НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ

Кафедра математичних методів системного аналізу

До захисту допущено:

Завідувач кафедри
_____ Оксана ТИМОЩУК
«___»_____2025 р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Системний аналіз і управління»

спеціальності 124 «Системний аналіз»

на тему: «Інформаційна система «Прогнозування фінансових часових рядів»

Виконав:
студент IV курсу, групи КА-12
Шинкарчук Денис Дмитрович _____

Керівник:
Доцент, к.т.н.
Селін Юрій Миколайович _____

Консультант з економічного розділу:
Доцент кафедри ТПЕ, к.е.н.,
Рощина Надія Василівна _____

Консультант з нормоконтролю:
к.ф.-м.н. Статкевич Віталій Михайлович _____

Рецензент:
Професор кафедри ІСТ, д.т.н.
Корнієнко Богдан Ярославович _____

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____

Київ – 2025 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий інститут прикладного системного аналізу

Кафедра математичних методів системного аналізу

Рівень вищої освіти – перший (бакалаврський)
Спеціальність – 124 «Системний аналіз»
Освітньо-професійна програма «Системний аналіз і управління»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Оксана ТИМОЩУК

«__» травня 2025 р.

ЗАВДАННЯ

на дипломну роботу студенту
Шинкарчуку Денису Дмитровичу

1. Тема роботи «Інформаційна система «Прогнозування фінансових часових рядів», керівник роботи доцент, к.т.н. Селін Ю.М. затверджені наказом по університету від «_26_»____05_____ 2025 р. №1759-с
2. Термін подання студентом роботи 11.06.2025.
3. Вихідні дані до роботи: моделі прогнозування, часові ряди, що описують зміни ВВП країн.
4. Зміст роботи: аналітичний огляд задач прогнозування економічних процесів, застосування авторегресій та нейронних мереж для побудови прогнозів, порівняльний аналіз різних методів прогнозування.
5. Перелік ілюстративного матеріалу: актуальність дослідження, постановка задачі, авторегресії, результати точностей, нейромережі, висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Рощина Н.В., доцент кафедри ТПЕ		

7. Дата видачі завдання

Календарний план

	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Затвердження теми БДР. Ознайомлення зі структурою БДР згідно з Положення про випускну атестацію студентів КПІ ім. Ігоря Сікорського та з державним стандартом України ДСТУ 3008:2015	21.04.2025	виконано
2	Проведений аналіз існуючих підходів, досліджена література, сформульовані етапи роботи, обраний за затвердженою темою та опрацьований набір даних під керівництво керівника БДР. Підготовлений перший розділ дослідження предметної області.	28.04.2025	виконано
3	Робота над математичними основами роботи: алгоритмами МН в цілому, попередня обробка даних, критерії якості тощо.	05.05.2025	виконано
4	Завершення роботи над другим розділом, сформований перший варіант основної частини БДР. Продовжена робота над експериментальною частиною і розробкою програмного продукту	12.05.2025	виконано
5	Доопрацювання основної частини БДР, розробка програмного продукту, аналіз результатів	25.05.2025	виконано
6	Написання функціонально-вартісного аналізу розробленої практичної частини, аналіз результатів, оформлення дипломної роботи	31.05.2025	виконано

Студент Денис ШИНКАРЧУК

Керівник Юрій СЕЛІН

РЕФЕРАТ

Дипломна робота: 135 с.,19 рис.,12 табл., 2 дод., 69 джерел.

ПРОГНОЗУВАННЯ, ВВП, ЧАСОВІ РЯДИ, ARIMA, МАШИННЕ НАВЧАННЯ, PYTHON, ФІНАНСОВІ ІНФОРМАЦІЙНІ СИСТЕМИ.

Об’єкт дослідження – фінансові часові ряди, зокрема історичні дані валового внутрішнього продукту (ВВП) країн світу з 1960 по 2022 рік.

Предмет дослідження – інформаційні моделі прогнозування часових рядів ВВП та засоби їх реалізації.

Мета роботи – розробити гнучку інформаційну систему на мові Python для прогнозування ВВП на основі реальних часових рядів, застосовуючи як класичні статистичні методи (ARIMA), так і моделі машинного навчання.

Методи дослідження – математичне моделювання, статистичний аналіз, авторегресійні моделі, алгоритми машинного навчання, лінійна регресія, візуалізація даних.

Актуальність – в умовах геополітичної нестабільності, глобальних економічних змін і пандемічних наслідків критично важливо мати точні прогнози основних макроекономічних показників. Побудова адаптивних моделей прогнозування дозволяє органам державного управління, інвесторам і підприємствам приймати обґрунтовані рішення.

Результати роботи – реалізовано інформаційну систему, що дозволяє імпортувати, очищати, обробляти та аналізувати фінансові часові ряди ВВП. Побудовано сім моделей прогнозування, результати яких порівнювалися за метриками MAE, MSE, RMSE та MAPE.

Шляхи подальшого розвитку предмету дослідження – інтеграція моделі LSTM для покращення довгострокових прогнозів, розширення інформаційної системи до інших економічних індикаторів, впровадження автоматичної оцінки.

ABSTRACT

Diploma work: 135 pages, 19 figures, 12 tables, 2 appendices, 69 references.

FORECASTING, GDP, TIME SERIES, ARIMA, MACHINE LEARNING, PYTHON, FINANCIAL INFORMATION SYSTEMS

Object of study – financial time series, particularly historical GDP data of countries from 1960 to 2022.

Subject of study – information models for GDP time series forecasting and tools for their implementation.

Aim of the thesis – to develop a flexible information system in Python for forecasting GDP based on real-world time series using both classical statistical methods (ARIMA) and modern machine learning models.

Methods of research – mathematical modeling, statistical analysis, autoregressive models, machine learning algorithms, linear regression, and data visualization.

Relevance – in the context of geopolitical instability, global economic shifts, and post-pandemic recovery, it is critically important to obtain accurate forecasts of key macroeconomic indicators.

Results of the thesis – an information system was implemented that allows for importing, cleaning, processing, and analyzing GDP time series data. Seven forecasting models were built and evaluated using MAE, MSE, RMSE, and MAPE metrics.

Directions for further research – integration of LSTM models to improve long-term forecasting accuracy, extension of the information system to cover other economic indicators, implementation of automated model performance assessment, and development of a more user-friendly interface.

ЗМІСТ

ВСТУП

У сучасному світі фінансові ринки відіграють ключову роль у забезпеченні стабільності та розвитку економічних систем. З огляду на стрімкий розвиток інформаційних технологій та зростаючий обсяг фінансових даних, що генеруються в режимі реального часу, постає гостра потреба у створенні ефективних інформаційних систем, здатних здійснювати аналіз та прогнозування фінансових часових рядів. Ці часові ряди відображають поведінку складних економічних процесів, таких як зміни валового внутрішнього продукту (ВВП), інфляційних показників, валютних курсів, цін на акції та інші фінансові індикатори.

Фінансові часові ряди, на відміну від звичайних статистичних даних, мають динамічну природу, що характеризується сезонністю, циклічністю, наявністю трендів та стохастичних компонент. Прогнозування таких рядів є надзвичайно складною задачею, оскільки потребує урахування багатьох факторів, що впливають на економічні процеси, включаючи як внутрішні, так і зовнішні економічні, політичні та соціальні чинники. Проте саме ці труднощі зумовлюють високу наукову та практичну цінність відповідних досліджень.

Зокрема, у сфері прогнозування ВВП як одного з основних макроекономічних показників, достовірні моделі прогнозування дозволяють приймати обґрунтовані рішення на рівні державної політики, планування бюджету, інвестиційної стратегії тощо. Таким чином, розвиток методів прогнозування ВВП з використанням сучасних інформаційних систем стає **актуальною задачею** як у теоретичному, так і в прикладному контексті.

Наукова новизна даного дослідження полягає в поєднанні класичних методів статистичного аналізу часових рядів з інструментами машинного навчання, що дозволяє підвищити точність прогнозів і розширити сферу застосування інформаційної системи. Крім того, у роботі запропоновано реалізацію гнучкої інформаційної системи, побудованої на мові програмування Python, яка дозволяє досліджувати та порівнювати різні моделі прогнозування

на реальному фінансовому датасеті – історичних значеннях ВВП за країнами світу з 1960 по 2022 рік.

Об’єктом дослідження у даній роботі є фінансові часові ряди, зокрема динаміка валового внутрішнього продукту (ВВП) у різних країнах світу.

Предметом дослідження виступають інформаційні моделі та методи прогнозування, а також програмні інструменти для їхньої реалізації.

Особлива увага приділяється методам обробки, аналізу та прогнозування часових рядів, серед яких: ARIMA, SARIMA, моделі нейронних мереж (LSTM) та методи машинного навчання.

Метою даної наукової **роботи** є розробка та реалізація інформаційної системи, здатної здійснювати прогнозування фінансових часових рядів на основі сучасних математичних моделей і засобів обчислювальної техніки. Така система повинна забезпечити гнучкість у використанні різних моделей, надавати зручний інтерфейс для аналізу результатів і бути застосовною до різних типів фінансових показників.

Для досягнення поставленої мети необхідно вирішити такі **завдання**:

- провести аналіз предметної області та наявних підходів до прогнозування фінансових часових рядів;

- визначити класи моделей, що найкраще підходять для прогнозування ввп;

- побудувати формальну математичну постановку задачі прогнозування;

- розробити інформаційну систему для реалізації моделей аналізу та прогнозування;

- провести експериментальну оцінку точності прогнозування з використанням обраного датасету;

- здійснити візуалізацію результатів і сформулювати практичні висновки щодо ефективності запропонованого підходу.

Методологічну основу дослідження становлять математичні методи аналізу часових рядів, теорія ймовірностей і математична статистика, а також алгоритми машинного навчання й елементи теорії нейронних мереж. Для

реалізації інформаційної системи використано мову програмування Python, а також відповідні бібліотеки – pandas, statsmodels, scikit-learn, keras, matplotlib та інші.

Актуальність цього дослідження обумовлена необхідністю впровадження точних і адаптивних систем прогнозування у сферу економічного планування, яка постійно стикається з викликами, пов’язаними з глобалізацією, нестабільністю ринків і технологічними змінами. Особливо в умовах постпандемічного відновлення та геополітичної нестабільності, які значно впливають на економіки різних країн, важливо мати інструменти, що дозволяють швидко реагувати на зміни.

Таким чином, дана наукова робота спрямована на вирішення актуальної задачі створення інформаційної системи для прогнозування фінансових часових рядів, що має як наукову, так і прикладну цінність. Результати дослідження можуть бути використані у сферах макроекономічного планування, фінансового аналізу, освітнього процесу, а також слугувати основою для подальших досліджень у галузі інтелектуального аналізу фінансових даних.

1 ПРЕДМЕТНА ОБЛАСТЬ І ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз сучасного стану досліджень у сфері прогнозування фінансових часових рядів

У сучасній науковій літературі прогнозування фінансових часових рядів посідає одне з провідних місць серед задач економетрики, статистики, фінансової математики та прикладного машинного навчання. Дана галузь знань об’єднує численні теоретичні підходи та прикладні методи, які дозволяють аналізувати поведінку складних економічних систем у часі та здійснювати оцінювання майбутніх значень фінансових показників на основі їх попередніх спостережень. Актуальність цього напряму підтверджується зростаючим попитом на аналітичні інструменти для підтримки прийняття рішень в умовах невизначеності на фінансових ринках.

Фінансові часові ряди являють собою послідовності числових значень, які відображають зміну фінансових показників у часі з певною періодичністю (день, місяць, рік тощо). Класичними прикладами є котирування цін на акції, обмінні курси валют, рівень інфляції, валовий внутрішній продукт, обсяг державного боргу тощо. Всі ці індикатори є ключовими елементами у процесі фінансового аналізу та прийняття рішень. Прогнозування таких рядів дозволяє не лише виявити закономірності у поведінці економічних параметрів, але й моделювати їх розвиток у майбутньому, що є надзвичайно важливим для планування, інвестування, кредитування, державного регулювання тощо [1].

Ранні дослідження у сфері прогнозування часових рядів базувались на класичних статистичних підходах. Так, у 1970-х роках активно розвивалися авторегресійні моделі, зокрема AR (autoregressive), MA (moving average), ARMA (autoregressive moving average), а пізніше – ARIMA (autoregressive integrated moving average), яку вперше систематизував Джордж Бокс разом з Гвіл'ямом Дженкінсом [2]. Ці моделі дозволили формалізувати процес прогнозування часових рядів із чіткими математичними критеріями та стали

основою для багатьох практичних застосувань у фінансовій галузі.

У подальшому розвиток економетрики та збільшення обчислювальних можливостей призвели до появи більш складних моделей, здатних моделювати не лише лінійні, але й нелінійні залежності. Серед таких варто відзначити моделі ARCH (Autoregressive Conditional Heteroskedasticity) та GARCH (Generalized ARCH), які особливо ефективні при моделюванні фінансових рядів з високою волатильністю, зокрема цін на біржові активи [3]. Вказані підходи застосовуються в задачах управління ризиками, побудови фінансових деривативів, аналізу динаміки інфляції, обсягів торгівлі тощо.

Починаючи з 2010-х років, у наукових дослідженнях усе частіше почали застосовуватись методи штучного інтелекту, зокрема машинного навчання (ML) та глибокого навчання (DL). Ці підходи відкрили нові горизонти в моделюванні складних взаємозв'язків у даних, які не завжди вдається описати лінійними статистичними моделями. Одним із найбільш популярних напрямів стало використання рекурентних нейронних мереж (RNN) та їхніх модифікацій, зокрема LSTM (Long Short-Term Memory), GRU (Gated Recurrent Unit), які демонструють високу ефективність у задачах передбачення динамічних процесів, включаючи фінансові часові ряди [4].

