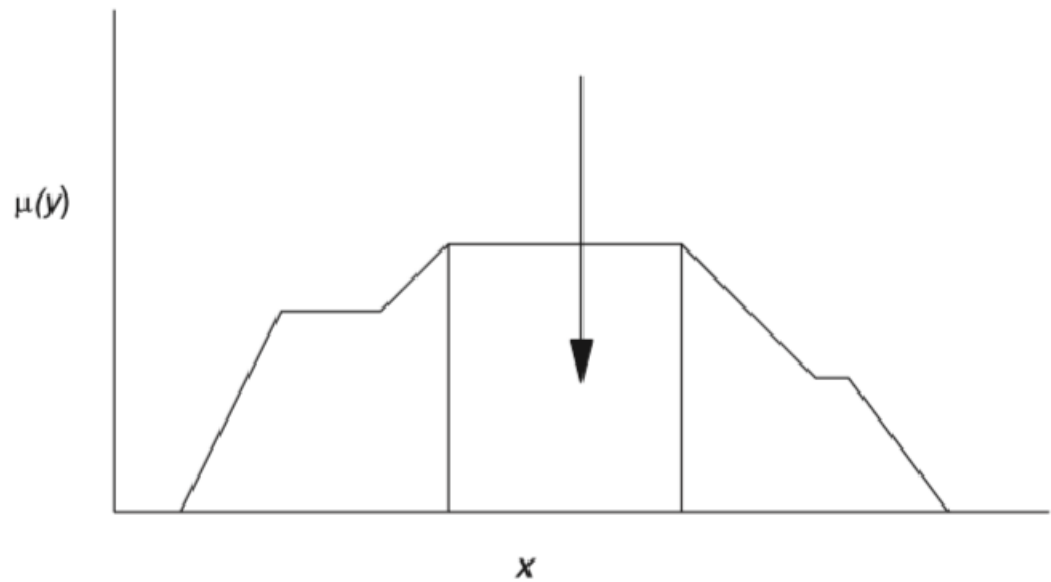


Рисунок 2.7. Метод максимальної декомпозиції.

Метод максимальної декомпозиції: у цьому методі defuzzifier аналізує агреговані нечіткі множини та вибирає такий вихід y , для якого $\mu_B(y)$ є максимальним. На відміну від методу центроїдів, метод максимального

розкладання має деякі властивості, які застосовуються до вузького класу проблем. Крім того, вихідне значення має тенденцію переходити від одного кадру до іншого у міру зміни форми нечіткої області.

Центр максимумів: у багаторежимній нечіткій області техніка центру максимумів знаходить найвище плато, а потім наступне найвище плато. І вибирається середня точка між центрами цих плато.

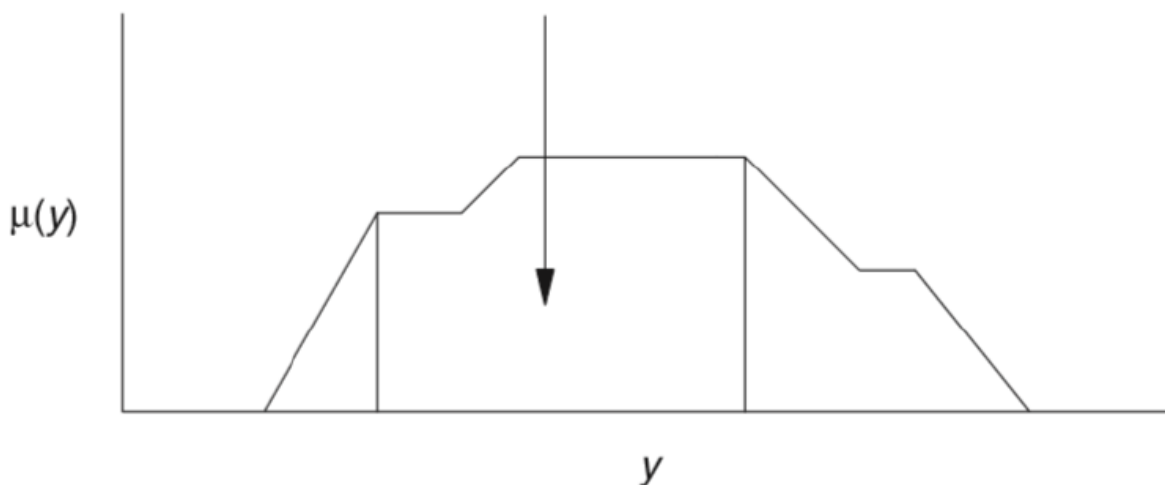


Рисунок 2.8. Метод центру максимумів.

Дефазифікація за висотою. У цьому методі дефазифікатор спочатку оцінює $\mu_{B(y)}$ на y'_1 , а потім обчислює вихід FLS, де визначається центр тяжкості нечітких множин B_i . Вихід y_h у цьому випадку задається виразом:

$$y_h = \frac{\sum_{i=1}^m y_i \mu_B(y_i)}{\sum_{i=1}^m \mu_B(y_i)} \quad (2.7)$$

де m являє кількість вихідних нечітких множин, отриманих після імплікації, а y'_1 є центр ваги нечіткої області i . Цей метод простий у використанні, оскільки центри тяжкості членів функцій, що широко використовуються, відомі заздалегідь.

Незалежно від того, використовуються чи мінімальні або виробничий вихід істотно визначає нечіткого логічного висновку процесу відображення хрустких значень вектора по значенню ясно з використанням нечітких правил, що зберігаються в базі знань.

Розглянутий процес нечіткого виведення відомий як метод нечіткого виводу Мамдані. Сугено (1977) запропонував метод нечіткого виведення, аналогічний методу Мамдані.

2.3 Штучний інтелект

Виконання інтелектуальної роботи дозволяє людині використовувати прогрес людства у повній мірі, одним з вінців цього є штучний розум, що полегшує прийняття рішень у різних сферах промисловості та торгівлі. Штучні нейронні мережі вже навчилися отримувати відповіді на завдання класифікації закономірностей, проблеми регресії, ідентифікації зображень та багато іншого.

Але все одно залишаються проблеми складні навіть для пояснення, щоб зробити отримані дані очікуваними у типових ситуаціях й виникло поняття нечітких систем, що дозволяють отримати інтерпретацію в тільки ще отриманих, не класифікованих даних.

Шукаючи узгоджену синергію між навчальною здатністю штучних нейронних мереж і можливостей представлення проблем у більше інтерпретаційний спосіб, й виникли моделі, що беруть найкраще з обох

концепцій для формування гібридних моделей під назвою нейро-нечіткі мережі.

Перша згадка про них виникла ще у 1960-х, їх охарактеризували як ефективне вирішення різноманітних проблем нашого суспільства. Мотивація розробки цих мереж полягає в тому, що вони легко інтерпретуються, завдяки чому можна отримати знання про топологію мережі. Ці мережі формуються шляхом співпраці теорії нечітких множин і нейронних мереж, що забезпечує широкий спектр здібностей до навчання. Вони надають моделі можливість інтегрувати й обробляти ще невизначену інформацію. Ці мережі дуже перспективні в різних областях застосування таких як нечітка кластеризація, моделювання нелінійних динамічних систем.

За останні кілька десятиліть нечіткі системи та їх гібриди деривації можуть імітувати людську здатність міркувати в обчислювальне-ефективний спосіб. Важлива область дослідження – це розробка таких систем на високому рівні гнучкості та самостійності для розвитку своїх структур і знань заснованих на змінах навколишнього середовища, вмючи впоратися в моделюванні, контролі, передбаченні та класифікації закономірностей в ситуаціях, схильності до постійних змін.

Ці моделі вбирали концепції еволюції навчання нейронних мереж і обробки даних для формування високої точністю моделей для побудови систем на основі правил, експертних систем, класифікаторів та універсальних апроксиматорів. Наша основна увага буде зосереджена на характеристиках різних архітектур і конструкцій моделей, які було визначено в контексті нечіткої нейронної мережі.

Інтелектуальні системи прагнуть моделювати поведінку людини в моделях для вирішення повсякденних проблем. У людини мозок обробляє інформацію паралельно, розподіляючи завдання на нервові клітини. Він

складається з багатьох нервових клітин, де нейрони є основними структурами, що реагують на подразники навколишнього середовища.

Вони відповідають за лікування спонукань, тобто обробку сигналів від ланцюга нейронів або на зовнішні ситуації. У своєму складі, нейрон представлений дендритами, аксонами, клітинним тілом та синапсами. Вони отримують вхідні сигнали від інших клітин через дендрити, обробляють, і генерувати вихідні сигнали, які передаються іншим нейронам через аксони та їх розгалуження

Штучні нейронні мережі в свою чергу є обчислювальними методами, які представляють математичну модель, натхненну нервовою структурою розумних організмів і тим, як отримуються знання через досвід, що дозволяє виконувати загальні завдання розумними істотами для виконання в своїх середовищах.

Існують категорії проблем, які не можна сформулювати як алгоритм. Проблеми, пов'язані із зменшенням кількості факторів, наприклад, покупна ціна нерухомості, яку наш мозок може (приблизно) розрахувати. Без алгоритму комп'ютер не може зробити те саме. Мозок містить 10^{11} нейронів, але вони мають час перемикання близько 10^{-3} секунд. Більшість мозку працює безупинно, тоді як більшість комп'ютера – це лише пасивне сховище даних. Мозок як біологічна нейронна мережа може бути реорганізований в протязом його «тривалості життя» і, отже, здатний вчитися, компенсувати помилки тощо.

Таким чином, вивчення штучних нейронних мереж мотивоване їхньою схожістю з успішною роботою біологічних систем, які, порівняно із загальною системою, складаються з дуже простих, але численних нервових клітин, які працюють у широких межах (що, ймовірно, є одним із найбільш значущих аспектів), мають можливість навчатися. Немає потреби явно програмувати нейронну мережу.

Наприклад, він може вчитися на навчальних зразках або через заохочення, так би мовити (посилення навчання). Одним із результатів цієї навчальної процедури є здатність нейронних мереж узагальнювати та асоціювати дані: після успішного навчання нейронна мережа може знайти розумні рішення для аналогічних проблем того ж класу, які не були явно навчені. Це, у свою чергу, призводить до високого ступеня відмовостійкості від шумних вхідних даних. Відмовостійкість яка тісно пов'язана з біологічними нейронними мережами, в яких ця характеристика дуже різна: як відомо, людина має близько 10^{11} нейронів, які постійно реорганізуються.

Поле нейронних мереж, як і будь-який інший напрямок науки, має довгу історію розвитку з багатьма злетами та падіннями. Історія нейронних мереж починається на початку 1940-х років і, отже, майже одночасно з історією програмованих електронних комп'ютерів, і вивчається і застосовується в сучасний час.

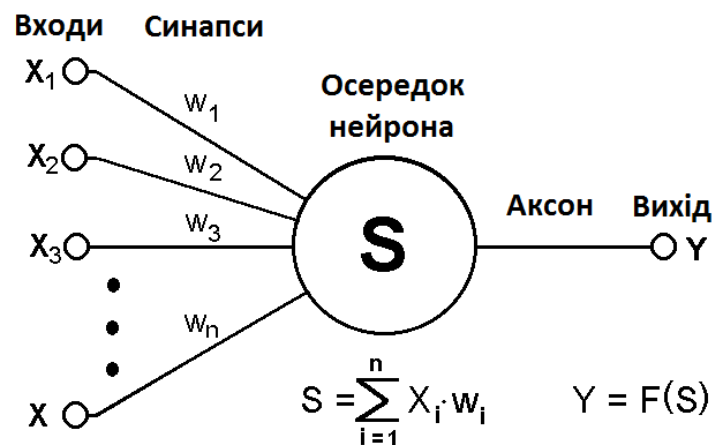


Рисунок 2.9. Біологічна нейронна мережа та штучна нейронна мережа.

Перехід від біологічних нейронів до штучних. У біологічних нейронах є прихильність один до одного, і коли вони стимулюються, вони електрично передають свій сигнал через аксон. З аксона вони не переносяться безпосередньо

в наступні нейрони, але спочатку їм доводиться перетинати синоптичну щілину, де сигнал знову змінюється за допомогою змінних хімічних процесів.

У приймальному нейроні різні входи, оброблені постобробкою в синоптичній щілині, підсумовуються або накопичуються до одного імпульсу. Короткий звіт точно відповідає декільком елементам біологічних нейронних мереж, які ми хочемо взяти в технічне наближення:

Введення технічних нейронів складається з багатьох компонентів, тому він є вектором. У природі нейрон отримує імпульси від 10^3 до 10^4 інших нейронів у середньому.

Скалярний вихід: вихід нейрона є скаляром, що означає, що нейрон складається лише з одного компонента. Декілька скалярних виходів у свою чергу утворюють векторний вхід іншого нейрона. Це, зокрема, означає, що десь у нейроні різні входні компоненти можуть бути об'єднані таким чином, що залишається лише один компонент.

Введення змін у синапсах: у технічних нейронних мережах введення також попередньо обробляється. Вони множачись на число (вага) – зважуються. Набір таких ваг є зберігання інформації нейронної мережі - як у біологічній оригінальній, так і в технічній адаптації.

Накопичувальні входи: в біології входів підсумовуються з імпульсом відповідно з хімічним зміною, тобто вони накопичуються, з технічного боку це часто реалізується виваженою сумою, про яку ми дізнаємось пізніше. Це означає, що після накопичення ми отримуємо лише одне значення – скаляр замість вектора.

Нелінійна характеристика: введення наших штучних нейронів також не пропорційне виходу.

Регульовані ваги: ваги, що зважують входи, є змінними, аналогічними хімічним процесам у синоптичній щілині. Це додає велику динаміку в мережу,

оскільки значна частина «знань» нейронної мережі зберігається у терезах, а також у формі та силі хімічних процесів у синаптичній щілині.

Таким чином, наша поточна, лише випадково сформульована і дуже проста нейронна модель отримує векторний вхід із компонентами.

Вони множаться на відповідні вагові коефіцієнти та накопичуються:

$$\sum_i \omega_i x_i \quad (2.8)$$

Вищезгаданий термін називається виваженою сумою. Тоді нелінійне відображення f визначає скалярний вихід y :

$$y = f \left(\sum_i w_i x_i \right) \quad (2.9)$$

Поняття часу в нейронних мережах У деяких визначеннях цього тексту ми використовуємо термін час або кількість циклів нейронної мережі відповідно. Час поділяється на окремі тимчасові кроки:

- поточний час (нині) визначається як (t) ,
- наступний крок часу як $(t + 1)$,
- попередній крок як $(t - 1)$.

Решта тимчасових кроків згадується аналогічно.

