УДК 004.032.26

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
Факультет інформатики та обчислювальної техніки
Кафедра технічної кібернетики

«До захисту допущено»

Завідувач кафедри

_____ _____
(підпис) (ініціали, прізвище)

“___”_____20__ р.

Дипломна робота

освітньо-кваліфікаційного рівня «Магістр»
(назва ОКР)

з напряму підготовки (спеціальності) 8.05020102 «Комп’ютеризовані та робототехнічні системи» _____
_____

на тему: Побудова та дослідження інтелектуальних систем прогнозування _____
_____
_____

Виконав (-ла): студент (-ка) VI курсу, групи ІК-31м
(шифр групи)

Горбатюк Владислав Сергійович _____ _____
(прізвище, ім’я, по батькові) (підпис)

Керівник к.т.н., доцент, Чумаченко Олена Іллівна _____ _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант ОППЦБ доцент кафедри «ОППЦБ» Праховнік Н .А. ___ _____
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент_____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Рецензент_____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цьому дипломному проекті
немає запозичень з праць інших авторів
без відповідних посилань.
Студент _____
(підпис)

Київ – 2015 року

Національний технічний університет України «Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки
Кафедра технічної кібернетики

Освітньо-кваліфікаційний рівень «Магістр»

Напрям підготовки 8.050201 «Системна інженерія»
Спеціальність 8.05020102 «Комп'ютеризовані та робототехнічні системи»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Ткач М. М.
(підпис) (ініціали, прізвище)
«___»_____20__ р.

ЗАВДАННЯ
на дипломну роботу студента
Горбатюка Владислава Сергійовича
(прізвище, ім'я, по батькові)

1. Тема роботи Побудова та дослідження інтелектуальних систем прогнозування

керівник роботи к.т.н., доцент, Чумаченко О. І.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від « 10 » березня 2015 р. № 292/2С

2. Строк подання студентом роботи 02.06.2015 р.

3. Вихідні дані до роботи технічні специфікації програмної системи прогнозування; список завдань, що система повинна виконувати; вимоги до методів прогнозування (а саме: середньоквадратична похибка прогнозу на попередньо визначених тестових даних має бути не більшою за похибку, отриману за допомогою МГУА та ШНМ)

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити) аналіз проблем, що виникають при вирішенні задачі прогнозування; аналіз існуючих методів прогнозування; розроблення нових методів прогнозування; дослідження структури програмної системи; Проектування інтерфейсів програмної системи

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) структурна схема входів та виходів задач, виконуваних системою; структурна схема алгоритмів задач системи; структурна схема системи (загальна).

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	доцент кафедри «ОППЦБ» Праховнік Н .А.		

7. Дата видачі завдання 26.09.2013

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Строк виконання етапів роботи	Примітка
1.	Вибір теми дипломної роботи	22-24 вересня 2013 р.	
2.	Постановка задачі та розробка плану роботи	25 вересня – 17 жовтня 2013 р.	
3.	Збір та аналіз матеріалу	2–12 листопада 2013 р.	
4.	Розробка нових методів прогнозування	20 листопада – 18 грудня 2013 р.	
5.	Проектування програмної системи	21 січня – 20 лютого 2014 р.	
6.	Розробка та тестування програмної системи	10-25 березня 2015 р.	
7.	Оформлення пояснювальної записки	2 квітня – 15 травня 2015 р.	
8.	Оформлення розділу з охорони праці	16-30 травня 2015 р.	

Студент _____ _____
(підпис) (ініціали, прізвище)

Керівник роботи _____ _____
(підпис) (ініціали, прізвище)

АННОТАЦІЯ

Пояснювальна записка дипломної роботи складається з 6 розділів, 17 рисунків, 3 таблиць, 6 додатків, 55 джерел.

Метою цієї дипломної роботи є дослідження існуючих інтелектуальних методів та систем прогнозування і розробка нових для покращення якості прогнозу. Об’єктом дослідження є системи прогнозування в цілому.

У дипломній роботі було розглянуто основні проблеми, що виникають при вирішенні задачі прогнозування. На основі аналізу проблем були розроблені нові алгоритми прогнозування, що були перевірені на реальних даних і показали якість прогнозу (у вигляді середньоквадратичної помилки прогнозу на тестових даних) кращу, ніж у таких методів, як штучні нейронні мережі, лінійна регресія та інші. Також було розроблено програмний комплекс прогнозування часових рядів, що дозволяє будувати складні багаторівневі системи прогнозування.

Розроблені методи прогнозування та програмний комплекс прогнозування можна використовувати для різноманітних практичних задач, в тому числі: прогнозування попиту на виробництві, прогнозування ризиків, прогнозування фінансових показників та інші.

Ключові слова:

ШТУЧНІ НЕЙРОННІ МЕРЕЖІ, ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ, МЕТОД ГРУПОВОГО УРАХУВАННЯ АРГУМЕНТІВ, ПРОГНОЗУВАННЯ РИЗИКІВ, АВТОМАТИЗОВАНИЙ ПРОГРАМНИЙ КОМПЛЕКС ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

ABSTRACT

Explanatory note of the degree work consists of 6 chapters containing 17 figures, 3 tables, 6 appends and 55 references.

The aim of this work is to study existing intelligent forecasting methods and systems and develop new ones to improve the forecast quality. The object of the study is forecasting system as a whole.

In the thesis work the main problems arising when solving the forecasting problem were considered. Based on the problem analysis new algorithms were developed and tested on the real data showing better forecasting accuracy (in a form of mean squared error on a test data) compared to the methods like artificial neural networks, linear regression and others. Also, the software package for time series forecasting was designed and developed. The package is capable of building complex multilevel forecasting systems.

The developed forecasting methods and software package can be used for a variety of practical tasks including: demand forecasting, risk prediction, forecasting of financial indicators and others.

Key words:

ARTIFICIAL NEURAL NETWORKS, TIME SERIES PREDICTION, GROUP METHOD OF DATA HANDLING, RISK PREDICTION, AUTOMATED SOFTWARE PACKAGE FOR TIME SERIES FORECASTING

Пояснювальна записка
до дипломного проекту
на тему: Побудова та дослідження інтелектуальних систем прогнозування

Київ – 2015 року

ЗМІСТ

ВСТУП

Прогнозування є одним з найбільш складних і затребуваних завдань, що цікавлять людство. Людям завжди було цікаво знати, що ж чекає їх у майбутньому. Однак, не тільки інтерес робить це завдання настільки затребуваним – в умовах глобалізації та прискорення ритму життя, коли за долю секунди відбуваються тисячі угод, можливість як-небудь передбачити розвиток ринку хоча б на найближчий час може стати результатом мільйонних прибутків. З іншого боку, дії, засновані на помилковому прогнозі, можуть призвести до ще більших втрат. Усе це свідчить про високу **актуальність** задачі прогнозування.

З точки зору математики, завдання прогнозування зводиться до задачі ідентифікації об'єкта – де об'єктом є прогнозований процес. Тобто передбачається, що прогнозований процес має деяку «модель поведінки», яка взаємодіє із зовнішнім світом і, знаючи яку ми б змогли оцінити виходи моделі в майбутньому. Тут важливим є слово «оцінити» – воно спочатку попереджає, що ми можемо лише передбачити деяку оцінку майбутніх виходів моделі.

На сьогоднішній день розроблений дуже солідний математичний апарат для вирішення задачі прогнозування. Однак навряд чи можна назвати єдиний метод прогнозування, який варто застосовувати при вирішенні усього розмаїття можливих практичних завдань прогнозування – кожен метод найкраще підходить для вирішення деякого кола завдань, що володіють певними властивостями. Найбільш універсальними виявилися методи, що використовують елементи штучного інтелекту – інтелектуальні системи прогнозування. Найвідоміший представник цієї групи методів – штучні нейронні мережі (ШНМ), що є системою взаємозалежних штучних нейронів. Кількість завдань, для вирішення яких були успішно застосовані ШНМ, дійсно вражає, як і різноманітність цих завдань: завдання

прогнозування, розпізнавання, управління і багато інших входять в цей список. Саме тому головним фокусом цієї роботи є ШНМ.

Метою роботи є дослідження існуючих інтелектуальних методів та систем прогнозування і розробка нових для покращення якості прогнозу.

Об'єктом дослідження є системи прогнозування в цілому. **Предмет дослідження** – шляхи покращення існуючих методів та систем прогнозування.

Основні завдання роботи:

- дослідження та аналіз сучасних інтелектуальних методів прогнозування;
- розробка нових підходів до прогнозування на основі використання елементів штучного інтелекту;
- розробка програмного комплексу, що забезпечуватиме просте використання існуючих та розроблених методів для вирішення задачі прогнозування часових рядів.

Наукова новизна роботи:

- запропонована нова постановка задачі прогнозування, що враховує можливу неоднорідність даних;
- запропоновано декілька нових гібридних методів прогнозування, заснованих на ШНМ;
- запропонована методика модифікації методів прогнозування, які володіють певними властивостями, що поліпшує точність прогнозу цих методів на неоднорідних даних.

Крім цього, була реалізована спеціалізована програмна система для вирішення задачі прогнозування.

Результати, які мають наукову новизну, були представлені в наступних журналах і конференціях:

- стаття «An algorithm for solving the problem of forecasting» в журналі Aviation (входить в бібліографічну базу даних SCOPUS), 2013 [1];

- стаття «Using a mixture of experts' approach to solve the forecasting task» в журналі Aviation, 2014 [2];
- стаття «A method for building a forecasting model with dynamic weights» в журналі «Eastern-European Journal of Enterprise Technologies» (входить до бібліографічної бази даних CiteFactor, Index Copernicus та інших), 2014 [3];
- стаття «Алгоритм решения задачи прогнозирования» в журналі «Искусственный интеллект», 2012 рік [4];
- та інші – [5] – [11].

РОЗДІЛ 1. ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

1.1. Постановка задачі

У більшості постановок задачі прогнозування вважається, що значення прогнозованого процесу $y(t)$ в будь-який момент часу t однозначно визначається як деяка функція h від виходу істинної моделі процесу в цей момент часу $f(t)$, і деякої випадкової складової $\varepsilon(t)$:

$$y(t) = h(f(t), \varepsilon(t)).$$

Найчастіше, у якості функції h використовують або функцію підсумовування $h(x, y) = x + y$, або мультиплікації $h(x, y) = x * y$.

Передбачається, що випадкова величина (ВВ) це білий шум [12], що має кінцеву дисперсію, але будь-які інші характеристики даної ВВ нам невідомі. У разі наявності будь-якої додаткової інформації про $\varepsilon(t)$ її слід обов'язково враховувати, оскільки це може значно поліпшити точність прогнозу.

Розглянемо основні постановки задачі прогнозування часових рядів.

1.1.1. Регресійна постановка задачі прогнозування

Задана тимчасова вибірка даних виду $\vec{x}^T = \{x(t)\}, t = 1, \dots, T; x(t) \in \mathbb{R}$, де x – прогнозований випадковий процес. Також можуть бути задані вибірки $\overrightarrow{z_i}$ с випадкових процесів z_i: $\overrightarrow{z_i}^T = \{z_i(t)\}, t = 1, \dots, T; i = 1, \dots, N$, від яких, можливо, залежить прогнозований процес. Ці додаткові процеси називаються екзогенними процесами або змінними. Важливо, що прогнозована величина є безперервною, у той час як екзогенні змінні можуть бути будь-якого типу: безперервні, дискретні, нечіткі і т.д. Потрібно побудувати модель процесу, що залежить як від значень самого прогнозованого процесу, значень процесів, що впливають на прогнозований, так і від часу t, яка б з задовільною точністю давала оцінку істинного значення $x(T + T_y)$:

$$\hat{x}_{T+T_y} = f(\vec{x}, \overrightarrow{z_1}, \dots, \overrightarrow{z_N}, t),$$

де $\hat{x}_{T+T_y}$ – оцінка моделі, а область значень моделі – вся множина дійсних чисел:

$$E(f) = \mathbb{R}.$$

1.1.2. Постановка задачі прогнозування як задачі класифікації

Класифікація вибірки даних означає поділ її на класи, число яких визначено заздалегідь – тобто в результаті класифікації кожна точка з вибірки буде віднесена до певного класу. Класифікацію можна розглядати як дискретний опис об'єкта.

Завдання класифікації вибірки даних може виникати при прогнозуванні деякої дискретної величини, що має обмежений набір значень.

Наприклад, потрібно передбачити наявність у пацієнта раку. У даному випадку мається навчальна вибірка даних про пацієнтів, що включає обидва класи пацієнтів: ті, у кого є рак, і ті, у кого немає, і потрібно визначити приналежність даного конкретного пацієнта до одного з класів.

Також, при прогнозуванні сильно зашумленої величини, зважаючи на більшу складність прогнозу, можливою є розбивка набору всіх прийнятих на вихідній вибірці значень на класи, і подальше прогнозування не самого значення величини, а його приналежності до деякого класу.

Вихідні дані ті ж, що й у регресійній постановці завдання прогнозування, тільки прогнозований процес є дискретним (як і будь-які його вибірки): $\vec{x} = \{x(t)\}, t = 1, \dots, T; x(t) \in C_k; k = 1, \dots, K$, де C_k – k-ий клас, K – кількість класів.

Потрібно побудувати модель процесу, що залежить від наявної інформації, яка б визначала належність до певного класу деякого майбутнього значення в момент часу: $T + T_y$:

$$\hat{x}_{T+T_y} = f(\vec{x}, \overrightarrow{z_1}, \dots, \overrightarrow{z_N}, t),$$

де область значень моделі – множина всіх можливих класів:

$$E(f) = \{1, \dots, K\}.$$

1.1.3. Задача безмодельного прогнозування процесів

Як було відмічено в попередніх постановках, стан складного об'єкта може описуватися вектором характеристичних змінних представлених у вибірці даних спостережень, де одне спостереження в момент часу t буде мати вигляд $\overrightarrow{o_t} = [z_1(t), z_2(t), \dots, z_N(t)]^T$, а прогнозованою величиною $x(t)$ буде вихід об'єкта. І якщо розмірність вектора характеристичних змінних велика, то для здійснення прогнозу за поточним спостереженням $\overrightarrow{o_t}$ може виявитися достатнім використання методу аналогії: необхідно знайти один або кілька близьких спостережень в передісторії і подивитися що було далі. При цьому модель об'єкта не потрібна, її роль виконує сам об'єкт. Прогноз за аналогами з передісторії розроблявся головним чином в метеорології для прогнозу погоди. Формалізована постановка задачі безмодельного прогнозування за аналогією.

1. Є навчальна вибірка, що складається з N векторів спостережень характеристичних змінних об'єкта $\overrightarrow{o_t} = [z_1(t), z_2(t), \dots, z_N(t)]^T, t = 1, \dots, N$ та N та відповідних значень виходу об'єкта $x(t), t = 1, \dots, N$.

2. Необхідно задати::

а. Критерій близькості спостережень $L(\overrightarrow{o_i}, \overrightarrow{o_j}), i, j = 1 \dots, T; L \geq 0$ – зазвичай його задають у вигляді відстані, і чим менша відстань – тим більша близькість.

б. Кількість аналогів k, що враховуються при прогнозі. Під аналогами маються на увазі спостереження, найближчі до заданого. У результаті, матимемо вектор з k векторів спостережень:

$$\vec{r} = \left[\overrightarrow{o_{w_1}}, \dots, \overrightarrow{o_{w_k}}\right]^T,$$

де $\vec{w} = [w_1, \dots, w_k]^T$ – вектор індексів найближчих k спостережень в наявній вибірці.

в. Функцію «усереднення» виходів об'єкта для знайдених аналогів:

$$\widehat{x_l} = f(x_{w_1}, x_{w_2}, \dots, x_{w_k}),$$

яка буде використовуватися для отримання остаточного прогнозу.

1.2. Огляд методів

Наведемо огляд основних методів, які розроблені і застосовуються для вирішення задачі прогнозування.

1.2.1. Регресійний аналіз

У разі застосування регресійного аналізу [13] для вирішення задачі прогнозування регресором буде або час t, або деякі незалежні змінні$\boldsymbol{z}_i$, а а критеріальною (залежною) змінною – сам прогнозований процес.

У випадку, коли регресором є час, вихідними даними є вибірка $\boldsymbol{x} = \{x_t\}, t = 1, \dots, T,$ і потрібно знайти рівняння регресії виду $x = f(t)$. У найпростішому випадку використовується лінійна регресія виду $x = b_0 + b_1 * t + b_2 * t^2 + \cdots + b_n * t^n,$ де вектор коефіцієнтів $\boldsymbol{b} = [b_0, \dots, b_n]^T$ знаходиться за допомогою методу найменших квадратів [13]. Відповідно, для прогнозування значення процесу в деякий момент часу $t_{T+T_y},$ потрібно просто підставити цей час в рівняння регресії.

У другому випадку вихідними даними буде вибірка виду $Z = \{z_{ik}\};\ i = 1, \dots, N; k = 1, \dots, K,$ і набір відповідних значень $\boldsymbol{x} = \{x_k\}$, ге N – N – кількість незалежних змінних, K – кількість точок у вибірці (кожен вектор виду $[z_{1k}, \dots, z_{Nk}]^T$ є однією точкою вибірки). Потрібно знайти рівняння регресії виду $x = f(\boldsymbol{z_1}, \dots, \boldsymbol{z}_N)$. Знову ж таки, у загальному випадку використовується лінійна регресія $x = b_0 + b_1 * \boldsymbol{z_1} + b_2 * \boldsymbol{z_2} + \cdots + b_N * \boldsymbol{z_N},$ де коефіцієнти знаходяться аналогічно. Для прогнозування значення $x(z_1, \dots, z_N)$ при деяких певних значеннях незалежних змінних $z_1, \dots, z_N$ потрібно підставити ці значення в рівняння регресії.

Згідно з теоремою Гаусса-Маркова, якщо виконується певний набір умов, то оцінки параметрів лінійної моделі, отримані за допомогою методу найменших квадратів, будуть ефективні (тобто будуть володіти найменшою дисперсією) в класі незміщених оцінок (тобто таких оцінок, математичне очікування яких дорівнює істинним значенням параметрів). Ось 5 умов,

необхідних для отримання ефективних оцінок при використанні методу найменших квадратів:

Перша умова: правильно задана модель, тобто істинна залежність між вхідними і вихідною змінними дійсно лінійна, помилка впливає на вихід моделі за адитивним законом і відсутня недовизначеність (упущені важливі фактори) або перевизначеність (враховані непотрібні фактори).

Друга умова: всі вхідні змінні детерміновані і не всі рівні між собою.

Третя умова: помилки не носять систематичного характеру. Випадковий член може бути іноді позитивним, іноді негативним, але він не повинен мати систематичного зсуву ні в якому з двох можливих напрямків. Якщо рівняння регресії включає постійний член, то ця умова найчастіше виконується автоматично, так як постійний член відображає будь-яку систематичну, але постійну складову у вихідної змінної, якою не враховують пояснюючі змінні, включені в рівняння регресії.

Четверта умова: дисперсія помилок однакова. Однаковість дисперсії помилок також прийнято називати гомоскедастичністю. Не повинно бути апріорної причини для того, щоб випадковий член породжував більшу помилку в одних спостереженнях, ніж в інших. Так як $E(\varepsilon_i) = 0\ \forall i$ і теоретична дисперсія відхилень ε_i дорівнює $E(\varepsilon_i^2)$, то цю умову можна записати так: $E(\varepsilon_i^2) = \sigma_\varepsilon^2\ \forall i$. Одне із завдань регресійного аналізу полягає в оцінці стандартного відхилення випадкового члена. Якщо умова, що розглядається, не виконується, то коефіцієнти регресії, знайдені за методом найменших квадратів, будуть неефективні, а більш ефективні результати будуть виходити шляхом застосування модифікованого методу оцінювання (зважений МНК або оцінка коваріаційної матриці за формулою Уайта [14] або Девідсона-Маккінона).

