

МАТЕМАТИЧНІ МЕТОДИ, МОДЕЛІ
ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ЕКОНОМІЦІ

УДК 338.45:004.8:658.7:334.716

ВИКОРИСТАННЯ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ
ДЛЯ ПРОГНОЗУВАННЯ ЛОГІСТИЧНИХ ВИТРАТ
МАШИНОБУДІВНИХ ПІДПРИЄМСТВUSING ARTIFICIAL NEURAL NETWORKS FOR PREDICTION
OF LOGISTICS COSTS OF ENGINEERING ENTERPRISES

Ковшик В.І.

аспірант кафедри менеджменту та оподаткування,
Національний технічний університет
«Харківський політехнічний інститут»

У статті розглянуто можливість застосування штучних нейронних мереж для прогнозування логістичних витрат на основі інформації про витрати попередніх періодів. Запропоновано використання тришарової мережі з навчанням за методом зворотного розповсюдження помилки. Визначено оптимальну конфігурацію такої нейронної мережі для використання із щомісячною інформацією щодо логістичних витрат машинобудівних підприємств.

Ключові слова: штучні нейронні мережі, логістичні витрати, логістика, управління витратами, прогноз, математичні моделі, машинобудівні підприємства.

В статье рассмотрена возможность применения искусственных нейронных сетей для прогнозирования логистических затрат на основе информации предыдущих периодов. Предложено использование трехслойной сети с обучением методом обратного распространения ошибки. Определена оптимальная конфигурация такой нейронной сети для использования с ежемесячной информацией о логистических затратах машиностроительных предприятий.

Ключевые слова: искусственные нейронные сети, логистические расходы, логистика, управление затратами, прогноз, математические модели, машиностроительные предприятия.

The article deals with investigation of possibility of using artificial neural networks to predict the logistics costs. The forecasting based on the information of previous periods is considered. Author proposes to use a three-layer feedforward neural network with learning by backpropagation algorithm. An optimal configuration of the neural network for use on monthly logistics cost information is defined. Article emphasizes the urgency of the approach usage at the variety of machine-building enterprises.

Keywords: artificial neural networks, logistics costs, logistics, cost management, forecasting, mathematical models, engineering enterprises.

Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями. Нині підприємства машинобудування стикаються з великою кількістю проблем, що потребують ефективного та своєчасного рішення. Економічна криза та вплив різноманітних факторів міжнародного середовища, що швидко змінюється, ставлять перед керівництвом таких компаній завдання щодо модернізації підходів до управління всіма аспектами роботи підприємства. Управління логістичними витратами підприємства не є винятком.

Одним із перспективних шляхів удосконалення роботи у цій сфері є впровадження автоматизованих систем управління витратами. Важливим завданням при цьому є розробка, вдосконалення та адаптація підходів до автоматизованого прогнозування витрат, можливим варіантом реалізації якого є застосування штучних нейронних мереж.

Штучна нейронна мережа (ШНМ) – це математична модель, що найчастіше виконується у вигляді комп'ютерної програми. Вона має на меті створення імітації роботи мозку. ШНМ є

системою, яка навчається, що відкриває широкі можливості для її використання у багатьох практичних та наукових завданнях.

Аналіз останніх досліджень і публікацій, в яких започатковано розв'язання даної проблеми і на які спирається автор. Теорія та практика ШНМ бере свій початок з другої половини ХХ ст. і сьогодні переживає новий пік популярності через розповсюдженість комп'ютерних технологій, підвищення обчислювальних можливостей техніки та розвиток ідей удосконалення штучного інтелекту [1]. Перспективи використання ШНМ для прогнозування числових рядів, у тому числі в економіці, є темою великої кількості наукових робіт іноземних та вітчизняних дослідників, серед них: І.О. Калініна [2], О.І. Чумаченко, В.С. Горбатюк [3], Н.В. Кудрицька [4], Н.Я. Савка [5], Р. Карбоню, К. Лафрамбуаз, [6], І. Каастра [7], Б. Оанча, Ш.К. Чуйку [8], С.Ф. Кроун [9] та багато інших.

Так, І.О. Калініна [2] розглядає застосування ШНМ для прогнозування обсягів продажів порівняно з регресійним алгоритмом ARIMA та обґрунтовує переваги ШНМ у точності отриманих значень. О.І. Чумаченко та В.С. Горбатюк [3] досліджують власне алгоритм прогнозування за допомогою поєднання ШНМ та методу групового урахування аргументів.

