

УДК 004.415.2

ПРОГНОЗУВАННЯ ПАРАМЕТРІВ ДОВКІЛЛЯ НА ОСНОВІ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ

О. Дзендзелюк, З. Любунь, В. Рабик

*Львівський національний університет імені Івана Франка,
вул. Ген. Тарнавського, 107, 79017 Львів, Україна
RabykV@ukr.net*

Розглянуто загальний алгоритм прогнозування часових рядів на основі штучних нейронних мереж. Цей алгоритм складається з таких етапів: опрацювання вхідних даних, визначення кількості нейронів у прихованому шарі, побудова штучної нейронної мережі для прогнозування метеоданих. Для навчання багат шарової нейронної мережі використовують алгоритм оберненого поширення похибки (“backprop”, “trprop+”, “trprop-”). Як приклад реалізовано штучну нейронну мережу для короткочасного прогнозування параметрів довкілля (температури і тиску на Чорногірському географічному стаціонарі Львівського національного університету імені Івана Франка) з використанням мови програмування R та пакета “neuralnet”.

Ключові слова: штучна нейронна мережа, параметри довкілля, пакет “neuralnet”, нейрон, мова програмування R, алгоритм оберненого поширення похибки, температура і тиск повітря.

Підвищення ефективності методів прогнозування метеорологічних даних було і є актуальною проблемою. Особливо високі вимоги ставлять до точності короткочасних (на термін до однієї доби) прогнозів, що потребує наявності в районі локальної станції спостережень метеорологічних даних. Останнім часом з огляду на появу автономних портативних метеостанцій [1] проблема локального прогнозування знову стає об’єктом пильної уваги дослідників.

Використання штучних нейронних мереж (ШНМ) для прогнозування часових рядів ґрунтується на здатності нейронних мереж апроксимувати нелінійні функції. Існують різні підходи до прогнозування часових рядів за допомогою ШНМ. До них належать багат шарові нейронні мережі, рекурентні нейронні мережі, нейронні мережі радіальних базисних функцій, узагальнені регресивні нейронні мережі. Ми розглянемо алгоритм прогнозування часових рядів за допомогою багат шарової ШНМ та реалізацію його мовою програмування R з використанням пакета “neuralnet”. Алгоритм складається з трьох етапів: попереднє опрацювання вхідних даних; визначення кількості нейронів у прихованому шарі або шарах; побудова нейронної мережі прогнозування [2].

На першому етапі виконують попереднє опрацювання часових рядів, зокрема процедуру видалення шумових і флуктуаційних складових у часових рядах; урахування специфічних характеристик прогнозованого процесу (сезонність значень у випадку прогнозування параметрів довкілля); нормування даних (зведення до інтервалів [0...1] або

$[-1 \dots 1]$) з метою підвищення точності подання інформації перед наступними етапами прогнозування.

Для попереднього опрацювання вхідних даних виконують їхнє згладжування з використанням методу ковзного середнього. Нехай $x(t_k)$, $k = 1, 2, \dots, N$ – часовий ряд довжиною N . Тоді згладжений часовий ряд $\bar{x}(t_k)$ можна отримати так:

$$\bar{x}(t_k) = \begin{cases} x(t_k), & k = 1, 2, \dots, p \\ \frac{1}{p} \sum_{i=1}^p x(t_{k-i}), & k = p+1, \dots, N \end{cases} \quad (1)$$

де p – період ковзання.

У ході побудови ШНМ як функцію активації часто використовують сигмоїдальну функцію або гіперболічний тангенс. Для сигмоїдальної функції активації значення на виході нейрона може змінюватися в діапазоні $[0 \dots 1]$, тому для нормування часового ряду використовували залежність

$$x_n(t_k) = \alpha + \frac{\bar{x}(t_k) - x_{\min}}{x_{\max} - x_{\min}} \cdot \beta, \quad (2)$$

де x_{\min} , x_{\max} – мінімальне і максимальне значення часового ряду $\bar{x}(t_k)$, $k = 1, 2, \dots, N$; $\alpha \dots \alpha + \beta$ – діапазон зміни значень нормованого часового ряду $x_n(t_k)$, $k = 1, 2, \dots, N$.

У разі використання як функції активації гіперболічного тангенса значення на виході нейрона змінюється в межах $[-1 \dots 1]$, і нормування часового ряду виконують за допомогою залежності

$$x_n(t_k) = \frac{2\bar{x}(t_k) - x_{\max} - x_{\min}}{x_{\max} - x_{\min}} \cdot \beta, \quad (3)$$

де скаляр β ($0 < \beta \leq 1$) задає діапазон зміни ($-\beta \dots \beta$) значень нормованого часового ряду $x_n(t_k)$, $k = 1, 2, \dots, N$.

Також для нормування даних часового ряду $\bar{x}(t_k)$, $k = 1, 2, \dots, N$ можна скористатися стандартним виразом

$$x_n(t_k) = \frac{\bar{x}(t_k) - Mx}{\sqrt{Dx}}, \quad (4)$$

де $Mx = \frac{1}{N} \sum_{i=1}^N \bar{x}(t_i)$ – вибіркова оцінка математичного сподівання,

$Dx = \frac{1}{N-1} \sum_{i=1}^N (\bar{x}(t_i) - Mx)^2$ – вибіркова оцінка дисперсії.

