

Міністерство освіти і науки України  
Запорізький національний технічний університет

**МЕТОДИЧНІ ВКАЗІВКИ**  
до виконання лабораторних робіт  
з дисципліни  
**«ПРОЕКТУВАННЯ**  
**НЕЙРОМЕРЕЖ»**  
для студентів напряму підготовки  
6.050101 "Комп'ютерні науки"  
усіх форм навчання

2013

Методичні вказівки до виконання лабораторних робіт з дисципліни "Проектування нейромереж" для студентів напряму підготовки 6.050101 "Комп'ютерні науки" усіх форм навчання / Уклад.: С.О. Субботін, Є.М. Федорченко. – Запоріжжя: ЗНТУ, 2013. – 60 с.

Автори:                   Сергій Олександрович Субботін,  
професор кафедри програмних засобів,  
кандидат технічних наук, доцент;

Євген Миколайович Федорченко, асистент

Рецензент:           А.О. Олійник, кандидат технічних наук

Відповідальний  
за випуск:           В.І. Дубровін, зав. каф. програмних засобів

Затверджено  
на засіданні кафедри про-  
грамних засобів

Протокол № 11  
від 4.06.2013 р.

## ЗМІСТ

|                                                                                         |    |
|-----------------------------------------------------------------------------------------|----|
| Загальні положення . . . . .                                                            | 4  |
| 1. Лабораторна робота № 1. Нейромережі прямого<br>поширення . . . . .                   | 5  |
| 2. Лабораторна робота № 2. Нейромережі зі зворотними<br>зв'язками. . . . .              | 12 |
| 3. Лабораторна робота № 3. Нейромережі з латеральними<br>зв'язками. . . . .             | 16 |
| Література. . . . .                                                                     | 20 |
| Додаток А. Моделювання нейронних мереж у пакеті<br>MATLAB . . . . .                     | 24 |
| Додаток Б. Моделювання нейронних мереж у пакеті Statistica<br>Neural Networks . . . . . | 37 |
| Додаток В. Задачі для побудови нейромоделей. . . . .                                    | 48 |

## ЗАГАЛЬНІ ПОЛОЖЕННЯ

Дане видання призначене для вивчення та практичного освоєння студентами усіх форм навчання основ нейроінформатики та еволюційних алгоритмів.

Відповідно до графіка студенти перед виконанням лабораторної або самостійної роботи повинні ознайомитися з конспектом лекцій та рекомендованою літературою.

Для одержання заліку по кожній роботі студент здає викладачу цілком оформлений звіт, а також 3,5-дюймову дискету у форматі MS – DOS / Windows, перевірену на відсутність вірусів, з текстами розроблених програм, файлами програм, що виконуються, файлами даних і текстом звіту.

Звіт (приклад оформлення - див. додаток А) має містити:

- титульний аркуш (на ньому вказують назву міністерства, назву університету, назву кафедри, номер, вид і тему роботи, виконавця та особу, що приймає звіт, рік);

- мету, варіант і завдання роботи;
- лаконічний опис теоретичних відомостей;
- текст програми, що обов'язково містить коментарі;
- вхідні та вихідні дані програми;
- змістовний аналіз отриманих результатів та висновки.

Звіт виконують на білому папері формату А4 (210 x 297 мм). Текст розміщують тільки з однієї сторони листа. Поля сторінки з усіх боків – 20 мм. Аркуші скріплюють за допомогою канцелярських скріпок. Для набору тексту звіту використовують редактор MS Word 97: шрифт Times New Roman, 12 пунктів. Міжрядковий інтервал: полуторний – для тексту звіту, одинарний – для листингів програм, таблиць і роздруків даних.

Під час співбесіди студент повинний виявити знання про мету роботи, по теоретичному матеріалу, про методи виконання кожного етапу роботи, по змісту основних розділів розробленого звіту з демонстрацією результатів на конкретних прикладах. Студент повинний вміти правильно аналізувати отримані результати. Для самоперевірки при підготовці до виконання і здачі роботи студент повинний відповісти на контрольні питання, приведені наприкінці опису відповідної роботи. Загальний залік студент одержує після виконання і здачі останньої роботи.

## 1 Лабораторна робота №1

### НЕЙРОМЕРЕЖІ ПРЯМОГО ПОШИРЕННЯ

**Мета роботи:** Вивчити архітектури формального нейрона, багат шарового персептрона і радіально-базисної мережі, а також методи їхнього навчання; ознайомитися з програмними продуктами, що моделюють персептрони та радіально-базисні мережі.

#### Завдання до роботи

1. Ознайомитися з конспектом лекцій та рекомендованою літературою.

2. Вивчити архітектури формального нейрона, багат шарового персептрона та радіально-базисної мережі, а також методи їхнього навчання.

3. Використовуючи документацію рекомендованого викладачем програмного засобу, вивчити його архітектуру і компоненти, призначені для моделювання і навчання нейромереж прямого поширення, введення і виведення даних, підготовки звітів (виведення і відображення результатів роботи).

4. Визначити за табл. 1.1 згідно з номером варіанта студента (обирається за журналом) номер підрозділу Додатку В, що містить опис прикладної задачі, та номери контрольних питань, відповіді на які належить навести у звіті. Визначити за табл. 1.2 кількості шарів та нейронів у шарах для побудови багат шарових персептронів, а також функцію активації для завдання 6.

5. Використовуючи пакети MATLAB та Statistica Neural Networks для відповідних варіанту студента вхідних даних вирішити прикладну задачу на основі одношарового та багат шарового персептронів, використовуючи різні методи навчання та різні функції активації. Вирішити ту саму задачу на основі радіально-базисної мережі.

6. Змінюючи значення кроку навчання, для одношарового та багат шарового персептронів з заданою функцією активації, що відповідає номеру варіанту студента (див. табл. 1.2) дослідити, як впливає величина кроку навчання на час навчання. Побудувати графіки залежності часу навчання персептронів від величини кроку навчання.

7. Зберегти у файлі на диску початкові параметри нейромоделей, результати їхнього навчання (матрицю ваг та структуру з зазначенням функцій активації і кількості нейроелементів у шарах) та ро-

боти для навчальної та контрольної вибірок (помилку класифікації (оцінювання), значення на входах і виході, час навчання, час класифікації).

8. Результати виконання пп. 5 - 7 занести в таблицю, стовпці якої повинні мати назви: назва архітектури нейромережі, кількість шарів, кількість нейронів у шарах, функції активації нейронів у шарах, метод навчання, час навчання, час класифікації (оцінювання) навченої мережі для навчальної вибірки, помилка класифікації (оцінювання) для навчальної вибірки, час класифікації (оцінювання) навченої мережі для тестової вибірки, помилка класифікації (оцінювання) для тестової вибірки.

Таблиця 1.1 - Практичні задачі та контрольні питання для варіантів

| Номер варіанту | Задача | Номер контрольного питання |    |    |    |    |
|----------------|--------|----------------------------|----|----|----|----|
| 1              | B.1    | 1                          | 11 | 21 | 31 | 41 |
| 2              | B.2    | 2                          | 12 | 22 | 32 | 42 |
| 3              | B.3    | 3                          | 13 | 23 | 33 | 43 |
| 4              | B.4    | 4                          | 14 | 24 | 34 | 44 |
| 5              | B.5    | 5                          | 15 | 25 | 35 | 45 |
| 6              | B.6    | 6                          | 16 | 26 | 36 | 46 |
| 7              | B.7    | 7                          | 17 | 27 | 37 | 47 |
| 8              | B.8    | 8                          | 18 | 28 | 38 | 48 |
| 9              | B.9    | 9                          | 19 | 29 | 39 | 49 |
| 10             | B.10   | 10                         | 20 | 30 | 40 | 50 |
| 11             | B.10   | 19                         | 11 | 21 | 38 | 41 |
| 12             | B.9    | 10                         | 20 | 22 | 40 | 42 |
| 13             | B.8    | 9                          | 1  | 23 | 39 | 43 |
| 14             | B.7    | 8                          | 18 | 24 | 31 | 44 |
| 15             | B.6    | 7                          | 17 | 25 | 37 | 45 |
| 16             | B.5    | 6                          | 16 | 26 | 36 | 46 |
| 17             | B.4    | 5                          | 15 | 27 | 35 | 47 |
| 18             | B.3    | 4                          | 14 | 28 | 34 | 48 |
| 19             | B.2    | 3                          | 13 | 29 | 33 | 49 |
| 20             | B.1    | 2                          | 12 | 30 | 32 | 50 |
| 21             | B.5    | 1                          | 11 | 20 | 29 | 40 |
| 22             | B.6    | 41                         | 2  | 12 | 21 | 30 |
| 23             | B.7    | 39                         | 31 | 3  | 13 | 22 |
| 24             | B.8    | 42                         | 50 | 32 | 4  | 14 |
| 25             | B.9    | 43                         | 49 | 33 | 23 | 5  |
| 26             | B.1    | 44                         | 34 | 24 | 6  | 15 |
| 27             | B.2    | 35                         | 25 | 7  | 16 | 45 |
| 28             | B.3    | 26                         | 8  | 17 | 36 | 46 |
| 29             | B.4    | 9                          | 18 | 27 | 37 | 47 |
| 30             | B.10   | 19                         | 10 | 28 | 38 | 48 |

Таблиця 1.2 - Параметри нейромереж для варіантів

| Номер<br>варіанта | Багатoshаровий персептрон |                               | назва функції активації |
|-------------------|---------------------------|-------------------------------|-------------------------|
|                   | Кількість<br>шарів        | Кількості нейронів у<br>шарах |                         |
| 1                 | 2                         | 3-1                           | логістична сигмоїдна    |
| 2                 | 3                         | 3-3-1                         | тангенційна сигмоїдна   |
| 3                 | 4                         | 3-3-3-1                       | логістична сигмоїдна    |
| 4                 | 2                         | 2-1                           | тангенційна сигмоїдна   |
| 5                 | 3                         | 2-2-1                         | логістична сигмоїдна    |
| 6                 | 4                         | 2-2-2-1                       | тангенційна сигмоїдна   |
| 7                 | 2                         | 5-1                           | логістична сигмоїдна    |
| 8                 | 3                         | 5-5-1                         | тангенційна сигмоїдна   |
| 9                 | 4                         | 5-5-5-1                       | логістична сигмоїдна    |
| 10                | 2                         | 4-1                           | тангенційна сигмоїдна   |
| 11                | 3                         | 4-4-1                         | логістична сигмоїдна    |
| 12                | 4                         | 4-4-4-1                       | тангенційна сигмоїдна   |
| 13                | 2                         | 6-1                           | логістична сигмоїдна    |
| 14                | 3                         | 6-3-1                         | тангенційна сигмоїдна   |
| 15                | 4                         | 6-4-3-1                       | логістична сигмоїдна    |
| 16                | 2                         | 7-1                           | тангенційна сигмоїдна   |
| 17                | 3                         | 7-7-1                         | логістична сигмоїдна    |
| 18                | 4                         | 7-3-4-1                       | тангенційна сигмоїдна   |
| 19                | 2                         | 8-1                           | логістична сигмоїдна    |
| 20                | 3                         | 8-8-1                         | тангенційна сигмоїдна   |
| 21                | 4                         | 8-7-6-1                       | логістична сигмоїдна    |
| 22                | 2                         | 2-1                           | тангенційна сигмоїдна   |
| 23                | 3                         | 2-3-1                         | логістична сигмоїдна    |
| 24                | 4                         | 2-4-4-1                       | тангенційна сигмоїдна   |
| 25                | 2                         | 3-1                           | логістична сигмоїдна    |
| 26                | 3                         | 6-2-1                         | тангенційна сигмоїдна   |
| 27                | 4                         | 7-6-3-1                       | логістична сигмоїдна    |
| 28                | 2                         | 4-1                           | тангенційна сигмоїдна   |
| 29                | 3                         | 9-2-1                         | логістична сигмоїдна    |
| 30                | 4                         | 9-3-2-1                       | тангенційна сигмоїдна   |

9. Проаналізувати отримані результати і зробити висновки про те, як впливає вид функції активації формального нейрона на час навчання і час класифікації (оцінювання), а також величину помилки навчання і класифікації (оцінювання) одношарового персептрона; як впливають вид функції активації і вид коригувального правила ваг (метод навчання) на час навчання і класифікації (оцінювання), а також величину помилки навчання та класифікації багатoshарового персептрона; яка модель нейромереж прямого поширення краще підходить для вирішення задачі, що відповідає варіанту.

10. Порівняти одношаровий та багатошаровий персептрони, радіально-базисну мережу та багатошаровий персептрон за швидкістю навчання і точністю класифікації; принципами побудови архітектури.

11. Відповісти на контрольні питання.

12. Оформити звіт з роботи.

### **Зміст звіту**

1. Тема та мета роботи.

2. Номер варіанту та завдання до роботи, що містить стислий опис вирішуваної практичної задачі (не більше 0,5 сторінки).

3. Короткі теоретичні відомості (не більше 2 сторінок), що містять стислий опис архітектур формального нейрона, багатошарового персептрона і радіально-базисної мережі та методів їхнього навчання..

4. Опис процесу виконання роботи (не більше 3 сторінок).

5. Тексти програм, розроблених (модифікованих) студентом.

6. Вхідні дані та результати роботи програм, узагальнюючі таблиці, графіки.

7. Висновки, що містять відповіді на контрольні запитання (див. табл. 1.1), а також відображують результати виконання роботи та їх критичний аналіз.

### **Контрольні питання**

1. Поняття: нейрон, нейромережа, нейрокомп'ютер, нейроінформатика.

2. Класифікація та види моделей нейромереж.

3. Властивості штучних нейромереж.

4. Загальне уявлення про навчання нейромереж.

5. Загальна характеристика та принципи побудови нейромереж.

6. Характеристики процесу навчання нейромереж.

7. Лінійна роздільність і лінійна нероздільність класів.

8. Порівняння моделей та алгоритмів навчання нейромереж прямого поширення.

9. Нейронні мережі у пакеті MATLAB. Модуль Neural Network Toolbox.

10. Пакет Statistica Neural Networks.

11. Моделі нейроелементів у пакеті MATLAB.

12. Біологічні нейрони. Математичні моделі нейроелементів.

13.Поняття: формальний нейрон, синапс, вага (ваговий коефіцієнт), поріг, дискримінантна (вагова) функція, функція активації, бажаний і реальний вихід нейромережі, навчання нейромережі, класифікація, апроксимація, оцінювання, помилка навчання / класифікації, час навчання / класифікації, цільова функція навчання.

14.Метод найменших квадратів як основа алгоритму Уідроу-Хоффа. Чи завжди збігаються алгоритми навчання одношарового персептрона?

15.Можливості і властивості одношарових персептронів. Які задачі можна вирішувати на основі одношарових персептронів, а які не можна? Чи доцільно застосовувати одношаровий персептрон для класифікації складно (нелінійно) роздільних образів? Обґрунтуйте і доведіть відповідь. Приведіть приклади.

16.Чи впливає величина кроку навчання на час навчання одношарового персептрона, багатошарової мережі? Відповідь обґрунтуйте теоретично та доведіть експериментально.

17.Які функції активації нейронів найчастіше використовують і чому? Чи впливає вид функції активації нейрона на тривалість навчання і роботи персептрона, величину помилки навчання і класифікації (оцінювання) і, якщо впливає, то яким чином?

18.Чи впливає кількість використаних ознак на швидкість навчання персептрона? Відповідь обґрунтуйте теоретично та доведіть експериментально.

19.У чому подібність і відмінність біологічних і формальних нейронів?

20.Чи можна навчити дискретний одношаровий персептрон обчислювати значення функцій:  $y = (x_1 \text{ and } x_2)$ ,  $y = (x_1 \text{ xor } x_2)$ ,  $y = \text{not}((\text{not } x_1) \text{ and } x_2)$ , а дійсний одношаровий персептрон – обчислювати значення функцій:  $y = 3x_1 - 0,05x_2$ ,  $y = \sin(x_1) + 0,3x_2$ ,  $y = 0,5x_1 + 2x_2 - 2,5(x_1 - x_2) + 5,5$ ? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

21.Багатошаровий персептрон: модель і принципи побудови архітектури.

22.Які задачі можна вирішувати на основі багатошарових персептронів, а які не можна? Обґрунтуйте і доведіть відповідь. Приведіть приклади.