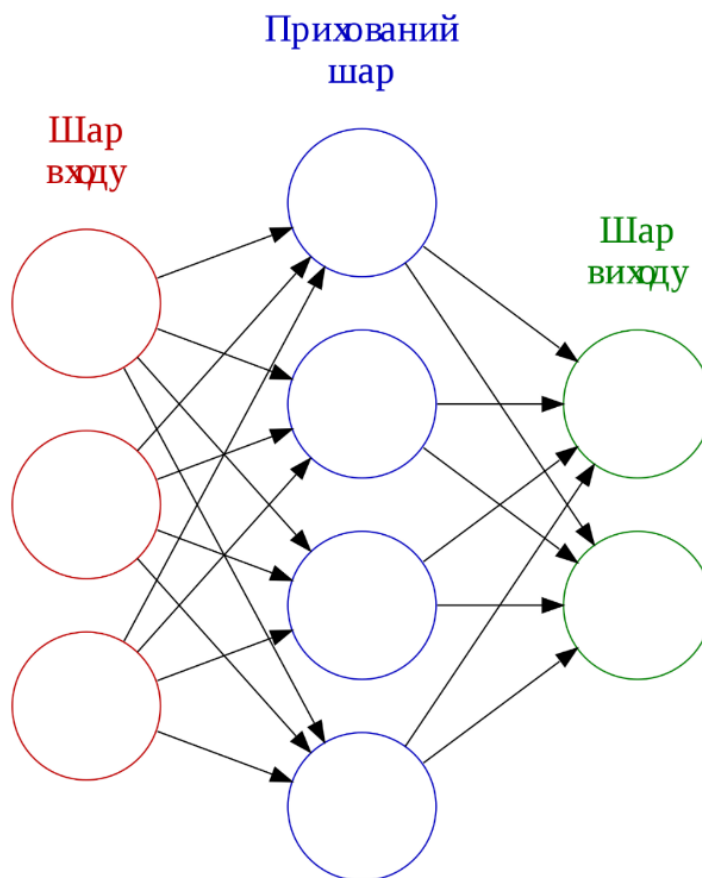


Рисунок 2.10. Шари нейронних мереж.

Штучна нейронна мережа складається з простих блоків обробки, нейронів та спрямованих зважених зв'язків між цими нейронами. Тут сила з'єднання (або сполучної ваги) між двома нейронами i та j визначається як w_{ij}

Нейронна мережа є відсортованою трійкою (N, V, w) з двома множинами N , V і функцією w , де N - безліч нейронів і V безліч $\{(i, j) \mid i, j \in N\}$, елементи називають зв'язками між нейроном i і нейроном j .

Функція $\omega: v \rightarrow \mathbb{R}$ визначає ваги, де w_{ij} , вага зв'язку між нейроном i та нейроном j скорочується до w_{ij} . Для з'єднань, які не існують у мережі, воно

приймає значення 0. З'єднання містять інформацію, оброблену нейронами. Дані переносяться між нейронами через ваги зв'язку.

Нейронні мережі складаються з шарів та з'єднань по відношенню до кожного наступного шару. Нейрони згруповані у такі шари:

- один вхідний шар;
- n прихованих шарів обробки (невидимий зовні, тому нейрони також називаються прихованими нейронами);
- один вихідний шар.

Топологія нейронних мереж.

Існують топології нейронних мереж:

- Feedforward (з прямим зв'язком);
- Feedback (зі зворотним зв'язком);
- Мережа "Hopfield" (з симетричним зв'язком).

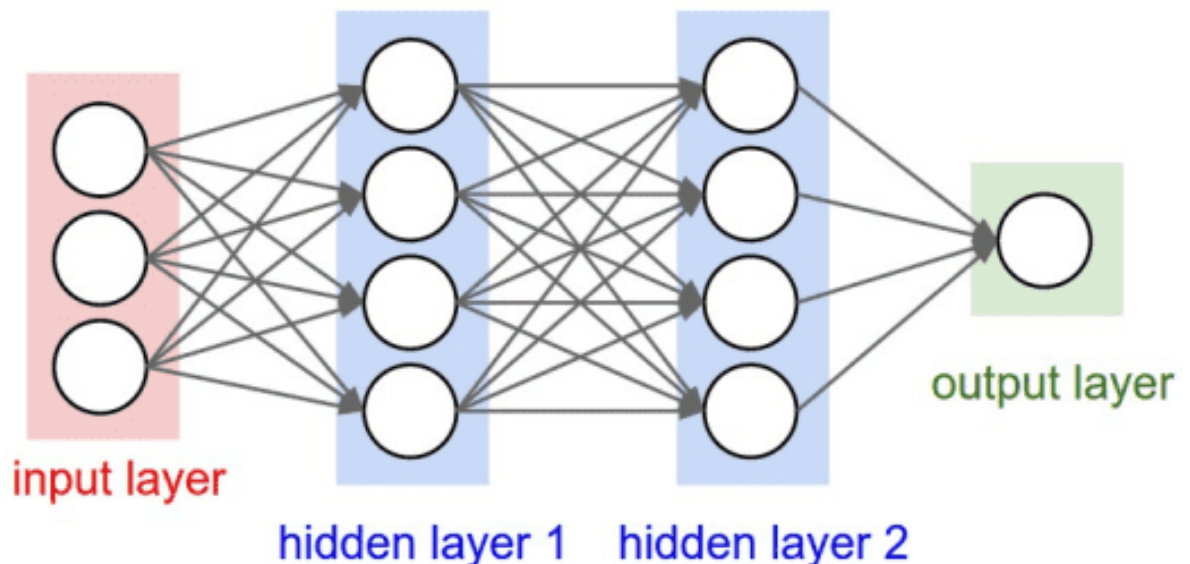


Рисунок 2.11. Нейронна мережа "feedforward".

У прямій мережі кожен нейрон в одному шарі має лише спрямовані з'єднання з нейронами наступного шару (до вихідного шару). Ми часто стикаємося з прямими мережами, у яких кожен нейрон пов'язаний з усіма нейронами наступного шару (ці шари називаються повністю пов'язаними).

Рівень нейронів мережі *обумовлений feedforward* чітко розділені: один вхідний шар, один вихідний шар і один або кілька шарів обробки, які невидимі зовні (також звані прихованими шарами). З'єднання дозволені лише для нейронів наступного шару. У мережі зі зворотним зв'язком вихідний нейрон передає результат іншому нейрону, що знаходиться на тому або на попередньому рівні.

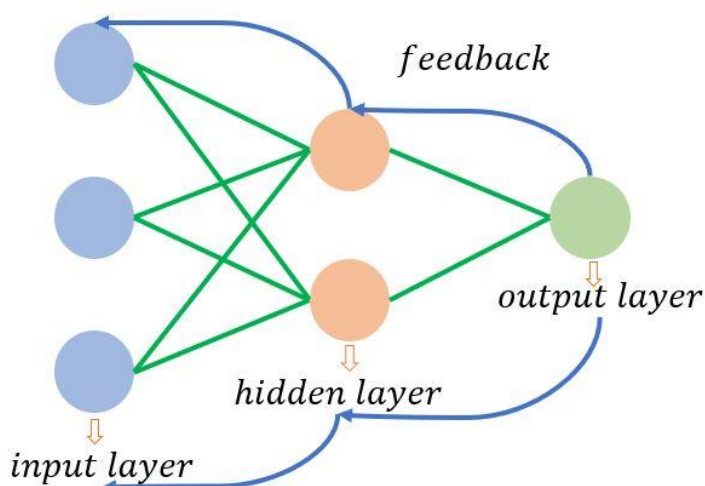


Рисунок 2.12. Нейронна мережа "feedback".

Нейронна мережа " **Hopfield** " має один шар, де кожен нейрон може бути як входом, так і виходом. Коротше кажучи, «мережа **Хопфілда** » складається з множини K повністю пов'язаних нейронів з бінарною активацією, симетричним балансом між окремими нейронами і залежить від того, з яким нейроном пов'язаний . Функція активації нейронів є бінарною граничною функцією з

виходами $\in \{1, -1\}$. Отже, $|K|$ нейрони з двома можливими станами $\in \{-1, 1\}$ можуть бути описані $x \in \{-1, 1\}^{|K|}$.

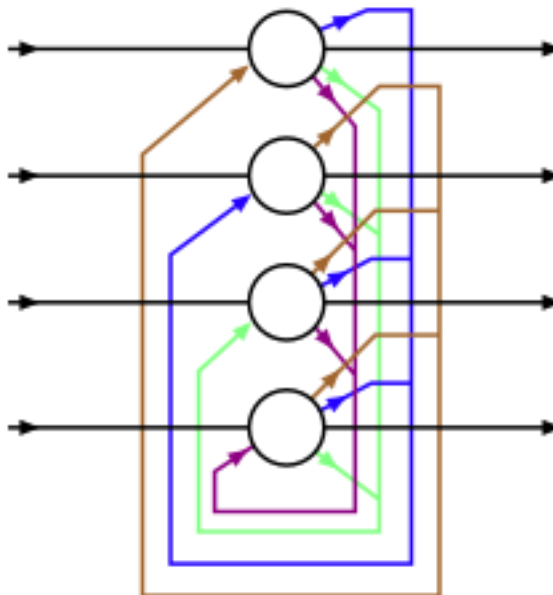


Рисунок 2.13. Мережа “*Hopfield*” або мережа із симетричним

2.4 Нейромережеве моделювання інформаційних процесів

Нейронні мережі стають дедалі популярнішими для завдання моделювання інформаційних процесів. Розглянемо моделі нейронних мереж, які використовуються як окремо, так і комбіновано для моделювання.

Модель мережі "Хеммінга"

Мережі Хеммінга були спроектовані явно для вирішення проблеми розпізнавання двійкових образів (де кожен елемент вхідного вектора має тільки два можливі значення, такі як 1 або -1).

Ця мережа називається мережею Хеммінга, тому що нейрон у шарі з прямим зв'язком з найбільшим виходом буде відповідати шаблону прототипу, який найбільш близький у відстані Хеммінга до шаблону введення. (Відстань

Хеммінгу між двома векторами дорівнює числу елементів, які відрізняються одна від одної. Це визначені тільки для двійкових векторів.)

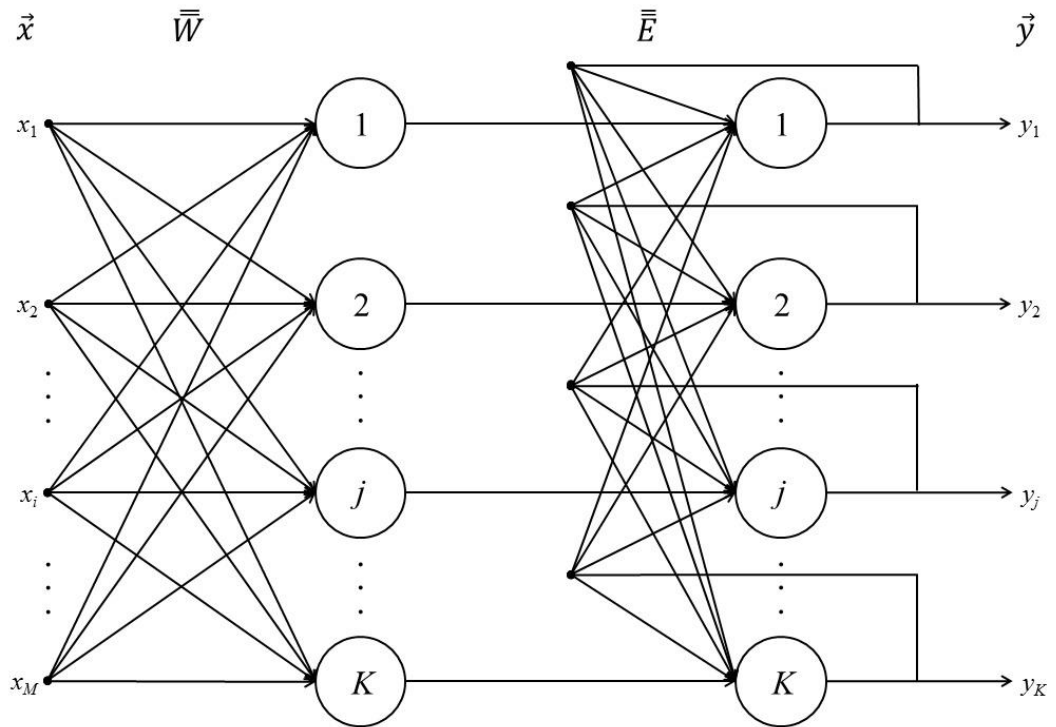


Рисунок 2.14. Мережа "Хеммінгу".

Нейрони в шарі feedback ініціалізуються виходами шару feedforward, які вказують на кореляцію між шаблонами прототипів та вхідний вектор. Потім нейрони конкурують один з одним, щоб визначити переможця. Після цього тільки один нейрон має відмінний вихід від нуля. Який переміг нейрон вказує, яка категорія введення була представлена в мережу. Кореляція - це одиниця, яка описує ступінь подібності між двома змінними / зразками:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (2.10)$$

Мережа включає як feedforward, і feedback шари.

Модель мережі "Кохонена".

Найвідомішою і найпопулярнішою моделлю мереж, що самоорганізуються, є карта збереження топології, запропонована Теуво Кохоненом. Так звані мережі Кохонена є втіленням деяких ідей, розроблених Розенблаттом, Фон-дер-Малсбургом та іншими дослідниками. Якщо вхідний простір має оброблятися нейронною мережею, першим питанням важливості є структура цього простору.

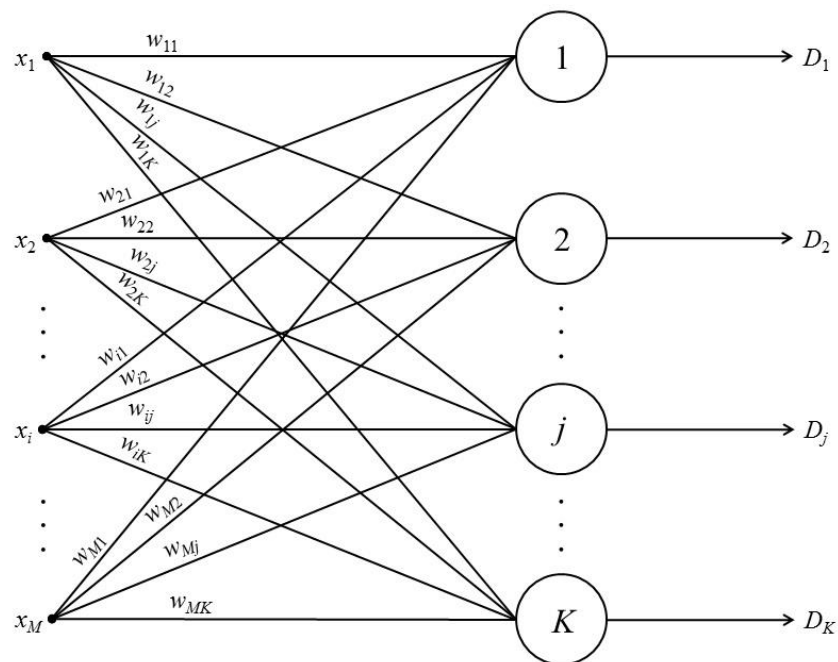


Рисунок 2.15. Мережа "Кохонена"

Нейронна мережа з реалістичними обчисленнями обчислює функцію f , що визначається з вхідного простору A у вихідний простір B . Область, де f визначена, може бути покрита мережею Кохонена. Мережі Кохонена вчать створювати карти вхідного простору способом, що самоорганізується.