У контексті глобалізованих фінансових ринків особливу увагу дослідники звертають на багатоваріантне прогнозування (multivariate forecasting), яке враховує не лише часовий контекст змін одного показника, а й вплив суміжних факторів – наприклад, взаємозв’язок між ВВП, рівнем безробіття, інфляцією, процентною ставкою, обсягом експорту тощо. Такий підхід дозволяє підвищити точність прогнозів і сформувати більш адекватні економічні сценарії [5].

Не менш важливою є роль сезонності та циклічності у фінансових часових рядах. Для моделювання сезонних компонентів активно застосовуються SARIMA-моделі (Seasonal ARIMA), які включають додаткові параметри для врахування періодичних коливань. Подібні моделі ефективні, наприклад, у задачах прогнозування ВВП, де щорічна сезонність (наприклад, зростання споживчої активності в грудні) відіграє помітну роль [2].

Наукові дослідження у сфері прогнозування фінансових рядів постійно інтегрують новітні досягнення інших галузей знань, зокрема обробки природної мови (NLP), графових мереж, трансформерів, які спочатку були розроблені для задач перекладу або розпізнавання зображень, але згодом були адаптовані й до фінансового аналізу. Наприклад, трансформерні моделі (Transformer-based models), такі як Temporal Fusion Transformer (TFT), демонструють багатообіцяючі результати при обробці складних часових рядів із багатьма змінними [6].

У межах міждисциплінарних досліджень зростає популярність гібридних моделей, які комбінують класичні статистичні підходи з методами глибокого навчання. Наприклад, моделі типу ARIMA-LSTM, де прогнозування тренду здійснюється класичною моделлю, а залишковий ряд обробляється LSTM-мережею, дозволяють суттєво покращити якість прогнозу за рахунок синергії двох підходів [7].

Сучасний стан досліджень у сфері прогнозування фінансових часових рядів відзначається високим рівнем розвитку як у теоретичному, так і в прикладному аспекті. Сучасна наука має у своєму арсеналі широкий спектр методів – від класичних авторегресійних моделей до складних глибоких нейронних мереж і гібридних підходів. Водночас, вибір конкретного методу значною мірою залежить від природи даних, мети прогнозування, доступності обчислювальних ресурсів та вимог до точності.

Поряд із розвитком математичних і алгоритмічних методів прогнозування фінансових часових рядів, спостерігається інтенсивне впровадження інструментів інформаційних систем, здатних автоматизувати процес обробки даних, побудови моделей і генерації прогнозів. Ці системи активно використовуються не лише в науковому середовищі, але й у комерційному секторі – банках, аналітичних агентствах, фінансових службах підприємств, державних інституціях. Ключову роль у цьому процесі відіграє розвиток програмних платформ, мов програмування та обчислювальних інфраструктур.

Одним із найбільш популярних інструментів для розробки інформаційних

систем фінансового аналізу є мова програмування Python. Завдяки наявності великої кількості спеціалізованих бібліотек (наприклад, pandas, statsmodels, scikit-learn, prophet, tensorflow, keras), Python дозволяє швидко імплементувати різноманітні моделі прогнозування часових рядів, включно з класичними статистичними моделями та методами глибокого навчання. Особливістю Python як платформи є його відкритість, багатофункціональність і активна підтримка з боку наукової спільноти [1].

Серед сучасних готових програмних рішень, що дозволяють реалізовувати прогнозування часових рядів, варто відзначити такі інструменти:

Facebook Prophet – бібліотека, розроблена командою Meta для побудови точних прогнозів із урахуванням сезонних коливань, трендів, свят та змін у поведінці користувачів. Вона показала свою ефективність у сфері фінансового прогнозування завдяки зручному інтерфейсу та високій точності [8].

AutoTS, NeuralProphet, Darts – новітні бібліотеки з автоматичним підбором моделей, які дозволяють здійснювати експерименти з десятками варіантів конфігурацій без глибокого занурення у математичні деталі. Це відкриває двері для нетехнічних фахівців до практичного застосування прогнозних методів [9].

Google Cloud AI Forecasting, Amazon Forecast, Azure Time Series Insights – комерційні хмарні рішення з інтерфейсами для побудови фінансових прогнозів із використанням масштабованої інфраструктури.

ВВП має стійку річну сезонність, часто демонструє довгострокові тенденції (тренди), але при цьому є досить інертним, тобто його зміни відбуваються поступово. Це зумовлює використання моделей, орієнтованих на середньо- та довгостроковий прогноз, таких як ARIMA, SARIMA, Prophet, а також нейронних мереж, здатних моделювати залежності в довгих часових рядах [6].

На практиці також застосовується так званий ensemble modeling – метод, що передбачає побудову кількох моделей і поєднання їхніх прогнозів за допомогою агрегуючих функцій або моделей другого рівня (stacking). Це

дозволяє компенсувати слабкості окремих моделей і суттєво підвищити точність фінансових прогнозів [10].

Слід зауважити, що у зарубіжній науковій літературі активно розвиваються підходи до інтерпретованого машинного навчання (Explainable AI, XAI), що дозволяють не лише прогнозувати, але й аналізувати, чому модель приймає ті чи інші рішення. Це особливо важливо у фінансовій сфері, де аналітики повинні пояснити свої висновки замовникам або регуляторам [6].

Аналіз сучасного стану досліджень у сфері прогнозування фінансових часових рядів свідчить про значний прогрес у розвитку як математичних моделей, так і прикладних інформаційних систем. Разом із тим, існують виклики, що вимагають подальшого дослідження – зокрема в напрямах стабільності моделей до структурних змін, адаптації до нових типів даних, побудови інтерпретованих моделей і застосування хмарних технологій для великомасштабного прогнозування.

1.2 Огляд інформаційних систем для роботи з фінансовими даними

Сучасні фінансові ринки характеризуються надзвичайною складністю, високою динамікою та великим обсягом інформації, яка потребує обробки у реальному часі. У зв’язку з цим особливого значення набувають інформаційні системи, орієнтовані на аналіз та прогнозування фінансових даних. Ці системи є невід’ємною частиною інфраструктури фінансових установ, органів державного управління, міжнародних організацій, інвестиційних компаній, банківських установ та підприємств реального сектору економіки. Їх метою є надання точних, своєчасних і обґрунтованих рішень на основі обробки великих масивів фінансової інформації.

Інформаційна система у сфері фінансів – це сукупність методів, засобів та технологій, яка забезпечує збирання, зберігання, обробку, аналіз, прогнозування та візуалізацію фінансових даних. Її архітектура зазвичай

включає кілька рівнів: рівень даних (сховище фінансової інформації), рівень аналітики (алгоритми аналізу та прогнозування), рівень подання результатів (інтерфейс користувача або автоматизовані звіти). Ефективність такої системи залежить від якості моделей прогнозування, гнучкості у налаштуванні параметрів, підтримки сучасних форматів даних та інтеграції з іншими інформаційними ресурсами [11].

Існує велика кількість програмних рішень та платформ, які реалізують вказані функції. Їх умовно можна поділити на три основні групи:

- корпоративні (промислові) системи прогнозування;
- науково-дослідницькі платформи та мови програмування;
- хмарні сервіси з вбудованими інструментами аналітики.

Корпоративні системи охоплюють такі продукти, як SAS Forecast Server, IBM SPSS Forecasting, SAP Analytics Cloud, Oracle Financial Services Analytical Applications. Ці системи характеризуються високим рівнем інтеграції з базами даних, зручним графічним інтерфейсом, автоматизованими засобами побудови прогнозів, функціоналом побудови сценаріїв та управління ризиками. Наприклад, SAS підтримує ARIMA, UCM, експоненційне згладжування, нейронні мережі та дозволяє обирати найкращу модель на основі крос-валідації [12].

SPSS Forecasting, у свою чергу, забезпечує побудову прогнозів на основі візуального аналізу часових рядів та інтеграцію з іншими модулями IBM SPSS для багатовимірного статистичного аналізу. Проте такі продукти є комерційними, що обмежує їх доступність для наукових досліджень та навчальних цілей [13].

Універсальним інструментом для науковців і розробників є мови програмування та бібліотеки відкритого коду. Найпопулярнішими серед них є Python та R, які забезпечують величезний спектр бібліотек для статистичного моделювання, машинного навчання та побудови моделей часових рядів. Зокрема, у Python використовуються бібліотеки:

- statsmodels – реалізація класичних моделей ARIMA, VAR, SARIMA;

- scikit-learn – стандартні методи машинного навчання;

- prophet – спеціалізований пакет для фінансового прогнозування з урахуванням сезонності та свят;

- tensorflow та keras – глибокі нейронні мережі для складних багатовимірних задач [1, 2, 4].

Подібні можливості має і мова R, де існують пакети forecast, tseries, caret, які широко застосовуються в економетриці та фінансовій статистиці. Перевагою таких платформ є відкритість, широке ком'юніті, доступ до академічних розробок, можливість інтеграції з сучасними інструментами візуалізації даних (matplotlib, ggplot2, seaborn).

Упродовж останніх років активно розвиваються хмарні сервіси, що дозволяють виконувати складні фінансові обчислення на віддалених серверах з використанням високопродуктивних обчислювальних ресурсів. Це, зокрема, Google Cloud AI Platform, Amazon Forecast, Microsoft Azure Machine Learning Studio. Їхньою перевагою є масштабованість, інтерфейс, орієнтований на користувача, можливість швидкого впровадження моделей у промислове середовище. Наприклад, Amazon Forecast базується на моделі DeepAR, яка використовує LSTM для прогнозування на основі великих обсягів часових рядів [14].

Також не слід забувати про спеціалізовані інформаційні платформи для збирання та обробки фінансових даних, такі як Bloomberg Terminal, Thomson Reuters Eikon, Quandl, Yahoo Finance API, World Bank API. Ці інструменти забезпечують не лише доступ до історичних фінансових даних, але й актуальні макроекономічні показники, які можуть бути безпосередньо використані у прогнозних моделях [15].

Саме можливість швидко та інтерактивно дослідити поведінку часових рядів, побудувати графіки, виявити закономірності – одна з ключових вимог до сучасних систем. Серед популярних інструментів візуалізації – matplotlib, plotly, seaborn, а також інтерактивні панелі типу Power BI, Tableau, Google Data Studio.

У результаті аналізу наявних рішень можна зробити висновок, що сучасні інформаційні системи для роботи з фінансовими часовими рядами представляють собою багатофункціональні інструменти, які об’єднують можливості збирання, обробки, аналізу, прогнозування та візуалізації даних. Вибір конкретного рішення залежить від потреб користувача, обсягу даних, рівня технічної підготовки та фінансових можливостей. Однак у контексті наукової роботи, де ключовими є гнучкість, масштабованість і відкритість коду, найбільш доцільним є використання мови Python та бібліотек з відкритим кодом.

1.3 Класифікація існуючих методів прогнозування

Сучасна наука пропонує велику кількість методів прогнозування часових рядів, що застосовуються у фінансовій сфері. Вони різняться як за математичною природою, так і за глибиною обробки даних, ступенем складності, вимогами до обчислювальних ресурсів, точністю та інтерпретованістю результатів. Вибір методу безпосередньо залежить від природи даних, їх обсягу, наявності трендів і сезонності, а також від типу прогнозу (коротко- чи довгостроковий, одно- чи багатовимірний).

Усі методи прогнозування фінансових часових рядів можна умовно поділити на три великі класи:

- класичні статистичні методи;
- методи машинного навчання;
- гібридні та інтегровані моделі.

Ця класифікація дозволяє краще зрозуміти сильні та слабкі сторони кожного підходу, а також визначити їх доцільність у конкретному контексті.

1.3.1 Класичні статистичні методи

Ці методи базуються на строгих математичних припущеннях про лінійність, стаціонарність та нормальний розподіл залишків. Їх головною перевагою є теоретична обґрунтованість, простота інтерпретації та висока точність у випадках, коли часовий ряд не містить сильних нелінійних взаємозв'язків. У таблиці 1.1 наведено основні класичні статистичні методи.

Таблиця 1.1 – Класичні статистичні методи

Метод	Короткий опис	Основні переваги	Основні обмеження
AR (авторегресія)	Прогноз на основі попередніх значень самого ряду	Простота, швидкість обчислень	Потребує стаціонарності
MA (ковзне середнє)	Прогноз ґрунтується на залишках від попередніх прогнозів	Добре працює з шумом	Не враховує тенденцій
ARMA	Комбінація AR і MA	Гнучкість для стаціонарних рядів	Не підходить для нестаціонарних рядів
ARIMA	Розширення ARMA для нестаціонарних даних	Підходить для трендових даних	Складність ідентифікації порядку
SARIMA	ARIMA з урахуванням сезонності	Ефективна для сезонних рядів	Багато параметрів
ARCH / GARCH	Моделі для рядів із змінною волатильністю	Добре описують фінансові коливання	Висока чутливість до початкових умов

[1, 2, 3, 5]

1.3.2 Методи машинного навчання

Машинне навчання дозволяє моделювати складні, часто нелінійні залежності між змінними. Ці методи не завжди потребують стаціонарності ряду та можуть навчатися без жорстких припущень щодо розподілу даних. Їх ефективність особливо помітна на великих обсягах даних. У таблиці 1.2 наведено основні методи машинного навчання.