П'ята умова: ε_i розподілені незалежно від ε_j при $i \neq j$. Ця умова передбачає відсутність систематичного зв'язку між значеннями випадкового члена в будь-яких двох спостереженнях. Якщо один випадковий член

великий і позитивний в одному напрямку, не повинно бути систематичної тенденції до того, що він буде таким же великим і позитивним (те ж можна сказати і про малі, і про негативні залишках). Теоретична коваріація $\sigma_{\varepsilon_i,\varepsilon_j}$ повинна дорівнювати нулю, оскільки:

$$\sigma_{\varepsilon_i,\varepsilon_j} = E\{[\varepsilon_i - E(\varepsilon_i)][\varepsilon_j - E(\varepsilon_j)]\} = E(\varepsilon_i * \varepsilon_j) - E(\varepsilon_i) * E(\varepsilon_j) = 0.$$

Теоретичні середні для ε_i і ε_j дорівнюють нулю в силу третьої умови теореми. При невиконанні цієї умови оцінки, отримані за методом найменших квадратів, будуть також неефективні.

Перевагами використання лінійної регресії як моделі і методу найменших квадратів для знаходження параметрів цієї моделі є простота, обчислювальна та практична ефективність. Під практичною ефективністю розуміється гарна якість прогнозу (з точки зору мінімуму середньоквадратичної помилки (СКП)), отриманого з використанням цього методу для багатьох реальних задач прогнозування.

Недоліками даного підходу є необхідність повного перерахунку параметрів при появі нової інформації і необхідність явного вибору моделі прогнозування – при невідповідності обраної та істинної моделі отримані параметри будуть зміщеними – тобто їх математичне очікування буде відрізнятися від дійсних значень параметрів. Для ілюстрації цього недоліку розглянемо наступний приклад.

1) Згенеруємо навчальні дані, задавши справжню модель як $y = 0.8 * x + 2$, для всіх $x = 1,2,\dots,100$, додаючи нормальну випадкову перешкоду ε з параметрами $E(\varepsilon) = 0, \sigma(\varepsilon) = 5$. Графік одного з нескінченної кількості можливих прикладів буде виглядати наступним чином (рис. 1.1):

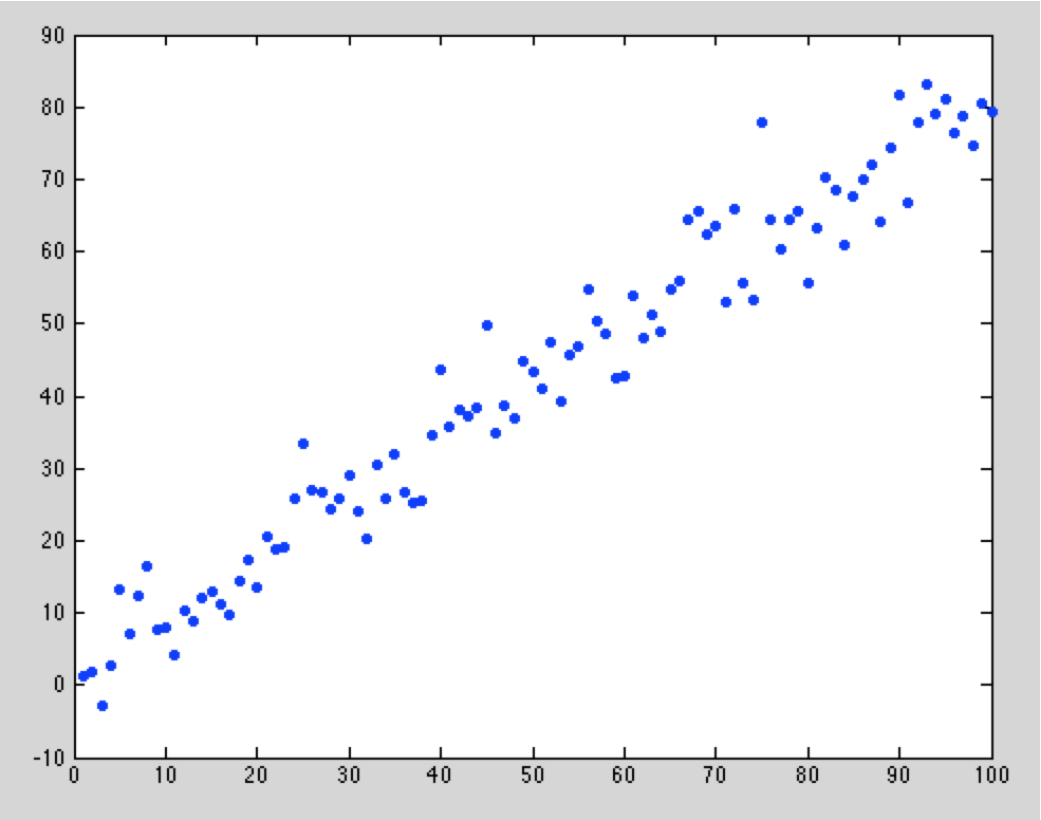

Рис. 1.1. Графік згенерованих навчальних даних

2) Для згенерованого прикладу отримаємо параметри двох моделей, використовуючи метод найменших квадратів:

$$y_1 = a_{11} * x + a_{10},$$

$$y_2 = a_{23} * x^3 + a_{22} * x^2 + a_{21} * x + a_{20}.$$

3) Запам'ятаємо отримані параметри, і повторимо пункти 1-2 ще 9999 разів, щоразу запам'ятовуючи параметри.

4) Усереднимо отримані вектори параметрів за всіма експериментами. Таким чином, ми отримаємо оцінку математичного очікування параметрів обраних моделей. Результати:

$$a_{11} = 0.8, a_{10} = 2.0039$$

$$a_{23} = -0.00001, a_{22} = 0.0000, a_{21} = 0.7985, a_{20} = 2.0211$$

Як добре видно з результатів, математичне очікування оцінок параметрів моделі y_2 дійсно злегка зміщене від істинних значень, у той час як математичне очікування оцінок параметрів моделі y_1 (яка збігається з істинною) збігається з істинними значеннями параметрів.

1.2.2. ARIMA

Інтегрована модель авторегресії-ковзаючого середнього (модель Бокса-Дженкінса, англ. AutoRegressive Integrated Moving Average) [15] – одна з математичних моделей, що використовуються для аналізу і прогнозування стаціонарних часових рядів в статистиці. Є розширенням моделей ARMA (AutoRegressive Moving Average) [15] для нестаціонарних часових рядів, які можна зробити стаціонарними взяттям різниць деякого порядку від вихідного часового ряду (так звані інтегровані або різнецевоо-стаціонарні часові ряди). Модель $ARIMA(p, d, q)$ означає, що різниці часового ряду порядку d підкоряються моделі $ARMA(p, q)$.

Моделлю $ARMA(p, q)$, де p і q – цілі числа, що задають порядок моделі, називається наступний процес генерації часового ряду $\{x_t\}$:

$$x_t = c + \sum_{i=1}^{p} \alpha_i * x_{t-i} + \sum_{i=1}^{q} \beta_i * \varepsilon_{t-i},$$

де c – константа, $\{\varepsilon_t\}$ – білий шум, тобто послідовність незалежних і однаково розподілених випадкових величин (як правило, нормальних), з нульовим середнім, а $\alpha_1, \dots, \alpha_p$ і $\beta_1, \dots, \beta_q$ – дійсні числа, авто регресійні коефіцієнти і коефіцієнти змінного середнього, відповідно.

Модель $ARIMA(p, d, q)$ для нестаціонарного часового ряду X_t має вигляд:

$$\Delta^d X_t = c + \sum_{i=1}^{p} a_i \, \Delta^d X_{t-i} + \sum_{j=1}^{q} b_j \, \varepsilon_{t-i} + \varepsilon_t$$

де ε_t – стаціонарний часовий ряд;

c, a_i, b_j – параметри моделі;

Δ^d – оператор різниці часового ряду порядку d (послідовне взяття d разів різниць першого порядку – спочатку від часового ряду, потім від отриманих різниць першого порядку, потім від другого порядку і т.д.).

Також дана модель інтерпретується як $ARMA(p + d, q)$-модель з d одиничними рішеннями. При $d = 0$ маємо звичайні ARMA-моделі.

ARIMA-моделі дозволяють моделювати інтегровані або разнецево-стаціонарні часові ряди (DS-ряди, diference stationary).

Часовий ряд X_t називається інтегрованим порядку k (зазвичай пишуть $X_t \sim I(k)$), якщо різниці ряду порядку k, тобто $\Delta^d X_t$ є стаціонарними, у той час як різниці меншого порядку (включаючи нульового порядку, тобто сам часовий ряд) не є стаціонарними щодо деякого тренду рядами (TS-рядами, trend stationary). Зокрема $I(0)$ – це стаціонарний процес.

Порядок інтегрованості часового ряду і є порядок d моделі $ARIMA(p, d, q)$.

Для побудови ж моделі ARMA за серією спостережень необхідно визначити порядок моделі (числа p і q), а потім і самі коефіцієнти. Для визначення порядку моделі може застосовуватися дослідження таких характеристик часового ряду, як його автокореляційна функція і приватна автокореляційна функція. Для визначення коефіцієнтів застосовуються такі методи, як метод найменших квадратів і метод максимальної правдоподібності [16].

1.2.3. Метод групового урахування аргументів

Метод групового урахування аргументів (МГУА) [17] – сімейство індуктивних алгоритмів для математичного моделювання мультипараметричних даних. Метод заснований на рекурсивному селективному відборі моделей, на основі яких будуються більш складні моделі. Точність моделювання на кожному наступному кроці рекурсії збільшується за рахунок ускладнення моделі.

Найпростіший ітераційний багаторядний алгоритм МГУА полягає в наступних етапах.

1. Маємо дані спостережень: $\vec{x}, y$. Необхідно побудувати найкращу в певному сенсі модель $Y(x_1, \dots, x_n)$.

2. Важливим етапом МГУА є розбиття вихідної вибірки $\vec{x}$ на дві (в деяких випадках три) підвибірки: навчальну і перевірочну (іноді ще

виділяють екзаменаційну вибірку). Навчальна вибірка буде надалі використовуватися для знаходження параметрів моделей, перебираються, а перевірочна – для перевірки їх придатності та відбору найкращих моделей (за наявності екзаменаційної вибірки на ній буде перевірятися остаточна модель). Існує кілька основних способів розділення даних на перевірочну і навчальну вибірки.

- У навчальну вибірку відбираються перші $n * k$ прикладів, n – загальна кількість прикладів, k – деякий коефіцієнт, $0 < k < 1$; в перевірочну вибірку – усі приклади, що залишилися.
- Всі приклади ранжуються за дисперсією, і в навчальну вибірку відбираються $n * k$ вузлів з найбільшою дисперсією, а в перевірочну – решта вузлів.
- Вузли відбираються в перевірочну і навчальну вибірку через один. У загальному випадку, те, яким чином вихідна вибірка розбивається на підвибірки, може значно вплинути на прогнозуючі якості майбутньої моделі

3. Вибирається загальний вигляд моделей, що перебираються, так звані опорні функції. Часто використовується поліном Колмогорова-Габора:

$$Y(x_1, \dots, x_n) = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ijk} x_i x_j x_k + \cdots$$

4. Використовуючи опорні функції будуються різні варіанти моделей для деяких або всіх аргументів. Наприклад, будуються поліноми з однією змінною, поліноми з усіма можливими парами змінних, поліноми з усіма можливими трійками змінних, і т.д., поліном з усіма змінними. Для кожної моделі визначаються її коефіцієнти методом регресійного аналізу.

5. Серед усіх моделей вибираються кілька найкращих. Якість моделей визначається коефіцієнтом детермінації, або средньоквадратичним відхиленням помилки, або кореляцією Y і вихідних даних.

6. Якщо знайдена достатньо «хороша» модель або досягнута максимально допустима складність моделей, то алгоритм закінчується. Інакше, знайдені на 3-му кроці моделі використовуються як аргументи $(x_1, \dots, x_n)$ для опорних функцій наступного етапу ітерації (перехід на 2-ий пункт). Тобто вже знайдені моделі беруть участь в формуванні більш складних.

Зазвичай ступінь полінома опорної функції вибирається не вище N-1, де N – кількість точок вибірки. Часто буває достатнім використати в якості опорних функції поліноми другого ступеня. У такому випадку на кожному кроці ітерації ступінь результуючого полінома подвоюється.

1.2.4. Штучні нейронні мережі

Штучні нейронні мережі (ШНМ) [18] являють собою систему з'єднаних і взаємодіючих між собою простих процесорів – штучних нейронів.

Нейронні мережі не програмуються у звичному розумінні цього слова, вони навчаються. З математичної точки зору, навчання нейронних мереж – це багатопараметричне завдання нелінійної оптимізації. Можливість навчання – одна з головних переваг нейронних мереж в порівнянні з традиційними алгоритмами. Технічно навчання полягає в знаходженні коефіцієнтів зв'язків між нейронами. У процесі навчання нейронна мережа здатна виявляти складні залежності між вхідними даними і вихідними, а також виконувати узагальнення.

Здібності нейронної мережі до прогнозування безпосередньо випливають з її здатності до узагальнення і виділення прихованих залежностей між вхідними та вихідними даними.

Великою перевагою ШНМ перед іншими методами прогнозування є непотрібність явно задавати рівняння моделі прогнозованого процесу, що значно спрощує процес прогнозування, а також прибирає можливість вибору неправильного рівняння, яке погано б відповідало реальному процесу.

Другим вагомим аргументом у бік вибору ШНМ для вирішення задачі прогнозування є їх універсальність: апарат нейронних мереж успішно використовується як для прогнозу часових рядів, так і для прогнозу значень залежної змінної від вектора значень незалежних змінних, кластеризації вибірки даних і багатьох інших завдань. Це дозволяє використовувати єдиний математичний апарат ШНМ для вирішення декількох різних завдань прогнозування (схожою універсальністю володіє МГУА, але в ньому потрібно явно задавати рівняння приватних описів).

Виходячи з вищевикладеного, у якості основних методів для дослідження були обрані ШНМ і МГУА – обидва методи дозволяють знаходити нелінійні залежності між даними без необхідності явного задавання моделі.

1.3. Основні проблеми, що виникають при вирішенні задачі прогнозування

Вирішення задачі прогнозування супроводжується великою кількістю різноманітних проблем, які, власне, і роблять цю задачу складною. Перечислимо основні:

1) неможливо врахувати всі чинники, що впливають на процес, який ми намагаємося прогнозувати; більше того, їх вплив може змінюватися з плином часу - фактор, що не був важливий сьогодні, може відіграти важливу роль завтра;
2) завжди є багато (іноді нескінченне число) можливих моделей, що добре відповідають навчальним даними - ми повинні вирішити, яку модель або множину моделей використовувати, і такі рішення зазвичай дуже схильні до помилок;
3) часто буває важко (якщо не неможливо) знайти оптимальну складність моделі.

Якщо проаналізувати перераховані існуючи методи, то можна зробити висновок, що більшість методів ніяк не вирішують описані проблеми –

тільки МГУА певним чином вирішує третю проблему, знаходячи оптимальну складність обраної моделі шляхом перебору усіх можливих «варіацій» моделі починаючи з самої простої.

Висновки до розділу

У цьому розділі було розглянуто основні постановки задач прогнозування разом з основними існуючими методами прогнозування. На основі аналізу методів було обрано найбільш перспективні методи для подальшого дослідження: ШНМ та МГУА, оскільки обидва методи дозволяють знаходити нелінійні залежності між вхідними та вихідними змінними та не потребують явного вибору істинної моделі об’єкту.

РОЗДІЛ 2. АНАЛІЗ ДАНИХ ПРИ ВИРІШЕННІ ЗАДАЧІ ПРОГНОЗУВАННЯ

2.1. Моделі випадкових перешкод

Як було сказано при розгляді постановок задачі прогнозування, вибірка, що використовується для прогнозування, майже завжди містить деяку випадкову перешкоду, яка взаємодіє з «чистим» сигналом за деяким законом (найчастіше адитивним або мультиплікативним). Далі розглянемо найбільш поширені моделі випадкових перешкод.

2.1.1. Білий шум

Білий шум – стаціонарний шум, спектральні складові якого рівномірно розподілені по всьому діапазону задіяних частот.

Безперервний у часі випадковий процес $w(t)$, де $t \in \mathbb{R}$, є білим шумом тоді і тільки тоді, коли його математичне очікування і автокореляційна функція задовольняють наступні рівняння відповідно:

$$\mu_w(t) = E\{w(t)\} = 0$$

$$R_{ww}(t_1, t_2) = E\{w(t_1)w(t_2)\} = \sigma^2\delta(t_1 - t_2)$$

Тобто, це випадковий процес з нульовим математичним очікуванням, що має автокореляційну функцію, яка є дельта-функцією Дірака. Така автокореляційна функція передбачає наступну спектральну щільність потужності:

$$S_{ww}(w) = \sigma_w^2$$

так як перетворення Фур'є дельта-функції дорівнює одиниці на всіх частотах. Через те, що спектральна щільність потужності однакова на всіх частотах, білий шум і отримав свою назву (за аналогією з частотним спектром білого світла).

Багато розроблених методів прогнозування та обробки часових рядів є оптимальними за умови, якщо перешкода, що впливає на чистий сигнал, є білим шумом (або деяким його підвидом, наприклад нормальним білим шумом). Як приклад можна привести метод найменших квадратів і фільтр

Калмана [19]. Саме тому, якщо вважається, що перешкода є білим шумом, зазвичай достатньо просто використовувати відповідний метод без будь-якої фільтрації.

2.1.2. Кольорові шуми

Рожевий шум

Рожевий шум або шум $\frac{1}{f}$ (іноді також званий флікер-шум) є сигналом або процесом з таким спектром частот, що спектральна щільність потужності (енергія або потужність на Гц) обернено пропорційна частоті сигналу. При рожевому шумі, кожна октава містить рівну кількість потужності шуму.

У науковій літературі термін рожевий шум іноді використовується трохи більш вільно і може відноситися до будь-якого шуму зі спектральною щільністю потужності у вигляді

$$S(f) \sim \frac{1}{f^{\alpha}}$$

де f – частота і $0 < \alpha < 2$, з показником α зазвичай близько до 1. Ці схожі на рожевий шуми широко поширені в природі.

Через те, що потужність рожевого шуму велика на низьких частотах, його практично неможливо відфільтрувати, не «зачепивши» чистий інформаційний сигнал, оскільки тренд інформаційного сигналу зазвичай «займає» низькі частоти.

Броунівський (червоний) шум

Спектральна щільність червоного шуму пропорційна $\frac{1}{f^2}$, де f – частота. Це означає, що на низьких частотах цей шум має навіть більше енергії, ніж рожевий шум.

Червоний шум може бути отриманий шляхом інтегрування білого шуму. Тобто, тоді як білий (дискретний) шум може бути отриманий шляхом випадкового вибору кожного відліку незалежно один від одного, червоний шум може бути отриманий шляхом додавання до кожного відліку сигналу випадкової величини для отримання наступного відліку:

$$x_k = x_{k-1} + \mu_k,$$

де μ_k – білий шум.

Чорний шум

Під чорним шумом зазвичай розуміють випадковий процес з спектральної щільністю виду $\frac{1}{f^\beta}$, де $\beta > 2$. Цю модель часто використовують для моделювання катастроф природного походження.

2.2. Первинна обробка даних

При вирішенні задачі прогнозування даний етап часто є важливішим від етапу складання моделі прогнозу. Неправильна обробка даних (або її відсутність) може значно погіршити прогнозуючі якості побудованої моделі, і навпаки.

Розглянемо постановку задачі попередньої обробки даних для подальшого прогнозування.

1) Постановка задачі

Задана вибірка даних виду $\boldsymbol{x} = \{x_t\}, t = 1, \dots, T$, де $\boldsymbol{x}$ – прогнозований процес, і передбачається, що значення прогнозованого процесу в будь-який момент часу однозначно визначається деякою функцією, залежною від значення істинної моделі процесу в цей момент час f_t, і деякої випадкової складової ε_t:

$$y_t = h(f_t, \varepsilon_t).$$

Загальна задача попередньої обробки даних розбивається на 2 підзадачі:

1) по можливості враховуючи інформацію про випадкову величину ε_t, спробувати таким чином перетворити вектор вихідних даних $\boldsymbol{x}$, щоб максимально позбавитися від випадкової складової;

2) враховуючи особливості обраного методу розв'язання задачі прогнозування, перетворити вектор $\boldsymbol{x}'$, отриманий внаслідок вирішення

попередньої підзадачі, щоб максимально збільшити ефективність використання обраного методу.