Модель Кохонена має біологічний та математичний фон. У нейробіології добре відомо, що багато структур мозку мають лінійну або плоску топологію, тобто вони поширюються в одному або двох вимірах. Мережі Кохонена є

пристрої обчислювальних вузлів в одно-, дво- або багатовимірних решітках. У вузлів є бічні зв'язки із кількома сусідами.

Прикладами таких бічних зв'язків є інгібуючі зв'язки. Вхідний нейрон з'єднується з першим вузлом (нейроном) шару Кохонена, вагою . Кожен вузол виробляє - один вихідний нейрон.

$$OUT_k = w_{A1-Bk} \times x_1 + w_{A2-Bk} \times x_2 + w_{A3-Bk} \times x_3 \quad (2.11)$$

Сітка обчислювальних елементів дає змогу ідентифікувати безпосередніх сусідів. Це дуже важливо, оскільки під час навчання вагові обчислювальні одиниці та їхні сусіди було оновлено. Об'єктом такого підходу навчання є те, що сусідні підрозділи вчаться реагувати на тісно пов'язані сигнали.

Перцептрон: одношаровий та багатошаровий.

Як згадувалося історія нейронних мереж, перцептрон був описаний Френком Розенблаттом в 1958 року. Спочатку Розенблатт визначав вже обговорювану виважену суму та нелінійну активаційну функцію як компоненти перцептрона.

Не існує встановленого визначення перцептрона, але більшу частину часу цей термін використовується для опису мережі з прямим з'єднанням за допомогою ярликів. Ця мережа має шар нейронів сканера (сітківка) із статично зваженими сполуками з наступним шаром, званим вхідним шаром; але вага решти верств може бути змінено. Усі нейрони, підлеглі сітківці, є детекторами.

Тут ми спочатку використовуємо бінарний перцептрон з кожним вихідним нейроном, що має рівно два можливі значення (наприклад,

$\{0,1\}$ або $\{-1,1\}$).

Таким чином, бінарна функція порога використовується як функція активації залежно від значення порогового вихідного нейрона. У певному сенсі функція двійкової активації є запитом

IF , який також можна заперечувати за допомогою негативних ваг. Таким чином, перцептрон можна використовувати для обробки логічної інформації.

Перцептрон є прямою мережею, що містить сітківку (сканер), яка використовується тільки для збору даних і має фіксовані зважені з'єднання з першим нейронним шаром (вхідним шаром). За фіксованим ваговим шаром слід щонайменше один навчальний ваговий шар. Один шар нейрона пов'язаний з наступним шаром.

Перший шар перцептрона складається із вхідних нейронів. Тут з'єднання з вагами, що навчаються, йдуть від вхідного рівня до вихідного нейрона, який повертає інформацію про те, був визнаний шаблон, введений у вхідні нейрони, чи ні.

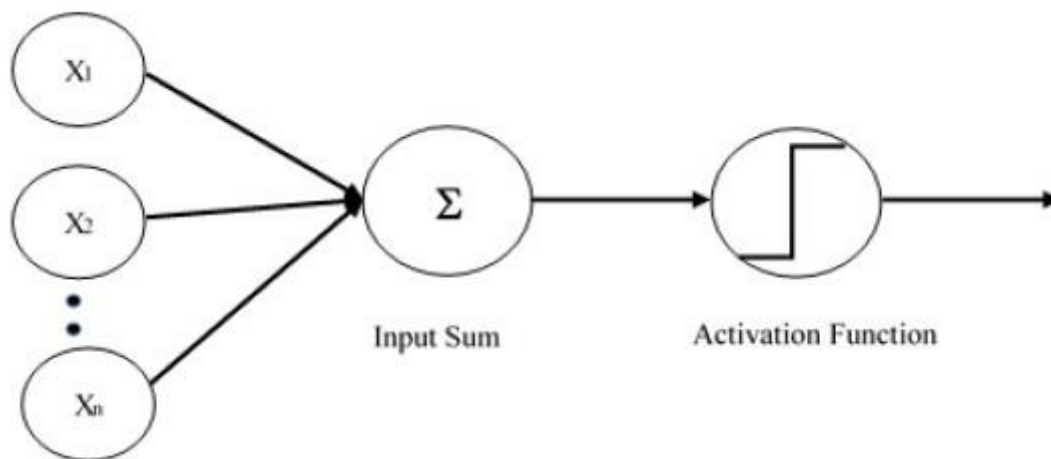


Рисунок 2.16. Одношаровий перцептрон.

Таким чином, одношарове сприйняття (скорочено SLP) має лише один рівень тренованих ваг. Одношаровий перцептрон (SLP) є перцептроном, що має лише один шар змінних ваг та один шар вихідних нейронів.

Перцептрон з двома або більше ваговими шарами, що тренуються, є більш потужним, ніж SLP. Перцептрони з більш ніж одним шаром змінно зважених сполук називаються багатошаровими перцептронами (MLP).

Таким чином, n - шаровий або n - ступінчастий перцептрон має рівно n шарів зі змінною вагою і $n + 1$ нейронних шарів (тут не враховується сітківка), причому шар нейрона 1 є вхідним шаром.

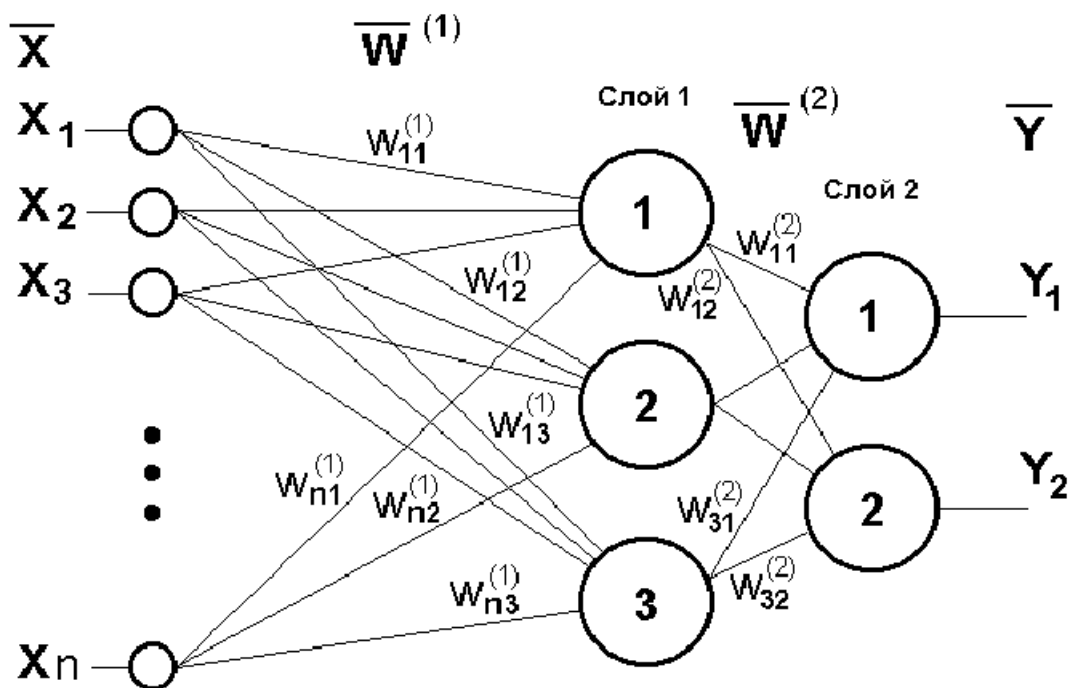


Рисунок 2.17. Багатошаровий перцептрон.

З цього випливає що n -етапний перцептрон має рівно n вагових шарів, що навчаються.

Усі перелічені моделі нейронної мережі застосовують у системах для моделювання інформаційних процесів, незалежно від застосування цих моделей як окремо, і злиттям, тобто. гібридами цих моделей.

РОЗДІЛ 3. НЕЙРО-НЕЧІТКИ МЕТОДИ ТА ЗАСОБИ МОДЕЛЮВАННЯ СЛАБОСТРУКТУРОВАНИХ ІНФОРМАЦІЙНИХ ПРОЦЕСІВ

3.1. Нейро-нечітке моделювання

Soft-computing включають в нечітку логіку, нейронні мережі, розподіл усіх міркування і генетичні алгоритми. Сьогодні методи чи комбінація методів із усіх цих областей використовуються для розробки розвідувальної системи. Нейронні мережі забезпечують алгоритми навчання, класифікації та оптимізації, тоді як нечітка логіка має справу з такими питаннями, як формування вражень та міркувань на семантичному чи лінгвістичному рівні.

Ймовірні міркування стосуються невизначеності. Хоча існують суттєві області перекриття між нейронними мережами, FL і ймовірні міркування, в цілому вони швидше доповнюють один одного, ніж конкурують. Останнім часом використовувалися багато інтелектуальних систем, звані нейронечіткими системами. Існує безліч способів комбінування нейронних мереж та методів FL.

Нейро-нечітке моделювання було визнано потужним інструментом, який може сприяти ефективній розробці моделей шляхом поєднання інформації з різних джерел, таких як емпіричні моделі, евристика та дані.

Нейро-нечіткі моделі описують системи за допомогою нечітких правил if-then, таких як «Якщо x маленький, то y є великим», представленим у мережевій структурі, до якої можуть бути застосовані алгоритми навчання, відомі в галузі ІНС. Завдяки цій структурі, нейро-нечіткі моделі в певній мірі прозорими для інтерпретації, аналізу, тобто можуть бути краще застосовані для пояснення рішень для користувачів, ніж "чорна скринька", із застосуванням нейронних мереж.

І нейронні мережі, і нечіткі системи мотивовані імітацією процесів людського мислення. У нечітких системах відносини явно представлені як правил if-then. У нейронних мережах відносини не зазначені явно, а «закодовані» у мережі та її параметрах.

На відміну від методів, що базуються на знаннях, явне знання не потрібне для застосування нейронних мереж. Нейро-нечіткі системи об'єднують семантичну прозорість нечітких систем на основі правил із можливостями навчання нейронних мереж.

На обчислювальному рівні нечітку систему можна розглядати як багаторівневу структуру (мережу), подібну до штучних нейронних мереж типу RBF. Для оптимізації параметрів у нечіткій системі можуть використовуватися градієнтні методи навчання, відомі в галузі нейронних мереж.

Отже, цей підхід зазвичай згадується як нейронечітке моделювання. Розглянемо спочатку простий приклад нечіткої моделі TS нульового порядку з наступними двома правилами:

$$\begin{aligned} \text{If } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{21} \text{ then } y &= b_1 \\ \text{If } x_1 \text{ is } A_{12} \text{ and } x_2 \text{ is } A_{22} \text{ then } y &= b_2 \end{aligned} \quad (3.1)$$

На рисунку 3.1 показано мережне уявлення цих двох правил. Вузли у першому шарі обчислюють ступінь належності входів в антецедентних нечітких множинах. Вузли продукту Π у другому шарі є антецедентною зв'язкою (тут оператор «і»). Вузол нормування N та вузол підсумовування Σ реалізують нечіткий середній оператор.

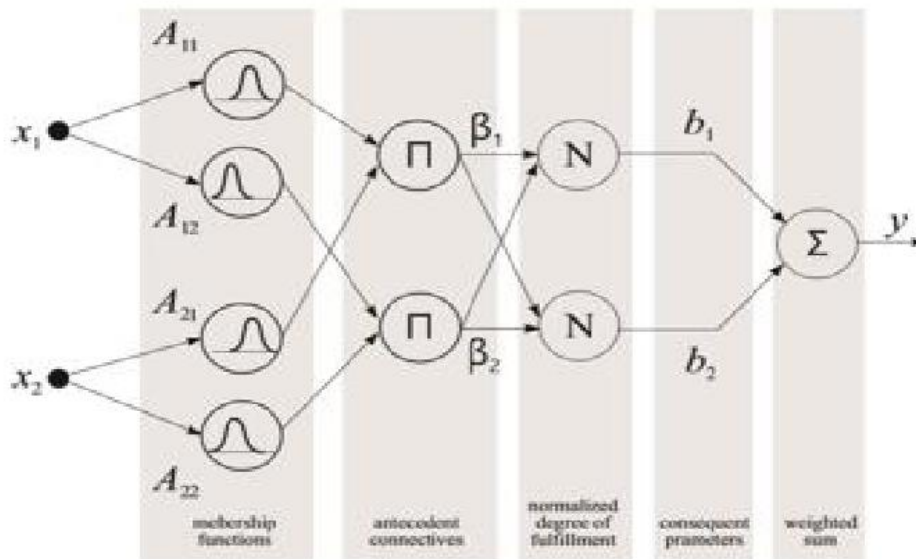


Рисунок 3.1. Приклад нечіткої моделі TS нульового порядку з двома правилами, представленими у вигляді нейронечіткої мережі.

Ця система називається ANFIS – Adaptive NeuroFuzzy Inference System.

Як правило, використовуються гладкі антецедентні функції приладдя, такі як гаусівські функції:

$$\mu_{A_{ij}}(x_j; c_{ij}, \sigma_{ij}) = \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right) \quad (3.2)$$

Рівняння Вхід вихід спільної моделі TS нульового порядку з кон'юнктивною формою антецеденту:

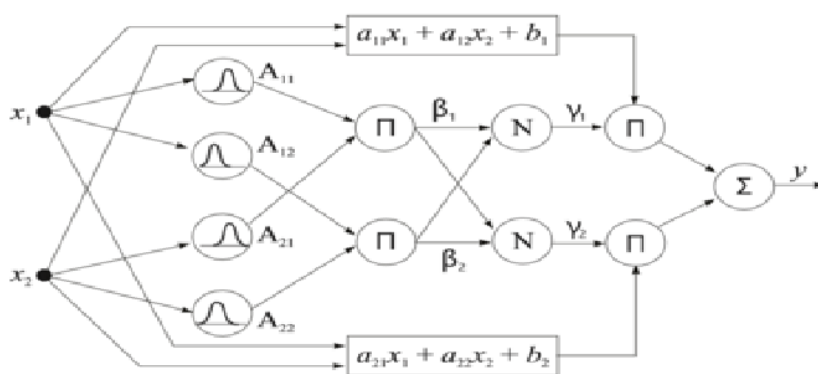
$$y = \sum_{i=1}^K \gamma_i(\mathbf{x}) b_i \quad \text{with} \quad \gamma_i(\mathbf{x}) = \frac{\prod_{j=1}^p \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right)}{\sum_{i=1}^K \prod_{j=1}^p \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right)} \quad (3.3)$$

Аналогічну модель можна у вигляді нечіткої моделі TS першого порядку.