Для того, щоб отримані значення на виході нейронної мережі відповідали реальним даним, потрібно виконати обернене до нормування перетворення. Зокрема, у випадку використання виразів (2) або (3) обернені перетворення матимуть, відповідно, такий вигляд:

$$x_{\text{вих}}(t_k) = x_{\text{мін}} + \frac{(x_n(t_k) - \alpha)(x_{\text{макс}} - x_{\text{мін}})}{\beta}, \quad (5)$$

$$x_{\text{вих}}(t_k) = \frac{x_{\text{макс}} + x_{\text{мін}}}{2} + \frac{x_n(t_k)(x_{\text{макс}} - x_{\text{мін}})}{2\beta}, \quad (6)$$

де $x_{\text{вих}}(t_k)$ – отримані значення на виході ШНМ після денормування.

Для визначення кількості нейронів у прихованому шарі можна скористатися правилом Baum-Haussler [2]:

$$N_{\text{hidden}} \leq \frac{N_{\text{train}} \cdot \varepsilon_{\text{tol}}}{N_{\text{pts}} + N_{\text{out}}}, \quad (7)$$

де N_{train} – кількість навчальних вибірок; ε_{tol} – похибка навчання мережі; N_{pts} – кількість елементів навчальної вибірки; N_{out} – кількість нейронів у вихідному шарі.

Формування навчальних вибірок для ШНМ виконують за принципом “ковзного вікна”. Цей принцип передбачає використання двох вікон W_1 і W_0 з фіксованими розмірами p і q , відповідно. Вікна переміщуються з деяким кроком ковзання s по даних часового ряду, починаючи з першого елемента. У цьому разі перше вікно $W_1(p)$ формує вхідний вектор ШНМ, а друге вікно $W_0(q)$ – вихідний вектор. Пару вхідного і вихідного векторів приймають за реалізацію однієї навчальної пари. У разі зсуву часових вікон за часовим рядом з кроком s отримуємо другу і наступні навчальні пари.

Нехай вихідне вікно має розмірність $q = 1$ і крок ковзання $s = 1$. Тоді сформована множина навчальних пар матиме такий вигляд:

Номер н. п.	Входи				Виходи
1	$x(t_1)$	$x(t_2)$...	$x(t_p)$	$x(t_{p+1})$
2	$x(t_2)$	$x(t_3)$...	$x(t_{p+1})$	$x(t_{p+2})$
3	$x(t_3)$	$x(t_4)$...	$x(t_{p+2})$	$x(t_{p+3})$
...
n	$x(t_n)$	$x(t_{n+1})$...	$x(t_{p+n-1})$	$x(t_{p+n})$

Для прогнозування часових рядів найчастіше використовують ШНМ прямого поширення сигналів без обернених зв'язків (FeedForward). У загальному випадку ШНМ прямого поширення сигналу з алгоритмом навчання у вигляді оберненого поширення похибки (Back Propagation) складаються з вхідного, вихідного й одного або декількох прихованих шарів. До складу кожного шару входить один або декілька нейронів. У мережах прямого поширення інформація передається тільки від вхідного шару до прихованих шарів, а від них – до вихідного шару нейронів (виходи нейронів i -го шару з'єднані з усіма входами всіх нейронів наступного $i+1$ -го шару).

Кількість вагових коефіцієнтів N_w у ШНМ можна визначити за допомогою такого виразу:

$$N_w = \sum_{i=1}^{N_f-1} n_i \cdot n_{i+1}, \quad (8)$$

де N_i – кількість шарів у ШНМ; n_i – кількість нейронів у i -му шарі. Наприклад, ШНМ, яка має вхідний шар з чотирма нейронами, один прихований шар з п'ятьма нейронами і вихідний шар, що містить один нейрон, складатиметься з 25 вагових коефіцієнтів.

Одна з основних властивостей ШНМ – це здатність до навчання. Процес навчання полягає в зміні вагових коефіцієнтів між нейронами ШНМ так, щоб сумарна середньо-квадратична похибка нейронної мережі для p -ї навчальної пари була мінімальною:

$$E_p = 0.5 \sum_{j=1}^m (y_{pj} - o_{pj}^{(N)})^2, \quad (9)$$

де m – кількість нейронів у вихідному шарі мережі; $o_{pj}^{(N)}$ і y_{pj} – дійсний і бажаний вихід j -го нейрона N -го вихідного шару ШНМ для p -ї навчальної пари.

Зміну вагових коефіцієнтів w_{ij} від i -го нейрона шару $n-1$ до j -го нейрона наступного шару n у разі використання градієнтного методу вибирають пропорційно до частинної похідної від функції похибки $E(w)$ за коефіцієнтом w_{ij} :

$$\Delta w_{ij}^{(n)} = -\mu \frac{\partial E(w)}{\partial w_{ij}}, \quad (10)$$

де μ – параметр, що визначає швидкість навчання мережі ($0 < \mu < 1$).

