Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Факультет комп'ютерних наук
Кафедра теоретичної та прикладної системотехніки

«Затверджую»
Зав. кафедри теоретичної та
прикладної системотехніки
_____ д.т.н., проф. С. І. Шматков
«___» _____ 2022 р.

Пояснювальна записка

до кваліфікаційної роботи
магістра

на тему: «**МЕТОД ВИЯВЛЕННЯ ТА ОЦІНКИ ПАРАМЕТРІВ ТРЕНДУ У ТРАФІКУ КОМП'ЮТЕРНОЇ МЕРЕЖІ**»

Захищено на засіданні
Атестаційної комісії № 44
протокол № __ від __.12.2022 р.
Оцінка _____ / _____
Голова Атестаційної комісії
_____ **МІНУХІН С. В.**
(підпис)

Виконав:

студент 2 курсу, групи КІ – 61
Галузь знань: 12 – Інформаційні технології
Спеціальність: 123 – «Комп'ютерна інженерія»
ДОРОШЕНКО Максим Ігорович

Керівник:

к.т.н., доцент кафедри теоретичної та прикладної системотехніки
СТРІЛЕЦЬ Вікторія Євгенівна

Рецензент:

д.т.н., доцент, професор кафедри математичного моделювання і штучного інтелекту
СКОБ Юрій Олексійович

Харків – 2022

АНОТАЦІЯ

Пояснювальна записка до магістерської атестаційної роботи складається зі вступу, чотирьох розділів, висновків, списку використаних джерел і чотирьох додатків. Загальний обсяг роботи складає 91 сторінка, із яких 67 сторінок основної частини з 24 рисунками, 1 таблицею, 27 найменуваннями списку використаних джерел. ***Метою роботи*** є розробка комп'ютерної моделі виявлення трендів у трафіку комп'ютерних мереж та їх подальше оцінювання для підвищення якості прогнозування трафіку та характеристик мереж.

В роботі було виконано збір та попередню обробку даних про стан комп'ютерної мережі протягом певного часу, їх формалізація й подальший кількісний та якісний аналіз. Для аналізу даних використовувались методи статистичного аналізу. Для побудови моделі прогнозування трафіку використовувалась мова програмування Python з наявними зовнішніми пакетами такими як ThymeBoost, XGBoost, pymannkendall, statsmodels, scikit-learn, pmdarima та pandas. Під час прогнозування трафіку комп'ютерних мереж була застосована оцінка існуючих математичних моделей та обчислювальних методів.

В результаті роботи було запропоновано метод виявлення тренду, а також розроблено модель прогнозування ARIMA часового ряду з використанням методів машинного навчання (градієнтного бустингу).

Результати роботи можуть бути використані для аналізу та оцінки поточного трафіку комп'ютерних мережі для подальшої оптимізації, пошуку перспектив для подальшого розвитку у структурі мережі, виявлення проблем, що негативно впливають на систему.

Ключові слова: ARIMA, тренд, часові аналіз, комп'ютерні мережі, веб-трафік, моделі прогнозування, градієнтний бустинг, машинне навчання, регресійні моделі, статистичні методи.

ABSTRACT

The explanatory note to the master's attestation work consists of an introduction, four sections, conclusions, a list of used sources and four appendices. The total volume of the work is 91 pages, of which 67 pages are the main part with 24 figures, 1 table, 27 names of the list of used sources.

The purpose of the work is the development of a computer model for detecting trends in the traffic of computer networks and their further evaluation to improve the quality of traffic state forecasting.

Collection and pre-processing of input data, their formalization and subsequent quantitative and qualitative analysis were used in the work. Statistical analysis methods were used for data analysis. To build the traffic forecasting model, the Python programming language was used with available external packages such as ThymeBoost, pymannkendall, statsmodels, scikit-learn, pmdarima and pandas. Also, during the forecasting of computer network traffic, an assessment of existing mathematical models and computational methods was applied. As a result of the work, a trend detection method was proposed, as well as an ARIMA time series forecasting model was developed using machine learning methods (gradient boosting).

The results of the work can be used to analyze and evaluate the current traffic of computer networks for further optimization, search for prospects for further development in the network structure, detection of problems that negatively affect the system.

Keywords: ARIMA, trend, time analysis, computer networks, web traffic, prediction models, gradient boosting, machine learning, regression models, statistical methods.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ І УМОВНИХ ПОЗНАЧЕНЬ

МК – тест Манн-Кендалла;

КМ – комп’ютерна мережа;

ЧР – часовий ряд;

ARIMA – Autoregressive Integrated Moving Average;

MAPE – Mean Average Percentage Error;

MAE – Mean Average Error;

RMSE – Root Mean Squared Error;

ADF – аугментований тест Дікі-Фулера.

ВСТУП

Сучасність характеризується стрімким зростанням обсягів трафіку комп'ютерних мереж. Основним завданням при модернізації комп'ютерної мережі є аналіз мережевого трафіку, а саме його структури та обсягу. Особливої уваги потребує виділення тенденцій зміни трафіку мереж за певний період часу, що дасть змогу більш точно спрогнозувати навантаження на канали зв'язку. Саме для вирішення цієї задачі пропонується застосувати методи виявлення трендів трафіку комп'ютерних мереж для розробки комп'ютерної моделі прогнозування показників трафіку комп'ютерної мережі.

Актуальність роботи. В сучасній статистичній теорії існує багато різноманітних методів прогнозування інформації. Значна їх частина відноситься до прогнозування часових рядів. Особливістю прогнозування часових рядів є те, що аналізуються лише дані спостережень без додаткової інформації, без аналізу впливу зовнішніх сил. Звичайно, такий аналіз виглядає досить неповним, але доволі часто прогнози часових рядів є більш точними. Набір методів аналізу досить широкий. Деякі методи є більш-менш універсальними, інші – більш спеціалізованими та вимагають подальшого розроблення й апробування. Велика різноманітність наявних методів, недостатня обізнаність фахівців про особливості використання тих чи інших методів, складність застосування математичного апарату створюють для аналітиків труднощі та навіть призводять до формулювання неправильних висновків. Дослідженню часових рядів приділяється багато уваги в роботах вітчизняних та зарубіжних науковців.

Метою дослідження є розробка комп'ютерної моделі виявлення трендів трафіку комп'ютерних мереж та їх подальше оцінювання для підвищення якості прогнозування трафіку та характеристик мереж.

Об'єкт дослідження – процеси передачі даних у комп'ютерних мережах.

Предмет дослідження – моделі і методи оцінки параметрів характеристик комп’ютерних систем, зокрема трафіку.

Завдання дослідження:

1. Визначення постановки задачі оцінки тренду часового ряду.
2. Формування та формалізація вхідних даних для побудови моделі прогнозування, а також їх підготовка та обробка для покращення ефективності методу.
3. Розробка (або модифікація існуючого) методу виявлення тренду даних.
4. Розробка моделі прогнозування часового ряду з використанням сучасних засобів моделювання.
5. Покращення моделі за допомогою методів машинного навчання, а також порівняння ефективності моделей до та після застосування обраного методу машинного навчання.

Методи дослідження: теорія множин, методи системного аналізу, евристичні методи (деякі методи машинного навчання), методи аналізу інформативності, методи оцінки якості класифікації.

РОЗДІЛ 1
АНАЛІЗ ЧАСОВИХ РЯДІВ

Часові ряди — це актуальний інструмент, який застосовується в багатьох рішеннях, від прогнозування цін на акції, прогнозів погоди, планування бізнесу до розподілу ресурсів [1]. Незважаючи на те, що прогнозування може бути зведене до побудови контрольованої регресії, існують особливості, пов'язані з тимчасовим характером спостережень, які необхідно враховувати, рядів можна назвати:

- опис характеристик та закономірностей ряду;
- моделювання – побудова моделі досліджуваного процесу;
- прогнозування – передбачення майбутніх значень ЧР;
- управління – знаючи властивості часових рядів, можна виробити методи впливу на відповідні бізнес-процеси для керування ними.

Рішення будь-якої задачі з аналізу і прогнозуванню ЧР починається з побудови графіка досліджуваного показника. На етапі графічного аналізу можна досліджувати компонентний склад ЧР, а також зробити перші кроки до вибору моделі для опису динаміки і подальшого прогнозування. Під час аналізу пристальну увагу приділяють відокремленню тенденцій або трендів. Це допомагає при спробі прогнозування результату деякого явища ЧР.

1.1 Аналіз моделі часового ряду

Існують дві основні цілі аналізу часових рядів: визначення природи ряду та прогнозування (пророцтво майбутніх значень ЧР за справжніми та минулими значеннями). Обидві ці цілі вимагають, щоб модель ряду була ідентифікована і більш-менш формально описана. Як тільки модель визначена, можна з її допомогою інтерпретувати дані. Незважаючи на глибину розуміння

і справедливість теорії, можна екстраполювати ряд на основі знайденої моделі, тобто передбачити його майбутні значення [1, 2].

Як і більшість інших видів аналізу, аналіз часових рядів передбачає, що дані містять систематичну складову (зазвичай включає кілька компонентів) і випадковий шум (помилку), який ускладнює виявлення регулярних компонентів. Більшість методів дослідження часових рядів включає різні способи фільтрації шуму, що дозволяють побачити регулярну складову більш чітко. Більшість регулярних складових часових рядів належить до двох класів: вони є трендом, або сезонною складовою. Тренд є загальним систематичним лінійним або нелінійним компонентом, яка може змінюватися в часі [2]. Сезонна складова – це компонент, що періодично повторюється. Обидва ці види регулярних компонентів часто присутні в ряді одночасно.

На спостереження за показниками та їх систематизацією впливають тенденції та сезонні ефекти. Від умов залежить складність моделювання системи прогнозування. ЧР можна розділити за наявністю чи відсутністю тенденцій та сезонних ефектів на стаціонарні та нестаціонарні.

Ряди з трендом чи сезонністю нестаціонарні [3]. Ряди з неперіодичними циклами стаціонарні, оскільки не можна передбачити заздалегідь, де знаходитимуться максимуми та мінімуми.

У стаціонарних часових рядах статистичні властивості не залежить від часу, тому результат легко передбачити. Більшість статистичних методів припускають, що всі ЧР мають бути стаціонарними. Звичайно, вона залежить від багатьох факторів, але її спад або зростання можна передбачити: у народжуваності немає яскраво вираженої сезонності.

У нестаціонарних часових рядах статистичні властивості змінюються з часом. Вони показують сезонні ефекти, тренди та інші структури, які залежить від тимчасового показника [3]. Приклад – міжнародні перельоти авіакомпаній. Кількість пасажирів у тих чи інших напрямах змінюється залежно від сезонності.

Для класичних статистичних методів зручніше створювати моделі стаціонарних часових рядів. Якщо простежується чітка тенденція чи сезонність у часових рядах, слід змоделювати ці компоненти і видалити з спостережень.

У загальному випадку ЧР (часовий ряд) можна представити з чотирьох різних компонентів:

- сезонної компоненти (позначається S_t, де t означає момент часу);
- тренда (T_t);
- циклічні компоненти (C_t);
- випадкової, нерегулярної компоненти (E_t).

Різниця між циклічною та сезонною компонентою полягає в тому, що остання має регулярну (сезонну) періодичність, тоді як циклічні фактори зазвичай мають триваліший ефект, який, до того ж, змінюється від циклу до циклу. Тренд і циклічну компоненту зазвичай поєднують в одну тренд-циклічну компоненту (T_tC_t) (для простоти позначень далі T_tC_t -> T_t) [4]. Конкретні функціональні взаємозв'язки між цими компонентами можуть мати різний вигляд. Однак можна виділити два основні способи, за допомогою яких вони можуть взаємодіяти – адитивно та мультиплікативно.

Адитивна модель можна описати наступною формулою:

$$Y_t = T_tC_t + S_t + E_t. \quad (1.1)$$

Мультиплікативна модель:

$$Y_t = T_t * C_t * S_t * E_t. \quad (1.2)$$

Модель змішаного типу має наступне рівняння:

$$Y_t = T_t * C_t * S_t + E_t. \quad (1.3)$$

Вибір однієї з трьох моделей складає основі аналізу структури сезонних коливань. Якщо амплітуда коливань приблизно стала, будують адитивну модель ЧР, в якій значення сезонної компоненти передбачаються постійними для різних циклів [4]. Якщо амплітуда сезонних коливань зростає або зменшується, будують мультиплікативну модель ЧР, яка ставить рівні в залежність від значень сезонної компоненти. Побудова адитивної та

мультиплікативної моделей зводиться до розрахунку значень T, S та E для кожного рівня ряду [4, 5]. Процес побудови моделі включає такі кроки:

1. Вирівнювання вихідного ряду методом ковзної середньої.
2. Розрахунок значень сезонної компоненти S.
3. Усунення сезонної компоненти з вихідних рівнів ряду та отримання вирівняних даних (Y - S = T + E) в адитивної або (Y / S = T * E) мультиплікативної моделі.
4. Аналітичне вирівнювання рівнів (T + E) або (T * E) та розрахунок значень T з використанням отриманого рівняння тренду.
5. Розрахунок отриманих за моделлю значень (T + E) або (T * E).
6. Розрахунок абсолютних та/або відносних помилок. Якщо з часового ряду видалити тренд (T_t) та періодичні складові (C_t і S_t), то залишиться нерегулярна компонента (E_t), так звана помилка. Якщо отримані значення помилок не містять автокореляції, ними можна замінити вихідні рівні ряду і надалі використовувати ЧР помилок (E_t) для аналізу взаємозв'язку вихідного ряду та інших часових рядів.

1.1.1 Трендова компонента

Тренд – компонента часового ряду, що повільно змінюється, яка описує вплив на ЧР довготривалих факторів, що викликають плавні та тривалі зміни ряду. Тренди можуть бути описані різними рівняннями – лінійними, логарифмічними, степеневими і так далі.

Тенденції можуть посилюватися, зменшуватися або бути стабільними в різні відрізки часу. Але загальний тренд має бути висхідним, низхідним або стабільним.

1.1.2 Сезонна компонента

Сезонна компонента є складовою часового ряду, що описує регулярні зміни його значень в межах деякого періоду і представляє собою послідовність майже повторюваних циклів.

Багатьом процесам властива повторюваність у часі, причому періодичність таких повторень може змінюватись у дуже широкому діапазоні. Очевидно, що для опису таких періодичних змін, присутніх у ЧР тренд виявляється непридатним.

Сезонна компонента може бути прив'язана до певного календарного тимчасового інтервалу: дню, тижню, місяцю – або до якоїсь події, яка прямо не співвідноситься із конкретними календарними інтервалами. Сезонну компоненту з періодом, що змінюється, іноді називають плаваючою.

1.1.3 Циклічна компонента

Циклічність – періодично коливання, які спостерігаються на часових рядах. У разі дослідження процесів, зав'язаних на календарні зміни, використовують сезонну компоненту як окремий, але важливий випадок циклічності.

Часто ЧР містять зміни, надто плавні та помітні для випадкової складової. У той же час такі зміни не можна віднести ні до тренду, оскільки вони не є досить протяжними, ні до сезонної компоненти, оскільки вони не є регулярними. Подібні зміни називаються циклічною компонентою часового ряду. Циклічна компонента часового ряду – інтервали підйому чи спаду, які мають різну довжину, і навіть різну амплітуду розміщених у яких значень.

Вивчення циклічної компоненти часто виявляється корисним для прогнозування, особливо короткострокового.

1.1.4 Нерегулярна компонента

Нерегулярна компонента, яка відображає складову ЧР, яку не можна пояснити. Зазвичай це складова частина часового ряду, що залишилася після виділення систематичних компонентів. Вона відбиває вплив численних чинників випадкового характеру і є випадкову, нерегулярну компоненту. Вона є обов'язковою складовою будь-якого часового ряду економіки, оскільки випадкові відхилення неминуче супроводжують будь-якому економічному

явищу. Якщо систематичні компоненти часового ряду визначені правильно, то така решта послідовність (ряд залишків), що залишається після виділення з часового ряду цих компонентів, буде випадковою компонентою ряду. За наявності випадкової компоненти неможливо прогнозувати значення часового ряду без помилки.

1.2 Огляд існуючих моделей тренду

Для опису трендів використовуються лінійні та нелінійні рівняння. Вид тренду визначається за допомогою побудови його функціональної моделі статистичними методами або шляхом згладжування вихідного часового ряду.

Усі методи побудови моделей трендів можна поділити на параметричні та непараметричні.

В параметричних моделях вибирається вид функцій (наприклад, лінійна, квадратична, експоненційна) та оцінюються значення їх параметрів (наприклад, за допомогою методу найменших квадратів), за яких вони найкраще відповідають даним [5].

В свою чергу непараметричні моделі застосовуються, коли підібрати адекватну функцію моделі тренду не вдається. У цьому випадку використовують методи згладжування ряду – ковзного середнього або експоненційного згладжування.

Щоб уявити характер тренду, зазвичай досить поглянути графік часового ряду. Найбільш популярні моделі для опису тренду:

- проста лінійна модель;
- поліноміальна модель (у більшості реальних завдань ступінь полінома не перевищує 5);
- експоненційна модель (використовується у випадках, коли процес характеризується рівномірним збільшенням темпів зростання);
- логістична модель.

1.2.1 Лінійний тренд

Найпростішим типом лінії тренду є пряма лінія, описувана лінійним (тобто першого ступеня) рівнянням тренду:

$$\widehat{Y}_t = a + b * t_i, \quad (1.4)$$

де Y_i – вирівняні, тобто позбавлені коливань, рівні тренду для років із номером i;

a – вільний член рівняння, чисельно рівний середньому вирівняному рівню для моменту або періоду часу, прийнятого за початок відліку, тобто для $t = 0$;

b – середня величина зміни рівнів ряду за одиницю зміни часу;

t_i – номери моментів або періодів часу, до яких належать рівні часового ряду (рік, квартал, місяць, дата).

Середня зміна рівнів ряду за одиницю часу – головний параметр та константа прямолінійного тренду. Отже, цей тип тренду підходить для відображення тенденції приблизно рівномірних змін рівнів: рівних у середньому абсолютних приростів або абсолютних скорочень рівнів за рівні проміжки часу [6]. Практика показує, що такий характер динаміки трапляється досить часто.