Розглянемо знову приклад із двома правилами:

$$\begin{aligned}
 &\text{If } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{21} \text{ then } y_1 = a_{11}x_1 + a_{12}x_2 + b_1 \\
 &\text{If } x_1 \text{ is } A_{12} \text{ and } x_2 \text{ is } A_{22} \text{ then } y_2 = a_{21}x_1 + a_{22}x_2 + b_2
 \end{aligned}
 \tag{3.4}$$

для яких відповідна мережа наведена на рисунку 3.2.



(3.5)

Рисунок 3.2. Приклад нечіткої моделі TS першого порядку з двома правилами, представленими як нейро-нечіткої мережі, званої ANFIS.

Рівняння вхід-вихід моделі TS першого порядку:

Далі розглянемо нейро-нечіткі моделі з їхньою архітектурою.

$$y = \sum_{i=1}^K \gamma_i(\mathbf{x}) (\mathbf{a}_i^T \mathbf{x} + b_i)
 \tag{3.6}$$

Узагальнено-орієнтовний підхід на основі інтелектуального управління (модель GARIC).

GARIC – це розширена версія орієнтовного підходу Berenji на основі інтелектуального управління, який реалізує нечіткий контролер, використовуючи кілька спеціалізованих нервових мереж з прямою передачею. Він складається з мережі оцінки стану дії (AEN) та мережі вибору дій (ASN). Архітектура GARICASN зображена на рисунку 3.3.

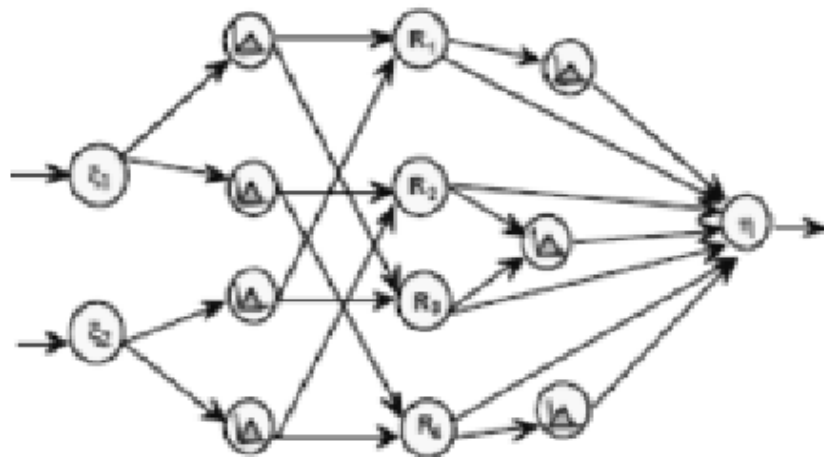


Рисунок 3.3. Архітектура GARICASN.

ASN GARIC - це мережа з прямим зв'язком із п'ятьма шарами.

1-й прихований шар зберігає лінгвістичні значення всіх вхідних змінних. Кожна вхідна одиниця пов'язана тільки з тими одиницями 1-го прихованого шару, які представляють пов'язані з ним мовні значення.

2-й прихований шар представляє вузли нечітких правил, які визначають рівень виконання правила за допомогою операції *softmin*.

3-й прихований шар представляє лінгвістичні значення керуючої вихідної змінної η .

Висновки правила обчислюються залежно від сили антецедентів правил, обчислених лише на рівні вузла правила. GARIC використовує метод локального середнього максимуму обчислення виходів правил. Цей метод вимагає чіткого вихідного значення кожного правила.

Таким чином, висновки повинні бути деактивовані перед тим, як вони накопичуються до кінцевого вихідного значення контролера. Garic використовує суміш градієнтного спуску і підготовки для підвищення точного налаштування параметрів вузла. Гібридне навчання припиняється, якщо АЕН вихід перестає змінюватися. Процес навчання дуже трудомісткий і складний архітектура Garic можна розглядати основні недоліки Гарика.

Нейро-нечіткий контроль (модель NEFCON).

Алгоритм навчання, визначений для NEFCON, здатний вивчати нечіткі множини, а також нечіткі правила, що реалізують FIS типу Mamdani. Цей метод можна розглядати як розширення для GARIC, яке також використовує навчання підкріплення, але потребує раніше певної бази правил. На рисунку 3.4 показана базова архітектура NEFCON з двома входами та п'ятьма нечіткими правилами.

Внутрішні вузли R_1, \dots, R_5 являють собою правила, вузли ξ_1, ξ_2 і η - вхідні та вихідні значення, а μ_r, V_r - нечіткі множини, що описують антецеденти та наслідки. На відміну від нейронних мереж, з'єднання NEFCON зважуються з нечіткими множинами замість реальних чисел. Правила з тим самим антецедентом використовують так звані загальні ваги, представлені еліпсами, намальованими навколо сполук, як показано на рисунку 3.4. Вони забезпечують цілісність основи правил. База знань нечіткої системи неявно визначається структурою мережі.

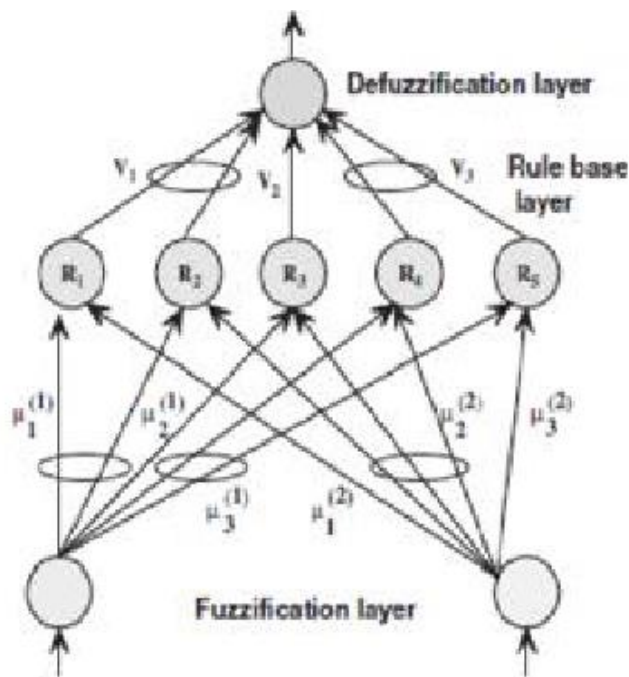


Рисунок 3.4. Архітектура NEFCON.

Модулі введення приймають від завдання інтерфейсу фазифікації, логіка виведення представлена функціями поширення, а вихідним модулем є інтерфейс дефазифікації.

Процес навчання моделі NEFCON можна поділити на дві основні фази. Навчання інкрементного правила використовується, коли коректний висновок невідомий та правила створюються на основі оцінних вихідних значень. У міру проходження навчання відповідно до вимоги додається більше правил.

Для навчання декрементного правила спочатку правила створюються з нечітких розділів змінних процесів, а непотрібні правила усуваються у процесі навчання. Освіта декрементное правило є менш ефективним для порівняння з поетапним підходом.

Через складність необхідних обчислень правило декрементного навчання може бути використане тільки в тому випадку, якщо є лише кілька

вхідних змінних з невеликою кількістю нечітких множин. Для великих систем правило інкрементного навчання буде оптимальним.

Попереднє знання, коли воно є, може бути включене для зниження складності навчання. Функції власності у основі правил змінюються відповідно до алгоритму поширення нечіткої помилки (FEBP). Алгоритм FEBP може адаптувати функції власності і може застосовуватися тільки в тому випадку, якщо існує основа нечітких правил.

Ідея алгоритму навчання ідентична: збільшує вплив правила, якщо його дія йде у правильному напрямку (винагорода) і зменшує його вплив, якщо правило поводиться продуктивно (карає). Якщо немає абсолютно ніякої інформації про початкову функцію власності, слід використовувати рівномірне нечітке розбиття змінних.

Нечіткі нейронні мережі (моделі EFuNN і dmEFuNN), що розвиваються (еволюційні).

EFuNN та dmEFuNN засновані на структурі ECOS (еволюціонуючих систем з'єднань) для адаптивних інтелектуальних систем, сформованих через еволюцію та інкрементне, гібридне (контрольоване/неконтрольоване) онлайн навчання. Вони можуть розмістити нові вхідні дані, включаючи нові функції, нові класи і т.д. за допомогою локального налаштування елементів. У EFuNN усі вузли створюються під час навчання. EFuNN має п'ятишарову архітектуру, як показано на рисунку 3.5. Вхідний рівень є буферний шар, що представляє вхідні змінні. Другий рівень вузлів є нечітким квантування кожного вхідного простору змінних. Кожен вхідний змінний представлено тут група просторового розташованих нейронів для подання нечіткого квантування цієї змінної. Вузли, які мають функції приналежності (трикутні, гауссовские тощо. буд.), може бути змінено під час навчання.

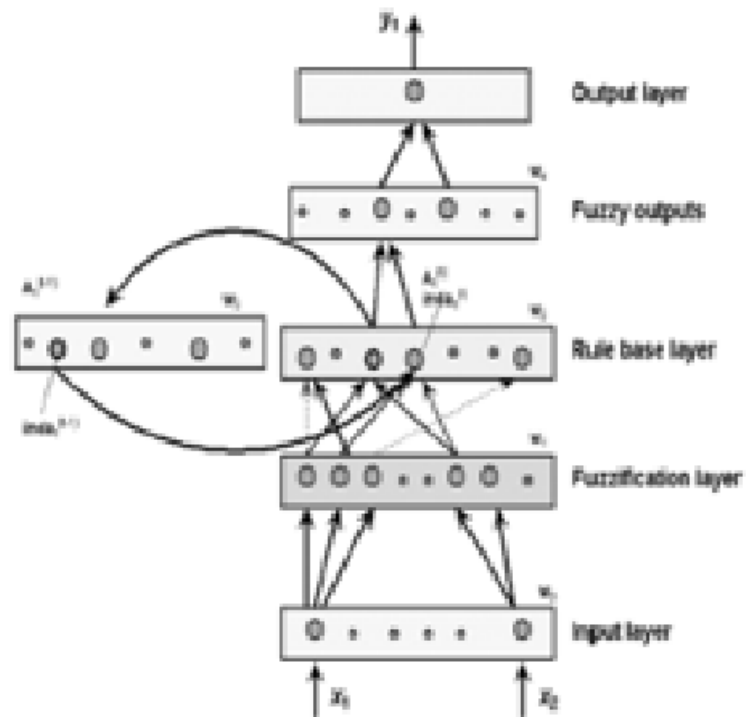


Рисунок 3.5. Архітектура EFuNN.

Третій рівень містить вузли правил, що розвиваються через гібридне (контрольоване/неконтрольоване) навчання.

Вузли правил є прототипи асоціацій даних введення-виведення, графічно представлених як асоціація гіперсфер з нечітких вхідних і нечітких вихідних просторів. Кожен вузол правила r визначається двома векторами ваги з'єднання: $W_1(r)$ і $W_2(r)$, причому останній коригується за допомогою контрольованого навчання на основі помилки виходу, а перша коригується за допомогою неконтрольованого навчання на основі вимірювання подібності в локальній області простору проблеми введення.

Четвертий шар нейронів є нечітким квантування для вихідних змінних.

П'ятий рівень є реальними значеннями для вихідних змінних. У разі "oneof-n" EFuNN максимальна активація вузла правила поширюються на наступний рівень. У разі «багато-оф-п» режимі, всі значення активації правил

вузлів, які перевищують поріг активації додатково розмножують в структурі сполуки.

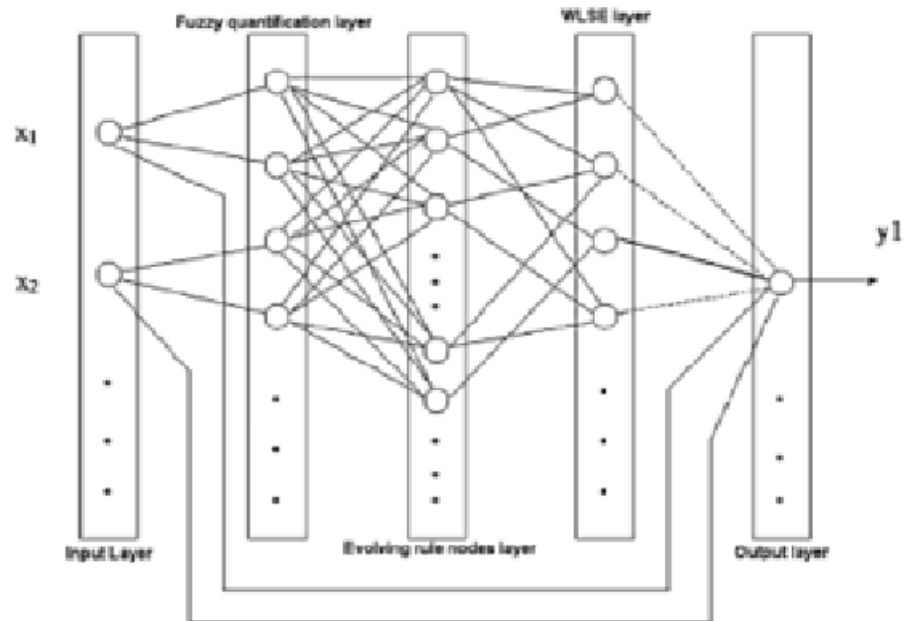


Рисунок 3.6. Архітектура dmEFuNN.

Динамічна еволюційна модель нечітких нейронних мереж (dmEFuNN) розроблена з думкою, що поширюється як активація вузла виграшного правила, а й група вузлів правил динамічно обраних кожного нового вхідного вектора, та його значення активації використовуються розрахунку динамічних параметрів вихідний функції. У той час як EFuNN використовує нечіткі правила типу Mamdani, dmEFuNN використовує нечіткі правила Takagi-Sugeno. Архітектура зображено рисунку 3.6.

Перший, другий і третій рівні dmEFuNN мають такі самі структури та функції, що й EFuNN.

Четвертий шар, шар нечіткого виведення, вибирає m вузлів правил з третього рівня, які мають найближчу нечітку нормовану локальну відстань до нечіткого вхідного вектора, а потім нечітке правило Takagi Sugeno сформується з використанням зваженої оцінки найменших квадратів.

Останній рівень обчислює висновок $dmEFuNN$. Число m активованих вузлів, що використовуються для обчислення вихідних значень $dmEFuNN$, не менше кількості вхідних вузлів плюс один. Як і $EFuNN$, $dmEFuNN$ можна використовувати як для автономного навчання, так і для онлайн-навчання, таким чином оптимізуючи глобальну помилку узагальнення або локальну помилку узагальнення.