Особливо важливою є попередня обробка даних при використанні ШНМ в якості методу прогнозування. Правильна обробка може значно прискорити процес навчання мережі і навіть допомогти їй знайти більш глибокий мінімум СКП. Також, у роботі [20] показано вплив неврахованих викидів у вихідній вибірці даних на прогнозні властивості мережі (зокрема на СКП прогнозу на перевірочній вибірці, яка значно збільшується за наявності викидів в більш ніж 10% вихідних даних).

Далі будуть розглянуті основні методи попередньої обробки даних для ШНМ.

Поліпшення інформативності вибірки

Оскільки нейронна мережа навчається, то кожен приклад в навчальній вибірці даних вносить свій внесок у зміну ваг між нейронами мережі. Як наслідок, наявність великої кількості однакових прикладів призведе до того, що мережа фактично підлаштується під ці приклади. Тому для поліпшення інформативності вибірки вихідних даних слід виключити часто повторювані однакові приклади (якщо тільки не ставиться вимога незмінності вихідної вибірки).

Масштабування вихідних даних

Даний метод використовується при більшості основних методів прогнозування, оскільки він спочатку «зрівнює» вплив незалежних змінних на прогнозований процес.

Існує кілька основних метод центрування і нормалізації змінних.

1) $x_i^{\text{н}} = \frac{x_i - \bar{x}}{\sigma_x}$, де $x_i^{\text{н}}, x_i$ – нормовані і вихідні значення змінної відповідно, $\bar{x}$ – середнє значення, σ_x – середньоквадратичне відхилення.

2) $x_i^{\text{н}} = \frac{x_i - \bar{x}}{\bar{x}}$.

3) $x_i^{\text{н}} = \frac{x_i - x_{min}}{x_{max} - x_{min}}$, де x_{min}, x_{max} – мінімальне і максимальне значення змінної відповідно.

Згладжування вихідних даних

Згладжування вихідних даних застосовується для спроби позбавлення від випадкової складової і викидів в прогнозованому процесі. Існує безліч методів згладжування – перерахуємо основні.

Метод ковзаючого середнього

Метод [21] ґрунтується на простій моделі, яка передбачає, що поточне значення y_t, $(t = 1...n)$ ряду $\{y_1, y_2, ..., y_n\}$ є сумою середнього арифметичного деякої кількості попередніх значень і деякої випадкової складової:

$y_t = C + \varepsilon_t$, де: $C = \frac{1}{p} * \sum_{i=1}^{p} y_{t-i}$, де p – кількість врахованих попередніх значень, так званий «розмір передісторії»; ε_t – випадкова складова.

Експоненціальне згладжування

Наступним кроком у модифікації моделі є припущення про те, що більш пізні значення ряду більш адекватно відображають ситуацію. Тоді кожному значенню присвоюється вага тим більша, чим більше «свіже» значення додається:

$$y_t = C + \varepsilon_t,$$

$$C = \frac{\sum_{i=1}^{p} \alpha_i * y_{t-i}}{\sum_{i=1}^{p} \alpha_i},$$

де α_i – відповідні ваги значень.

Медіанний фільтр

Медіанний фільтр [21] – один з видів цифрових фільтрів, широко використовуваний в цифровій обробці сигналів і зображень для зменшення рівня шуму. Медіанний фільтр є нелінійним КІХ-фільтром.

Значення відліків усередині вікна фільтра сортуються в порядку зростання (спадання), і значення, що знаходиться у середині упорядкованого

списку, надходить на вихід фільтра. У разі парного числа відліків у вікні, вихідне значення фільтра дорівнює середньому значенню двох відліків у середині упорядкованого списку. Вікно переміщується вздовж сигналу, що фільтрується, і обчислення повторюються.

Алгоритм Tukey 53H

Алгоритм Tukey 53H [22] використовує принцип, згідно з яким медіана є робастною оціночною функцією середнього значення, щоб обчислити «гладку» тимчасову послідовність, яка може бути в подальшому вирахувана з початкового сигналу для знаходження викидів. Алгоритм складається з наступних кроків.

1. Складається послідовність $x_i^{(1)}$ як медіана 5 точок з x_{i-2} по x_{i+2}.
2. Складається послідовність $x_i^{(2)}$ як медіана 3 точок з $x_{i-1}^{(1)}$ по $x_{i+1}^{(1)}$.
3. Складається послідовність $x_i^{(3)} = \frac{1}{4} * \left(x_{i-1}^{(2)} + 2 * x_i^{(2)} + x_{i+1}^{(2)}\right)$.
4. Складається послідовність $\Delta_i = \left|x_i - x_i^{(3)}\right|$, тоді точка x_i вважається викидом, якщо $\Delta_i > k * \sigma$, де σ – середньоквадратичне відхилення ряду x_i, а k – наперед заданий поріг, зазвичай $k = 3 \ldots 9$.

Слід зауважити, що згладжування вихідних даних слід застосовувати з обережністю, оскільки можливе згладжування інформативних і важливих значень.

Перетворення вихідних даних

У теорії обробки сигналів важливе місце займає перетворення Фур'є [23], яке дає можливість переходу від часових координат до частотних, що дозволяє витягувати нову інформацію про сигнал і певним чином обробляти його.

Наступним кроком після перетворення Фур'є стало вейвлетне перетворення [24], яке здійснювало перехід від часових координат до координат масштаб-час, що, по суті, давало чітку прив'язку спектру різних особливостей сигналів до часу.

Таким чином, можливі два підходи попередньої обробки даних для прогнозування при використанні перетворень.

1) Вихідні дані обробляються перетворенням, після чого проводиться будь-яка обробка (наприклад, придушення високих частот для позбавлення від шуму при використанні перетворення Фур'є), і виконується зворотне перетворення, після чого вже оброблені дані використовуються для побудови прогнозуючої моделі.

2) Вихідні дані обробляються перетворенням, після чого отримані дані будуть використовуватися для побудови моделі, яка буде прогнозувати процес використовуючи в якості входів перетворені дані (наприклад, деталізуючі коефіцієнти і коефіцієнти апроксимації у випадку вейвлетного перетворення). Таким чином, перед безпосереднім прогнозуванням потрібно буде виконувати відповідне перетворення.

Висновки до розділу

У цьому розділі було розглянуто:

2) основні типи випадкових перешкод, що можуть впливати на вихід об’єкту, поведінка якого прогнозується;
3) основні методи попередньої обробки даних, що дозволяють:
    a) зменшити вплив випадкових перешкод;
    b) трансформувати наявні дані для виділення певної інформації, наприклад розподіл енергії по частотам у випадку перетворення Фур’є.

РОЗДІЛ 3. ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ З ВИКОРИСТАННЯМ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ

3.1. Використання тільки ШНМ

Оскільки у загальному випадку істинна модель прогнозованого процесу невідома, то використання ШНМ, здатних за минулим спостереженням відтворити нелінійне відображення [25-33] виду:

$$x(k) = F\big(x(k-1), x(k-2), \dots, x(k-n_A)\big) + e(k) = \hat{x}(k) + e(k),$$

у якості методу прогнозування є доволі логічним вибором (тут $\hat{x}(k)$ – оцінка-прогноз значення $x(k)$, отримана на виході нейронної мережі, що в даному випадку представляє собою нелінійну авторегресійну NAR [x] модель, $e(k)$ – помилка прогнозу).

Можливість і ефективність використання NAR-моделі в задачах прогнозування визначається теоремою Такенса про дифеоморфізм [34], яка встановлює існування порядку моделі n_A, який забезпечує як завгодно мале значення помилки $e(k)$, і універсальними апроксимуючими властивостями ШНМ.

Згідно з монографією [35], при використанні ШНМ у якості методу прогнозування, найбільш розповсюдженими є 2 підходи:

1. Використання багатошарових мереж прямого розповсюдження [36], вхідний шар яких складається з елементів чистої затримки z^{-1} з відводами.

На рис. 3.1 приведена архітектура багатошарової мережі прямого розповсюдження:

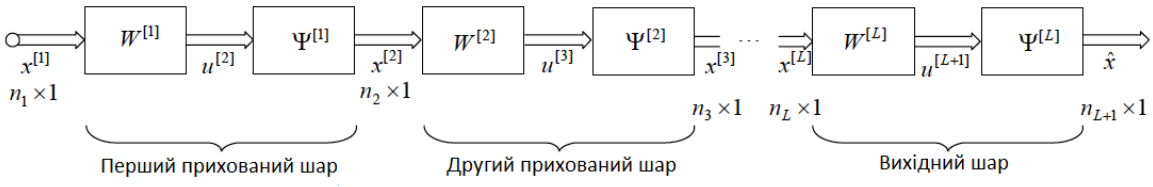

Рис. 3.1. L -шарова нейронна мережа прямого розповсюдження

На перший прихований шар такої мережі поступає $n = n_1 = n_A = n_A^{[1]}$-мірний вектор $x^{[1]}$, сформований у вхідному шарі за допомогою елементів затримки z^{-1} $(z^{-1}x(k) = x(k-1))$, і який має вигляд $x(k-1), x(k-2), \dots, x\left(k - n_A^{[1]}\right)$. Вихідним сигналом першого прихованого шару є $(n_2 \times 1)$−мірний вектор $x^{[2]}$, що подається на вхід другого прихованого шару і т.д. На виході L-го шару (вихідного) з'являється прогнозний $m = n_{L+1}$-мірний вектор $\hat{x}$. Таким чином, кожен шар має n_l входів і n_{l+1} виходів, і

характеризується $(n_{l+1} \times (n_l + 1))$-мірною матрицею синаптичних ваг $W^{[l]}$ і $(n_{l+1} \times n_{l+1})$-мірним діагональним оператором $\Psi^{[l]}$, що сформований з нелінійних активаційних функцій $\psi_j^{[l]}, j = 1,2, \dots, n_{l+1}$.

На рис. 3.2 приведена схема стандартного формального статичного нейрона *l*−го шару, *l*= 1,2,…, L:

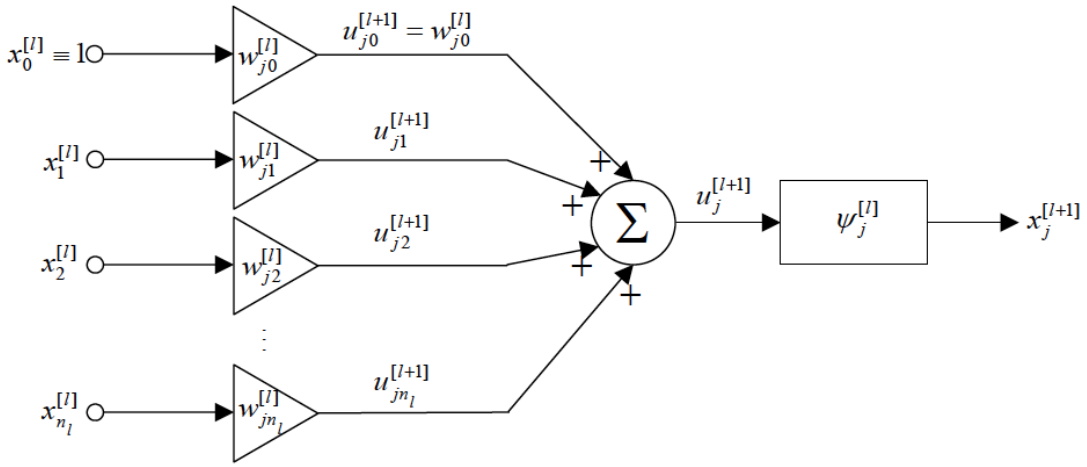

Рис. 3.2. Статичний нейрон

Така ШНМ складається зі стандартних статичних нейронів, що реалізують наступне нелінійне відображення:

$$x_j^{[l+1]} = \psi_j^{[l]}\left(u_j^{[l+1]}\right) = \psi_j^{[l]}\left(\sum_{i=0}^{n_l} u_{ji}^{[l+1]}\right) = \psi_j^{[l]}\left(\sum_{i=0}^{n_l} w_{ji}^{[l]} x_i^{[l]}\right).$$

Кожен нейрон має $n_l + 1$ синаптичних ваг $w_{ji}^{[l]}$, що будуть налаштовуватися у процесі навчання нейронної мережі.

Усього мережа складається з $\sum_{l=1}^{L}(n_l + 1)\, n_{l+1}$ невідомих параметрів, що налаштовуються за допомогою процедури навчання (зазвичай, певної модифікації методу зворотного поширення помилки [37]).

Загальним недоліком прогнозуючих нейронних мереж на статичних нейронах є надто велика кількість параметрів, що налаштовуються, і низька швидкість навчання.

2. У зв’язку з цим, Е. Ваном було запропоновано [38-41] у прогнозуючих нейронних мережах замість статичних нейронів використовувати їх динамічні аналоги, у яких синаптичні ваги є цифровими адаптивними нерекурсивними фільтрами с кінцевою імпульсною характеристикою (КІХ-фільтри, FIR-фільтри) [42] так, як це показано на рис. 3.3.

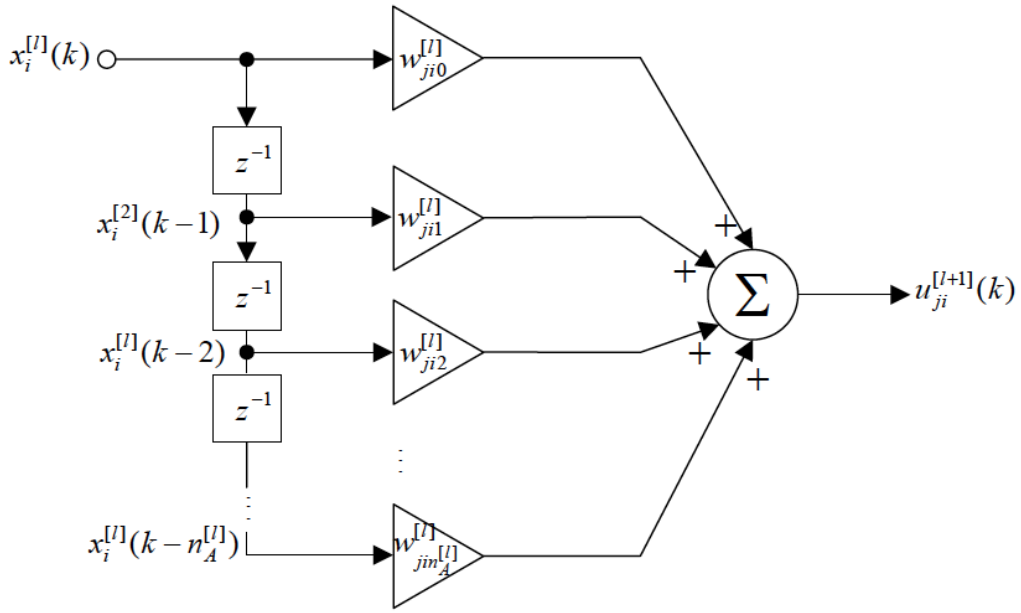

Рис. 3.3. Синапс–КІХ-фільтр

Тоді динамічний нейрон, що використовує ці синаптичні КІХ-ваги буде мати вигляд, як показано на рис. 3.4.

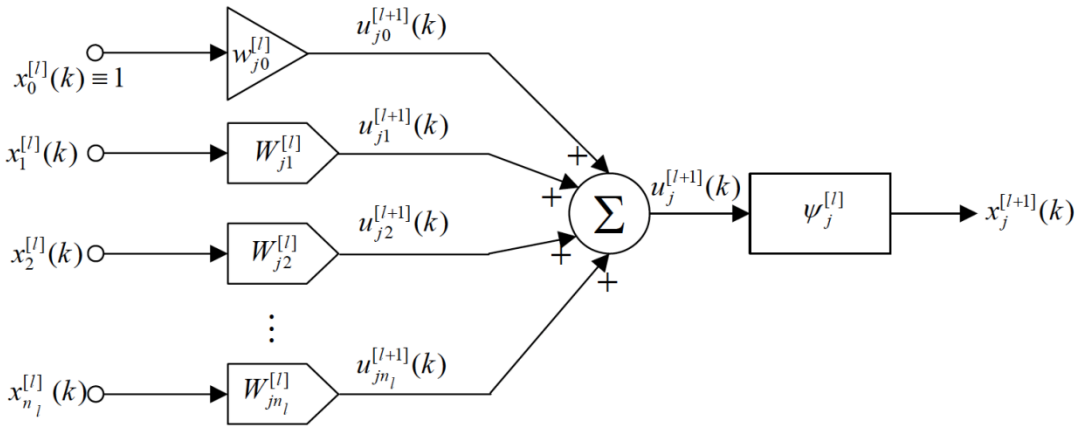

Рис. 3.4. Динамічний КІХ-нейрон

І хоча динамічний нейрон складається з $n_l\left(n_A^{[l]}+1\right)+1$ параметрів, що перевищує кількість синаптичних ваг звичайного нейрону, мережа, побудована з таких вузлів, має набагато менше параметрів, ніж мережа зі стандартною архітектурою на статичних нейронах з лініями затримки на вході. У роботі [39] було доведено, що в мережі на статичних нейронах з лініями затримки кількість параметрів зростає у геометричній залежності від n_A, у той час як в ШНМ на динамічних нейронах кількість налаштовуваних параметрів є лінійною функцією від n_A і L. Для навчання ШНМ на динамічних нейронах в роботі [40] була введена градієнтна процедура, що отримала назву зворотного поширення помилки у часі.

3.2. Об'єднання підходів ШНМ і МГУА

Після аналізу різних існуючих на даний момент методів побудови прогнозуючої моделі, найбільш «гнучкими» методами можна однозначно назвати ШНМ і МГУА. Під гнучкістю розуміється відсутність необхідності визначення експертом чіткої моделі прогнозування, як наприклад, у лінійної регресії або методології Бокса-Дженкінса. Однак, при використанні ШНМ постає питання вибору оптимальної архітектури мережі, а при використанні МГУА – оптимальних опорних функцій.

Розглянемо алгоритм [1], який об'єднує ці два підходи – для цього ШНМ будуть використовуватися в якості опорних функцій МГУА. Завдяки цьому даний алгоритм поєднує в собі переваги як ШНМ, так і МГУА: поступове збільшення складності моделі МГУА і можливість навчання ШНМ, і складається з наступних кроків.

1. З даних виду $\mathrm{X}=\{\mathrm{x_n}, \mathrm{n}=1\ldots\mathrm{N}\}$, де N – кількість відліків, отриманих в результаті етапу попередньої обробки даних, методом ковзаючого вікна формуються дві наступні матриці:

$$X^{(0)} = \begin{bmatrix} x_1 & x_2 & \dots & x_k \\ x_{k+2} & x_{k+3} & \dots & x_{2*k+1} \\ \dots & \dots & \dots & \dots \\ x_{N-k-1} & x_{N-k} & \dots & x_{N-1} \end{bmatrix}, y = \begin{bmatrix} x_{k+1} \\ x_{2*k+2} \\ \dots \\ x_N \end{bmatrix},$$

де k – розмір вікна. За допомогою цих матриць надалі здійснюватиметься навчання мереж – кожен вектор-рядок $\bar{x}_n = \{x_{n1}, x_{n2}, \dots, x_{nk}\}, n = 1 \dots m$ (де m – кількість рядків матриці) матриці $X^{(0)}$ і відповідне йому значення y_n є незалежним прикладом, а вектор-стовпець $\vec{x}_z = \{x_{z1}, x_{z2}, \dots, x_{zm}\}, z = 1 \dots k$ – окремою змінною.

2. Отримані приклади діляться в деякому співвідношенні (зазвичай $0.7m : 0.3m$) на навчальну і перевірочну вибірки.

3. Визначається вид опорних функцій МГУА, а саме від яких змінних вони будуть залежати, наприклад $f_l = f(x_i, x_j)$ або $f_l = f(x_i, x_j, x_i * x_j, x_i^2, x_j^2)$, де $l = 1 \dots C_k^2$ – кількість варіантів вибору двох змінних з k можливих (в загальному випадку $l = 1 \dots C_k^o$, де o – кількість змінних, що відбираються як аргументи опорних функцій). На відміну від МГУА, де також визначається конкретний вид функцій (наприклад $f_l = a_0 + a_1 * x_i + a_2 * x_j$), у даному алгоритмі вибираються лише змінні, від яких залежатимуть опорні функції (ступені змінних і їх коваріації можна представити як додаткові змінні), а знаходження залежності значення функції від змінних буде здійснюватися нейронною мережею.