Причина близьких до рівномірних абсолютних змін рівнів ряду полягає в наступному: багато явищ, як, наприклад, урожайність сільськогосподарських культур, чисельність населення регіону, міста, сума доходу населення, середнє споживання будь-якого продовольчого товару та ін., залежать від великої кількості різних факторів. Одні з них впливають на прискорене зростання досліджуваного явища, інші – на уповільнення зростання, треті – на скорочення рівнів тощо. Вплив різноспрямованих та різноприскорених (уповільнених) сил факторів взаємно усереднюється, частково взаємно погашається, а рівнодіючий їх вплив набуває характеру, наближеного до рівномірної тенденції. Отже, рівномірна тенденція динаміки (або застою) – це результат складання впливу великої кількості факторів на

зміну показника, що вивчається. Графічне зображення прямолінійного тренду – пряма лінія у системі прямокутних координат з лінійним (арифметичним) масштабом обох осях.

Приклад лінійного тренду подано на рис. 1.1.

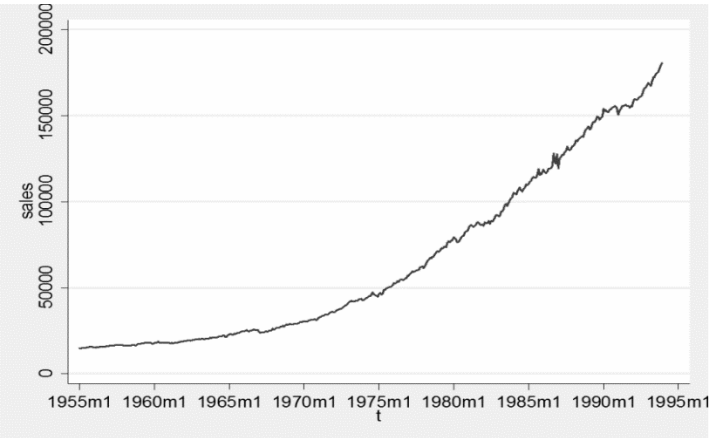

Рисунок 1.1 – Графік моделі лінійного тренду.

Основні властивості тренду у формі прямої лінії такі:

- рівні зміни за рівні проміжки часу;
- якщо середній абсолютний приріст – позитивна величина, то відносні прирости чи темпи приросту поступово зменшуються;
- якщо середня абсолютна зміна – негативна величина, то відносні зміни або темпи скорочення поступово збільшуються за абсолютною величиною зниження до попереднього рівня;
- якщо тенденція до скорочення рівнів, а величина, що вивчається, є за визначенням позитивною, то середня зміна b не може бути більшою за середній рівень a;
- при лінійному тренді прискорення, тобто різниця абсолютних змін за послідовні періоди, дорівнює нулю.

1.2.2 Експоненційний тренд

Експоненційним трендом називають тренд, виражений рівнянням:

$$\widehat{Y}_i = a * k^{t_i} . \quad (1.5)$$

Вільний член експоненти дорівнює вирівняному рівню, тобто. рівнем тренду в останній момент чи період, прийнятий початку відліку часу, тобто. при t = 0. Основний параметр експоненційного тренду k є постійним темпом зміни рівнів (цінним). Якщо k > 1, маємо тренд з зростаючими рівнями, причому це не просто прискорене, і з зростаючим прискоренням і зростаючими похідними всіх вищих порядків. Якщо k < 1, то маємо тренд, що виражає тенденцію постійного, але скорочення рівнів, що сповільнюється, причому уповільнення безперервно посилюється. Екстремуму експонента немає і за t → ∞ прагне або до ∞ при k > 1, або до 0 при k < 1.

Експоненційний тренд характерний для процесів, що розвиваються в середовищі, що не створює жодних обмежень для зростання рівня. З цього випливає, що на практиці він може розвиватися тільки на обмеженому проміжку часу, оскільки будь-яке середовище рано чи пізно створює обмеження, будь-які ресурси з часом вичерпні [6, 7].

Основні властивості експоненційного тренду:

- абсолютні зміни рівнів тренду пропорційні самим рівням;
- експонента екстремумів немає: при k > 1 тренд прагне +∞ , при k<1 тренд прагне нулю;
- рівні тренду є геометричною прогресією: рівень періоду з номером $t = a * k_m$;
- при k > 1 тренд відображає прискорене нерівномірне зростання рівнів, при k < 1 тренд відбиває уповільнене нерівномірне зменшення рівнів;

Графічний приклад експоненційного тренду зображено на рис. 1.2.

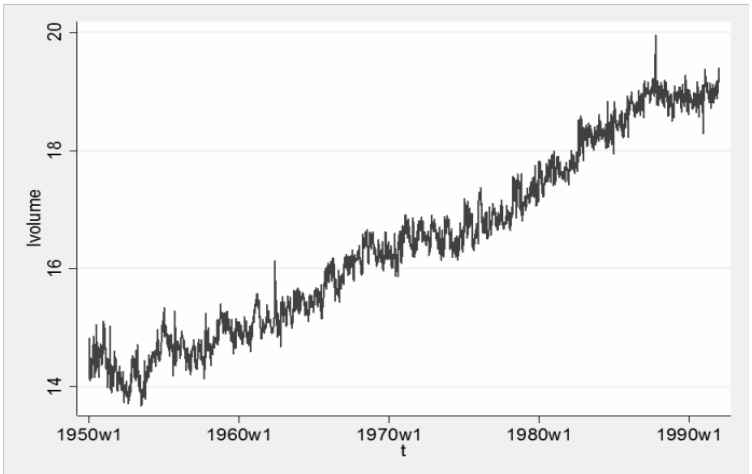

Рисунок 1.2 – Графік моделі експоненційного тренду.

1.2.3 Логарифмічний тренд

Якщо процес, що вивчається, призводить до уповільнення зростання довільного показника, але при цьому зростання не припиняється, не прагне будь-якої обмеженої межі, то гіперболічна форма тренду вже не підходить. Тим більше не підходить парабола з негативним прискоренням, за якою зростання, що сповільнюється, перейде з часом у зниження рівнів. У зазначеному випадку тенденція зміни найкраще відображається логарифмічною формою тренду:

$$\widehat{Y}_t = a + b \ln (t_i). \qquad (1.6)$$

Логарифми зростають значно повільніше, ніж самі числа (номери періодів), але зростання логарифмів необмежене. Підбираючи початок відліку періодів (моментів) часу, можна знайти таку швидкість зниження абсолютних змін, яка найкраще відповідає фактичному часовому ряду [6].

Основні властивості логарифмічного тренду:

- якщо b > 0, то рівні зростають, але з уповільненням, а якщо b < 0, то рівні тренду зменшуються, також із уповільненням;
- абсолютні зміни рівнів за модулем завжди зменшуються з часом;
- прискорення абсолютних змін мають знак, протилежний абсолютним змінам, а по модулю поступово зменшуються;

- темпи зміни (ланцюгові) поступово наближаються до 100% при t → ∞.

Можна зробити загальний висновок у тому, що логарифмічний тренд, як і гіперболічний тренд, має поступово згасаючий процес змін. Відмінність полягає в тому, що загасання по гіперболі відбувається швидко при наближенні до кінцевої межі, а при логарифмічному тренді згасаючий процес триває без обмеження набагато повільніше.

Приклад логарифмічного тренду можна побачити на рис. 1.3.

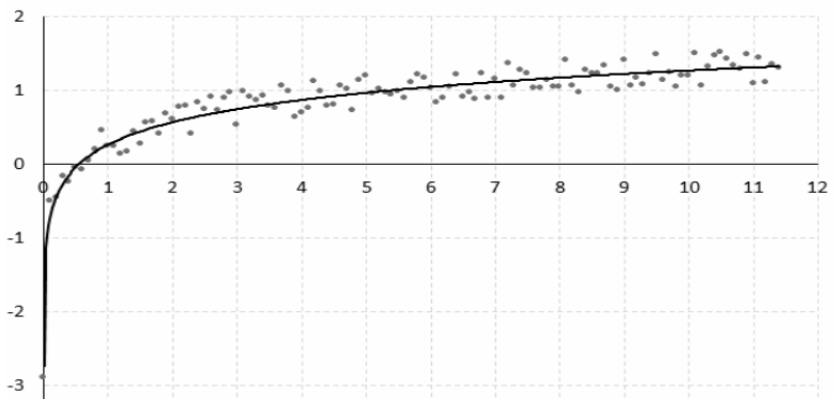

Рисунок 1.3 – Графік моделі логарифмічного тренду.

1.2.4 Логістичний тренд

Логістична форма тренду підходить для опису такого процесу, при якому досліджуваний показник проходить повний цикл розвитку, починаючи, як правило, від нульового рівня, спочатку повільно, але з прискоренням зростаючи, потім прискорення стає нульовим у середині циклу, тобто. зростання відбувається за лінійним трендом, потім, у завершальній частині циклу, зростання уповільнюється по гіперболі в міру наближення до граничного значення показника.

Рівняння логістичного тренду має вигляд:

$$\widehat{Y}_i = \frac{y_{max} - y_{min}}{e^{a_0 + a_1 t_{i+1}}} + y_{min}. \qquad (1.7)$$

При логістичному тренді зі рівнями, що знижуються, показники динаміки змінюються в наступному порядку: негативні абсолютні зміни по

модулю зростають до середини ряду і знижуються до кінця, прагнучи до нуля при $t \to \infty$ [7].

Прискорення у першій половині періоду негативні та за модулем зростаючі; у другій половині періоду прискорення позитивні та зменшуються в межах до нуля. Темпи змін дедалі менше 100%, наприкінці першої половини періоду найменші, у другій половині зростають із уповільненням до 100% у межі.

Приклад логістичного тренду зображено на рис. 1.4.

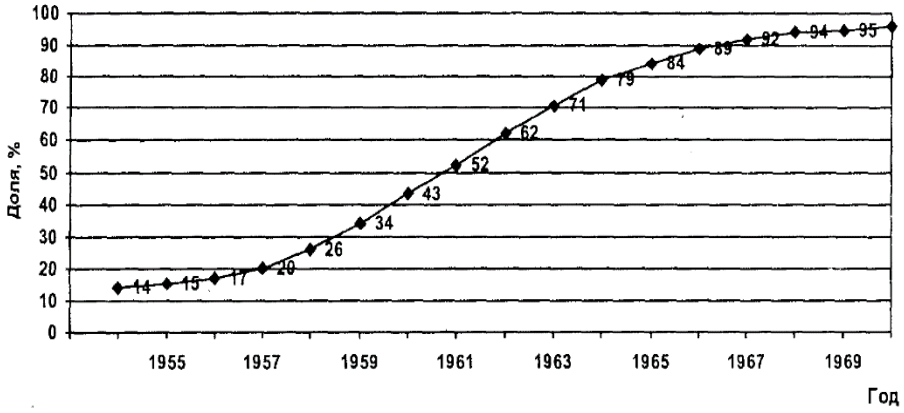

Рисунок 1.4 – Графік логістичної моделі тренду.

1.3 Прогнозування даних в часових рядах

Прогнозування часових рядів – це процес аналізу даних часових рядів із використанням статистики та моделювання для прогнозування та прийняття стратегічних рішень. Це не завжди точне передбачення, і ймовірність прогнозів може різко відрізнятися, особливо коли йдеться про типові коливання змінних у даних часових рядів, а також фактори, які ми не контролюємо. Однак прогнозування розуміння того, які результати є більш імовірними або менш імовірними, ніж інші потенційні результати [8]. Часто, чим повніші дані використовуються для аналізу, тим точнішими можуть бути прогнози. Хоча прогнозування та «передбачення» загалом означають те саме, існує помітна відмінність. У деяких галузях прогнозування може стосуватися

даних на певний майбутній момент часу, тоді як прогнозування стосується майбутніх даних у цілому.

Прогнозування рядів часто використовується в поєднанні з аналізом часових рядів. Аналіз часових рядів включає розробку моделей для розуміння даних, щоб зрозуміти основні фактори впливу.

Моделі зазвичай оцінюються за допомогою середньоквадратичної помилки (MSE) або середньоквадратичної помилки (RMSE) [9].

1.3.1 Аналіз існуючих моделей прогнозування

Авторегресійні моделі

Y_t називається авторегресійним рядом порядку p, AR(p) якщо він задовольняє:

$$Y_t = \varphi_1 Y_{t-1} + \cdots + \varphi_p Y_{t-p} + \varepsilon_t, \tag{1.8}$$

де ε_t – білий шум, а φ_u – коефіцієнти параметрів. Наступне значення, яке спостерігається в ряду, є невеликим збуренням простої функції останніх спостережень.

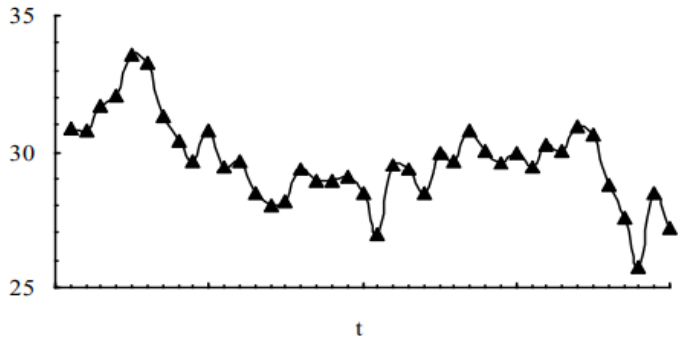

Рисунок 1.5 – Приклад процесу AR(1), створеного за допомогою генератора випадкових чисел.

У випадку k-го порядку кореляція між Y_t і Y_{t-k} частково може бути через кореляцію цих спостережень із інтервенційними лагами Y_{t-1}, Y_{t-2}, … Y_{t-k+1}. Для коригування цієї кореляції розраховуються часткові автокореляції – PACF.

Моделі середнього ковзного

Y_t називається процесом ковзного середнього порядку q, MA(q), якщо він задовольняє:

$$Y_t = \varepsilon_t + \theta_u \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}, \quad (1.9)$$

де θ_u – коефіцієнти параметрів. На практиці це легко відрізнити і серії AR за поведінкою їх ACF: MA ACF різко обривається, тоді як AR ACF спадає експоненціально.

Важливо зауважити, що скінченна AR-модель еквівалентна нескінченній моделі MA, а скінченна MA-модель еквівалентна нескінченній моделі AR [9].

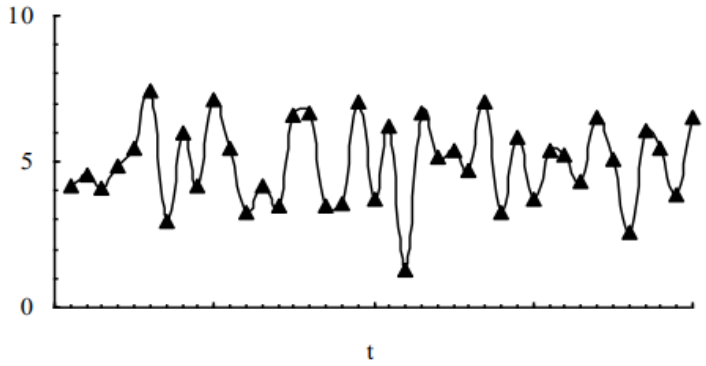

Рисунок 1.6 – Приклад процесу MA(1), створеного за допомогою генератора випадкових чисел.

Інтегровані моделі

Інтегрований ряд – це ряд, у якому значення Y_t є простою сумою випадкових поштовхів. Загалом, порядок інтегрування d можна розглядати як кількість різниць, яку ряд вимагає, щоб стати стаціонарним.

Процес випадкового блукання є прикладом $I(d)$:

$$\Delta Y_t = Y_t - Y_{t-1} = \varepsilon_t \quad (1.10)$$

де різницевий ряд ΔY_t є просто функцією випадкового члена ε_t.

Авторегресійний ряд ковзних середніх

Y_tназивається авторегресійним ковзним середнім процесом порядку (p, q), ARMA(p, q), якщо він задовольняє:

$$Y_t = \varphi_1 Y_{t-1} + \cdots + \varphi_p Y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}. \quad (1.11)$$

Інтегрований авторегресійний ряд ковзних середніх

Якщо $W_t = \Delta^d Y_t$, замість Y_t в ARMA(p, q), то кажуть, що це інтегрований авторегресійний ряд ковзного середнього ARIMA(p, d, q).

Багато часових рядів демонструють сильні сезонні характеристики. Сезонні моделі можуть проявлятися в багатьох контекстах (наприклад, тижневі моделі в щоденних спостереженнях або щоденні моделі в погодинних даних) [8, 9].

Сезонні ефекти можна моделювати шляхом включення коефіцієнтів при лагах, кратних сезонному періоду.

Щоб визначити відповідну ARIMA модель для часового ряду, необхідно визначити порядок (порядки) розбіжностей, необхідних для стаціонарізації ряду та усунення грубих ознак сезонності. Якщо результуючий ЧР показує сильну тенденцію (зростання або спад), то процес явно не є стаціонарним, і його слід хоча б раз розрізнити (differencing).

Другим тестом, який можна використати, є перевірка оціненої автокореляції часових рядів. Для стаціонарного часового ряду автокореляції зазвичай швидко спадають до 0. Для нестаціонарних часових рядів автокореляції зазвичай спадають повільно, якщо взагалі спадають.

1.3.2 Прогнозування з використанням трендової компоненти часового ряду

Модель авторегресії та ковзного середнього, запропонована Боксом і Дженкінсом включає як параметри авторегресії, так і параметри ковзного середнього. Саме, є три типи параметрів моделі: параметри авторегресії (p), порядок різниці (d), параметри ковзного середнього (q). В позначеннях Бокса

і Дженкінса модель записується як ARIMA (p, d, q). Наприклад, модель (0, 1, 2) містить 0 (нуль) параметрів авторегресії (p) і 2 параметри ковзного середнього (q), які обчислюються для ряду після взяття різниці з лагом, що дорівнює 1.

Як зазначено раніше, для моделі ARIMA необхідно, щоб ряд був стаціонарним, це означає, що його середнє постійно, а вибіркова дисперсія та автокореляція не змінюються у часі [9]. Тому зазвичай необхідно брати різниці ряду доти, доки він не стане стаціонарним (часто також застосовують логарифмічне перетворення для стабілізації дисперсії). Число різниць, які були взяті для досягнення стаціонарності, визначаються параметром d (див. попередній розділ). Для того, щоб визначити необхідний порядок різниці, потрібно дослідити графік ряду та автокорелограму. Сильні зміни рівня (сильні стрибки вгору чи вниз) зазвичай вимагають взяття несезонної різниці першого порядку (лаг дорівнює 1). Сильні зміни нахилу вимагають взяття різниці другого порядку. Сезонна складова вимагає взяття відповідної сезонної різниці (див. нижче). Якщо є повільне зменшення вибіркових коефіцієнтів автокореляції залежно від лага, зазвичай беруть різницю першого порядку. Однак слід пам'ятати, що для деяких часових рядів потрібно брати різниці невеликого порядку або зовсім не брати їх. Зауважимо, що надмірна кількість взятих різниць призводить до менш стабільних оцінок коефіцієнтів.