23. Алгоритм зворотного поширення помилки. Чи завжди збігається алгоритм зворотного поширення помилки для багатошарового персептрона?

24. Які методи навчання багатошарових нейромереж Вам відомі? Дайте порівняльну характеристику відомих Вам методів навчання багатошарових нейромереж.

25. Градієнтні алгоритми навчання багатошарових нейромереж. Який з градієнтних методів навчання на основі зворотного поширення помилки Ви рекомендували б для застосування, якщо критерієм оптимальності навчання НМ є: а) швидкість навчання, б) простота обчислень, в) універсальність?

26. Чи може дійсний багатошаровий персептрон моделювати функцію  $y = x_1^2 + 0.5x_1 + x_2$ , якщо число його шарів дорівнює: а) 1, б) 2, в) 3? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

27. Чи можливе використання неградієнтних методів багатовимірної безумовної оптимізації для настроювання ваг багатошарових нейромереж, і, якщо можливе, то наскільки це доцільно робити?

28. Чи можна навчити дискретний багатошаровий персептрон обчислювати значення функцій: 1)  $y = x_1$  and  $x_2$ , 2)  $y = x_1 \text{ xor } x_2$ , 3)  $y = \text{not}((\text{not } x_1) \text{ and } x_2)$ , а дійсний багатошаровий персептрон – обчислювати значення функцій: 1)  $y = 3x_1 - 0.05x_2$ , 2)  $y = \sin(x_1) + 0.3x_2$ , 3)  $y = 0.5x_1 + 2x_2 - 2.5(x_1 - x_2) + 5.5$ , 4)  $y = x_1 / (x_2 \sin(\pi))$ ? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

29. Опишіть суть узагальненого градієнтного алгоритму. Окремі випадки узагальненого градієнтного алгоритму.

30. Нейронні мережі прямого поширення та градієнтні алгоритми навчання у пакеті MATLAB.

31. Моделі та принципи синтезу архітектури радіально-базисних нейромереж.

32. У чому подібність і відмінність радіально-базисних нейромереж і багатошарових нейромереж?

33. Застосування кластер-аналізу при навчанні радіально-базисних нейромереж.

34. Методи навчання радіально-базисних нейромереж.

35. Чи завжди збігаються алгоритми навчання радіально-базисних нейромереж?

36. Які задачі можна вирішувати на основі радіально-базисних нейромереж, а які не можна? Обґрунтуйте і доведіть відповідь. Наведіть приклади.

37. Радіально-базисні нейромережі у пакеті MATLAB.

38. Чи доцільно застосовувати радіально-базисних нейромереж для класифікації складно (нелінійно) роздільних образів?

39. Чи можна навчити радіально-базисну нейромережу обчислювати значення функцій: 1)  $y = x_1$  and  $x_2$ , 2)  $y = x_1 \text{ xor } x_2$ , 3)  $y = \text{not}((\text{not } x_1) \text{ and } x_2)$ , 4)  $y = 3x_1 - 0,05x_2$ , 5)  $y = \sin(x_1) + 0,3x_2$ , 6)  $y = 0,5x_1 + 2x_2 - 2,5(x_1 - x_2) + 5,5$ , 7)  $y = x_1 / (x_2 \sin(\pi))$ ? Відповіді обґрунтуйте і, якщо можливо, поясніть рисунками.

40. Переваги і недоліки радіально-базисних мереж.

41. Що таке генеральна сукупність, вибірка, екземпляр, ознака?

42. Вимоги до навчальних вибірок даних.

43. Що таке репрезентативна вибірка даних?

44. Чи повинна навчальна вибірка бути репрезентативною?

45. Чи повинна тестова вибірка бути репрезентативною?

46. Чи впливає обсяг навчальної вибірки на швидкість навчання нейромереж?

47. Чи впливає репрезентативність навчальної вибірки на точність класифікації екземплярів тестової вибірки?

48. Чи впливає репрезентативність тестової вибірки на точність класифікації екземплярів тестової вибірки?

49. Чи впливає репрезентативність тестової вибірки на точність навчання персептрона по навчальній вибірці?

50. Чи залежить якість навчання нейромереж від якості та обсягу навчальної вибірки?

## **Лабораторна робота № 2**

### **НЕЙРОМЕРЕЖІ ЗІ ЗВОРОТНИМИ ЗВ'ЯЗКАМИ**

**Мета роботи** – вивчити моделі і методи навчання нейромереж зі зворотними зв'язками, розглянути приклади їхнього практичного використання; порівняти їхні можливості з можливостями нейромереж прямого поширення; ознайомитися зі стандартними програмними засобами для моделювання мереж зі зворотними зв'язками.

#### **Завдання до роботи**

1. Ознайомитися з конспектом лекцій та рекомендованою літературою.

2. Вивчити архітектури нейромереж Хопфілда та Ельмана, а також методи їхнього навчання.

3. Використовуючи документацію рекомендованого викладачем програмного засобу, вивчити його архітектуру і компоненти, призначені для моделювання і навчання нейромереж зі зворотними зв'язками, введення і виведення даних, підготовки звітів (виведення і відображення результатів роботи).

4. Визначити за табл. 1.1 згідно з номером варіанта студента (обирається за журналом) номер підрозділу Додатку В, що містить опис прикладної задачі, та номери контрольних питань до даної роботи, відповіді на які належить навести у звіті.

5. Використовуючи пакет MATLAB для відповідних варіанту студента вхідних даних вирішити прикладну задачу на основі нейромереж Хопфілда та Ельмана.

6. Зберегти у файлі на диску початкові параметри нейромоделей, результати їхнього навчання (матрицю ваг та структуру з зазначенням функцій активації і кількості нейроелементів у шарах) та роботи для навчальної та контрольної вибірок (помилку класифікації (оцінювання), значення на входах і виході, час навчання, час класифікації).

7. Результати виконання пп. 5 - 6 занести в таблицю, стовпці якої повинні мати назви: назва архітектури нейромережі, кількість шарів, кількість нейронів у шарах, функції активації нейронів у шарах, метод навчання, час навчання, час класифікації навченої мережі для навчальної вибірки, помилка класифікації для навчальної вибірки, час

класифікації навченої мережі для тестової вибірки, помилка класифікації для тестової вибірки.

8. Проаналізувати отримані результати і дати порівняльну характеристику мереж Хопфілда та Ельмана. Дати порівняльну характеристику мереж зі зворотними зв'язками та мереж прямого поширення за швидкістю навчання і точністю класифікації; принципами побудови архітектури.

9. Відповісти на контрольні питання.

10. Оформити звіт з роботи.

### **Зміст звіту**

1. Тема та мета роботи.

2. Номер варіанту та завдання до роботи, що містить стислий опис вирішуваної практичної задачі (не більше 0,5 сторінки).

3. Короткі теоретичні відомості (не більше 2 сторінок), що містять стислий опис архітектур та методів навчання нейромереж зі зворотними зв'язками.

4. Опис процесу виконання роботи (не більше 3 сторінок).

5. Тексти програм, розроблених (модифікованих) студентом.

6. Вхідні дані та результати роботи програм, узагальнюючі таблиці, графіки.

7. Висновки, що містять відповіді на контрольні запитання до даної роботи (див. табл. 1.1), а також відображають результати виконання роботи та їх критичний аналіз.

### **Контрольні питання**

1. Дайте визначення понять: повнозв'язна мережа Хопфілда, функція обчислювальної енергії, асоціативна пам'ять, псевдоінверсне навчальне правило, проективний алгоритм настроювання ваг, ефект рознасичення (ефект Городничего), еталон.

2. Бінарні повнозв'язні нейромережі Хопфілда.

3. Псевдоінверсне навчальне правило, проективний алгоритм настроювання ваг.

4. Чи дозволяє модель і традиційні методи навчання мереж Хопфілда побудувати на її основі асоціативний запам'ятовуючий

пристрій, здатний запам'ятовувати стільки образів, скільки нейронів у мережі? Відповідь поясніть.

5. Які задачі можна вирішувати на основі бінарних НМ Хопфілда, а які не можна? Обґрунтуйте і доведіть відповідь. Приведіть приклади.

6. Чи можна навчити бінарну НМ Хопфілда відновлювати за ключем 1101\*\*\*\* запам'ятовані набори кодових слів: а) 11011001, 01010010, 11001111, б) 11011001, 11011010, 01011111, в) 11011001, 11010010, 11011111? Відповідь поясніть.

7. Чи доцільно застосовувати бінарні мережі Хопфілда для класифікації складно (нелінійно) роздільних образів?

8. Чи завжди збігаються проєкційні алгоритми навчання мережі Хопфілда?

9. Опишіть ефект Городничего і розгляньте перспективи його використання.

10. Метод рознасичення синаптичної матриці мережі Хопфілда.

11. Нейромережа Ельмана.

12. Машина Больцмана.

13. Чи доцільно застосовувати мережі Хопфілда при вирішенні задач, для рішення яких може використовуватися одношаровий перцептрон?

14. Застосування нейромереж для асоціативного пошуку інформації.

15. Мережі Хопфілда у задачах комбінаторної оптимізації.

16. Порівняння мереж Хопфілда та Ельмана

17. Нейронні мережі Хопфілда та Ельмана у пакеті MATLAB.

18. Порівняйте можливості бінарних мереж Хопфілда і дискретного одношарового перцептрона.

19. Порівняйте можливості бінарних мереж Хопфілда і багатошарового перцептрона.

20. Порівняйте можливості бінарних мереж Хопфілда і радіально-базисних мереж.

21. Вимоги до навчальних вибірок даних.

22. Чи повинна навчальна вибірка бути репрезентативною?

23. Чи повинна тестова вибірка бути репрезентативною?

24. Чи впливає обсяг навчальної вибірки на швидкість навчання нейромереж?

25. Чи впливає репрезентативність навчальної вибірки на точність класифікації екземплярів тестової вибірки?

26. Чи впливає репрезентативність тестової вибірки на точність класифікації екземплярів тестової вибірки?

27. Чи впливає репрезентативність тестової вибірки на точність навчання мережі по навчальній вибірці?

28. Чи залежить якість навчання нейромереж від якості та обсягу навчальної вибірки?

29. Що таке генеральна сукупність, вибірка, екземпляр, ознака?

30. Що таке репрезентативна вибірка даних?

31. Характеристики процесу навчання нейромереж.

32. Порівняння моделей та алгоритмів навчання нейромереж прямого поширення.

33. Поняття: нейрон, нейромережа, нейрокомп'ютер, нейроінформатика.

34. Поняття: формальний нейрон, синапс, вага (ваговий коефіцієнт), поріг, дискримінантна (вагова) функція, функція активації.

35. Поняття: бажаний і реальний вихід нейромережі, навчання нейромережі, класифікація, апроксимація, оцінювання, помилка навчання / класифікації, час навчання / класифікації, цільова функція навчання.

36. Нейронні мережі у пакеті MATLAB. Модуль Neural Network Toolbox.

37. Моделі нейроелементів у пакеті MATLAB.

38. Лінійна роздільність і лінійна нерозділеність класів.

39. Класифікація та види моделей нейромереж.

40. Загальне уявлення про навчання нейромереж.

41. Загальна характеристика та основні принципи побудови нейромереж.

42. Властивості штучних нейромереж.

43. Біологічні нейрони. Математичні моделі нейроелементів.

44. Використання мереж зі зворотними зв'язками при вирішенні практичних задач.

45. Розпізнавання зображень букв тексту за допомогою мереж Хопфілда.

46. Доцільність та обмеження практичного використання мереж зі зворотними зв'язками.

47.Порівняйте мережі прямого поширення та мережі зі зворотними зв'язками.

48.Обмеження методів синтезу мереж Хопфілда та засоби їх подолання.

49.Розпізнавання образів: постановка задачі та можливість застосування мереж зі зворотними зв'язками.

50.Асоціативний пошук інформації на основі мереж Хопфілда.

## **Лабораторна робота № 3**

### **НЕЙРОМЕРЕЖІ З ЛАТЕРАЛЬНИМИ ЗВ'ЯЗКАМИ**

**Мета роботи** – вивчити моделі нейронних мереж з латеральними зв'язками та методи їхнього навчання; порівняти можливості нейромереж з латеральними зв'язками з можливостями нейромереж прямого поширення та можливостями мереж зі зворотними зв'язками; ознайомитися з програмними засобами, що моделюють нейромережі з латеральними зв'язками.

#### **Завдання до роботи**

1. Ознайомитися з конспектом лекцій та рекомендованою літературою.
2. Вивчити архітектури нейромереж з латеральними зв'язками, а також методи їхнього навчання.
3. Використовуючи документацію рекомендованого викладачем програмного засобу, вивчити його архітектуру і компоненти, призначені для моделювання і навчання нейромереж з латеральними зв'язками, введення і виведення даних, підготовки звітів (виведення і відображення результатів роботи).
4. Визначити за табл. 1.1 згідно з номером варіанта студента (обирається за журналом) номер підрозділу Додатку В, що містить опис прикладної задачі, та номери контрольних питань до даної роботи, відповіді на які належить навести у звіті.
5. Використовуючи пакет MATLAB для відповідних варіанту студента вхідних даних вирішити прикладну задачу на основі нейромереж Кохонена LVQ, при цьому окремо дослідити властивості різних методів навчання. Для тієї ж задачі сформулювати кластери на основі карт ознак, що самоорганізуються.
6. Зберегти у файлі на диску початкові параметри нейромоделей, результати їхнього навчання (матрицю ваг та структуру з зазначенням функцій активації і кількості нейроелементів у шарах) та роботи для навчальної та контрольної вибірок (помилку класифікації (оцінювання), значення на входах і виході, час навчання, час класифікації).
7. Результати виконання пп. 5 - 6 занести в таблицю, стовпці якої повинні мати назви: назва архітектури нейромережі, кількість

шарів, кількість нейронів у шарах, функції активації нейронів у шарах, метод навчання, час навчання, час класифікації навченої мережі для навчальної вибірки, помилка класифікації для навчальної вибірки, час класифікації навченої мережі для тестової вибірки, помилка класифікації для тестової вибірки.

8. Проаналізувати отримані результати і дати порівняльну характеристику мереж з латеральними зв'язками SOFM та LVQ. Дати порівняльну характеристику мереж з латеральними зв'язками, мереж зі зворотними зв'язками та мереж прямого поширення за швидкістю навчання і точністю класифікації; принципами побудови архітектури.

9. Відповісти на контрольні питання.

10. Оформити звіт з роботи.

### **Зміст звіту**

1. Тема та мета роботи.

2. Номер варіанту та завдання до роботи, що містить стислий опис вирішуваної практичної задачі (не більше 0,5 сторінки).

3. Короткі теоретичні відомості (не більше 2 сторінок), що містять стислий опис архітектур та методів навчання нейромереж з латеральними зв'язками.

4. Опис процесу виконання роботи (не більше 3 сторінок).

5. Тексти програм, розроблених (модифікованих) студентом.

6. Вхідні дані та результати роботи програм, узагальнюючі таблиці, графіки.

7. Висновки, що містять відповіді на контрольні запитання (див. табл. 1.1), а також відображують результати виконання роботи та їх критичний аналіз.

### **Контрольні питання**

1. Дайте визначення понять: нейронна мережа, метрика, асоціативна пам'ять, SOFM, LVQ.

2. Які задачі можна вирішувати на основі SOFM і LVQ, а які не можна? Обґрунтуйте і доведіть відповідь. Приведіть приклади.

3. Порівняйте можливості SOFM, мереж Хопфілда та одношарового дискретного персептрона.

4. Як впливають вид і параметри метрики на точність визначення центрів зосередження екземплярів. Відповідь поясніть і проілюструйте.

5. Запропонуйте способи урахування апіорної інформації про значимість ознак при навчанні SOFM.

6. Чи доцільно використовувати SOFM для попереднього аналізу даних у задачах діагностики і прогнозування?

7. Запропонуйте стратегію (алгоритм) визначення необхідної кількості нейронів конкуруючого шару LVQ.

8. Карти ознак самоорганізації Кохонена: нейронна мережа SOFM.

9. Нейронна мережа LVQ.

10. Нейромережа "SOFM-АЗП".

11. Застосування мереж Кохонена у задачах кластер-аналізу та геоінформаційних системах.