Таблиця 1.2 – Методи машинного навчання

Метод	Короткий опис	Основні переваги	Основні обмеження
Лінійна регресія	Модель залежності між змінною і її факторами	Простота, інтерпретованість	Погано працює з нелінійними залежностями
Рішення дерев	Дерево, що розбиває дані за правилами розгалуження	Інтерпретованість	Схильність до перенавчання
Випадковий ліс (Random Forest)	Ансамбль дерев, що агрегує прогнози	Висока точність	Важче інтерпретувати
Градієнтний бустинг	Послідовна побудова моделей для зменшення помилок	Надзвичайна точність	Повільне навчання, складність
K-найближчих сусідів (KNN)	Прогноз ґрунтується на схожих прикладах	Не потребує припущень	Повільність при великих наборах

[4, 6, 10]

1.3.3 Нейронні мережі та гібридні моделі

Цей клас методів найновіший і найпотужніший. Він включає штучні нейронні мережі (ANN), рекурентні мережі (RNN), зокрема LSTM та GRU, трансформерні моделі, а також гібридні підходи, що поєднують класичні моделі з глибоким навчанням. У таблиці 1.3 наведено основні нейронні мережі.

Таблиця 1.3 – Нейронні мережі та гібридні моделі

Метод	Короткий опис	Основні переваги	Основні обмеження
ANN	Багатошарові нейронні мережі для загального прогнозування	Потужність при складних патернах	Потребують великих обсягів даних
RNN	Мережі, що враховують часову послідовність даних	Урахування послідовності	Важко навчаються, проблема градієнтного спаду
LSTM	Модифіковані RNN з довготривалою пам’яттю	Добре прогнозують довгі залежності	Велика кількість параметрів
GRU	Спрощена версія LSTM	Менше обчислень	Може втрачати частину інформації
ARIMA + ML / LSTM	Комбіновані моделі: тренд – класично, залишки – ML	Синергія підходів	Складність реалізації
Temporal Transformer	Модель уваги для часових рядів з багатьма змінними	Точність, масштабованість	Велика обчислювальна складність

[4, 6, 7, 10]

Кожна з наведених груп методів має свої сфери застосування. Так, ARIMA та SARIMA добре працюють у задачах із чітко вираженою сезонністю та лінійними трендами, що часто має місце при моделюванні ВВП. Натомість LSTM та глибокі мережі демонструють вищу ефективність у задачах довгострокового прогнозування із сильними нелінійними залежностями, наприклад, при аналізі поведінки фондового ринку або криптовалют. Гібридні моделі дозволяють об’єднати кращі риси обох підходів і дедалі частіше впроваджуються у практику.

Варто зазначити, що підвищення точності прогнозів не завжди означає підвищення їхньої цінності. Інтерпретованість результатів, надійність моделі, адаптивність до нових умов – це ті критерії, що часто важать не менше, ніж похибка прогнозу. У цьому контексті класичні моделі, незважаючи на обмеження, залишаються актуальними, особливо у сфері макроекономічного аналізу та прогнозування показників державного рівня.

Отже, класифікація методів прогнозування дає змогу сформувати обґрунтований вибір моделей у процесі створення інформаційної системи. Для наукової роботи, орієнтованої на прогнозування валового внутрішнього продукту, доцільно використовувати комбінацію статистичних моделей (наприклад, SARIMA) та нейромереж (наприклад, LSTM), що забезпечить баланс між теоретичною обґрунтованістю та практичною точністю прогнозу.

1.4 Виявлення недоліків і обмежень діючих рішень

Попри значний прогрес у розробці методів прогнозування фінансових часових рядів та створенні відповідних інформаційних систем, на сьогодні існує низка об’єктивних обмежень і недоліків, які суттєво впливають на точність прогнозів, адаптивність моделей до зміни зовнішнього середовища, інтерпретованість результатів і загальну ефективність практичного застосування. У цьому підрозділі проаналізуємо найбільш поширені проблеми

та виклики, з якими стикаються як дослідники, так і практики.

Одним із ключових недоліків є обмежена здатність моделей до адаптації у випадку структурних змін у часових рядах. Наприклад, події типу глобальних економічних криз, пандемій, збройних конфліктів чи інших макроекономічних шоків здатні кардинально змінити поведінку фінансових ринків і економічних показників. Класичні моделі (ARIMA, SARIMA, GARCH), зазвичай побудовані на припущенні стаціонарності або умовної стаціонарності, не можуть адекватно відреагувати на подібні переломні моменти [1, 2]. Навіть складні нейронні мережі, навчені на історичних даних, мають схильність до “катастрофічного забування” нових патернів, якщо не виконано процедур регулярного донавчання.

Наступна проблема – перенавчання моделей (overfitting). Це явище особливо характерне для методів машинного навчання, зокрема при використанні глибоких нейронних мереж або ансамблевих моделей. У таких випадках модель дуже точно відтворює навчальні дані, однак втрачає здатність до узагальнення і генерує низькоякісні прогнози на нових, раніше не спостережених даних [17]. Причиною цього часто є недостатній обсяг навчальної вибірки або її надмірна “зашумленість”, що особливо характерно для фінансових даних з високою волатильністю.

Ще однією важливою проблемою є складність у підготовці даних та передобробці часових рядів. Практика показує, що якість прогнозу на 70–80% залежить від якості вхідних даних. При роботі з фінансовими часовими рядами аналітики стикаються з такими проблемами, як пропущені значення, наявність викидів (аномалій), різна періодичність записів (щоденно, щотижнево, щоквартально), а також необхідність узгодження валют, індексів або економічних агрегатів [11]. Усі ці аспекти потребують значної частки ручної роботи, що знижує ефективність автоматизації та обмежує масштабованість інформаційних систем.

Ще одним викликом є низька інтерпретованість складних моделей, зокрема нейронних мереж. У фінансовій сфері важливо не лише отримати

прогноз, а й розуміти, чому саме модель дійшла такого висновку. Це особливо актуально у банківській сфері, страхуванні, державному плануванні, де регулятори або інвестори вимагають прозорості обґрунтувань. Методи explainable AI (XAI), такі як LIME, SHAP або attention-механізми, частково розв'язують цю проблему, проте інтеграція цих методів у повноцінні інформаційні системи ще перебуває на стадії активного дослідження [6, 18].

Не менш важливою є проблема вибору моделі. У наявності безліч підходів (ARIMA, Prophet, LSTM, GRU, LightGBM, XGBoost тощо), однак вибір оптимального методу часто є задачою емпіричного підбору, що вимагає глибоких знань у сфері статистики та машинного навчання. Автоматизовані фреймворки, які здійснюють підбір моделей, такі як AutoML або AutoTS, частково вирішують цю проблему, однак усе ще не позбавлені обмежень: зокрема, вони не завжди правильно обробляють специфічні особливості макроекономічних рядів [9].

Деякі системи можуть мати упередження, пов'язані із навчальними даними, або бути чутливими до маніпуляцій з боку зацікавлених сторін. У випадках, коли йдеться про прогнозування ВВП або інфляції, результати прогнозів можуть суттєво впливати на громадську думку, політичні рішення або ділову активність. Тому важливо забезпечити прозорість, надійність і реплікабельність результатів таких інформаційних систем [18].

Ще один аспект – проблема обчислювальної складності. Деякі моделі, особливо трансформери та ансамблеві мережі, потребують значних обчислювальних ресурсів, що унеможливлює їх використання у реальному часі або на пристроях з обмеженою продуктивністю. Це обмежує їхнє застосування в оперативному управлінні або в системах з великим потоком даних (наприклад, у фінансовому трейдингу чи моніторингу в реальному часі) [6].

Хоча сучасні моделі прогнозування фінансових часових рядів демонструють високу точність у лабораторних умовах, їх широке практичне впровадження наштовхується на низку викликів. Серед них: складність роботи з реальними даними, обмеження моделей щодо адаптивності, складність

інтерпретації результатів, проблема перенавчання, а також етичні й технічні ризики. Ці аспекти потребують подальшого дослідження, а також розвитку нових підходів до побудови більш адаптивних, пояснюваних і надійних інформаційних систем.

1.5 Формальна та змістовна постановка задачі

1.5.1 Формальна постановка задачі

Формальна постановка задачі прогнозування фінансових часових рядів є ключовим етапом, що визначає математичну структуру проблеми, дозволяє чітко сформулювати вхідні та вихідні дані, встановити критерії якості та межі допустимих рішень. У контексті цієї роботи – де об’єктом аналізу виступає валовий внутрішній продукт (ВВП) країн світу – завдання прогнозування можна розглядати як задачу регресійного типу, де потрібно знайти функцію, що апроксимує часову залежність змінної на основі її історичних значень.

Позначимо часовий ряд, що описує динаміку ВВП певної країни, як послідовність скалярних значень

$$Y = \{y_t\}_{t=1}^{T}, \tag{1.1}$$

де $y_t \in \mathbb{R}$ – значення ВВП у момент часу t;

T – кількість моментів спостереження.

Завдання полягає у побудові моделі $\mathcal{M}$, яка для заданого вектору історичних значень $\{y_{t-k},\ldots,y_t\}$ прогнозує значення $y_{t+1}, y_{t+2},\ldots,y_{t+h}$, де $h \in \mathbb{N}$ – горизонт прогнозування

$$\hat{y}_{t+h} = \mathcal{M}(y_t, y_{t-1},\ldots,y_{t-k+1}), \tag{1.2}$$

де $\hat{y}_{t+h}$ – прогнозоване значення;

k – кількість попередніх значень (розмір вікна).

У загальному випадку модель прогнозування часових рядів можна представити у вигляді

$$\hat{y}_{t+1} = f(y_t, y_{t-1}, \ldots, y_{t-k+1}; \theta), \tag{1.3}$$

де f – функціональна залежність, параметризована вектором параметрів θ.

Тип функції f визначається обраною моделлю: для ARIMA це буде лінійна комбінація лагів і залишків, для нейронної мережі – композиція нелінійних функцій, для регресійної моделі – вагова сума з регуляризацією.

Формально завдання прогнозування фінансового часового ряду зводиться до мінімізації функції втрат, яка вимірює відхилення прогнозованого значення від істинного. Однією з найпоширеніших є квадратична функція втрат (MSE – Mean Squared Error)

$$\mathcal{L}(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{1.4}$$

або, для більш стійких до викидів моделей, можуть застосовуватися MAE (Mean Absolute Error)

$$\mathcal{L} = \frac{1}{n}\sum|y_i - \hat{y}_i|, \tag{1.5}$$

або MAPE (Mean Absolute Percentage Error)

$$\mathcal{L} = \frac{1}{n}\sum\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\%, \tag{1.6}$$

яка дозволяє враховувати відносну похибку.

У випадку багатоваріантного прогнозу, тобто коли розглядаються залежності між кількома макроекономічними показниками одночасно, задача набуває вигляду векторного прогнозування

$$\mathbf{Y}_t = [y_t^{(1)}, y_t^{(2)}, \ldots, y_t^{(m)}]^T, \tag{1.7}$$

де m – кількість змінних (наприклад, ВВП, інфляція, безробіття), і відповідна функція прогнозування має враховувати міжзмінні зв'язки

$$\mathbf{Y}_{t+1} = f(\mathbf{Y}_t, \mathbf{Y}_{t-1}, \ldots, \mathbf{Y}_{t-k+1}; \theta). \tag{1.8}$$

Також у постановці задачі необхідно враховувати наявність сезонності. Якщо фінансовий ряд містить періодичні коливання (наприклад, щорічне зростання ВВП у певному кварталі), модель має бути здатна виявити та врахувати ці закономірності. У класичних моделях (SARIMA) сезонність враховується через додаткові лаги, кратні періоду. В глибокому навчанні – через включення ознак-календарів (індикаторів часу), таких як "квартал", "рік", "місяць".

Формальна постановка задачі дозволяє звести складну економічну проблему до чітко структурованої математичної моделі, яка може бути реалізована алгоритмічно в рамках інформаційної системи.

1.5.2 Змістовна постановка задачі

Поряд із формальною постановкою, яка визначає математичну структуру задачі, важливо також сформулювати змістовну постановку, що передбачає розкриття економічного, прикладного та методологічного контексту завдання. Такий підхід дозволяє не лише чітко розуміти, «що» і «як» прогнозується, але й навіщо це робиться, які цілі переслідує аналіз, та як результати можуть бути інтерпретовані у контексті прийняття рішень.

У межах даної наукової роботи об’єктом дослідження є валовий внутрішній продукт (ВВП) – один із основних макроекономічних показників, що характеризує ринкову вартість усіх кінцевих товарів та послуг, вироблених в економіці за певний період часу. Згідно з методологією Світового банку, ВВП вимірюється як у поточних, так і у постійних доларах США, і може бути представленим як річний або квартальний часовий ряд [15].

Змістовно задача прогнозування ВВП полягає у визначенні очікуваного рівня економічної активності країни у наступному періоді на основі її попередніх значень. Такий прогноз дає змогу уряду, бізнесу, інвесторам та міжнародним інституціям приймати стратегічні рішення: планувати бюджет, розраховувати податкові надходження, формувати грошово-кредитну політику тощо.

У цьому контексті прогнозування ВВП слід розглядати як частину ширшої інформаційної системи, призначеної для підтримки рішень. Основні змістовні компоненти задачі можна представити так:

1. Ціль прогнозування: динаміка ВВП країни (наприклад, України, Німеччини, США) на наступний рік або квартал.

2. Аналітична основа: багаторічний часовий ряд ВВП, отриманий з офіційних джерел (World Bank, IMF, OECD).

3. Контекст: економічна політика країни, демографічні зміни, зовнішні ринки, податкове навантаження тощо.

4. Мета аналізу: побудова прогнозної моделі, що дозволяє оцінити майбутні значення ВВП із заданим рівнем точності, адаптивності та інтерпретованості.

Щоб побудувати ефективну модель, слід також розглянути змістовні властивості фінансового часового ряду:

- тренд: довгострокова тенденція зміни ВВП. Наприклад, поступове зростання внаслідок економічного розвитку.

- сезонність: повторювані закономірності протягом року, пов’язані з циклами виробництва, бюджетними витратами, політикою центрального банку.

- випадкові компоненти: непередбачувані зовнішні шоки, включно з політичними чи природними подіями.