У $dmEFuNN$ для нового вхідного вектора (для якого вихідний вектор невідомий) знайдено підпростір із m вузлів правила, нечітке правило Такагі Сугено першого порядку формується за допомогою методу найменших квадратів. Це правило використовується для обчислення вихідного значення $dmEFuNN$.

Таким чином, $dmEFuNN$ діє як універсальний апроксиматор функції, використовуючи m лінійних функцій у малому m -мірному вузлі. Точність наближення залежить від розміру вузлових підпросторів, чим менше підпростір, тим вища точність.

3.2. Моделювання слабоструктурованих процесів методами та засобами ANFIS

Як попередні знання, так і дані процесу можуть бути використані для побудови нейронечітких систем. Попереднє знання може мати досить приблизний характер (якісний, евристичний). Можна виділити два основних підходи до інтеграції знань та даних:

Знання експерта формулюється як сукупність правил *if-then*. У такий спосіб створюється вихідна модель. Параметри цієї моделі (функції приладдя, наступні параметри) потім налаштовуються за допомогою даних процесу.

Нечіткі правила (включно з пов'язаними параметрами) будуються з нуля з використанням числових даних. У цьому випадку перевагою використання

нейронечіткої моделі є можливість інтерпретувати отриманий результат (що неможливо з “чорними ящиками”, як нейронні мережі).

Експерт може впоратися з інформацією, що зберігається в базі правил, власними знаннями, змінювати правила або надавати додаткові, щоб продовжити термін дії моделі тощо.

Згадані вище методи, звичайно, можуть бути об'єднані в залежності від проблеми.

Структура та параметри.

Двома основними етапами системної ідентифікації є структурна ідентифікація та оцінка параметрів. Вибір структури моделі дуже важливий, оскільки визначає гнучкість моделі в наближенні (невідомих) систем. Модель з багатою структурою може апроксимувати складніші функції, але водночас матиме найгірші властивості узагальнення. Хороше узагальнення означає, що модель, прив'язана до однієї множини даних, також добре працює на іншій множині даних у тому самому процесі.

У нейро-нечітких моделях процес вибору структури включає такі основні варіанти:

- Вибір вхідних змінних. Це пов'язано не тільки з фізичними входами u , але і з динамічними регресорами, що визначаються вхідними та вихідними лагами, n_y та n_u відповідно.
- Кількість та тип функцій власності, кількість правил. Ці два структурні параметри взаємопов'язані (для більшої кількості функцій приналежності необхідно визначити більше правил) та визначити рівень деталізації, званий деталізацією моделі.

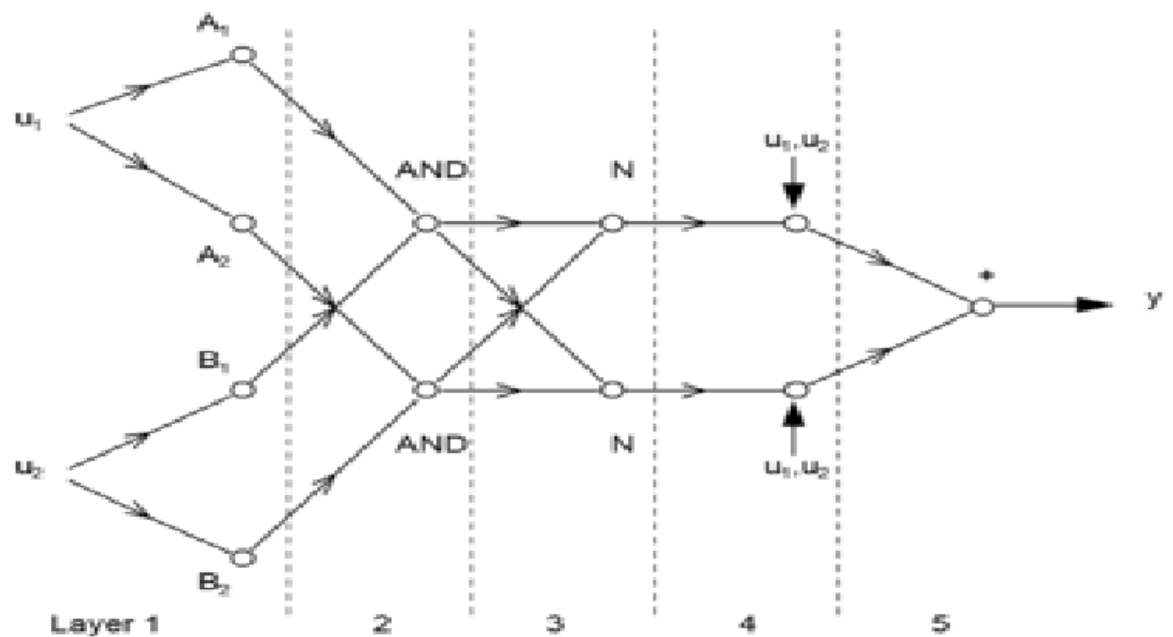


Рисунок 3.7. Структура мережі ANFIS

Попередні знання, розуміння поведінки процесу та мета моделювання є типовими джерелами інформації для вибору вихідного набору можливих входів. Автоматичний вибір даних може бути використаний для порівняння різних структур з погляду деяких певних критеріїв ефективності.

Мета моделювання та обсяг доступної інформації (знань та даних) будуть визначати цей вибір. Автоматизовані методи можуть використовуватись для додавання або видалення функцій та правил приладдя.

Алгоритм навчання ANFIS.

Коли вміщені параметри фіксовані, загальний вихід є лінійною комбінацією наступних параметрів. У символах вихід y можна записати у

вигляді:

$$\begin{aligned}
 y &= \frac{\alpha_1}{\alpha_1 + \alpha_2} y_1 + \frac{\alpha_2}{\alpha_1 + \alpha_2} y_2 \\
 &= \bar{\alpha}_1 (c_{11} u_1 + c_{12} u_2 + c_{10}) + \bar{\alpha}_2 (c_{21} u_1 + c_{22} u_2 + c_{20}) \\
 &= (\bar{\alpha}_1 u_1) c_{11} + (\bar{\alpha}_1 u_2) c_{12} + \bar{\alpha}_1 c_{10} + (\bar{\alpha}_2 u_2) c_{21} + (\bar{\alpha}_2 u_2) c_{22} + \bar{\alpha}_2 c_{20}
 \end{aligned}
 \tag{3.7}$$

яка, лінійна в наступних параметрах: c_{ij} ($i = 1, 2; j = 0, 1, 2$) гібридний алгоритм коригує наступні параметри c_{ij} у прямому проході та вміщені параметри $\{a_i, b_i, c_i\}$ у зворотному проході. При прямому проході вхідні дані мережі поширюються вперед на рівень 4, де наступні параметри визначаються методом найменших квадратів. У зворотному проході сигнали помилки розповсюджуються назад, а наведені параметри оновлюються за допомогою градієнтного спуску.

Оскільки правила оновлення для передумови та наступні параметри розв'язані в гібридному правилі навчання, прискорення обчислень може бути можливим за допомогою варіантів градієнтного методу або інших методів оптимізації на вихідних параметрах.

Оцінка глобальних найменших квадратів.

Глобальний метод оцінки найменших квадратів дає параметри, що мінімізують наступний критерій помилки прогнозування:

$$\theta = \arg \min \sum_{k=1}^N \left(y_k^* - \sum_{i=1}^K \gamma_i(\mathbf{x}_k) [\mathbf{x}_k^T \mathbf{1}] \theta_i \right)^2
 \tag{3.8}$$

де конкатенацією всіх індивідуальних правил векторних параметрів. Для матриць даних цей критерій можна переписати у матричній формі:

$$\begin{aligned}\theta &= \arg \min (\mathbf{y}^* - \Lambda \theta)^T (\mathbf{y}^* - \Lambda \theta) \\ \Lambda &= [\Gamma_1 \varphi \dots \Gamma_K \varphi],\end{aligned}\tag{3.9}$$

де Λ , тобто діагональна матриця, що має як її k -го діагонального елемента. Тоді оптимальне рішення виходить безпосередньо з використанням матричного псевдозворотного:

$$\theta = (\Lambda^T \Lambda)^{-1} \Lambda^T \mathbf{y}^*$$

Оцінка локальних найменших квадратів.

Хоча глобальне рішення дає мінімальну помилку прогнозу, він може змістити оцінки ефектів в якості параметрів локальних моделей. Якщо потрібні локально відповідні параметри моделі, слід використовувати зважений підхід найменших квадратів до кожного правила. Це робиться шляхом мінімізації множин K -зважених локальних критеріїв LS:

$$\theta_i = \arg \min (\mathbf{y}^* - \varphi \theta_i)^T \Gamma_i (\mathbf{y}^* - \varphi \theta_i), \quad i = 1, 2, \dots, K\tag{3.10}$$

для яких рішенням є:

$$\theta_i = (\varphi^T \Gamma_i \varphi)^{-1} \varphi^T \Gamma_i \mathbf{y}^*, \quad i = 1, 2, \dots, K\tag{3.11}$$

У цьому випадку наступні індивідуальні правила параметрів оцінюються незалежно один від одного, і тому результат не залежить від взаємодії правил.

Проте, водночас, виходить більша помилка прогнозування, ніж із глобальними найменшими квадратами.

При інтерпретації моделей ANFIS, отриманих із даних, необхідно знати компроміси між локальною та глобальною оцінкою. Обмежена та багатокритеріальна оптимізація також може застосовуватися для обмеження незалежності параметрів.

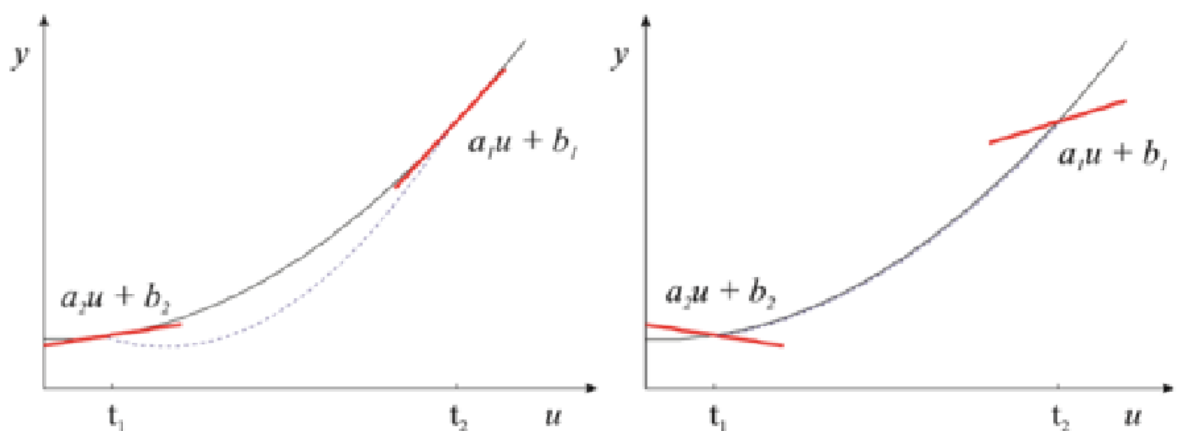


Рисунок 3.8. Результати локальної (лівої) та глобальної (правої) оцінки відповідних параметрів. Пунктирна лінія – це вихід моделі.

Обмежена оцінка.

Знання про динамічну систему, такі як її стабільність, мінімальний або максимальний статичний підсилення або час її встановлення, можуть бути переведені в опуклі обмеження на наступні налаштування. Використовуючи дані введення-виводу, оптимальні значення параметрів визначаються квадратичним програмуванням, а не найменшими квадратами.

Існує два типи обмежень: глобальний та локальний. Локальні обмеження є докладні знання, що стосуються кожного конкретного правила, тоді як глобальні обмеження відносяться до всієї моделі і тому повинні посилатися деякі глобальні системні властивості, такі як загальна стабільність.

Щоб переконатися у цьому, зрозуміємо, що можливу TS-модель можна як одну квазілінійну систему

$$y = \left(\sum_{i=1}^K \gamma_i(\mathbf{x}) \mathbf{a}_i^T \right) \mathbf{x} + \sum_{i=1}^K \gamma_i(\mathbf{x}) b_i = \mathbf{a}^T(\mathbf{x}) \mathbf{x} + b(\mathbf{x}) \quad (3.12)$$

з вхідними залежними «параметрами» $a(x)$, $b(x)$, які є опуклими лінійними комбінаціями індивідуальних послідовних параметрів a_i та b_i , тобто:

$$\mathbf{a}(\mathbf{x}) = \sum_{i=1}^K \gamma_i(\mathbf{x}) \mathbf{a}_i, \quad b(\mathbf{x}) = \sum_{i=1}^K \gamma_i(\mathbf{x}) b_i. \quad (3.13)$$

Це власність дозволяє нам визначати глобальні опуклі обмеження всієї моделі.

Помилка зворотного розповсюдження (Error Backpropagation).

Backpropagation - це процедура з градієнтним спусками (включаючи всі сильні та слабкі сторони градієнтного спуску) з функцією помилки, яка отримує всі n ваги як аргументи і передає сигнал про помилку, тобто. будучи n - мірним. На функції помилки, точка малої помилки або навіть точка найменшої помилки запитується за допомогою градієнтного спуску.

Різниця між бажаним виходом y^* та виходом мережі y називається помилкою. Ця помилка використовується для налаштування ваги в мережі шляхом мінімізації наступної функції витрат:

$$J = \frac{1}{2} \sum_{k=1}^N e_k^2 \quad e_k = y_k^* - y_k \quad (3.14)$$

Ініціалізація антецеденту функцій власності

Для успішного застосування градієнтного спуску до параметрів функції приналежності важливою є хороша ініціалізація. Декілька методів ініціалізації розглянуті в цьому розділі.

Функції приладдя на основі шаблонів.

З допомогою цього методу домени попередніх змінних є попередньо виділеними поруч функцій власності. Вони зазвичай рівномірно розподілені та мають форму. Потім створюється база правил, що охоплює всі комбінації антецедентних термінів.

Суворим недоліком такого підходу є те, що число правил моделі зростає експоненційно. Крім того, якщо не відомо, які змінні викликають нелінійність системи, всі попередні змінні зазвичай розподіляються рівномірно. Проте складність поведінки системи, зазвичай, нерівномірна. Деякі робочі області можна добре апроксимувати локальною лінійною моделлю, в той час як інші області вимагають відносно тонкого розділу.

Щоб отримати ефективне уявлення з мінімальною кількістю правил, функції приналежності мають бути розміщені так, щоб вони фіксували неоднорідну поведінку системи.

Дискретні способи пошуку.

Ітераційні алгоритми пошуку дерева можуть бути застосовані для декомпозиції антецедентного простору на прямокутники за допомогою ортогональних уламків. На кожній ітерації область з найгіршою локальною помилкою ділиться на дві половини (або інші частини). Вибираються розриви

у всіх вимірах вхідного сигналу та вибирається той, який має найбільше покращення продуктивності з найгіршою локальною відповідністю на даному етапі.