Співвідношення (10) у разі зміни вагових коефіцієнтів для p -ї навчальної пари можна записати у вигляді

$$\Delta_p w_{ij}^{(n)} = -\mu \cdot \delta_{pj}^{(n)} \cdot o_{pj}^{(n-1)}, \quad (11)$$

причому похибка δ_{pj} нейрона з сигмоїдальною функцією активації для вихідного шару визначена виразом

$$\delta_{pj} = o_{pj} (1 - o_{pj}) (y_{pj} - o_{pj}), \quad (12)$$

а для прихованого шару – виразом

$$\delta_{pj} = o_{pj} (1 - o_{pj}) \sum_k \delta_{pk} w_{jk}. \quad (13)$$

Одним із недоліків алгоритму з оберненим поширенням похибки на основі градієнтного методу є дуже тривалий процес навчання. Існує низка алгоритмів, що прискорюють процес їхнього навчання. До них належить і алгоритм, який отримав назву Resilient Propagation (RProp) [3]. На відміну від стандартного алгоритму BackProp, RProp використовує тільки знаки частинних похідних для корекції вагових коефіцієнтів, яка виконується після подачі на вхід нейронної мережі всіх навчальних пар. Для кожної ітерації нові значення вагових коефіцієнтів обчислюють за допомогою виразу

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \Delta w_{ij}^{(t)}, \quad (14)$$

де індекс t – номер ітерації. Розміри кроків визначають за допомогою виразу

$$\Delta w_{ij}^{(t)} = -\text{sign} \left(\frac{\partial E^{(t)}}{\partial w_{ij}} \right) \cdot \Delta_{ij}^{(t)}. \quad (15)$$

Для кожного вагового коефіцієнта $w_{ij}^{(t)}$ на t -й ітерації значення його корекції визначають таким виразом:

$$\Delta_{ij}^{(t)} = \begin{cases} \min(\eta^+ \cdot \Delta_{ij}^{(t-1)}, \Delta_{\max}), & \text{якщо } \frac{\partial E^{(t)}}{\partial w_{ij}} \cdot \frac{\partial E^{(t-1)}}{\partial w_{ij}} > 0 \\ \max(\eta^- \cdot \Delta_{ij}^{(t-1)}, \Delta_{\min}), & \text{якщо } \frac{\partial E^{(t)}}{\partial w_{ij}} \cdot \frac{\partial E^{(t-1)}}{\partial w_{ij}} < 0, \\ \Delta_{ij}^{(t-1)}, & \text{в іншому випадку} \end{cases} \quad (16)$$

де η^- , η^+ – константи ($0 < \eta^- < 1 < \eta^+$, $\eta^- = 0.5$, $\eta^+ = 1.2$ [4]). Якщо частинна похідна $\frac{\partial E^{(t)}}{\partial w_{ij}}$ має той самий знак для двох послідовних ітерацій, то значення кроку корекції збі-

льшується. У разі зміни знака похідної $\frac{\partial E^{(t)}}{\partial w_{ij}}$ значення кроку корекції зменшується. Кро-

к корекції обмежений параметрами $\Delta_{\max} = 50$ і $\Delta_{\min} = 10^{-06}$ [4].

У праці [5] запропоновано поліпшення алгоритму RProp для навчання ШНМ з урахуванням глобальної зміни похибки. Алгоритм отримав назву RProp з Weight Backtracking (з “відкатом” значення вагового коефіцієнта). Суть модифікації така: у разі зміни знака градієнта “відкат” виконується тільки у випадку, якщо загальна похибка збільшилася з попереднього кроку. Після зміни кроку згідно з (16) оновлюються вагові коефіцієнти. Розрізняють два випадки. Якщо знак частинної похідної не змінився, то оновлення вагових коефіцієнтів виконується за виразом

$$\Delta w_{ij}^{(t)} = -\text{sign}\left(\frac{\partial E^{(t)}}{\partial w_{ij}}\right) \cdot \Delta_{ij}^{(t)}, \quad \text{якщо } \frac{\partial E^{(t)}}{\partial w_{ij}} \cdot \frac{\partial E^{(t-1)}}{\partial w_{ij}} > 0. \quad (17)$$

У випадку зміни знака частинної похідної попереднє оновлення ваг виконується згідно з виразом

$$\left\{ \Delta w_{ij}^{(t)} = -\Delta w_{ij}^{(t-1)}, \frac{\partial E^{(t)}}{\partial w_{ij}} = 0 \right\}, \quad \text{якщо } \frac{\partial E^{(t)}}{\partial w_{ij}} \cdot \frac{\partial E^{(t-1)}}{\partial w_{ij}} < 0. \quad (18)$$

Прогнозування часових рядів на основі ШНМ можна розділити на одно- та багатокрокове. Під час однокрокового прогнозування на вхід ШНМ подаються значення останнього навчального вхідного вектора $x(t_{N-p+1})$, $x(t_{N-p+2})$, ..., $x(t_N)$, а на виході формується прогнозована величина $\hat{x}(t_{N+1})$. У разі врахування специфіки часових рядів такий тип прогнозування не завжди доцільний, проте в певних випадках короткотривалих прогнозів ним можна скористатися.

Під час багатокрокового прогнозування часового ряду перший крок збігається з однокроковим прогнозуванням. Отримане прогнозоване значення виходу $\hat{x}(t_{N+1})$ одночасно додається до значень навчального вхідного вектора. На другому кроці прогнозування на вхід ШНМ подається вектор $x(t_{N-p+2})$, $x(t_{N-p+3})$, ..., $\hat{x}(t_{N+1})$, а на виході отримують значення $\hat{x}(t_{N+2})$. Наступні кроки прогнозування аналогічно повторюються. Багатокрокове прогнозування дає змогу виконувати коротко- і середньотривале прогно-

зування. Суттєвий вплив на точність має нагромаджена похибки на кожному кроці прогнозування.