Методологія Бокса-Дженкінса підбору ARIMA-моделі для певного ряду спостережень складається з чотирьох етапів:

- ідентифікація моделі – процес вибору моделі, що найкраще відповідає аналізованому реальному процесу;

- оцінювання моделі – використання регресійних методів для отримання оцінок параметрів, включених в модель;

- тестування моделі – перевірка основних передумов використання регресійного аналізу, перевірка адекватності моделі з використанням тестів на нормальність залишків (тест Жарка-Бера), на автокореляцію залишків (тест

Дарбіна Вотсона), на сталість дисперсій випадкових залишків (критерії Кохрана та Голдфалда Кванта) якість специфікації моделі (F-тест);

- використання моделі для прогнозування.

На етапі ідентифікацією порядку моделі необхідно вирішити, як багато параметрів авторегресії (p) і ковзного середнього (q) має бути присутнім в ефективній та економній моделі процесу (економність моделі означає, що в ній є найменша кількість параметрів та найбільша кількість ступенів свободи серед усіх моделей, що підганяються до даних). Насправді дуже рідко буває, що число параметрів p чи q більше 2.

Наступний, після ідентифікації, крок полягає в оцінюванні параметрів моделі. Отримані оцінки параметрів використовуються на останньому етапі, щоб обчислити нові значення ряду і побудувати довірчий інтервал для прогнозу. Процес оцінювання проводиться за перетвореними даними (підданим застосуванню оператора різниці). До побудови прогнозу необхідно виконати зворотну операцію (інтегрувати дані). Таким чином, прогноз методології порівнюватиметься з відповідними вихідними даними. На інтегрування даних вказує буква “I” у загальній назві моделі (Autoregressive Integrated Moving Average).

Додатково моделі ARIMA можуть містити константу, інтерпретація якої залежить від моделі, що підганяється. Саме, якщо у моделі немає параметрів авторегресії, то константа є середнє значення ряду, якщо параметри авторегресії є, то константа є вільним членом. Якщо бралася різниця ряду, то константа є середнім або вільним членом перетвореного ряду. Наприклад, якщо бралася перша різниця (різниця першого порядку), а параметрів авторегресії в моделі немає, то константа є середнім значенням перетвореного ряду i, отже, коефіцієнт нахилу лінійного тренда вихідного.

Висновки за розділом 1

Після огляду на аналіз часових рядів та їх особливостей можна сказати, що ЧР – це хронологічна послідовність спостережень за певною змінною. Часові ряди мають складові компоненти, які дозволяють отримати цілковито повний опис даних та правильно підібрати модель для подальшого прогнозування. Зазвичай спостереження проводяться через рівні проміжки часу (дні, місяці, роки), але вибірка може бути нерегулярною. Аналіз часових рядів складається з трьох етапів:

1) створення моделі, яка представляє ЧР;
2) перевірка запропонованої моделі;
3) використання моделі для передбачення (прогнозування) майбутніх значень та/або врахування відсутніх значень.

Якщо ЧР має регулярний шаблон, тоді значення ряду має бути функцією попередніх значень.

Мета побудови моделі часових рядів така ж, як і для інших типів прогнозних моделей, тобто створити модель так, щоб помилка між прогнозованим значенням цільової змінної та фактичним значенням була якомога меншою. Основна відмінність між моделями часових рядів та іншими типами моделей полягає в тому, що значення лагу цільової змінної використовуються як предикторні змінні, тоді як традиційні моделі використовують інші змінні як предиктори, і концепція значення лагу не застосовується, оскільки спостереження не не представляють хронологічної послідовності.

Першим кроком при аналізі часового ряду для розробки прогностичної моделі є виявлення та розуміння закономірностей, що лежать в основі даних з часом. Ці основні закономірності зазвичай класифікуються як чотири компоненти: загальний тренд, сезонність, циклічні коливання і випадкова складова (помилка чи шум).

Також слід зазначити, що прогнозування часових рядів — це завдання адаптації моделі до історичних даних із мітками часу, щоб передбачити майбутні значення. Традиційні підходи включають ковзне середнє, експоненціальне згладжування та ARIMA.

Виділення тренду мережевого трафіку є дуже важким і водночас дуже важливим завданням, оскільки його рішення дозволяє здійснювати прогноз, у своїй випадкова складова часового ряду використовується з метою оцінки точності прогнозу стану трафіку.

Використання сучасних методів аналізу та оцінки трафіку відіграють велику роль у розвитку комп’ютерних мереж, тож необхідне постійне вдосконалення існуючих методів та впровадження нових.

РОЗДІЛ 2
МЕТОД ОЦІНКИ ТРЕНДУ ЧАСОВОГО РЯДУ

Виділення тренду мережевого трафіку є дуже важким і водночас дуже важливим завданням, оскільки його вирішення дозволяє здійснювати прогноз, у якому випадкова складова часового ряду використовується з метою оцінки точності прогнозу стану трафіку.

В даний час відомі два методи виділення тренду. Перший метод полягає в тому, що за емпіричними даними ЧР підбирається крива (математична модель), що вирівнює, яка з найбільшою точністю описує ЧР. При цьому в якості математичних моделей використовуються різні функції: рівняння прямої та експоненти, парабола (квадратична, кубічна і більш високих ступенів), логістична крива, крива Гомперца та ін. Другий метод виділення тренду полягає в згладжуванні ряду за методом ковзної середньої [10]. При цьому зазвичай знаходять середнє значення трьох (або п’яти) перших членів, далі беруться наступні три члени зі зміщенням на одиницю і середнє. Таким чином, вдається зменшити випадкову складову.

2.1 Критерії наявності тренду

При аналізі часового ряду є доцільним виявити трендову складову. Для аналізу вибірок випадкових величин (часових рядів) на предмет відсутності тренду в характеристиках вимірюваної величини в додатках використовується цілий ряд параметричних та непараметричних критеріїв перевірки гіпотез [11].

Серед критеріїв виявлення трендової компоненти популярності набули наступні:

- критерій Аббе-Лінника;
- критерій Кокса-Стюарта;
- критерій Фостера-Стюарта.

Кожен з цих критеріїв використовується індивідуально для кожного часового ряду, в залежності від властивостей даних, на яких цей критерій буде застосовуватись.

Критерії наявності тренду та випадковості призначені для перевірки гіпотез про випадковість розташування отриманих вибіркових даних, тобто відсутності взаємозв'язку між значеннями спостережуваної випадкової величини та їх номерами у вибірковій послідовності.

Найчастіше критерії тренду знаходять застосування при статистичному контролі і запобіжному регулюванні технологічних процесів у промисловості, дозволяючи заздалегідь статистично обґрунтовано виявити тенденцію, що намічається, погіршення якості продукції [11]. Також дана група критеріїв має велике практичного значення для медицини. Наявність тренду в досліджуваному ряду даних про захворювання є об'єктивним критерієм оцінки епідемії, що насувається, і темпів її зростання. Кількість можливих ситуацій, у яких виявлення тренду дає практично корисну інформацію, надзвичайно велике, і кожен інженер чи дослідник повсякденно зустрічається з необхідністю використовувати критерії тренду та випадковості у своїй роботі.

Слід зазначити, що однією з основних передумов застосування багатьох критеріїв даного класу є припущення про належність аналізованих даних до нормального закону [11, 12]. Проте, практично таке припущення досить часто порушується. Для багатьох статистичних критеріїв порушення припущення нормальності призводить до істотних змін у законі розподілу статистики критерію.

Критерій Аббе-Лінніка призначений для перевірки гіпотези про те, що всі вибіркові значення належать до однієї генеральної сукупності з постійним середнім проти альтернативи тренду.

Нехай $x_1, \dots, x_n$ – ряд значень взаємно незалежних нормально розподілених випадкових величин з математичними очікуваннями $\mu_1, \dots, \mu_n$ відповідно і однаковими (але невідомими) дисперсіями. Перевіряється

гіпотеза у тому, що це вибіркові значення належать однієї генеральної сукупності із середнім μ:

$$H_0: \mu_i = \mu, i = 1, \dots, n, \quad (2.1)$$

проти альтернативи тренду:

$$H_1: |\mu_{i+1} - \mu_i| > 0,\ i = 1, \dots, n-1. \quad (2.2)$$

Статистика критерію Аббе-Лінніка має вигляд:

$$q = \frac{1}{2}\frac{\sum_{i=1}^{n-1}(x_{i+1}-x_i)^2}{\sum_{i=1}^{n}(x_i-\bar{x})}, \text{ де } \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i. \quad (2.3)$$

Якщо $q > q_\alpha$, то нульова гіпотеза випадковості ряду $x_1, \dots, x_n$ відхиляється з довірчою ймовірністю α.

При n > 60 справлива апроксимація, заснована на тому, що випадкова величина

$$Q^* = -(1-q)\sqrt{\frac{2n+1}{2-(1-q)^2}} \quad (2.4)$$

має стандартний нормальний розподіл. Тому нульова гіпотеза відхиляється, якщо $Q^* < u_{1-\alpha}$.

Критерій Кокса-Стюарта призначений для перевірки тренда середніх та дисперсій у послідовності спостережень. Критерій передбачає наявність гіпотези H0 – існування тренду.

Для критерію середнього у вибірці обсягу n запропоновано статистику

$$S_1 = \sum_{i=1}^{\left[\frac{n}{2}\right]}(n-2i+1)h_{i,n-i+1}, \quad (2.5)$$

де

$$h_{i,j} = \begin{cases} 0, & x_i > x_j \\ 1, & x_i \le x_j \end{cases}, (i < j). \quad (2.6)$$

Критерій, заснований на статистиці S_1, має ефективність приблизно 0.86 по відношенню до найкращого параметричного критерію. Для перевірки гіпотези тренду застосовується нормалізована статистика:

$$S_1^* = \frac{S_1 - M(S_1)}{\sqrt{D(S_1)}}, \quad (2.7)$$

де

$$\mathrm{M}(\mathrm{S}_1) = \frac{\mathrm{n}^2}{8}, \tag{2.8}$$

$$\mathrm{D}(\mathrm{S}_1) = \frac{\mathrm{n}(\mathrm{n}^2-1)}{24}. \tag{2.9}$$

При $|S_1^*| < u_{\frac{1+\alpha}{2}}$ нульова гіпотеза H_0 існування тренда середнього відхиляється, інакше гіпотеза H_0 приймається.

Критерій для перевірки гіпотези про тренд дисперсії у вибірці будується в такий спосіб. Вибірка $x_1, \dots, x_n$ розбивається на n/k підвибірок $x_1, \dots, x_k;\ x_{k+1}, \dots, x_{2k};\ x_{2k+1}, \dots, x_{3k}; \dots; x_{n-k+1}, \dots, x_n$ (якщо n не ділиться на k відкидається необхідну кількість спостережень у центрі). Для кожної i-тої підвиборки знаходиться розмах ω_i $(1 \le i \le r)$, де $r = \left[\frac{n}{k}\right]$. Далі розмахи ω_i перевіряються на тренд критерієм S_1.

Критерій Фостера-Стюарта призначений для перевірки тренду середніх і дисперсій. Метод Фостера-Стюарта крім визначення наявності тренду в часовому ряду дозволяє виявити основну тенденцію дисперсії рівнів низки динаміки, що важливо знати під час аналізу та прогнозування його майбутніх значень [13].

Нульова гіпотеза H_0 – існування тренду.

Статистики критерію мають вигляд:

$$S = \sum_{i=2}^{n} S_i,$$
$$d = \sum_{i=2}^{n} d_i, \tag{2.10}$$

де

$$S_i = u_i + l_i,$$
$$d_i = u_i - l_i. \tag{2.11}$$

- якщо $x_i > x_{i-1}, \dots, x_1$, тоді $u_i = 1$, інакше $u_i = 0$;
- якщо $x_i < x_{i-1}, \dots, x_1$, тоді $l_i = 1$, інакше $l_i = 1$.

Статистика S використовується для перевірки тренда в дисперсіях, статистика d – для виявлення тренда у середніх:

$$0 \le S \le n - 1,$$

та

$$-(n-1) \le d \le n-1.$$

За відсутності тренду величини дані мають розподіл Стьюдента з ступенями свободи:

$$t = \frac{d}{f}, \tag{2.12}$$

та

$$\tilde{t} = \frac{S-f^2}{l}, \tag{2.13}$$

де

$$l = \sqrt{2\ln(n) - 3.4253}, \tag{2.14}$$

$$f = \sqrt{2\ln(n) - 0.8456}. \tag{2.15}$$

Якщо $|t|, |\tilde{t}| > t_{\frac{1+\alpha}{2}}$, то з довірчою ймовірністю α нульова гіпотеза H_0 існування тренду приймається, інакше гіпотеза H_0 відкидається.

2.2 Метод оцінки тренду

Наразі існує два основні види методів оцінки тренду: параметричні та непараметричні.

Параметричні – розглядають ЧР як гладку функцію від t: $X_t = f(t, \theta)$, $t = 1, \dots, n$ потім різними методами оцінюються параметри функції theta, наприклад, методом найменших квадратів. Виділяють лінеаризувані тренди, тобто приведені до лінійного вигляду щодо параметрів тренду на основі тих чи інших перетворень алгебри.

Серед параметричних тестів, які ґрунтуються на припущеннях нормального розподілу та незалежності, є тести на зміну кроку:

- t-критерій Стьюдента (стандартний параметричний тест для перевірки того, чи дві вибірки мають різні середні значення, і передбачається відомий час точки зміни);
- тест співвідношення правдоподібності Уорслі (схожий на t-критерій Стьюдента, але підходить для використання, коли час точки зміни невідомий);

- лінійна регресія, або тест на поступову тенденцію (один із найпоширеніших тестів на тенденцію — він використовує градієнт регресії як тестову статистику та припускає, що дані розподілені нормально).

Непараметричні – це різного роду ковзні середні (проста, зважена) [14]. Такі методи застосовуються з метою оцінки тренда, але з прогнозуванням. Такі методи бувають корисний у разі, коли для оцінки тренда не вдається підібрати потрібну функцію. Припустимо, що основний процес — неповністю вивчена фізична система. Можна побудувати модель незалежно від природи процесу, щоб пояснити поведінку показників. Зокрема, можна дізнатися, зростає чи зменшується тенденція показників.

Параметричні тести базуються на припущеннях розподілу та незалежності [14]. Однак більшість гідрологічних рядів мають ненормальний розподіл, і тому має сенс використовувати методи тестування без розподілу. Методи без розповсюдження — це методи, у яких не потрібно робити жодних припущень щодо основного розподілу даних. Однак припущення про незалежність все ще залишається. Передбачається, що елементи вибірки є незалежними та однаково розподіленими випадковими змінними.

Методами виявлення тренду є:

- метод укрупнення інтервалів;
- метод ковзної середньої;
- аналітичне вирівнювання.

Метод укрупнення інтервалів ґрунтується на укрупненні періодів часу, до якого належить рівні ряду. При підсумовуванні рівнів за укрупненими інтервалами коливання у рівнях, обумовлені випадковими причинами, взаємопогашуються і чіткіше виявляється загальна тенденція.

Метод ковзної середньої полягає в тому, що розглядається середній рівень із певної кількості рівнянь, але починаючи з другого за рахунком і т.д. Таким чином, середня хіба що «ковзає» з низки динаміки, просуваючись однією термін.

Метод аналітичного вирівнювання: фактичні рівні ряду замінюються рівними рівнями, що плавно змінюються, отриманими з рівняння регресії $Y_t = f(t) + e$. При аналітичному вирівнюванні використовуються різні види трендових моделей.

2.2.1 Аналіз існуючих методів оцінки тренду

Після того, як встановлено наявність тенденції в часовому ряду, необхідно її описати, тобто визначити тип перебігу процесу, що має місце в даному явищі, напрям зростання і зміна, що проходять у ньому [15].

Можна виділити такі типи процесів:

- за зростанням або зменшенням рівнів ряду: монотонно-зростаючі, монотонно-зменшувальні, комбіновані.
- за наявністю насичення та прагнення до деякої граничної величини: що мають межі насичення, що не мають меж насичення.
- за наявністю екстремальних значень та перегинів: процеси, що мають екстремальні значення; процеси, що мають переходи від зростання до спадання та навпаки.

Для виділення типу розвитку можуть використовуватися різні методи та критерії, зокрема відомі способи згладжування:

- згладжування або механічне вирівнювання окремих членів низки динаміки з використанням фактичних значень сусідніх рівнів;
- вирівнювання із застосуванням кривої, проведеної між конкретними рівнями таким чином, щоб вона відображала тенденцію, властиву ряду, і одночасно звільнила його від незначних коливань.

Вибір методу виявлення основний тенденції розвитку залежить від завдань, що стоять перед дослідженням. Якщо треба дати загальну картину розвитку, його грубу модель, засновану на механічному повторенні тих самих дій зі збільшення інтервалу часу, то можна обмежитися методом ковзної середньої. Якщо ж дослідження вимагає докладного аналітичного вираження руху в часі, то методу ковзної середньої недостатньо.

Методи виявлення основної тенденції розвитку мають різний логічний зміст і тому застосовуються до часових ряду для різних цілей. Основна їх мета, як уже говорилося, полягає в тому, щоб розкривати загальні закономірності розвитку, затушовані окремими, іноді випадковими обставинами.

Припустимо, що дані складаються зі спостережень за послідовністю взаємно незалежних випадкових величин $X_1, X_2, \dots, X_n$, розташованих у порядку, в якому спостерігаються випадкові величини. Шкала вимірювання X, є принаймні порядковою. Значки X або розподілені однаково, або існує тенденція.

Тест Кокса-Стюарта можна використовувати для виявлення будь-якого конкретного типу невипадкової моделі, такої як синусоїда або інша періодична картина [16]. Ідея тесту Кокса-Стюарта заснована на порівнянні першої та другої половини вибірки. Якщо є тенденція до зниження, спостереження у другій половині вибірки мають бути меншими, ніж у першій половині. Якщо вони більші, то є підозра на висхідну тенденцію. Якщо тенденції немає, слід очікувати лише невеликих відмінностей між першою та другою половиною вибірки через випадковість.

Таким чином, для проведення аналізу трендів необхідно обчислити вибірку відмінностей:

$$y_1 = x_{1+c} - x_1, y_2 = x_{2+c} - x_2, \dots, y_c = x_n - x_{n*c}, \tag{2.16}$$

де якщо n парне $c = n/2$, а якщо n непарне $c = (n+1)/2$. Різниці, що дорівнюють нулю, не враховуються. Для простоти позначимо вибірку позитивних різниць за $y_1, \dots, y_m$.