12. Нейронні мережі Кохонена SOFM та LVQ у пакеті MATLAB.

13. У чому полягає навчання без учителя?

14. У чому полягає навчання з учителем?

15. Наведіть структуру мережі Кохонена SOFM.

16. У чому полягає навчання мережі Кохонена LVQ?

17. Що таке нормалізація даних?

18. З якою метою проводиться масштабування даних?

19. Назвіть основні принципи створення навчальної вибірки.

20. Що означає термін "кластеризація"?

21. Чи впливає репрезентативність навчальної вибірки на точність класифікації екземплярів тестової вибірки?

22. Чи впливає репрезентативність тестової вибірки на точність класифікації екземплярів тестової вибірки?

23. Чи впливає репрезентативність тестової вибірки на точність навчання мережі по навчальній вибірці?

24. Чи залежить якість навчання нейромереж від якості та обсягу навчальної вибірки?

25. Що таке генеральна сукупність, вибірка, екземпляр, ознака?

26. Вимоги до навчальних вибірок даних.

27. Що таке репрезентативна вибірка даних?

28. Чи повинна навчальна вибірка бути репрезентативною?

29. Чи повинна тестова вибірка бути репрезентативною?

30. Чи впливає обсяг навчальної вибірки на швидкість навчання нейромереж?

31. Опишіть структуру мережі SOFM.

32. Опишіть структуру мережі LVQ.

33. Опишіть структуру мережі SOFM-A3У.

34. У чому полягає метод навчання LVQ1?

35. У чому полягає метод навчання LVQ2?

36. У чому полягає метод навчання OLQV?

37. Які метрики можна використовувати для побудови мереж SOFM?

38. Обмеження мереж LVQ.

39. Застосування мереж SOFM для кластер-аналізу.

40. Застосування мереж SOFM у картографії.

41. Застосування мереж SOFM для створення пошукових систем.

42. Застосування мереж Кохонена для розпізнавання мови.

43. Застосування мереж SOFM для планування експерименту.

44. Нейронні мережі з латеральним гальмуванням.

45. Як впливає репрезентативність навчальної вибірки на результат побудови мережі SOFM?

46. Порівняйте мережі Кохонена з персептронами.

47. Порівняйте мережі Кохонена з бінарною мережею Хопфілда.

48. Порівняйте мережі Кохонена з мережею Ельмана.

49. Застосування мереж Кохонена у технічній діагностиці.

50. Застосування мереж Кохонена у медичній діагностиці.

## ЛІТЕРАТУРА

1. Demuth H. Neural Network Toolbox for use with MATLAB: user's guide / H. Demuth, M. Beale. – Natick: Mathworks Inc, 1997. – 700 p.
2. Интеллектуальные средства диагностики и прогнозирования надежности авиадвигателей: Монография / Дубровин В.И., Субботин С.А., Богуслаев А.В., Яценко В.К. – Запорожье: ОАО "Мотор-Сич", 2003. – 279 с.
3. Каллан Р. Основные концепции нейронных сетей / Р. Каллан. – М.: Издательский дом "Вильямс", 2001. – 287 с.
4. Круглов В.В. Искусственные нейронные сети: теория и практика / В.В. Круглов, В.В. Борисов. – М.: Горячая линия – Телеком, 2001. – 382 с.
5. Руденко О.Г. Основы теории искусственных нейронных сетей / О.Г. Руденко, Е.В. Бодянский – Харьков Телетех, 2002. – 317с.
6. Нейросетевые системы управления / В.А. Терехов, Д.В. Ефимов, И.Ю. Тюкин, В.Н. Антонов. – СПб.: Издательство Санкт-Петербургского университета, 1999. – 265 с.
7. Субботін С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень : Навчальний посібник / С. О. Субботін. – Запоріжжя: ЗНТУ, 2008. – 341 с.
8. Субботін С. О. Ітеративні, еволюційні та мультиагентні методи синтезу нечіткологічних і нейромережних моделей : Монографія / С. О. Субботін, А. О. Олійник, О. О. Олійник ; під заг. ред. С. О. Субботіна. – Запоріжжя: ЗНТУ, 2009. – 375 с.
9. Прогрессивные технологии моделирования, оптимизации и интеллектуальной автоматизации этапов жизненного цикла авиационных двигателей: Монография / А. В. Богуслаев, Ал. А. Олейник, Ан. А. Олейник, Д. В. Павленко, С. А. Субботин; Под ред. Д. В. Павленко, С. А. Субботина. – Запорожье: ОАО "Мотор Сич", 2009. – 468 с.
10. Хайкин С. Нейронные сети: полный курс, 2-е изд. / С. Хайкин: Пер. с англ. – М.: Вильямс, 2006. – 1104 с.

## Додаток А

### Моделювання нейронних мереж у пакеті MATLAB

MATLAB являє собою математичний пакет, призначений для вирішення задач обчислювальної математики та побудови чисельних моделей складних об'єктів і процесів.

Пакет MATLAB складається з інтерпретатора – модельного середовища, що має термінальний інтерфейс, ядра (набору найпростіших стандартних операцій, функцій і процедур для обчислень), а також бібліотек функцій (Toolbox).

Перевагами пакета є: багаті графічні можливості, великий набір математичних функцій, простота вбудованої мови MATLAB, можливість автоматичного перетворення текстів програм мовою MATLAB у тексти програм на мові Сі, а також те, що тексти бібліотечних функцій постачаються у вихідному виді. До недоліків пакета варто віднести відсутність дружнього інтерфейсу користувача і низьку швидкість роботи.

Для моделювання НМ за допомогою пакета MATLAB необхідно встановити і використовувати бібліотеку Neural Network Toolbox. Розглянемо приклади побудови моделей і навчання НМ мовою пакета MATLAB (табл. А.1 – А.6).

Таблиця А.1 – Моделювання і навчання радіально-базисних НМ

|                          |                                                                                                                                            |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| $x = [1 \ 2 \ 3];$       | Задаємо значення ознак екземплярів навчальної вибірки: 3 екземпляри (стовпці), 1 ознака (рядки).                                           |
| $y = [2.0 \ 4.1 \ 5.9];$ | Задаємо значення параметра, що прогнозується, для 3 екземплярів навчальної вибірки.                                                        |
| $net = newrb(x,y);$      | Створюємо і навчаємо радіально-базисну НМ                                                                                                  |
| $a = sim(net,x)$         | Обчислюємо по навченій мережі $net$ значення параметра, що прогнозується, для екземплярів, що характеризуються набором значень ознак $x$ . |

Таблиця А.2 – Моделювання і навчання НМ Хопфілда

|                                                                 |                                                                                                  |
|-----------------------------------------------------------------|--------------------------------------------------------------------------------------------------|
| $x = \begin{bmatrix} -1 & 1 \\ -1 & -1 \\ 1 & 1 \end{bmatrix};$ | Задаємо значення ознак екземплярів навчальної вибірки: 2 екземпляри (стовпці), 3 ознаки (рядки). |
| $net = newhop(x);$                                              | Створюємо і навчаємо НМ Хопфілда.                                                                |

Таблиця А.3 – Моделювання і навчання НМ LVQ

|                                                              |                                                                                                                                                                                                                                                                                                                               |
|--------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>x=[1 -2 2 0 4 -5 3];</code>                            | Задаємо значення ознак екземплярів навчальної вибірки: 7 екземплярів (стовпці), 1 ознака (рядки).                                                                                                                                                                                                                             |
| <code>y=[1 2 1 2 1 2 1];</code>                              | Задаємо номери класів для 7 екземплярів навчальної вибірки.                                                                                                                                                                                                                                                                   |
| <code>yc=ind2vec(y);</code>                                  | Перетворюємо номери класів у внутрішній формат.                                                                                                                                                                                                                                                                               |
| <code>net=newlvq(minmax(x),4, [0.6 0.4], 'learnlv1');</code> | Створюємо нейронну мережу <i>net</i> і визначаємо її топологію: діапазон зміни значень ознак визначається функцією <i>minmax</i> , кількість схованих нейронів (кластерів) – 4, апіорна імовірність віднесення екземплярів до одного класу – 0.6, до іншого – 0.4, у якості методу навчання мережі використовуємо метод LVQ1. |
| <code>net.trainParam.epochs= 1000;</code>                    | Задаємо максимально припустиму кількість циклів навчання (епох).                                                                                                                                                                                                                                                              |
| <code>net.trainParam.show= 100;</code>                       | Задаємо період відображення інформації про процес навчання на екрані в циклах (епохах) навчання.                                                                                                                                                                                                                              |
| <code>net.trainParam.lr=0.05;</code>                         | Задаємо крок навчання.                                                                                                                                                                                                                                                                                                        |
| <code>net=train(net,x,yc);</code>                            | Навчаємо нейронну мережу <i>net</i> на основі навчальної вибірки, представленій набором значень ознак екземплярів <i>x</i> і набором значень відповідних їм номерів класів <i>y</i> .                                                                                                                                         |
| <code>a=sim(net,x);</code>                                   | Обчислюємо по навченій мережі <i>net</i> номери класів для екземплярів, що характеризуються набором значень ознак <i>x</i> .                                                                                                                                                                                                  |
| <code>ac=vec2ind(a)</code>                                   | Перетворюємо номери кластерів у зручний для сприйняття формат і видаємо на екран.                                                                                                                                                                                                                                             |

Таблиця А.4 – Моделювання і навчання персептрона

|                                                                                                                            |                                                                                                                                                                                                                                                                                                                                                                                      |
|----------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $x = [0.1 \ 0.5 \ 0.2 \ 0.4 \ 0.3 \ 0.9;$<br>$0.9 \ 0.5 \ 0.8 \ 0.6 \ 0.7 \ 0.1;$<br>$0.3 \ 0.0 \ 0.6 \ 0.1 \ 0.2 \ 0.9];$ | Задаємо значення ознак екземплярів навчальної вибірки: 6 екземплярів (стовпці), 3 ознаки (рядки).                                                                                                                                                                                                                                                                                    |
| $y = [1 \ 0 \ 1 \ 0 \ 1 \ 0];$                                                                                             | Задаємо номери класів для 6 екземплярів навчальної вибірки.                                                                                                                                                                                                                                                                                                                          |
| $net=newff(repmat([0 \ 1], 3, 1),$<br>$[2,1], \ {'logsig', \ 'logsig'},$<br>$'trainlm');$                                  | Створюємо нейронну мережу <i>net</i> і визначаємо її топологію: діапазон зміни значень ознак $[0 \ 1]$ , кількість ознак – 3, кількість вихідних змінних – 1, на першому шарі – 2 нейрони, на другому шарі 1 – нейрон, нейрони 1 і 2 шарів мають сигмоїдні функції активації ( <i>logsig</i> ), для навчання мережі використовується метод Левенберга-Марквардта ( <i>trainlm</i> ). |
| $net.trainparam.show=25;$                                                                                                  | Задаємо період відображення інформації про процес навчання на екрані в циклах (епохах) навчання.                                                                                                                                                                                                                                                                                     |
| $net.trainparam.lr= 0.01;$                                                                                                 | Задаємо крок навчання.                                                                                                                                                                                                                                                                                                                                                               |
| $net.trainparam.epochs=500;$                                                                                               | Задаємо максимально припустиму кількість циклів навчання (епох).                                                                                                                                                                                                                                                                                                                     |
| $net.trainparam.goal=0.01;$                                                                                                | Задаємо максимально припустиме значення критерію навчання (помилки навчання).                                                                                                                                                                                                                                                                                                        |
| $ct=cputime;$                                                                                                              | Визначаємо і запам'ятовуємо поточне значення лічильника часу в змінній <i>ct</i> .                                                                                                                                                                                                                                                                                                   |
| $net=train(net, x, y);$                                                                                                    | Навчаємо нейронну мережу <i>net</i> на основі навчальної вибірки, представленої набором значень ознак екземплярів <i>x</i> і набором значень відповідних їм номерів класів <i>y</i> .                                                                                                                                                                                                |
| $ct=cputime-ct$                                                                                                            | Визначаємо поточне значення лічильника часу, віднімаємо від нього значення змінної <i>ct</i> – визначаємо час навчання НМ, що заносимо в змінну <i>ct</i> і видаємо на екран (ознака друку на екран – відсутність символу “,” наприкінці оператора).                                                                                                                                 |
| $a=round(sim(net, x));$                                                                                                    | Обчислюємо по навченій мережі <i>net</i> номер класів для екземплярів, що характеризуються набором значень ознак <i>x</i> .                                                                                                                                                                                                                                                          |

Таблиця А.5 – Моделювання і навчання НМ SOM

|                                            |                                                                                                                                                                         |
|--------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>x = rand(1,400);</code>              | Задаємо набір значень для однієї ознаки 400 екземплярів за допомогою генератора випадкових чисел                                                                        |
| <code>net=newsom(minmax(x), [2 5]);</code> | Створюємо нейронну мережу <i>net</i> і визначаємо її топологію: діапазон зміни значень ознак визначається функцією <i>minmax</i> , у першому шарі масив нейронів – 2х5. |
| <code>net=train(net, x);</code>            | Навчаємо нейронну мережу <i>net</i> (формуємо кластери) на основі навчальної вибірки, представленої набором значень ознак екземплярів <i>x</i> .                        |
| <code>y=sim(net,x);</code>                 | Обчислюємо по навченій мережі <i>net</i> номери кластерів, зіставлених нейронам вихідного шару НМ для екземплярів, що характеризуються набором значень ознак <i>x</i> . |
| <code>yc=vec2ind(y)</code>                 | Перетворюємо номери кластерів у зручний для сприйняття формат і видаємо на екран.                                                                                       |

Таблиця А.6 - Моделювання і навчання мереж Ельмана

|                                                                |                                                                                                                                                                                     |
|----------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>x = [1 0 1 1 1 0 1 1]</code>                             | Задаємо значення ознак екземплярів навчальної вибірки: 8 екземплярів, 1 ознака.                                                                                                     |
| <code>y = [0 0 0 1 1 0 0 1]</code>                             | Задаємо значення цільової ознаки екземплярів.                                                                                                                                       |
| <code>net = newelm([0 1], [5 1], {'tansig', 'logsig'});</code> | Створюємо мережу Ельмана, що складається з 5 нейронів на першому шарі та 1 нейрона на другому шарі, які мають функції активації ' <i>tansig</i> ' та ' <i>logsig</i> ', відповідно. |
| <code>net = train(net, x, y);</code>                           | навчаємо мережу Ельмана.                                                                                                                                                            |
| <code>Y = sim(net, x);</code>                                  | Обчислюємо по навченій мережі <i>net</i> номери класів для екземплярів, що характеризуються набором значень ознак <i>x</i> .                                                        |

Окрім розглянутих засобів моделювання НМ пакет MATLAB, починаючи з версії 6.0, містить візуальний інтерфейсний модуль *nntool*, який входить до бібліотеки Neural Network Toolbox.

Використання *nntool* дозволяє більш зручними засобами, ніж написання програми вручну, будувати нейромережеві моделі об'єктів та процесів. Розглянемо деякі основні можливості та прийоми роботи із засобом *nntool*.

Після запуску Matlab.exe в командному вікні для початку роботи із nntool треба ввести: nntool. Після цього завантажиться засіб nntool (рис. А.1).

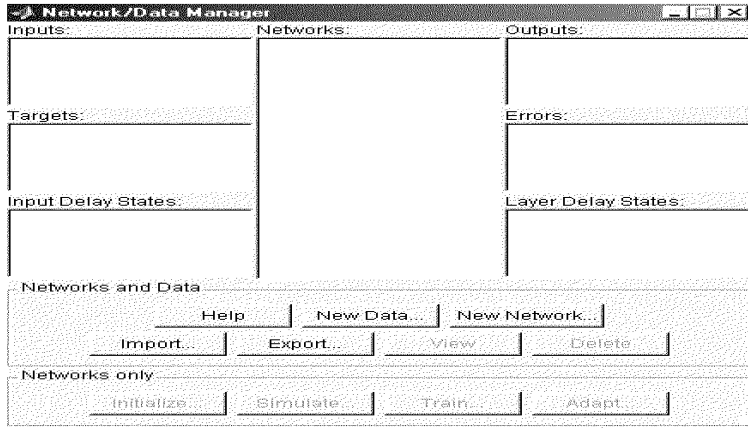


Рисунок А.1 – Головна діалогова форма засобу nntool

На панелі Network and Data ("Нейромережі та дані") користувач має натиснути кнопки для завдання вихідних даних для побудови нейромережевої моделі.