4. Складаються C_k^o (o – кількість змінних, що враховуються в опорних функціях, для описаного вище прикладу o = 2) нейронних мереж з одним виходом, одним прихованим шаром з малою кількістю нейронів (близько 3) в ньому, і потрібною кількістю входів (для опорних функцій виду $f_l = f(x_i, x_j)$ мережа повинна мати 2 входи, для $f_l = f(x_i, x_j, x_i * x_j, x_i^2, x_j^2)$ – 5 входів і т.д.).

5. Кожна нейронна мережа зіставляється з конкретною опорною функцією, а саме вибираються змінні, які подаватимуться на входи мережі (наприклад, для опорних функцій виду $f_l = f(x_i, x_j, x_i * x_j, x_i^2, x_j^2)$ одна мережа

буде працювати зі змінними $x_1, x_2, x_1 * x_2, x_1^2, x_2^2$, а інша зі змінними $x_1, x_k, x_1 * x_k, x_1^2, x_k^2$), і навчається, використовуючи тільки приклади з навчальної вибірки.

6. На цьому кроці слід скласти вихідні дані для наступної ітерації алгоритму. Для цього слід за деяким зовнішнім критерієм відібрати k кращих (також можна відібрати меншу кількість, але доповнивши їх найбільш «корисними» вихідними змінними) мереж, після чого скласти нову матрицю:

$$X^{(1)} = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1k} \\ h_{21} & h_{22} & \dots & h_{2k} \\ \dots & \dots & \dots & \dots \\ h_{m1} & h_{m2} & \dots & h_{mk} \end{bmatrix},$$

де h_{ij} – значення виходу j-ої з k найкращих мереж при подачі на її входи і-го прикладу, $i = 1 \dots m, j = 1 \dots k$ (або вихідна змінна).

7. Виконується наступна ітерація, але в якості вихідних прикладів береться вже матриця $X^{(1)}$. Ітерації виконуються, поки значення зовнішнього критерію поліпшується, або поки не буде досягнуто необхідне значення критерію.

8. У ході алгоритму на кожній ітерації слід запам'ятовувати структуру і ваги мереж (та / або вихідних змінних), які були відібрані для отримання матриці вихідних прикладів для наступної ітерації. Після досягнення необхідного значення критерію (або досягнення ітерації, після якої значення критерію починає погіршуватися – рости) виконання ітерацій припиняється, і відбирається єдина мережа, яка в подальшому і буде прогнозувати остаточне значення. Для подальшого прогнозування деякого значення з використанням отриманих результатів необхідно:

- скласти вихідний приклад $inp = \{x_{c-k-1}, x_{c-k}, \dots, x_{c-1}\}$;
- подати даний приклад на входи тих мереж, які були використані для отримання матриці вихідних прикладів другої ітерації, і, використовуючи отримані значення виходів (та / або вихідних змінних), скласти новий вхідний приклад для другої ітерації;

- повторювати другий крок поки не буде складено вхідний приклад для ітерації, на якій було здійснено зупинку алгоритму, після чого подати складений приклад на входи відібраної мережі, вихід якої і буде прогнозом значення x_c.

Переваги та недоліки запропонованого алгоритму

Перевагою даного алгоритму в порівнянні з МГУА є те, що не потрібно явно задавати вид опорних функцій, необхідну залежність знаходитимуть нейронні мережі, які, як відомо, дуже добре справляються з цим завданням.

Даний алгоритм також позбавлений відомого недоліку ШНМ: при побудові мережі заздалегідь невідома її оптимальна складність, і занадто проста нейронна мережа може погано змоделювати процес, а надто складні мережі схильні до так званого «оверфіттінгу», або перенавчання, у результаті якого мережу починає моделювати шум, присутній у навчальній вибірці, і як наслідок він показує погані результати на перевірочної вибірці. Пропонований же алгоритм на кожній ітерації використовує прості мережі, не схильні до перенавчання, але за рахунок каскадного ускладнення здатний прогнозувати дуже складні процеси.

Головним недоліком запропонованого методу є його ресурсомісткість, так як на кожній ітерації будується деяка кількість нейронних мереж, що може зайняти чимало часу. Але за рахунок використання чисельно-оптимізованих алгоритмів навчання нейронних мереж і загальної оптимізації алгоритму, можна досягти порівняно швидкої роботи методу.

- **Визначення параметрів запропонованого методу**

Щоб побудувати остаточну прогнозуючу модель, використовуючи запропонований метод, необхідно визначити його параметри. Оскільки даний алгоритм є сумішшю підходу МГУА та штучних нейронних мереж, то завдання визначення його параметрів розпадається на дві підзадачі:

1) задача структурно-параметричного синтезу нейронної мережі;
2) задача визначення потрібних параметрів МГУА.

Для вирішення першої підзадачі необхідно більш детально розглянути апарат ШНМ.

Як вже говорилося вище, нейронні мережі складаються зі штучних нейронів. Штучний нейрон зазвичай представляють як деяку нелінійну функцію від єдиного аргументу – лінійної комбінації всіх вхідних сигналів. Дану функцію називають функцією активації або функцією спрацьовування, передавальною функцією (рис. 3.5).

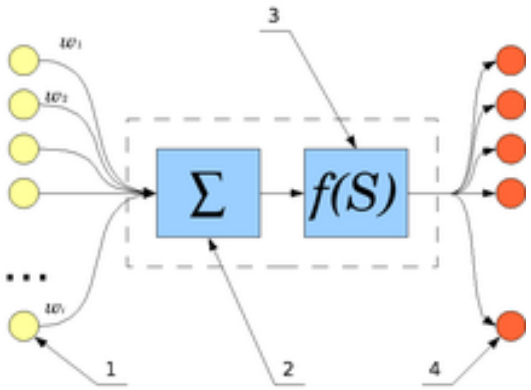

Рис. 3.5. Схема штучного нейрона: 1. Нейрони, вихідні сигнали яких надходять на вхід даному нейрону. 2. Суматор вхідних сигналів. 3.Обчислювач передавальної функції.4.Нейрони, на входи яких подається вихідний сигнал даного нейрона

Математично нейрон являє собою зважений суматор, єдиний вихід якого визначається через його входи і матрицю ваг наступним чином:

$$y = f(u), \text{ де } u = \sum_{i=1}^{n} \omega_i * x_i + \omega_0 * x_0.$$

Тут x_i і ω_i – сигнали на входах нейрона і ваги входів відповідно, функція u називається індукованим локальним полем, а $f(u)$ – активаційною функцією. Додатковий вхід x_0 і відповідна йому вага ω_0 використовуються для ініціалізації нейрона.

Оскільки ШНМ – це мережа взаємопов'язаних нейронів, то одним з основних параметрів, що характеризують нейронну мережу, є те, як саме здійснюється зв'язок між нейронами, тобто топологія мережі.

Далі розглянемо основні топології ШНМ.

Багатошаровий персептрон (БП)

Ця топологія являє собою мережу, що складається з декількох послідовно з'єднаних шарів нейронів, причому кожен нейрон шару i з'єднаний з усіма нейронами шару і $-$ 1, де i – номер шару. Найперший шар є входами мережі, у процесі навчання або роботи мережі на нього подаються вихідні дані. За ним слідує один або кілька прихованих шарів, і останній шар – вихідний шар, який являє собою виходи мережі (рис. 3.6).

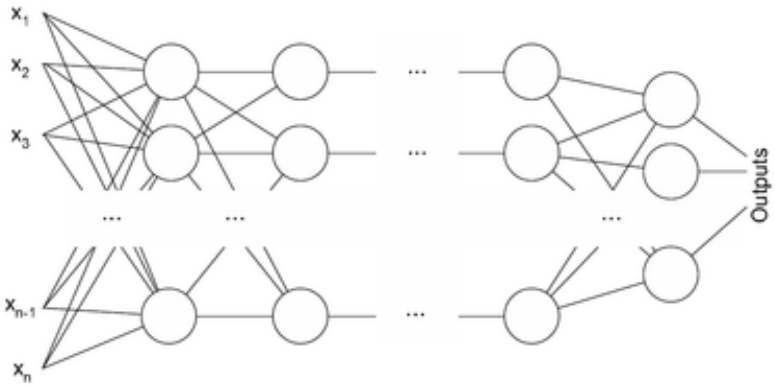

Рис. 3.6. Архітектура багатошарового персептрона

Переваги: найбільша вивченість, простота, стійкість.

Недоліки: розглядає навчальні приклади незалежно один від одного, тим самим упускаючи можливий зв'язок між прикладами.

Рекурентні нейронні мережі

У рекурентних мережах зв'язки між нейронами можуть бути не тільки послідовними, а й зворотними, у тому числі зв'язок може бути між нейроном і самим собою.

Найбільш популярним окремим випадком рекурентних мереж є мережа Елмана [43]. Рекурентна мережа Елмана характеризується частковою рекурентністю у вигляді зворотного зв'язку між прихованим і вхідним шаром, реалізованою за допомогою одиничних елементів запізнювання (рис. 3.7).

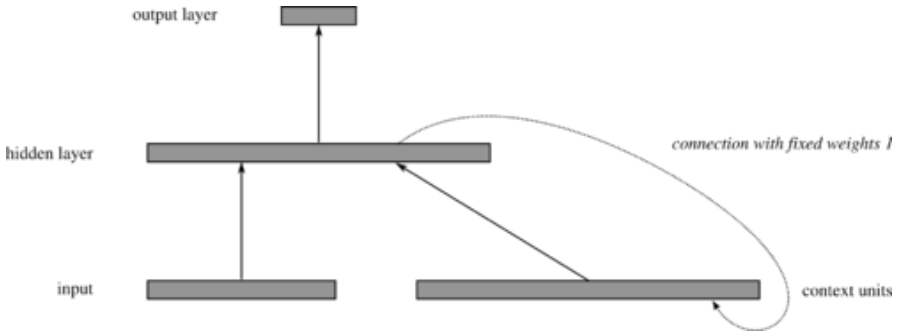

Рис. 3.7. Архітектура нейронної мережі Елмана

Переваги: рекурентні ШНМ враховують можливий зв'язок між прикладами, як наслідок краще підходять для моделювання послідовностей; їх здатність до узагальнення більша, ніж у БП.

Недоліки: нестійкість, складність, навчання в загальному випадку більш повільне, ніж у БП.

Радіально-базисні нейронні мережі

Так називаються ШНМ, що використовують в якості активаційних радіально-базисні функції (такі мережі скорочено називаються RBF-мережами). Загальний вигляд радіально-базисної функції: $f(x)=\phi\left(\frac{x^2}{\sigma^2}\right)$, наприклад, $f(x)=e^{-\frac{x^2}{\sigma^2}}$, де x – вектор вхідних сигналів нейрона, σ – ширина вікна функції, $\phi(y)$ – спадна функція (найчастіше, рівна нулю поза деяким відрізком).

Радіально-базисна мережа характеризується трьома особливостями:

1) єдиний прихований шар;

2) тільки нейрони прихованого шару мають нелінійну активаційну функцію;

3) ваги зв'язків між вхідним і прихованим шаром рівні одиниці.

Переваги: велика швидкість навчання, містять всього один прихований шар, вихідний шар має лінійну активаційну функцію.

Недоліки: володіють поганими екстраполюючими властивостями, усі зв'язки між входами і нейронами прихованого шару рівні 1, тобто всі вхідні змінні мають рівну значущість при знаходженні виходів мережі.

Постановка задачі структурно-параметричного синтезу нейронних мереж для вирішення задачі прогнозування

Побудова ШНМ полягає в послідовному вирішенні наступних завдань:

- вибір топології;
- структурний синтез ШНМ: визначення кількості і типів (числові, бінарні, нечіткі) входів мережі, кількості прихованих шарів і кількості нейронів в них, кількості виходів мережі;
- вибір алгоритму навчання;
- параметричний синтез ШНМ: вибір типу активаційних функцій для шарів (сигмоїда, лінійна, порогова), визначення значень вагових коефіцієнтів.

Вирішення задачі структурно-параметричного синтезу нейронних мереж для вирішення задачі прогнозування

- У роботах [44] – [46] обґрунтовується вибір БП в якості топології нейронної мережі при вирішенні задачі прогнозування, а в роботах [47], [43] – вибір рекурентних нейронних мереж. У даній роботі в якості топології нейронної мережі була обрана топологія багатошарового персептрона через більшу її вивченість, а також стійкість і простоту.
- Теоретично було доведено [48], що багатошаровий персептрон з одним прихованим шаром і достатньою кількістю нейронів у ньому може апроксимувати як завгодно складну безперервну функцію. Таким чином, для завдання прогнозування не має сенсу вибирати більш складну структуру.

Кількість і тип входів може змінюватися в залежності від розміру навчальної вибірки і типу змінних в ній. Так як у запропонованому методі кількість входів мережі зазвичай (якщо не вводити додаткові змінні)

дорівнює розміру передісторії, тобто кількості попередніх значень, що враховуються при прогнозі майбутнього значення, то починати слід з малої кількості входів ($\approx 0.05 * m \dots 0.15 * m$, де – кількість прикладів в навчальній вибірці). Це дозволить побудувати більш просту прогнозуючу модель, а більшість практичних випробувань показують, що простіша модель зазвичай менш схильна до великих помилок прогнозу (хоча і здається, що більша кількість інформації має дати більш точну модель).

Кількість виходів в загальному випадку дорівнює одиниці.

Кількість нейронів в прихованому шарі може варіюватися, але знову ж, бажано не вибирати дуже велику кількість, тому що це, по-перше, ускладнить структуру мережі, а по-друге, швидше за все, призведе до перенавчання. Так як у запропонованому алгоритмі ускладнення моделі відбувається з ряду в ряд аналогічно підходу МГУА, то кількість нейронів в прихованому шарі зазвичай вибирається малим ($\approx 3 \dots 10$).

- Найбільш відомий і поширений алгоритм навчання ШНМ – це алгоритм зворотного поширення помилки, який ґрунтується на методі градієнтного спуску. Однак, на сьогоднішній день цей алгоритм в чистому вигляді використовувати небажано, тому що розроблено безліч куди більш ефективних методів. У даній роботі в якості алгоритму навчання буде використовуватися алгоритм Левенберга-Марквардта [49].
- Найбільш поширеною активаційною функцією є сигмоїда:

$$f(s) = \frac{1}{1+e^{-a*s}},$$

Також використовуються такі активаційні функції:

- гіперболічний тангенс: $f(s) = \frac{e^{a*s}-e^{-a*s}}{e^{a*s}+e^{-a*s}}$.
- порогова: $f(s) = \begin{cases} 0, s < 0 \\ 1, s \geq 0 \end{cases}$.
- лінійна: $f(s) = a * s$.

Для задачі прогнозування прийнято для нейронів в прихованому шарі вибирати сигмоїду в якості активаційної функції, а для нейрона вихідного

шару – лінійну активаційну функцію, оскільки значення прогнозованого процесу в загальному випадку можуть змінюватися від $(-\infty; +\infty)$, у той час як область значень сигмоїд лежить в межах $(0; 1)$.

Значення ж вагових коефіцієнтів зв'язків між нейронами визначаються в результаті навчання мережі.

Постановка задачі визначення параметрів МГУА для вирішення задачі прогнозування запропонованим методом

У рамках даного методу потрібно визначити такі параметри МГУА:

- конкретний алгоритм МГУА (як було сказано, МГУА – це сімейство алгоритмів);
- зовнішній критерій, за яким відбиратимуться «кращі» моделі для переходу на наступну ітерацію, і за яким буде визначатися момент зупинки алгоритму;
- вид опорних функцій.

Вирішення задачі визначення параметрів МГУА для вирішення задачі прогнозування запропонованим методом

- Два основних алгоритмиу МГУА – це комбінаторний алгоритм і багаторядний алгоритм.

1) Комбінаторний алгоритм.

У комбінаторних алгоритмах виконується перебір всіляких моделей із заданого базису з вибором кращої з цих моделей за заданим критерієм селекції.

При переборі складність часткових моделей, тобто число аргументів поступово нарощується від 1 до максимального числа n (числа аргументів базисного набору функцій).

Таким чином, загальна схема комбінаторного алгоритму включає наступні операції.

- Визначаються коефіцієнти всіх приватних моделей при складності s=1..n.
- Для кожної з них обчислюється значення зовнішнього індивідуального або комбінованого критерію складності селекції.
- Єдина модель оптимальної складності вибирається за мінімальним значенням критерію.

2) Багаторядний алгоритм.

Багаторядні алгоритми працюють за наступною схемою.

1-ий ряд – на основі даних таблиці спостережень будуються часткові описи об'єкта (найчастіше попарні), що наближають вихідну змінну q:

$$y_1=f_1(x_1,x_2), y_2=f_2(x_1,x_3),\ldots,y_k=f_k(x_{n-1},x_n).$$

З цих моделей вибирається деяке число кращих за зовнішнім критерієм.

2-ий ряд – отримані змінні приймаються як аргументи другого ряду, і знову будуються всі часткові описи:

$$z_1=\varphi_1(y_1,y_2), z_2=\varphi_2(y_1,y_3),\ldots,z_l=\varphi_l(y_{F_l-1},y_F).$$

З них за зовнішнім критерієм відбирається F_2 кращих моделей в якості змінних наступного ряду і т.д. Ряди нарощуються до тих пір, поки знижується значення зовнішнього критерію.

У даній роботі пропонується використовувати багаторядний підхід МГУА, оскільки йому віддається перевага при невеликій кількості прикладів в навчальній вибірці, і до того ж, він дозволяє використовувати ШНМ в якості часткових описів.

- Перелічимо найбільш поширені зовнішні критерії.

1) Критерій регулярності.

Критерій регулярності $\Delta^2(\mathrm{C})$ включає середньоквадратичну помилку на тестовій вибірці C, отриману при параметрах моделі, налаштованих на навчальній вибірці l:

$$\Delta^2(\mathrm{C}) = |y_C - f(A_C, \widehat{\omega_l})|^2,$$

де y_C – вектор значень залежної змінної, взятий з тестової вибірки; A_C – матриця значень незалежних змінних, взята з тестової вибірки; $\widehat{\omega_l}$ – матриця параметрів моделі, отримана з навчальної вибірки; f – певна функція.

2) Критерій мінімального зміщення.

Інакше критерій несуперечності моделі: модель, яка має на навчальній вибірці одну неув'язку, а на контрольній – іншу, називається суперечливою. Цей критерій включає різницю між залежними змінними моделі, обчисленими на двох різних вибірках l і C. Критерій не включає помилку моделі в явній формі. Він вимагає, щоб оцінки коефіцієнтів в оптимальній моделі, обчислені на множинах l і C, розрізнялися мінімально.

Критерій несуперечності як критерій мінімуму зміщення має вигляд:

$$\eta_\alpha^2 = |\widehat{\omega_l} - \widehat{\omega_C}|^2,$$

де $\widehat{\omega_C}$ – матриця параметрів моделі, отриманих на тестовій вибірці.

3) Комбінований критерій.

Цей критерій дозволяє використовувати при виборі моделей лінійну комбінацію декількох критеріїв. Комбінований критерій $k^2 = \sum_{i=1}^{K} \alpha_i * k_i$, за умови нормування $\sum_{i=1}^{K} \alpha_i = 1$, де k_i – прийняті на розгляд критерії, а α_i – ваги цих критеріїв, призначені на початку обчислювального експерименту.

У даній роботі пропонується використовувати критерій регулярності в якості зовнішнього критерію, оскільки він є найбільш простим і швидко обчислюваним, а як було вже сказано вище, одним з недоліків запропонованого алгоритму є його складність і ресурсомісткість.

- Як було зазначено раніше, у цьому методі на відміну від класичного методу групового обліку аргументів не потрібно явно задавати вид опорних функцій (як-то лінійна функція, функція з коваріацією змінних, функція зі ступенями змінних і т.д.), але потрібно задати від скількох аргументів будуть залежати опорні функції, а необхідну залежність знаходитимуть ШНМ.