Тест Кокса-Стюарта – це тест знака, застосований до вибірки ненульових різниць $y_1, \dots, y_m$. Нехай $sgn(\alpha) = 1$, якщо $\alpha > 0$, і $sgn(\alpha) = -1$, якщо $\alpha < 0$. Статистика тесту Кокса-Стюарта має вигляд:

$$T = \sum_{i=1}^{m} sgn(y_i) \tag{2.17}$$

Правило прийняття рішення: на рівні значущості a треба відхилити гіпотезу H_0 і прийняти альтернативну гіпотезу H, якщо $T > t(\alpha)$ (тенденція до

зростання), якщо $T < t(\alpha)$ (тенденція до зменшення), де $t(\alpha)$ є належним квантилем біноміального розподілу. Для $m > 20$ — наближення

$$t(\alpha) = \frac{1}{2}[m + w(\alpha)]\sqrt{m} \qquad (2.18)$$

де $w(\alpha)$ є α-квантиль стандартного нормального розподілу, який можна застосувати.

Статистичний тест Манна-Кендалла (МК) на тенденцію використовується для оцінки того, чи набір значень даних збільшується з часом чи зменшується з часом, і чи є тенденція в будь-якому напрямку статистично значущою. МК не оцінює величину зміни [16].

Тест можна використовувати для індикаторів із різними одиницями вимірювання та періодами часу та не потребує довірчих інтервалів (які доступні не для всіх індикаторів). Крім того, перевірку можна виконати, навіть якщо в наборі відсутні значення.

Щоб обчислити тестову статистику S, треба порівнювати кожне значення з усіма наступними значеннями періоду часу для індикатора. Для кожної пари порівняння присвоюється оцінка «+1», якщо останнє значення перевищує перше.

Якщо останнє значення нижче за перше, то порівнянню присвоюється оцінка «-1». Потім усі бали підсумовуються для обчислення тестової статистики S. Позитивне значення S означає, що тенденція зростає, а від'ємне значення S означає, що тенденція зменшується.

Тест МК перевіряє, чи слід відхилити нульову гіпотезу (H_0) і прийняти альтернативну гіпотезу (H_a), де

- H_0: немає монотонної тенденції;
- H_a: монотонна тенденція присутня.

Початкове припущення тесту МК полягає в тому, що дані є правдивими та що дані мають бути переконливими поза розумним сумнівом, перш ніж вони будуть відхилені та прийняті [16, 17]. Регресійний аналіз вимагає, щоб залишки від підігнаної лінії регресії були нормально розподілені, припущення,

яке не вимагається тестом Манн-Кендалла, тобто є непараметричним (без розподілу) тестом.

Обчислення S статистики має наступний вигляд:

$$S = \sum_{k=1}^{N-1} \sum_{l=k+1}^{N} sign(x_l - x_k). \quad (2.19)$$

Далі обчислюється $VAR(S)$:

$$VAR(S) = \frac{1}{18}\Big(n(n-1)(2n+5) - \sum_{p=1}^{g} t_p\big(t_p-1\big)\big(2t_p+5\big) -$$

$$- \sum_{q=1}^{h} u_q\big(u_q-1\big)\big(2u_q+5\big) + \frac{\sum_{p=1}^{g} t_p(t_p-1)(2t_p-2) - \sum_{q=1}^{h} u_q(u_q-1)(2u_q-2)}{9n(n-1)(n-2)} +$$

$$+ \frac{\sum_{p=1}^{g} t_p(t_p-1) \sum_{q=1}^{h} u_q(u_q-1)}{2n(n-1)}\Big) \quad (2.20)$$

де g – кількість груп зв’язаних даних;

t_p – кількість зв'язаних даних у p-й групі;

h – кількість разів вибірки, які містять кілька даних;

u_p – кількість множинних даних за q-й період часу.

Z – статистика має наступне рівняння:

$$Z = \begin{cases} \frac{S-1}{\sqrt{VAR(S)}}, & \text{якщо } S > 0, \\ 0, & \text{якщо } S = 0, \\ \frac{S+1}{\sqrt{VAR(S)}}, & \text{якщо } S < 0. \end{cases} \quad (2.21)$$

Тест Лапласа спочатку був розроблений як параметричний тест для певного неоднорідного пуассонівського процесу, що підпорядковується функції інтенсивності:

$$\lambda(t|\alpha,\beta) = e^{\alpha+\beta_t}, \quad (2.22)$$

де α і β є дійсними параметрами.

Якщо $\beta > 0$, то інтенсивність зростає ($\beta < 0$ - зменшується), що означає, що існує тенденція.

Статистику тесту Лапласа, що відповідає наведеній вище функції інтенсивності, можна визначити для двох випадків: передбачається, що спостереження за процесом припиняються або в момент часу t, або в момент

часу T_n, де відбувається n-та точка [18]. Якщо спостереження припинено в момент часу t, то тестова статистика має вигляд:

$$U_\tau = \frac{S_n - \frac{1}{2}n\tau}{\sqrt{\frac{n\tau^2}{12}}}, \quad (2.23)$$

де

$$S_n = \sum_{i=1}^{n} T_i. \quad (2.24)$$

Якщо спостереження зупинено на події T_n, статистика визначається за допомогою наступної формули:

$$U_n = \frac{S_{n-1} - \frac{1}{2}(n-1)T_n}{\sqrt{\frac{(n-1)T_n^2}{12}}}. \quad (2.25)$$

На практиці наведені вище версії тестової статистики істотно не відрізняються. Для великих значень n тестова статистика розподілена приблизно нормально, і відповідно визначаються критичні значення статистики. Наближення є адекватним на 5% рівні значущості для $n > 3$.

Статистика U використовується з наступними гіпотезами:

- відсутність тенденції проти зниження інтенсивності: відхилити нульову гіпотезу $\beta > 0$ на рівні значущості γ, якщо $u < l_\gamma$;
- відсутність тенденції проти зростання інтенсивності: відхилити нульову гіпотезу на рівні значущості γ, якщо $u > m_\gamma$;
- відсутність тенденції проти зниження інтенсивності або зростання інтенсивності: відхилити нульову гіпотезу $\beta = 0$ на рівні значущості γ, якщо $|u| > n_\gamma$.

Для γ = 5 % критичними значеннями є l_γ = -1,645, m_γ= 1,645, n_γ = 1,960.

Загальні моделі точкових процесів, які обговорюються в цьому розділі, призначені для опису явищ, які відбуваються точково в часі [19]. Моделі точкових процесів також можна використовувати для опису деяких явищ, пов’язаних із безперервними процесами, наприклад явища перетину рівня кумулятивних процесів.

Архетипом моделей точкових процесів є однорідний процес Пуассона, в якому події, як передбачається, відбуваються з постійною інтенсивністю. Як обговорювалося вище, модель процесу Пуассона можна легко узагальнити. Тестування трендів точкових процесів зазвичай розглядається як перевірка гіпотез щодо інтенсивності процесу [20]. Коли функція інтенсивності параметризована, тестування тенденції є типовою проблемою параметричного тестування.

Параметричні тести можуть бути розроблені для різних функцій інтенсивності. Критерій Лапласа, розглянутий вище, спочатку є параметричним тестом для певної функції інтенсивності. На практиці тест Лапласа часто використовується як непараметричний тест тренду. Існують моделі, для яких тест має досить хороші статистичні властивості. Крім того, різниця між варіантами різних правил зупинки спостережень практично незначна. З практичної точки зору, тест Лапласа є хорошим показником існування тенденції.

Разом із графічними представленнями статистика трендів Лапласа дає досить чітке уявлення про можливі тренди. Для аналізу точкових явищ практично рекомендується використання статистики Лапласа. Однак, якщо явище не описується належним чином точковою моделлю процесу, не можна використовувати критерій Лапласа. На практиці не завжди фактично використовуються дуже низькі значення рівня статистичної значущості, але тестова статистика скоріше буде застосована для вказівки на можливу тенденцію або для порівняння тенденцій. Оскільки статистика Лапласа має також теоретичні основи, вона дуже корисна для такого роду використання.

2.2.2 Розробка методу оцінки тренду

Для оцінки наявності трендової компоненти в роботі було використано метод Манн-Кендала. Цей метод був взятий за основу оскільки має декілька переваг перед іншими, а саме:

- він не передбачає розподіл даних відповідно до якогось конкретного правила, тобто, наприклад, не вимагає, щоб дані розподілялися нормально;
- на результат не впливають відсутні дані, окрім факту, що кількість точок вибірки зменшено і, отже, може негативно вплинути на статистичну значущість;
- на результат не впливає нерівномірний розподіл часових точок вимірювання;
- на результат не впливає довжина часового ряду.

Окрім тесту Манн-Кендалла використовується також його модифікація, запропонована Хамедом та Рао. На основі модифікованого значення дисперсії статистики тренду Манна-Кендалла використовується модифікований непараметричний тест тренду, який підходить для автокорельованих даних. Точність модифікованого тесту з точки зору рівня його емпіричної значущості є вищою, ніж оригінальний тест тренду Манна-Кендалла без будь-якої втрати потужності.

Для перевірки гіпотези про віднесення ЧР до класу стаціонарних (щодо лінійного тренду) чи нестаціонарних процесів є низка різних тестів. Однак всі тести мають деякі недоліки або обмеження: наприклад, часто не відкидається вихідна (нульова) гіпотеза, коли вона насправді не виконується; або буває зміщена статистика тесту, може навіть відкидатися нульова гіпотеза, коли вона насправді вірна. Для моделі часових рядів усі залежні змінні, незалежні змінні та залишки мають бути перевірені на стаціонарність за допомогою тесту ADF [21].

Розширений тест Дікі-Фуллера (ADF) є модифікацією тесту Дікі-Фуллера в тих випадках, коли передбачається автокорелювання відхилень моделі.

Результати роботи етапу перевірки даних на стаціонарність можна побачити на рис. 2.1.

```
ADF Test Statistic : -3.540124708284948
p-value : 0.007014457876836656
#Lags Used : 0
Number of Observations : 47
Strong evidence against the null hypothesis(Ho), reject the null hypothesis. Data is stationary!
```

Рисунок 2.1 – Вихідні дані етапу перевірки датасету на стаціонарність.

Відхилення нульової гіпотези означає, що процес не має одиничного кореня і, у свою чергу, що ЧР є стаціонарним або не залежить від часу структури.

Загальна оцінка наявності тренду використовує методику консенсусу, тобто складається на основі як мінімум трьох тестів: тест Манн-Кендала, модифікований тест Манн-Кендалла (Yue and Wang) та модифікований тест Манн-Кендала (Hamed and Rao). Результат методу інтерпретується як ціле число в інтервалі [-1;1]. Якщо результат є від'ємне число - то тренд існує, та є низхідним. В випадку, коли результат 0 - це свідчить про те, що тренду не існує. Якщо результат дорівнює 1 - це говорить про те, що в даних є тренд, який має висхідний напрямок. Таким чином можна зробити комплексну оцінку наявності тренду, та отримати об’єктивні результати аналізу.

Схему роботи методу можна побачити на рис. 2.2.

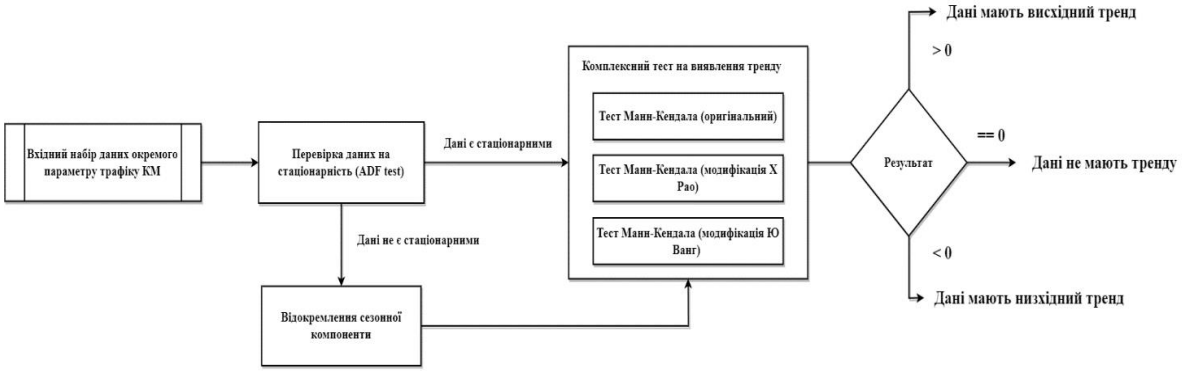

Рисунок 2.2 – Схема роботи розробленого методу виявлення тренду.

Для імплементації розробленого методу було використано мову програмування Python. Завдяки підключенню додаткової зовнішньої

бібліотеки pymannkendall, вдалося досить швидко реалізувати метод виявлення тренду у часовому ряді.

Висновки за розділом 2

В даному розділі було проведено аналіз методів оцінки трендів та критеріїв наявності трендів у часових рядах. А також розроблено новий метод оцінки наявності тренду у часовому ряді.

Графічні методи необхідні, щоб надати дослідницьке уявлення про спостереження та визначити потребу в більш глибокому аналізі тенденцій. Часто графічних методів достатньо, щоб показати, що тенденції немає. Проте графічних інструментів недостатньо для оцінки статистичної значущості та для створення статистичних висновків щодо тенденційних явищ. Тому необхідно застосовувати формальні статистичні тести або моделі. З іншого боку, хоча графічні методи можуть вказувати на тенденцію, розмір вибірки статистичних спостережень може бути настільки малим, що статистично неможливо прийняти жодну гіпотезу щодо тенденції.

Після огляду існуючих критеріїв виявлення тренду можна констатувати, що застосування параметричних критеріїв виявлення тренду в середніх (критерій автокореляції, Аббе) буде коректним і в тих випадках, коли ми маємо справу із законом, що істотно відрізняється від нормального. Однак параметричні критерії виявлення тренду в середніх лише небагатьом перевищують потужності непараметричні. Параметричні критерії виявлення зсуву в дисперсії потужніші за непараметричні, але дуже чутливі до порушення припущень про нормальність випадкових величин (як і будь-які критерії, пов'язані з перевіркою гіпотез про дисперсії. Проведений аналіз критеріїв наявності трендів дозволяють судити про здатність критеріїв виявляти наявність лінійного та нелінійного тренду в середньому або в характеристиках розсіювання.

Розроблений метод оцінки тренду використовує базовий тест Манн-Кендалла в поєднанні з модифікованими тестами Манн-Кендалла, що дозволяє в свою чергу отримати об’єктивну оцінку трендової складової ряду. Для інтерпретації результату було застосовано метод консенсусу (тобто вихідні результати кожного з тестів додавались до спільної відповіді й враховувались як єдиний результат). Метод в подальшому може використовуватись при аналізі монотонного тренду у часових рядах.

РОЗДІЛ 3
РОЗРОБКА МОДЕЛІ ПРОГНОЗУВАННЯ ПАРАМЕТРІВ МЕРЕЖЕВОГО ТРАФІКУ

3.1 Набір вхідних даних моделі

Для побудови будь-якої моделі необхідно формалізувати та обрати дані на вхід, щоб отримати результати моделювання та інтерпретувати їх.

3.1.1 Формування та формалізація вхідних даних

Для розробки моделі прогнозування необхідно було обрати датасет для аналізу, який мав би всі необхідні дані для аналізу та відповідав області використання результатів дослідження. Вибірка повинна була налічувати більше 100 000 рядків унікальних даних та складатися би із значень параметрів мережевого трафіку.

Після аналізу багатьох існуючих наборів даних виявлось, що наразі мережа не налічує безкоштовних та відповідних до вимог датасетів.

За основу для вхідної вибірки було обрано логи (інформаційні файли моніторингу мережі) комерційної компанії, яка профілюється на аналізі мережевого трафіку для крупних клієнтів в США [22]. З отриманих логів за допомогою мови програмування Python та регулярних виражень було сформовано вибірку яка наразі налічує більше 300 000 рядків.

Сам набір даних містить 9 параметрів (рис. 3.1):

- Max packet size (bytes) – відповідає за показник максимального розміру пакету, що передається по мережі.
- Send latency (usec) – показник затримки відправника у мілісекундах.
- Recv latency (usec) – відповідає за показник затримки отримувача у мілісекундах.

- Min packet spacing (usec) – відповідає за часову різницю між пересиланням пакетів (інтервал передачі пакетів) у мілісекундах.
- Max rate (Mbps) – відповідає за показник максимальної швидкості даних в мережі на момент заміру.
- ADR (Mbps) – показник адаптивного бітрейту мережі.
- Grey bandwidth resolution – показник сірого трафіку мережі.
- Measurement date – дата, коли було сформовано дані вимірювання показників трафіку мережі.
- Available bandwidth (Mbps) – наявна пропускна здатність мережі.

	A	B	C	D	E	F	G	H	I
1	Max packet size (bytes)	Send latency (usec)	Recv latency (usec)	Min packet spacing (usec)	Max rate (Mbps)	ADR (Mbps)	Grey bandwidth resolution	Measurement date	Available bandwidth (Mbps)
2	1472	7	1	14	857.14	0.65	0.03	06/03/2009	0.65
3	1472	33	1	66	181.82	0.10	0.00	06/02/2009	0.10
4	1464	169	1	338	35.31	0.25	0.01	06/01/2009	0.25
5	1472	90	1	180	66.67	0.30	0.01	06/01/2009	0.30
6	1472	22	1	44	272.73	0.54	0.03	06/01/2009	0.54
7	1464	39	1	78	153.03	0.28	0.01	06/01/2009	0.28
8	1442	47	1	94	125.11	0.30	0.02	06/01/2009	0.30
9	1472	43	1	86	139.53	5.82	0.29	06/01/2009	2.93
10	1472	43	1	86	139.53	6.29	0.31	06/01/2009	3.16

Рисунок 3.1 – Приклад перших 10 записів сформованого датасету.

3.1.2 Попередня обробка даних

Попередня обробка даних відіграє велику роль у моделюванні. Правильно підготовлені дані покращують роботу будь-якої моделі та не допускають неправильного процесу моделювання.

Обробка даних проводилась з використанням статистичних методів (усі N/A комірки були замінені на середнє значення всього набору для відповідного параметру мережевого трафіку).

Оскільки дані налічують велику кількість рядків та носять щоденний характер, під час попередньої обробки дані були згруповані помісячно кожного року (за допомогою функції resample() з параметром групування «M». Це дозволило скоротити розмірність вхідних даних для побудови моделі.