Кнопка New Data ("Нові дані") викликає редактор для створення нових даних (рис. А.2).

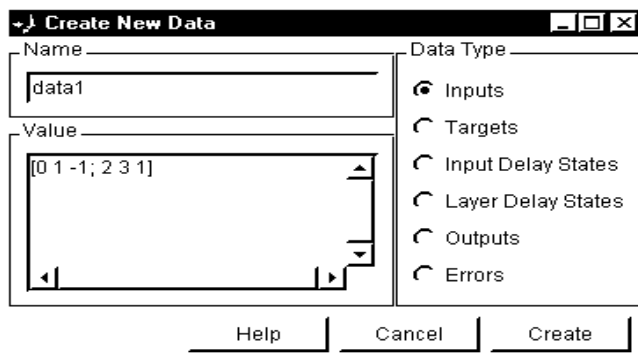


Рисунок А.2 - Редактор даних засобу nntool

Поле Name ("Ім'я") задає ім'я нової змінної середовища MATLAB, до якої зберігається масив нових даних, що вводяться у полі Value ("Значення"). Панель Data Type ("Тип даних") визначає призначення введених даних: Inputs - входи мережі, Targets - цільові значення виходів мережі, Input Delay States та Layer Delay States - описи затримок на входах та у шарах мережі, Outputs - фактичні значення на виходах мережі, Errors - помилки мережі.

Якщо дані уже існують у вигляді зовнішніх файлів або містяться у середовищі MATLAB у вигляді змінних, вони можуть бути імпортовані за допомогою кнопки Import ("Імпорт"). При цьому з'являється діалогова форма (рис. А.3).

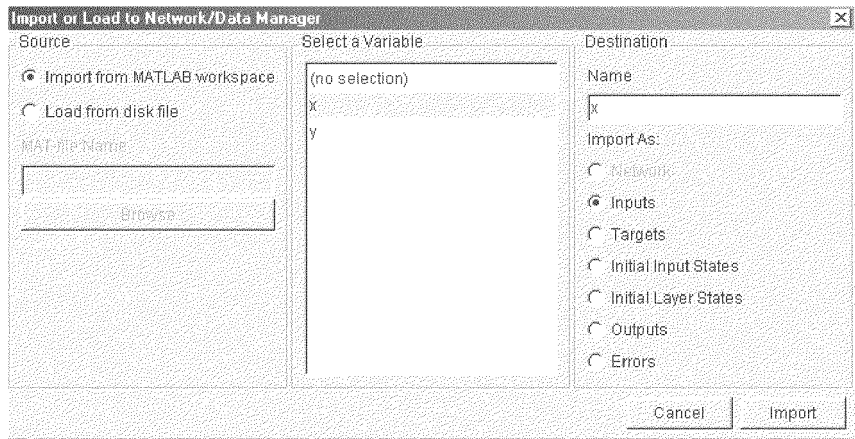


Рисунок А.3 - Діалогова форма імпорту даних

Поле Source ("Джерело") дозволяє обрати джерело введення даних: Import from Matlab workspace (імпорт даних із середовища MATLAB) або Load from disk file (завантаження даних із файлу на диску). Кнопка Browse дозволяє обрати необхідний файл.

Поле Select a Variable ("Вибір змінної") дозволяє вказати засобу nntool, яку змінну треба використовувати для імпорту даних.

Панель Destination ("Приймач") дозволяє задати змінну для прийому даних, що імпортуються. Її ім'я вказується у полі Name ("Ім'я"), а призначення (Import as) обирається із наведеного меню.

Кнопка Export ("Експорт") головної діалогової форми дозволяє зберегти дані із середовища nntool у файлі на диску, або передати їх до середовища MATLAB.

Кнопка "New Network" ("Нова мережа") викликає діалогову форму для конструювання нейромережі та визначення її параметрів (рис. А.4).

Рисунок А.4 - Форма конструювання нейромережі

Поле Network Name ("Ім'я мережі") визначає ім'я змінної, де зберігається мережа. Список вибору Network Type ("Тип мережі") дозволяє обрати тип архітектури мережі (наприклад, Feed-forward backprop - багатошарова нейромережа прямого поширення), поля Training Function, Adaptation Learning Function, Performance Function та Number of Layers визначають, відповідно, тип алгоритму навчання мережі, тип алгоритму адаптації ваг мережі, цільову функцію та кількість шарів мережі.

Панель Properties for Layer K дозволяє задати властивості для нейронів K-го шару мережі. У полі Number of Neurons вказують кількість нейронів для поточного шару, а у полі Transfer Function - тип функції активації нейронів поточного шару мережі.

Кнопка "View" дозволяє отримати графічне зображення схеми побудованої нейромережі (рис. А.5).

Кнопка "Delete" головної форми дозволяє видалити непотрібний елемент даних (змінну або мережу).

Кнопка "Help" дозволяє викликати довідкову службу MATLAB з описом необхідних компонентів та поясненнями щодо їхнього використання.

Після побудови нейромережі у нижній частині головної діалогової форми nntool стає доступною панель Networks only, що призначена для роботи із побудованою мережею (рис. А.6).

При натисненні будь-якої з кнопок цієї панелі викликається діалогова форма Network ("Мережа"), яка містить набір закладок-панелей для роботи з мережею (рис. А.7-А.10).

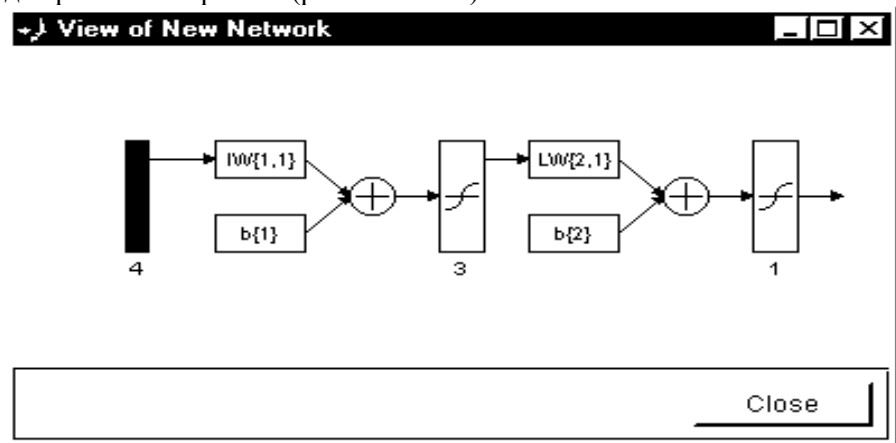


Рисунок А.5 - Приклад зображення форми нейромережі, побудованої за допомогою засобу nntool

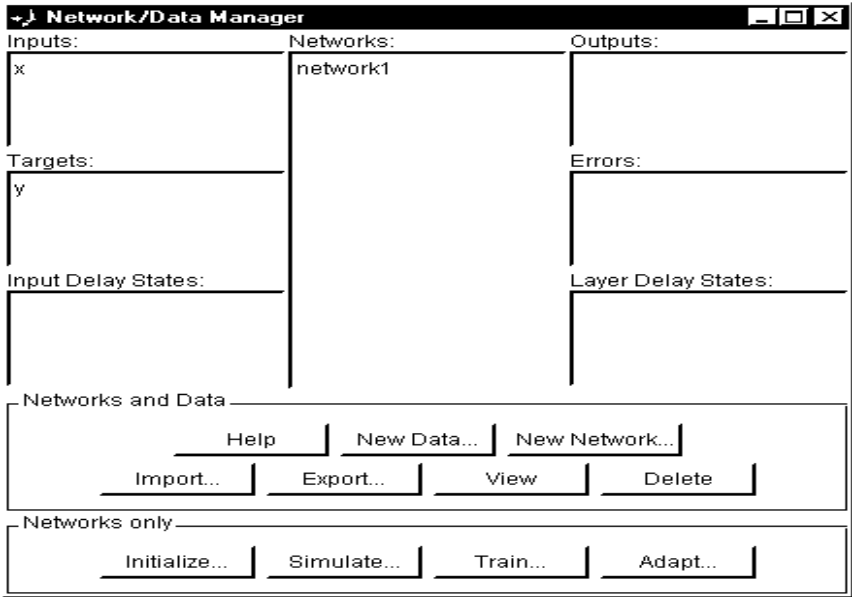


Рисунок А.6 - Головна діалогова форма nntool після побудови мережі

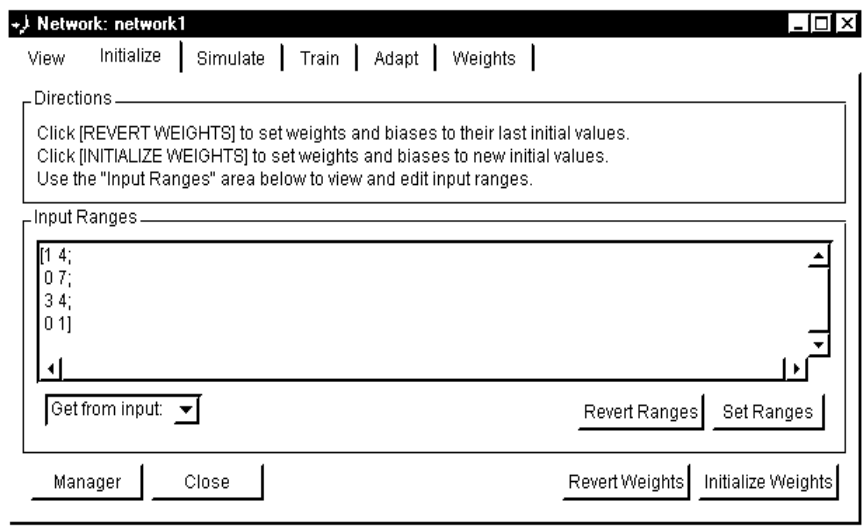


Рисунок А.7 - Форма Network: закладка Initialize

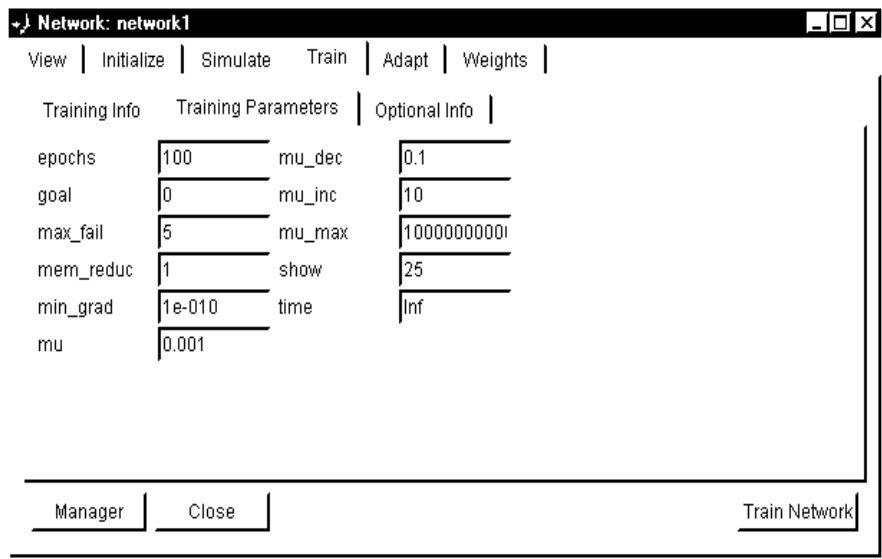


Рисунок А.8 - Форма Network: закладка Train

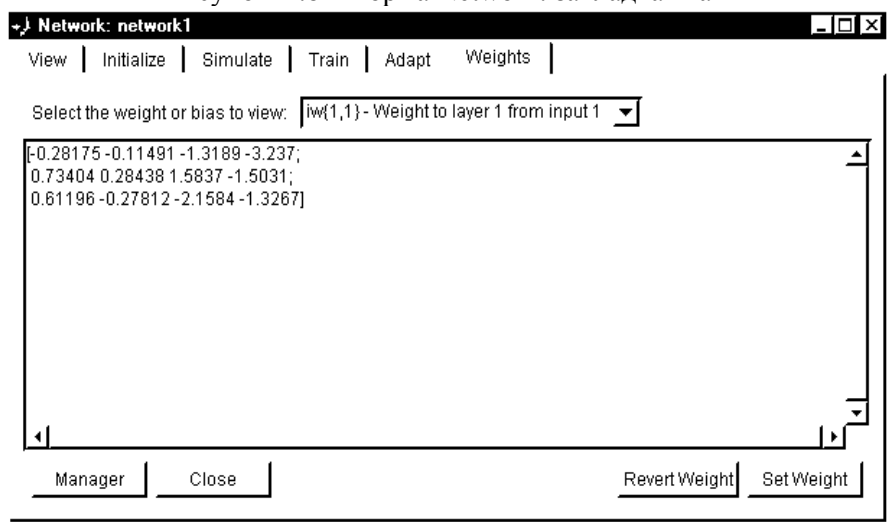


Рисунок А.9 - Форма Network: закладка Weights

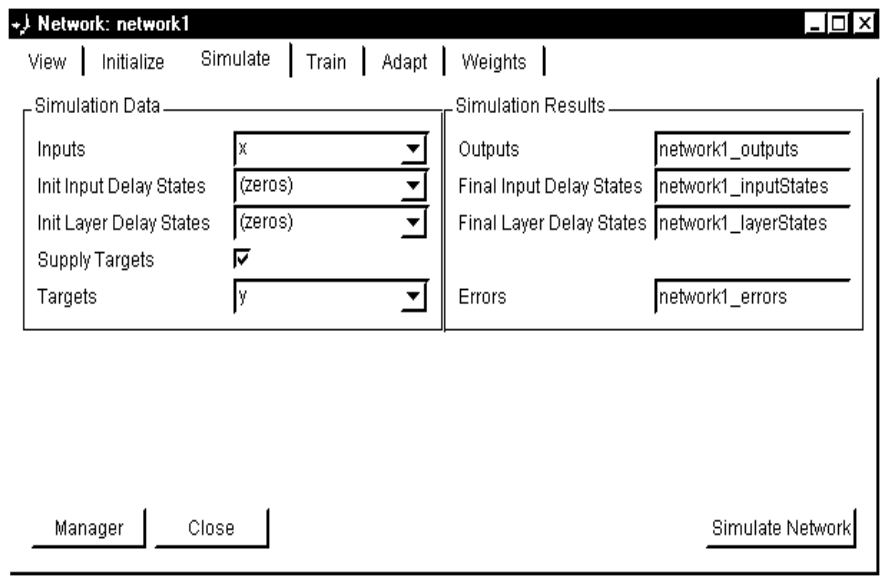


Рисунок А.10 - Форма Network: закладка Simulate

Закладка Initialize ("Ініціалізація") дозволяє задати межі, в яких змінюються вхідні дані та розрахувати на їхній основі початкові значення ваг мережі.

Ініціалізована мережа може бути навчена за допомогою закладки Train ("Тренування, навчання"). Серед параметрів навчання, доступних на цій закладці обов'язково необхідно задати: goal - максимально припустиме значення цільової функції, epochs - максимальна припустима кількість циклів навчання мережі, show - шаг виводу на екран інформації про навчання мережі, задається в циклах навчання.

В процесі навчання середовище MATLAB будує графік зміни значення цільової функції по епохах - циклах навчання (рис. А.11).