Порівняння якості прогнозу методу і ШНМ

Для перевірки алгоритму використовуємо публічно доступні дані про продажі літаків (загальна кількість проданих в США літаків за роками) з 1947 по 2011 рік [50]. Дані попередньо опрацюємо за допомогою алгоритму Tukey 53H.

При побудові моделі за допомогою алгоритму використовувалися наступні параметри:

- розмірність, що використовується при вкладенні часового ряду для отримання відповідних матриць прикладів: $k = 5$;
- співвідношення розмірів навчальної і перевірочної вибірок: $0.7m / 0.3m$, де m – загальна кількість прикладів. Для отримання навчальної вибірки використовувалися перші $0.7m$ прикладів;
- вид опорних функцій: $f_l = f\left(x_i, x_j, x_i * x_j, x_i^2, x_j^2\right)$, $l = 1 \ldots C_5^2$, $i = 1 \ldots 5, j = 1 \ldots 5, i \neq j$;
- структура нейронних мереж: один нейрон у вихідному шарі, один прихований шар з 3 нейронами, 5 нейронів у вхідному шарі (відповідно до розмірності вкладення);
- при відборі мереж СКП розраховувалася на всіх наявних прикладах;
- для створення матриці прикладів для наступної ітерації використовувалися виходи 3 мереж з мінімальною СКП і дві вхідні змінні, які були входами мережі з найменшою СКП.

Для порівняння якості прогнозу алгоритму використовуємо прогноз, отриманий за допомогою БП з одним вихідним нейроном, одним прихованим шаром з 10 нейронами і 10 вхідними нейронами. БП, так як і алгоритм, навчався використовуючи приклади тільки з навчальної вибірки. Для навчання використовувався алгоритм Левенберга-Марквардта. Відомо, що під час навчання, ШНМ схильні «застрявати» в локальних мінімумах функції помилки, тому було навчено близько 10 ШНМ і обрано мережу з мінімальною СКП.

Порівняння прогнозів, отриманих за допомогою ШНМ і алгоритму, показано на рис. 3.8.

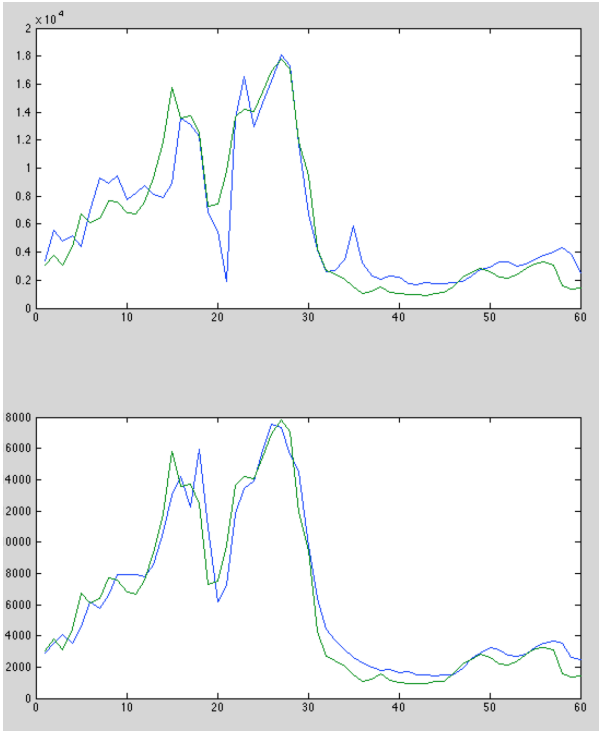

Рис. 3.8. Порівняння прогнозів, отриманих за допомогою ШНМ (зверху) і алгоритму (знизу). Зелений графік – вихідні дані, сині графіки – прогнози

Значення СКП: ШНМ – 0.0613, алгоритму – 0.0255.

На графіках можна помітити, що навчений персептрон прогнозує значення перевірочної вибірки гірше (права частина графіків), у той час як помилки прогнозу, отриманого за допомогою алгоритму, практично однакові для обох – навчальної та перевірочної – вибірок, що зазвичай свідчить про більш точну модель прогнозованого процесу. Алгоритм також більш точно спрогнозував «екстремальні» значення, що є дуже важливим фактором при прогнозуванні попиту.

3.3. Комплексування декількох методів прогнозування

Для поліпшення якості прогнозу можливо використовувати комплексування оцінок, отриманих за допомогою різних моделей. Під комплексуванням розуміється зважена сума оцінок, отриманих за допомогою

згенерованого набору моделей. Вагові коефіцієнти визначаються за допомогою зовнішнього критерію оптимальності моделей – дисперсії на екзаменаційній вибірці. Множина моделей отримується перебором варіантів розбиття вихідної вибірки на підвибірки і перебором різних методів прогнозування. Таким чином, маючи $k1$ варіантів розбиття на підвибірки і $k2$ методів прогнозування, отримуємо $k1 * k2$ різних моделей. Для отримання остаточного прогнозу, маючи вектор вхідних даних, необхідно:

1. Подати цей вектор на вхід кожної моделі, таким чином отримавши вектор оцінок $\hat{\boldsymbol{y}} = [\hat{y}_1, \dots, \hat{y}_{k1*k2}]$;
2. Отримати остаточний прогноз $\hat{y}_f$ як зважену суму елементів вектора оцінок $\hat{\boldsymbol{y}}$.

Ці кроки вимагають визначення вагових коефіцієнтів $\alpha_i, i = 1 \dots k1 * k2$.

3.3.1. Розбиття вихідної вибірки на підвибірки

- По порядку.

У навчальну вибірку відбираються перші $C_1 * N$ точок, у перевірочну – наступні $C_2 * N$ точок, і в екзаменаційну – точки, що залишилися, тобто $(1 - C_1 - C_2) * N$ (де N – загальна кількість точок, $C_1 + C_2 < 1; C_1, C_2 > 0$ – коефіцієнти, зазвичай вибирають $C_1 = 0.6, C_2 = 0.2$).

- Випадковим чином.

Аналогічно попередньому методу, тільки точки відбираються не по порядку, а випадковим чином, але пропорції між підвибірками зберігаються.

- Кожен і-й.

У перевірочну вибірку відбирається кожна і-а точка, з решти точок в екзаменаційну відбирається кожна j-а, всі точки, що залишилися, відбираються в навчальну вибірку (зазвичай вибирають $i = 3, j = 4$).

- За дисперсією.

Усі точки ранжуються за дисперсією (під точкою розуміється один приклад) і потім відбираються в вибірки аналогічно першому методу.

3.3.2. Отримання моделей за допомогою перебору методів

Для кожного з відібраних методів розбиття на вибірки і методів прогнозування будується модель прогнозованого процесу. У роботі [5] використовуються такі методи прогнозування: ШНМ, МГУА, комбінація МГУА та ШНМ. Таким чином, при 4 способах розбиття на вибірки і 3 методах прогнозування отримуємо 4 * 3 = 12 різних моделей.

3.3.3. Комплексування отриманих результатів

Для отримання реального прогнозу використовується комплексування прогнозів усіх отриманих на попередньому етапі моделей. Тобто, маючи оцінки прогнозованого значення $\hat{\boldsymbol{y}} = [\hat{y}_1, \dots, \hat{y}_{k1*k2}]$, прогноз самого значення буде дорівнювати $\hat{y}_f = \sum_i \alpha_i * \hat{y}_i$. Коефіцієнти α_i визначаються використовуючи зовнішній критерій – дисперсію прогнозу і-ої моделі на екзаменаційній вибірці: $\alpha_i = \frac{1}{\sigma_i}$, де σ_i – СКП прогнозів і-ої моделі на екзаменаційній вибірці від реальних значень прогнозованого процесу. Після обчислення всіх α_i їх слід нормалізувати за формулою $\alpha_i^n = \frac{\alpha_i}{\sum_i \alpha_i}$.

Щоб оцінити застосовність комплексування для поліпшення якості прогнозу, порівняємо нормовану середньоквадратичну помилку (НСКП) прогнозу (на деяких штучних даних), отриманого за допомогою запропонованого в даній роботі алгоритму прогнозування і за допомогою комплексування прогнозів декількох методів (у тому числі і запропонованого). Згенеруємо штучний часовий ряд з наступною істинною моделлю:

$$f(t) = 0.1t^3 - 10t^2 + 6 + 300 * \sin(t), t = 1 \dots 100.$$

Додамо до математичної моделі нормальний шум $\varepsilon(t)$ з характеристиками $E(\varepsilon) = 1000, \sigma(\varepsilon) = 500$.

Графік штучного часового ряду з істинною моделлю та доданим нормальним шумом наведений на рис. 3.9.

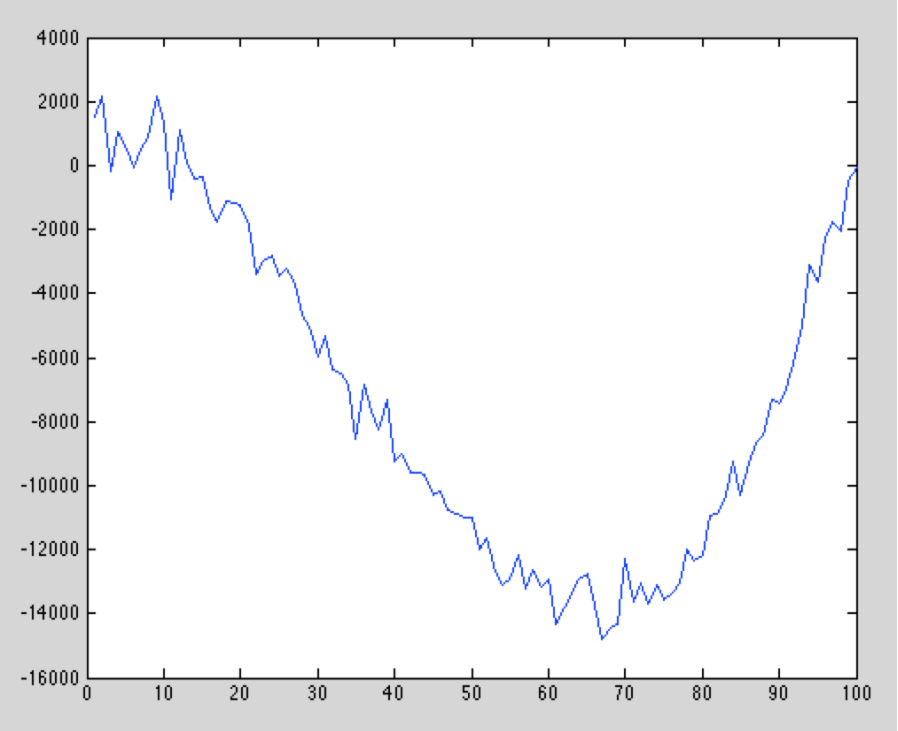

Рис. 3.9. Штучний часовий ряд

Після цього побудуємо прогнозуючу модель із застосуванням алгоритму МГУА + ШНМ та із застосуванням комплексування оцінок декількох методів. НСКП прогнозу, отриманого за допомогою алгоритму МГУА + ШНМ, дорівнює 0.0038 (рис.3.10).

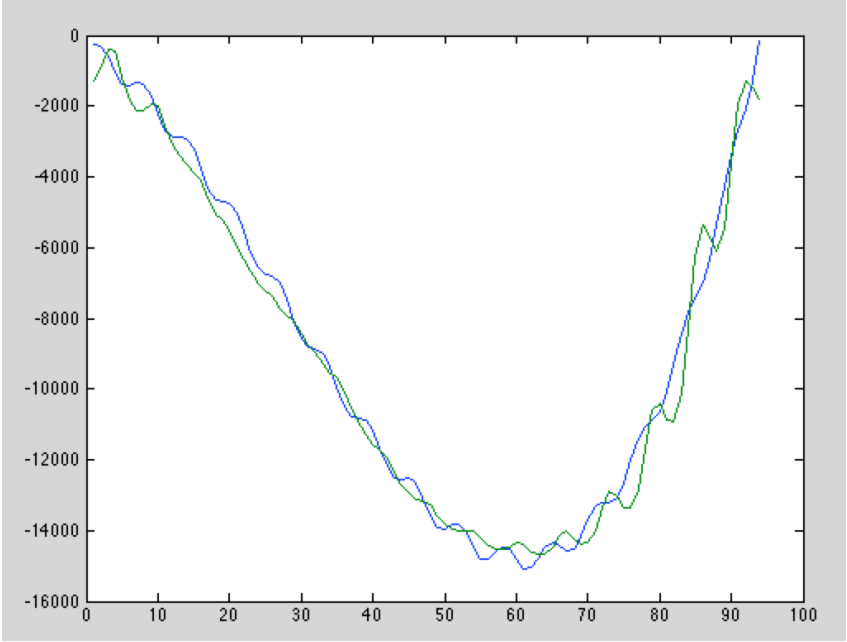

Рис. 3.10. Прогноз, отриманий за допомогою алгоритму МГУА + ШНМ

НСКП прогнозу, отриманого за допомогою комплексування декількох алгоритмів (МГУА + ШНМ, МГУА, ШНМ), дорівнює 0.0020 (рис. 3.11).

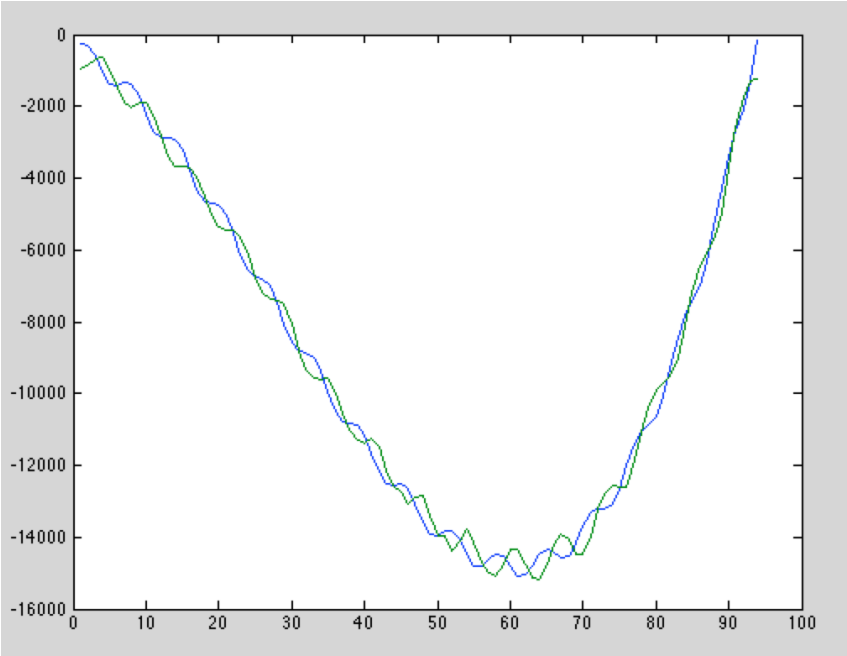

Рис. 3.11. Прогноз, отриманий за допомогою комплексування декількох алгоритмів

Як добре видно з рисунків, комплексування дійсно значно покращує якість отриманого прогнозу.

Висновки до розділу

У цьому розділі було розглянуто певні існуючі методи прогнозування, що базуються на використанні ШНМ та запропоновано два нових:

1) метод, що об’єднує підходи ШНМ та МГУА для знаходження оптимальної структури нейронної мережі для конкретної задачі прогнозування;

2) комплексування декількох прогнозуючих моделей, отриманих шляхом розбиття наявних даних на певну кількість вибірок і використання декількох різноманітних методів прогнозування для навчання моделей (кожна модель навчається на одній з отриманих вибірок).

Нові методи були перевірені на реальних даних, та дали середньоквадратичну помилку, меншу, ніж у таких існуючих методів як ШНМ та МГУА.

РОЗДІЛ 4. ПРОГНОЗУВАННЯ У ВИПАДКУ НЕОДНОРІДНИХ ДАНИХ

4.1. Постановка задачі

При постановці завдання прогнозування найчастіше робиться припущення, що поведінка прогнозованого процесу постійна, і, таким чином, завдання можна вирішити шляхом знаходження потрібної моделі та оцінки її параметрів, використовуючи всі наявні дані. Проте, найчастіше поведінка прогнозованого процесу мінлива, що робить таку постановку задачі неправильною. Розглянемо постановку задачі прогнозування, яка явно враховує проблему непостійної поведінки прогнозованого процесу.

Нехай є:

а) множина прикладів $S = \{(\vec{x}_i, y_i) : \mathrm{i} = 1 \ldots \mathrm{n}\}$, де кожен приклад $(\vec{x}_i, y_i)$ складається з вектора значень вхідних факторів і відповідного вихідного значення;

б) модель f, яка залежить від деякого вектора параметрів $\vec{\theta} = [\theta_1, \ldots, \theta_m]^T$ і найбільш загальним способом описує прогнозований процес – мається на увазі, що залежно від значень параметрів модель здатна описати всі можливі стани об'єкта, який прогнозується.

Необхідно знайти:

а) множину векторів параметрів $\Theta, |\Theta| = k$, таких, що вектор $\vec{\theta}_j$ оптимальний згідно критерію L_j на деякій підмножині вихідних даних S_j : $\Theta = \left\{ \vec{\theta}_j : \vec{\theta}_j = \underset{\theta}{\arg\min}(L_j[f, \vec{\theta}, S_j]), S_j \subseteq S, j = 1 \ldots k \right\}$. Кожен вектор параметрів таким чином буде описувати один стан прогнозованого процесу. Дана постановка задачі робить припущення, що наявні дані описують кілька різних станів процесу, і кожен такий стан може бути описаний певним набором ваг. Однак кількість таких станів невідома – це завдання відповідного методу прогнозування;

б) спосіб визначення кінцевого вектора θ_s параметрів для нових вхідних даних $\vec{x}_s$ виходячи із знайденої множини векторів параметрів і самих вихідних даних: $\vec{\theta}_s = F(\vec{x}_s, \Theta, S)$; такі, що мінімізують деяку оцінку помилки прогнозу $\hat{e} = e(F, \Theta, S)$.

4.2. Класифікація методів

Усі методи прогнозування можна поділити на 3 групи.

1) Глобальні методи. До цієї групи входить найбільша кількість методів: усі методи даної групи знаходять єдиний набір параметрів деякої прогнозуючої моделі, використовуючи всі наявні дані. З точки зору формалізованої постановки задачі методи в цій групі мають наступні параметри:

а. $k = 1$ – використовується лише один вектор параметрів $\vec{\theta}_1$, при цьому $S_1 = S$ – значення цих параметрів знаходяться шляхом оптимізації деякого критерію L_1 на всіх вихідних даних;

б. у якості кінцевого вектора параметрів θ_s для нових вхідних даних $\vec{x}_s$ просто використовується вже знайдений вектор $\vec{\theta}_1$: $\vec{\theta}_s = \vec{\theta}_1$. При такому записі стає очевидно, що значення нових даних ніяк не використовуються при визначенні параметрів моделі.

Найбільш відомі методи в даній групі: лінійна регресія, ARIMA, нейронні мережі, МГУА.

2) Локальні методи є протилежністю глобальним: замість однієї глобальної будується кілька локальних моделей, параметри для яких підбираються на деяких підмножинах вихідних даних. Дані методи враховують проблему непостійної поведінки прогнозованого процесу, але й мають свої недоліки:

- вони більш вимогливі до кількості навчальних даних – чим більше локальних моделей, тим більше потрібно даних;

- втрачається глобальна інформація – кожна локальна модель навчається тільки на своїх даних, що може призвести до зайвої «спеціалізації» та перенавчання моделей.

Найбільш відомі методи в даній групі: локальна регресія, кластеризація з регресією, локальна апроксимація та ін.