Деякі роботи пов'язані з логістикою та управлінням ланцюгами постачання. Так, С.Ф. Кроун [9] розглядає можливість розробки нового підходу до прогнозування з використанням ШНМ у процесі управління товарно-матеріальними запасами. Н.В. Кудрицька [4] прогнозує обсяги перевезень залізничним транспортом, використовуючи готовий програмний продукт (реалізацію ШНМ). Р. Карбоню та ін. [6] застосовують машинне навчання для прогнозування попиту у ланцюзі постачання.

Таким чином, питання прогнозування певних показників, що змінюються у часі, є достатньо опрацьованим науковцями. Проте можливість використання ШНМ для прогнозування саме логістичних витрат у машинобудуванні є недостатньо вивченою, що й зумовлює актуальність обраної теми статті.

Формулювання цілей статті (**постановка завдання**). Мета роботи – адаптація алгоритму застосування ШНМ для прогнозування логістичних витрат та розробка методологічних рекомендацій щодо її застосування в роботі відповідних підрозділів машинобудівних підприємств. Для досягнення зазначеної мети необхідно: розглянути можливість застосування ШНМ для прогнозування витрат; обрати й адаптувати алгоритм навчання мережі для потреб управління логістичними витратами; експериментальним шляхом установити рекомендовану конфігурацію ШНМ із заданою точністю.

Виклад основного матеріалу дослідження з повним обґрунтуванням отриманих наукових результатів. У сфері управління підприємством

ШНМ може ефективно виступати як інструмент прогнозування показників діяльності підприємства, у тому числі з урахуванням факторів внутрішнього і зовнішнього середовища [6, с. 1153]. Особливістю та основною перевагою ШНМ є можливість її навчання на основі попередніх результатів. Отримуючи дані на вхід, ШНМ може працювати у двох режимах: навчання, коли результат порівнюється з очікуваним значенням і мережа підлаштовується з кожною ітерацією, та використання вже навченої мережі. При цьому формування внутрішньої моделі даних відбувається без участі людини, що є перевагою над традиційними методами прогнозування [2, с. 133].

Крім того, перевагами ШНМ порівняно з іншими методами прогнозування є: більше можливих напрямів застосування через те, що ШНМ є непараметричною нелінійною моделлю; можливість використання тоді, коли кількість факторів впливу є суттєво більшою, ніж можливості побудови чіткої математичної моделі; простота автоматизації та використання в роботі відділів логістики; можливість інтеграції з комп'ютерною системою підприємства; виявлення складних і прихованих нелінійних зв'язків.

Для потреб управління логістичними витратами машинобудівних підприємств пропонується використати ШНМ, зображену схематично на рис. 1. Вона складається з трьох шарів нейронів, позначених на схемі буквами. $X = \{x_1, x_2, \dots, x_i\}$ – шар вхідних нейронів (англ. input), $H = \{h_1, h_2, \dots, h_j\}$ – прихований шар (англ. hidden), $O = \{o_1, o_2, \dots, o_k\}$ – шар виходу (англ. output). При цьому навчання мережі відбувається за допомогою методу зворотного поширення помилки (англ. backpropagation) у нейронній мережі прямого розповсюдження, що є одним із найбільш широко використовуваних підходів до побудови структури ШНМ.

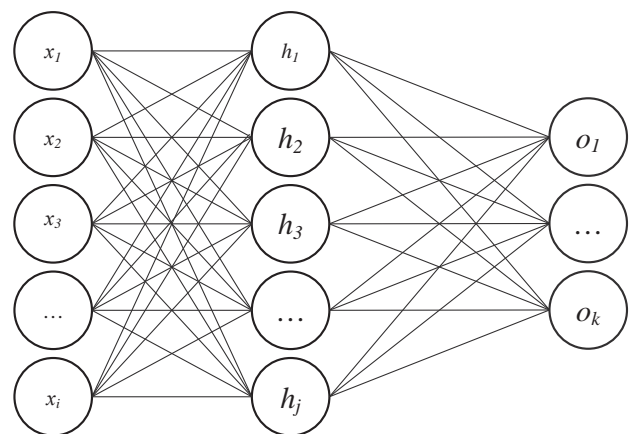


Рис. 1. Структура запропонованої тришарової ШНМ для використання

в управлінні логістичними витратами

Кількість нейронів вхідного шару i зазвичай відповідає кількості одиниць інформації, що мають бути враховані для класифікації, клас-

теризації, прийняття рішення чи прогнозування певної величини [10, с. 5]. У разі прогнозування витрат це кількість попередніх періодів, що враховується під час прогнозу. Інші фактори у цій моделі не враховуються, хоча це також є можливим різновидом використання ШНМ у прогнозуванні.