- циклічність: економічні фази підйому і спаду, що виходять за межі сезонності.

У змістовному розумінні прогнозування можна представити через функцію залежності

$$\text{ВВП}_{t+1} = f(\text{ВВП}_t, \text{ВВП}_{t-1}, \ldots, \text{ВВП}_{t-k}) + \varepsilon_t, \tag{1.9}$$

де ε_t – залишкова (непередбачувана) складова, що може мати випадковий характер або бути частково поясненою іншими факторами.

Таким чином, модель прогнозування може набувати вигляду множинної регресії або багатовхідної нейронної мережі, де вхід включає не лише попередні значення ВВП, але й супутні макроекономічні індикатори.

Ще одним важливим аспектом є пояснення результатів. У змістовному контексті важливо не лише отримати прогноз, але й надати аргументацію, чому модель вважає, що, наприклад, наступного року ВВП зросте на 3%, а не зменшиться. Для цього слід використовувати інструменти інтерпретації моделей (наприклад, SHAP-значення), що дозволяють визначити вплив кожної змінної на кінцевий прогноз [21].

Таким чином, змістовна та формальна постановки задачі є взаємодоповнюючими: перша – надає аналітичний та прикладний контекст, друга – закладає основу для математичної реалізації в інформаційній системі.

Висновки до розділу 1

У першому розділі наукової роботи було здійснено комплексне дослідження предметної області прогнозування фінансових часових рядів із акцентом на динаміку валового внутрішнього продукту (ВВП) як ключового

макроекономічного показника. Проведений аналіз дозволив окреслити теоретичні та практичні основи задачі, сформувати уявлення про існуючі методології та оцінити їхні сильні й слабкі сторони у контексті побудови інформаційної системи прогнозування.

У підрозділі 1.1 було висвітлено сучасний стан наукових досліджень у сфері прогнозування часових рядів. Зокрема, було показано, що у фокусі уваги як українських, так і зарубіжних учених перебувають моделі ARIMA, GARCH, VAR, а також сучасні підходи з використанням нейронних мереж (LSTM, GRU) та ансамблевих методів машинного навчання. Доведено, що гібридні моделі, які поєднують класичні статистичні та глибокі алгоритмічні підходи, мають потенціал для підвищення точності прогнозування.

У підрозділі 1.2 було здійснено систематичний огляд сучасних інформаційних систем, що реалізують інструменти прогнозування фінансових показників. Виокремлено три головні категорії рішень: корпоративні аналітичні платформи (SAS, SPSS, SAP), відкриті програмні середовища (Python, R), а також хмарні сервіси (Google Cloud AI, Amazon Forecast). Особливу увагу приділено інструментам Python, як найбільш гнучкому і придатному середовищу для наукової розробки.

У підрозділі 1.3 запропоновано класифікацію методів прогнозування фінансових часових рядів на три основні групи: класичні статистичні, методи машинного навчання та гібридні моделі. Таблична форма представлення дозволила наочно порівняти підходи за критеріями точності, складності, обчислювальної вартості та інтерпретованості. Така класифікація є підґрунтям для раціонального вибору методів, що застосовуватимуться в наступних розділах роботи.

У підрозділі 1.4 було виявлено низку обмежень сучасних підходів. Серед них: низька здатність до адаптації в умовах структурних зсувів, чутливість до якості даних, складність інтерпретації глибоких моделей, а також потреба в значних обчислювальних ресурсах. Це створює виклики, які вимагають подальшого розвитку адаптивних та пояснюваних інформаційних систем.

У підрозділі 1.5 було здійснено як формальну, так і змістовну постановку задачі прогнозування ВВП. Математична формалізація включає опис вхідних параметрів, функціональних залежностей та функцій втрат, а змістовна – пояснює роль прогнозу в економічному плануванні, визначає контекст, впливові фактори та очікувану корисність результатів. У результаті сформовано чітке бачення завдання, яке підлягає розв’язанню в теоретичному та практичному блоках роботи.

Таким чином, розділ 1 заклав методологічний фундамент для побудови інформаційної системи прогнозування ВВП. Подальше дослідження буде зосереджене на математичних моделях, теоретичних принципах аналізу часових рядів і практичній реалізації прогнозного механізму за допомогою сучасних інструментів програмування.

2 ТЕОРЕТИЧНІ ОСНОВИ ПРОГНОЗУВАННЯ ФІНАНСОВИХ ЧАСОВИХ РЯДІВ

2.1 Основні поняття теорії часових рядів

Теорія часових рядів є фундаментальною математичною дисципліною, що досліджує залежності між послідовними спостереженнями змінної у часі. Часовий ряд – це впорядкована у часі послідовність числових значень випадкової величини, яка моделює реальний процес. У прикладному аспекті, до таких процесів належать фінансові показники, зокрема ціни на акції, обсяги продажів, індекси споживчих цін, валютні курси та валовий внутрішній продукт.

Формально, часовий ряд представляється як послідовність

$$Y = \{y_t\}_{t \in T}, \tag{2.1}$$

де $y_t \in \mathbb{R}$ – значення спостереження у момент часу t;

$T \subset \mathbb{Z}$ – дискретна множина моментів часу.

Залежно від контексту, час може бути виміряний у днях, місяцях, кварталах або роках. Якщо значення спостережень подано у фіксованих часових інтервалах, йдеться про регулярний часовий ряд. Інакше – про нерегулярний.

Ключовими поняттями при дослідженні часових рядів є стаціонарність, автокореляція, тренд, сезонність та білі шуми. Визначення та властивості цих компонентів є критично важливими для побудови адекватних моделей прогнозування.

Стаціонарність ряду означає, що його статистичні характеристики (математичне сподівання, дисперсія, автокореляційна функція) не змінюються у часі. Формально, слабка (вторинна) стаціонарність виконується, якщо

$$\mathbb{E}[y_t] = \mu = \text{const}, \quad \text{Var}(y_t) = \sigma^2 = \text{const}, \quad \text{Cov}(y_t, y_{t+h}) = \gamma(h), \tag{2.2}$$

де $\gamma(h)$ – автоковаріаційна функція, що залежить лише від лагу h, а не від моменту часу t [1, 2].

У практиці фінансового аналізу більшість часових рядів (зокрема ВВП) не є стаціонарними, оскільки мають довготривалі тренди, сезонність або структурні зсуви. Тому часто використовують методи приведення ряду до стаціонарного вигляду, наприклад, диференціювання

$$\Delta y_t = y_t - y_{t-1}, \tag{2.3}$$

або, у випадку сезонності з періодом s

$$\Delta_s y_t = y_t - y_{t-s}. \tag{2.4}$$

Тренд – це довгострокова тенденція зростання або спадання значень часового ряду. Якщо його не врахувати, модель може суттєво втратити точність у довгострокових прогнозах. Тренд може бути детермінованим (лінійним чи поліноміальним)

$$y_t = \beta_0 + \beta_1 t + \varepsilon_t, \tag{2.5}$$

або стохастичним, тобто модель має вигляд випадкового блуждання

$$y_t = y_{t-1} + \varepsilon_t, \tag{2.6}$$

де $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$ – білий шум, незалежний і рівномірно розподілений [2].

Сезонність – це компонент ряду, що повторюється через певні фіксовані інтервали часу, наприклад, щороку. Сезонна структура враховується або

шляхом включення сезонних дамі-змінних до регресії, або через включення сезонних лагів у моделі ARIMA. Наприклад, модель SARIMA має вигляд

$$\Phi_p(B)\Phi_P(B^s)y_t = \Theta_q(B)\Theta_Q(B^s)\varepsilon_t, \tag{2.7}$$

де B – оператор зсуву назад;

s – період сезонності;

Φ, Θ – поліноми автокореляції та ковзного середнього відповідно [3].

Автокореляція та часткова автокореляція – це інструменти для оцінки залежності значень часового ряду між собою на різних лагах. Автокореляційна функція визначається як

$$\rho(h) = \frac{\mathrm{Cov}(y_t, y_{t+h})}{\mathrm{Var}(y_t)}. \tag{2.8}$$

Автокореляція та часткова автокореляція є головними для ідентифікації порядків моделей ARIMA – значення лагів, що слід включити у модель. Типові графіки ACF дозволяють встановити довжину пам’яті часового ряду, що визначає його прогнозну структуру [2, 4].

У багатьох випадках реальні фінансові часові ряди є комбінацією кількох компонент: тренду, сезонності, циклічних коливань і випадкових збурень. Це відображається у так званому адитивному представленні

$$y_t = T_t + S_t + C_t + \varepsilon_t, \tag{2.9}$$

або мультиплікативному представленні

$$y_t = T_t \cdot S_t \cdot C_t \cdot \varepsilon_t, \tag{2.10}$$

де T_t – трендова компонента;

S_t – сезонна;

C_t – циклічна;

ε_t – випадкове збурення [1].

Слід звернути особливу увагу на поняття стохастичних процесів, білих шумів, інтегрованих рядів та спектрального аналізу, які є фундаментальними для розуміння природи часових змін та побудови адекватних моделей прогнозування.

Стохастичний процес у контексті часових рядів – це сімейство випадкових величин $\{Y_t : t \in T\}$, де кожне Y_t визначено на спільному ймовірнісному просторі. Таким чином, часовий ряд розглядається як реалізація стохастичного процесу. Центральним класом процесів у фінансовому аналізі є процеси з незалежними однаково розподіленими (i.i.d.) випадковими величинами та процеси з автокорельованою структурою, наприклад, процеси AR(p) або MA(q).

Поняття білого шуму є критично важливим для побудови адекватних моделей. Білий шум – це послідовність незалежних і однаково розподілених випадкових величин із нульовим математичним сподіванням і скінченною дисперсією

$$\varepsilon_t \sim WN(0,\sigma^2), \quad \mathbb{E}[\varepsilon_t] = 0, \quad \mathrm{Cov}(\varepsilon_t, \varepsilon_{t+h}) = 0, \forall h \neq 0. \tag{2.11}$$

Білий шум є базовим припущенням для залишків адекватно побудованої моделі: якщо після моделювання залишки не є білим шумом, модель вважається неспроможною до опису процесу.

Поняття стаціонарності другого порядку розширює клас стаціонарних процесів. У такому випадку вимагається лише сталість математичного сподівання і коваріаційної функції, але не розподілу. Це важливо для фінансових рядів, які рідко задовольняють жорсткі умови строгої стаціонарності

$$\mathbb{E}[Y_t] = \mu, \quad \mathrm{Cov}(Y_t, Y_{t+h}) = \gamma(h). \tag{2.12}$$

Це дозволяє будувати моделі типу ARMA або GARCH, навіть коли розподіли не є нормальними або симетричними, що часто зустрічається у фінансових даних із жирними хвостами.

Однією з важливих категорій є інтегровані часові ряди, які утворюються шляхом диференціювання нестабільного (нестійкого) ряду. Якщо послідовне диференціювання ряду d разів призводить до стаціонарного ряду, то кажуть, що початковий ряд є інтегрованим порядку d, і позначається як $I(d)$. Наприклад, якщо $y_t \sim I(1)$, то

$$\Delta y_t = y_t - y_{t-1} \sim I(0). \tag{2.13}$$

Це – основа моделі ARIMA, де перша буква "I" позначає інтеграцію.

Більшість макроекономічних рядів, зокрема ВВП, є інтегрованими порядку 1 або 2. Тобто вони мають стійкий тренд, який необхідно видалити для адекватного моделювання залишкових коливань. В іншому випадку модель буде мати штучно завищену кореляцію (спуріозну залежність) і занижену прогностичну здатність [26].

Поняття спектрального аналізу також має важливе значення для вивчення циклічності в рядах. У класичному підході часовий ряд представляється як сума гармонічних складових (т.зв. фур'є-розклад), і спектральна густина визначає, які частоти найбільше впливають на динаміку ряду. Формально спектральна густина $f(\lambda)$ обчислюється як

$$f(\lambda) = \frac{1}{2\pi} \sum_{h=-\infty}^{\infty} \gamma(h) e^{-i\lambda h}, \quad \lambda \in [-\pi, \pi], \tag{2.14}$$

де $\gamma(h)$ – автоковаріаційна функція.

Аналіз спектру дозволяє виявити приховані періодичності та цикли, що

недоступні при звичайному візуальному аналізі [27].

Ще одне важливе поняття – коваріаційна матриця лагів, яка формується на основі зсунутих версій ряду. Вона є основою для розрахунку оптимальних фільтрів (наприклад, Калмана) або побудови векторних моделей (VAR), коли враховується взаємозв'язок між кількома часовими рядами. При цьому, високий рівень мультиколінеарності між лагами може свідчити про необхідність застосування регуляризації або відсікання неінформативних змінних.

Описані поняття формують базис теоретичної частини дослідження часових рядів. Їх інтеграція в інформаційні системи дозволяє створити інструменти, які не лише здійснюють механічне прогнозування, але й враховують глибоку математичну природу процесів, що моделюються. Це є особливо важливим у контексті макроекономічних показників, де необхідна висока точність, інтерпретованість і стійкість моделей до змін середовища.

2.2 Класичні статистичні моделі

Математичне моделювання часових рядів традиційно ґрунтується на класичних статистичних моделях, які дозволяють формалізувати залежність поточних значень від попередніх спостережень. Ці моделі вирізняються високим рівнем теоретичної обґрунтованості, формальною простотою і відносною легкістю реалізації, що робить їх базовими інструментами в економетричному аналізі. Найбільш вживаними є авторегресійна модель (AR), модель ковзного середнього (MA), комбінована модель ARMA, а також її узагальнення – модель авторегресії з інтегрованими складовими та ковзним середнім (ARIMA), яка дозволяє моделювати нестаціонарні ряди.