3) Глобально-локальні методи. Методи з цієї групи намагаються об'єднати найкраще з обох підходів: всі вихідні вхідні дані використовуються для отримання глобальної інформації про прогнозований об'єкт (часто званої трендом), у той час як деякі підмножини даних використовуються для знаходження локальних особливостей. У роботі [51] в якості прикладу глобально-локального методу названий метод MARS. Авторами також був розроблений підхід «перетворення» глобальних параметричних методів у глобально-локальні методи [3]. Суть підходу полягає в навчанні спочатку однієї глобальної моделі, після чого навчається безліч локальних моделей з додатковим критерієм близькості параметрів локальних моделей до параметрів глобальної моделі; на стадії прогнозу просто вибираються параметри кількох «найближчих» локальних моделей, потім вони усереднюються і застосовуються до нових входів (аналогічно деяким локальним методам). Завдяки такому підходу, а саме використанню критерію близькості параметрів локальної та глобальної моделей, при побудові локальних моделей не втрачається глобальна інформація і самим локальним моделям не потрібна велика кількість даних. Даний підхід був застосований до лінійної регресії, а отриманий метод названий «лінійна регресія з динамічними вагами» – ще один приклад глобально-локального методу.

Найбільш перспективними є глобально-локальні методи, так як вони позбавлені недоліків інших 2 груп:

- на відміну від глобальних методів, параметри прогнозуючої моделі залежать від значень вхідних змінних – тобто використовується локальна інформація про прогнозований процес;

- на відміну від локальних методів, не губиться глобальна інформація, що дозволяє уникнути перенавчання локальних моделей і знизити необхідну кількість навчальних даних.

Саме тому розроблений авторами підхід дозволяє переходити від глобальних до глобально-локальних методів, дає можливість отримувати якіснішу прогнозуючу модель (з точки зору мінімуму оцінки помилки прогнозу моделі), застосовуючи його до існуючих глобальних методів. Для демонстрації розглянемо глобально-локальний метод, отриманий шляхом застосування описаного підходу до глобального методу прогнозування з використанням нейронних мереж прямого поширення – найбільш відомою і застосовуваною архітектурою нейронних мереж для вирішення задачі прогнозування [44-46],[7].

4.3. Кластеризація даних

Дуже важливим кроком у локальних і глобально-локальних методах є розбиття вихідної вибірки на підвибірки, які будуть використані для побудови локальних моделей. Очевидно, що в найкращому випадку кожна підвибірка буде містити набір прикладів, схожих між собою і які відрізняються від прикладів з інших груп. Така неформальна постановка задачі повністю відповідає завданню кластеризації. Для формалізації завдання досить ввести лише функцію відстані між двома прикладами $D = f: X \times X \to \mathbb{R}^+$ і задати деякий критерій оптимальності розбиття на кластери I, який би враховував відстань між прикладами одного кластера і між прикладами з різних кластерів.

Одним з перших і найпростіших алгоритмів кластеризації є алгоритм k-середніх (k-means [52]). Не дивлячись на його простоту, алгоритм дає дуже хороші результати для реальних практичних проблем.

Алгоритм є версією EM-алгоритму (expectation maximization algorithm), застосовуваного також для поділу суміші Гауссіан. Він розбиває безліч елементів векторного простору на заздалегідь відоме число кластерів *k*.

Основна ідея полягає в тому, що на кожній ітерації перерозраховується центр мас для кожного кластера, отриманого на попередньому кроці, потім вектори розбиваються на кластери знову відповідно до того, який з нових центрів виявився ближче за обраною метрикою.

Алгоритм завершується, коли на якийсь ітерації не відбувається зміни центру мас кластерів. Це відбувається за кінцеве число ітерацій, так як кількість можливих розбиттів кінцевої множини звичайна, а на кожному кроці сумарне квадратичне відхилення V не збільшується, тому зациклення неможливе.

Інші відомі алгоритми кластеризації: нейронні мережі Кохонена, ієрархічна кластеризація, генетичні алгоритми та ін. У даному алгоритмі для кластеризації прикладів використовувався саме алгоритм k-середніх через його простоту та ефективність.

4.4. Методика перетворення методів для врахування неоднорідності даних

Опишемо етапи отриманого глобально-локального методу прогнозування:

1. Навчається нейронна мережа прямого поширення з використанням усієї наявної множини прикладів $S = \{(\vec{x}_i, y_i) : \mathrm{i} = 1 \ldots \mathrm{n}\}$. Як функція помилки мережі використовується стандартна середньоквадратична помилка:

$$E = \sum_{i=1}^{n} \left[net(\vec{x}_i) - y_i \right]^2$$

Ваги $\vec{\theta}^g$ даної мережі являють собою інформацію про глобальний характер прогнозованого процесу. Питання вибору оптимальної структури нейронної мережі та оптимального алгоритму навчання в роботі не розглядаються – було використано звичайний багатошаровий персептрон з одним прихованим шаром з 10 нейронів, з гіперболічним тангенсом в якості активаційної функції прихованого шару і лінійної функції в якості

активаційної функції вихідного шару; для навчання використовувався алгоритм Rprop.

2. Виконується кластеризація множини прикладів S. У ідеалі, кожен кластер C_j повинен містити приклади, які найкращим чином описуються деяким єдиним набором ваги мережі $\vec{\theta}_j, j=1...k$, де k – кількість кластерів, тобто, виконуючи кластеризацію, ми сподіваємося, що схожі приклади повинні описуватися схожими вагами моделі. Звичайно, якість прогнозування в такому випадку буде сильно залежати від використаного методу кластеризації. Для ілюстрації підходу використаємо один з найбільш простих методів кластеризації – k-means, з фіксованою кількістю кластерів – 10. Як буде видно з аналізу результатів, навіть при такому простому методі кластеризації вдалося значно поліпшити якість прогнозу.

3. Навчаються k нейронних мереж – по одній на кожен кластер. Навчання виконується таким чином:

а. Ваги кожної мережі ініціалізуються попередньо знайденими вагами $\vec{\theta}^g$.

б. Мережа net_j навчається тільки на прикладах з кластера C_j з використанням наступної функції помилки:

$$E_j = \gamma * \sum_{\vec{x}_i \in C_j} (net_j(\vec{x}) - y_i)^2 + \sum_{\theta_l \in \vec{\theta}_j} (\theta_l - \theta_l^g)^2$$

Таким чином, при навчанні, ваги мережі будуть прагнути як зменшити помилку мережі на прикладах зі свого кластера, так і не віддалятися занадто сильно від початкових ваг $\vec{\theta}^g$. Тобто не буде втрачена інформація про глобальний характер процесу.

4. Після навчання локальних мереж, прогнозування для нових прикладів виконується таким чином: для вхідного вектора $\vec{x}_s$ знаходиться відповідний йому кластер C_h (шляхом знаходження найближчого центру кластера), і для прогнозу використовується мережа, навчена на прикладах цього кластера.

Для оцінки запропонованого методу було виконано його порівняння з наступними методами:

1. «Метод-предок» – нейронна мережа прямого поширення.
2. Повністю локальний варіант методу прогнозування з використанням нейронних мереж – розбиття всіх даних на кластери з подальшою побудовою і навчанням нейронної мережі для кожного кластера.
3. Покращений варіант чисто локального методу прогнозування з використанням нейронних мереж. Спочатку навчається нейронна мережа на всіх даних, і запам'ятовуються її ваги; після цього виконується розбиття даних на кластери та навчання нейронної мережі для кожного кластера, при цьому ваги кожної нейронної мережі будуть ініціалізовані запам’ятованою вагами мережею, навченою на всіх даних.

Порівняння виконувалося шляхом перевірки якості прогнозу вищеназваних методів на 11 різних наборах даних (часових рядах), які знаходяться у відкритому доступі в мережі Інтернет (більшість наборів було взято з сайту [53]). Для перевірки якості прогнозу кожного окремого методу використовувалися наступні параметри:

- кількість попередніх значень часового ряду використовуються в якості вхідних змінних (розмірність вкладення): 5;
- співвідношення розміру навчальної вибірки до розміру всієї вибірки: 0.7;
- метод розбиття вибірки на навчальну і перевірочну: випадковим чином;
- величина попередження прогнозу – на 2 значення вперед.

Всі методи були поставлені в рівні умови: використовувалися нейронні мережі однакової архітектури, однакові алгоритми навчання і один і той же метод розбиття на кластери – k-means. Результати порівняння, представлені у вигляді нормалізованої середньоквадратичної помилки моделі (отриманої за допомогою методу) на всіх даних після навчання, наведені в таблиці 4.1:

Таблиця 4.1. Нормалізована СКП методів

Часовий ряд / метод	Нейронна мережа	Кластеризація + нейронна мережа	Кластеризація + нейронна мережа з ініціалізацією ваг	Запропонований метод
Виробництво електрики в Австралії	0.0226	0.0312	0.0137	0.0213
Бенчмарк CATS [54]	0.0041	0.0169	0.0101	0.0041
Курс долара до євро	0.0763	0.0868	0.0979	0.0660
Курс долара до фунта	0.0621	0.2130	0.0643	0.0876
Індекс споживчих цін	0.0007	0.0535	0.0201	0.0005
Споживання електроенергії в Іспанії	0.0288	0.0520	0.0185	0.0274
Середні депозитні ставки в Іспанії	0.0842	0.1019	0.0743	0.0609
Індекс фондової біржі в Іспанії	0.0037	0.0115	0.0032	0.0038
Кількість плям на сонці за місяцями	0.2043	0.4424	0.1623	0.2016
Поставки авіації в США	0.6157	0.1453	0.2260	0.1967
Зимовий Північноатлантичний індекс	1.1040	1.0164	1.1725	1.0001
Сумарна помилка	2.2063	2.1709	1.8629	1.6698

Як видно з таблиці, сумарна нормалізована СКП запропонованого методу на 25% нижча, ніж у «методу-предка» – звичайної нейронної мережі, навченої на всіх даних, і на 11% менше, ніж у методу прогнозування з використанням кластеризації і навчанням нейронних мереж для кластерів з попередньою ініціалізацією ваг мереж. До речі, цей метод також належить до

групи глобально-локальних методів, оскільки ваги, використовувані для ініціалізації «локальних» ШНМ є, по суті, глобальною інформацією.

Висновки до розділу

У цьому розділі було розглянуто проблему прогнозування при неоднорідних даних – тобто коли поведінка прогнозованого об’єкту змінюється (можливо, декілька разів) на спостережуваному періоді. В результаті було запропоновано нову постановку задачі прогнозування, що явно враховує можливу неоднорідність даних, та нову методику прогнозування, що дозволяє використовувати існуючі алгоритми прогнозування, які не враховують неоднорідність даних у навчальній вибірці, коли ця неоднорідність існує. Запропонована методика була застосована до нейронних мереж і перевірена на реальних даних: отримана середньоквадратична помилка була менша ніж у таких методів, як ШНМ, локальні ШНМ та інші.

РОЗДІЛ 5. ПРОГРАМНА СИСТЕМА ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

Використовуючи вищеописані методи прогнозування була розроблена програмна система прогнозування часових рядів (ПСПЧР).

Дана система складається з п'яти основних компонентів.

- Компонент попередньої обробки даних – ці компоненти дають користувачеві можливість застосовувати до даних різні перетворення та / або фільтри (такі як фільтр змінного середнього, медіанний фільтр і так далі).
- Компоненти побудови прогнозуючої моделі – дані компоненти дозволяють будувати прогнозуючу модель, використовуючи різні методи прогнозування (ШНМ, МГУА).
- Компонент побудови графіків – компонент дозволяє візуалізувати часові ряди, у тому числі кілька рядів на одному графіку.
- Компонент завантаження даних – дозволяє завантажувати часові ряди з файлів.
- Компонент збереження даних – дозволяє зберігати часові ряди в файл.

Далі будуть описані: структура системи, список основних вирішуваних завдань, опис входів-виходів для кожного завдання, схема алгоритму роботи системи, опис інтерфейсу користувача і приклади роботи з системою.

5.1. Структура системи

Структура системи відповідає стандартній трирівневій схемі «Інтерфейс користувача» ⇔ «Логіка додата» ⇔ «Доступ до даних», де кожен рівень може «спілкуватися» тільки з сусідніми рівнями. Також, для спілкування між рівнями використовуються «інтерфейси» рівнів – тобто, не кожен клас або компонент певного рівня може безпосередньо спілкуватися з будь-яким класом (компонентом) рівня-сусіда – це призвело б до занадто сильної і непередбачуваної пов'язаності між компонентами додатка – а тільки

одному спеціально призначеному «интерфейсному» класу дозволено спілкуватися з інтерфейсними класами рівнів-сусідів.

Високорівнева структура додатка виглядає наступним чином.

- Контролер головної форми додатка реагує на дії користувача (перетягування елементів, з'єднання елементів, кліки по елементах), і передає (при необхідності) інформацію про дії користувача в інтерфейсний клас рівня «Логіка додатка».
- Інтерфейсний клас рівня «Логіка додатка» також реагує на дії користувача і сигналізує (при необхідності) контролеру головної форми про необхідність оновити форму. Для реагування на дії користувача інтерфейсний клас рівня «Логіка додатка» може також використовувати інтерфейсний клас рівня «Доступ до даних» або інтерфейси внутрішніх компонентів рівня «Логіка додатка» – кожен елемент («Вхідні дані», «Обробка», «Побудова моделі») має власний інтерфейс, і викликається (при необхідності) інтерфейсним класом рівня «Логіка додатка». Цей клас також відповідає за злагоджену роботу всіх компонентів рівня «Логіка додатка» в цілому, оскільки самі компоненти не знають про існування інших компонентів – вони являють собою відокремлені одиниці, які можна використовувати повторно (reusable components).
- Інтерфейс рівня «Доступ до даних» взагалі «не знає» нічого про логіку додатка, він лише надає методи виду «Дістати дані з файлу», «Завантажити дані в файл» і т.д.
- Форма налаштування кожного елемента працює тільки з відповідним елементом за чітко визначеним інтерфейсому виду «Встановити значення за ключем» і «Отримати значення за ключем». Таким чином, коли користувач налаштовує значення певного параметра, контролер форми налаштування просто викликає інтерфейсний метод «Встановити значення за ключем» для відповідного ключа, а вже сам компонент реагує потрібним чином.

5.2. Список основних вирішуваних завдань

1. Завантаження даних із зовнішнього джерела.
2. Відображення даних у вигляді графіка.
3. Попередня обробка даних (згладжування даних, отримання похідної ряду, дискретне перетворення Фур'є).
4. Прогнозування часового ряду.

4.1. Побудова прогнозуючої моделі, використовуючи навчальні дані.

4.2. Використання побудованої моделі для прогнозування нових даних.

5. Налаштування параметрів систем, які використовуються (прогнозуючої системи, систем попередньої обробки даних).

6. Збереження даних у зовнішнє джерело (файл).

5.3. Опис входів-виходів завдань

Задача «Завантаження даних із зовнішнього середовища»:

- **вхід:** дані із зовнішнього середовища у певному форматі;
- **вихід:** дані у внутрішньому форматі програмного комплексу.

Задача «Відображення даних у вигляді графіка»:

- **вхід:** набір даних у внутрішньому форматі програмного комплексу;
- **вихід:** вікно з відповідними графіками.

Задача «Попередня обробка даних»:

- **вхід:** дані у внутрішньому форматі програмного комплексу;
- **вихід:** оброблені дані у внутрішньому форматі програмного комплексу.

Задача «Прогнозування часового ряду»:

- **вхід:** набір даних у внутрішньому форматі програмного комплексу;

- **вихід:** прогноз, представлений у вигляді даних у внутрішньому форматі програмного комплексу та модель, що була використана для цього прогнозу.

Задача «Налаштування параметрів систем»:

- **вхід:** система, що налаштовується та нові параметри;
- **вихід:** налаштована система.

Задача «Збереження даних у зовнішнє джерело»:

- **вхід:** дані у внутрішньому форматі програмного комплексу;
- **вихід:** дані, збережені у зовнішнього середовища у певному форматі.

Для схематичного представлення входів та виходів задач та під задач дивись «Додаток А».

5.4. Схема алгоритму роботи системи

Найбільш інтенсивне застосування всієї структури програми починається після запуску змодельованої користувачем системи на виконання. При цьому:

- інтерфейсний клас рівня «Логіка додатку» перевіряє побудовану користувачем схему на наявність помилок. У випадку, якщо помилки наявні – запуск припиняється і користувачу повідомляється про відповідні помилки; в іншому випадку, інтерфейсний клас рівня «Логіка додатку» відшукує всі «вхідні» (ті, що тільки генерують дані) елементи зі списку всіх елементів, і асинхронно викликає у них метод «Виконати» (всі елементи реалізують відповідний інтерфейс базового елементу);
- вхідні елементи відпрацьовують (при цьому вони можуть викликати методи інтерфейсного класу «Доступ до даних»), кожен елемент після закінчення сигналізує інтерфейсному класу «Логіка додатка», що він закінчив. Якщо під час виконання виникає виняткова ситуація, то робота елемента припиняється негайно, і про це також сигналізується інтерфейсному класу «Логіка додатку». Крім того, якщо в результаті

виконання елементу користувач повинен отримати певну інформацію – у вигляді повідомлення, або графіку – ця інформація відображається на екрані;

- інтерфейсний клас рівня «Логіка додатка» реагує на закінчення роботи наступного елемента, і якщо відбувся виняток – причина виключення запам'ятовується, після чого він очікує на закінчення всіх вже запущених елементів, і повідомляє контролеру головної форми про «аварійне» закінчення разом з його ймовірною причиною. Якщо ж робота закінчилася нормальним чином – відшукуються всі елементи, які чекали закінчення цього елемента і якщо вони готові почати – у них також асинхронно викликається метод «Виконати». Якщо елемент, який закінчив роботу, останній – про це повідомляється контролеру головної форми, щоб він міг відключити режим очікування.

Для схематичного представлення алгоритмів роботи системи дивись «Додаток Б».

5.5. Приклад роботи з додатком

Складемо найпростішу схему, у якій перші 180 точок деякого часового ряду будуть використані для побудови моделі, використовуємо простий багатошаровий персептрон, а 120 точок, що залишилися, будуть прогнозуватися побудованою моделлю. Так само, для позбавлення від непотрібних шумів слід попередньо обробити часовий ряд медіанним фільтром.

Отже, спочатку подивимося, як будуть виглядати вихідні і згладжені медіанним фільтром дані. Для цього перетягнемо компоненти «Джерело даних», «Обробка даних» та «Графік» і з'єднаємо їх потрібним чином (рис.5.1).

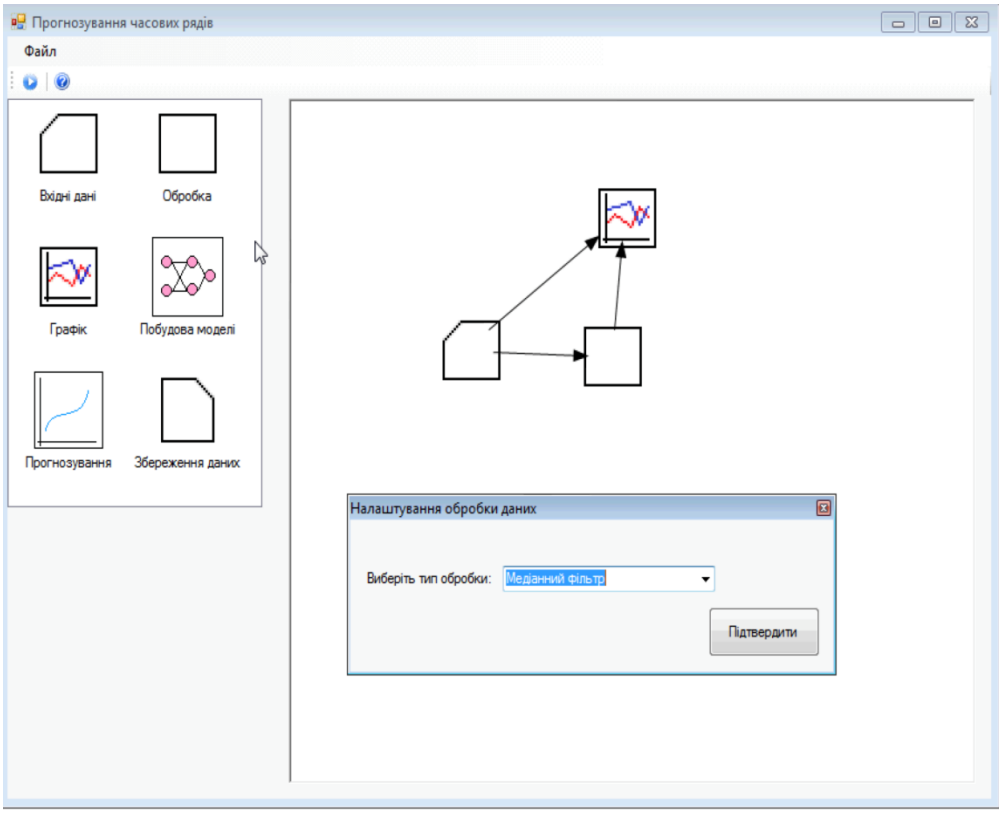

Рис. 5.1. Приклад роботи з додатком – побудова графіка

Після цього запустимо побудовану схему і отримаємо наступний графік (рис. 5.2).