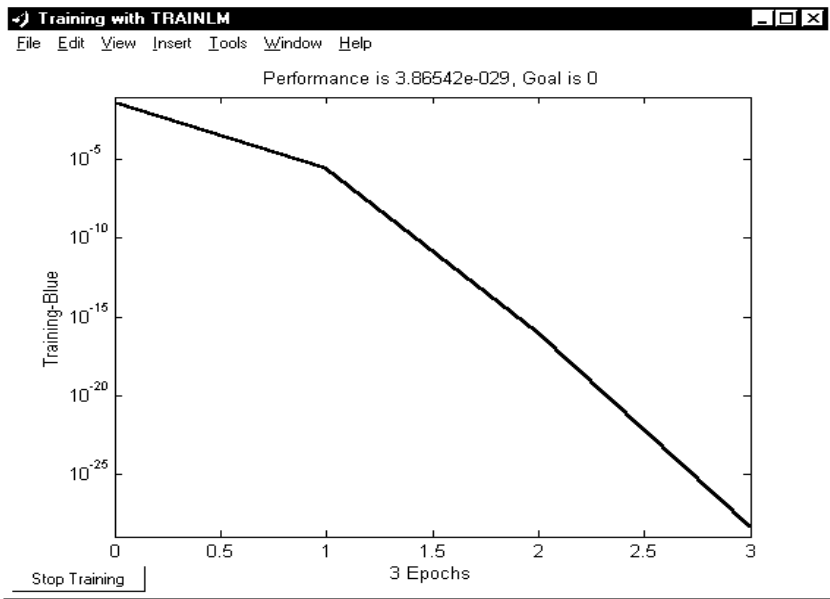


Рисунок А.11 - Графік зміни значення цільової функції в процесі навчання

Ваги мережі, що була навчена, можна переглянути, використовуючи закладку Weights ("Ваги").

Після того, як мережа навчилася, її можна використовувати для розпізнавання за допомогою закладки Simulate ("Моделювання").

Нейронні мережі, створені за допомогою функцій MATLAB, мають єдиний формат представлення, як це показано у табл. А.7.

Таблиця А.7 - Представлення структури нейромережі у MATLAB

| Поле структури                                                        | Опис, значення поля                                                     |
|-----------------------------------------------------------------------|-------------------------------------------------------------------------|
| numInputs                                                             | кількість входів мережі                                                 |
| numLayers                                                             | кількість (схованих) шарів мережі без урахування вхідного шару          |
| biasConnect, inputConnect, layerConnect, outputConnect, targetConnect | булеві масиви, що визначають зв'язки між структурними елементами мережі |

Продовження табл. А.7

|                                                                                 |                                                                                        |
|---------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| numOutputs                                                                      | кількість виходів мережі                                                               |
| numTargets                                                                      | кількість цільових ознак                                                               |
| numInputDelays                                                                  | кількість затримок вхідного шару                                                       |
| numLayerDelays                                                                  | кількість затримок схованих шарів                                                      |
| inputs                                                                          | входи мережі                                                                           |
| layers                                                                          | шари мережі                                                                            |
| outputs                                                                         | виходи мережі                                                                          |
| targets                                                                         | цільові ознаки                                                                         |
| biases                                                                          | масив порогів нейронів мережі                                                          |
| inputWeights                                                                    | масив вагових коефіцієнтів вхідного шару мережі                                        |
| layerWeights                                                                    | масив вагових коефіцієнтів схованих шарів мережі                                       |
| adaptFcn                                                                        | ім'я функції адаптації нейронів                                                        |
| initFcn                                                                         | ім'я функції ініціалізації мережі                                                      |
| performFcn                                                                      | ім'я цільової функції навчання (помилки)                                               |
| trainFcn                                                                        | ім'я функції, що реалізує процес навчання                                              |
| adaptParam                                                                      | параметри адаптації мережі                                                             |
| initParam                                                                       | параметри ініціалізації мережі                                                         |
| performParam                                                                    | параметри цільової функції мережі                                                      |
| trainParam.epochs                                                               | максимально допустима кількість ітерацій навчання (епох)                               |
| trainParam..goal                                                                | максимально допустиме значення цільової функції навчання                               |
| trainParam.max_fail                                                             | максимально допустима кількість відмов у процесі навчання мережі                       |
| trainParam.mem_reduc                                                            | коефіцієнт, що регулює (зменшує) використання пам'яті при навчанні нейромереж          |
| trainParam.min_grad                                                             | мінімально допустиме значення градієнту цільової функції                               |
| trainParam.mu,<br>trainParam.mu_dec,<br>trainParam.mu_inc,<br>trainParam.mu_max | параметри методу Левенберга-Марквардта                                                 |
| trainParam.show                                                                 | кількість ітерацій, через яку будуть відображатися зміни стану процесу навчання мережі |
| trainParam.time                                                                 | максимально допустимий час навчання мережі у секундах                                  |
| IW                                                                              | масив значень ваг вхідного (першого) шару                                              |
| LW                                                                              | масив значень ваг схованих шарів                                                       |
| b                                                                               | масив значень порогів нейронів                                                         |
| userdata                                                                        | дані користувача                                                                       |

## **Додаток Б**

### **Моделювання нейронних мереж у пакеті Statistica Neural Networks**

Statistica Neural Networks - це універсальний пакет нейромережевого аналізу фірми StatSoft. Він може працювати як самостійний додаток, а також у рамках системи Statistica або Quick Statistica.

Пакет має ряд відмінних рис.

1. Найсучасніші і могутні алгоритми навчання мережі (включаючи методи сполучених градієнтів та Левенберга-Марквардта); можливість створювати складні, практично не обмежені в розмірах комбінації з мереж різних архітектур; вибіркове навчання фрагментів нейронної мережі; повний контроль над вибором функцій активації; убудована система Нейро-Майстер - Network Wizard, що допомагає користувачу приймати рішення і вибирати параметри.

2. Виняткова простота у використанні плюс неперевершена аналітична міць; так наприклад, Автоматичний Конструктор Мережі - Automatic Network Designer знайде найкращу архітектуру для Вашої мережі (вибір потрібної архітектури мережі - це важкий процес проб і помилок; Statistica Neural Networks зробить це за користувача).

3. Могутні методи аналізу, у тому числі, унікальний Інструмент нейро-генетичного відбору вхідних даних - Neuro-Genetic Input Selection (вибір потрібних вхідних змінних при розвідницькому аналізі даних - області, де найчастіше застосовуються нейромережі - займає багато часу; модуль Statistica Neural Networks здатний виконувати цю роботу за користувача).

4. Statistica Neural Networks може працювати в середовищі Statistica (чи Quick Statistica), але також може використовуватися і як самостійний додаток (є можливість уведення-виведення файлів даних системи Statistica; матриць вихідних даних і графіки).

5. Statistica Neural Networks цілком підтримує API (інтерфейс прикладного програмування), так що досвідчені користувачі (і розроблювачі користувацьких систем "пошуку знань" і "видобутку даних") можуть використовувати могутні обчислювальні можливості модуля Statistica Neural Networks у своїх власних додатках.

Простий і зручний інтерфейс системи Statistica Neural Networks дозволяє швидко створювати нейромережеві додатки для вирішення користувацьких задач.

Інтерфейс прикладного програмування Statistica Neural Networks' API дозволяє вмонтувати ці додатки у вже наявну в систему, наприклад, зробити їх частиною більш широкого обчислювального середовища (це можуть бути процедури, розроблені окремо й вбудовані в комп'ютерну мережу фірми). API поставляється в двох варіантах. Скорочена версія API дозволяє запускати створені і навчені в середовищі Statistica Neural Networks нейронні мережі з інших програм-додатків, написаних на C, C++, Delphi чи Visual Basic. Повний API надає користувачу доступ до всіх можливостей ядра пакета Statistica Neural Networks, включаючи створення, редагування, уведення/виведення даних і архітектур мереж, навчання мереж.

Пакет має багатовіконний Windows інтерфейс. Головне і дочірнє вікна пакета зображені на рис. Б.1.

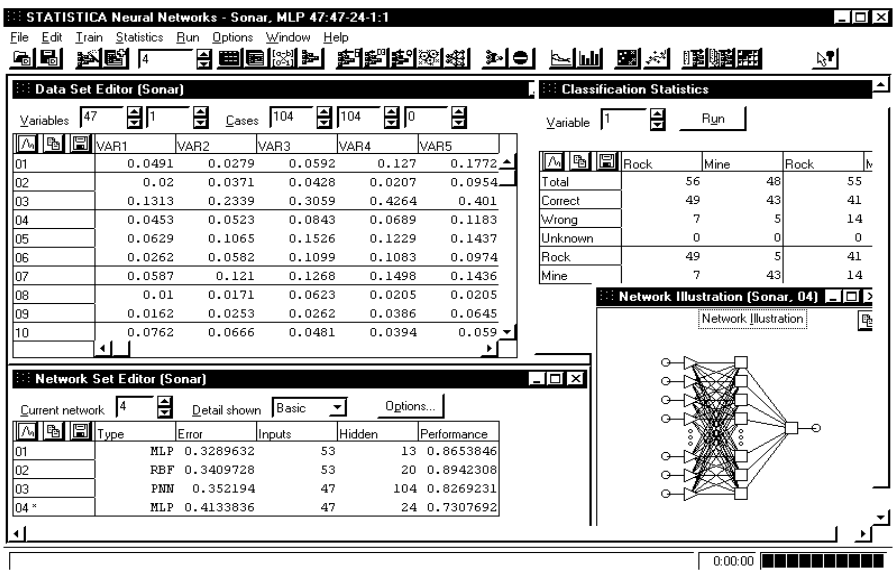


Рисунок Б.1 - Головне вікно пакета Statistica Neural Networks

Розглянемо призначення основних пунктів меню пакета.

Меню File (Файл) містить пункти: 1) New - новий(ва): Network - нейронна мережа, Intelligent Problem Solver - інтелектуальний вирішувач задач, Data Set - вибірка даних; 2) Open - відкрити файл, 3) Close -

закрити файл, 4) Save - зберегти в поточний файл, 5) Save as - зберегти у файл з ім'ям, 6) Network Set - набір нейромереж, 7) Exit - вихід.

Меню Edit (Редагувати) містить пункти: 1) Variables - змінні, ознаки, 2) Cases - випадки, екземпляри; 3) Undo - скасувати дію; 4) Cut - вирізати; 5) Copy - копіювати до буфера; 6) Paste - вставити з буфера; 7) Clear - очистити; 8) Data Set - набір даних; 9) Network Set - набір нейромереж; 10) Pre/post Processing - перед / пост-обробка; 11) Network - нейромережа.

Меню Train (Навчання) містить пункти: 1) Multilayer Perceptrons - багатошарові перцептрони; 2) Radial-Basis Functions - радіально-базисні мережі; 3) Linear - лінійні мережі; 4) Kohonen - мережі Кохонена (SOM); 5) Class labels - мітки класів; 6) LVQ - мережі LVQ; 7) Probabilistic - імовірнісні мережі; 8) General Regression - узагальнено-регресійні мережі; 9) Principal Components - аналіз головних компонент; 10) Auxiliary - помічник: Best Network - добір кращої мережі, Stopping Conditions - критерії зупинення, Set Weights - установка ваг, Weigend Regularization - регуляція ваг, Feature Selection - відбір ознак.

Меню Statistics (Статистика) містить пункти: 1) Training Graph - графік навчання, 2) Case Errors - помилки для екземплярів, 3) Classification - класифікація, 4) ROC Curve - крива характеристики дії одержувача, 5) Regression - регресія, 6) Weight Distribution - розподіл ваг, 7) Win Frequencies - частоти переможців, 8) Sensivity - чутливість, 9) Network Illustration - зображення мережі, 10) Messages - повідомлення.

Меню Run (Запуск) містить пункти: Single Case - запуск для одного екземпляра, Data Set - запуск для набору даних, One-off - запуск для одного з прикладів, обраних інтерактивно, Activations - активація мережі, Time Series - часові ряди, Response Graph - графік відгуку, Response Surface - поверхня відгуку, Topological Map - топологічна карта, Cluster Diagram - діаграма кластерів, Code Generation - генерація коду.

Вибір вхідних змінних - це винятково важливий етап при побудові нейронної мережі. Перед тим, як безпосередньо починати працювати з пакетом Statistica Neural Networks, має сенс зробити попередній відбір змінних, використовуючи при цьому свої знання в предметній області та стандартні статистичні критерії. Потім, уже засобами пакета Statistica Neural Networks можна буде спробувати різні комбінації вхідних змінних.

У пакеті Statistica Neural Networks є можливість "ігнорувати" деякі змінні, так що отримана мережа не буде використовувати їх як входи. Можна по черзі експериментувати з різними комбінаціями входів, будуючи кожного разу нові варіанти мереж.

При таких експериментуванні дуже корисними виявляються імовірнісні й узагальнено-регресійні мережі. Незважаючи на те, що вони працюють повільніше більш компактних багат шарових перцептронів і радіально-базисних мереж, вони навчаються майже миттєво, і це важливо, оскільки при переборі великого числа комбінацій вхідних змінних приходить ся щораз будувати нові мережі. Крім того, імовірнісні й узагальнено-регресійні мережі - це радіальні мережі (у першому шарі вони мають радіальні елементи, і апроксимуючі функції будуються у виді комбінацій гаусівських функцій). При відборі вхідних змінних це є перевагою, оскільки радіальні мережі в меншому ступені страждають від прокльону розмірності, чим мережі, побудовані на лінійних елементах.

У пакеті Statistica Neural Networks є інструмент, що може автоматизувати пошук найбільш удакої комбінації вхідних змінних на основі генетичного алгоритму. Генетичні алгоритми добре пристосовані для задач такого типу, оскільки вони дозволяють робити пошук серед великого числа комбінацій при наявності внутрішніх залежностей у змінних.

Існує й інший підхід до проблеми розмірності, що може використовуватися як альтернатива або як доповнення до методів відбору змінних: це зниження розмірності. Суть його полягає в тому, що вихідна сукупність змінних перетворюється в нову сукупність, що складається з меншого числа змінних, але при цьому (як ми сподіваємося) утримуючу по можливості всю інформацію, закладену у вихідних даних.

Найпоширеніший метод зниження розмірності - це аналіз головних компонентів. Метод полягає в наступному: до даних застосовується лінійне перетворення, при якому напрямкам нових координатних осей відповідають напрямку найбільшого розкиду вихідних даних. Як правило, вже перший компонент відбиває велику частину інформації, що міститься в даних. Оскільки аналіз головних компонентів (АГК) являє собою лінійний метод, його можна реалізувати за допомогою лінійної мережі, і в пакеті Statistica Neural Networks передбачена можливість навчати лінійну мережу для виконання АГК. Дуже часто ме-

тод АГК виділяє з багатомірних вихідних даних зовсім невелике число компонентів, зберігаючи при цьому структуру інформації.

Один з недоліків методу головних компонентів (АГК) полягає в тому, що це чисто лінійний метод, і через це він може не враховувати деякі важливі характеристики структури даних.

У пакеті Statistica Neural Networks реалізований також варіант "нелінійного АГК", заснований на використанні так називаної автоасоціативної мережі. Це така нейронна мережа, навчається видавати як виходи свої власні вхідні дані, але при цьому в її проміжному шарі міститься менше нейронів, чим у вхідному і вихідному шарах. Тому, щоб відновити свої вхідні дані, мережа повинна навчитися представляти їх у більш низькій розмірності. Мережа перетворює спостереження у формат проміжного шару і тільки потім видає їх на виході. Після навчання автоасоціативної мережі її зовнішній інтерфейс може бути збережений і використаний для зниження розмірності.

Як правило, у якості автоасоціативної мережі береться багатошаровий персептрон із трьома проміжними шарами. При цьому середній шар відповідає за представлення даних у малій розмірності, а два інших схованих шари служать відповідно для нелінійного перетворення вхідних даних у середній шар і виходів середнього шару у вихідний шар. Автоасоціативна мережа з єдиним проміжним шаром може виконувати тільки лінійне зниження розмірності, і фактично здійснює АГК у стандартному варіанті.

Процес побудови мережі (після вибору вхідних змінних) складається з таких кроків.

1. Вибрати початкову конфігурацію мережі (наприклад, один проміжний шар з числом елементів у ньому, рівним напівсумі числа входів і числа виходів - Наставник (Network Advisor) пакета Statistica Neural Networks запропонує таку конфігурацію за замовчуванням).

2. Провести ряд експериментів з різними конфігураціями, запам'ятовуючи при цьому кращу мережу (у змісті контрольної помилки). У пакеті Statistica Neural Networks передбачене автоматичне запам'ятовування кращої мережі під час експерименту. Для кожної конфігурації варто провести кілька експериментів, щоб не одержати помилковий результат через те, що процес навчання потрапив у локальний мінімум.

3. Якщо в черговому експерименті спостерігається недонавчання (мережа не видає результат прийнятної якості), спробувати додати

додаткові нейрони в проміжний шар (шари). Якщо це не допомагає, спробувати додати новий проміжний шар.

