

## Лабораторна робота №9

### Штучні нейронні мережі. Прогнозування часових рядів

**Мета:** Отримати початкові навички по створенню штучних нейронних мереж, що здатні прогнозувати часові ряди.

### Теоретичні відомості

#### *Прогнозування часових рядів за допомогою нейрону з сигмоїдальною функцією активації*

Задача прогнозування часових рядів, в яких є певні закономірності, може бути вирішена за допомогою нейромережі, яка може навчатися. Відомо, що людський мозок здатний до самонавчання, причому досягає успіхів найчастіше, не знаючи природи процесів, що лежать в основі виконуваних дій.

Наприклад, щоб потрапити м'ячем у баскетбольне кільце, робот-баскетболіст повинен виміряти відстань до кільця й напрямок, розрахувати параболічну траєкторію, і зробити кидок з урахуванням маси м'яча й опору повітря. Людина ж обходиться без цього тільки через тренування. Багаторазово здійснюючи кидки й спостерігаючи результати, вона коректує свої дії, поступово вдосконалюючи свою техніку. При цьому в її мозку формуються відповідні структури нейронів, відповідальні за техніку кидків.



Рисунок 1.9 – Алгоритм навчання нейронних мереж

**1. Вибір структури нейромережі,** це складна задача. В даній лабораторній роботі візьмемо мережу, що складається з одного нейрону, зображеного на рисунку 1.10.

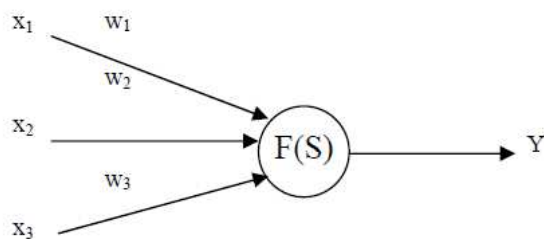


Рисунок 1.10 – Штучний нейрон для прогнозування значень часового ряду

Зважена сума та функція активації даного нейрона:

$$S_i = x_{i-3} \cdot w_1 + x_{i-2} \cdot w_2 + x_{i-1} \cdot w_3, \quad (1.9)$$

$$Y_i = 1/(1+\exp(-S_i)) \cdot 10, \quad (1.10)$$

де  $w_1, w_2, w_3$  – синаптичні ваги;

$x_{i-3}, x_{i-2}, x_{i-1}$  – вхідні сигнали – відомі попередні значення часового ряду ( $i$ -й набір вхідних даних);

$S_i$  – зважена сума  $i$ -го набору вхідних даних;

$Y_i$  – прогнозоване значення  $i$ -го члена часового ряду  $x_i$ ;

10 – масштабний множник.

**2. Навчання** полягає в тому, що на вхід мережі подаються спеціальні тренувальні дані, тобто такі вхідні дані, вихідний результат для яких відомий. На виході формуються результуючі дані, результати порівнюються з очікуваними, і обчислюється значення помилки. Після цього в певній послідовності виконується корекція параметрів нейронної мережі із метою мінімізації функції помилки. Якщо задовільної точності досягти не вдається, варто змінити структуру мережі й повторити навчання на множині тренувальних даних.

Таблиця 1.5 – Приклад тренувальних даних для нейронної мережі, що здійснює прогнозування значень часового ряду

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	$x_{13}$	$x_{14}$	$x_{15}$
1,59	5,73	0,48	5,28	1,35	5,91	0,77	5,25	1,37	4,42	0,26	4,21	1,90	4,08	1,40

Перші 13 чисел будемо використовувати для навчання мережі як тренувальний набір даних. Останні два члени ряду в навчанні не будуть брати участь, а служитимуть для тестування мережі.

Прогнозування полягає в тому, щоб на основі  $x_i, x_{i+1}, x_{i+2}$  обчислити  $x_{i+3}$ .

Іншими словами, нейронна мережа «ковзає» уздовж часового ряду, «обмацуючи» синапсами по три сусідніх числа, та намагається прогнозувати значення наступні за ними. Таким чином, для наведеного вище прикладу вхідними й вихідними величинами будуть наступні (див. табл. 1.6).

Таблиця 1.6 – Очікувані значення часового ряду на кожному кроці навчання

i-тий набір даних	Вхід нейрона	Вихід (очікуваний результат)
1	1,59 5,73 0,48	5,28
2	5,73 0,48 5,28	1,35
3	0,48 5,28 1,35	5,91
4	5,28 1,35 5,91	0,77
	і т.д.	