Також для виявлення аутлаєрів було побудовано гістограми, на яких можна побачити, чи дані мають аномальні виплески. Результат побудови гістограми можна побачити на рис. 3.2.

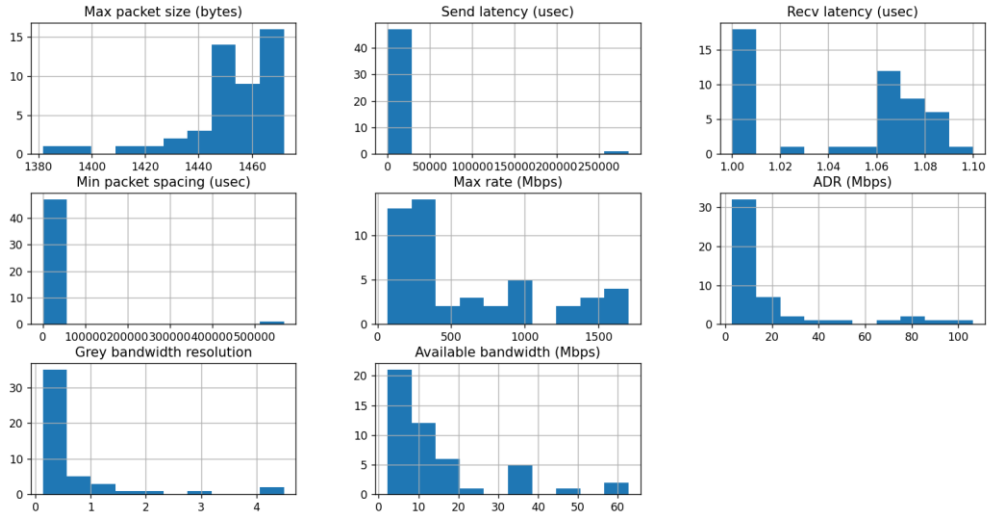

Рисунок 3.2 – Гістограми показників параметрів трафіку.

Є примітним те, що дані не мають нормальний розподіл, але для моделі прогнозування авторегресії інтегрованого ковзного середнього це не несе інформаційної навантаження, тож нормалізацію даних проводити не потрібно.

3.2 Вибір параметрів мережевого трафіку для побудови моделі

Після обробки даних постає питання: чи потрібно використовувати всі параметри для моделювання, або можна відокремити тільки ті параметри, які мають найбільший вплив. Оскільки в сформованій вибірці є параметри, які майже не змінюються з часом, то доцільним є видалити їх для подальшого моделювання.

За допомогою пакету Python statsmodel було розраховано коефіцієнти кореляції для кожного з параметрів, та обрано декілька параметрів, які мали найбільші показники кореляційності. На рис. 3.3 можна побачити побудовану heat map, завдяки якій добре видно, що найвпливовішими параметрами є параметр наявної пропускної здатності мережі (Available bandwidth) та показник сірого трафіку мережі (Grey bandwidth resolution).

Рисунок 3.3 – Кореляційна таблиця (heat map) параметрів трафіку КМ.

Ці два показники також були вибрані через наявність чіткої трендової компоненти, модель тренду має логістичний тип (змішаний).

3.3 Вибір програмного комплексу для побудови моделі

Для розробки моделі використовується мова програмування Python з використанням зовнішніх бібліотек для статистичного аналізу даних, а також для побудови моделей прогнозування.

У роботі були використані наступні бібліотеки: statsmodel, pandas, numpy, pmdarima, pymannkendall, seaborn, scikit-learn, matplotlib, XGBoost.

Пакет функцій statsmodels — це пакет Python, який є доповненням до scipy для статистичних обчислень, включаючи описову статистику та оцінку та висновок для статистичних моделей.

Пакети pandas та numpy забезпечують швидкі, гнучкі та виразні структури даних, розроблені для того, щоб зробити роботу з «реляційними» або «міченими» даними одночасно легкою та інтуїтивно зрозумілою. Вони мають на меті стати основним будівельним блоком високого рівня для практичного аналізу реальних даних у Python.

Пакет pmdarima – це статистична бібліотека, призначена для заповнення пустоти в можливостях аналізу часових рядів Python.

Пакет pymannkendall – пакет, що реалізує базовий тест МК, та має також в арсеналі його модифікації.

Пакет seaborn – це бібліотека візуалізації Python на основі matplotlib. Він забезпечує інтерфейс високого рівня для малювання привабливої статистичної графіки.

Пакет scikit-learn – це модуль Python для машинного навчання, створений на основі SciPy.

XGBoost – це оптимізована розподілена бібліотека яка імплементує метод градієнтного бустингу. XGBoost забезпечує паралельне прискорення дерева (також відоме як GBDT, GBM), яке швидко й точно вирішує багато проблем науки про дані.

Увесь перелічений набір бібліотек дозволяє швидко побудувати модель прогнозування часових рядів та забезпечити вхідні дані моделі необхідною попередньою обробкою, що дозволяє отримати кращі результати моделювання.

3.4 Побудова моделі ARIMA

Стаціонарний ЧР визначається як ряд, властивості якого не залежать від часу спостереження ряду. Таким чином, часові ряди з тенденціями або сезонністю не є стаціонарними, тоді як ряди білого шуму стаціонарні. У більш математичному сенсі ЧР називається стаціонарним, якщо він має постійне середнє значення та дисперсію, а коваріація не залежить від часу [23]. Загалом, стаціонарний ЧР не матиме довгострокових передбачуваних моделей. Можна перевірити стаціонарність ряду двома способами. З одного боку, можна перевірити це вручну, перевіривши середнє значення та дисперсію часового ряду. З іншого боку, можна оцінити стаціонарність за допомогою тестової функції.

Деякі випадки можуть заплутати. Наприклад, ЧР без тенденції та сезонності, але з циклічною поведінкою є стаціонарним, оскільки цикли не мають фіксованої тривалості.

Щоб проаналізувати тенденцію та сезонність часових рядів за допомогою кількох тестів, таких як тест Дікі-Фуллера (ADF) і Квятковського, Філіпса, Шмідта та Шина (KPSS):

Результат тесту ADF (p-значення нижче 0,05) свідчить про те, що нульову гіпотезу про наявність одиничного кореня можна відхилити з довірчим рівнем 95%. Отже, якщо p-значення нижче 0,05, ЧР є стаціонарним [24].

Результат тесту KPSS (p-значення вище 0,05) свідчить про те, що нульову гіпотезу про відсутність одиничного кореня, наявність одиничного кореня, не можна відхилити з 95% рівнем довіри. Отже, якщо p-значення нижче 0,05 - ЧР не є стаціонарним [24, 25].

Хоча ці тести описуються для перевірки стаціонарності даних, вони корисні для аналізу тенденції часових рядів, а не сезонності. Результати виконання тесту Діка-Фулера можна побачити на рис. 3.1 і 3.2.

```
ADF Test Statistic : -4.150093722323407
p-value : 0.0007994771447168711
#Lags Used : 0
Number of Observations : 47
```

Рисунок 3.1 – Результат тесту Діка-Фулера на стаціонарність для параметру *Grey bandwidth resolution*.

```
ADF Test Statistic : -3.540124708284948
p-value : 0.007014457876836656
#Lags Used : 0
Number of Observations : 47
```

Рисунок 3.2 – Результат тесту Діка-Фулера на стаціонарність для параметру *Available bandwidth (Mbps)*.

Для додаткового аналізу датасету також є доцільним провести декомпозицію часового ряду. Результати декомпозиції можемо побачити на рис. 3.3 та 3.4.

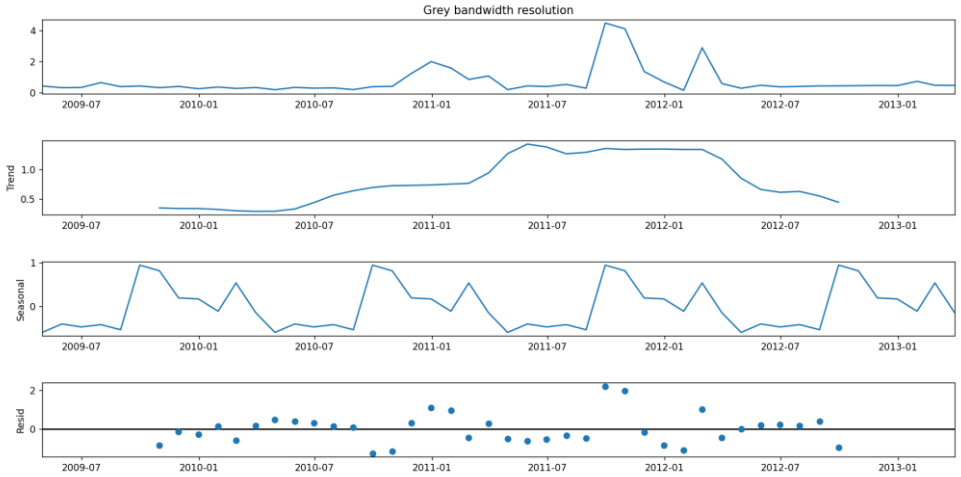

Рисунок 3.3 – Декомпозиція часового ряду параметру *Grey bandwidth resolution*.

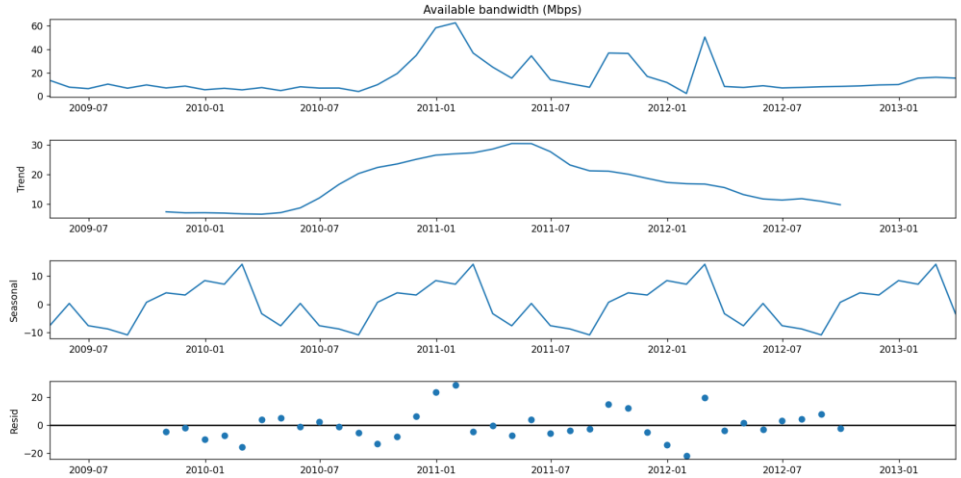

Рисунок 3.4 – Декомпозиція часового ряду параметру *Available bandwidth (Mbps)*.

Як ми бачимо, сезонна компонента не є очевидною у декомпозиційному графіку обох параметрів, що вказує також на їх стаціонарний характер. Щодо тренду можна сказати, що трендова компонента присутня, але не є яскравовираженою.

Для того, щоб оцінити методи прогнозування було розділено дані на стандартний поділ на тестові та навчальні, де останні займають обсяг у 30% даних.

Модель визначається сімома параметрами:

- p – порядок авторегресії моделі. ЧР вважається авторегресійним, якщо попередні спостереження добре описують наступні;
- d – порядок диференціювання;
- q – порядок ковзної середньої моделі, тобто розмір «ковзного вікна».

Для побудови моделі ми будемо використовувати метод підбору вхідних параметрів p, d, q та оцінювати роботу моделі за критерієм Акаіке [25]. Результат ітеративного методу визначення вхідних параметрів моделі зображено на рис. 3.5 та рис. 3.6.

```
Performing stepwise search to minimize aic
 ARIMA(2,0,2)(0,0,0)[0]             : AIC=93.695, Time=0.12 sec
 ARIMA(0,0,0)(0,0,0)[0]             : AIC=113.183, Time=0.01 sec
 ARIMA(1,0,0)(0,0,0)[0]             : AIC=91.477, Time=0.01 sec
 ARIMA(0,0,1)(0,0,0)[0]             : AIC=95.525, Time=0.02 sec
 ARIMA(2,0,0)(0,0,0)[0]             : AIC=92.358, Time=0.02 sec
 ARIMA(1,0,1)(0,0,0)[0]             : AIC=91.770, Time=0.02 sec
 ARIMA(2,0,1)(0,0,0)[0]             : AIC=93.677, Time=0.05 sec
 ARIMA(1,0,0)(0,0,0)[0] intercept   : AIC=89.811, Time=0.01 sec
 ARIMA(0,0,0)(0,0,0)[0] intercept   : AIC=99.587, Time=0.02 sec
 ARIMA(2,0,0)(0,0,0)[0] intercept   : AIC=88.590, Time=0.03 sec
 ARIMA(3,0,0)(0,0,0)[0] intercept   : AIC=90.575, Time=0.04 sec
 ARIMA(2,0,1)(0,0,0)[0] intercept   : AIC=90.534, Time=0.05 sec
 ARIMA(1,0,1)(0,0,0)[0] intercept   : AIC=88.791, Time=0.03 sec
 ARIMA(3,0,1)(0,0,0)[0] intercept   : AIC=92.481, Time=0.08 sec

Best model:  ARIMA(2,0,0)(0,0,0)[0] intercept
Total fit time: 0.520 seconds
```

Рисунок 3.5 – Результат ітеративного методу підбору параметрів моделі за допомогою критерію Акаіке для параметру *Grey bandwidth resolution*.

```
Performing stepwise search to minimize aic
 ARIMA(2,0,2)(0,0,0)[0]             : AIC=inf, Time=0.15 sec
 ARIMA(0,0,0)(0,0,0)[0]             : AIC=309.573, Time=0.01 sec
 ARIMA(1,0,0)(0,0,0)[0]             : AIC=263.220, Time=0.01 sec
 ARIMA(0,0,1)(0,0,0)[0]             : AIC=285.956, Time=0.02 sec
 ARIMA(2,0,0)(0,0,0)[0]             : AIC=264.068, Time=0.02 sec
 ARIMA(1,0,1)(0,0,0)[0]             : AIC=264.106, Time=0.03 sec
 ARIMA(2,0,1)(0,0,0)[0]             : AIC=266.029, Time=0.03 sec
 ARIMA(1,0,0)(0,0,0)[0] intercept   : AIC=262.031, Time=0.02 sec
 ARIMA(0,0,0)(0,0,0)[0] intercept   : AIC=285.195, Time=0.01 sec
 ARIMA(2,0,0)(0,0,0)[0] intercept   : AIC=261.343, Time=0.04 sec
 ARIMA(3,0,0)(0,0,0)[0] intercept   : AIC=263.273, Time=0.08 sec
 ARIMA(2,0,1)(0,0,0)[0] intercept   : AIC=263.316, Time=0.07 sec
 ARIMA(1,0,1)(0,0,0)[0] intercept   : AIC=262.150, Time=0.06 sec
 ARIMA(3,0,1)(0,0,0)[0] intercept   : AIC=264.033, Time=0.10 sec

Best model:  ARIMA(2,0,0)(0,0,0)[0] intercept
Total fit time: 0.644 seconds
```

Рисунок 3.6 – Результат ітеративного методу підбору параметрів моделі за допомогою критерію Акаіке для параметру *Available bandwidth (Mbps)*.

Як ми бачимо для обох обраних параметрів трафіку КМ найкращі результати показала модель ARIMA(2,0,0)(0,0,0).

Після того, як ми обрали вхідні параметри моделі ми можемо її побудувати. Результати моделювання можна побачити на рис. 3.7 та рис. 3.8.

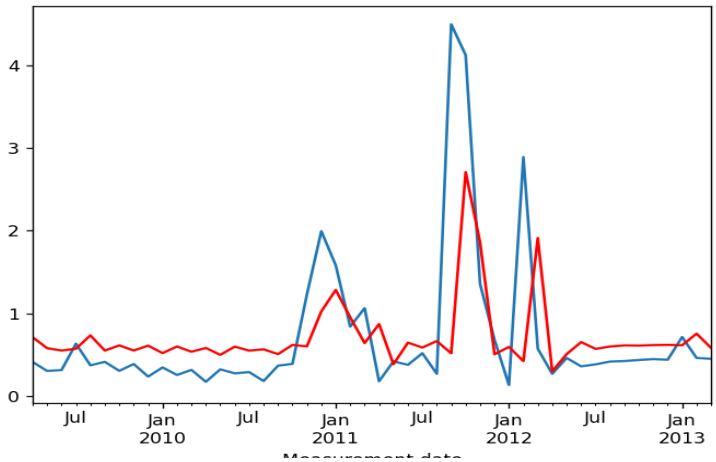

Рисунок 3.7 – Графік роботи моделі прогнозування параметру *Grey bandwidth resolution*.

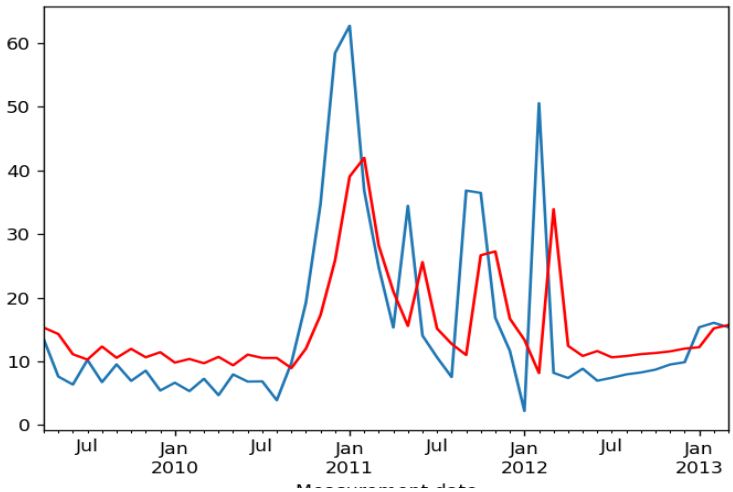

Рисунок 3.8 – Графік роботи моделі прогнозування параметру *Available bandwidth (Mbps).*

Синім виділені актуальні дані, червоним – результати моделі прогнозування. Як можна побачити, за попередньою оцінкою точність прогнозування не є високою.