Авторегресійна модель порядку p, позначена як AR(p), моделює значення часового ряду як лінійну комбінацію його попередніх значень. Математично вона описується рівнянням

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \varepsilon_t, \tag{2.15}$$

де ϕ_i – коефіцієнти авторегресії

$\varepsilon_t \sim WN(0, \sigma^2)$ – білий шум.

У операторній формі, використовуючи оператор зсуву назад B, модель записується як

$$\Phi(B) y_t = \varepsilon_t, \quad \Phi(B) = 1 - \phi_1 B - \ldots - \phi_p B^p. \tag{2.16}$$

Модель AR(p) добре описує ряди, де спостерігається автокореляція значень, однак вона є доцільною лише для стаціонарних процесів.

Модель ковзного середнього порядку q, позначена як MA(q), будує поточне значення як функцію лінійної комбінації попередніх значень випадкових збурень

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}. \tag{2.17}$$

У операторному вигляді це рівняння набуває форми

$$y_t = \Theta(B) \varepsilon_t, \quad ma\ \Theta(B) = 1 + \theta_1 B + \ldots + \theta_q B^q. \tag{2.18}$$

Модель MA добре підходить для опису короткострокових змін, які не мають тривалого тренду, і є особливо ефективною, коли автокореляція обмежується невеликою кількістю лагів.

Комбінація моделей AR і MA приводить до моделі ARMA(p, q), яка дозволяє одночасно враховувати як внутрішню структуру автокореляції, так і вплив залишків попередніх періодів

$$y_t = \phi_1 y_{t-1} + \ldots + \phi_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \ldots + \theta_q \varepsilon_{t-q}. \tag{2.19}$$

В операторному записі

$$\Phi(B)y_t = \Theta(B)\varepsilon_t. \tag{2.20}$$

ARMA-моделі ефективні для моделювання стаціонарних процесів і часто застосовуються до трансформованих рядів, які було приведено до стаціонарного вигляду за допомогою диференціювання або логарифмування.

Проте на практиці більшість економічних і фінансових часових рядів не є стаціонарними. Наприклад, ВВП майже завжди демонструє тренд і, можливо, сезонність, що вимагає застосування розширених моделей. Саме тому було введено модель ARIMA(p, d, q), де параметр d означає порядок диференціювання, необхідний для приведення ряду до стаціонарного вигляду

$$\Delta^d y_t = (1-B)^d y_t. \tag{2.21}$$

Модель ARIMA(p, d, q) формалізується як

$$\Phi(B)\Delta^d y_t = \Theta(B)\varepsilon_t. \tag{2.22}$$

Таким чином, ARIMA дозволяє моделювати не лише автокорельовані ряди, але й ті, що мають тренд (лінійний або логарифмічний). Для прикладу, якщо ряд ВВП є інтегрованим порядку 1, тобто $I(1)$, то його перші різниці (Δy_t) будуть стаціонарними і придатними для ARMA-моделювання.

Класичні статистичні моделі мають важливу властивість – лінійність. Це означає, що прогнозне значення є лінійною функцією попередніх спостережень і шумів. Попри це, вони здатні ефективно описувати ряди з помірною динамікою, стабільними патернами та відносно короткою пам’яттю. Саме тому ARIMA є стандартною моделлю для макроекономічних рядів, зокрема таких як валовий внутрішній продукт, які змінюються повільно та демонструють лінійні

тренди у тривалих періодах.

Одним із розширень моделі ARIMA, що дозволяє враховувати сезонні коливання у часових рядах, є модель SARIMA (Seasonal ARIMA), яка поєднує параметри лінійної авторегресії, диференціювання, ковзного середнього та сезонності. SARIMA-модель має вигляд

$$\Phi(B)\Phi_s(B^s)\Delta^d\Delta_s^D y_t = \Theta(B)\Theta_s(B^s)\varepsilon_t, \tag{2.23}$$

де $\Phi(B)$, $\Theta(B)$ – поліноми авторегресійної та MA-частин;

$\Phi_s(B^s)$, $\Theta_s(B^s)$ – сезонні поліноми порядку P і Q;

Δ^d, Δ_s^D – відповідно, звичайне та сезонне диференціювання;

s – період сезонності (наприклад, 4 для квартальних, 12 для місячних рядів).

Наприклад, у випадку квартального ВВП, сезонність може бути чітко вираженою, оскільки деякі галузі економіки активніші в певні періоди року (сільське господарство, роздрібна торгівля, туризм). SARIMA дозволяє вбудувати цю інформацію без необхідності додаткового вручного формування дамі-змінних.

Параметри SARIMA зазвичай позначаються як $(p,d,q)\times(P,D,Q)_s$, де перша трійка – параметри звичайної ARIMA, а друга – сезонної частини. Підбір параметрів здійснюється аналогічно до ARIMA, з використанням ACF, PACF, а також інформаційних критеріїв AIC і BIC. У багатьох сучасних бібліотеках (наприклад, pmdarima в Python) існує функціонал автоматичного підбору параметрів, що істотно полегшує роботу аналітика.

Наступним етапом є оцінювання параметрів моделей, яке зазвичай здійснюється методом максимізації функції правдоподібності (MLE – Maximum Likelihood Estimation) або методом найменших квадратів. Наприклад, для AR(p) параметри $\phi_1,\ldots,\phi_p$ обираються так, щоб мінімізувати функцію

$$\mathcal{L}(\phi_1,\ldots,\phi_p) = \sum_{t=p+1}^{T} (y_t - \hat{y}_t)^2. \quad (2.24)$$

При використанні MLE формується ймовірнісна модель шуму, і обчислюється ймовірність спостереження наявних даних при заданих параметрах. Обидва методи дають наближено однакові результати, але MLE забезпечує кращу теоретичну інтерпретацію і дозволяє будувати довірчі інтервали для прогнозів [34].

Після побудови моделі важливо провести перевірку її адекватності. Це передбачає:

- перевірку залишків на відповідність білому шуму (тест Люнга–Бокса);

- перевірку нормальності розподілу залишків (тест Шапіро–Уілка або Колмогорова–Смирнова);

- оцінку автокореляційної структури залишків (ACF/PACF залишків);

- стабільність коефіцієнтів у часі (CUSUM-тест або порівняння моделей на різних підвибірках).

Модель вважається адекватною, якщо її залишки є незалежними, неавтокорельованими та нормально розподіленими. У противному разі модель потребує перегляду, зміни порядку або трансформації вхідного ряду.

Попри свою ефективність у багатьох задачах, класичні статистичні моделі мають обмеження. Найбільш суттєві з них:

- лінійність: моделі ARIMA не здатні моделювати складні нелінійні структури, характерні для фінансових ринків;

- обмежена адаптивність: моделі погано реагують на структурні зсуви та зміни в поведінці процесу;

- вимога стаціонарності або приведення до неї: деякі процеси погано піддаються стабілізації навіть після кількох рівнів диференціювання;

- слабка масштабованість: складність моделей швидко зростає з ростом кількості параметрів, що ускладнює роботу з багатоваріантними рядами.

У зв’язку з цим у практиці дедалі частіше використовуються гібридні

підходи, які комбінують ARIMA-моделі з нелінійними моделями машинного навчання – наприклад, ARIMA + LSTM або ARIMA + XGBoost. Такий підхід дозволяє зберегти інтерпретованість базової моделі, водночас використовуючи прогностичну потужність глибокого навчання для опису залишків.

У контексті задачі прогнозування валового внутрішнього продукту ARIMA та її варіації є логічним вибором, особливо на етапі початкового аналізу. ВВП, як правило, має регулярну річну сезонність, стійкий тренд і помірну волатильність – тобто ті властивості, які добре описуються класичними статистичними моделями. Надалі ARIMA-прогнози можуть бути розширені за допомогою більш гнучких і потужних моделей, що буде розглянуто в наступних підрозділах.

2.3 Розширені моделі та методи

2.3 Розширені моделі та методи (SARIMA, ARCH/GARCH, VAR)

Класичні моделі ARIMA, попри свою ефективність у багатьох прикладних задачах, мають суттєві обмеження при роботі з часовими рядами, що демонструють сезонність, волатильність або взаємозалежність між кількома змінними. У відповідь на ці виклики було розроблено низку розширених моделей, серед яких найбільш вживаними є: сезонна модель SARIMA, моделі умовної гетероскедастичності ARCH та GARCH, а також векторна авторегресійна модель VAR. Ці моделі дозволяють описати більш складні структури часових рядів, що характерні для економіки та фінансів.

SARIMA (Seasonal ARIMA) є природним розширенням ARIMA, коли часовий ряд демонструє чітко виражену сезонність. Як зазначалося раніше, SARIMA враховує як регулярну компоненту, так і сезонну через додаткові параметри авторегресії, інтегрування та ковзного середнього

$$SARIMA(p,d,q)(P,D,Q)_s, \tag{2.25}$$

де s – сезонний період;

P, D, Q – сезонні параметри.

У більшості макроекономічних задач, таких як прогнозування ВВП, обсягів продажів, попиту на ресурси, сезонні коливання відіграють ключову роль, і SARIMA дозволяє описати їх у рамках лінійного підходу. Застосування SARIMA також передбачає використання методів Бокса–Дженкінса для вибору параметрів на основі ACF, PACF та інформаційних критеріїв [34].

Проте не всі характеристики фінансових рядів можна ефективно описати за допомогою SARIMA. Зокрема, моделі ARIMA та SARIMA припускають, що дисперсія залишків є сталою в часі. Це припущення порушується у багатьох фінансових даних, де спостерігається волатильність – тобто змінна дисперсія, яка збільшується або зменшується в певні періоди. Для таких випадків застосовуються моделі умовної гетероскедастичності – ARCH (Autoregressive Conditional Heteroskedasticity) та GARCH (Generalized ARCH), запропоновані Робертом Енглом та Тімом Боллерслевом відповідно.

Модель ARCH(p) описує дисперсію залишків як функцію квадратів попередніх помилок

$$\varepsilon_t \sim N(0,\sigma_t^2), \quad \sigma_t^2 = \alpha_0 + \alpha_1\varepsilon_{t-1}^2 + \ldots + \alpha_p\varepsilon_{t-p}^2. \tag{2.26}$$

Модель GARCH(p,q) є узагальненням, де дисперсія залежить також від власних лагів

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p}\alpha_i\varepsilon_{t-i}^2 + \sum_{j=1}^{q}\beta_j\sigma_{t-j}^2. \tag{2.27}$$

GARCH-моделі дозволяють моделювати періоди підвищеної і зниженої волатильності, які типові для ринків капіталу, валютних курсів і цін на

енергоресурси. У випадку макроекономічних змінних, таких як ВВП, ці моделі використовуються рідше, але можуть бути корисними для аналізу економічної нестабільності або посткризового відновлення [3].

Іншим важливим напрямом є моделювання багатоваріантних часових рядів, де враховується взаємозалежність між кількома змінними. Наприклад, ВВП, рівень безробіття, інфляція та державні витрати можуть мати складну кореляційну структуру. У таких випадках застосовується векторна авторегресійна модель (VAR). На відміну від ARIMA, яка моделює одну змінну, VAR описує систему рівнянь, де кожна змінна моделюється через власні лаги і лаги всіх інших змінних у системі

$$\mathbf{Y}_t = c + \sum_{i=1}^{p} A_i \mathbf{Y}_{t-i} + \varepsilon_t, \tag{2.28}$$

де $\mathbf{Y}_t \in \mathbb{R}^k$ – вектор із k змінних;

$A_i \in \mathbb{R}^{k \times k}$ – матриці коефіцієнтів;

ε_t – вектор білих шумів.

VAR-моделі широко застосовуються у макроекономіці для аналізу політичних впливів, ефектів шоків, а також побудови сценаріїв. Наприклад, можна оцінити, як зміна грошово-кредитної політики вплине на реальний ВВП і рівень цін. Для побудови VAR необхідно забезпечити стаціонарність усіх рядів, що входять до моделі; у випадку нестаціонарності використовуються моделі VECM (Vector Error Correction Model) для інтегрованих рядів з коінтеграцією.

Розширені моделі SARIMA, ARCH/GARCH та VAR дозволяють значно розширити можливості класичних підходів. Вони здатні моделювати сезонність, дисперсійні ефекти та взаємозалежність між змінними – тобто охоплюють ключові аспекти реальної економічної динаміки. Їх включення до аналітичного арсеналу інформаційної системи прогнозування ВВП є необхідною умовою для досягнення високої точності й адекватності

моделювання.

2.4 Підходи машинного навчання

Методи машинного навчання суттєво розширюють традиційні підходи до прогнозування фінансових часових рядів, дозволяючи будувати моделі без суворих припущень щодо стаціонарності, нормальності розподілу, лінійності залежностей тощо. Вони стали особливо актуальними з розвитком комп'ютерної потужності та появою великих відкритих фінансових датасетів. Застосування алгоритмів машинного навчання в економіці дозволяє враховувати ширший спектр факторів, будувати складніші моделі взаємозв'язків та адаптуватися до змін у динаміці даних.

Однією з базових моделей, яка має широке застосування як у класичній статистиці, так і в машинному навчанні, є лінійна регресія. Незважаючи на свою простоту, вона часто виступає стартовою точкою для побудови більш складних моделей. У випадку часових рядів, лінійну регресію можна використовувати як autoregressive model exogenous (ARX), де прогнозована змінна залежить не лише від власних лагів, а й від зовнішніх предикторів

$$y_t = \beta_0 + \sum_{i=1}^{p} \beta_i y_{t-i} + \sum_{j=1}^{q} \gamma_j x_{t-j} + \varepsilon_t, \tag{2.29}$$

де x_{t-j} – зовнішні змінні (наприклад, інфляція, рівень зайнятості, курс валют);

β_i, γ_j – параметри моделі, $\varepsilon_t \sim N(0, \sigma^2)$.