Оцінку якості прогнозування виконували за допомогою стандартних критеріїв:

– середньої абсолютної відсоткової похибки (MAPE):

$$MAPE = \frac{100}{N} \cdot \sum_{i=1}^N \left| \frac{x_i - \hat{x}_i}{x_i} \right|, \% \quad (19)$$

де x_i – фактичне значення часового ряду; \hat{x}_i – прогнозоване значення; N – кількість значень часового ряду;

– середньоквадратичної похибки (MSE):

$$MSE = \frac{1}{N-1} \cdot \sum_{i=1}^N (x_i - \hat{x}_i)^2 \quad (20)$$

Навчання багатшарової ШНМ для прогнозування параметрів довкілля (температури, тиску) виконували за допомогою мови програмування R (версія 3.1.1). У разі реалізації програмного забезпечення для прогнозування часових рядів за допомогою ШНМ мовою програмування R використовували функції, які входять до складу пакета “neuralnet” [7].

Зокрема, функцію *neuralnet()*, що входить до складу пакета “neuralnet”, використовували для навчання ШНМ прямого поширення сигналу з застосуванням алгоритмів оберненого поширення похибки (“backprop”, “grprop+”, “grprop-”) або алгоритму глобальної збіжності (“grgor”). Функція *neuralnet()* дає змогу виконувати гнучкі налаштування для оптимального навчання ШНМ через вибір функції для обчислення похибки і функції активації.

Численні експерименти з побудови багатшарової ШНМ прямого поширення з різними функціями активації та прогнозування на її основі виконували для періодичного часового ряду, отриманого з модельної функції

$$y(t) = 7 + 11 \sin\left(\frac{2\pi}{365}t\right) + 3 \sin\left(\frac{2\pi}{7}t\right) \quad (21)$$

Можна вважати, що цей ряд грубо моделює зміну середньодобової температури протягом року. Його вигляд зображений на рис. 1. Досліджуваний часовий ряд складався з 380 елементів. З них 340 елементів утворюють навчальну множину, а 40 – тестову. Вибір навчальних і тестових пар виконувався випадковим способом.

Структуру нейронної мережі вибирали експериментально. Досліджували ШНМ з такою структурою: вхідний шар – 4–16 входів; прихований шар – 4–16 нейронів; вихідний шар – 1 нейрон.

Нормування часового ряду, утвореного з функції (21), виконували за допомогою виразу (3), де $\beta = 0,6$. Навчання, тестування ШНМ для цього ряду і короткочасне прогнозування на її основі проводили за допомогою функції *neuralnet()*. Для навчання нейронної мережі використовували алгоритм “grprop+”, функцію активації вибирали як сигмоїдальну функцію (*act.fct* = “logistic”) і у вигляді гіперболічного тангенса (*act.fct* = “tanh”). Критерій закінчення навчання нейронної мережі задається як деякий поріг (*threshold* = 0,0001) для частинних похідних цільової функції.

Результати побудови ШНМ і прогнозування модельного часового ряду наведені в табл. 1. У дужках відображено дані, отримані для ШНМ з функцією активації у вигляді

гіперболічного тангенса. Похибки прогнозування, наведені в табл. 1, відповідають прогнозуванню на сім часових кроків.

Таблиця 1

Результати навчання, тестування ШНМ та прогнозування на її основі часового ряду

N_{in}	N_{hidden}	Навчання ШНМ		Тестування ШНМ		Прогноз ($N_{pred} = 7$)	
		MAPE, %	MSE	MAPE, %	MSE	MAPE, %	MSE
4	4	7,50 (0,78)	0,03 (0,53E-3)	0,91 (0,09)	0,40E-2 (0,36E-3)	9,04 (0,65)	1,13 (0,72E-2)
	8	3,03 (1,09)	0,61E-2 (0,98E-3)	0,25 (0,08)	0,89E-3 (0,69E-4)	6,98 (0,93)	0,73 (0,04)
	12	5,09 (0,89)	0,02 (0,56E-3)	0,65 (0,14)	0,25E-2 (0,65E-4)	11,48 (2,55)	1,78 (0,23)
	16	7,53 (0,84)	0,03 (0,64E-3)	0,90 (0,06)	0,483E-3 (0,48E-4)	9,11 (1,96)	1,14 (0,12)
8	4	1,93 (0,56)	0,30E-2 (0,19E-3)	0,09 (0,04)	0,27E-3 (0,30E-4)	0,16 (0,10)	0,82E-3 (0,20E-3)
	8	1,86 (0,75)	0,40E-2 (0,53E-3)	0,13 (0,06)	0,30E-3 (0,58E-4)	0,43 (0,09)	0,32E-2 (0,30E-3)
	12	1,79 (0,90)	0,40E-2 (0,51E-3)	0,169 (0,05)	0,31E-3 (0,48E-4)	0,55 (0,32)	0,0072 (0,17E-2)
	16	2,46 (0,75)	0,40E-2 (0,86E-3)	0,12 (0,07)	0,47E-3 (0,11E-3)	0,36 (0,28)	0,26E-3 (0,32E-2)
12	4	1,24 (0,62)	0,30E-2 (0,62E-3)	0,08 (0,04)	0,46E-3 (0,71E-4)	0,35 (0,09)	0,23E-3 (0,17E-4)
	8	10,45 (0,50)	0,06 (0,21E-3)	0,45 (0,02)	0,74E-2 (0,37E-4)	2,04 (0,08)	0,16 (0,17E-3)
	12	1,40 (0,63)	0,38E-2 (0,35E-3)	0,10 (0,04)	0,69E-3 (0,37E-4)	0,35 (2,40)	0,22E-2 (0,39)
	16	1,25 (1,00)	0,45E-2 (0,60E-3)	0,08 (0,04)	0,79E-3 (0,52E-4)	0,14 (6,71)	0,37E-3 (1,69)
16	4	1,16 (0,70)	0,28E-2 (0,61E-3)	0,10 (0,04)	0,20E-3 (0,52E-4)	0,12 (0,11)	0,23E-3 (0,27E-3)
	8	1,51 (0,45)	0,28E-2 (0,20E-3)	0,07 (0,02)	0,20E-3 (0,11E-4)	0,23 (2,74)	0,13E-2 (0,46)
	12	1,24 (0,96)	0,19E-2 (0,48E-3)	0,056 (0,05)	0,12E-3 (0,44E-4)	0,29 (1,02)	0,17E-2 (0,06)
	16	0,82 (0,82)	0,72E-3 (0,34E-3)	0,06 (0,07)	0,41E-3 (0,32E-4)	0,16 (0,50)	0,41E-3 (0,01)