**Навчання нейронної мережі полягає в** знаходженні таких значень ваг  $w$ , при яких нейромережа буде здатна видавати на основі вхідних даних вірні вихідні дані з певною наперед заданою точністю.

Дана задача задовільно вирішується за допомогою **алгоритму зворотного поширення (*back propagation*)**, що полягає в наступному:

1) Спочатку всі вагові коефіцієнти нейронної мережі встановлюються довільно. Можна скористатися функцією `random`, або просто присвоїти всім ваговим коефіцієнтам 1.

2) Через мережу пропускаються тренувальні дані (перший набір вхідних даних), і обчислюється сумарна функція помилки (сума квадратів помилки):

$$E = \sum_{i=1}^N (Y_i - y_i)^2, \quad (1.11)$$

де  $Y_i$  – обчислене значення виходу нейрона;

$y_i$  – правильне значення наступного члену часового ряду.

3) Обчислюється значення похідної функції помилки  $E'_t$  для кожного вагового коефіцієнта:

$$E'_i = (Y_i - y_i) \cdot (\exp(-s_i) / (1 + \exp(-s_i))^2) \cdot x_i \quad (1.12)$$

4) На основі  $E'_{i1}$ ,  $E'_{i2}$ , та  $E'_{i3}$ , здійснюється розрахунок виправлень  $\Delta w_{i1}$ ,  $\Delta w_{i2}$ , та  $\Delta w_{i3}$  до відповідних вагових коефіцієнтів за наступною формулою:

$$\Delta w_i = -v \cdot E'_i \quad (1.13)$$

де  $v$  – коефіцієнт швидкості навчання. Виправлення необхідно знайти для кожного  $i$ -го набору вихідних даних, і обчислити середні значення  $\Delta w_{\text{середнє}1}$ ,  $\Delta w_{\text{середнє}2}$ , та  $\Delta w_{\text{середнє}3}$  для всього набору:

$$\Delta w_{\text{середнє}} = \frac{1}{N} \sum_{i=1}^N \Delta w_i, \quad (1.14)$$

де  $N$  – кількість наборів вхідних даних для навчання.

5) Вагові коефіцієнти коректуються на величину обчислених виправлень:

$$w = w + \Delta w_{\text{середнє}}, \quad (1.15)$$

6) Поточне значення сумарної функції помилки  $E$  зберігається в іншій змінній:

$$E_0 = E. \quad (1.16)$$

Кроки 2–5 алгоритму зворотного поширення повторюються для кожного  $i$ -того набору вхідних даних (назвемо це *цикли навчання*), поки функція помилки не знизиться до заданого рівня, наприклад:

$$|E - E_0| < 0,0001 \quad (1.17)$$

Кількість ітерацій у процесі навчання мережі може досягати сотень і навіть тисяч. Тому доречно зробити додаткову умову виходу із циклу, на випадок якщо навчання з заданим рівнем точності буде тривати не примусово довго, або відбудеться зациклення. Додатковою умовою виходу може бути натиснення користувачем кнопки "Стоп", або досягнення певної кількості циклів навчання, наприклад:  $i > 1000000$ .

**3. Тестування**, тобто контроль точності на спеціальних тестових даних, виконується після того, як нейронна мережа навчена. Це означає, що всі дані варто розбити на дві підмножини: на першій з них виконується навчання мережі, а на другій - тестування. За аналогією з навчанням людини тестування можна вподібнити іспиту. В нашому випадку для тестових даних ми залишили визначення нейронною мережею чисел  $x_{14}$  та  $x_{15}$  нашого часового ряду.

### Завдання:

Написати програму для реалізації штучних нейронів та нейронних мереж для прогнозування часового ряду (варіанти часових рядів див. в таблицю А до лабораторної роботи).

### Контрольні питання:

1. З яких основних блоків складається алгоритм навчання нейронних мереж? Як ці блоки пов'язані між собою?
2. В чому заключається алгоритм зворотного поширення помилки?
3. Яким чином можна налаштувати нейронну мережу на прогнозування значень величин часового ряду?