Регресійні моделі легко розширюються за допомогою регуляризаційних технік. Зокрема, Lasso-регресія (Least Absolute Shrinkage and Selection Operator) дозволяє виконувати не лише оцінювання коефіцієнтів, але й автоматичний відбір змінних. Її функція втрат має вигляд

$$\mathcal{L}(\beta) = \sum_{t=1}^{T} (y_t - \hat{y}_t)^2 + \lambda \sum_{i=1}^{p} |\beta_i|, \tag{2.30}$$

де λ – гіперпараметр, що контролює ступінь регуляризації.

Якщо λ велике, багато коефіцієнтів зануляються, і модель стає більш простою.

Інший підхід – Ridge-регресія, де регуляризація здійснюється через L2-норму

$$\mathcal{L}(\beta) = \sum_{t=1}^{T} (y_t - \hat{y}_t)^2 + \lambda \sum_{i=1}^{p} \beta_i^2. \tag{2.31}$$

Ridge зменшує варіацію моделі при наявності мультиколінеарності, що часто трапляється при великій кількості лагів або корельованих змінних. У випадку фінансових рядів, це може бути корисно при роботі з квартальними ВВП, коли одночасно враховуються лаги на кілька періодів назад і додаткові індикатори.

Лінійні регресійні моделі мають обмежену здатність до моделювання нелінійних залежностей, які часто трапляються у фінансових часових рядах. У зв’язку з цим виникає потреба у застосуванні деревоподібних моделей, перш за все – рішень дерев (decision trees). Дерево будує рекурсивні розбиття простору ознак за критерієм мінімізації помилки (найчастіше середньоквадратичної або ентропійної). Розбиття відбувається доти, доки не буде досягнуто максимальної глибини або мінімального розміру вузла.

Формально, для кожного вузла m знаходиться оптимальна ознака j та поріг s, які мінімізують суму квадратів відхилень у дочірніх вузлах

$$\min_{j,s} \left[\sum_{i \in R_1(j,s)} (y_i - \bar{y}_1)^2 + \sum_{i \in R_2(j,s)} (y_i - \bar{y}_2)^2 \right], \tag{2.32}$$

де R_1, R_2 – підмножини спостережень, що потрапляють у ліву або праву гілку;

$\bar{y}_1, \bar{y}_2$ – середні значення у відповідних групах.

Незважаючи на простоту реалізації, дерева рішень мають ряд недоліків: вони нестабільні до невеликих змін у даних, мають низьку узагальнювальну здатність та схильні до перенавчання. Для вирішення цих проблем у машинному навчанні широко застосовують ансамблеві методи, які об'єднують велику кількість дерев або інших слабких моделей у потужний прогнозний механізм.

Одним із таких методів є випадковий ліс (Random Forest), що поєднує декілька рішень дерев, побудованих на різних випадкових підвибірках даних та підмножинах ознак. Кожне дерево виконує прогноз, а результатом є середнє (у випадку регресії) або мода (у випадку класифікації)

$$\hat{y}_t = \frac{1}{N} \sum_{i=1}^{N} T_i(x_t), \tag{2.33}$$

де T_i – окреме дерево в ансамблі;

N – загальна кількість дерев.

Завдяки випадковості та агрегуванню, випадковий ліс значно зменшує варіацію і забезпечує високу точність навіть при великій кількості предикторів.

Інший підхід – бустинг, який полягає у послідовному навчанні слабких моделей, кожна з яких фокусується на помилках попередніх. Основна ідея бустингу полягає в побудові адитивної моделі

$$\hat{y}_t = \sum_{m=1}^{M} \nu h_m(x_t), \tag{2.34}$$

де h_m – базові предиктори (наприклад, дерева);

$\nu \in (0,1)$ – коефіцієнт навчання (learning rate), що контролює вклад кожного нового дерева.

На кожному кроці обчислюється градієнт функції втрат і відповідно до нього будується нове дерево, що компенсує помилку попередніх.

Застосування бустингових моделей, таких як XGBoost, LightGBM або CatBoost, дозволяє досягати дуже високої точності прогнозування завдяки:

- підтримці пропущених значень;
- автоматичному кодуванню категоріальних ознак;
- регуляризації (через L1 і L2);
- паралельній обробці та ефективному використанню пам’яті.

Ці моделі особливо ефективні у задачах прогнозування ВВП, коли в систему вводяться численні макроекономічні індикатори: рівень інфляції, процентні ставки, обсяг торгівлі, індекси промисловості тощо. У поєднанні з часовими лагами (тобто попередніми значеннями) вони дозволяють побудувати високоадаптивні моделі, що враховують як довгострокові тренди, так і короткострокові флуктуації.

Практичне впровадження ансамблевих методів у фінансове прогнозування передбачає кілька ключових кроків: інженерію ознак, вибір лагів, оптимізацію гіперпараметрів і забезпечення інтерпретованості результатів. Особливість використання моделей машинного навчання в контексті часових рядів полягає в необхідності трансформувати часову структуру у формат, прийнятний для “табличних” методів, таких як дерева або бустинг. Це вимагає побудови матриці ознак, де кожен рядок відповідає моменту часу t, а стовпці – лагам і супутнім змінним.

Наприклад, для ряду y_t, із вибором 3 лагів, матриця навчання матиме вигляд

$$X_t = [y_{t-1} \quad y_{t-2} \quad y_{t-3}], \quad \textit{ціль}: y_t. \tag{2.35}$$

За аналогією, можна включити також x_{t-1}, x_{t-2} – інші економічні показники (наприклад, інфляцію або рівень безробіття), сформувавши багатовимірну

матрицю ознак. Усе це перетворює задачу прогнозування часового ряду у класичну регресійну задачу, для якої доступний широкий спектр алгоритмів.

Наступним важливим кроком є оптимізація гіперпараметрів моделей. Наприклад, у випадку XGBoost параметри включають: глибину дерев (max_depth), кількість дерев (n_estimators), швидкість навчання (learning_rate), регуляризаційні коефіцієнти (alpha, lambda) тощо. Їх можна підібрати вручну або за допомогою методів автоматичного пошуку: Grid Search, Random Search, Bayesian Optimization.

Для задач фінансового прогнозування з переважанням довгострокових трендів RMSE вважається однією з найнадійніших метрик, оскільки сильніше карає великі відхилення, що можуть мати критичне значення при стратегічному плануванні (наприклад, бюджету країни на рік).

Після тренування моделі особливої уваги вимагає інтерпретація результатів. Це особливо актуально у сфері макроекономіки, де прогнози часто стають основою для прийняття політичних або бюджетних рішень. Одним із найбільш популярних сучасних підходів є SHAP-аналіз (SHapley Additive exPlanations). Він дозволяє оцінити внесок кожної ознаки в прогноз

$$\hat{y}_t = \phi_0 + \sum_{i=1}^{n} \phi_i, \tag{2.36}$$

де ϕ_0 – базове значення (середній прогноз);

ϕ_i – внесок i-тої ознаки. SHAP-аналіз побудований на ідеях кооперативної теорії ігор і дозволяє обчислювати справедливий вклад кожної ознаки незалежно від її взаємодій з іншими.

Окрім SHAP, також використовують Feature Importance, яка відображає, наскільки часто певна змінна використовувалась у розбиттях дерев і наскільки вони знижували функцію втрат. Проте цей підхід менш стійкий до мультиколінеарності, яка властива економічним даним (наприклад, ВВП часто корелює з державними витратами та промисловим виробництвом).

Ще одним важливим напрямком є гібридизація моделей, тобто поєднання класичних і ML-методів. Найбільш поширеним є сценарій, коли базовий тренд прогнозується за допомогою ARIMA або SARIMA, а залишки (помилки прогнозу) передаються до ML-моделі (наприклад, XGBoost), яка моделює нелінійні залежності, не враховані класичною частиною. Такий підхід формалізується як

$$y_t = \hat{y}_t^{ARIMA} + \hat{y}_t^{ML} + \varepsilon_t. \tag{2.37}$$

Це дозволяє зберегти інтерпретованість основної динаміки і водночас підвищити точність через адаптацію до складніших залежностей.

У сучасних інформаційних системах такі гібридні підходи можна реалізовувати у вигляді каскадних моделей, мета-моделей (stacking), або через модульну архітектуру, де окремі компоненти відповідають за окремі аспекти прогнозу – тренд, сезонність, шокові зміни, екзогенні фактори тощо. Все це сприяє побудові більш надійних, масштабованих і гнучких аналітичних рішень.

Методи машинного навчання не лише доповнюють класичні підходи до прогнозування фінансових часових рядів, але й здатні забезпечити новий рівень аналітики, враховуючи

- глибину структури даних;
- високу адаптивність до змін;
- автоматизацію побудови моделей;
- можливість інтеграції з зовнішніми джерелами (новини, настрої ринку, соціальні сигнали).

Для задач прогнозування ВВП це означає здатність враховувати не лише “жорсткі” економічні показники, але й сигнали, що формуються внаслідок змін політичної кон’юнктури, глобальних ринків, або соціальних очікувань. Таким чином, методи машинного навчання мають критичне значення для побудови сучасних інформаційних систем прогнозування, орієнтованих на складні економічні процеси.

2.5 Нейронні мережі для часових рядів

Застосування нейронних мереж у задачах прогнозування часових рядів стало однією з найвагоміших тенденцій у сучасній аналітиці, зокрема в області фінансів та макроекономіки. На відміну від класичних статистичних моделей або методів машинного навчання, які працюють із табличними даними, нейронні мережі мають здатність моделювати складні динамічні залежності в послідовних даних. Їх ключовою перевагою є можливість автоматичного виявлення довгострокових залежностей, адаптація до нерегулярностей у поведінці ряду та навчання без потреби в суворій трансформації вхідного сигналу.

Класичні нейронні мережі, такі як багатошарові перцептрони (MLP), працюють із фіксованою кількістю входів і не враховують порядок подачі даних. Це унеможливлює їхнє ефективне використання в задачах, де послідовність і часовий контекст є ключовими. Тому було розроблено спеціальні архітектури, пристосовані до послідовних структур. Основною з них є рекурентна нейронна мережа (RNN – Recurrent Neural Network).

Рекурентна нейронна мережа будується на ідеї зберігання прихованого стану h_t, який передається від одного кроку часу до наступного. На кожному кроці мережа отримує нове значення x_t та попередній стан h_{t-1}, обчислюючи новий стан і вихід

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b), \tag{2.38}$$

$$\hat{y}_t = f(W_y h_t + c), \tag{2.39}$$

де σ – нелінійна активаційна функція (наприклад, tanh або ReLU);

W_h, W_x, W_y – вагові матриці;

b, c – зміщення.

Таким чином, RNN моделює залежність вихідного сигналу не лише від

поточного вхідного значення, але й від усього попереднього контексту, що особливо корисно у задачах прогнозування фінансових рядів, де актуальні впливи минулих періодів.

Проте базова архітектура RNN має суттєві обмеження. Головна з них – проблема зникнення або вибуху градієнтів. У процесі навчання методом зворотного поширення через час (Backpropagation Through Time, BPTT), похідні, що проходять через велику кількість шарів (часових кроків), можуть наближатися до нуля або нескінченності

$$\frac{\partial \mathcal{L}}{\partial \theta} = \sum_t \frac{\partial \mathcal{L}}{\partial h_t} \prod_{k=1}^{t} \frac{\partial h_k}{\partial h_{k-1}} \frac{\partial h_{k-1}}{\partial \theta}, \tag{2.40}$$

де θ – параметри мережі.

Коли $\frac{\partial h_k}{\partial h_{k-1}}$ містить градієнти близькі до нуля (через насиченість tanh або sigmoid), довготривалі залежності не передаються, а навчання стає неефективним.

Для подолання цієї проблеми були розроблені розширені рекурентні архітектури, які включають внутрішню механіку контролю передачі інформації. Найвідомішою з них є LSTM (Long Short-Term Memory) – модель із пам’яттю довгого та короткого терміну, запропонована Гохрайтером і Шмідхубером у 1997 році.

LSTM-модуль складається з стану пам’яті C_t, прихованого стану h_t та трьох «воріт», які керують потоком інформації:

1. ворота забування f_t

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \tag{2.41}$$

2. ворота оновлення i_t, кандидат нового стану $\tilde{C}_t$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \quad \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C), \tag{2.42}$$

3. ворота виходу o_t

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \tag{2.43}$$

і оновлення стану пам’яті

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t, \quad h_t = o_t \odot \tanh(C_t), \tag{2.44}$$

де $\odot$ – покомпонентне множення.

Така структура дозволяє LSTM зберігати інформацію протягом тривалих часових інтервалів і уникати згаданих вище проблем градієнтів.

LSTM виявилися надзвичайно ефективними в задачах фінансового прогнозування: вони здатні виявляти затримки впливу зовнішніх подій, працювати з нестабільними ринками та адаптуватися до зміни поведінки часових рядів. У практиці прогнозування ВВП LSTM дозволяють враховувати ефекти попередніх років, що часто є визначальними у формуванні поточного макроекономічного стану.

Окрім LSTM, існує спрощена варіація – GRU (Gated Recurrent Unit), яка зберігає більшість переваг LSTM, але має менше параметрів і швидше навчається. Вона об’єднує деякі ворота в один механізм, зменшуючи складність, але зберігаючи здатність до моделювання довгострокових залежностей

$$z_t = \sigma(W_z[h_{t-1}, x_t]), \quad r_t = \sigma(W_r[h_{t-1}, x_t]), \tag{2.45}$$

$$\tilde{h}_t = \tanh(W[r_t \odot h_{t-1}, x_t]), \quad h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t. \tag{2.46}$$

GRU часто використовуються там, де обчислювальні ресурси обмежені або коли дані не демонструють надто тривалі залежності. У задачах прогнозування середньо- або короткострокових коливань економічних

показників (наприклад, місячні індекси або квартальні ВВП) GRU забезпечують оптимальний баланс між продуктивністю й точністю.