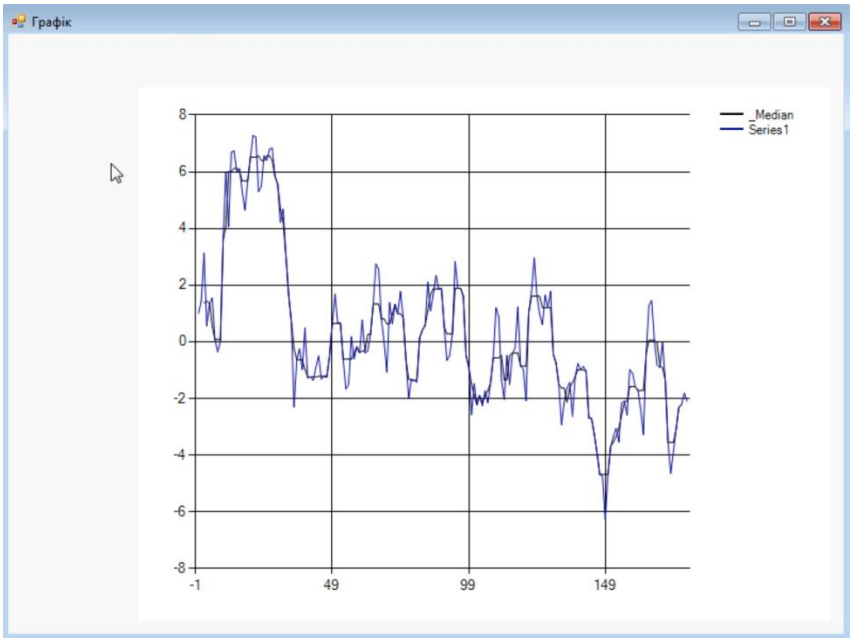

Рис. 5.2. Приклад роботи з додатком – побудований графік

Далі добудуємо схему таким чином, щоб модель будувалася за допомогою БП з цих 180 точок, а прогноз робився за рештою 120 точкам, які знаходяться в іншому файлі (рис.5.3).

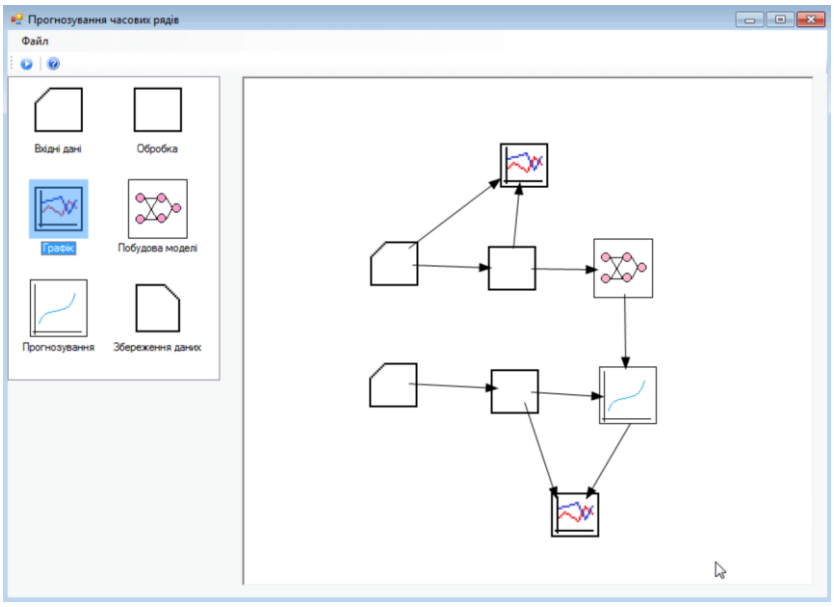

Рис. 5.3. Приклад роботи з додатком – побудова системи для прогнозування часового ряду

Запустимо побудовану схему на виконання і отримаємо наступний результат (рис.5.4).

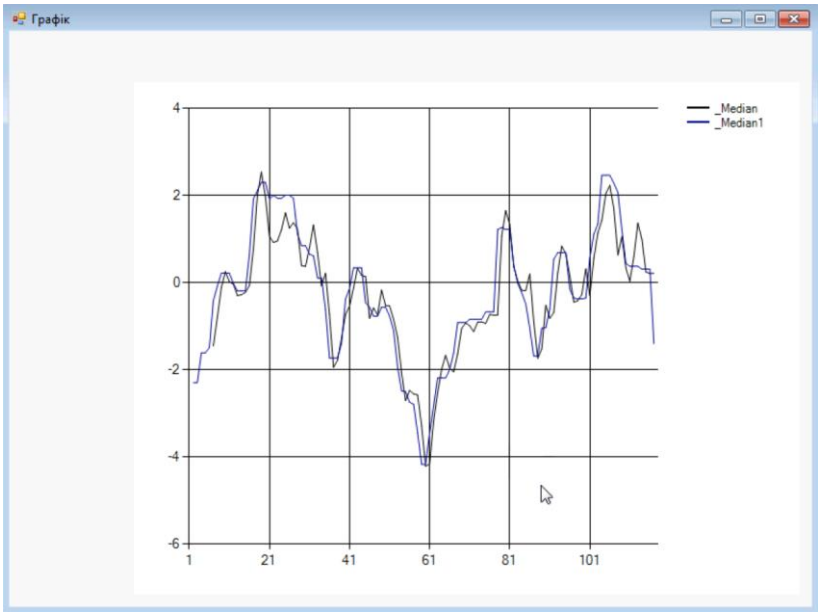

Рис. 5.4. Приклад роботи з додатком – результат прогнозу

Як бачимо, побудована модель дійсно є адекватною і добре виконала узагальнення даних.

Висновки до розділу

У цьому розділі було описано розроблений програмний комплекс прогнозування часових рядів, що має наступні властивості:

- інтуїтивно-зрозумілий drag-n-drop графічний інтерфейс;
- компонента структура, що дозволяє користувачу з легкістю будувати складні багаторівневі системи прогнозування;
- включає до свого складу розроблені в роботі нові методи прогнозування.

РОЗДІЛ 6. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

У цьому розділі розглядаються умови працівника під час роботи за електронно-обчислювальної машини при використанні програмного забезпечення, яке проектувалось під час написання дипломної роботи.

Передбачається, що працівник буде взаємодіяти безпосередньо з персональним комп’ютером, на якому буде встановлене необхідне програмне забезпечення в офісному приміщенні.

При роботі з персональним комп'ютером користувач відчуває значні навантаження на зір, м'язові навантаження, піддається електромагнітному випромінюванню, в зв'язку з переробкою великого обсягу інформації у користувача настає стомлюваність, що призводить до зниження працездатності і збільшення кількості помилок у виконуваній роботі. Так само негативно впливають: неправильне освітлення, неправильне планування робочого місця, погана вентиляція і обігрів приміщення.

Таким чином, при організації робочого місця важливо приділити увагу питанням охорони праці. Від умов роботи працівників залежить швидкість та якість роботи.

6.1. Характеристика виробничого приміщення

Досліджуваним робочим приміщенням є кімната (рис.6.1), що містить вісім робочих місць програмістів.

Загальна площа приміщення складає 51 квадратний метр, а загальний об‘єм – 163,2 метрів кубічних. Висота стелі – 3,2 метри. Площа на одне робоче місце становить 6,375 м2, а об'єм – 20,4 м3.

Робочі столи виготовлені з ДСП та мають два метри завдовжки та один метр завширшки. Сидіння робочих місць відповідають ергономічним вимогам для найбільш зручного положення тіла при роботі з ВДТ. Відстань

між користувачем та ВДТ складає приблизно 700 мм. Висота робочої поверхні столу становить 750 мм.

Відеодисплейний термінал на кожному з робочих місць розташований на відстані 600 мм від краю поверхні стола. Окрім терміналу, кожне робоче місце містить власне комп‘ютер, клавіатуру та мишу.

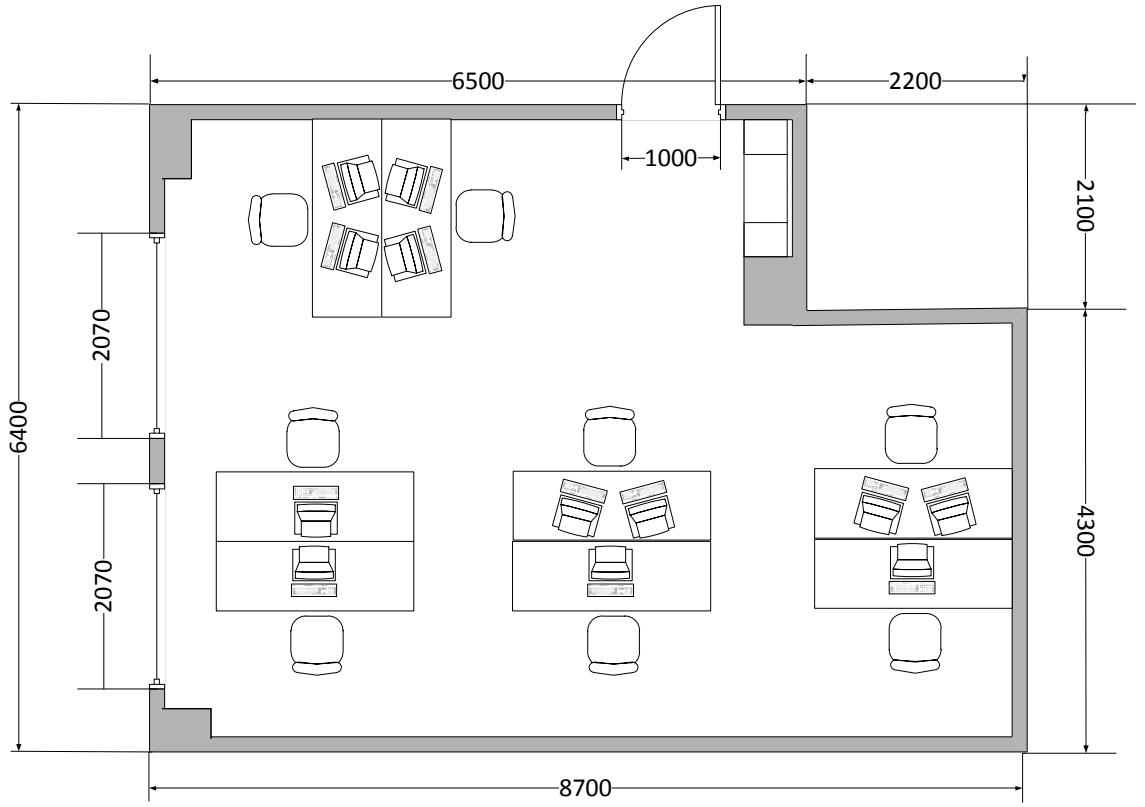

Рис. 6.1. План приміщення

6.2. Ергономічна безпека

Правильна організація робочих місць запобігає передчасній втомлюваності користувача і сприяє збереженню здоров'я. Організація робочого місця передбачає:

• правильне розміщення робочого місця у виробничому приміщенні. Робочі місця у приміщенні групуються по 2 стола та знаходяться на відстані 1 м одне від одного. Площа на одне робоче місце становить не менше ніж 6,375 м2, а об'єм – не менше ніж 20,4 м3. Загальна площа приміщення складає 51 квадратний метр, а загальний об‘єм – 163,2 метрів кубічних. Висота стелі – 3,2 метри;

• вибір ергономічно обґрунтованого робочого положення, виробничих меблів з урахуванням характеристик людини. Робочі столи виготовлені з ДСП та мають два метри завдовжки та один метр завширшки. Сидіння робочих місць відповідають ергономічним вимогам для найбільш зручного положення тіла при роботі з ВДТ: тулуб розташований вертикально, кут, що утворюється між тулубом та стегновими кустками, складає 90 градусів, кут між стегном та гомілкою складає 90-110 градусів, кут між тулубом та передпліччям складає приблизно 0 градусів. Руки лежать на столі, відстань між користувачем та ВДТ складає приблизно 700 мм. Висота робочої поверхні столу повинна регулюватися у межах 680 – 800 мм; у середньому вона становить 725 мм;

• раціональне компонування обладнання на робочих місцях. Відеодисплейний термінал на кожному з робочих місць розташований на відстані 600 мм від краю поверхні стола. Окрім терміналу, кожне робоче місце містить власне комп‘ютер, клавіатуру та мишу;

• урахування характеру й особливостей трудової діяльності. Організація робочого місця користувача ВДТ має забезпечуватися відповідно до НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин».

Що стосується умов праці користувачів ВДТ, пов'язаних з гігієною площі і виробничих приміщень, то вона також має відповідати НПАОП 0.00-1.28-10, згідно з якими площа, де має розташовуватися одне робоче місце з ВДТ, має становити не менш ніж 6,0 м2, а об'єм приміщення бути не менше 20,0 м3.

6.3. Мікроклімат виробничого приміщення

Згідно ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень», нормування параметрів проводиться в залежності від періоду року та категорії важкості виконуваних робіт. Для постійних робочих місць, якими є робочі місця операторів ПК, встановлені оптимальні параметри

мікроклімату, а при неможливості їх дотримання використовують допустимі параметри. Робота за персональним комп'ютером по енерговитратами відноситься до категорії легких Іб. В таблиці 6.1. наведені оптимальні параметри мікроклімату в приміщеннях, де виконуються роботи операторського типу.

Таблиця 6.1. Параметри мікроклімату для приміщень з ПК

Період року	Параметр мікроклімату	Величина
Холодний	Температура повітря в приміщенні	21...23°С
	Відносна вологість	40... 60%
	Швидкість руху повітря	до 0,1 м/с
Теплий	Температура повітря в приміщенні	22...24°С
	Відносна вологість	40…60%
	Швидкість руху повітря	0,1...0,2 м/с

Виміряні за допомогою приладів температура та вологість у приміщенні відповідають вказаним у таблиці для теплого періоду року.

Слід зазначити, що для нормалізації параметрів мікроклімату слід використовувати у приміщеннях кондиціювання повітря, або забезпечити подачу свіжого повітря системами вентиляції. Норми подачі свіжого повітря наведені у табл. 6.2.

Таблиця 6.2. Норми подачі свіжого повітря в приміщення з ПК

Характеристика приміщення	Об'ємна витрата свіжого повітря, що подається в приміщення, $м^3$ на одну людину в
Об'єм до 20$м^3$ на людину	Не менше 30
20... 40 $м^3$ на людину	Не менше 20
Більше 40 $м^3$на людину	Може біти використана природна вентиляція

6.4. Освітлення

Наведемо для даного робочого місця характеристики зорових робіт :

- мінімальний розмір об'єкта розпізнавання – більше 5 мм; (мінімальний об’єкт розпізнавання в даній кімнаті є комп’ютер, текст на моніторах);
- контраст між об'єктом розпізнавання і фоном – є досить різний, оскільки він залежить від кольорів на лінії зору монітора комп’ютера;
- характеристику фону – також є різний;
- відносна тривалість зорової роботи в напрямку зору на робочу поверхню – менше 70%.

1. Розряд зорової роботи – середньої точності (оскільки об’єкт розпізнавання більше 0.5 мм);

2. Підрозряд – 2 (оскільки відносна тривалість зорової роботи в напрямку зору на робочу поверхню складає менше 70%);

3. Система природного освітлення – бокова;

4. Система штучного освітлення – загальна;

5. КПО для природного освітлення $e_н = 2\%$;

6. Нормоване значення освітленості для загальної системи штучного освітлення $E_н = 150$ лк.

Роботи в даному приміщенні ведуться у світлий час доби при природному освітленні, у темне – при штучному.

Природне освітлення здійснюється через два віконних прорізи розмірами 1,5х1,6 м та 1,4х1,6 м.

Одне вікно робочого приміщення виходить на південь, інше на північний схід.

Тип природного освітлення – бічне, однобічне.

Розрахуємо нормативний коефіцієнт природної освітленості для робочого приміщення.

$$e_н = e_n \cdot m_N$$

де $e_n = 0.5$ – значення КПО за таблицями 1 і 2 державних будівельних норм;

$m_N = 0.85$ – коефіцієнт світлового клімату за таблицею 4 державних будівельних норм;

N – номер групи забезпеченості природним світлом за таблицею 4 державних будівельних норм.

$$e_н = 0.5 * 0{,}85 = 0{,}425\%$$

Перевіримо відповідність розрахованого значення $e_н$ дійсному $e_{дійсн}$ за допомогою приблизного проектного розрахунку.

Знайдемо світловий коефіцієнт природного освітлення за формулою:

$$e_{дійсн} = \frac{S_{ВІК}}{S_П} * 100$$

де $S_{вік}$ – площа вікон, м^2;

$S_п$ – площа підлоги, м^2.

$$e_{дійсн} = \frac{S_{ВІК}}{S_П} = \frac{1{,}5*1{,}6+1{,}4*1{,}6}{18} * 100 = \frac{4{,}64}{18} * 100 = 25{,}7\%$$

Оскільки 25,7% більше за 0,425 %, то природного світла достатньо для роботи в розглянутому приміщенні у світлий час доби.

В відповідності з [55], якщо в приміщенні не вистачає до норми природного освітлення та для освітлення в темний період доби, то воно повинне бути доповнене штучним.

Аналіз штучного освітлення

Для штучного освітлення використовуються люмінесцентні лампи – типу ЛБ 20, 8 штук. По чотири лампи в одному світильнику, всього два світильники.

Виконаємо аналіз штучного освітлення, який полягає у визначенні фактичної освітленості ($E_ф$), що створює система загального штучного освітлення, у порівнянні її з нормованим значенням $E_н$.

Для визначення фактичної освітленості скористаємося методом коефіцієнта світлового потоку, основне розрахункове рівняння якого має такий вигляд :

$$E_ф = \frac{F_л N n \eta}{S k_З z}$$

де $E_ф$ – нормована освітленість, лк;

N = 2 – кількість світильників;

п = 4 – кількість ламп у світильнику;

η = 0.53 – коефіцієнт використання світлового потоку;

S = 51 м^2 – площа приміщення;

$к_З$ = 1.3 – коефіцієнт запасу;

z = 1.15 коефіцієнт нерівномірності освітлення;

$F_л$ = 1020 лк. – світловий потік лампи.

$$E_ф = \frac{1020 \cdot 2 \cdot 4 \cdot 0{,}53}{18 \cdot 1{,}3 \cdot 1{,}15} = 160 \text{ лк} > E_н = 150 \text{ лк.}$$

Отже, визначивши і порівнявши фактичну освітленість ($E_ф$) з нормованим значенням $E_н$, яке визначається згідно державним будівельним норми України [55], було визначено що штучного освітлення достатньо.

6.5. Характеристика випромінювання

Усе обладнання, що використовується в даному приміщенні, відповідає нормам, щодо електромагнітного випромінювання.

6.6. Аналіз рівня шуму

Можливими джерелами шуму в даному приміщенні є кулери процесорів, клавіатур. Сумарний рівень шуму в приміщенні складає близько

48дБА(з них 45.9дБА – шум від кулерів процесорних блоків, 2.1дБА – шум від клавіатур). Згідно ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку», еквівалентний рівень шуму має становити менше ніж 50дБА, тобто рівень шуму в приміщенні нижче за допустиму норму.

6.7. Електробезпека

Приміщення відноситься до приміщень буз підвищеної електронебезпеки.