4. Якщо має місце перенавчання (контрольна помилка стала зростати), спробувати видалити кілька схованих елементів (а можливо і шарів).

Багаторазове повторення евристичних експериментів у кращому випадку є досить утомливим, і тому в пакет Statistica Neural Networks включений спеціальний алгоритм автоматичного пошуку.

Автоматичний конструктор мережі - Automatic Network Designer проведе експерименти з різним числом схованих елементів, для кожної спробної архітектури мережі виконає кілька прогонів навчання, відбираючи при цьому найкращу мережу по показнику контрольної помилки з виправленням на розмір мережі.

В Автоматичному конструкторі мережі реалізовані складні алгоритми пошуку, у тому числі метод "імітації віджигу" (simulated annealing), за допомогою яких можна перепробувати сотні різних мереж, виділяючи з них особливо перспективні, або швидко знаходити "грубе і просте" рішення.

У задачі класифікації мережа повинна віднести кожне спостереження до одного з декількох класів (чи, у більш загальному випадку, оцінити імовірність приналежності спостереження до кожного з класів). У пакеті Statistica Neural Networks для класифікації використовуються номінальна вихідна змінна - різні її значення відповідають різним класам.

У пакеті Statistica Neural Networks класифікацію можна здійснювати за допомогою мереж наступних типів: багат шарового перцептрона, радіально-базисних функцій, мережі Кохонена, імовірнісної нейромережі та лінійної мережі. Єдина з мереж пакета Statistica Neural Networks, не призначена для задач класифікації, - це узагальнено-регресійна мережа.

Номінальні змінні представляються в пакеті Statistica Neural Networks в одному з двох видів (перший з них підходить тільки для змінних із двома значеннями): 1) бінарному (два стани) та 2) один-з-N. При бінарному представленні змінній відповідає один вузол мережі, при цьому значення 0,0 означає активний стан, а 1,0 - неактивний. При кодуванні 1-з-N на кожен стан виділяється один елемент, так що кожен конкретний стан представляється як 1,0 у відповідному елементі та 0,0 у всіх інших.

Номинальні вхідні змінні в пакеті Statistica Neural Networks можуть бути перетворені одним з цих методів як на етапі навчання мережі, так і при її роботі. Цільові вихідні значення для елементів, що відповідають номінальним змінним, також легко визначаються під час навчання. Більш великі зусилля вимагаються для того, щоб за результатами роботи мережі визначити вихідний клас.

Кожний з вихідних елементів буде містити числові значення в інтервалі від 0,0 до 1,0. Щоб упевнено визначити клас по наборі вихідних значень, мережа повинна вирішити, чи "досить близькі" вони до нуля або одиниці. Якщо такої близькості не спостерігається, клас вважається "невизначеним".

Крім того, у пакеті Statistica Neural Networks для інтерпретації вихідних значень використовуються довірчі рівні (пороги прийняття і відкидання). Ці граничні значення можна корегувати, щоб змусити мережу бути більш, чи, навпаки, менш "рішучою" при оголошенні класу. Схеми тут не сильно розрізняються для випадків бінарного та 1-3-N представлень.

Бінарне. Якщо вихідне значення елемента перевищує поріг прийняття, то вибирається клас 1,0. Якщо вихідне значення лежить нижче порога відкидання, вибирається клас 0,0. Якщо вихідне значення лежить між порогами, клас вважається не визначеним.

Один-3-N. Визначений клас вибирається тільки в тому випадку, якщо значення відповідного вихідного елемента вище порога прийняття, а всіх інших вихідних елементів - нижче порога відкидання. Якщо ж дана умова не виконана, клас не визначається.

При кодуванні методом 1-3-N є одна особливість. На перший погляд здається, що "найбільш рішучою" буде мережа з порогами прийняття і відкидання, рівними 0,5. Це дійсно так для бінарного кодування, але вже не так для кодування 1-3-N. Можна зробити так, щоб поріг прийняття був нижче порога відкидання, і найбільш рішучою буде мережа, у якої поріг прийняття 0,0, а поріг відкидання 1,0. При такому, на перший погляд, дивному настроюванню мережі елемент із найвищим рівнем активації буде визначати клас поза залежністю від того, що відбувається в інших елементах.

Схема дії алгоритму визначення класу в пакеті Statistica Neural Networks полягає у виконанні таких кроків.

1. Вибирається елемент із найвищим вихідним сигналом. Якщо його вихідний сигнал вище або дорівнює порогові прийняття, а вихід-

ні сигнали всіх інших елементів нижче порога відкидання, то як відповідь видати клас, обумовлений цим елементом.

2. При порозі прийняття 0,0 вихідний сигнал елемента, що виграв, завжди буде прийнятий, а при порозі відкидання 1,0 всі інші елементи неминуче будуть відкинуті, і тому алгоритм зводиться до простого вибору елемента, що виграв. Якщо ж обое граничних значення - прийняття та відкидання - установити на рівні 0,5, мережа цілком може залишитися в нерішучості (у випадках, коли в переможця результат нижче 0.5 чи в когось із тих, що програли - вище 0,5).

Усе сказане відноситься до механізму вибору класу для більшості типів мереж. На відміну від них, мережа Кохонена діє зовсім інакше.

У мережі Кохонена елементом топологічної карти (вихідного шару), що виграв, є той, у якого найвищий рівень активації (він вимірює відстань від вхідного приклада до точки, координати якої зберігаються в елементі мережі). Деякі чи навіть всі елементи топологічної карти можуть бути позначені іменами класів. Якщо ця відстань є достатньо малою, то даний випадок зараховується до відповідного класу (за умови, що зазначене ім'я класу). У пакеті Statistica Neural Networks значення порога прийняття - це найбільша відстань, на якій приймається позитивне рішення про класифікацію спостереження. Якщо ж вхідний випадок лежить від елемента, що виграв, на більш далекій відстані або якщо елемент, що виграв, не був позначений (або якщо його мітка не відповідає жодному зі значень вихідний номінальної змінної), то випадок залишається некласифікованим. Поріг відкидання в мережах Кохонена не використовується.

У наших розглядах ми припускали, що "позитивному" рішенням про класифікацію повинне відповідати значення, близьке до 1,0, а "негативному" - близьке до 0,0. Це дійсно так у тому випадку, якщо на виході використовуються логістичні функції активації. Крім того, це зручно, оскільки імовірність може приймати значення від 0,0 до 1,0. Однак, у деяких ситуаціях може виявитися більш зручним використовувати інший діапазон. Іноді застосовується зворотна упорядкованість, так що позитивне рішення відповідає малим вихідним значенням. Пакет Statistica Neural Networks підтримує кожної з цих варіантів роботи.

Спочатку як границі діапазону для кожної змінної використовуються значення мінімум/середнє і максимум/стандартне відхилення. Для логістичної вихідної функції активації гарними значеннями за за-

мовчуванням є 0,0 і 1,0. Деякі автори радять використовувати як функцію активації гіперболічний тангенс, що приймає значення в інтервалі  $(-1,0, +1,0)$ . Таким прийомом можна поліпшити навчання, тому що ця функція (на відміну від логістичної) симетрична. У цьому випадку потрібно змінити значення мінімум/середнє і максимум/стандартне відхилення, і програма Statistica Neural Networks автоматично буде правильно інтерпретувати класи.

Зворотна упорядкованість, як правило, застосовується в двох ситуаціях. Одна з них - це мережі Кохонена, у яких вихідне значення є міра далекості, і її мале значення відповідає більшій довірі. Друга ситуація виникає при використанні матриці втрат (яка може бути додана у імовірнісну мережу на етапі її побудови або вручну - до мереж інших типів). Якщо використовується матриця втрат, то виходи мережі означають очікувані утрати від вибору того чи іншого класу, і ціль полягає в тім, щоб вибрати клас з найменшими втратами. Упорядкованість можна звернути, оголосивши вихідний сигнал не рівнем довіри, а мірою помилки. У такому випадку поріг прийняття буде нижче порога відкидання.

При виборі порогів прийняття/відкидання й оцінці здібностей мережі до класифікації дуже допомагає інформація, що міститься у вікні Статистики класифікації - Classification Statistics. У ньому вказується, скільки спостережень було класифіковано правильно, скільки неправильно взагалі чи не класифіковано. Крім того, видається інформація про те, скільки спостережень кожного класу було віднесено до інших класів. Усі ці дані видаються окремо для навчальної, контрольної і тестової множин.

У задачах оцінювання (регресії) метою є оцінка значення числової вихідної змінної за значеннями вхідних змінних. Задачі регресії в пакеті Statistica Neural Networks можна вирішувати за допомогою: багатошарових перцептронів, мереж радіально-базисних функцій, узагальнено-регресійних і лінійних мереж. При цьому вихідні дані повинні мати стандартний числовий (не номінальний) тип.

Особливу важливість для регресії мають масштабування (шкалювання) вихідних значень і ефекти екстраполяції.

Нейронні мережі найбільше часто використовуваних архітектур видають вихідні значення в деякому визначеному діапазоні (наприклад, на відрізку  $[0,1]$  у випадку логістичної функції активації). Для задач класифікації це не створює труднощів. Однак для задач регресії

зовсім очевидно, що отут є проблема, і деякі її деталі виявляються дуже тонкими.

Для початку застосуємо алгоритм масштабування, щоб вихід мережі мав "прийнятний" діапазон. Найпростішою з масштабуючих функцій пакета Statistica Neural Networks є мінімаксна функція: вона знаходить мінімальне і максимальне значення змінної по навчальній множині і виконує лінійне перетворення (із застосуванням коефіцієнта масштабу і зсуву), так щоб значення лежали в потрібному діапазоні (як правило, на відрізку  $[0,0, 1,0]$ ). Якщо ці дії застосовуються до числової вихідної змінної, тобто є гарантія, що всі навчальні значення після перетворення потраплять в область можливих вихідних значень мережі, тоді мережа може бути навчена. Крім того, ми знаємо, що виходи мережі повинні знаходитися у визначених межах. Цю обставину можна вважати достоїнством або недоліком - тут ми приходимо до питань екстраполяції.

Припустимо, наприклад, що ми використовуємо багат шаровий персептрон. Застосування мінімакса по описаній вище схемі є дуже обмеженим. По-перше, функція не буде екстраполюватися, як би близько ми не знаходилися до навчальних даних (у дійсності ж, якщо ми лише мало-мало вийшли за область навчальних даних, екстраполяція цілком виправдана). По-друге, оцінка по середньому також не буде виконуватися: замість цього буде братися мінімум або максимум дивлячись по тому, зростала чи убувала в цьому місці оцінювана крива.

Щоб уникнути цих недоліків у персептроні використовується ряд прийомів.

По-перше, логістичну функцію активації у вихідному шарі можна замінити на лінійну, котра не змінює рівня активації (помітимо, що функції активації змінюються тільки у вихідному шарі; у проміжних шарах як і раніше залишаються логістичні та гіперболічні функції активації). Лінійна функція активації не насичується, і тому здатна екстраполювати (при цьому логістичні функції попередніх рівнів усе-таки припускають насичення на більш високих рівнях). Лінійні функції активації у персептроні можуть викликати визначені обчислювальні труднощі в алгоритмі зворотного поширення, тому при його використанні варто брати малі (менш 0,1) швидкості навчання. Описаний підхід придатний для цілей екстраполяції.

По-друге, можна змінити цільовий діапазон мінімаксної масштабуючої функції (наприклад, зробити його  $[0,25, 0,75]$ ). У результаті навчальні спостереження будуть відображатися в рівні, що відповідає се-

редній частини діапазону вихідних значень. Цікаво помітити, що якщо цей діапазон обраний маленьким, і обидві його границі знаходяться поблизу значення 0,5, то він буде відповідати середній ділянці сигмоїдної кривої, на якому вона "майже лінійна", - тоді ми будемо мати практично ту ж схему, що й у випадку лінійного вихідного шару. Така мережа зможе виконувати екстраполяцію у визначених межах, а потім буде насичуватися. Усе це можна добре собі представити так: екстраполяція припустима у визначених границях, а поза ними вона буде припинятися.

Якщо застосовується перший підхід і у вихідному шарі поміщені лінійні елементи, то може вийти так, що взагалі немає необхідності використовувати алгоритм масштабування, оскільки елементи і без масштабування можуть видавати будь-який рівень вихідних сигналів. У пакеті Statistica Neural Networks мається можливість для більшої ефективності узагалі відключити всі масштабування. Однак, на практиці повне відмовлення від масштабування приводить до труднощів в алгоритмах навчання. Дійсно, у цьому випадку різні ваги мережі працюють у різних масштабах, і це ускладнює початкову ініціалізацію ваг і (частково) навчання. Тому не рекомендується відключати масштабування, за винятком тих випадків, коли діапазон вихідних значень дуже малий і розташований поблизу нуля. Це ж розуміння свідчить на користь масштабування і при передобробці у персептронах (при який, у принципі, ваги першого проміжного шару можна легко коректувати, домагаючись цим будь-якого потрібного масштабування).

Радіальні мережі по своїй природі нездатні до екстраполяції. Чим далі вхідний приклад розташований від точок, що відповідають радіальним елементам, тим менше стають рівні активації радіальних елементів і (зрештою) тим менше буде вихідний сигнал мережі. Вхідний приклад, розташований далеко від центрів радіальних елементів, дасть нульовий вихідний сигнал. Прагнення мережі не екстраполювати дані можна вважати достоїнством (це залежить від предметної області і думки користувача), однак убування вихідного сигналу (на перший погляд) не є достоїнством. Якщо ми прагнемо уникати екстраполяції, то для вхідних точок, що відрізняються великим ступенем новизни, у якості виходу ми, як правило, хочемо мати усереднене значення.

Для радіальних мереж у задачах регресії цього можна досягти за допомогою масштабуючої функції середнє/стандартне відхилення. Навчальні дані масштабуються таким чином, щоб середнє вихідне

значення дорівнювало 0,0, а всі інші значення були б промасштабовані на стандартне відхилення вихідних сигналів. При обробці вхідних точок, що лежать поза областями дії радіальних елементів, вихідний сигнал мережі буде приблизно дорівнює середньому значенню.

Якість роботи мережі в задачі регресії можна перевірити декількома способами.

По-перше, мережі можна повідомити вихідне значення, що відповідає будь-якому спостереженню (чи якомусь новому спостереженню, що необхідно перевірити). Якщо це спостереження містилося у вихідних даних, то видасться значення різниці (нев'язання).

По-друге, можуть бути отримані підсумкові статистики. До них відносяться середнє значення і стандартне відхилення, обчислені для навчальних даних і для помилки прогнозу. У загальному випадку середнє значення помилки прогнозу буде дуже близьким до нуля (зрештою, нульове середнє для помилки прогнозу можна одержати, попросту оцінивши середнє значення навчальних даних і зовсім не звертаючись до значень вхідних змінних). Найбільш важливим показником є стандартне відхилення помилки прогнозу. Якщо воно не виявиться істотно менше стандартного відхилення навчальних даних, це буде означати, що мережа працює не краще, ніж проста оцінка по середньому. Далі, у пакеті Statistica Neural Networks користувачу видасться відношення стандартного відхилення помилки прогнозу до стандартного відхилення навчальних даних. Якщо воно істотно менше одиниці (наприклад, нижче 0.1), то це говорить про гарну якість регресії. Це регресійне відношення (точніше, величину одиниця мінус це відношення) іноді називають часткою поясненої дисперсії моделі.

По-третє, можна вивести зображення поверхні відгуку. Насправді, зрозуміло, ця поверхня являє собою  $N+1$ -мірний об'єкт, де  $N$  - число вхідних елементів, а вимір, що залишився, відповідає висоті точки на поверхні. Зрозуміло, що безпосередньо візуально представити таку поверхню при  $N$  більшому двох неможливо (а реально  $N$  завжди більше двох). Проте, у пакеті Statistica Neural Networks можна виводити зрізи поверхні відгуку по будь-яким двом вхідним змінним. При цьому значення всіх інших вхідних змінних фіксуються, і міняються тільки два обрані. Всім іншим змінним можна додати будь-яке значення за своїм розсудом (за замовчуванням система Statistica Neural Networks візьме для них середні значення). Значення

двох досліджуваних змінних можна змінювати в довільному діапазоні (за замовчуванням - у діапазоні зміни навчальних даних).