Варіанти часових рядів

Таблиця А

№ варіанту	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	x <sub>4</sub>	x <sub>5</sub>	x <sub>6</sub>	x <sub>7</sub>	x <sub>8</sub>	x <sub>9</sub>	x <sub>10</sub>	x <sub>11</sub>	x <sub>12</sub>	x <sub>13</sub>	x <sub>14</sub>	x <sub>15</sub>
1	2,56	4,20	1,60	4,29	1,17	4,40	0,88	4,14	0,07	4,77	1,95	4,18	0,04	5,05	1,40
2	0,20	5,14	0,47	4,37	1,22	4,29	1,89	4,51	0,32	5,80	1,37	5,77	0,88	4,86	1,94
3	1,92	4,01	1,48	5,45	1,56	5,42	1,28	4,34	1,51	5,49	1,32	4,00	0,49	4,19	1,53
4	0,13	5,97	0,57	4,02	0,31	5,55	0,15	4,54	0,65	4,34	1,54	4,70	0,58	5,83	0,03
5	2,16	3,19	1,85	4,84	0,55	4,20	1,68	4,74	0,14	5,68	0,48	5,03	0,18	5,99	0,09
6	2,54	5,28	0,78	5,72	0,58	4,65	0,91	5,80	1,76	5,67	1,73	5,70	1,03	5,00	1,79
7	1,69	3,38	1,40	5,56	1,86	5,62	0,46	5,51	0,26	5,13	1,18	5,98	1,36	5,09	1,29

8	1,19	5,61	0,89	6,00	1,04	5,98	0,03	6,00	1,83	4,23	0,60	4,15	0,13	5,01	1,87
9	0,87	4,12	0,93	4,62	1,51	5,76	0,50	5,48	0,95	4,03	0,92	5,15	1,66	5,01	0,40
10	2,82	3,48	0,60	4,76	1,51	5,51	1,48	5,19	0,48	5,22	0,21	4,19	0,07	4,63	0,49
11	2,64	4,66	1,87	4,05	1,73	5,31	1,67	5,96	0,13	5,64	1,52	4,07	0,22	4,79	0,73
12	2,65	5,60	1,21	5,48	0,73	4,08	1,88	5,31	0,78	4,36	1,71	5,62	0,43	4,21	1,21
13	2,37	4,85	1,97	4,17	1,39	4,66	1,26	4,40	0,46	5,54	1,34	5,80	1,61	5,97	1,95
14	1,88	4,52	1,91	5,66	1,23	5,50	1,14	5,29	1,60	4,31	0,06	5,33	0,07	4,62	0,69
15	0,78	4,95	1,19	4,08	0,80	4,25	0,22	4,63	1,48	4,97	0,53	5,50	1,28	5,79	0,44
16	0,58	3,38	0,91	5,80	0,91	5,01	1,17	4,67	0,60	4,81	0,53	4,75	1,01	5,04	1,07
17	0,51	4,82	0,43	4,71	1,92	5,86	1,24	4,69	0,72	5,26	0,90	4,55	1,46	5,21	1,50
18	0,07	3,58	0,44	5,33	0,56	5,24	1,99	4,38	0,89	4,53	1,82	4,13	1,88	5,97	1,18
19	1,44	4,60	1,22	5,90	1,34	4,31	1,02	4,35	0,82	4,18	1,60	4,86	1,45	4,97	1,00
20	2,57	4,35	1,27	5,46	1,30	4,92	1,31	4,14	1,97	5,67	0,92	4,76	1,72	4,44	1,49
21	0,79	3,84	0,92	4,50	0,96	5,51	1,14	5,32	0,39	4,99	1,36	5,81	1,90	4,79	1,41
22	0,99	4,72	1,59	5,29	1,53	5,58	0,84	5,79	0,21	5,94	0,42	5,98	1,18	5,55	0,11
23	2,92	3,56	0,15	5,11	1,38	4,44	1,61	4,11	1,97	4,50	1,37	5,08	1,76	5,19	1,58
24	0,48	4,30	0,91	4,85	0,53	4,51	1,95	5,88	0,63	5,79	0,92	5,18	1,88	4,84	0,22
25	1,88	4,98	0,06	5,26	1,16	5,06	0,58	5,28	1,41	5,57	1,19	5,36	1,40	4,30	0,09
26	2,57	5,77	0,38	4,73	0,10	5,93	1,35	4,70	1,62	5,51	1,78	5,66	1,47	5,52	1,88
27	0,11	4,87	1,52	4,47	0,34	5,44	1,20	5,21	1,48	5,93	0,62	5,48	1,34	4,25	0,65
28	1,07	3,17	1,08	5,99	1,28	4,11	0,25	5,82	0,96	4,83	1,10	4,31	0,81	5,49	1,92
29	1,59	5,74	0,48	5,28	1,34	5,91	0,77	5,25	1,37	4,42	0,26	4,21	1,90	4,08	1,40
30	0,68	5,78	0,25	5,58	1,31	4,28	1,57	5,75	0,41	5,55	0,90	5,86	0,03	5,57	0,30