Це послідовне поділ припиняється, коли досягається конкретна метапомилки або коли досягається бажана кількість правил.

Перевагою такого підходу є його ефективність для високорозмірних даних та прозорості отриманого розділу.

Недоліком є те, що процедура побудови дерева є субоптимальною (жадібною), і, отже, кількість отриманих правил може бути досить великою з найгіршою локальною відповідністю на даному етапі.

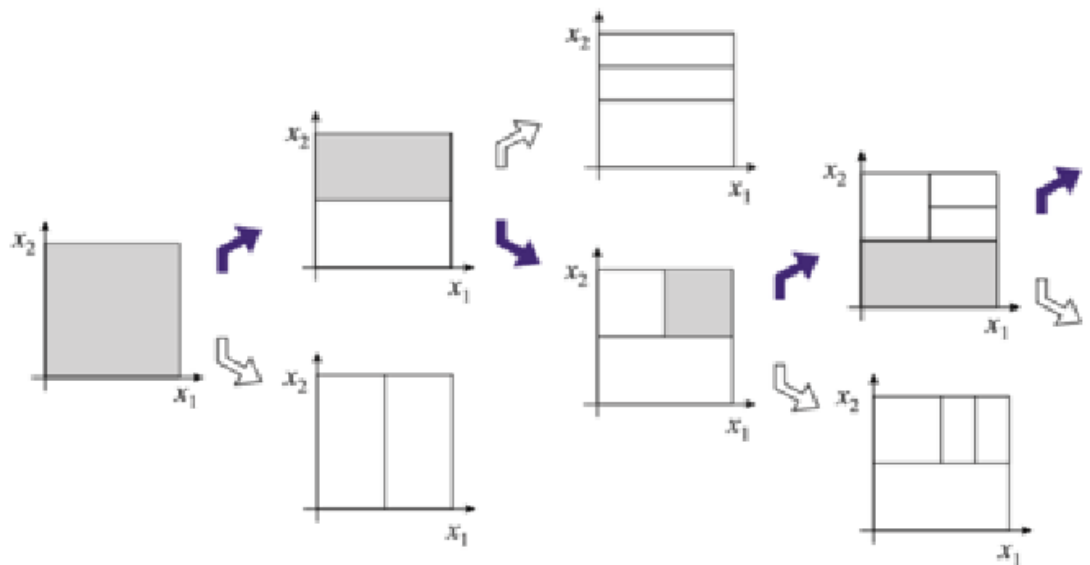


Рисунок 3.9. Антецедентне розкладання простору за допомогою

Нечітка кластеризація (Fuzzy Clustering).

Методи побудови, засновані на нечіткій кластеризації, виходять з аналізу даних та розпізнавання образів, де поняття нечіткої приналежності використовується для уявлення ступеня, в якому дані об'єкта схожі на якийсь прототипний об'єкт. Ступінь подібності може бути розрахована з використанням відповідного заходу відстані. Грунтуючись на схожості, ці вектори згруповані

таким чином, що дані в кластері у вигляді можливо подібним, і дані з різних кластерів, так різноманітні, як це можливо.

На рисунку 3.10 наведено приклад двох кластерів R^2 з прототипами v_1 і v_2 . Поділ даних виявляється у матриці нечіткого розбиття $U = [\mu_{ij}]$, елементами якої є належність ступеня векторів даних x_k у нечітких кластерах з прототипами v_j .

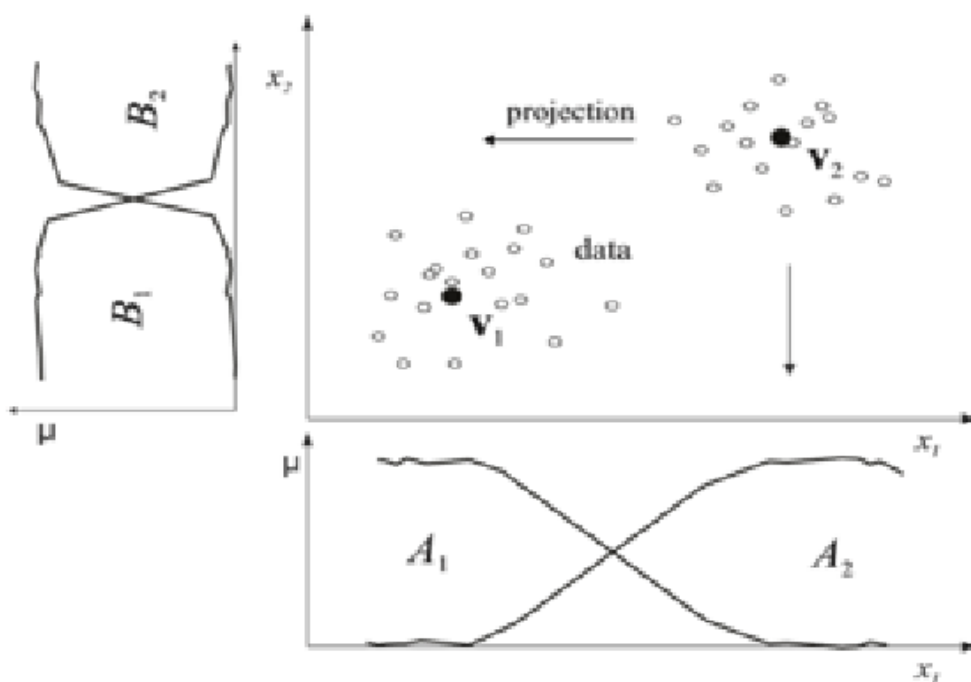
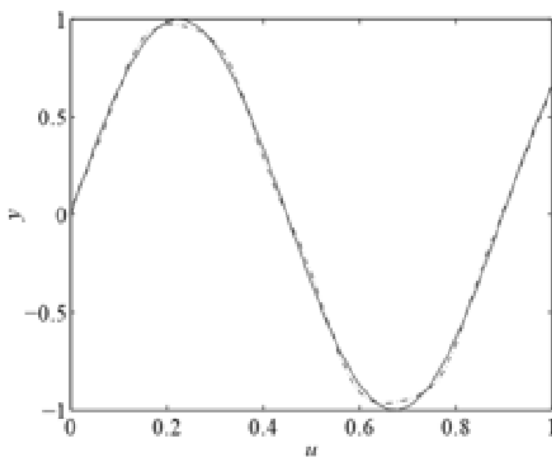


Рисунок 3.10. Ідентифікація функцій власності через нечітку кластеризацію.

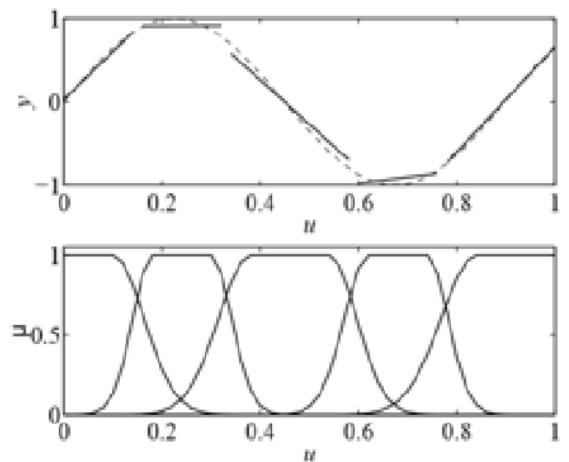
Потім антецедентні функції приналежності витягуються шляхом проектування кластерів деякі змінні. Для ініціалізації моделей ANFIS першого порядку прототипи можуть бути визначені як лінійні підпростори, або кластери є еліпсоїдами адаптивно визначеною формою. Кількість кластерів даних може бути визначено як пріоритет, який запитується автоматично, використовуючи критерії валідності кластерів і методи злиття.

Приклад моделювання (Статична функція)

У цьому розділі наводиться приклад моделювання, що ілюструє кілька важливих питань, пов'язаних із навчанням нейро-нечітких систем. Приклад простої задачі про нерухому статичну функцію. Він демонструє типову процедуру побудови нейронечіткої моделі.



3.11a Дані (суцільні) та моделі (Пунктирні).



3.11b Локальні моделі (зверху) та функції приналежності (внизу)

Численні результати показують, що покращення продуктивності досягається за рахунок отримання правил if-then, які не є повністю релевантними як локальний опис системи. Опишемо одновимірну функцію $y = \sin(u)$ моделлю ANFIS з наступними лінійними функціями. Ми вибираємо кількість правил, яких має бути п'ять, і побудуємо вихідну модель шляхом кластеризації даних $U \times Y$, використовуючи методологію, що базується на алгоритмі Густафсона-Кесселя. Отримано такі правила:

Якщо u дорівнює A_1 , то $y = 5$

$.721u + 0.030$ Якщо u дорівнює A_2 , то y

$= 0.035u + 0.904$ Якщо $u \in A_3$, то $y = -$

$5.302u + 2.380$ Якщо u дорівнює A_4 ,

$$\begin{aligned} \text{тоді } y &= 0.734u - 1.413 \\ \text{Якщо } u &\in A5, \text{ то } y \\ &= 6.283u - 5.623 \end{aligned}$$

Функція цієї вихідної моделі показано рисунку 3.11а. Функції приладдя та відповідні локальні моделі наведено на рисунку 3.11б.

Функції приладдя позначаються ліворуч від A1 до A5.

3.3 Опис роботи системи

Розглянемо моделювання за допомогою ANFIS-моделі такого слабоструктурованого процесу, як оцінка кредитного ризику. Для моделювання використовують програмне забезпечення MATLAB для створення предиктора ANFIS. Процедура та навчання ANFIS реалізуються з використанням стандартного нечіткого інструментарію у програмному забезпеченні MATLAB.

Кредит є основним продуктом, який банки пропонують своїм клієнтам, який генерує більшу частину прибутку, але водночас спричиняє найбільші втрати у разі дефолту.

Кредитний ризик – це ризик втрат через дефолт позичальника. Просте збільшення ризику може негативно вплинути на вартість портфеля. Ось чому рішення про надання чи відмову від кредиту є ключовим рішенням для банку. Кредитний ризик є складною проблемою виміру. Простого знання про непогашений кредит недостатньо, щоб оцінити його ризик. Можливі збитки залежать від розгляду зобов'язань, а також ймовірності дефолту контрагентів, що неможливо легко виміряти. Цей розділ містить опис наборів даних, що використовуються, і надає попередню обробку даних.

Набір даних використовується в взятий з банку. Багато містить безперервні змінні, такі як вік, стаж роботи, здатність погашення, які складають вхідні дані ANFIS. Зібрані конфіденційні дані, пов'язані з погашенням кредиту,

108 клієнтів у банку. Деякі з цих клієнтів відкладають погашення своїх позичок і вважаються клієнтом із високим ризиком.

Таблиця 3.1. Опис змінного "Вік".

Вік	Число	Процентне співвідношення
< 30 років	23	21,29%
30 років <= and < 60 років	81	75%
>= 60 років	4	3,71%
Усього	108	100%

Таблиця 3.2. Опис змінного «Стаж роботи».

Стаж роботи	Число	Процентне співвідношення
< 2 роки	6	5,55%
2 роки <= and < 10 років	64	59,26%
>= 10 років	38	35,19%
Усього	108	100%

Таблиця 3.3. Опис змінного «Здатність погашення».

Здатність погашення	Число	Процентне співвідношення
< 200 чоловік	7	6,5%

200 \leq and $<$ 400 людей	55	50,9%
\geq 400 чоловік	46	42,6%
Усього	108	100%

Таблиця 3.4. Значення використаних змінних у графіку функції власності.

Змінні	Низький	Середній	Високий
Вік	30	45	60
Стаж роботи	2	6	10
Здатність погашення	200	300	400

Нечітка система має три вхідні змінні. Кожна змінна має три функції приладдя.

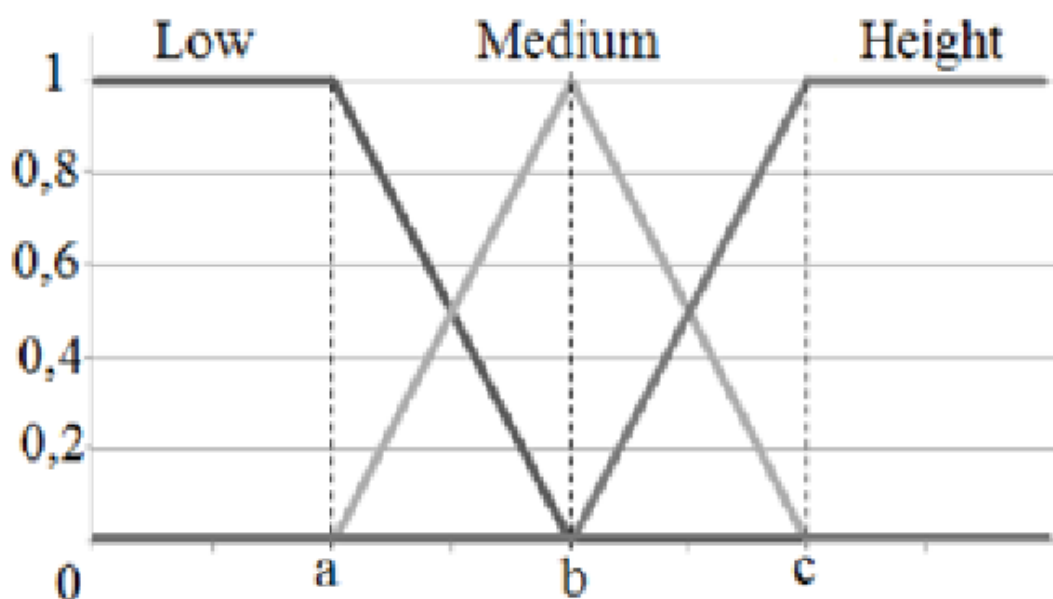


Рисунок 3.12. Графік функції ANFIS.

Можливі комбінації приналежності потім досягають 27. Під час запуску процесу навчання ANFIS автоматично обчислює ці комбінації, щоб вивести правила нечіткого виведення.

Грунтуючись на інтерв'ю з банківськими експертами, в тому числі 108 клієнтів, 51 клієнтів вважаються високим ризиком клієнтами. Ризикований клієнт призначив оцінку, що вказує на можливість затримки у погашенні боргів.

Мета моделі ціноутворення кредиту — передбачити цей показник для кожного клієнта на основі деяких пояснювальних змінних.

В якості виходу використовується DelProb (ймовірність затримки - ймовірність затримки). DelProb знаходиться відношенням кількості відкладених платежів на загальну кількість платежів.