Кількість нейронів прихованого шару j визначається емпірично шляхом порівняння результативності ШНМ у кожному конкретному випадку залежно від вхідних даних та точності отриманого результату. Рекомендації щодо кількості нейронів прихованого шару суттєво відрізняються у різних наукових джерелах [7; 8; 10; 11]. Рекомендовані значення, що є відправною точкою використання на налаштування ШНМ, включають такі значення, як $j = 2i$; $j = 0,75i$; $j = (4i^2 + 3)/(i^2 - 8)$; $j = i + k + 1$; $j = \sqrt{ik}$ тощо.

Кількість нейронів шару виходу відповідає потребам застосування ШНМ у разі прогнозування $k = 1$.

Складники пропонованої мережі (нейрони) використовують як функції активації так звану сигмо їду, або логістичну функцію (1). Вона є неперервно диференційованою монотонною нелінійною функцією, що дає на виході значення $\sigma(n) \in (0;1)$

$$\sigma(n) = \frac{1}{1 + e^{-n}}, \quad (1)$$

де n – значення, що подається на вхід нейрона.

Можливим також є використання інших функцій активації. Проте функція (1) та її похідна є зручними для обчислення та автоматизації. Крім того, доведено, що сигмоїдні нейрони є найменш схильними до накопичення помилок за умови наявності систематичної помилки на вході нейронної мережі [12, с. 419].

З урахуванням вищезазначеного встановимо додаткові особливості запропонованої ШНМ та її конфігурацію, прийняті у цьому дослідженні:

1) ШНМ має три шари, серед яких один – прихований;

2) функцію активації мають нейрони прихованого шару та шару виходу;

3) числові вхідні значення та цільові показники x попередньо нормалізуються до величин $x \in [0;1]$ за формулою:

$$x^* = \frac{x - \min x}{\max x - \min x}; \quad (2)$$

4) додатковий коефіцієнт зміщення (англ. bias) є окремим для кожного нейрону прихованого шару та шару виходу, початкове значення $b = 1$;

5) кожен нейрон одного шару пов'язаний із кожним нейроном наступного;

6) для навчання ШНМ використовується метод зворотного поширення помилки (англ. backpropagation) [7, с. 217; 8, с. 1406];

7) початкова вага зв'язку між нейронами встановлюється випадковим дійсним числом $w \in [-0,5; 0,5]$, що відповідає потребі мати почат-

ковий набір ваг, середнє арифметичне якого буде наближуватися до 0 [13, с. 22];

8) коефіцієнт швидкості навчання (англ. learning rate) приймається однаковим та незмінним для всієї штучної нейронної мережі.

Навчання та застосування вже навченої мережі включають розрахунок значень вихідних нейронів. Спочатку розраховується чисте вхідне значення для кожного прихованого нейрону $neth_j$ за формулою (3). Воно, своєю чергою, є аргументом функції активації (1) при обчисленні чистого вхідного значення для нейрону виходу $neto_k$ за формулою (4).

$$neth_j = b_j + \sum_i x_i w_{ij}, \quad (3)$$

де b_j – коефіцієнт зміщення нейрону h_j прихованого шару;

w_{ij} – вага зв'язку між нейронами x_i та h_j .

$$neto_k = b_k + \sum_j \sigma(neth_j) w_{jk} \quad (4)$$

де b_k – коефіцієнт зміщення нейрону o_k шару виходу;

w_{jk} – вага зв'язку між нейронами h_j та o_k .