У задачах аналізу часових рядів метою є прогноз майбутніх значень змінної, залежної від часу, на основі попередніх значень її та/або інших змінних.

Як правило, прогнозована змінна є числовою, тому прогнозування часових рядів - це окремий випадок регресії. Однак таке обмеження не закладене в пакет Statistica Neural Networks, так що в ньому можна прогнозувати і часові ряди номінальних змінних.

Звичайно чергове значення часового ряду прогнозується по деякому числу його попередніх значень (прогноз на один крок уперед у часі). У пакеті Statistica Neural Networks можна виконувати прогноз на будь-яке число кроків. Після того, як обчислене чергове передбачуване значення, воно підставляється назад і з його допомогою (а також попередніх значень) виходить наступний прогноз - це називається проєкцією часового ряду. У пакеті Statistica Neural Networks можна здійснювати проєкцію часового ряду і при покроковому прогнозуванні. Зрозуміло, що надійність такої проєкції тим менше, чим більше кроків уперед ми намагаємося пророчити. У випадках, коли потрібно зовсім визначена дальність прогнозу, розумно буде спеціально навчити мережу саме на таку дальність.

У пакеті Statistica Neural Networks для рішення задач прогнозу часових рядів можна застосовувати мережі всіх типів (тип мережі повинний підходити, у залежності від задачі, для регресії чи класифікації). Мережа конфігурується для прогнозу часового ряду установкою параметрів Часове вікно - Steps та Обрій - Lookahead.

Параметр Часове вікно задає число попередніх значень, які варто подавати на вхід, а параметр Обрій указує, як далеко потрібно будувати прогноз. Кількість вхідних і вихідних змінних може бути довільним. Однак, найчастіше в якості вхідної й одночасно (з урахуванням обрію) вихідної виступає єдина змінна.

При конфігуруванні мережі для аналізу часових рядів змінюється метод передобробки даних (витягаються не окремі спостереження, а їхні блоки), але навчання і робота мережі відбуваються точно так само, як і в задачах інших типів.

## **Додаток В**

### **Задачі для побудови нейромоделей**

#### **В.1 Діагностика лопаток газотурбінних авіадвигунів**

Високі вимоги до якості і надійності виробів пред'являються в моторобудуванні. Відмовлення авіаційного двигуна в польотних умовах, зв'язаний з їхнім руйнуванням, може привести до катастрофічних наслідків. Тому важливо вчасно виявити й усунути дефекти і причини їхнього виникнення в процесі експлуатації двигунів літальних апаратів.

Найбільш навантаженими деталями газотурбінних двигунів є лопатки. Найкраща працездатність навантажених деталей газотурбінних двигунів забезпечується при сполученні високих границь текучості, тривалій міцності і високій деформаційній здатності (пластичності) матеріалу. Однак у більшості випадків таке сполучення міцносних і пластичних властивостей не може бути досягнуто.

У зв'язку з тим, що лопатки працюють при значних вібраційних навантаженнях, основною вимогою до їх матеріалів є високий опір утоми.

Тріщини в лопатках турбіни, що виникають у процесі експлуатації, є одним з найбільш характерних і розповсюджених дефектів. Цей вид дефекту, як правило, з'являється і розвивається протягом визначеного тимчасового проміжку в процесі вироблення двигуном його ресурсу. Це дає реальну можливість здійснювати діагностичні заходи для своєчасного виявлення дефектних лопаток. Однак, звичайно ця процедура, здійснювана традиційними методами і засобами, є дуже трудомісткої для обслуговуючого персоналу. Тому автоматизація процесу виявлення дефектних лопаток украй важлива для спрощення і прискорення процесу обслуговування авіадвигунів, підвищення надійності їхнього діагностування.

Одним з можливих методів діагностування є метод виміру параметрів вільних загасаючих коливань лопаток у процесі їх широкополосного імпульсного збудження шляхом простукування. Для визначення дефектів збираються дані - спектри вільних загасаючих коливань лопаток після ударного збудження або різниці напівперіодів загасаючих коливань лопаток після ударного збудження. На основі

цих даних необхідно вміти здійснювати класифікацію лопаток на групи кондиційних і дефектних.

Для дослідження взаємозв'язку між параметрами і класом лопаток були проведені експерименти, у результаті яких були отримані набори значень параметрів усереднених спектрів потужності вільних коливань для кондиційних лопаток і лопаток із тріщинами.

Експерименти проводилися для робочих лопаток першої ступіні турбіни високого тиску двигуна. Спектри вільних загасаючих коливань містили 10240 спектральних ліній у частотному діапазоні до 25000 Гц, з кроком по частоті 2.44 Гц. Значення спектрів характеризують усереднений частотний склад вільних загасаючих коливань у виді спектральної щільності потужності амплітуд виброприскорення.

Формалізувавши задачу маємо:  $10^9$  вхідних ознак, що містять параметри коливань лопатки та 1 цільову ознаку - номер класу лопатки (1 - дефектна, 1 - гідна).

## **В.2 Прогнозування надійності виробів електронної техніки**

Вироби електронної техніки (ОСТ 11 ОДО.052.005), далі ВЕТ, являють собою ненадійні вироби, що можуть бути охарактеризовані наступними параметрами:  $x_1$  – струм сітки першої зворотний, мкА;  $x_2$  – струм сітки першої зворотний при перегарті, мкА;  $x_3$  – струм анода, мА;  $x_4$  – струм сітки другий, мА;  $x_5$  – крутість характеристики, мА/В;  $x_6$  – крутість характеристики при недожару, мА/В;  $x_7$  – струм розжарення, мА;  $x_8$  – ємність вхідна, пф;  $x_9$  – ємність вихідна, пф;  $x_{10}$  – ємність прохідна, пф;  $x_{11}$  – час готовності, с;  $x_{12}$  – коефіцієнт підсилення в тріодному включенні;  $x_{13}$  – напруга контактної різниці потенціалів, В;  $x_{14}$  – відносна зміна струму анода, %;  $x_{15}$  – струм емісії при недожару, А;  $x_{16}$  – опір оксидного шару, Ом;  $x_{17}$  – струм емісії, А;  $x_{18}$  – відносна зміна крутості характеристики після іспитів на довговічність 5000 год., %.

Будучи піддані навантаженню в процесі експлуатації ці вироби можуть виходити з ладу в різний час. Задачею контролю якості цих виробів при завершенні виробництва, а також в експлуатації є прогнозування ресурсу (терміну служби) ВЕТ. Причому на практиці цілком достатньо якісного прогнозу, тобто досить спрогнозувати, чи зможе конкретний виріб прослужити заданий чи термін ні. Для побудови моделі якісної залежності між параметрами і класом ВЕТ можна використовувати експериментально отримані і пронормовані дані.

Формалізувавши задачу маємо: 18 вхідних ознак, що містять параметри ВЕТ 1 цільову ознаку - номер класу ВЕТ (1 - дефектний, 0 - гідний).

### **В.3 Прогнозування надійності трансформаторів**

Розглянемо задачу прогнозування довговічності трансформатора по двох ознаках. З попередніх досліджень і аналізу фізичних причин відмовлення, випливає, що трансформатори відносяться до навантажених електрорадіо-елементів.

Їхні відмовлення в основному визначаються тепловим старінням ізоляційних матеріалів, використовуваних у конструкції, тому як першу ознаку для прогнозування можна запропонувати величину температури перегріву  $T$ , стосовно навколишнього середовища за рахунок втрат у міді і стали, тому що перегрівши погіршує тепловий режим, а значить знижує довговічність трансформатора. За другу ознаку може бути узятий опір ізоляції трансформатора  $R$ . Результати попередніх іспитів показали, що початкове значення  $R$  є малоінформативним для прогнозування. Опір ізоляції значно змінюється від одного циклу іспитів до іншого, погіршуючи іноді в  $10^3$ - $10^4$  разів. Це свідчить про процеси старіння, що протікають, ізоляційних матеріалів. Отже, інформативною ознакою, що характеризує довговічність трансформатора може служити зміна опору чи ізоляції значення опору ізоляції через визначений інтервал часу іспитів. Однак цей інтервал з погляду витрат на прогнозування не повинний бути значним.

Для виявлення можливості прогнозування по обраних ознаках був виконаний навчальний експеримент. До класу 0 віднесені трансформатори, що відмовили після 10 циклу (тривалість кожного циклу 500 годин), тобто мають нормальну надійність. До класу 1 віднесені трансформатори, що відмовили до 10 циклу, тобто мають знижену надійність.

Формалізувавши задачу маємо: 2 вхідних ознаки, що містять параметри трансформаторів та 1 цільову ознаку - номер класу надійності трансформатору (1 - ненадійний, 0 - гідний).

#### **В.4 Моделювання параметрів сендастових магнітних голівок**

Одними з найбільш значимих показників якості магнітних голівок є їхні акустичні характеристики. До основних акустичних характеристик магнітних голівок із сендастовими сердечниками відносять індуктивності лівого і правого каналів, електрорушійну силу (віддачу) відтворення лівого і правого каналів і амплітудно-частотні характеристики лівого і правого каналів.

Як відомо, у процесі магнітного звукозапису власне записуючим елементом є протяжне і нерівномірне розподілене поле голівки в області її робочого зазору, утвореного двома сендастовими сердечниками. Просторовий розподіл напруженості зовнішнього магнітного поля голівки, що дає можливість робити звукозапис крім струму в її обмотці залежить ще і від величини і форми робочого зазору між сердечниками, форми полюсних наконечників сердечника поблизу зазору, магнітної проникності і товщини робочого шару носія запису.

У процесі відтворення зовнішній магнітний потік стрічки замикається через сердечник голівки і індукє у ній електрорушійну силу, що змінюється пропорційно швидкості зміни магнітного потоку біля робочого зазору.

У такий спосіб можна вважати, що основні акустичні характеристики магнітних голівок визначаються параметрами сендастового сердечника, які умовно можна розділити на два класи: магнітні властивості сендаста і геометричні параметри сердечника.

Магнітні властивості сендаста, як правило, є однаковими для всіх голівок партії, не залежать від геометричних параметрів сердечника і характеризуються індукцією насичення, магнітною проникністю і коефіцієнтом сили.

Однак, заливання компаунда робить деякий вплив на індуктивності каналів, тому на практиці необхідно для кожної голівки контролювати індуктивності лівого і правого каналів до і після заливання компаунда.

Більшість геометричних параметрів сердечника мають, як правило, незначні відхилення від норми. Однак, особливу увагу при виробництві голівок варто приділяти тим геометричним параметрам, найменша зміна яких здатна впливати на акустичні характеристики магнітних голівок. Серед геометричних параметрів сендастових сердечників будемо виділяти відносну висоту робочої зони після дове-

дення і шліфування і товщину робочої зони сердечника, що характеризується шириною лівого і правого каналів.

Геометричні параметри сердечника й індуктивності каналів можна виміряти ще до заливання компаунда після зборки магнітної голівки, а оскільки вони впливають на акустичні властивості голівки, то виникає необхідність прогнозування з чисельною оцінкою акустичних характеристик голівки за значеннями цих параметрів.

Таке прогнозування дозволить здійснювати діагностику магнітних голівок ще в процесі виробництва комплектуючих, що здатне на практиці принести визначений прибуток за рахунок скорочення браку і виробничих операцій по виготовленню голівок з недостатньо якісних комплектуючих.

Задача прогнозування значень акустичних характеристик магнітних голівок за значеннями параметрів сендастового сердечника по суті представляє собою задачу апроксимації багатовимірної залежності.

У якості вихідних даних для моделювання використовувалися значення вищеописаних параметрів і акустичних характеристик магнітних голівок ЗД24.310 ДМЕ 3.253.000, у яких використовуються сендастові сердечники ДСВ 6.653.002.

Сендаст, з якого були виготовлені сердечники, характеризувався такими значеннями параметрів: магнітна проникність - 25000 ; індукція насичення - 0,93 Тл, коерцитивна сила 2,6 А/м.

При зборці магнітної голівки до заливання компаунда здійснювався вимір геометричних параметрів і індуктивностей каналів. Вимір геометричних параметрів здійснювався за допомогою мікрометра МІТUTOYO (Японія). Контроль і вимір індуктивності здійснювався вироблялися на контрольно-вимірювальному приладі WAYNEKERR (Німеччина). Вимір акустичних характеристик здійснювався на устаткуванні 50062 фірми Ferrotronic (Німеччина).

При випробуванні магнітної голівки на віддачу використовується стандартний підсилювач, що утворює лінійну характеристику. На магнітну стрічку еталонною голівкою записуються сигнали частотою 315 Гц і 12500 Гц. Тривалість сигналів контролюється комп'ютером. Значення вимірів представляються в децибелах. Початком відліку 0 d вважається віддача на опорній частоті 315 Гц. відхилення на частоті 12500 Гц повинні бути не більш 2 d, причому різниця двох каналів не повинна складати більш 2 d.

При контролі амплітудно-частотної характеристики відтворення магнітної голівки на контрольну стрічку еталонною магнітною голівкою записуються сигнали частотою 315 Гц і 14 кГц. У залежності від значення амплітудно-частотної характеристики відтворення на частоті 14 кГц щодо частоти 315 Гц магнітну голівку відносять до одного з трьох класів якості (див. табл. В.1).

Таблиця В.1 - Класи якості сендастових голівок

| Найменування класу | Позначення класу | Амплітудно-частотна характеристика на частоті 14 кГц щодо частоти 315 Гц, d |
|--------------------|------------------|-----------------------------------------------------------------------------|
| вищий              | А                | не менш +10                                                                 |
| перший             | В                | не менш +8                                                                  |
| другий             | С                | не менш +6                                                                  |

Експериментально отримані дані містять наступні позначення:  $x_1$  - Індуктивність лівого каналу до заливання компаундом, мГн;  $x_2$  - Індуктивність правого каналу до заливання компаундом, мГн;  $x_3$  - Відносна висота, мм;  $x_4$  - Ширина лівого каналу,  $10^{-3}$  мм;  $x_5$  - Ширина правого каналу,  $10^{-3}$  мм;  $y_1$  - Віддача відтворення лівого каналу, d;  $y_2$  - Віддача відтворення правого каналу, d;  $y_3$  - АХЧ відтворення лівого каналу, d;  $y_4$  - АХЧ відтворення правого каналу, d;  $y_5$  - Індуктивність лівого каналу після заливання компаундом, мГн;  $y_6$  - Індуктивність правого каналу після заливання компаундом, мГн.

Формалізувавши задачу маємо 5 вхідних ознак та 6 вихідних ознак

## В.5 Моделювання стрічкопротягувального механізму

Задачею стрічкопротягувального механізму (СПМ) є транспортування стрічки з заданою номінальною швидкістю. Однак, через вплив дестабілізуючих факторів дійсна швидкість відрізняється від номінального значення (причому, відхилення може бути постійним і змінним). При звукозаписі коливання швидкості стрічки приводять до миттєвих змін висоти струму, називаними детонацією. СПМ характеризують коефіцієнтом детонації, що представляє собою коефіцієнт

коливань швидкості, обмірюваний при умовах оцінки, що відповідає сприйняттю детонації людським вухом.

Аналіз факторів, що збурюють коливання миттєвої швидкості транспортування магнітної стрічки в непрямому приводі СПМ серійних магнітофонів, дозволяє зробити висновок, що джерелами коливань швидкості стрічки є вузол привода ведучого вала і вузол транспортування магнітної стрічки.

Метою досліджень є побудова моделі СПМ, що дозволить на основі значень заданих параметрів визначати значення коефіцієнта детонації і токів споживання.

На підставі приведеного аналізу до зборки магнітофонів були виконані виміри ряду характеристик елементів і вузлів трьох серійних СПМ. Потім були проведені експериментальні дослідження коефіцієнта детонації та токів споживання магнітофонів.

Експериментально отримані дані містять наступні позначення:  $x_1$  – биття канавки маховика ведучого вала, мм;  $x_2$  – биття осі ведучого вала, мм;  $x_3$  – биття притискного ролика (у зборі з втулкою, без важеля – неповний складальний вузол), мм;  $x_4$  – детонація приводного ремня на вертикальній грані, %;  $x_5$  – детонація приводного ремня на горизонтальній грані, %;  $x_6$  – зусилля натягу приводного ремня, Н;  $y_1$  – коефіцієнт детонації, %;  $y_2$  – струм споживання, А.