Отримані результати дають підставі стверджувати, що похибки MAPE, MSE є меншими в разі використання гіперболічного тангенса порівняно з сигмоїдальною функцією активації, у випадку як побудови ШНМ для модельного часового ряду, так і прогнозування на її основі.

У цьому разі для побудови ШНМ часовий ряд отримували моделюванням функції (21) без урахування шумів вимірювань. Для практичного використання важливим є прогнозування часового ряду за допомогою ШНМ, отриманого вимірюваннями.

Розглянемо побудову ШНМ для прогнозування параметрів довкілля, яка охоплює попередню підготовку даних та етапи її навчання, тестування і прогнозування. Параметри довкілля – це температура і тиск повітря, виміряні на метеостанції Чорногірського

географічного стаціонару Львівського національного університету імені Івана Франка за період з січня 2009 р. по січень 2010 р. Інтервал вимірювань даних $t = 15$ хв. Довжина досліджуваних часових рядів температури, тиску упродовж 13 місяців – $n = 38\,016$. Прогнозування температури і тиску виконували на період один–сім днів.

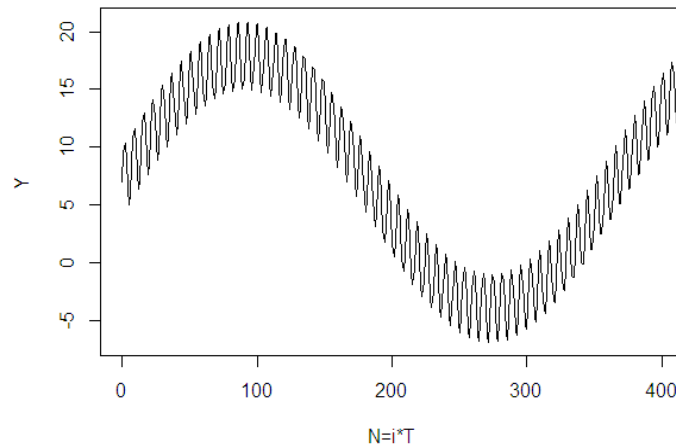


Рис. 1. Графік часового ряду, отриманого з модельної функції (21)

Для навчання нейронної мережі використовували той же алгоритм “trgor+”, функцію активації вибрали у вигляді гіперболічного тангенса. Критерій закінчення навчання нейронної мережі: $\text{threshold} = 0,001$. Зі зменшенням цього порога суттєво зростає кількість кроків для навчання ШНМ.

Перед побудовою ШНМ виконано усереднення по днях ($t = 1$ день) часових рядів температури і тиску. Після цього довжина часових рядів $n = 396$. Для отриманих часових рядів виконано попереднє опрацювання даних, яке полягало в їхньому згладжуванні та нормуванні. Для згладжування даних використовували метод ковзного середнього – вираз (1) з періодом ковзання $p = 4$.

Нормування часових рядів температури і тиску виконували за допомогою виразу (3), де $\beta = 0,6$. Діапазон зміни значень нормованих часових рядів становив $(-0,6 \dots 0,6)$. Денормування виконували згідно з виразом (6).

Вигляд часових рядів температури і тиску повітря зображено на рис. 2, де суцільною лінією (Y1) позначено виміряні дані, а штриховою лінією з кружечками (Y2) – згладжені.

Як базову архітектуру для прогнозування часових рядів вибрано багатошарову ШНМ з одним прихованим шаром нейронів. Також дослідження проводили для ШНМ з декількома прихованими шарами (2, 3), проте отримані результати суттєво не відрізнялися від ШНМ з одним прихованим шаром. Розбиття часових рядів температури і тиску на навчальні й тестові підвибірки виконувалося випадково.

Розмірність навчальної вибірки становила 340 елементів, тестової – 40, що відповідає приблизно співвідношенню 90 : 10 %. Решту останніх елементів часових рядів використовували для порівняння з отриманими прогнозованими значеннями температури і тиску.