Таким чином, результатом роботи алгоритму нейронної мережі, що виводиться нейроном o_k , є значення $\sigma(neto_k)$, формула (5).

$$\begin{aligned} o_k &= \sigma(neto_k) = \sigma\left(b_k + \sum_j \sigma(neth_j) w_{jk}\right) \\ &= \sigma\left(b_k + \sum_j \left(\sigma(b_j + \sum_i x_i w_{ij}) w_{jk}\right)\right) \end{aligned} \quad (5)$$

Процес навчання штучної нейронної мережі полягає в повторюваному розрахунку результату алгоритму на низці прикладів зі заздалегідь визначеними цільовими значеннями виходу $T = \{t_1, t_2, \dots, t_k\}$ та корекції ваг зв'язків між нейронами шляхом зворотного поширення помилки. Розглянемо цей алгоритм детальніше. На першому кроці відбувається визначення значення помилки E за формулою (6), яка дає змогу оцінити ступінь відповідності фактичних результатів цільовим значенням [13, с. 11; 14, с. 156].

$$E = \frac{1}{2} \sum_k (t_k - o_k)^2 \quad (6)$$

Наступник крок – розрахунок часткової похідної $\frac{\partial E}{\partial w_{jk}}$, що визначає ступінь впливу зміни ваги зв'язку w_{jk} між нейронами прихованого шару та шару виходу на зміну відхилення від цільових показників. Розрахунок ведеться за допомогою ланцюгового правила диференціювання складної функції за формулою (7).

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial neto_k} \times \frac{\partial neto_k}{\partial w_{jk}} \quad (7)$$

Похідна $\frac{\partial E}{\partial o_k}$ є похідною функції (6) і, відповідно, розраховується так:

$$\frac{\partial E}{\partial o_k} = 2 \times \frac{1}{2} (t_k - o_k) \times (-1) + 0 = o_k - t_k \quad (8)$$

Похідна $\frac{\partial o_k}{\partial neto_k}$ є похідною логістичної функції (1):

$$\frac{\partial o_k}{\partial neto_k} = o_k (1 - o_k) \quad (9)$$

Своєю чергою, наступна частина формули (7) $\frac{\partial neto_k}{\partial w_{jk}}$ є похідною функції (4) і розраховується так:

$$\frac{\partial neto_k}{\partial w_{jk}} = 1 \times h_j w_{jk}^{(1-1)} + 0 = h_j \quad (10)$$

Отже, величина зміни ваги зв'язку w_{jk} між нейронами h_j та o_k на поточному кроці навчання штучної нейронної мережі розраховується за формулою (11). При цьому вага зв'язку на наступній ітерації обчислюється шляхом віднімання отриманого значення, помноженого на швидкість навчання від поточного значення ваги обраного зв'язку, формула (12).

$$\frac{\partial E}{\partial w_{jk}} = (o_k - t_k) o_k (1 - o_k) h_j \quad (11)$$

$$w_{jk}^{\wedge} = w_{jk} - \eta \frac{\partial E}{\partial w_{jk}}, \quad (12)$$

де w_{jk}^{\wedge} – вага зв'язку для наступної ітерації;
 η – коефіцієнт швидкості навчання.

Розрахунок оновлення значення коефіцієнту зміщення b_k відбувається аналогічним чином за винятком того, що третій компонент формули (7) дорівнює 1 як похідна числової константи:

$$\frac{\partial E}{\partial b_k} = (o_k - t_k) o_k (1 - o_k) \quad (13)$$

$$b_k^{\wedge} = b_k - \eta \frac{\partial E}{\partial b_k} \quad (14)$$

Наступна частина алгоритму навчання ШНМ – оновлення ваг w_{ij} зв'язків між нейронами шару входу (x_i) та прихованого шару (h_j). Розрахунок відбувається також за допомогою ланцюгового правила за формулою (15):

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial h_j} \times \frac{\partial h_j}{\partial neto_k} \times \frac{\partial neto_k}{\partial w_{ij}} \quad (15)$$

Оскільки результат обчислення функції активації прихованого нейрону h_j впливає на всі нейрони o_k шару виходу, а отже, і на результат обчислення значення E , то потрібно врахувати цей вплив під час розрахунку $\frac{\partial E}{\partial h_j}$. Ураховуючи вищевикладене та вже розглянуті похідні (8, 9, 10) аналогічних функцій, виразимо перший множник добутку (15) так (16):

$$\begin{aligned} \frac{\partial E}{\partial h_j} &= \sum_k \frac{\partial E_{o_k}}{\partial h_j} = \sum_k \left(\frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial neto_k} \times \frac{\partial neto_k}{\partial h_j} \right) = \\ &= \sum_k (o_k - t_k) o_k (1 - o_k) w_{jk} \end{aligned} \quad (16)$$

Варто зазначити, що значення w_{jk} у формулі (16) береться до оновлення його на попередньому кроці. Заміна ваг зв'язків w на w^{\wedge} відбувається наприкінці ітерації навчання ШНМ.