## **В.6 Розпізнавання сільськогосподарських рослин**

Класифікація рослин за результатами дистанційного зондування на культурні і бур'яни, хворі та здорові, по біологічних видах використовується при автоматизованій дешифрації супутникових і аерофотознімків, визначенні засміченості сільськогосподарських угідь, ступеня поразки лісових насаджень.

Нова технологія обробки просапних культур у рослинництві також використовує процедуру розпізнавання класів рослин. За цією технологією прополка міжрядь здійснюється звичайним шляхом, а в рядку кожна рослина спочатку відноситься до свого класу (культурна або бур'ян) і після цього обробляється відповідним чином. Можливість такого поділу визначається біологічними особливостями різних видів рослин, що при дистанційному методі зондування виявляються в спектрах відображення, поглинання і люмінесценції. При цьому висувається допущення, що інформація про приналежність об'єктів до того

чи іншого класу має радіометричні прояви -«сигнатури». Приклад сигнатури — це відображення, як функція довжини хвилі, прийняте багатодіапазонними датчиками. У цьому випадку кожна вимірювана величина, що відповідає спектральному коефіцієнту яскравості (СКЯ), є ознакою. Тоді образ можна представити у виді деякого вектора, що складає з числових характеристик.

Для поділу рослин таким вектором може слугувати СКЯ у видимій і ближній інфрачервоній частині спектра. При дешифруванні типів земного покриття з космосу найбільш інформативними є такі спектральні смуги: 440...460 нм, 620...660 нм, 800...820нм. Для характеристики форми кривої СКЯ рекомендують використовувати відносини коефіцієнтів яскравості. По експериментальним даним специфічними смугами спектра при розпізнаванні природних об'єктів є: 550...560, 660...680 і 800...820 нм. Залежність коефіцієнта відбиття від рослини, як функція довжини хвилі, у діапазоні 350...850 нм визначається пігментною системою листів, товщиною, щільністю й анатомією листової пластинки. Ці характеристики є індивідуальними для різних видів рослин. Показано, що внесок хлорофілу і каротиноїдів у поглинання і відображення променистої енергії найбільше відчувається у видимому діапазоні (350...750 нм). При цьому роль каротиноїдів істотна в діапазоні 350...550 нм, а для довжин хвиль більше 550 нм основну роль грає хлорофіл. У діапазоні спектра 750 нм і вище рослина характеризується високим невідбирчим відбиттям і невеликим поглинанням променистої енергії. На довжинах хвиль великих 1200 нм зростає поглинання водою (до 96-98% на 3 мкм), а відбиття зменшується. Тому, з біологічної точки зору, для розпізнавання найбільш доцільне використання діапазону довжин хвиль від 300 нм до 800 нм.

Для одержання інформації про поведінку СКЯ для різних рослин був проведений ряд експериментальних досліджень. Як об'єкти для дослідження використовувалися: культурні рослини - кукурудза (*Zea mays*), соняшник (*Heliantus cultus*); бур'яни - щириця (*Amaranthus retroflexus*), мишей (*Setaria viridis*), пирій (*Agropyrum repens*), осот (*Sonchus arvensis*), лобода (*Shenopodium album*) , молочай (*Euphorbia falcata*).

У ході експериментів методом дистанційного зондування були отримані характеристики відбиття цих рослин. Для кожного виду рослини проводилося 12 дослідів, що відповідають різним зразкам і різній орієнтації щодо об'єктива експериментальної установки. Основний

вузол установки - монохроматор, що входить у фотоелектричний спектрофотометр "Spescol", забезпечував випромінювання сигналу смугою 12 нм у діапазоні 310...850 нм із кроком у 1нм. Для кожного зразка знімалося 55 спектральних точок у цьому діапазоні з кроком 10 нм.

### **В.7 Прогнозування стану форми спортсменів на основі даних біохімічного контролю тренувальних навантажень**

Метою досліджень було вивчення в ритмобіологічному аспекті динаміки реорганізації метаболічної функціональної системи, що забезпечує структурно-енергетичний гомеостаз, на етапах річного тренувального циклу в академічному веслуванні і з'ясування можливості використання класифікації отриманих даних для прогнозування стану спортсменів.

Основними показниками спортивних характеристик людини є наявні в крові три специфічних речовини: сечовина, лактат і неетерифіковані жирні кислоти (НЕЖК). У процесі добових навантажень концентрація цих речовин у крові змінюється в межах деякого середнього значення. Однак із плином тренувального процесу це середнє значення змінюється.

Метою будь-якого тренувального процесу є не тільки підвищити середньодобове значення концентрації цих речовин, але і домогтися прояву максимальної фази середньодобового коливання безпосередньо під час тренування або змагань.

Для досягнення даної мети кожен тренер має свої власні методи тренувань. Ефективність методики цілком можна перевірити по графіках зміни вмісту відповідних речовин у крові, або просто довідавшись - чи витримав спортсмен весь цикл тренувань і чи продемонстрував високі результати на змаганнях, або був відрахований з команди в зв'язку з відсутністю успіхів.

Однак, можливо, існує спосіб визначення перспектив спортсмена і методології тренувань ще при перших аналізах крові - це дозволило б вчасно вжити заходів - змінити методику тренувань чи відправити спортсмена відпочивати, тим самим уникнувши зайвих витрат.

Біохімічний контроль здійснювали на етапах виконання річної тренувальної програми в умовах учбово-тренувальних зборів.

У капілярній крові визначали рівень сечовини - показник інтенсивності участі в роботі силового компонента, лактату - швидкісного і НЕЖК – витривалості.

Кров відбирали у вільний від тренувань день з 8 ранку до 8 вечора кожні 4 години. Час початку і закінчення добору крові визначалося розподілом тренувань протягом дня. Інтервал часу - між заборами крові 4 години - узяті нами як максимальний, що об'єктивно відбиває біоритм системи з періодом, рівним 12 і 24 години.

Дослідження проведені на групі дівчин 16-17 років, майстрів спорту. Тренувальна програма передбачала підготовку спортсменок для проходження дистанції 1500 м у межах 5-6 хвилин. Програма реалізувалася протягом листопаду - липня.

Формалізувавши задачу, маємо: 12 ознак (3 змінних по 4 значення) і 2 класи (готовий і не готовий).

## **В.8 Моделювання біогазової установки**

Інтенсивне збільшення енергетичних потреб людства в умовах обмежених запасів традиційно використовуваних видів палива висуває на перший план пошук і використання альтернативних джерел енергії, і, насамперед, поновлюваних.

Одним з таких джерел енергії є біогаз (метан), що утворюється в результаті розкладання органічних речовин у відходах тваринництва і рослинництва. Виділення біогазу здійснюється в спеціальних біогазових установках. Застосування таких установок доцільно як у місцях важкодоступних для доставки палива, так і поблизу сільськогосподарських підприємств, особливо тваринницьких ферм.

Утворення біогазу з органічних речовин — результат діяльності збалансованого анаеробного мікробної спільноти, що включає різні групи мікроорганізмів, які здійснюють поетапну деградацію складних органічних сполук.

Донедавна основні зусилля були зосереджені на вивченні метаногенних спільнот, що функціонують у мезофільному (33—40°) і термофільному (50—55°) режимах, що дозволяло застосовувати біогазові установки в основному в країнах з теплим кліматом.

У країнах же з помірним і холодним кліматом у зимовий період потрібно здійснювати теплоізоляцію і підігрів біореакторів, що істот-

но збільшує вартість біогазових установок і ускладнює їхню експлуатацію.

Відомо, що в природі значне місце займає утворення біогазу в системах, що знаходяться при більш низьких температурах. В останні роки широко вивчаються також можливості збражування відходів тваринництва при температурі навколишнього середовища і фактори, що впливають на утворення біогазу в цих умовах. Вважається, що збражування відходів тваринництва при температурах навколишнього середовища є більш дешевим, тому що скорочуються витрати на нагрівання.

Досліджувалася динаміка розкладання органічних залишків мікрофлорою при температурі 6—28°. Накопичувальна ємність, що містила органічні залишки була заглиблена на 6 м, на дні її температура узимку не опускається нижче 4°, а влітку не перевищує 20°. Вологість органічних залишків складала: стоки - 97—99%, осад - 90—94%. При цьому зміст сухої речовини в середовищі складало 10 г/л, а концентрація органічних речовин була 7—8 г/л. рН середовища в пробах вимірювали потенціометрично. Гази, що утворюються, визначали методом газової хроматографії на хроматографі «Chrom-5» з катарометром.

Отримані експериментальні дані мають невеликий обсяг, є точковими і містять значні пробіли, тому виникає необхідність апроксимації логістичної залежності обсягу утворення біогазу (ммоль/л) від часу, що пройшов з початку збражування (доб.), і температури навколишнього середовища (°C).

## **В.9 Моделювання сумарного показника якості життя хворих на хронічний бронхіт**

На теперішній час у медичних дослідженнях усе частіше фігурує поняття "якість життя". Це поняття інтегрує велику кількість фізичних, психологічних, соціальних, економічних характеристик хворого і відбиває його здатність адаптуватися до проявів хвороби. Однак у країнах СНД є вкрай мало повідомлень, переважно оглядового характеру, з вивчення якості життя (ЯЖ) хворих на хронічний обструктивний бронхіт (ХОБ). Дискусійним і дотепер невирішеним залишається питання про взаємозв'язок між ЯЖ пацієнтів та їх соматичним статусом, вплив ваги й особливостей перебігу хвороби на ЯЖ хворого.

Тому актуальним є створення комп'ютерної моделі, що дозволяє оцінювати взаємозв'язок сумарного показника якості життя (СПЯЖ) хворих ХОБ із клінічними і лабораторно-інструментальними показниками, що характеризують плин основного захворювання.

На основі експериментальних досліджень складено вибірку для хворих ХОБ у фазі загострення захворювання. Середній вік пацієнтів складав  $40 \pm 3$  років, чоловіків - 57%, жінок - 43%.

Оцінка соматичного статусу пацієнтів ґрунтувалася на наступних 47 клінічних, лабораторно-інструментальних показниках:  $x_1$ - обсяг форсованого видиху (ОФВ 1) за 1 сек, у % від належного;  $x_2$ - пікова об'ємна швидкість (ПОШ), у % від належного;  $x_3$ - середньодинамічний тиск у легеневої артерії (СДЛА), мм рт. ст.;  $x_4$ - число серцевих скорочень у 1 хв.;  $x_5$ -тривалість інтервалу S-T у II стандартному відведенні на електрокардіограмі (ЕКГ), у сек;  $x_6$ - вік пацієнтів, у роках;  $x_7$  - стать (жіночий – 0, чоловічий – 1);  $x_8$  - характер праці (фізичний – 0, розумовий – 1);  $x_9$  - тривалість лікування в стаціонарі (дні);  $x_{10}$  - тривалість хвороби, років;  $x_{11}$  - тривалість паління, років;  $x_{12}$  - вживання алкоголю (немає – 0, є – 1);  $x_{13}$  - наявність алергійних реакцій в анамнезі (немає – 0, є – 1);  $x_{14}$ -перенесені пневмонії (немає – 0, є – 1);  $x_{15}$ -часті простудні захворювання (ОРВІ) в анамнезі (немає – 0, так – 1);  $x_{16}$ -наявність ішемічної хвороби серця (ІХС): немає – 0, є – 1;  $x_{17}$ -наявність гіпертонічної хвороби (ГХ): немає – 0, є – 1;  $x_{18}$ -наявність захворювань шлунково-кишкового тракту (ШКТ): немає – 0, є – 1;  $x_{19}$ -наявність хвороб ЛОР-органів (немає – 0, є – 1);  $x_{20}$ -наявність емфіземи, пневмосклерозу (немає – 0, є – 1);  $x_{21}$ -наявність бронхоектазів (немає – 0, є – 1);  $x_{22}$ -особливі стани (після лобектомії, після туберкульозу легень і ін.): немає – 0, є – 1;  $x_{23}$ -наявність вторинної легеневої гіпертензії (ВЛГ), хронічного легеневого серця (ХЛС): немає – 0, є – 1;  $x_{24}$ -форсована життєва ємність легень, у % від належного;  $x_{25}$ -миттєва об'ємна швидкість (МОШ) на рівні великих бронхів, у % від належного;  $x_{26}$ -МОШ на рівні середніх бронхів, у % від належного;  $x_{27}$ -МОШ на рівні дрібних бронхів, у % від належного;  $x_{28}$ -індекс Тиффно (відношення ОФВ 1 до ФЖЕЛ), у %;  $x_{29}$ -тривалість зубця Р у II стандартному відведенні на ЕКГ, у сек.;  $x_{30}$ -індекс Макруза по ЕКГ;  $x_{31}$ -тривалість зубця Т у II стандартному відведенні на ЕКГ, у сек.;  $x_{32}$ -різниця між висотою зубця R і зубця Т в I відведенні на ЕКГ, у мм;  $x_{33}$ -різниця між висотою зубця Т в V1 і V6 відведеннях на ЕКГ;  $x_{34}$ -висота зубців Т в грудних відведеннях на ЕКГ менше норми (немає – 0, є – 1);  $x_{35}$ -наявність аритмії на ЕКГ (немає – 0, є – 1);  $x_{36}$ - систолічний артеріальний тиск (САТ), у мм рт. ст.;  $x_{37}$ -діастолічний артеріальний тиск (ДАТ), у мм рт. ст.;  $x_{38}$ -ударний обсяг крові, у мол ;  $x_{39}$ -хвилинний об-

сяг крові, у л/хв.;  $x_{40}$ -абсолютне число еритроцитів крові,  $\cdot 10^9$ / л;  $x_{41}$ -кількість гемоглобіну крові, г/л;  $x_{42}$ -абсолютне число лейкоцитів крові,  $10^9$ / л;  $x_{43}$ -абсолютне число паличкоядерних лейкоцитів крові;  $x_{44}$ -абсолютне число лімфоцитів крові;  $x_{45}$ -швидкість осідання еритроцитів (ШОЕ), мм/год.;  $x_{46}$ -характер запального процесу в бронхах (катаральний – 0, гнійний – 1);  $x_{47}$ -стадія бронхіту; у-сумарний показник якості життя (СПЯЖ), у.о.

Для оцінки ЯЖ досліджуваних хворих використовували російською мовою версію "Респіраторної анкети лікарні Святого Георгія (РАСГ). Відповідно до вимог анкети розраховували СПЯЖ, виражений в умовних одиницях і позначений умовно в ході моделювання як "у".

## **В.10 Прогнозування курсу валюти**

Динамічний характер економічних та соціально-політичних процесів у сучасному світі ускладнює розвиток підприємств, а в умовах жорсткої конкурентної боротьби вимагає прийняття рішень при мінімумі інформації. Особливо складним є прийняття рішень щодо майбутніх ситуацій.

До показників, що характеризують ситуацію на ринку можна віднести біржові індекси, курси валют та акцій. Передбачення значень цих показників у майбутньому дає змогу заздалегідь прийняти правильне рішення.

Нехай ми маємо спостереження за курсом валюти або акцій протягом певної кількості днів -  $D$ . Необхідно, знаючи курс валюти (акцій) за декілька днів, спрогнозувати курс у наступний день.

Навчальну вибірку даних створимо із застосуванням методу вікон. Задамо ширину вікна  $N$  - кількість днів у які курс відомий, а також довжину прогнозу  $L$  - відстань між поточним днем та днем прогнозу. Як перший екземпляр вибірки візьмемо  $N$  перших спостережень, котрі стануть значеннями ознак, а як значення цільової ознаки візьмемо  $(N+L)$ -те спостереження. Сформуємо подібним чином й інші екземпляри, взявши спостереження з номерами  $[i, (i+1), \dots, (i+N-1)]$  як ознаки, а  $(i+N-1+L)$ -те спостереження як значення цільової ознаки для відповідного екземпляра, де  $i=1, \dots, (D-N-L)$ . Усього буде сформовано  $S$  екземплярів.

У результаті отримаємо вибірку, що буде складатися з  $S$  екземплярів, які характеризуються  $N$  ознаками кожний. Кожному екземпляру співставлене одне значення цільової ознаки.