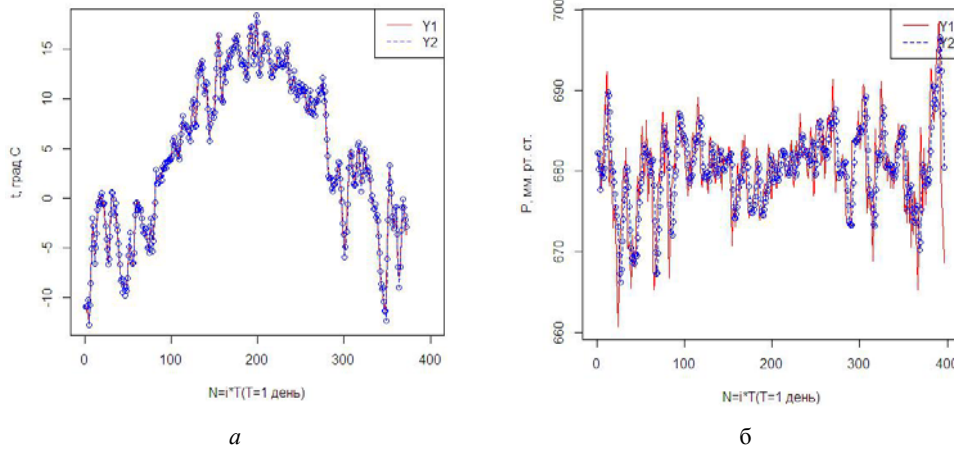


Рис. 2. Графіки температури (а) і тиску (б) повітря, зроблені на підставі даних метеостанції Чорногірського географічного стаціонару за січень 2009 – січень 2010 рр.

Дослідження ефективності ШНМ для прогнозування часових рядів виконували для різних значень її входів та кількості нейронів у прихованому шарі. Зокрема, проводили дослідження для $N_{in} = (4, 8, 12, 16)$, $N_{hidden} = (4, 8, 16, 24)$ при $N_{out} = 1$. Отримані результати наведені в табл. 2 (ШНМ для прогнозування температури повітря) і табл. 3 (ШНМ для прогнозування тиску повітря). У таблицях зведені результати з навчання ШНМ (340 елементів підвибірки), її тестування (40 елементів підвибірки) та апроксимації часових рядів (380 елементів вибірки). Як критерії якості в ході побудови ШНМ використано середню абсолютну відсоткову похибку (вираз (19)) і середньоквадратичну похибку (вираз (20)).

Таблиця 2

Результати навчання ШНМ для прогнозування температури повітря

N_{in} (N_{hidden})	Навчання ШНМ		Тестування ШНМ		Апроксимація	
	MAPE, %	MSE	MAPE, %	MSE	MAPE, %	MSE
4 (4)	51,10 (46,58)	3,343 (0,450)	7,89 (1,831)	0,809 (0,055)	53,34 (43,77)	3,754 (0,456)
4 (8)	52,21 (55,71)	3,344 (0,456)	8,36(1,71)	0,854 (0,056)	54,77 (51,93)	3,795 (0,463)
4 (16)	51,68 (46,14)	2,228 (0,453)	8,95(1,83)	4,641 (0,053)	54,82 (43,38)	6,210 (0,458)
4 (24)	38,84 (44,97)	1,455 (0,420)	14,62(1,97)	2,540 (0,064)	48,35 (42,45)	3,612 (0,438)
8 (4)	50,26 (59,51)	3,138 (0,308)	6,416 (3,063)	0,431 (0,063)	51,80 (57,19)	3,262 (0,339)
8 (8)	41,20 (36,13)	2,337 (0,278)	11,89 (2,964)	0,955 (0,059)	48,52 (35,73)	3,009 (0,308)
8 (16)	27,45 (22,57)	1,346 (0,220)	8,990 (2,738)	0,982 (0,102)	33,31 (23,13)	2,128 (0,294)
8 (24)	16,13 (31,88)	0,565 (0,226)	16,68 (2,607)	7,233 (0,046)	29,99 (31,52)	7,126 (0,249)
12 (4)	51,46 (43,39)	2,935 (0,287)	3,857 (13,33)	0,312 (0,025)	51,11 (52,40)	3,000 (0,289)
12 (8)	34,46 (19,97)	1,946 (0,230)	14,59 (1,597)	0,544 (0,034)	45,33 (19,92)	2,300 (0,243)
12 (16)	17,21 (23,84)	0,510 (0,194)	21,64 (21,60)	1,304 (0,026)	35,90 (41,99)	1,676 (0,204)
12 (24)	10,57 (11,11)	0,225 (0,174)	24,75 (17,93)	1,142 (0,058)	32,64 (26,83)	1,263 (0,214)
16 (4)	49,44 (60,29)	2,483 (0,274)	4,247 (0,986)	0,503 (0,031)	50,14 (57,23)	2,790 (0,285)
16 (8)	30,90 (44,37)	1,296 (0,220)	14,20 (0,914)	18,44 (0,042)	46,10 (42,30)	14,47 (0,244)
16 (16)	7,962 (17,41)	0,208 (0,130)	6,785 (1,711)	1,117 (0,096)	13,77 (17,86)	1,238 (0,211)
16 (24)	4,812 (17,60)	0,136 (0,116)	14,22 (1,397)	2,986 (0,061)	17,78 (17,74)	2,915 (0,166)

Побудову ШНМ виконували як для вимірних значень температури і тиску повітря, так і для їхніх згладжених значень (дані таблиць у дужках). З отриманих результатів навчання і тестування ШНМ (див. табл. 2, 3) випливає, що для згладжених згідно з виразом (1) метеоданих якість їхньої апроксимації є суттєво ліпшою.