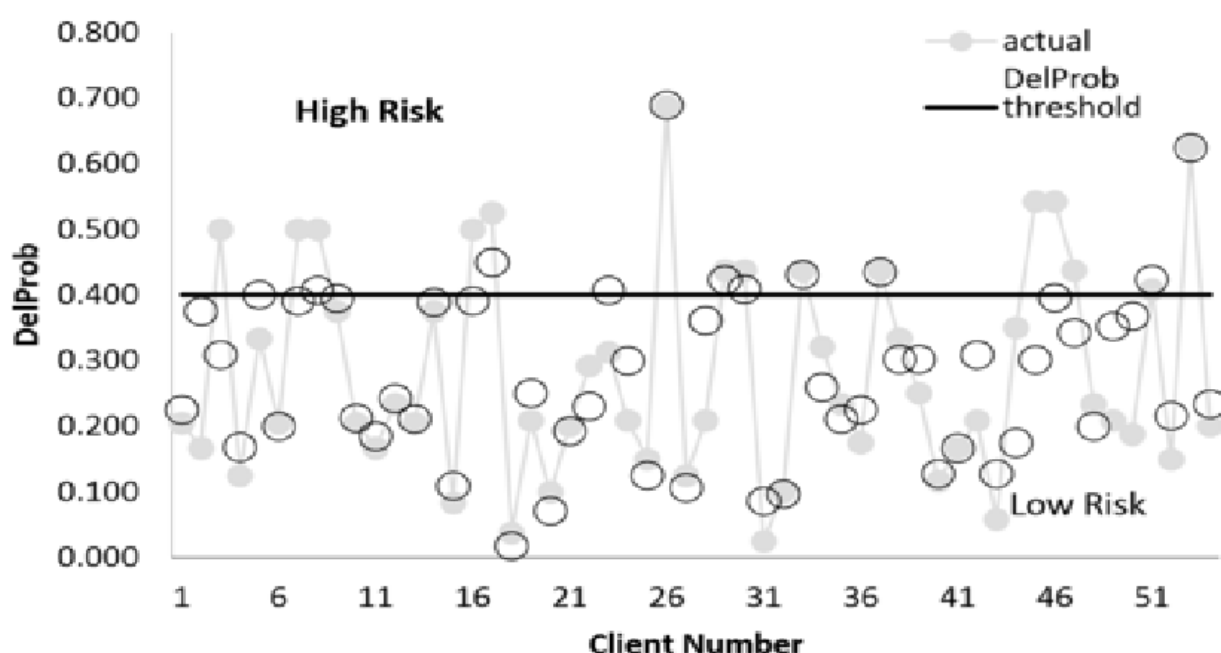


Рисунок 3.13. Результати ANFIS для кредитного скорингу.

Результати ANFIS для прогнозованого кредитного рейтингу для порівняння з фактичною ймовірністю затримки, показаною на рисунку 3.13.

Кредитний скоринг (рейтинг) зазвичай пов'язані з оцінкою потенційних ризиків, відповідних надання балів. Для порівняння результатів оцінки скорингу методом ANFIS використовується середня абсолютна відсоткова похибка (MAPE):

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (3.15)$$

де y_i і \hat{y}_i представляють фактичні та передбачувані ймовірності затримки i -го клієнта відповідно, а N – кількість клієнтів.

Як видно, предиктор ANFIS має задовільні результати для клієнтів із низьким рівнем ризику, проте його результати для клієнтів із високим рівнем ризику не дуже добрі та мають відносно більше помилок.

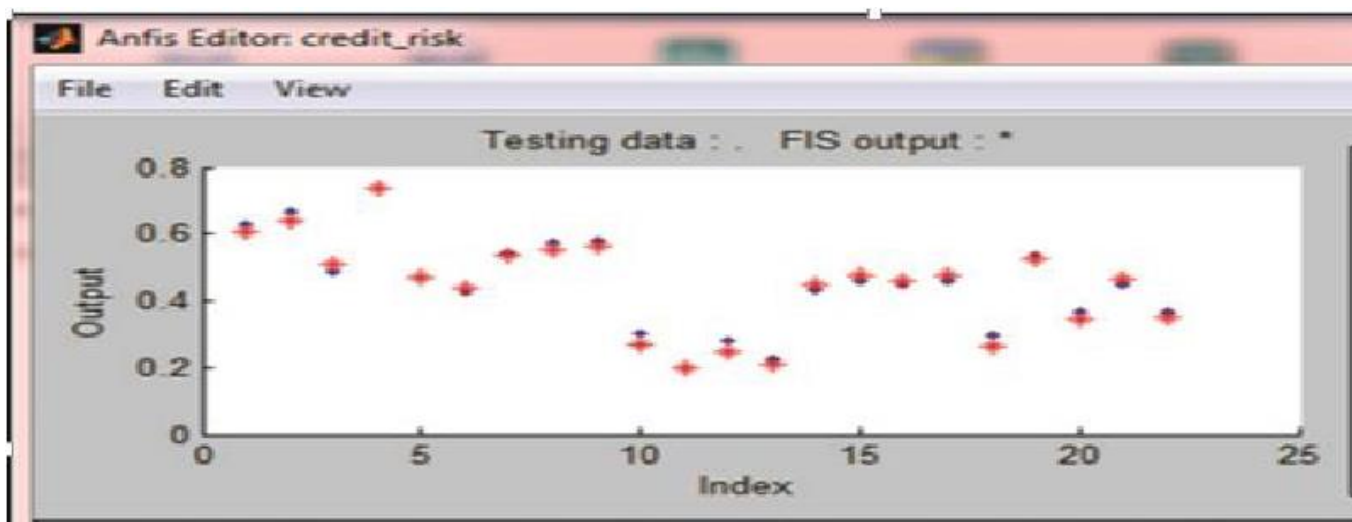


Рисунок 3.14. Виведення кредитного ризику для тестування даних .

Після завершення навчального процесу та етапів перевірки зовнішньої системи тестовані дані можуть бути введені в систему без представлення реальних результатів, щоб модель могла передбачити запитані значення.

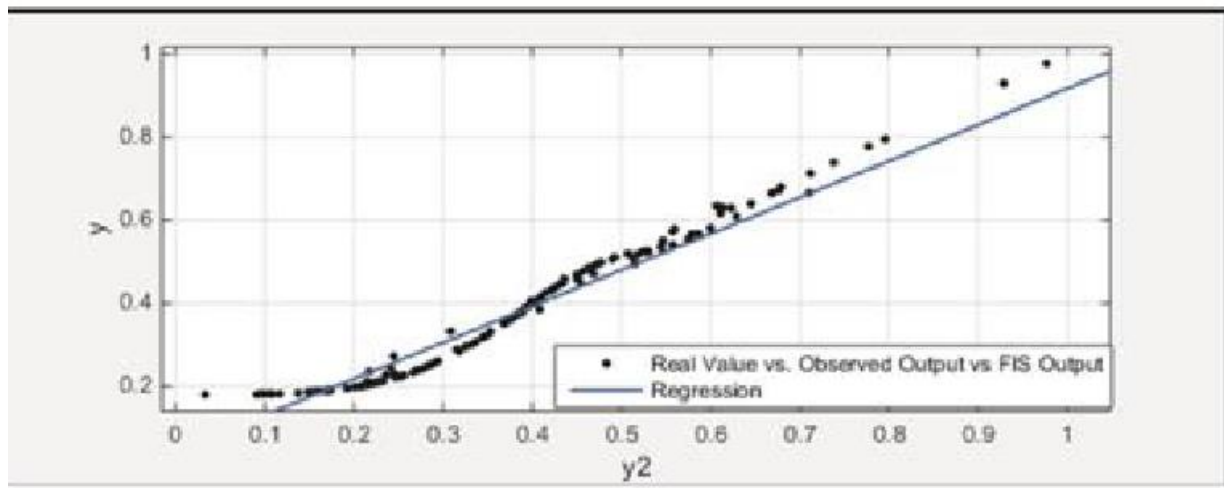


Рисунок 3.15. Порівняння результатів кредитного ризику ANFIS із реальними даними на етапі навчання.

Однак наступні діаграми показують, що вихідні значення були передбачені MATLAB на додаток до придатності реальних та прогнозованих виходів стосовно даних тестування.

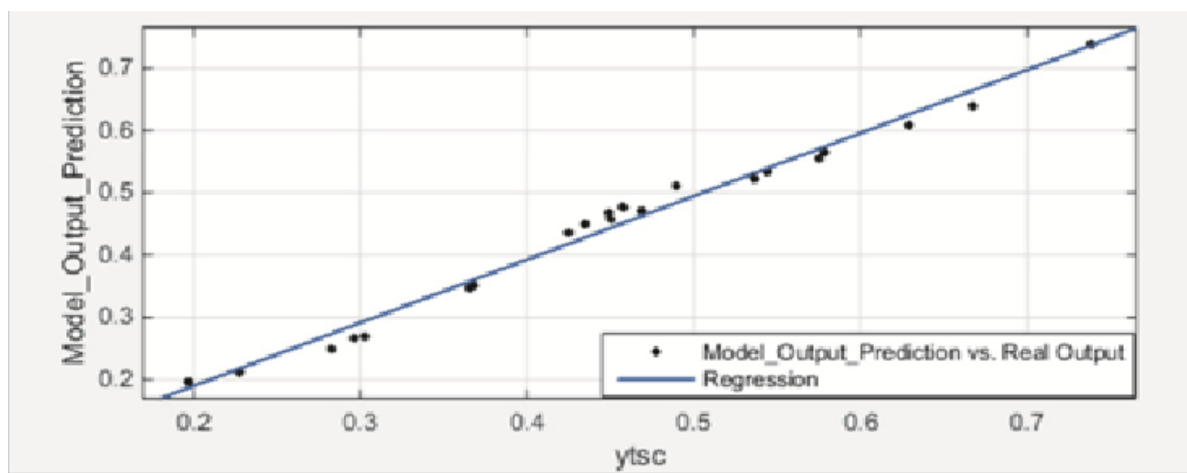


Рисунок 3.16. Порівняння результатів кредитного ризику з реальними даними ANFIS на етапі тестування.

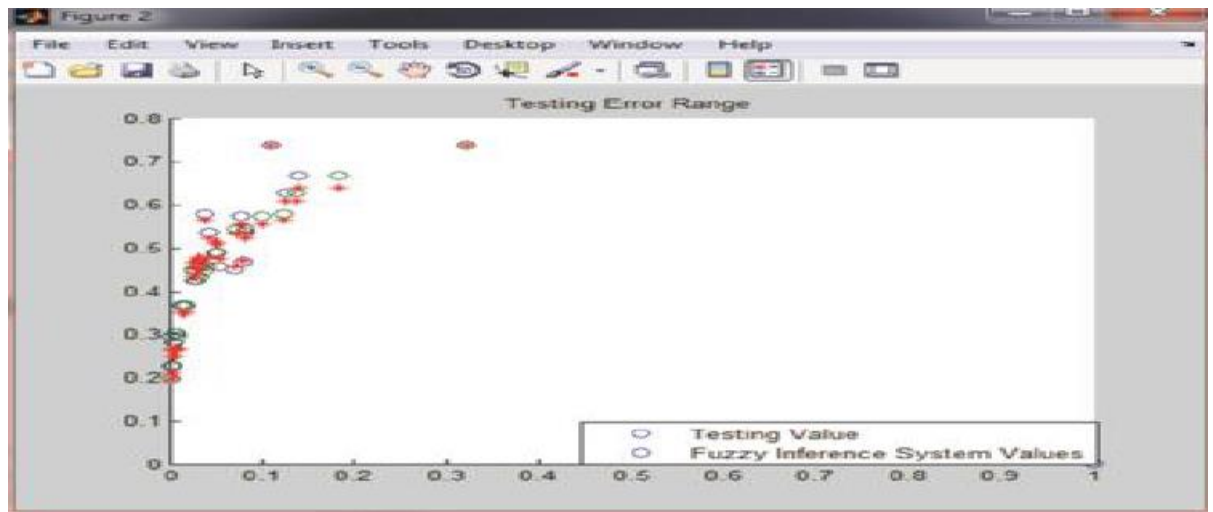


Рисунок 3.17. Відображення реальних значень та обчислених значень кредитного ризику моделі ANFIS для тестування даних.

Real Outputs	System Data
7.541863	7.300358194
8.0020411	7.671326058
5.8690365	6.124778497
8.8507744	8.854441146
5.6218553	5.642688195
5.0954197	5.227879389
6.5226524	6.401113842
6.8951929	6.65319125
6.9363679	6.78416766
3.6318102	3.223890057
2.3611756	2.353280617
3.3923974	2.983418511
2.7301778	2.529286792
5.213519	5.392508438
5.4902332	5.713954966
5.3981194	5.486986412
5.4845735	5.727758581
3.5514244	3.195259187

Рисунок 3.18. Порівняння результатів моделі оцінки кредитного ризику із реальними результатами.

Висновки показують, що застосування нейро-нечіткої моделі ANFIS є добре застосовним до оцінювання кредитного ризику з більш точними результатами.

ВИСНОВКИ

В результаті виконання магістерської роботи було удосконалення системи прийняття рішень для оптимізації баз даних на основі нейронних мереж з використанням засобів моделювання на основі нейро-нечіткої моделі до слабоструктурованих процесів.

При виконанні роботи було виконано наступні задачі.

1. Проаналізовано діючі алгоритми нечіткої логіки такі як модель Мамдані та Такагі-Сугено. Встановлено, що модель Мамдані дозволяє значно зменшити обсяги обчислень через агрегування передумов, що дозволяє зв'язати умови між собою та виявлення значення істинності всіх його підумов. В свою чергу Модель Такагі-Сугено зазвичай використовують у експертних системах знань, що передбачає спільне використання алгоритмів та евристик одночасно.

2. Досліджено сучасні проблеми слабоструктурованих процесів та встановлено, що нечітка логіка може допомогти в дослідженні суджень в умовах нечіткості, які схожі з судженнями у звичайному сенсі, та їх застосування у обчислювальних системах.

3. Проаналізовано існуючі архітектури нейронечітких мереж такі як GARICASN, NEFCON, EFuNN і dmEFuNN та ANFIS.

4. Розроблено методику прийняття рішень для оптимізації баз даних на основі нейронних мереж, з допомогою яких було виконано роботу по кредитного скорінгу, який допомагає банкам генерувати більше прибутків та зменшити ризики у разі дефолту.