3.5 Результати роботи моделі

Для оцінки лінійних моделей використовуються три основні показники. Це є середня абсолютна помилка (MAE), середньоквадратична помилка (MSE) або коренева середньоквадратична помилка (RMSE). Для розрахунків цих метрик було використано пакет sklearn. Результати виведення метрик для параметру Available Bandwidth (Mbps) та Grey bandwidth resolution можна побачити на рис. 3.9 та рис. 3.10 відповідно.

```
ARIMA MAE metric value: 7.973378535832609.
ARIMA RMSE metric value: 13.6265085842131.
ARIMA MAPE metric value: 0.6187122469061571.
```

Рисунок 3.9 – Результати метрик якості роботи моделі для параметру *Available Bandwidth (Mbps).*

```
ARIMA MAE metric value: 0.4547754766998217.
ARIMA RMSE metric value: 0.749160191520123.
ARIMA MAPE metric value: 0.7283062467149921.
```

Рисунок 3.10 – Результати метрик якості роботи моделі для параметру *Grey bandwidth resolution.*

Виходячи з значень метрик можна сказати, що модель показала погані результати, оскільки має значення показнику MAPE > 50, а саме 60%-70% передбачень прогнозування виявляються невірними, що є неприємним навіть для макроаналітичного моделювання прогнозів.

Висновки за розділом 3

Розуміння основних закономірностей часових рядів і уміння впроваджувати моделі прогнозування часових рядів є завданням під час аналізу часового ряду.

До очевидних переваг моделей класу ARIMA можна віднести те, що вони мають дуже чітке математико-статистичне обґрунтування, що робить їх одними з найбільш науково-обґрунтованих моделей з безлічі моделей прогнозування тенденцій у часових рядах. Ще однією перевагою є формалізована і найбільш докладно розроблена методика, дотримуючись якої можна підібрати модель, що найбільше підходить до кожного конкретного ЧР.

Серйозним недоліком є неадаптивність моделей авторегресії: при отриманні нових даних модель потрібно періодично переоцінювати, а іноді і переідентифікувати.

В даному розділі була розроблена модель прогнозування даних ARIMA для параметрів трафіку комп’ютерної мережі. Для побудови моделі використовувалась мова програмування Python та зовнішні набори бібліотек.

Вхідні дані для моделі були отримані нетривіальним шляхом – а тобто, через парсинг моніторингових файлів за допомогою мови програмування

Python. Результат парсингу дав необхідний для аналізу набір даних. Задля успішного застосування цих даних було також проведено їх попередню обробку, після якої була сформована вибірка, згрупована за місяцями.

Результати моделювання показали, що модель має свої недоліки й має завеликий процент помилок (показник MAPE – 60-70%), що означає в свою чергу неточність прогнозування точкових значень.

РОЗДІЛ 4
ВИКОРИСТАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ В МОДЕЛЯХ ПРОГНОЗУВАННЯ ПАРАМЕТРІВ МЕРЕЖЕВОГО ТРАФІКУ

4.1 Методи машинного навчання

4.1.1 Метод градієнтного бустингу

Градієнтний бустинг – це сучасний алгоритм машинного навчання для вирішення завдань класифікації та регресії. Він будує передбачення у вигляді ансамблю слабких моделей, що передбачають, якими в основному є дерева рішень. З кількох слабких моделей у результаті ми збираємо одну, але вже ефективну. Загальна ідея алгоритму – послідовне застосування предиктору в такий спосіб, що кожна наступна модель зводить помилку попередньої до мінімуму [25].

Метод градієнтного бустингу є актуальною темою у світі машинного навчання протягом багатьох років, але ще не набуло особливої уваги у аналізі часових рядів. Насамперед це пов’язано з тим, що більшість переваг градієнтного бустингу надходить від методів, які певним чином поділяють вхідні дані, наприклад дерева рішень [26]. Градієнтний бустинг може використовуватись також для параметричного моделювання часових рядів.

Алгоритм починає роботу з побудови початкової моделі та коригує її, покроково створюючи послідовність дерев регресії або використовуючи інші базові методи (лінійна регресія, нейронна мережа тощо).

Кожне дерево у послідовності створюється на підставі залишків моделі, яка зводиться на попередньому етапі. Залишки моделі по суті використовуються як цільова змінна.

Градієнтний бустинг можна використовувати якщо:

- є велика кількість спостережень (наближеної подібності) у тренувальному наборі даних;

- кількість ознак менша кількості спостережень в даних, які використовуються для навчання;
- дані включають в себе суміш кількісних та категоріальних ознак або тільки кількісні ознаки;
- необхідно розглянути метрики роботи моделі.

Плюси використання методу градієнтного бустингу:

- алгоритм працює з будь-якими функціями втрат;
- передбачення у середньому краще, ніж в інших алгоритмів;
- самостійно справляється із пропущеними даними;

Мінуси використання:

- алгоритм дуже чутливий до викидів і за їх наявності витрачатиме величезна кількість ресурсів на ці моменти. Однак, слід зазначити, що використання Mean Absolute Error (MAE) замість Mean Squared Error (MSE) значно знижує вплив викидів на вашу модель (вибір функції у параметрі criterion).
- модель буде схильна до перенавчання за дуже великої кількості дерев. Ця проблема є у будь-якому алгоритмі, пов'язаному з деревами і справляється правильним налаштуванням параметра **n_estimators**.
- обчислення можуть забрати багато часу. Тому, якщо є великий набір даних, завжди необхідно складати правильний розмір вибірки і не забувайте правильно налаштувати параметр **min_samples_leaf**.

4.2 Покращення існуючої моделі прогнозування

При розробці рішення на основі градієнтного бустингу використовувалися покращені версії алгоритму, зокрема, XGBoost [27]. Його відмінною особливістю є більш швидка реалізація розрахунків і використання регулярізації при побудові дерев рішень.

Найбільш важливими параметрами при побудові бустингу є:

- B – тип базового алгоритму для бустингу: лінійна модель, дерево рішень, нейронна мережа та інші. У цьому дослідженні застосовується класичний підхід до визначення базового алгоритму – дерева рішень.
- η – швидкість навчання алгоритму. Відповідає за ефективність збіжності алгоритму та його можливості потрапляння до глобального мінімуму.
- N - Число ітерацій алгоритму. Часто, коли в основі алгоритму приймаються дерева рішень, говорять про кількість дерев.
- MD – максимальна глибина дерев. Відповідає за структурну складність дерев. Зміна параметра допомагає уникнути перенавчання.
- MC – мінімальна кількість об'єктів у аркуші дерева. Також дозволяє регулювати складність дерев.
- δ – частка об'єктів вибірки, що використовуються на кожній ітерації алгоритму. Дозволяє збільшити стійкість алгоритму та значно підвищити якість моделі.
- c – частка ознак для кожної ітерації. δ та c дозволяють використовувати
- переваги випадкового лісу у межах градієнтного бустингу.

Це не повний перелік можливих параметрів, що говорить про достатній рівень складності представленого алгоритму. Тим не менш, він дає якісний результат, який призводить до підвищеної точності прийнятих рішень.

```
********** Round 1 **********
Using Split: None
Fitting initial trend globally with trend model:
median()
seasonal model:
None
cost: 1.1159969736454058
********** Round 2 **********
Using Split: None
Fitting global with trend model:
arima(auto)
seasonal model:
None
cost: 0.6487239071327346
==============================
Boosting Terminated
```

Рисунок 4.1 – Виконання алгоритму градієнтного бустингу для моделі прогнозування ARIMA параметру *Grey bandwidth resolution*.

```
********** Round 1 **********
Using Split: None
Fitting initial trend globally with trend model:
median()
seasonal model:
None
cost: 274.2195172739998
********** Round 2 **********
Using Split: None
Fitting global with trend model:
arima(auto)
seasonal model:
None
cost: 102.15555755013685
==============================
Boosting Terminated
```

Рисунок 4.2 – Виконання алгоритму градієнтного бустингу для моделі прогнозування ARIMA параметру *Available bandwidth (Mbps)*.

Графіки результатів виконання роботи моделі з використанням методу градієнтного бустингу зображено на рис. 4.3 та 4.4.

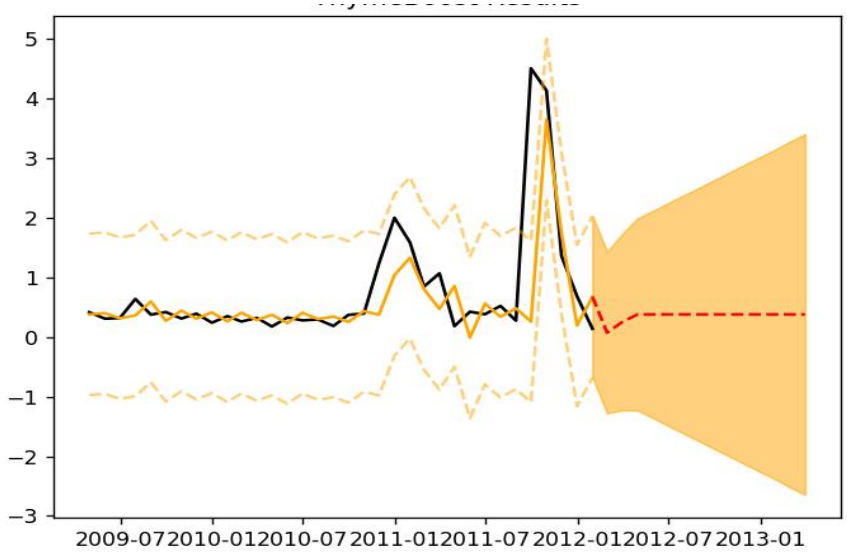

Рисунок 4.3 – Графік результатів роботи моделі з застосуванням градієнтного бустингу для параметру *Grey bandwidth resolution*.

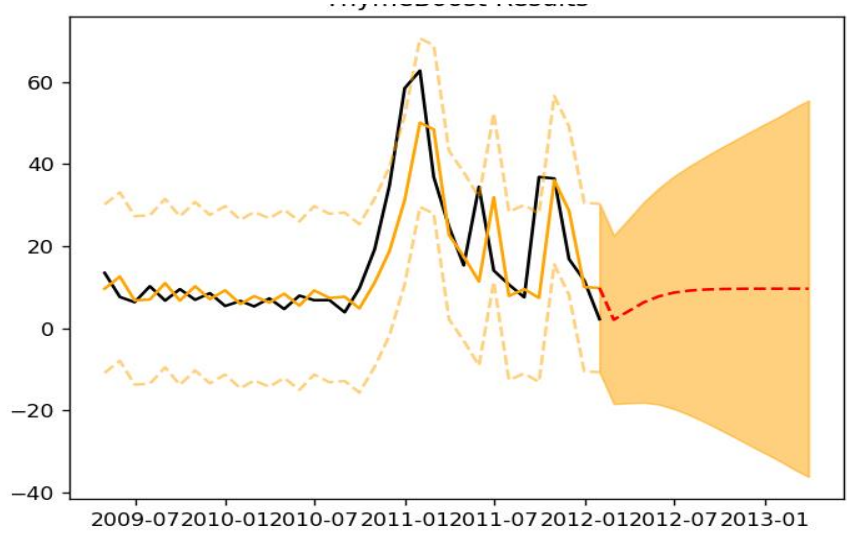

Рисунок 4.4 – Графік результатів роботи моделі з застосуванням градієнтного бустингу для параметру *Available bandwidth (Mbps)*.

4.3 Результат роботи моделі

Для аналізу результатів роботи моделі використано такі ж самі параметри, що застосовуються для аналізу роботи моделі у розділі 3.

```
Boosted ARIMA MAE metric value: 0.29660439359158175.
Boosted ARIMA RMSE metric value: 0.7665433051989342.
Boosted ARIMA MAPE metric value: 0.4750010581561023.
```

Рисунок 4.5 – Результати метрик якості роботи моделі після застосування градієнтного бустингу для параметру *Grey Bandwidth resolution.*

```
Boosted ARIMA MAE metric value: 5.712938677418449.
Boosted ARIMA RMSE metric value: 13.337079960643061.
Boosted ARIMA MAPE metric value: 0.44330833029659433.
```

Рисунок 4.6 – Результати метрик якості роботи моделі після застосування градієнтного бустингу для параметру *Available Bandwidth (Mbps).*

Дивлячись на результати можна сказати, що модель зазнала значних покращень, що дозволяє говорити про те, що метод градієнтного бустінгу може бути успішно застосований в подібних задачах.

Відносне покращення показників роботи моделі приведено в таблиці 4.1.

Таблиця 4.1

Метрики роботи моделей

	Grey bandwidth resolution			Available bandwidth (Mbps)		
Назва моделі	MAE	RMSE	MAPE	MAE	RMSE	MAPE
ARIMA	0.454	0.749	0.728	7.973	13.626	0.618
Boosted ARIMA	0.296	0.766	0.475	5.71	13.33	0.44

Як ми бачимо всі роди помилок було зменшено приблизно на 30-35%.

Висновки за розділом 4

У цьому розділі було застосовано покращення моделі прогнозування ARIMA за допомогою градієнтного бустингу. Зростаюча популярність методів машинного навчання, зокрема градієнтного бустингу, полягає в їх здатності витягувати ознаки з рядів і включати їх у моделі без вказівки параметрів, як у випадку зі стандартними статистичними алгоритмами. Емпіричне дослідження довело їхню перевагу щодо точності прогнозу, особливо для довгих вибірок. Крім того, ці моделі менш схильні до переобладнання та дозволяють користувачеві включати несуттєві змінні та параметри без втрати передбачуваності моделі.

Застосування сучасних методів машинного навчання можуть бути успішно застосовані у задачах прогнозування з використанням регресійних моделей. Доопрацьована модель на 30% має кращі показники роботи моделі.

ВИСНОВКИ

На сьогоднішній день існують різні методи виявлення та оцінки трендів у трафіку комп’ютерних мереж. Побудова моделей аналізу трафіку комп’ютерних мереж за допомогою дослідження трендів часових рядів, а також лінійних алгоритмів регресії дозволять підвищити точність та якість аналізу стану КМ. У роботі був сформований набір даних, який характеризує КМ, для його подальшого аналізу і виявлення тренду.

Після огляду методів аналізу часових рядів та їх особливостей можна сказати, що ЧР - це хронологічна послідовність спостережень за певною змінною. Часові ряди мають складові компоненти, які дозволяють отримати цілковито повний опис даних та правильно підібрати модель для подальшого прогнозування.

Мета побудови моделі часових рядів така ж, як і для інших типів прогнозних моделей, тобто створити модель так, щоб помилка між прогнозованим значенням цільової змінної та фактичним значенням була якомога меншою.

Розроблений метод оцінки тренду використовує базовий тест МК в поєднанні з модифікованими тестами МК, що дозволяє в свою чергу отримати об’єктивну оцінку трендової складової ряду. Для інтерпретації результату було застосовано метод консенсусу (тобто вихідні результати кожного з тестів додавались до спільної відповіді й враховувались як єдиний результат). Метод в подальшому може використовуватись при аналізі монотонного тренду у часових рядах.

Однією з задач дослідження було обрати датасет для аналізу, який мав би всі необхідні дані для аналізу та відповідав області використання результатів дослідження. За основу для даних було обрано логи комерційної компанії, яка профілюється на аналізі мережевого трафіку для крупних клієнтів. З отриманих логів за допомогою розробленого мовою програмування

Python програмного забезпечення було сформовано вибірку (за допомогою парсингу текстових файлів на основі регулярних виразів), яка наразі налічує більше 300 000 рядків.

Була побудована модель прогнозування на основі ковзного середнього з використанням основних рішень для регресійних моделей – ARIMA. При виборі параметрів моделі p, d, q було застосовано ітеративний метод підбору цих параметрів на основі критерію Акаіке. Після отримання результатів моделювання було запропоновано покращити модель за допомогою методів машинного навчання, а саме методу градієнтного бустинга.

Інтерпретуючи результати моделювання можна сказати, що сучасні моделі прогнозування можуть цілком успішно використовуватись в поєднанні з методами машинного навчання, що дає змогу покращувати результати моделювання.

Результати роботи можуть бути використані в різних типах задач, таких як задачах аналізу трафіку КМ, прогнозування станів технічних систем, розробка моделей КМ, оцінювання функціонування комп’ютерних мереж, а також використані для аналізу та оцінки поточного трафіку КМ для подальшої оптимізації, пошуку перспектив для подальшого розвитку у структурі мережі, виявлення проблем, що негативно впливають на систему.

Використання сучасних методів аналізу та оцінки трафіку відіграють велику роль у розвитку комп’ютерних мереж, тож необхідне постійне вдосконалення існуючих методів та впровадження нових.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Часові ряди. [Електронний ресурс] Режим доступу: https://kstat.pnu.edu.ua/wp-content/uploads/sites/63/2018/04/%D0%A7%D0%B0%D1%81%D0%BE%D0%B2%D1%96-%D1%80%D1%8F%D0%B4%D0%B8.pdf

2. Андерсон Т. Статистичний аналіз часових рядів. 1976. - 756 с.

3. Андрієнко В.М. Теоретичні та методологічні аспекти моделювання статистичних рядів даних. Развитие науки в XXI веке: сборник статей VII Международной конференции. Ч. 2. Харьков. С. 68-76.

4. Андриенко В.М., Арсирий Е.А. Интеллектуальный анализ временных рядов со стохастическим трендом. Восточно-европейский журнал передовых технологий. 2011. № 4/4 (52). С. 4-8.

5. Баклан І.В., Степанкова Г.А. Імовірнісні моделі для аналізу та прогнозування часових рядів. Искусственный интеллект. 2008. № 3. С. 505-515.

6. Березька К.М., Маслій В.В. Методологічні аспекти застосування моделі нечітких часових рядів для прогнозування податкових надходжень. Актуальні проблеми економіки. 2011. № 1. С. 227-235.

7. Андриенко В.М., Арсирий Е.А. Комплексная методология анализа, моделирования и прогнозирования временных рядов. Современный научный вестник. Серия «Математика». 2010. № 13 (95). С. 71-92.

8. Ковтун Н. В. Теорія статистики: Курс лекцій, практикум. - К.: ІмексЛТД, 2007. - 276 с

9. Баклан І.В., Степанкова Г.А. Імовірнісні моделі для аналізу та прогнозування часових рядів. Искусственный интеллект. 2008. № 3. С. 505-515.

10. Gray, Katharine Lynn, “Comparison of Trend Detection Methods” (2007). Graduate Student Theses, Dissertations, & Professional Papers. 228.

11. Ханк, Д.Э. Бизнес-прогнозирование, 7-е изд.: Пер. с англ. / Д.Э. Ханк, Д.У. Уичерн, А.Дж. Райтс. - М.: Издательский дом «Вильямс», 2003. - 458 с.