За малої кількості входів ШНМ ($N_{in} < 8$) збільшення кількості нейронів у прихованому шарі не приводить до поліпшення точності як їхнього навчання, так і їхнього тестування. У разі більшої кількості входів ШНМ ($N_{in} \geq 8$) збільшення кількості нейронів у прихованому шарі однозначно поліпшує точність навчання. Збільшення кількості входів ШНМ за фіксованої кількості нейронів у прихованому шарі теж поліпшує точність апроксимації метеоданих. У випадку невеликої кількості нейронів у прихованому шарі ($N_{hidden} \leq 8$) кількість входів ШНМ практично не впливає на точність апроксимації. Зі збільшенням кількості нейронів у прихованому шарі кількість входів ШНМ може поліпшити точність апроксимації часового ряду.

Зі збільшенням часового інтервалу прогнозування точність прогнозу погіршується. Це особливо чітко виявляється для згладжених метеоданих. Для випадку незгладжених метеоданих ця тенденція зберігається. У разі більшої кількості входів ШНМ ($N_{in} \geq 8$) збільшення кількості нейронів прихованого шару збільшує точність прогнозування.

Вибір структури ШНМ виконували емпірично на підставі залежностей похибок апроксимації від розміру нейронної мережі за заданого розміру вибірки вимірних метеоданих.

Таблиця 3

Результати навчання ШНМ для прогнозування тиску повітря

N_{in} (N_{hidden})	Навчання ШНМ		Тестування ШНМ		Апроксимація	
	MAPE, %	MSE	MAPE, %	MSE	MAPE, %	MSE
4 (4)	0,292 (0,097)	12,036 (1,362)	0,033 (0,011)	1,287 (0,159)	0,294 (0,097)	12,045 (1,375)
4 (8)	0,264 (0,088)	10,694 (1,216)	0,029 (0,012)	1,131 (0,189)	0,265 (0,091)	10,690 (1,271)
4 (16)	0,204 (0,075)	6,698 (0,843)	0,037 (0,016)	1,616 (0,418)	0,218 (0,082)	7,516 (1,140)
4 (24)	0,180 (0,064)	5,570 (0,602)	0,054 (0,017)	4,213 (0,458)	0,211 (0,073)	8,844 (0,959)
8 (4)	0,254 (0,084)	5,586 (0,538)	0,033 (0,011)	0,847 (0,410)	0,262 (0,087)	5,879 (0,866)
8 (8)	0,218 (0,072)	4,049 (0,412)	0,033 (0,009)	0,956 (0,104)	0,230 (0,074)	4,573 (0,472)
8 (16)	0,143 (0,0429)	1,747 (0,174)	0,061 (0,013)	3,099 (0,279)	0,186 (0,052)	4,429 (0,415)
8 (24)	0,061 (0,039)	0,418 (0,145)	0,057 (0,018)	2,517 (0,671)	0,107 (0,053)	2,682 (0,746)
12 (4)	0,249 (0,075)	4,938 (0,420)	0,033 (0,007)	2,525 (0,055)	0,261 (0,075)	6,894 (0,420)
12 (8)	0,180 (0,065)	2,648 (0,339)	0,034 (0,007)	1,118 (0,047)	0,197 (0,067)	3,479 (0,357)
12 (16)	0,099 (0,041)	0,896 (0,145)	0,051 (0,011)	2,378 (0,142)	0,139 (0,049)	3,026 (0,265)
12 (24)	0,020 (0,025)	0,056 (0,062)	0,080 (0,013)	7,665 (0,251)	0,092 (0,036)	7,132 (0,289)
16 (4)	0,232 (0,067)	4,486 (0,376)	0,038 (0,008)	1,399 (0,063)	0,251 (0,070)	5,497 (0,410)
16 (8)	0,157 (0,049)	2,104 (0,194)	0,029 (0,011)	0,934 (0,196)	0,175 (0,057)	2,838 (0,364)
16 (16)	0,039 (0,026)	0,151 (0,059)	0,087 (0,017)	11,222 (0,361)	0,117 (0,040)	10,622 (0,393)
16 (24)	0,006 (0,015)	0,004 (0,025)	0,068 (0,018)	5,080 (0,359)	0,069 (0,031)	4,748 (0,359)

Результати короткочасного прогнозування температури і тиску повітря наведені в табл. 4. Отримані значення прогнозу порівнювали з вимірними метеостанцією значеннями температури і тиску. У цьому разі використовували ШНМ зі структурою (8,8,1). Під час навчання цієї мережі задана похибка становила $threshold = 0,0001$. Якість прогнозування оцінювали за допомогою виразів (19) і (20). Наприклад, для температури повітря похибки становили: $MAPE = 51,897\%$, $MSE = 0,261$, а для тиску повітря –

MAPE = 0,214 %, MSE = 3,336. Також у табл. 4 наведено значення відносної похибки ϵ для кожного з часових прогнозів. Виконані дослідження дають підстави стверджувати, що зі зменшенням похибки ϵ в певному діапазоні (0,005–0,0001) похибки прогнозування MAPE і MSE метеоданих зменшуються. У разі подальшого зменшення цієї похибки в десятки разів зростає кількість кроків, необхідних для навчання ШНМ, і якість прогнозування суттєво не поліпшується.