Таким чином, перша хвиля глибоких нейронних моделей для часових рядів базується на рекурентній архітектурі. Вони змінили підхід до прогнозування, дозволивши виявляти як миттєві ефекти, так і повільні тренди в економічних даних. Проте із зростанням обсягів даних та складністю міжчасових взаємодій з'явилися нові виклики, що потребують альтернативної архітектури – зокрема, трансформерів, які будуть розглянуті далі.

Практичне застосування RNN, LSTM та GRU у фінансових і макроекономічних задачах вимагає врахування низки технічних і аналітичних аспектів. Перш за все, часові ряди в економіці часто містять пропущені значення, нерегулярності у часових інтервалах, викиди та структурні зсуви. Тому перед передачею даних у нейронну мережу їх необхідно стандартизувати. Найчастіше використовують мінімаксну нормалізацію

$$x'_t = \frac{x_t - \min(x)}{\max(x) - \min(x)}, \tag{2.47}$$

або z-нормалізацію

$$x'_t = \frac{x_t - \mu}{\sigma}, \tag{2.48}$$

де μ – середнє значення;

σ – стандартне відхилення.

Це дозволяє забезпечити стабільне навчання, особливо коли активатори в мережі є чутливими до масштабу.

Ключовим етапом у застосуванні рекурентних мереж є формування вікон часу (time windows). Наприклад, для прогнозу y_{t+1} за попередні k значень $y_t, y_{t-1}, \ldots, y_{t-k+1}$, формується тензор

$$X = \left\{ \mathbf{x}^{(i)} \right\}_{i=1}^{N}, \quad \mathbf{x}^{(i)} = [y_i, y_{i+1}, \ldots, y_{i+k-1}], \quad y^{(i)} = y_{i+k}. \tag{2.49}$$

Це дозволяє подати ряд у вигляді послідовностей для мережі, яка передбачає наступне значення на основі попередніх. Такий підхід часто називається "sliding window" і є основою майже всіх сучасних застосувань LSTM до фінансових рядів.

Оскільки рекурентні мережі мають велику кількість параметрів, існує ризик перенавчання, особливо при обмеженому обсязі даних. Для боротьби з цим застосовують методи регуляризації. Найпоширеніший – Dropout: випадкове занулення частини нейронів під час навчання з імовірністю p. У випадку RNN використовують recurrent dropout, який застосовується до рекурентних з'єднань

$$h_t = f(Wh_{t-1} \cdot d + Ux_t + b), \tag{2.50}$$

де $d \sim \text{Bernoulli}(1-p)$.

Dropout допомагає уникнути надмірної залежності від окремих шляхів у мережі, стимулюючи загальну узагальнювальну здатність.

Ще один важливий аспект – довжина послідовностей. У фінансових задачах вибір оптимальної довжини "вікна" для моделі є критичним: надто коротке вікно не дозволить моделі вловити тенденцію, тоді як надто довге – призведе до розмиття важливих сигналів і зростання розмірності входу. Часто застосовується метод крос-валідації з різними довжинами k, де обирається значення, яке мінімізує помилку прогнозу на валідаційній вибірці.

Нейронні мережі також дають змогу моделювати багатовимірні часові ряди, що особливо актуально в макроекономіці. Наприклад, можна передавати в мережу не лише y_t (ВВП), а й супутні індикатори: рівень безробіття u_t, інфляцію π_t, індекс споживчих цін CPI_t, тощо. Таким чином, вхідний тензор має форму

$$X \in \mathbb{R}^{N \times T \times d}, \tag{2.51}$$

де N – кількість прикладів;

T – довжина послідовності;

d – кількість ознак на крок.

Це дозволяє LSTM моделювати взаємозв'язки між змінними в часі, що є особливо важливим при аналізі міжгалузевих економічних процесів.

Щоб забезпечити пояснюваність моделей, дослідники все частіше застосовують підходи, що аналізують важливість кроків послідовності. Наприклад, методи attention (увага) дозволяють мережі автоматично визначати, які з попередніх кроків є найважливішими для прогнозу. Для кожного кроку часу обчислюється коефіцієнт уваги

$$\alpha_t = \frac{\exp(e_t)}{\sum_{i=1}^{T} \exp(e_i)}, \quad e_t = \text{score}(h_t, \bar{h}), \tag{2.52}$$

де h_t – стан у момент t;

$\bar{h}$ – цільовий стан або контекст.

Увага дозволяє моделі сфокусуватися лише на релевантних кроках, що значно покращує інтерпретованість і стабільність у фінансових прогнозах.

Зокрема, застосування механізмів уваги в задачі прогнозу ВВП дає змогу мережі виявляти, що певні макроекономічні події чи зміни, які відбулися, наприклад, 6 або 8 кварталів тому, мають більший вплив на поточну економічну динаміку, ніж зміни в найближчому минулому. Це важко вловити без явно заданих правил, проте такі закономірності природно виникають у нейронних мережах із увагою.

Останнім важливим етапом є оцінка якості прогнозів, яка здійснюється на тестовій вибірці. Окрім стандартних метрик (MAE, RMSE), у фінансовій сфері

також важлива оцінка трендової адекватності (наприклад, відсоток правильно спрогнозованих напрямків зміни) та економічна значущість прогнозу – наприклад, чи достатній прогнозний горизонт для ухвалення бюджету або реалізації економічної стратегії.

Отже, використання RNN, LSTM і GRU в задачах прогнозування фінансових часових рядів забезпечує набагато глибше розуміння динаміки економіки, здатність до моделювання нелінійних процесів, виявлення довгострокових впливів і побудову гнучких прогнозів у складному середовищі. Проте ці архітектури мають певні обмеження, пов'язані з послідовною природою обробки, яка ускладнює масштабування. Саме для подолання цього обмеження і була створена принципово нова архітектура – трансформери.

Попри ефективність рекурентних архітектур, таких як LSTM і GRU, вони мають фундаментальне обмеження – послідовну обробку вхідної інформації. Це значно ускладнює паралелізацію навчання, особливо на довгих часових послідовностях, та уповільнює обчислення в реальному часі. Крім того, хоча механізми уваги покращують інтерпретованість та адаптивність, їхнє вбудовування в рекурентну структуру не повністю вирішує проблему масштабованості. У відповідь на ці виклики у 2017 році було запропоновано нову архітектуру – Transformer.

Трансформер, розроблений Васвані та співавторами [66], повністю відмовляється від рекурентності. Його головною інновацією є механізм самоуваги (self-attention), який дозволяє моделі одночасно аналізувати всі кроки вхідної послідовності та виявляти найважливіші залежності незалежно від відстані в часі. Формально, самоувага для кожної позиції t обчислюється так:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2.53}$$

де Q – запити (queries);

K – ключі (keys);

V – значення (values);

d_k – розмірність ключа.

Усі ці матриці формуються шляхом лінійних перетворень вхідної послідовності. На практиці використовують багатоголову увагу (multi-head attention), що дозволяє моделі паралельно фокусуватися на різних аспектах інформації.

Оскільки трансформери не мають вбудованого механізму обробки порядку елементів, у модель вводять позиційне кодування – додаткові вектори, які кодують положення елементів у послідовності. Найпоширеніше кодування – гармонічне

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_{model}}}\right), \quad PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{model}}}\right), \tag{2.54}$$

де pos – позиція;

i – індекс вектора.

Це дозволяє трансформеру дізнатися відстані між кроками часу, навіть без рекурентної структури.

У фінансових задачах трансформери почали застосовуватись порівняно нещодавно, але вже довели свою ефективність у таких задачах, як прогнозування цін на акції, обсягу продажів, попиту та економічних індикаторів. Особливо ефективними виявилися трансформери у випадках довгострокових залежностей, де традиційні LSTM починають втрачати інформацію.

Зокрема, для задач на кшталт прогнозування ВВП, де важливі як короткотермінові, так і довготермінові ефекти (наприклад, наслідки економічної політики, що діяла кілька років тому), трансформер дозволяє будувати глобальні залежності між кроками часу. Це особливо корисно при роботі з квартальними або річними даними, де вплив одного кроку може виявитися суттєвим через декілька періодів.

На базі трансформерів було розроблено низку спеціалізованих архітектур

для прогнозування часових рядів. Однією з найвідоміших є Temporal Fusion Transformer (TFT) [67], який був створений спеціально для інтерпретованого прогнозування послідовностей у багатоваріантному контексті. TFT поєднує такі компоненти:

- динамічне виділення важливих ознак (variable selection network),
- механізм уваги до часу (temporal attention),
- рекурентну частину (LSTM) для короткотермінової пам’яті,
- глобальні контексти через attention.

Це дозволяє моделі не лише формувати високоточний прогноз, а й пояснити, які змінні (наприклад, інфляція, курс валют, безробіття) були найважливішими у кожному періоді. Саме ця інтерпретованість робить TFT та подібні моделі придатними для практичного застосування в урядових аналітичних центрах, міжнародних організаціях та економічному плануванні.

Варто підкреслити, що на відміну від LSTM, трансформери можуть ефективно навчатися на непослідовних або нерегулярних даних, що відкриває нові горизонти в роботі з економічною інформацією. Наприклад, за допомогою відповідних масок або attention-механізмів можна побудувати модель, яка працює з неповними рядами, річними даними, або навіть враховує нерівномірні часові інтервали.

У реалізації трансформерів у задачах прогнозування фінансових рядів також активно застосовуються:

- Informer – ефективна архітектура зі зменшеним обчислювальним навантаженням для довгих послідовностей,
- Autoformer – модель, яка автоматично будує декомпозицію ряду на трендову і сезонну частини,
- FEDformer – спектрально-орієнтована модель для високочастотних економічних даних.

Усі ці моделі дозволяють будувати глибоко аналітичні, масштабовані та адаптивні системи прогнозування, які об’єднують можливості attention-механізмів, навчання представлень і нелінійного узагальнення.

Нейронні мережі – зокрема трансформери – є наразі найперспективнішим напрямком у прогнозуванні фінансових часових рядів. Їх здатність виявляти приховані структури, працювати з високою розмірністю даних, інтегрувати зовнішні сигнали і забезпечувати інтерпретованість робить їх ядром сучасних інформаційних систем економічного прогнозування.

Висновки до розділу 2

У другому розділі було здійснено глибоке теоретичне дослідження сучасних методів прогнозування фінансових часових рядів, з особливою увагою до економічних індикаторів, таких як валовий внутрішній продукт (ВВП). Теоретичне підґрунтя прогнозування базується на фундаментальних поняттях аналізу часових рядів – стаціонарності, сезонності, автокореляції, дисперсійної стабільності, а також основних класах стохастичних процесів. Ці властивості є критично важливими для вибору правильної моделі, способу трансформації даних і формування структур прогнозування, що враховують як короткотермінові, так і довготермінові залежності.

Було розглянуто класичні статистичні моделі AR, MA, ARMA та ARIMA, які продемонстрували свою ефективність у моделюванні стаціонарних процесів із обмеженим ступенем складності. Їх розширення у вигляді SARIMA дозволило врахувати сезонні компоненти, а моделі умовної гетероскедастичності ARCH і GARCH – адаптуватися до змінної дисперсії, яка часто виникає у фінансових і макроекономічних даних. VAR-моделі, у свою чергу, дозволяють формалізувати міжзмінні зв’язки у багатовимірних рядах. Усі ці моделі забезпечують добру інтерпретованість, але втрачають ефективність при складній, нелінійній або нерегулярній структурі даних.

Підходи машинного навчання, включно з регресійними методами, деревами рішень та ансамблевими алгоритмами (Random Forest, XGBoost, LightGBM), демонструють значно більшу гнучкість і точність. Вони

дозволяють будувати прогностичні моделі без суворих припущень про природу ряду, інтегрують велику кількість зовнішніх факторів і забезпечують адаптацію до змін середовища. Автоматизація підбору моделей, регуляризація, інтерпретація через SHAP і LIME – усе це робить машинне навчання потужним інструментом у задачах прогнозування економічних показників.

Особливу увагу було приділено глибинному навчанню, зокрема рекурентним нейронним мережам (RNN), їхнім модифікаціям LSTM і GRU, а також сучасній архітектурі трансформерів. Ці моделі здатні моделювати складні часові структури, виявляти довготривалі залежності, адаптуватися до динаміки ринку та враховувати взаємодію великої кількості економічних індикаторів у часі. Трансформери, зокрема Temporal Fusion Transformer, Informer та Autoformer, відкривають нові горизонти в побудові гнучких, масштабованих та інтерпретованих прогнозних систем.

Загалом, розділ 2 створює ґрунтовну теоретичну основу для побудови сучасної інформаційної системи прогнозування фінансових часових рядів. Отримані знання про моделі, їхні властивості, переваги й обмеження дозволяють переходити до практичної реалізації системи на основі мови програмування Python, що буде детально розглянуто у третьому розділі роботи.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ НА PYTHON

3.1 Вибір платформи та средовища розробки

Перш ніж перейти до реалізації інформаційної системи для прогнозування фінансових часових рядів, необхідно здійснити обґрунтований вибір технологічної платформи, інструментарію та середовища розробки. Від правильності цього вибору безпосередньо залежить ефективність реалізації моделі, зручність роботи дослідника, масштабованість коду, повторюваність експериментів, продуктивність обчислень і зрештою – якість прогнозів.

На сучасному етапі розвитку прикладної аналітики часових рядів використовується низка мов програмування та обчислювальних середовищ, кожне з яких має свої переваги та обмеження. До найпоширеніших мов, що використовуються для фінансового моделювання, належать Python, R, Matlab, Julia та C++. Вибір мови обумовлюється низкою факторів, серед яких: наявність бібліотек для статистики та машинного навчання, підтримка нейронних мереж, зручність візуалізації, швидкодія, інтеграція з базами даних та системами візуальної аналітики.