12. М. Кендалл, А. Стьюарт. Многомерный статистический анализ и временные ряды. М.: Главная редакция физико-математической литературы изд-ва «Наука», 1976.

13. Баклан І.В., Степанкова Г.А. Про деякі нові особливості використання прихованих марковських моделей для аналізу та прогнозування часових рядів. Искусственный интеллект. 2010. № 4. С. 337-341.

14. IBM Corporation Documentation, “Trend detection”, 2021. URL: https://www.ibm.com/docs/en/siffs/2.0.3?topic=learning-trend-detection

15. Robert Nau, “Statistical Forecasting: notes on regression and time-series analysis” (2020). ARIMA Models for time-series forecasting. URL: https://people.duke.edu/~rnau/arimreg.htm

16. Mann, H.B, Non-parametric tests against trend (1945), Econometrica, 13:163-171.

17. Рилова Н. Синтез ARIMA-моделей для прогнозування коефіцієнтів виходу кондиційних напівпровідникових матеріалів / Н. Рилова, І. Оксаніч // Системи обробки інформації. - 2015. - Вип. 5 (130). - С. 102-107.

18. Бокс Дж., Дженкинс Г. Анализ временных рядов, прогноз и управление: Пер. с англ. // Под ред. В.Ф. Писаренко. – М.: Мир, 1974, кн. 1. - 406 с.

19. Porn, K., Shen, K., Kjall, H., Occurrence trend analysis applied to commercial IFR air taxi safety in the Nordic countries - A new Bayesian approach. STUDSVIK/ES-93/36, ISBN 91-7010-228-7. Prepared for the Swedish Transport Research Board. Published by Studsvik EcoSafe, Nykoping, Sweden 1993.

20. Porn, K., Shen, K., Nyman, R., I-Book, Edition 2. Initiating Events at the Nordic Nuclear Power Plants (in Swedish). SKI Report 94:12, Swedish Nuclear Power Inspectorate, Stockholm 1994, ISSN 1104-1374.

21. Dickey D.A., Fuller W.A. Distribution of the Estimators for Autoregressive Time Series with a Unit Root. Journal of the American Statistical Association. 1979. № 74. P. 427-431.

22. Real World Networking datasets. [Електронний ресурс] Режим доступу:https://gist.github.com/stefanbschneider/96602bb3c8b256b90058d59f337a0e59

23. Dekel Shai. Wavelet Decomposition of Gradient Boosting / Dekel Shai, Oren Elisha, Ohad Morgan // CoRR abs/1805.02642 . - 2018 . - P.1-13.

24. Snyder, D.L., Random Point Processes, John Wiley & Sons, New York 1976. 485 p. Tanner, M.A., Tools for Statistical Inference. Springer-Verlag, New York 1991. 110 p.

25. Tibor, C, Some parameter-free tests for trend and their application to reliability analysis. Reliability Engineering and System Safety 41(1993), pp.225-230

26. Корнієнко Є.В. Методи прогнозування та прийняття рішень. Концеп - Жовтень 13, 2012 - 100 с.

27. Chen T. XGBoost: A scalable tree boosting system / T.Chen, C.Guestrin // 22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining . - San Francisco, CA, 2016 . - P.785-794.

ДОДАТКИ

Додаток А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В. Н. Каразіна

Факультет комп'ютерних наук
Кафедра теоретичної та прикладної системотехніки
Рівень вищої освіти (освітньо-кваліфікаційний рівень) Магістр
Галузь знань: 12 – Інформаційні технології
Спеціальність: 123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри теоретичної
та прикладної системотехніки
_____д.т.н., проф. Шматков С. І.

«_10_» _грудня__2021_року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дорошенко Максим Ігорович
(прізвище, ім'я, по батькові студента)

1. Тема роботи Метод виявлення і оцінки параметрів тренду у трафіку комп'ютерної мережі
керівник роботи Стрілець Вікторія Євгенівна, кандидат технічних наук,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від "10" грудня 2021 року №_____

2. Строк подання студентом роботи 30 листопада 2022 року

3. Перелік питань, які потрібно розробити

1. Аналіз методів і моделей виявлення тренду у трафіку комп'ютерної мережі.
2. Розробка методу виявлення тренду у трафіку комп'ютерної мережі та методу прогнозування трафіку.
3. Розробка моделі прогнозування трафіку, та аналіз результатів моделювання.
4. Покращення показників моделі прогнозування за допомогою методів машинного навчання.

4. План роботи

№ з/п	Назви етапів роботи	Термін виконання етапів роботи
1	Аналіз наукової та профільної літератури	До 07.01.2022
2	Аналіз існуючих моделей трендів	До 15.01.2022
3	Пошук вхідного датасету	До 27.01.2022
4	Огляд існуючих методів оцінки параметрів тренду	До 17.02.2022
5	Розробка методу виявлення тренду	До 11.03.2022
6	Аналіз існуючих моделей прогнозування даних часового ряду	До 26.03.2022
7	Вибір засобів та елементів для розробки моделі прогнозування даних часових рядів	До 01.04.2022
8	Розробка моделі прогнозування даних часового ряду	До 28.06.2022
9	Аналіз моделі та результатів моделювання	До 09.07.2022
10	Огляд методів машинного навчання для підвищення якості роботи моделі	До 22.08.2022
11	Розробка моделі прогнозування з використанням методів машинного навчання	До 27.09.2022
12	Аналіз результатів роботи моделі після застосування методу градієнтного бустингу	До 03.10.2022
13	Підготовка та оформлення пояснювальної записки	До 25.11.2022
14	Представлення пояснювальної записки керівнику та рецензенту	До 30.12.2022

5. Дата видачі завдання 10.12.2021

Студент Дорошенко М.І. _____
ініціали, прізвище підпис

Керівник роботи Стрілець В.Є. _____
ініціали, прізвище підпис

Додаток Б

ІНДИВІДУАЛЬНЕ ТЕХНІЧНЕ ЗАВДАННЯ.

Технічне завдання

на розробку програмного виробу

«Метод виявлення та оцінки параметрів тренду у трафіку комп’ютерної мережі»

Назва розділу	Назва і зміст підрозділу
1. Введення	1.1. Назва програмного виробу: Метод виявлення та оцінки параметрів тренду у трафіку комп’ютерної мережі. 1.2. Галузь застосування: Автоматизація та аналіз станів комп’ютерних систем.
2. Підстава для розробки	2.1. Навчальний план за спеціальністю 123 - «Комп’ютерна інженерія» 2.2. Завдання на кваліфікаційну роботу магістра затверджено наказом ректора № _____ від _____.
3.Призначення розробки	3.1. Мета розробки програмного виробу: підвищення якості прогнозування стану трафіку, покращення методів оцінки тренду часових рядів. 3.2. Призначення програмного виробу: використання виробу для підвищення рівня роботи моделей прогнозування за допомогою використання методів машинного навчання, а саме методу градієнтного бустингу. 3.3. Вихідні дані для розробки: комп’ютерна модель прогнозування параметрів мережевого трафіку, метод виявлення трендів. У якості прототипу взятий метод виявлення тренду Манн-Кендалла з використанням методу консенсусу, що складається з результату трьох окремих методів виявлення тренду у часовому ряду описаний у науковому збірнику «Праці міжнародної науково-технічної конференції «Комп'ютерне моделювання у наукоємних технологіях (КМНТ-2022)», (21-25

	листопада 2022 р., м. Харків, Україна). – Харків: ХНУ, 2022. // Дорошенко М.І., Стрілець В.Є. Метод аналізу та оцінки параметрів тренду у трафіку комп'ютерних мереж.
4. Технічні вимоги до програмного виробу	4.1. Вимоги до функціональних характеристик: можливість прогнозувати значення параметрів трафіку комп’ютерних мереж з використанням методів машинного навчання. 4.2. Вимоги до надійності: забезпечення працездатності методу виявлення та оцінки тренду у трафіку комп’ютерних мереж при подачі вхідних даних неправильного формату, а також працездатність комп’ютерної моделі прогнозування. 4.3. Вимоги до умов експлуатації: встановлення мови програмування Python та необхідних бібліотек для роботи моделі прогнозування та методу виявлення тренду. 4.4. Вимоги до складу і параметрів технічних засобів: комп’ютер або ноутбук із 4 ГБ оперативної пам’яті та процесором не нижче Intel Core i3 9-го покоління. 4.5. Вимоги до інформаційної та програмної сумісності: ОС Linux або Windows 8/10, підтримка Anaconda, Python 3. 4.6. Вимоги до маркування та упаковки: жорстка упаковка з пластмаси, маркування на українській та англійській мовах. 4.7. Вимоги до транспортування і зберігання: транспортування в упаковці будь-яким способом, температура транспортування/зберігання +5-+20 С. 4.8. Спеціальні вимоги: відсутні.

<table>
<tr><td>5. Вимоги до програмної документації.</td><td colspan="2">Програмною документацією до виробу «Метод виявлення та оцінки параметрів тренду у трафіку комп’ютерної мережі» вважати:
1) Справжнє Технічне завдання на розробку програмного виробу (представити у вигляді Додатку Б до пояснювальної записки до кваліфікаційної роботи).
2) Програму і методику випробувань розробленого програмного виробу (представити у вигляді Додатку В до пояснювальної записки до кваліфікаційної роботи).
3) Опис програмного виробу (представити в розділах 3, 4 пояснювальної записки до кваліфікаційної роботи).
4) Текст програми (представити в Додатку Г до пояснювальної записки до кваліфікаційної роботи).</td></tr>
<tr><td>6. Техніко економічні показники</td><td colspan="2">Орієнтовна оцінка якості виконуваного прогнозування після застосування методу градієнтонго бустингу склала 0.43 (показник MAPE). Порівняння ефективності з базовою моделлю прогнозування ARIMA представити у розділі 4.</td></tr>
<tr><td rowspan="2">7. Стадії і етапи розробки</td><td>Дата</td><td>Назва етапу</td></tr>
<tr><td>25.12.2021-07.01.2022

08.01.2022 - 15.01.2022

16.01.2022 - 27.01.2022

28.01.2022 - 18.02.2022

19.02.2022 -11.03.2022</td><td>1. Аналіз наукової та профільної літератури
2. Аналіз існуючих моделей трендів
3. Пошук вхідного датасету
4. Огляд існуючих методів оцінки параметрів тренду
5. Розробка методу</td></tr>
</table>

	12.03.2022 - 26.03.2022 27.03.2022 - 01.04.2022 02.04.2022 - 28.06.2022 30.06.2022 - 09.07.2022 11.07.2022 - 22.08.2022 23.08.2022 - 27.09.2022 29.09.2022 - 03.10.2022	виявлення тренду 6. Аналіз існуючих моделей прогнозування даних часових рядів 7. Вибір засобів та елементів для розробки моделі прогнозування даних часових рядів. 8. Розробка моделі прогнозування 9. Аналіз моделі та результатів моделювання 10. Огляд методів машинного навчання для підвищення якості роботи моделі 11. Доопрацювання моделі прогнозування з використанням методу градієнтного бустингу 12. Аналіз результатів роботи моделі після покращення
8. Порядок контролю і приймання	В даному розділі повинні бути вказані загальні вимоги до приймання розробленого програмного виробу наприклад: 1) Перевірка ходу розробки програмного виробу. Керівнику робіт виконувати 1 раз в 3 тижні. 2) Випробування програмного виробу відповідно до Програми і методики випробувань провести на базі комп'ютерного класу або приватного приміщення. 3) Захист розробленого програмного виробу провести на засіданні атестаційної комісії.	

	4) Пояснювальну записку надати на паперових носіях в одному примірнику, в електронному вигляді.

Виконавець:

студент групи КІ-61

Дорошенко М.І.

Замовник:

кандидат технічих наук

Стрілець В.Є.

Додаток В

Програма і методика випробувань програмного виробу

«Метод виявлення та оцінки параметрів тренду у трафіку комп’ютерної мережі»

1. Об'єкт випробувань

1. Назва програмного виробу: «Метод виявлення та оцінки параметрів тренду у трафіку комп’ютерної мережі».
2. Галузь застосування: Автоматизація та аналіз станів комп’ютерних систем.
3. Перераховані відомості запозичуються з відповідних розділів Технічного завдання.

2. Мета випробувань

Перевірка відповідності функціональності програмної реалізації системи заявленим функціональним можливостям в технічному завданні (Додаток Б до пояснювальної записки до дипломної роботи).

3. Загальні положення

3.1. Підстави для проведення випробувань

Підставою для проведення випробувань є наказ про призначення атестаційної комісії.

3.2. Місце і тривалість випробувань

Приймальні (приймально-здавальні) випробування проводяться на базі комп'ютерного класу кафедри або приватного приміщення в період роботи атестаційної комісії.

3.3. Обсяг випробувань

Приймальні випробування програмного вироби проводяться в обсязі відповідному цієї програми і методики випробувань.

3.4. Організації, які беруть участь у випробуваннях

Приймальні випробування проводяться атестаційною комісією напередодні засідання (або в процесі засідання) за участю Замовника, Виконавця та інших осіб, присутніх на засіданні.

4. Вимоги до програми або програмного виробу

Модель повинна задовольняти наступним вимогам:

– працювати на основних операційних системах: Windows, Linux, MacOS;

– вимоги до надійності;

– передбачити захист від некоректних дій користувача;

– сумісність з іншими програмними продуктами;

– бути легко розширюваною;

– елементи програми повинні бути ізольовані одне від одного для зменшення їх впливу на роботи програми під час редагування програмного коду;

– вимоги до складу і параметрів технічних засобів;

– вимоги до маркування та упаковки (не висуваються);

– вимоги до транспортування і зберігання (не висуваються).

– Спеціальні вимоги (не пред’являються).

5. Вимоги до програмної документації

Програмною документацією до виробу «Метод виявлення та оцінки параметрів тренду у трафіку комп’ютерної мережі» вважати:

– справжнє технічне завдання на розробку програми (представити як Додаток Б до пояснювальної записки до кваліфікаційної роботи);

– програму і методику випробувань розробленої програми (представити як Додаток В до пояснювальної записки до кваліфікаційної роботи);

– рекомендацій щодо застосування створеної програмної стандартизації у проектах (представити в Розділі 3 пояснювальної записки до кваліфікаційної роботи);

– текст програми(представити як Додаток Г до пояснювальної записки до кваліфікаційної роботи).

6. Засоби і порядок випробувань

6.1. Засоби випробувань

Для проведення випробувань необхідний проект для розробки програмної моделі за допомогою мові програмування Python з використання бібліотек sklearn, seaborn, XGBoost, pymannkendall, pmdarima, statsmodels та вхідним датасетом з параметрами трафіку комп'ютерної мережі.

6.2. Порядок проведення випробувань

Як правило, випробування проводяться в два етапи:

- ознайомчий (1-й етап);

- випробування програмного виробу (2-й етап).

Перелік перевірок, що проводяться на 1 етапі випробувань, включає в себе:

1. Перевірку комплектності програмної документації.

2. Перевірка комплектності складу програмної документації здійснюється за критерієм наявності зазначеної в ТЗ документації.

3. Перевірку комплектності складу технічних і програмних засобів.

4. Методику проведення перевірок на 1 етапі випробувань.

5. Якість програмної документації перевіряється на відповідність вимогам стандартів ЕСПД.

Перелік перевірок, що проводяться на 2 етапі випробувань, включає в себе:

1. Перевірку відповідності технічних характеристик програми вимогам технічного завдання.

2. Перевірку ступеня виконання функціональних вимог до програми.

3. Методику проведення перевірок, що входять до переліку по 2 етапу випробувань.

1. Програма працює відповідно до умов експлуатації операційних систем MS Windows, Linux та MacOS.

2. Для роботи необхідний компілятор мови програмування Python, версії не нижчої ніж 3.0.

3. Порядок проведення випробувань:

3.1. Спочатку перед запуском моделювання необхідно запустити програму парсингу вхідних даних. Запуск такої програми здійснюється за допомогою запуску скріпту для парсингу файлів в датасет на мові програмування python: “python <ім’я скрипту>”. Виконання програми здійснюється в директорії разом моніторинговими файлами (логами);

3.2. Після запуску програми парсингу необхідно у директорії проекту в навігації знайти директорію під назвою «output» в якому буде знаходитись результат роботи скріпту – файл з розширенням .csv.

3.3. Після чого необхідно запустити програму моделі прогнозування параметрів трафіку комп’ютерної мережі за допомогою запуску програми на мові програмування python “python <ім’я файлу>;

3.4. Після натискання на екрані з’явиться результат моделювання з графічними елементами.

Для проведення випробувань пропонується провести тест 1, тест 2, тест 3 та тест 4.

Тест 1

1. Перевірка виконання скріпту парсингу логів моніторингу.
2. Запуск скриптового файлу для початку аналізу та парсингу даних.
3. Отримання результатів у вигляді файлу, а також консольні логи про те, що програма була успішно виконана.

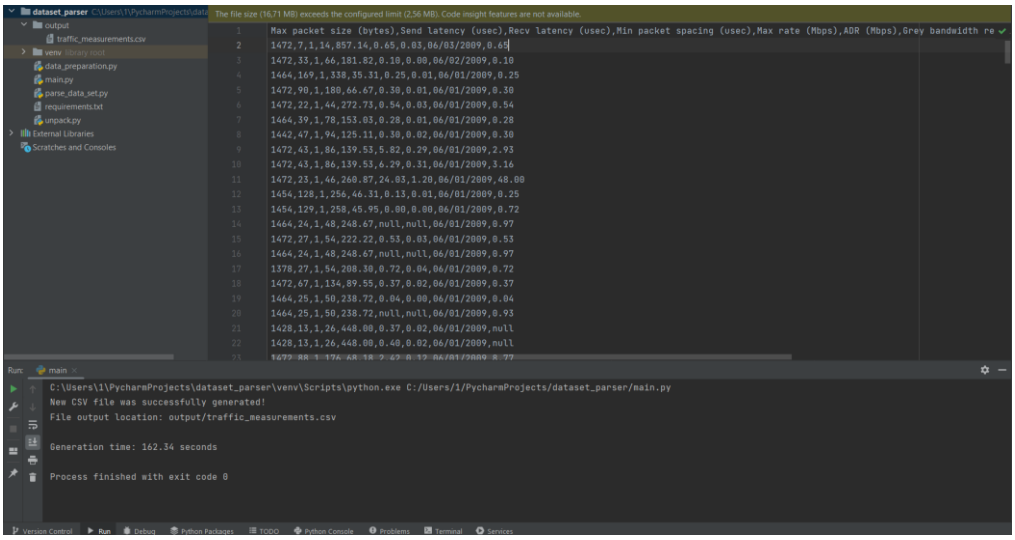

Рис. В.1 Тест 1

Тест 2

1. Перевірка виконання методу виявлення тренду у часовому ряді.
2. Виявлення тренду у часовому ряді, в якому є графічно представлений висхідний тренд.
3. Отримання результату виконання в консолі, а також графік декомпозиції з наявною трендовою компонентою.