Таблиця 4

Результати прогнозування температури і тиску повітря на тиждень

i	Температура повітря			Тиск повітря		
	$x_{msr}(t_i)$	$x_{pred}(t_i)$	$\epsilon, \%$	$x_{msr}(t_i)$	$x_{pred}(t_i)$	$\epsilon, \%$
1	-4,6	-3,8	17,391	682,3	682,1	-0,029
2	-5,1	-3,7	27,451	685,4	684,8	-0,087
3	-6,5	-4,2	35,384	688,3	686,3	-0,290
4	-8,0	-5,2	35,000	690,5	687,6	-0,419
5	-8,4	-6,6	21,428	690,4	688,4	-0,289
6	-9,5	-8,1	14,736	688,7	687,5	-0,174
7	-10,2	-8,5	16,667	687,6	686,0	-0,232

Отже, наведений алгоритм прогнозування метеоданих на основі багатосарової ШНМ реалізований у вигляді програми мовою програмування R з використанням пакета “neuralnet”. Проведено дослідження точності прогнозування за допомогою критеріїв (19) і (20) для ШНМ з різною кількістю нейронів у вхідному та прихованому шарах. Вибір їхньої кількості виконаний емпірично.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Дзендзелюк О. Автоматизована система моніторингу параметрів довкілля / О. Дзендзелюк, І. Мусійчук, В. Рабик // Теор. електротехніка. – 2010. – Вип. 61. – С. 90–98.
2. Lin Feng. Time Series Forecasting with Neural Networks / Lin Feng, Yu Xing Huo, Gregor Shirley // J. of Complexity International 2. 1995. [Електронний ресурс]. – Режим доступу: <http://journal-ci.ccse.monash.edu.au/ci/vol02/cmxxhk/cmxxhk.htm>
3. Riedmiller M. Rprop – Description and Implementation Details / M. Riedmiller // Technical Report. – 1994. January. [Електронний ресурс]. – Режим доступу: www.inf.fu-berlin.de/lehre/WS06/Mustererkennung/Paper/rprop.pdf
4. Riedmiller M., Brawn H. A direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm [Електронний ресурс]. Режим доступу: <http://davinci.fmph.uni-ba.sk/~uhliarik4.recognition/resources/rprop/rb-1993-rprop.pdf>
5. Igel C., Hüsken M. Empirical Evaluation of the Improved Rprop Learning Algorithms. [Електронний ресурс]. Режим доступу: <http://dblp.l3s.de/d2r/resource/publications/journals/igon.lgelH03>
6. Adler J. R in a NutShell – 2 nd ed. – O’Reilly Media, 2012. -722 pp. [Електронний ресурс]. – Режим доступу: <http://git.psylab.info/r-books/downloads/Adler2012.pdf>
7. Package “neuralnet”. [Електронний ресурс]. – Режим доступу: <http://cran.r-project.org/web/packages/neuralnet/neuralnet.pdf>bbk.ac.uk

Стаття: надійшла до редакції 14.04.2015,
доопрацьована 20.04.2015,
прийнята до друку 27.04.2015

PREDICTION ENVIRONMENTAL PARAMETERS BASED ON ARTIFICIAL NEURAL NETWORKS

O. Dzdzelyuk, Z. Lyubun, V. Rabyk

*Ivan Franko National University of Lviv,
Tarnavskogo Str. 107, UA - 79017 Lviv, Ukraine
RabykV@ukr.net*

The general algorithm for time series prediction based on artificial neural networks is considered. This algorithm consists of the following stages: the process of the incoming data, computation of the number of neurons in the hidden layer, the construction of artificial neural network for the forecasting of weather information. For training of multilayer neural network the algorithm of back propagation error ("backprop", "rprop +", "rprop-") is used. As an example, the artificial neural network for short-term forecasting of environmental parameters (temperature and pressure in Chornogirskyy geographical station of Lviv National Ivan Franko University) is implemented using the R programming language and the package "neuralnet".

Keywords: artificial neural network, environmental parameters, package "neuralnet", neuron, R programming language, algorithm of back propagation error, temperature and air pressure.

ПРОГНОЗИРОВАНИЕ ПАРАМЕТРОВ ОКРУЖАЮЩЕЙ СРЕДЫ НА ОСНОВАНИИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

О. Дзендзелюк, З. Любунь, В. Рабик

*Львовский национальный университет имени Ивана Франко,
ул. Ген. Тарнавского, 107, 79017, Львов, Украина
RabykV@ukr.net*

Рассмотрено общий алгоритм прогнозирования временных рядов на основе искусственных нейронных сетей. Этот алгоритм состоит из следующих этапов: обработки входных данных, определения количества нейронов в скрытом слое, построение искусственной нейронной сети для прогнозирования метеоданных. Для обучения многослойной нейронной сети использовано алгоритм обратного распространения ошибки ("backprop", "rprop+", "rprop-"). В качестве примера реализовано искусственную нейронную сеть для кратковременного прогнозирования параметров окружающей среды (температуры и давления на Черногорском географическом стационаре Львовского национального университета имени Ивана Франко) с использованием языка программирования R и пакета "neuralnet".

Ключевые слова: искусственная нейронная сеть, параметры окружающей среды, пакет "neuralnet", нейрон, язык программирования R, алгоритм обратного распространения ошибки, температура и давление воздуха.