Приміщення живиться електричною енергією від централізованого трансформатора, який знаходиться на поверсі поза межами приміщення і понижує напругу до 220 В, змінним струмом частотою 50 Гц. Комп’ютери мають власні блоки живлення, що перетворюють змінний струм мережі у постійний (5 В, 12 В). Потужність даних блоків живлення – 400 – 450 Вт. ПК у приміщенні відносяться до категорії І захисту від ураження електричним струмом, оскільки усі прилади при підключенні до електричної мережі яка заземляються, використовуючи систему TN. Тобто система, у якій живильні мережі мають глухе заземлення однієї точки струмопровідних частин джерела живлення, а електроприймачі і відкриті провідні частини електроустановки приєднуються до цієї точки за допомогою відповідно нейтрального і захисного провідників.

Корпуса всіх електричних пристроїв виготовляються з неструмопровідних матеріалів, а живлення здійснюється спеціальним кабелем, так щоб виключити ураження людини електричним струмом. Зазначені міри електробезпеки відповідають вимогам, пропонованим до побутових приладів відповідно до міжнародного стандарту СЕ.

У приміщенні електропроводка трифазна, чотирьохжильна, з подвійною ізоляцією. Всі кабелі сховані в коробах. Штепсельні розетки встановлені на висоті одного та 0,2 метри від підлоги. Вимикачі на стінах розташовані на висоті 1,2 метра від підлоги з боку ручки для відкривання

дверей. Таким чином, вимоги НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин» виконано.

6.8. Пожежна безпека

В приміщенні, що аналізується, наявні важкогорючі (елементи ЕОМ та інші електричні пристрої, з'єднувальні кабелі) та горючі (дерев'яні столи, шафи, паркет з натурального дерева та паперова документація). Горючі гази та горючі пиловидні речовини в завислому стані, що складають значну вибухову та пожежну небезпеку, а також горючі рідини в даному приміщенні відсутні.

Приміщення, для якого проводиться аналіз стану охорони праці, згідно НАПБ Б.03.002-2007 «Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» відноситься до категорії В, класу П-IIа по пожежонебезпеці.

В якості заходів щодо забезпечення пожежної безпеки прийняті наступні:

- призначений відповідальний за пожежну безпеку приміщення;
- розроблено план евакуації людей;
- у приміщенні заборонено користуватися нагрівальними приладами, курити, користуватися відкритим вогнем;
- є первинні засоби пожежогасіння у приміщенні (3 вогнегасника ВВ-2);
- встановлена пожежна сигналізація.

Відповідно до НАПБ Б.03.001-2004 «Типові норми належності вогнегасників» у приміщенні встановлено 3 вогнегасники типу ВВ-2 для тушіння невеликих джерел займання і устаткування під напругою до 1000 В. Відстань між місцями розташування вогнегасників не повинна перевищувати 15 м.

Для вчасного попередження про пожежу у приміщенні встановлена порогова пожежна сигналізація з пасивними сенсорами диму. У разі спрацювання системи сигнал надходить до чергового у лікарні.

Можливим джерелом запалювання є короткі замикання, що виникають внаслідок неправильного монтажу або експлуатації ЕОМ, старіння чи пошкодження ізоляції.

Для запобігання пожежі проводять планову заміну пошкоджених чи занадто старих кабелів. Монтаж нових ЕОМ та їх оновлення і ремонт проводяться за встановленою документацією. Експлуатація дозволяється тільки після проходження працівниками інструктажу від спеціаліста.

Ширина дверей 1м, при нормі не менше 0.9м, висота проходу 2,2 м при нормі не менше 2м, ширина коридору 2м (для одностороннього розташування дверей ширина евакуаційного шляху до світла, рівна ширині коридору, яку зменшено на половину ширини дверного полотна, повинна бути не менше 1м), що задовольняє вимогам.

6.9. Охорона праці та безпеки в умовах надзвичайних ситуацій

На відстані 1100 м від будівлі де розміщене робоче місце оператора ЕОМ в наслідок аварії або необережного поводження, може статися вибух приблизно 400т пропану. Тому, необхідно розрахувати можливі наслідки та вжити необхідні заходи щоб мінімізувати їх.

Вихідні дані:

1) Відстань від офісу до місця аварії (вибуху) – 1100 м;
2) Маса пропану – 400 т;
3) Характеристики елементів будівлі:

- будівля – зі збірного залізо бетону;
- кабельні лінії – наземні;
- контрольно-вимірювальна апаратура – наявна;
- границі вогнетривкості несучих стін – 3 год;
- границі вогнетривкості перегородок – 1 год.

4) Категорія виробництва з пожежної безпеки – В;

5) Щільність забудови об’єкту – 48%.

Розрахункова частина:

Зона I : $r1 = 17{,}5 \cdot \sqrt[3]{400} = 128.94\ м$ ΔРф = 1700 кПа

Зона II: $r2 = 1{,}7 \cdot r1 = 1{,}7 \cdot 128{,}94 = 219{,}2\ м$

$$\Delta PII = 1300 \cdot (r_1 / R_0)^3 + 50 = 1300 \cdot (128{,}94 / 1100)^3 + 50 = 50{,}002$$

Висновок: об’єкт знаходиться за межами I та II зон, тобто у зоні повітряної ударної хвилі (зона III)

$$\Delta P_\phi = \frac{262}{\sqrt{1 + 7{,}66 \cdot 10^{-5} \cdot \frac{L^3}{Q}} - 1} = \frac{262}{\sqrt{1 + 7{,}66 \cdot 10^{-5} \cdot \frac{1100^3}{400}} - 1} = 17{,}471$$

ΔРф = 17,471 кПа

При цьому маємо, наступні показники:

- Ступінь руйнування будівлі – слабкі;
- Ступінь руйнування контрольно-вимірювальної апаратури – середні;
- Ступінь руйнування кабельних ліній – слабкі;
- Ступінь ураження людей від прямої дії УХ – відсутні;
- Ступінь вогнестійкості – I;
- Небезпечна кількість вибухової речовини для уникнення руйнувань будівлі – менше 150 т;
- Небезпечна кількість вибухової речовини для уникнення будь-яких руйнувань – менше 40 т;

o Характеристика руйнувань будівлі: спостерігаються руйнування заповнень дверних та віконних прорізів, зривання покрівлі даху.

o Характеристика ураження людей: можливе отримання легких травм внаслідок вибиття вікон та дверей при дії вибухової хвилі

o Очікувана пожежна обстановка: для виробництва категорії пожежної небезпеки В та ступеня вогнестійкості будівель – І, при надмірному тиску 17,471 кПа пожежі не передбачуються.

Висновки:

На відстані 1100 м від будівлі стався вибух пропану, що призвело до часткових пошкоджень будівлі, а саме вибиття вікон, дверей та зривання покрівлі даху, а також наявність потерпілих людей з легкими травмами. В першу чергу необхідно надати першу медичну допомогу постраждалим та викликати швидку медичну допомогу.

Рекомендації що до зменшення заподіяної шкоди та уражень людей:

- установити на вікнах захисні металеві сітки, щоб розбите скло не потрапляло в приміщення;
- порушити питання перед відповідними органами про зменшення запасу вибухонебезпечної речовини до безпечної кількості.

Висновок до розділу

Виходячи із даного розділу, можна відзначити, що приміщення, яке використовується програмісти для створення програмного забезпечення, має достатню ізоляцію, хороше освітлення. Для забезпечення пожежної безпеки виконується всі необхідні дії, по її запобігання, а також має достатню кількість вогнегасників у у випадку непередбачених ситуацій. Приміщення відповідає всім вимогам, які висуваються стандартами.

ВИСНОВКИ

У ході роботи над магістерською дисертацією було зроблено наступні висновки та досягнуто наступних результатів.

1. Найбільш перспективними методами прогнозування для подальших досліджень є ШНМ та МГУА: обидва методи дозволяють знаходити складні нелінійні залежності між вхідними та вихідними змінними та не потребують явного вибору моделі. Крім того, ШНМ мають перевагу у вигляді можливості навчання, що дозволяє використовувати їх для онлайн-прогнозування, а МГУА дозволяє знаходити оптимальну структуру моделі.
2. Виділено основні проблеми, що виникають при вирішенні задачі прогнозування, а саме:
   - неможливо врахувати всі чинники, що впливають на процес, який ми намагаємося прогнозувати; більше того, їх вплив може змінюватися з плином часу - фактор, що не був важливий сьогодні, може відіграти важливу роль завтра;
   - завжди є багато (іноді нескінченне число) можливих моделей, що добре відповідають навчальним даними - ми повинні вирішити, яку модель або множину моделей використовувати, і такі рішення зазвичай дуже схильні до помилок;
   - часто буває важко (якщо не неможливо) знайти оптимальну складність моделі.
3. На основі аналізу проблем, що виникають при вирішенні задачі прогнозування було розроблено:
   - нову постановку задачі прогнозування, яка явно враховує можливу непостійність впливу вхідних факторів на поведінку об’єкту, що прогнозується;

- новий метод прогнозування, що об’єднує підходи ШНМ і МГУА, та дозволяє знаходити оптимальну структуру нейронної мережі для даної конкретної задачі прогнозування;
- нову методику прогнозування, що дозволяє використовувати існуючи методи прогнозування для практичних задач прогнозування, де наявна проблема неоднорідності даних.

4. Проведено тестування нових розроблених методів на наборі реальних даних, і отримана помилка прогнозу (у вигляді середньоквадратичної помилки) менша ніж у таких методів, як ШНМ, лінійна регресія та інші.
5. Реалізовано програмний комплекс прогнозування часових рядів. Основні характеристики та особливості розробленого комплексу:
   - інтуїтивно-зрозумілий інтерфейс, що використовує принцип drag-n-drop;
   - складається з набору компонентів (таких як компонент завантаження вхідних даних, компонент побудови прогнозуючої моделі та інші), які користувач може з’єднувати потрібним чином для побудови складних багаторівневих систем прогнозування;
   - у якості алгоритмів, що використовуються для побудови прогнозуючої моделі, можуть використовуватися як існуючи алгоритми, так і нові розроблені алгоритми.

ПЕРЕЛІК ПОСИЛАНЬ

1. An algorithm for solving the problem of forecasting / Victor Sineglazov, Elena Chumachenko, Vladyslav Gorbatyuk // Aviation. -2013. №1(17), p. 9-13.

2. Victor Sineglazov, Elena Chumachenko & Vladyslav Gorbatyuk (2014) Using a mixture of experts' approach to solve the forecasting task, Aviation, 18:3, p. 129-133.

3. Gorbatiuk, Vladyslav; Sineglazov, Victor; Chumachenko, Olena. A method for building a forecasting model with dynamic weights. Eastern-European Journal of Enterprise Technologies, [S.l.], v. 2, n. 4(68), p. 4-8, apr. 2014. ISSN 1729-4061.

4. Алгоритм решения задачи прогнозирования / Е.И. Чумаченко, В.С. Горбатюк // Штучний інтелект. – 2012. – № 2., С. 24-31.

5. Метод решения задачи прогнозирования на основе комплексирования оценок / В. М. Синеглазов, Е. И. Чумаченко, В. С. Горбатюк // Індуктивне моделювання складних систем, випуск 4, 2012,

С. 214-223.

6. Интеллектуальная система прогнозирования рисков послеоперационных осложнений / Синеглазов В. М., Чумаченко Е. И., Горбатюк В. С. // ISDMCI'2013, С. 289-291.

7. Использование искусственных нейронных сетей для задачи прогнозирования / Е. И. Чумаченко, В. С. Горбатюк // Електроніка та системи упр. – 2012. – № 1, С. 113-119.

8. Applying Different Neural Network's Topologies to the Forecasting Task – Gorbatiuk, Vladyslav; Sineglazov, Victor; Chumachenko, Olena. – International Conference in Inductive Modelling ICIM' 2013, С. 217-220.

9. One approach for the forecasting task solution – Gorbatiuk, Vladyslav; Sineglazov, Victor; Chumachenko, Olena. – V Всесвітній конгрес

«Авіація у XXI столітті» – «Безпека в авіації та космічні технології», p. 3.5.49-3.5.53.

10. Method for predicting failure risk of UAV navigation systems (2012) – Gorbatiuk, Vladyslav; Chumachenko, Olena. – Methods and systems of navigation and motion control (MSNMC 2012), p. 63-65.

11. Алгоритм решения задачи прогнозирования (2012) – Чумаченко, Горбатюк – Міжнародна наукова конференція «Інтелектуальні системи прийняття рішень і проблеми обчислювального інтелекту ISDMCI-2012», С. 423-425.

12. Francis X. Diebold (2007), Elements of Forecasting, 4th edition.

13. Ивахненко А.Г., Ивахненко Г.А. Обзор задач, решаемых по алгоритмам Метода Группового Учета Аргументов (МГУ А)// http://www.gmdh.net.

14. В.Феллер. Глава XI. Целочисленные величины. Производящие функции // Введение в теорию вероятностей и её приложения = An introduction to probability theory and its applicatons, Volume I second edition / Под ред. Е. Б. Дынкина. – 2-е изд. – М.: Мир, 1964. – С. 270 – 272.

15. Box, George; Jenkins, Gwilym (1970). Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day.

16. Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. Philos. Trans. Roy. Soc. London Ser. A 222 309–368.

17. Ивахненко А. Г. Индуктивный метод самоорганизации моделей сложный систем. – Киев: Наукова думка, 1982. – 245с.

18. Мак-Каллок У. С., Питтс В., Логическое исчисление идей, относящихся к нервной активности // В сб.: «Автоматы» под ред. К. Э. Шеннона и Дж. Маккарти. – М.: Изд-во иностр. лит., 1956. – С. 363-384.

19. Kalman, R.E. (1960). "A new approach to linear filtering and prediction problems". Journal of Basic Engineering 82 (1): pp. 35–45

20. Vogel,L., Owens,A., and Walsh,M., Iskustevennyj Intellekt i Evolyutsyonnoe Modelirovanie. (Artificial Intelligence and Evolutinary Modeling). Moscow: Nauka, 1969.

21. Алесинская Т.В. Методы скользящего среднего и экспоненциального сглаживания / Т.В. Алесинская // Экономико-математические методы и модели : уч. пособие по решению задач по курсу. – Таган-рог : Изд-во ТРТУ, 2002. – 153 с.

22. Klevecka Irina. Pre-Processing of Input Data of Neural Networks: The Case of Forecasting Telecommunication Network Traffic / Irina Klevecka, Janis // Telektronikk 3/4.2008. – C. 168-178.

23. Зорич В. А. Математический анализ. – М.: Физматлит, 1984. – 544 с.

24. Addison P.S. The Illustrated Wavelet Transform Handbook. – IOP, 2002.

25. Wong F. S. Time series forecasting using backpropagation neural networks // Neurocomputing. – 1990/91. – 2. – P. 147-159.

26. Groot de C., Wuertz D. Analysis of univariate time series with connectionist nets: a case study of two classical examples // Neurocomputing. – 1991. – 3. – P. 177-192.

27. Connor J. T., Martin R. D., Atlas L. E. Recurrent neural networks and robust time series prediction // IEEE Trans. on Neural Networks. – 1994. – 5. – P. 240–254.

28. Saxen H. Nonlinear time series analysis by neural networks. A case study // Int. J. Neural Systems. – 1996. – 7. – P. 195-201.

29. Madhavan P. G. A new recurrent neural network learning algorithm for time series prediction // J. of Intelligent Systems. – 1997. – 7. – P. 103 – 116.

30. Conway A. J., Macpherson K. P., Brown J. C. Delayed time series predictions with neural networks // Neurocomputing. – 1998. – 18. – P. 81 – 89.

31. Бодянський Є. В., Кулішова Н. Є., Руденко О. Г. Рекурентна прогнозуюча штучна нейронна мережа: архітектура та алгоритми навчання //

Адаптивні системи автоматичного управління. – Дніпропетровськ: Системні технології, 1999. – Вип. 2 (22). – С. 129-137.

32. Бодянский Е. В., Попов С. В., Штефан А. Нейросетевой упредитель многомерных стохастических последовательностей // Праці Міжнар. конф. з автоматичного управління «Автоматика-2000»:Т.2 – Львів: ДНДІ інформаційної інфраструктури, 2000. – С. 40-42.

33. Bodyanskiy Ye., Kolodyazniy V., Kulishova N. Generalized forecasting Sigma-Pi neural network // In “Intelligent Technologies – Theory and Applications”.–Amsterdam: IOS Press, 2002. – P.29-33.

34. Packard N., Crutchfield J., Farmer J., Shaw R. Geometry from a time series // Phys. Rev. Lett. – 1980. – 45. – P. 712-716.

35. Бодянский, Е. В. Искусственные нейронные сети: архитектуры, обучение, применения / Е. В. Бодянский, О. Г. Руденко // Харьков : ТЕЛЕТЕХ, 2004. – 369 с. : ил.

36. Rosenblatt, Frank. x. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC, 1961.

37. Rumelhart, David E.; Hinton, Geoffrey E.; Williams, Ronald J. (8 October 1986). "Learning representations by back-propagating errors". Nature 323 (6088): 533–536.

38. Wan E. Temporal backpropagation for FIR neural networks // Int. Joint Conf. on Neural Networks. – V.1. – San Diego, 1990. – P. 575 – 580.

39. Wan E. A. Time series prediction by using a connectionist network with integral delay lines / Eds. by A. Weigend, N. Gershenfeld “Time Series Prediction. Forecasting the Future and Understanding the Past”. – SFI Studies in the Sciences of Complexity. – V. XVII. – Reading: Addison-Wesley, 1994. – P. 195 – 218.

40. Back A. D., Wan E. A., Lawrence S., Tsoi A. C. A unifying view of some training algorithms for multilayer perceptrons with FIR filter synapses / Eds. by J. Vlontzos, J. Hwang, E. Wilson “Neural Networks for Signal Processing 4”. – N.Y.: IEEE Press, 1994. – P. 146 – 154.

41. Yu H.-Y., Bang S.-Y. An improved time series prediction by applying the layer-by-layer learning method to FIR neural networks // Neural Networks. – 1997. – 10. – P. 1717 – 1729.

42. Уидроу Б., Стирнз С. Адаптивная обработка сигналов. – М.: Радио и связь, 1989. – 440 с.

43. Elman, J.L. Finding structure in time. // Cognitive Science. – 1990. – С. 179-211.

44. Amir F. A. A comparison between neural-network forecasting techniques / F. A. Amir, I. S. Samir – case study: river flow forecasting. // IEEE Transactions on neural networks. –Vol. 10, No. 2. – 1999. – p. 402–409.

45. Mohsen H. Artificial neural network approach for short term load forecasting for Illam region / H. Mohsen, S. Yazdan // World Academy of Science, Engineering and Technology 28 2007. – p. 280–284.

46. Short term load forecasting with multilayer perceptron and recurrent neural networks / Muhammad Riaz Khan, Cestm Ondrusek // Journal of ELECTRICAL ENGINEERING, VOL. 53, NO. 1-2, 2002, p. 17-23.

47. Jerome T. C. Recurrent neural networks and robust time series prediction / T. C. Jerome, R. M. Douglas, L. E. Atlas // IEEE transactions on neural networks. – Vol. 5, No. 2. –1994. – p. 240–254.

48. Горбань А. Н., Обобщенная аппроксимационная теорема и вычислительные возможности нейронных сетей, Сибирский журнал вычислительной математики, 1998. Т.1, № 1. С. 12-24.

49. Kenneth Levenberg (1944). «A Method for the Solution of Certain Non-Linear Problems in Least Squares». Quarterly of Applied Mathematics 2: p. 164–168.

50. U.S. General Aviation Aircraft Shipments and Sales 2012: http://www.bga-aeroweb.com/database/Data3/US-General-Aviation-Aircraft-Sales-and-Shipments.xls.

51. Friedman, J. H. (1991). "Multivariate Adaptive Regression Splines". The Annals of Statistics 19: 1.

52. MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability 1. University of California Press. pp. 281–297.

53. Data Sets for Time-Series Analysis [online] 2005. Available from Internet: http://tracer.uc3m.es/tws/TimeSeriesWeb/repo.html.

54. Lendasse, A., Oja, E., Simula, O., Verleysen, M. (2004). Time Series Prediction Competition: The CATS Benchmark. *International Joint Conference on Neural Networks, Budapest (Hungary), IEEE,* 1615-1620.

55. ДБН В.2.5-28-2006. Природне і штучне освітлення.