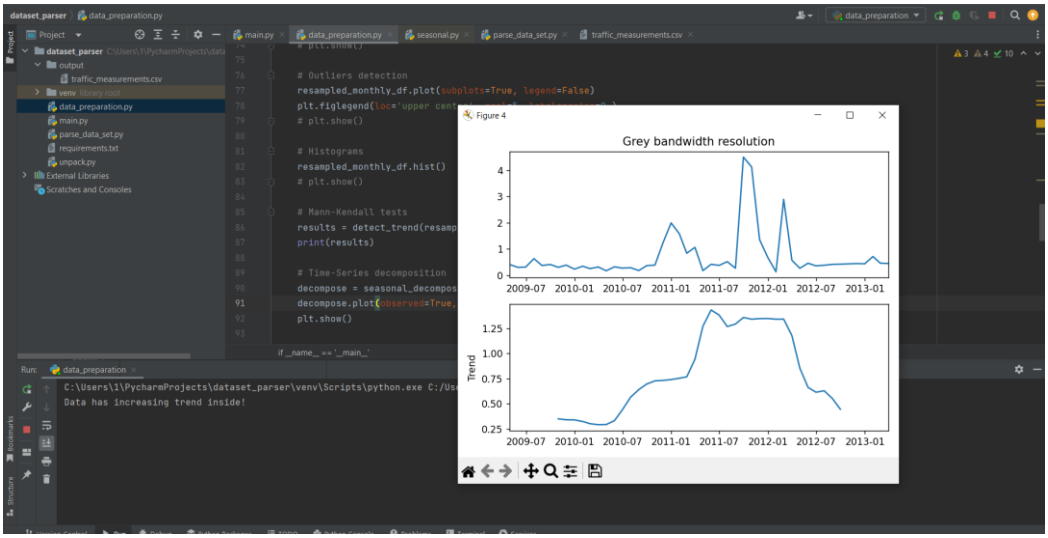

Рис. В.2 Тест 2

Тест 3

1. Перевірка виконання побудови моделі прогнозування.

2. Запуск методу main основного пайтон файлу для процесу моделювання ARIMA-моделі.

3. Отримання результатів виконання моделювання у вигляді логів та графіків.

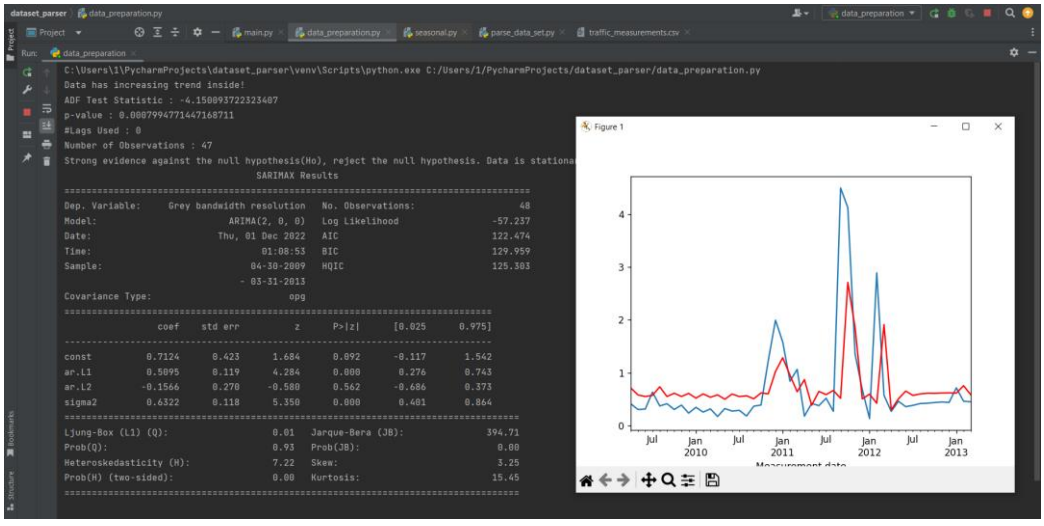

Рис. В.3 Тест 3

Тест 4

1. Перевірка виконання побудови моделі прогнозування з використанням методу градієнтного бустингу.

2. Запуск методу main основного пайтон файлу для процессу моделювання ARIMA-моделі з застосуванням градієнтного бустингу.

3. Отримання результатів виконання моделювання у вигляді логів та графіків.

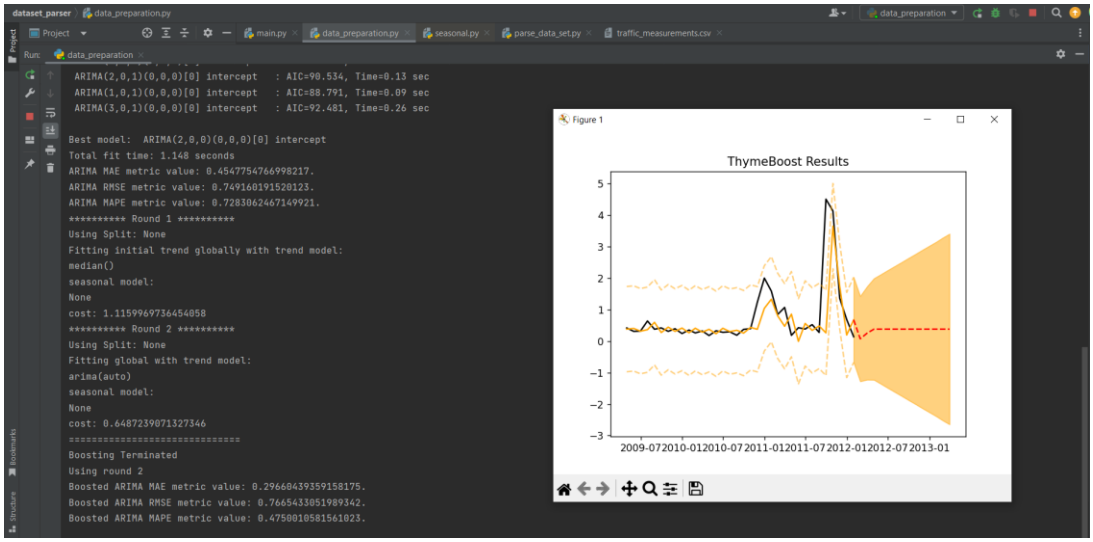

Рис. В.4 Тест 4

Тест вважається пройденим, якщо відбуваються вказані операції і їх відображення у програмному продукті.

Висновки: випробування пройшло успішно, оскільки кожен з тестів показав очікуванні результати.

Виконавець:

студент групи КІ-61, Дорошенко М. І.

Додаток Г

Лістинг програмного коду

Лістинг коду розробленої комп'ютерної моделі прогнозування:

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import pymannkendall
import seaborn as sns
from statsmodels.tsa.seasonal import seasonal_decompose
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.tsa.stattools import adfuller
from typing import Union
import pmdarima as pm
from ThymeBoost import ThymeBoost as tb

def fill_na_values(dataframe):
    for header in dataframe.select_dtypes(exclude=['datetime']).columns.values:
        dataframe[header].fillna(float(dataframe[header].mean()), inplace=True)



def adfuller_test(data):
    result = adfuller(data)
    labels = ['ADF Test Statistic', 'p-value', '#Lags Used', 'Number of Observations']
    for value, label in zip(result, labels):
        print(label+' : '+str(value))
    if result[1] <= 0.05:
        print("Strong evidence against the null hypothesis(Ho), reject the null hypothesis. Data is
stationary!")
    else:
        print("Weak evidence against null hypothesis,indicating it is non-stationary!")



def detect_trend(series: Union[np.ndarray, pd.Series, pd.DataFrame],
```

```
            test: str = 'consensus',
            tests: dict = {
                'hamed_rao': pymannkendall.hamed_rao_modification_test,
                'normal': pymannkendall.original_test,
                'yue_wang': pymannkendall.yue_wang_modification_test
            }):
    trend = 0.
    if test == "consensus":
        results = []
        for t, v in tests.items():
            results.append(v(series))
        for r in results:
            if r.trend == 'decreasing':
                trend -= 1
            elif r.trend == 'increasing':
                trend += 1

    # consensus average
    return interpret_results(float(round(trend / len(tests))))


def interpret_results(trend_value):
    if trend_value > 0:
        return 'Data has increasing trend inside!'
    elif trend_value < 0:
        return 'Data has decreasing trend inside!'
    else:
        return 'Data has no trend!'


if __name__ == '__main__':
    # Read time-series data set
    df = pd.read_csv('output/traffic_measurements.csv', index_col=['Measurement date'],
parse_dates=['Measurement date'])
```

```
    # Replace each null with mean value in specified column
    fill_na_values(df)

    # Resampling data to include only monthly results
    resampled_monthly_df = df.resample('M').mean()
    fill_na_values(resampled_monthly_df)

    # Correlation matrix
    corr = resampled_monthly_df.corr()
    sns.heatmap(corr, annot=True, xticklabels=corr.columns, yticklabels=corr.columns)
    # plt.show()

    # Outliers detection
    resampled_monthly_df.plot(subplots=True, legend=False)
    plt.figlegend(loc='upper center', ncol=5, labelspacing=0.)
    # plt.show()

    # Histograms
    resampled_monthly_df.hist()
    # plt.show()

    # Mann-Kendall tests
    results = detect_trend(resampled_monthly_df['Grey bandwidth resolution'])
    print(results)

    # Time-Series decomposition
    decompose = seasonal_decompose(resampled_monthly_df['Grey bandwidth resolution'],
model='additive')
    decompose.plot(observed=True, seasonal=False, trend=True, resid=False)
    plt.show()

    # Stationarity of time-series to fit ARIMA model
    adfuller_test(resampled_monthly_df['Grey bandwidth resolution'])

    # Data forecasting using ARIMA model
```

```
    y = resampled_monthly_df['Grey bandwidth resolution']

    arima_model = ARIMA(y, order=(2, 0, 0))
    model = arima_model.fit()
    print(model.summary())

    # Actual vs Fitted
    y.plot()
    model.fittedvalues.plot(color='red')
    plt.show()

    test_len = int(len(y) * 0.3)
    al_train, al_test = y.iloc[:-test_len], y.iloc[-test_len:]

    # Fit a simple auto_arima model
    arima = pm.auto_arima(al_train, seasonal=False, trace=True)
    pmd_predictions = arima.predict(n_periods=len(al_test))
    arima_mae = np.mean(np.abs(al_test - pmd_predictions))
    arima_rmse = (np.mean((al_test - pmd_predictions) ** 2)) ** .5
    arima_mape = np.sum(np.abs(pmd_predictions - al_test)) / (np.sum((np.abs(al_test))))
    print(f"ARIMA MAE metric value: {arima_mae}.")
    print(f"ARIMA RMSE metric value: {arima_rmse}.")
    print(f"ARIMA MAPE metric value: {arima_mape}.")

    boosted_model = tb.ThymeBoost(verbose=1)
    output = boosted_model.fit(al_train,
                               trend_estimator='arima',
                               arima_order='auto',
                               global_cost='mse')
    predicted_output = boosted_model.predict(output, len(al_test))
    tb_mae = np.mean(np.abs(al_test - predicted_output['predictions']))
    tb_rmse = (np.mean((al_test - predicted_output['predictions']) ** 2)) ** .5
    tb_mape = np.sum(np.abs(predicted_output['predictions'] - al_test))
(np.sum((np.abs(al_test))))
    print(f"Boosted ARIMA MAE metric value: {tb_mae}.")
```

```
    print(f"Boosted ARIMA RMSE metric value: {tb_rmse}.")
    print(f"Boosted ARIMA MAPE metric value: {tb_mape}.")
    boosted_model.plot_results(output, predicted_output)
```

Лістинг коду для розробленого парсеру даних параметрів трафіку комп'ютерної мережі:

```
import csv
import os
import re
import glob
import time
from datetime import datetime


class MeasurementRow:
    def __init__(self, max_packet_size, sndr_latency, recv_latency, min_packet_spacing,
max_rate, adr_value,
                 grey_bandwidth, date, bandwidth):
        self.max_packet_size = max_packet_size
        self.sndr_latency = sndr_latency
        self.recv_latency = recv_latency
        self.min_packet_spacing = min_packet_spacing
        self.max_rate = max_rate
        self.adr_value = adr_value
        self.grey_bandwidth = grey_bandwidth
        self.date = date
        self.bandwidth = bandwidth

    def get_parameters(self):
        return [self.max_packet_size, self.sndr_latency, self.recv_latency,
                self.min_packet_spacing, self.max_rate, self.adr_value,
                self.grey_bandwidth, self.date, self.bandwidth]

    def is_invalid(self):
```

```
        return self.max_packet_size == NO_DATA and self.sndr_latency == NO_DATA and
self.recv_latency == NO_DATA and \
               self.min_packet_spacing == NO_DATA and self.max_rate == NO_DATA and
self.adr_value == NO_DATA and \
               self.grey_bandwidth == NO_DATA and self.bandwidth == NO_DATA

    def __str__(self):
        return
f"{self.max_packet_size},{self.sndr_latency},{self.recv_latency},{self.min_packet_spacing},{s
elf.max_rate},{self.adr_value},{self.grey_bandwidth},{self.date},{self.bandwidth}"


HEADERS = ['Max packet size (bytes)', 'Send latency (usec)', 'Recv latency (usec)',
           'Min packet spacing (usec)', 'Max rate (Mbps)', 'ADR (Mbps)',
           'Grey bandwidth resolution', 'Measurement date', 'Available bandwidth (Mbps)']

NO_DATA = 'null'


def generate_csv_file():
    with open('traffic_measurements.csv', 'w', newline='', encoding='UTF8') as f:
        csv.writer(f).writerow(HEADERS)
    return f


def write_row(row):
    with open('traffic_measurements.csv', 'a', newline='', encoding='UTF8') as f:
        csv.writer(f).writerow(row.get_parameters())


def parse_logs(data_path):
    measurement_rows = []
    for filename in glob.iglob(data_path + '/' + '**/*.log', recursive=True):
        f = os.path.join(filename)
        with open(f, "r") as f:
```

```
            lines = f.readlines()
            row_to_add = MeasurementRow(
                find_max_packet_size(lines),
                find_sndr_latency(lines),
                find_recv_latency(lines),
                find_min_packet_spacing(lines),
                find_max_rate(lines),
                find_adr_value(lines),
                find_grey_bandwidth(lines),
                find_date(lines, f.name),
                find_bandwidth(lines)
            )
            if not row_to_add.is_invalid():
                write_row(row_to_add)
    return measurement_rows


def find_max_packet_size(lines):
    for line in lines:
        result = re.findall("\s+Maximum\s+packet\s+size\s+::\s+\d+\s+bytes", line)
        if not result:
            continue
        else:
            return re.findall("\d+", result[0].split('::')[1])[0]
    return NO_DATA


def find_sndr_latency(lines):
    for line in lines:
        result = re.findall("send\s+latency\s+@sndr\s+::\s+\d+\s+usec", line)
        if not result:
            continue
        else:
            return re.findall("\d+", result[0].split('::')[1])[0]
    return NO_DATA
```

```
def find_recv_latency(lines):
    for line in lines:
        result = re.findall("recv\s+latency\s+@rcvr\s+::\s+\d+\s+usec", line)
        if not result:
            continue
        else:
            return re.findall("\d+", result[0].split('::')[1])[0]
    return NO_DATA


def find_min_packet_spacing(lines):
    for line in lines:
        result = re.findall("Minimum\s+packet\s+spacing\s+::\s+\d+\s+usec", line)
        if not result:
            continue
        else:
            return re.findall("\d+", result[0].split('::')[1])[0]
    return NO_DATA


def find_max_rate(lines):
    for line in lines:
        result =
re.findall("Max\s+rate\(max_pktsz\/min_time\)\s+::\s+\d{1,5}[\,\.]{1}\d{1,5}Mbps", line)
        if not result:
            continue
        else:
            return re.findall("\d{1,5}[\,\.]{1}\d{1,5}", result[0].split('::')[1])[0]
    return NO_DATA


def find_adr_value(lines):
    for line in lines:
```

```
        result = re.findall("ADR\s+\[\.*\]\s+::\s+\d{1,5}[\,\.]{1}\d{1,5}Mbps", line)
        if not result:
            continue
        else:
            return re.findall("\d{1,5}[\,\.]{1}\d{1,5}", result[0].split('::')[1])[0]
    return NO_DATA


def find_grey_bandwidth(lines):
    for line in lines:
        result = re.findall("Grey\s+bandwidth\s+resolution\s+::\s+\d{1,5}[\,\.]{1}\d{1,5}", line)
        if not result:
            continue
        else:
            return re.findall("\d{1,5}[\,\.]{1}\d{1,5}", result[0].split('::')[1])[0]
    return NO_DATA


def find_date(lines, filename):
    for line in lines:
        result = re.findall("End\s+of\s+measurement:.*", line)
        if not result:
            continue
        else:
            return format_date(re.search("\w{3}\s+\w{3}\s+\d{1,2}\s+\d{2}\:\d{2}\:\d{2}\s+\d{4}",
result[0])[0])
    return extract_date_from_filename(filename)


def format_date(date_str):
    return datetime.strptime(date_str, '%a %b %d %H:%M:%S %Y').strftime('%m/%d/%Y')


def extract_date_from_filename(filename):
    extracted = re.search("\d{4}\d{2}\d{2}", filename)[0]
```

```
    return datetime.strptime(extracted, '%Y%m%d').strftime('%m/%d/%Y')


def find_bandwidth(lines):
    for line in lines:
        result = re.findall("Available\s+bandwidth.*", line)
        if not result:
            continue
        else:
            return re.findall("\d{1,5}[\,\.]{1}\d{1,5}\s+", result[0])[0].replace('(Mbps)', '')
    return NO_DATA


if __name__ == '__main__':
    desktop_path = os.path.join(os.path.join(os.environ['USERPROFILE']), 'Desktop')
    path_to_logs = os.path.join(desktop_path, 'extracted')

    start_time = time.time()

    generated_csv_file = generate_csv_file()
    parse_logs(path_to_logs)

    print(f"New CSV file was successfully generated!\nFile output location:
output/{generated_csv_file.name}")
    print(f"\nGeneration time: {round(time.time() - start_time, 2)} seconds")
```