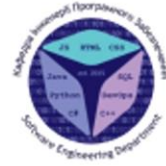
Перелік посилань

1. X. Duan, Y. Wang, W. Pedrycz, X. Liu, C. Wang, Z. Li, AFSNN: a classification algorithm using axiomatic fuzzy sets and neural networks, *IEEE Trans. Fuzzy Syst.* (2018).
2. J.E. Dayhoff, J.M. DeLeo, Artificial neural networks: opening the black box, *Cancer* 91 (S8) (2001) 1615–1635.
3. O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*, Springer Science & Business Media, 2013.
4. F. Da, Decentralized sliding mode adaptive controller design based on fuzzy neural networks for interconnected uncertain nonlinear systems, *IEEE Trans. Neural Netw.* 11 (6) (2000) 1471–1480.
5. J. Vieira, F. Dias, A. Mota, Neuro-fuzzy systems: a survey, in: *5th WSEAS NNA International Conference on Neural Networks and Applications*, Udine, Italia, 2004, pp. 1–6.
6. S. Kar, S. Das, P.K. Ghosh, Applications of neuro fuzzy systems: A brief review and future outline, *Appl. Soft Comput.* 15 (2014) 243–259.
7. S. Mitra, Y. Hayashi, Neuro-fuzzy rule generation: survey in soft computing framework, *IEEE Trans. Neural Netw.* 11 (3) (2000) 748–768.
8. B. Müller, M. Beer, *Fuzzy Randomness: Uncertainty in Civil Engineering and Computational Mechanics*, Springer Science & Business Media, 2013.
9. F. Zahedi, Z. Zahedi, A review of neuro-fuzzy systems based on intelligent control, 2018, arXiv preprint arXiv:1805.03138.
10. M. Knezevic, M. Cvetkovska, T. Hančk, L. Braganca, A. Soltesz, Artificial neural networks and fuzzy neural networks for solving civil engineering problems, *Complexity* 2018 (2018).

11. O.N. Sayaydeh, M.F. Mohammed, C.P. Lim, A survey of fuzzy min max neural networks for pattern classification: Variants and applications, *IEEE Trans. Fuzzy Syst.* (2018).
12. S. Mishra, S. Sahoo, B.K. Mishra, Neuro-fuzzy models and applications, in: *Emerging Trends and Applications in Cognitive Computing*, IGI Global, 2019, pp. 78–98. P.V. de Campos Souza / *Applied Soft Computing Journal* 92 (2020) 106275 19
13. K. Shihabudheen, G. Pillai, Recent advances in neuro-fuzzy system: A survey, *Knowl.-Based Syst.* 152 (2018) 136–162.
14. P. Chaves, T. Kojiri, Deriving reservoir operational strategies considering water quantity and quality objectives by stochastic fuzzy neural networks, *Adv. Water Resour.* 30 (5) (2007) 1329–1341.
15. Q.-Z. Zhang, W.-S. Gan, Y.-l. Zhou, Adaptive recurrent fuzzy neural networks for active noise control, *J. Sound Vib.* 296 (4–5) (2006) 935–948.
16. C.-H. Lu, C.-C. Tsai, Generalized predictive control using recurrent fuzzy neural networks for industrial processes, *J. Process Control* 17 (1) (2007) 83–92.
17. X. Deng, X. Wang, Incremental learning of dynamic fuzzy neural networks for accurate system modeling, *Fuzzy Sets and Systems* 160 (7) (2009) 972–987.
18. L. Yu, Y.-Q. Zhang, Evolutionary fuzzy neural networks for hybrid financial prediction, *IEEE Trans. Syst. Man Cybern. C* 35 (2) (2005) 244–249.
19. R.-J. Li, Z.-B. Xiong, Forecasting stock market with fuzzy neural networks, in: *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, Vol. 6, IEEE, 2005, pp. 3475–3479.
20. S. Zhao, G. Xu, T. Tao, L. Liang, Real-coded chaotic quantum-inspired genetic algorithm for training of fuzzy neural networks, *Comput. Math*



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ



Кафедра інженерії програмного забезпечення

МАГІСТЕРСЬКА РОБОТА
«РОЗРОБКА СИСТЕМИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ ОПТИМІЗАЦІЇ
БАЗ ДАНИХ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ

Виконав: студент групи ПДМ – 61, Копаєв Данііл Олександрович

Керівник: , к.т.н., доц. кафедри ІПЗ, Негоденко Олена Василівна

Київ - 2021

МЕТА, ОБ'ЄКТА ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

2

Мета роботи підвищення ефективності прийняття рішень до бази даних з використанням нейронних мереж та нечіткої логіки.

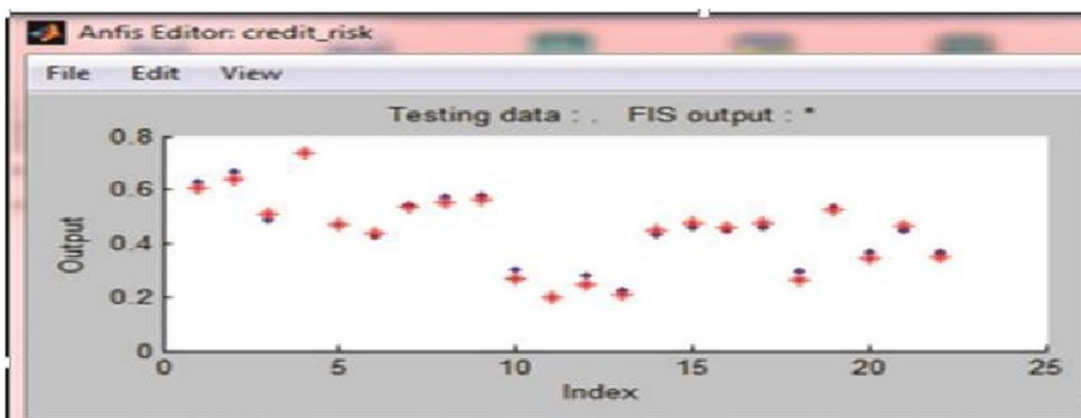
Об'єкт дослідження: оптимізація системи прийняття рішень до бази даних на основі нейронних мереж та нечіткої логіки.

Предмет дослідження: методи та алгоритми для розробки системи прийняття рішень на основі нейронних мереж та нечіткої логіки.

АКТУАЛЬНІСТЬ РОБОТИ

3

Актуальність роботи полягає в розробці комплексної методики роботи нейро-нечітких мереж з базами даних. За допомогою розробленої методики можна буде виконувати роботу по кредитного скорінгу, який допомагає банкам генерувати більше прибутків та зменшує ризики у разі дефолту.



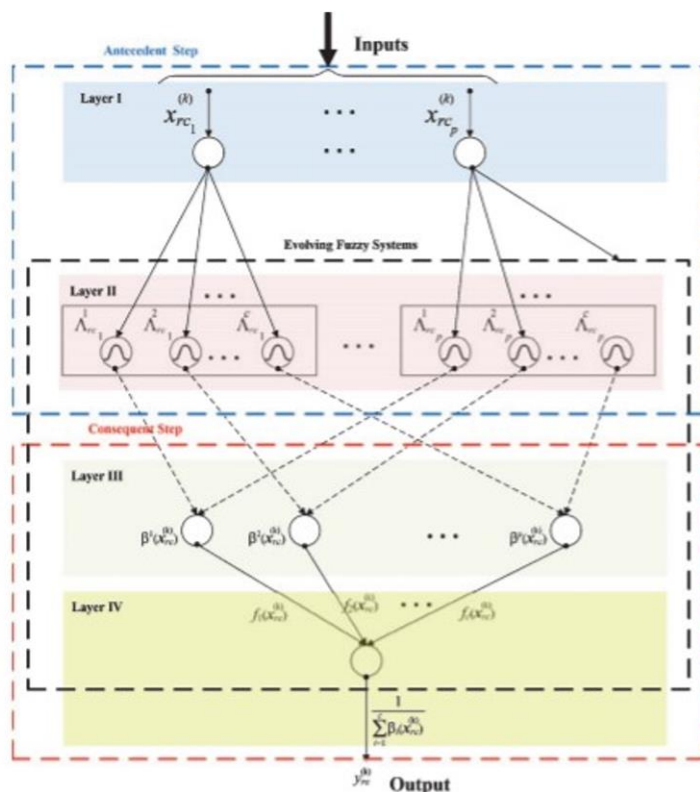
АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ.

4

НЕЙРОННІ МЕРЕЖІ	НЕЧІТКА ЛОГІКА	НЕЙРОНЕЧІТКІ МЕРЕЖІ
(+) широко застосовуються для ідентифікації та розпізнавання інформації;	(+) можливість нечіткої формалізації критеріїв оцінки і порівняння;	(+) сигнал, що плавно змінюється, на виході;
(+) можливість додаткового навчання у процесі роботи;	(+) можливість оперувати вхідними даними, заданими нечітко;	(+) точне налаштування функцій приналежності, можливість донавчання протягом роботи;
(-) не можна прискорити навчання виходячи з деякої інформації;	(-) неможливість внесення додаткової інформації у процесі роботи;	(+) можливість втручатися в внутрішню структуру мережі
(-) внутрішня структура мережі – «чорна скринька»;	(-) обмежена складність внутрішньої структури;	(+) можливість оперувати евристичними даними для донавчання.

УЗАГАЛЬНЕНИЙ АЛГОРИТМ ПРОВЕДЕННЯ МОДЕЛЮВАННЯ

5



ІНФОРМАЦІЯ ДО НАВЧАННЯ

6

Таблиця 1.1 Вік позичальника

Вік	Число	Процентне співвідношення
< 30 років	23	21,29%
30 років <= and < 60 років	81	75%
>= 60 років	4	3,71%
Усього	108	100%

Таблиця 1.2 Стаж роботи

Стаж роботи	Число	Процентне співвідношення
< 2 роки	6	5,55%
2 роки <= and < 10 років	64	59,26%
>= 10 років	38	35,19%
Усього	108	100%

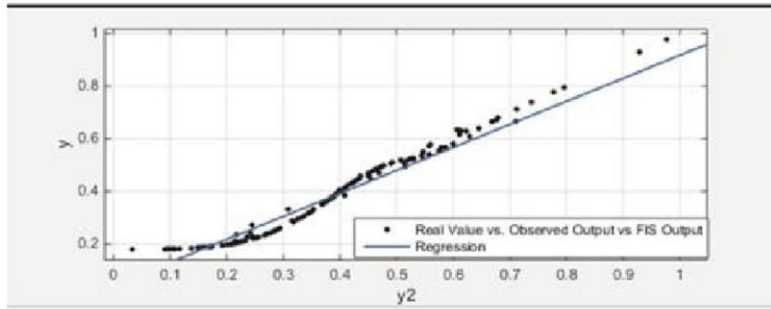
Таблиця 1.3 Здатність погашення

Здатність погашення	Число	Процентне співвідношення
< 200 чоловік	7	6,5%
200 <= and < 400 людей	55	50,9%
>= 400 чоловік	46	42,6%
Усього	108	100%

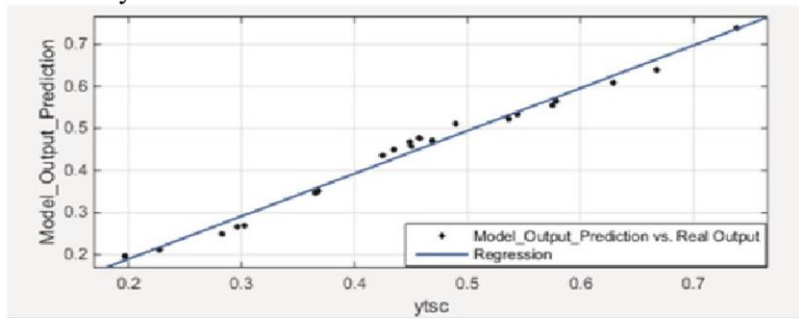
РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ

7

Порівняння результатів кредитного ризику ANFIS із реальними даними на етапі навчання.



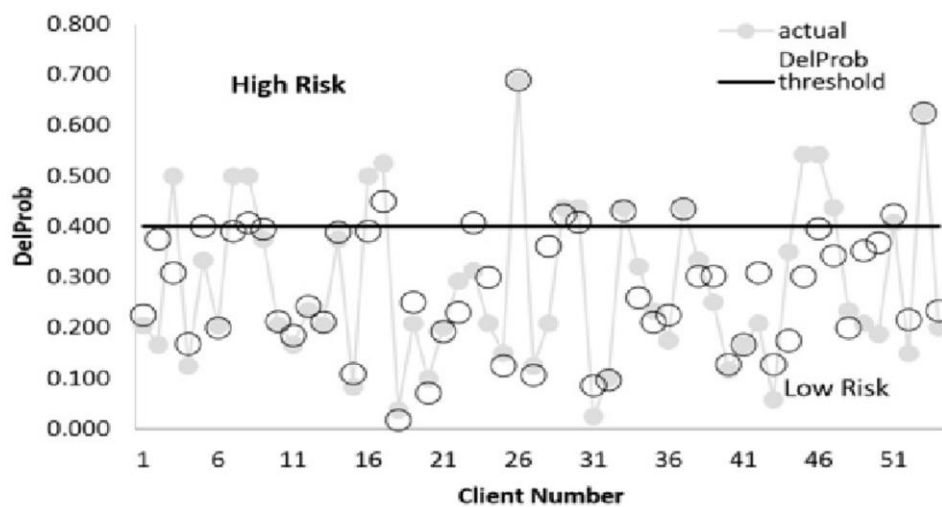
Порівняння результатів кредитного ризику з реальними даними ANFIS на етапі тестування.



ПРАКТИЧНІ РЕЗУЛЬТАТИ

8

Результати ANFIS для кредитного скорінгу.



ВИСНОВКИ

9

1. Проаналізовано діючі алгоритми нечіткої логіки з яких обрано модель ANFIS. Встановлено, що модель ANFIS дозволяє значно зменшити обсяги обчислень через агрегування передумов, що дозволяє зв'язати умови між собою та виявлення значення істинності всіх його підумов. Також встановлено що Модель ANFIS зазвичай використовують у експертних системах знань, що передбачає спільне використання алгоритмів та евристик одночасно.

2. Досліджено сучасні проблеми слабоструктурованих процесів та встановлено, що нечітка логіка може допомогти в дослідженні суджень в умовах нечіткості, які схожі з судженнями у звичайному сенсі, та їх застосування у обчислювальних системах.

3. Проаналізовано існуючі архітектури нейронечітких мереж такі як GARIC ASN, NEFCON, EFuNN і dmEFuNN та ANFIS.

4. Розроблено методику прийняття рішень для оптимізації баз даних на основі нейронних мереж, з допомогою яких було виконано роботу по кредитного скорінгу, який допомагає банкам генерувати більше прибутків та зменшити ризики у разі дефолту.

ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ

10

СЕРТИФІКАТ

про публікацію в міжнародному науковому журналі

«ВІСНИК НАУКИ» № 37 | ISSN 0236-4115

стаття (тези)

«ЗАСТОСУВАННЯ ANFIS

В НЕЙРО-НЕЧІТКОМУ МОДЕЛЮВАННІ»

володар сертифіката

Конаєв Даніїл Олександрович



Інтернет-адреса журналу
world-conf.com/journal
Адреса електронної пошти:
office@world-conf.com



Рисоювач

Головний редактор
Рассказова Л.Ф.

Одеса, Україна | Рига, Латвія
26 грудня 2021 р.

ДЯКУЮ ЗА УВАГУ!