У результаті аналітично до попереднього кроку маємо такі вирази:

$$\frac{\partial E}{\partial w_{ij}} = \left[\sum_k (o_k - t_k) o_k (1 - o_k) w_{jk} \right] h_j (1 - h_j) x_i \quad (17)$$

$$w_{ij}^{\wedge} = w_{ij} - \eta \frac{\partial E}{\partial w_{ij}} \quad (18)$$

Визначення ступеню корегування значення коефіцієнта зміщення $\frac{\partial E}{\partial b_j}$ та нової його вели-

чини b_j^{\wedge} відбувається за аналогічною процедурою виходячи з логічних побудов, наведених раніше:

$$\frac{\partial E}{\partial b_j} = \left[\sum_k (o_k - t_k) o_k (1 - o_k) w_{jk} \right] h_j (1 - h_j) \quad (19)$$

$$b_j^{\wedge} = b_j - \eta \frac{\partial E}{\partial b_j} \quad (20)$$

Навчання ШНМ є ітераційним процесом. Воно відбувається з використанням заздалегідь підготовленого набору даних, що циклічно подаються на входи мережі, та відповідних цільових значень. Важливою характеристикою є кількість циклів (т. зв. epoch): навчання відбувається або певну їх кількість, або ж до досягнення певного значення E .

Як фактичні дані для аналізу будуть використані дані щомісячних логістичних витрат машинобудівного підприємства за 2010–2016 рр. Попередньо всі 74 значення були нормалізовані за допомогою функції (2).

Навчання ШНМ для прогнозування змін величин у часі полягає у послідовному вводиті в мережу ряду значень $\{x_n, x_{n+1}, \dots, x_{n+i}\}$ для всіх n , для яких виконується нерівність $1 < n < (N - i)$. При цьому цільовим значенням t_1 нейрону виходу o_1 є значення x_{n+i+1} , тобто наступне число у числовому ряді.

Основними параметрами, що впливають на роботу ШНМ визначеної конфігурації, є такі складники: кількість нейронів входу i ; коефіцієнт швидкості навчання, η ; кількість нейронів прихованого шару, j . При цьому максимальна кількість ітерацій приймається за 5 000 із зупинкою навчання у разі виконання умови, що максимальна помилка за результатами, яка обраховується в процесі навчання за формулою (14), становить $E \leq 0,01$.

Як вибірку для тестування роботи мережі використано ряд із 20 значень (27% від загальної множини). Ці значення участі в навчанні ШНМ не беруть. Як характеристику якості навчання в даному дослідженні використано середньоквадратичну помилку (англ. mean squared error, MSE).

Передусім розглянемо зміну точності ШНМ від розміру вхідного шару. Для передбачення рядів чисел, що за гіпотезою мають певні закономірності, використовується різна кількість вхідних нейронів. Так, у роботі [6] використано $i = 5$ під час прогнозування попиту в ланцюзі постачання на основі щомісячних даних. Інші автори використовують 20 [8], 13 [9] нейронів тощо. Виходячи з припущення, що закономірність зміни логістичних витрат відповідає річному циклу, розглянемо значення i : 6, 8, 10, 12.

Мета порівняння – знаходження варіанту конфігурації, що дає змогу із заданою точністю визначити необхідні значення логістичних витрат.

Результати порівняння різних конфігурацій вхідного шару мережі наведено в табл. 1. Значне підвищення точності спостеріга-

ється у разі використання 12 нейронів: MSE на тестовій множині дорівнює 0,0146 за інших базових значень та кількості прихованих нейронів 6 (за формулою $j = i + k + 1$, рядок № 17 табл. 1). Для підвищення точності роботи ШНМ ввід масиву даних для навчання виконано в довільному порядку перед кожної ітерацією [13, с. 16]. Це дає змогу підвищити швидкість та ефективність навчання нейронної мережі шляхом збільшення варіативності даних на вході.

Порівняємо також інші можливі варіанти кількості нейронів прихованого шару: $j = 0,75i$ $j = \sqrt{ik}$ та варіант $j = \frac{(4i^2 + 3)}{(i^2 - 8)}$, який наведено як найбільш оптимальний у вичерпному дослідженні [11, с. 9] можливих способів розрахунку j . При цьому важливим є визначення оптимального коефіцієнту η . Як уже зазначалося, він приймається фіксованим для кожного процесу навчання ШНМ, а його початкове значення корегується до оптимального шляхом послідовних незначних змін ($\pm 0,2$) між тестами. Тести з показником $j = i + k + 1$, які проводилися в першу чергу,

показали, що найбільш оптимальним варіантом коефіцієнту є $\eta = 1,1$, який і був прийнятий за початковий в інших тестах.

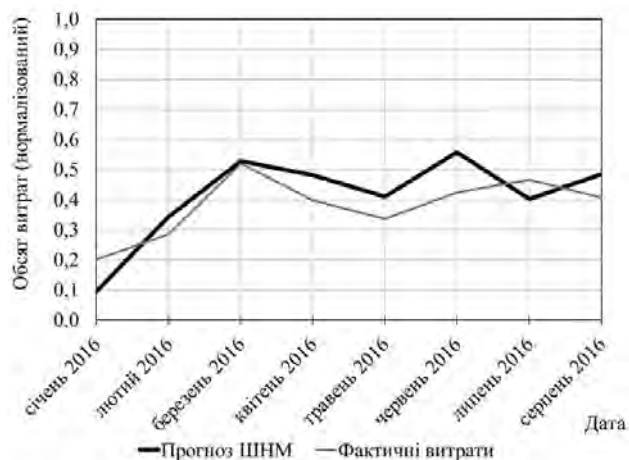


Рис. 2. Результати прогнозування логістичних витрат машинобудівного підприємства за допомогою ШНМ із рекомендованими параметрами

Таблиця 1

Результати навчання ШНМ за умови зміни ключових параметрів

№	Кількість вхідних нейронів, k	Формула кількості нейронів прихованого шару	Кількість нейронів прихованого шару, j	Коефіцієнт швидкості навчання	Результати навчання (середньоквадратична помилка, MSE)			Середнє арифметичне MSE	Середня кількість ітерацій (округлено)
					Тест 1	Тест 2	Тест 3		
1	6	$j = i + k + 1$	8	0,5	0,0279	0,0431	0,0260	0,0323	5000
2	8		10	0,5	0,0436	0,0518	0,0367	0,0440	4883
3	10		12	0,5	0,0554	0,0439	0,0441	0,0478	4081
4	12	$j = \sqrt{ik}$	3	1,3	0,0085	0,0380	0,0171	0,0212	932
5				1,1	0,0128	0,0143	0,0090	0,0120	1050
6				0,9	0,0170	0,0121	0,0205	0,0165	1653
7				0,7	0,0145	0,0234	0,0126	0,0168	2649
8		$j = \frac{(4i^2 + 3)}{(i^2 - 8)}$	4	1,1	0,0123	0,0197	0,0249	0,0189	1507
9				0,9	0,0288	0,0279	0,0191	0,0252	2150
10				1,3	0,0161	0,0147	0,0088	0,0132	993
11	1,5			0,0131	0,0170	0,0099	0,0133	1045	
12	1,7			0,0169	0,0246	0,0075	0,0163	753	
13	0,7	0,0138	0,0189	0,0197	0,0175	2375			
14	$j = 0,75i$	9	1,1	0,0124	0,0103	0,0086	0,0104	1127	
15			0,9	0,0225	0,0205	0,0175	0,0202	1850	
16			1,3	0,0131	0,0157	0,0141	0,0143	938	
17	$j = i + k + 1$	14	0,5	0,0109	0,0187	0,0143	0,0146	2090	
18			0,7	0,0146	0,0146	0,0172	0,0155	1754	
19			0,9	0,0102	0,0086	0,0084	0,0091	1251	
20			1,1	0,0069	0,0081	0,0086	0,0079	1005	
21			1,3	0,0101	0,0159	0,0103	0,0121	943	

За результатами проведених тестів алгоритму визначено найточніший варіант конфігурації: $i = 12$; $j = 14$; $k = 1$; $\eta = 1,1$, із середнім значенням $MSE = 0,0079$ після в середньому 1005 ітерацій. Графік, що ілюструє результати прогнозування логістичних витрат машинобудівного підприємства за допомогою ШНМ з рекомендованими параметрами, наведено на рис. 2.

Висновки з цього дослідження і перспективи подальших розвідок у даному напрямку. У статті було досліджено можливість адаптації алгоритму навчання та використання ШНМ для прогнозування логістичних витрат машинобудівних підприємств. Пропонується використання мережі з одним прихованим шаром, яка може бути охарактеризована такою конфігурацією: $i = 12$; $j = 14$; $k = 1$; $\eta = 1,1$. За допомогою використання масиву даних щодо щомісячних логі-

стичних витрат машинобудівного підприємства визначено, що за ~ 1000 ітерацій навчання така мережа досягає заданої точності $MSE \leq 0,01$ при тестах на підмножині, що не входила до навчального набору даних. Перевагою запропонованого підходу є висока точність та можливість самостійного навчання системи з використанням рекомендованої ШНМ у разі її інтеграції в інформаційну систему управління підприємством. Реалізація запропонованої ШНМ може бути впроваджена у вигляді комп'ютерної програми чи модуля до наявної системи на широкому діапазоні машинобудівних підприємств.

Подальші дослідження у цій сфері можуть бути пов'язані із визначенням можливості додавання до вхідної інформації ШНМ не тільки послідовного ряду чисел, що відповідають витратам, але й відповідних значень різноманітних факторів, що впливають на ці витрати.

БІБЛІОГРАФІЧНИЙ СПИСОК:

1. Schmidhuber J. Deep learning in neural networks: An overview / J. Schmidhuber // *Neural Networks*. – 2015. – vol. 61. – P. 85–117.
2. Калініна І.О. Дослідження нейромережових методів у задачах прогнозування / І.О. Калініна // *Наукові праці Чорноморського державного університету імені Петра Могили. Серія «Комп'ютерні технології»*. – 2009. – № 106, Вип. 93. – С. 132–138.
3. Чумаченко Е.И. Алгоритм решения задачи прогнозирования / Е.И. Чумаченко, В.С. Горбатюк // *Штучний інтелект*. – 2012.
4. Кудрицька Н.В. Прогнозування розвитку транспортно-дорожнього комплексу України за допомогою нейронних мереж / Н.В. Кудрицька // *Економіко-математичне моделювання соціально-економічних систем*. – 2014. – № 19. – С. 198–207.
5. Савка Н.Я. Застосування штучних нейронних мереж з радіально-базисними функціями для розв'язування задач прогнозування / Н.Я. Савка // *Коммунальное хозяйство городов*. – 2011. – № 97. – С. 349–353.
6. Carbonneau R. Application of machine learning techniques for supply chain demand forecasting / R. Carbonneau, K. Laframboise, R. Vahidov // *European Journal of Operational Research*. – 2008. – Vol. 184. – № 3. – P. 1140–1154.
7. Kaastra I. Designing a neural network for forecasting financial and economic time series / I. Kaastra, M. Boyd // *Neurocomputing*. – 1996. – Vol. 10. – № 3. – P. 215–236.
8. Oancea B. Time series forecasting using neural networks / B. Oancea, Ş. C. Ciucu // *Proceedings of the CKS 2013 International Conference*. – 2014.
9. Crone S.F. Artificial neural networks for time series prediction-A novel approach to inventory management using asymmetric cost functions. / S.F. Crone. // *IC-AI*. – 2003. – P. 193–199.
10. Heaton J. Programming Neural Networks with Encog3 in Java / J. Heaton. – St. Louis, MO, USA: Heaton Research, Inc. , 2011. – 242 p.
11. Sheela K.G. Review on methods to fix number of hidden neurons in neural networks / K.G. Sheela, S.N. Deepa // *Mathematical Problems in Engineering*. – 2013.
12. Sirisaengtaksin O. Sigmoid neurons are the safest against additive errors / O. Sirisaengtaksin, V. Kreinovich, H.T. Nguyen. – 1995.
13. LeCun Y.A. Efficient backprop / Y.A. LeCun, L. Bottou, G.B. Orr, K.-R. Müller // *Neural networks: Tricks of the trade*. – Springer, 2012. – P. 9–48.
14. Rojas R. Neural Networks: A Systematic Introduction / R. Rojas. – Springer Science & Business Media, 2013. – 511 p.