У таблиці 3.1 наведено порівняльну таблицю ключових мов програмування з точки зору застосування в задачах аналізу та прогнозування часових рядів.

З наведеного порівняння очевидно, що Python посідає провідні позиції за всіма ключовими критеріями. Він має широку екосистему бібліотек, активну спільноту, простий синтаксис і тісно інтегрується з інструментами візуалізації та нейронними мережами. Саме тому Python є найпопулярнішою мовою в сфері Data Science, зокрема й у фінансовій аналітиці та економетрії.

Особливо важливою є підтримка бібліотек для аналізу часових рядів. У Python існують потужні інструменти для роботи з економічними рядами – зокрема pandas, statsmodels, prophet, pmdarima, sklearn, tslearn, darts, gluonts. Для

побудови нейронних мереж доступні TensorFlow, PyTorch, Keras. Усі ці інструменти є відкритими, активно підтримуються спільнотою і широко документовані.

Таблиця 3.1 – Порівняння мов програмування

Критерій	Python	R	Matlab	Julia	C++
Наявність бібліотек для ML	Висока	Висока	Середня	Середня	Низька
Наявність бібліотек для часових рядів	Висока	Висока	Висока	Низька	Низька
Продуктивність обчислень	Висока	Середня	Висока	Висока	Дуже висока
Простота синтаксису	Висока	Висока	Висока	Середня	Низька
Візуалізація результатів	Висока	Висока	Висока	Середня	Низька
Підтримка нейронних мереж	Висока	Обмежена	Обмежена	Зростаюча	Висока
Вартість	Безкоштовна	Безкоштовна	Платна	Безкоштовна	Безкоштовна
Спільнота користувачів	Дуже велика	Велика	Обмежена	Менша	Вузька
Інтеграція з Jupyter Notebook	Повна	Обмежена	Можлива через сторонні засоби	Можлива	Немає

Наступним етапом є вибір середовища розробки. У практиці аналізу часових рядів найчастіше використовують наступні середовища: Jupyter Notebook, Spyder, PyCharm, RStudio, Google Colab, VSCode. Для оцінки доцільності використання кожного з них, розглянемо порівняльну таблицю 3.2.

Таблиця 3.2 – Порівняння середовищ розробки

Критерій	Jupyter Notebook	PyCharm	Spyder	RStudio	Google Colab
Підтримка Python	Повна	Повна	Повна	Часткова	Повна
Візуалізація графіків	Інтерактивна	Обмежена	Хороша	Обмежена	Інтерактивна
Підтримка блокового коду	Так	Ні	Ні	Ні	Так
Інтеграція з бібліотеками ML	Висока	Висока	Середня	Низька	Висока
Підтримка Markdown, Latex	Повна	Обмежена	Обмежена	Обмежена	Повна
Хмарна підтримка	Часткова	Немає	Немає	Обмежена	Повна
Навчальна зручність	Висока	Середня	Середня	Висока	Висока
Підтримка нейромереж	Висока	Висока	Середня	Обмежена	Висока

Як видно з таблиці, Jupyter Notebook має низку унікальних переваг: інтерактивність, можливість поєднання коду, формул, візуалізацій та пояснень

в одному документі. Така форма роботи ідеально підходить для дослідницької розробки та наукової комунікації результатів. Крім того, Jupyter підтримує імпорт та інтеграцію з усіма основними бібліотеками Python, включно з matplotlib, plotly, seaborn, scikit-learn, tensorflow, pmdarima, що є критично важливим у даній роботі.

Jupyter також дозволяє зберігати коди у вигляді .ipynb файлів, які легко передавати, архівувати, запускати повторно, що робить їх ідеальними для реплікаційних досліджень, а також для подальшої публікації моделей та результатів у відкритому доступі. Можливість запуску Jupyter через локальний сервер або у хмарному середовищі (наприклад, Google Colab) значно розширює його функціональність.

Крім того, у межах інформаційної системи, яка буде реалізована, необхідна можливість поетапного моделювання: спочатку – обробка даних, потім – побудова моделей, далі – валідація і нарешті – візуалізація результатів. Структура блоків Jupyter Notebook дає змогу логічно структурувати увесь процес, а Markdown-підтримка – коментувати кожен етап з поясненням алгоритмів, формул та висновків.

З огляду на вищевикладене, для реалізації інформаційної системи прогнозування фінансових часових рядів у цій роботі обрано мову програмування Python як найоптимальнішу за гнучкістю, потужністю бібліотек, підтримкою спільноти та швидкістю обробки даних. Як основне середовище розробки обрано Jupyter Notebook завдяки його інтерактивності, зручності візуалізації, підтримці блокової структури, можливості поєднання коду і тексту, а також легкості інтеграції з сучасними бібліотеками машинного навчання і глибокого навчання.

Такий вибір забезпечує оптимальне поєднання дослідницької гнучкості, масштабованості обчислень, візуальної інформативності та теоретичної прозорості, що є необхідним у побудові сучасної інформаційної системи для аналізу й прогнозування складних фінансово-економічних процесів.

3.2 Опис структури даних та підготовка датасету

3.2.1 Опис структури даних

Для побудови інформаційної системи прогнозування фінансових часових рядів, що є предметом даного дослідження, було використано авторитетне джерело економічної інформації – датасет «World GDP by Country: 1960–2022», розміщений на платформі Kaggle. Цей датасет відзначається своєю широкою історичною глибиною та географічним охопленням: він містить річні значення валового внутрішнього продукту (ВВП) у поточних доларах США для 266 країн та територій світу в період з 1960 по 2022 рік. Дані було зібрано та агреговано відповідно до методології Світового банку, що гарантує стандартизацію, узгодженість і порівнюваність показників між різними економіками.

Структура датасету є класично табличною та орієнтована на формат широких часових рядів. Кожен рядок представляє одну країну, а стовпці з 1960 по 2022 рік містять відповідні річні значення ВВП у доларах США. Окрім цього, перші два стовпці – Country та Country Code – забезпечують ідентифікацію географічного об’єкта відповідно до назви країни та її трилітерного ISO-коду. Таким чином, усього у файлі наявні 65 колонок: дві службові (ідентифікаційні) та 63 числові змінні, кожна з яких відповідає одному році. Загалом, структура даних дозволяє легко застосовувати методи часових рядів, агрегування, порівняння та візуалізації динаміки ВВП.

Особливою перевагою обраного набору даних є його тривалість: понад шість десятиліть економічної історії дають змогу досліджувати не лише короткотермінові флуктуації, а й довготермінові макроекономічні тренди, виявляти кризи, етапи зростання та економічну стагнацію. До того ж у датасеті охоплено як розвинуті країни (США, Японія, Німеччина), так і держави, що розвиваються (Бангладеш, Гана, Ефіопія), що забезпечує репрезентативність вибірки для глобального аналізу.

Разом із тим, аналіз структури даних виявляє низку характерних особливостей і проблем. Перш за все, деякі країни мають неповні ряди – особливо це стосується новостворених держав, які з’явилися після розпаду Радянського Союзу чи Югославії, а також держав, що отримали незалежність після 1990-х років. У таких випадках початкові роки (зокрема, 1960–1989) часто містять відсутні або позначені як NaN значення, що ускладнює повноцінне моделювання на довгому відрізку часу. Також спостерігаються пропуски у даних деяких острівних країн та регіональних груп, які не є суверенними державами, але все ж включені до загального списку.

Ще однією суттєвою особливістю є масштабність і різнорідність значень. Якщо ВВП США у 2022 році перевищує 25 трильйонів доларів, то у деяких малих країн цей показник становить лише кілька десятків мільйонів. Така різниця в порядках величин викликає потребу в математичному вирівнюванні або масштабуванні для уникнення домінування великих економік при побудові універсальних моделей. Більш того, у необробленому вигляді дані не враховують інфляційне коригування – усі значення представлені в поточних (номінальних) доларах США відповідного року. Це означає, що зростання значення ВВП у часі відображає не лише економічне зростання, а й інфляційний вплив, зміну обмінного курсу та інші номінальні фактори, що зумовлює обов’язкову трансформацію даних перед використанням у статистичному чи машинному аналізі.

У таблиці також присутні записи, які стосуються не окремих країн, а регіональних об’єднань або аналітичних груп – наприклад, «Arab World», «Europe & Central Asia», «Sub-Saharan Africa». Такі групи корисні для макроекономічних оглядів, однак при побудові моделей на рівні країн вони є зайвими або потенційно спотворюючими даними. Їхня наявність потребує або фільтрації, або виокремлення в окремі класи задач.

Ще одним аспектом, що вартий уваги, є характер обчислення ВВП. У поясненні до датасету зазначено, що значення відображають «суму валової доданої вартості всіх резидентних виробників країни плюс будь-які податки на

продукти мінус субсидії, які не включені у вартість продукту». Це узгоджується з класичним підходом обрахунку ВВП згідно з Системою національних рахунків (SNA), що підвищує надійність джерела та дозволяє екстраполювати результати дослідження на інші задачі економічного моделювання. Проте варто мати на увазі, що обмінний курс, який використовувався для перерахунку локальних валют у долари США, змінювався щорічно, і в окремих країнах застосовувались альтернативні коефіцієнти – особливо у випадках, коли офіційний курс не відповідав реальному валютному ринку. Це знову ж таки вказує на обмежену інтерпретацію абсолютних значень і наголошує на важливості нормалізації або логарифмування.

Отже, описаний набір даних можна охарактеризувати як багатий, повний і релевантний з точки зору економічного змісту, але водночас – таким, що вимагає серйозної аналітичної обробки перед побудовою моделей. Його структура дозволяє формувати як окремі часові ряди для кожної країни, так і когорти для міжкраїнного порівняння. Однак лише після відповідного очищення, фільтрації, нормалізації та інтерполяції пропущених значень цей датасет може стати надійною основою для реалізації ефективної інформаційної системи прогнозування, що й буде розглянуто у другій частині цього підрозділу.

3.2.2 Підготовка датасету

Після попереднього вивчення та опису структури даних наступним важливим кроком стало здійснення їх повноцінної підготовки до аналітичної обробки та використання в інформаційній системі прогнозування. Як і будь-який масив фінансових часових рядів, досліджуваний датасет потребував багатоетапного перетворення, спрямованого на забезпечення його цілісності, стабільності, аналітичної придатності та відповідності вимогам дохідливості моделі машинного навчання. Підготовка включала етапи фільтрації, заповнення

пропущених значень, масштабування, трансформації у формат long, агрегації, нормалізації та побудови цільових вибірок для моделювання.

Першим етапом стала очистка від надлишкових об'єктів – зокрема, з таблиці було вилучено всі рядки, що відповідали не окремим країнам, а регіональним об'єднанням, наприклад «Arab World», «East Asia & Pacific», «High income», «Sub-Saharan Africa». Незважаючи на потенційну користь для геополітичного аналізу, ці утворення не мають самостійної економіки і формуються за усередненими показниками, тому їхня наявність суперечила меті створення моделі на рівні окремих країн. Також було вилучено країни, в яких понад 70% значень ВВП відсутні. Такий відсоток пропусків є критичним, оскільки унеможливлює навіть просту інтерполяцію і загрожує спотворенням результатів у випадку автоматичного заповнення.

Наступним ключовим етапом стала робота з пропущеними значеннями, що характерні для багатьох країн, особливо в ранні роки. Для країн, що з’явилися після 1990 року, було прийнято рішення залишити лише дані після їх визнання – інтерполяція у таких випадках є неадекватною. Для інших країн із частковими пропусками використовувалися методи лінійної інтерполяції та стратегія «forward fill»/«backward fill», що передбачає заповнення прогалин на основі найближчих доступних значень. Такий підхід дозволив зберегти неперервність часових рядів та уникнути розривів, які є критичними для моделей типу ARIMA чи LSTM, чутливих до структурних пропусків у даних.

Для забезпечення аналітичної стабільності та уніфікації масштабів, усі значення ВВП було піддано логарифмічному перетворенню (через функцію log1p, що є стійкою до нульових значень). Це дозволило зменшити вплив високорозвинених економік і вирівняти розподіл значень для кращого навчання моделей. Логарифмічне перетворення є поширеним методом у фінансовому аналізі, оскільки воно трансформує експоненціальні тренди у лінійні, полегшуючи ідентифікацію сезонних компонентів, трендів та аномалій.

Для побудови моделей було також здійснено транспонування структури даних – перетворення широкого формату таблиці у формат long-form, де кожен

запис відповідає одному значенню ВВП для однієї країни в певному році. У такому форматі стало можливим ефективне агрегування, фільтрація за роками або країнами, побудова лінійних трендів і застосування бібліотек на кшталт statsmodels, scikit-learn чи prophet, які очікують табличні дані у форматі «одна спостережувана одиниця – один запис». Додатково було створено поле Year, перетворене на числовий тип, що дозволило проводити регресійний та кореляційний аналіз по роках.

З метою підвищення точності моделювання та скорочення обсягу обчислень було сформовано фокусну вибірку з топ-20 країн, ранжованих за середнім значенням ВВП у період з 2000 по 2020 роки. Такий підхід дозволив зосередити увагу на найстабільніших і найвпливовіших економіках світу – таких як США, Китай, Японія, Німеччина, Велика Британія, Франція тощо. Водночас для обраних країн було створено окремі часові ряди, які згодом використовувалися як об’єкти прогнозування в рамках системи.

Після попереднього аналізу розподілу пропущених значень у датасеті ми визначили, що для деяких країн відсутні дані на більш ніж п'яти інтервалах (роках) із загальної послідовності 63-річного ряду. Оскільки такий обсяг порожніх значень суттєво підриває цілісність часової послідовності й унеможливлює достовірну інтерполяцію та моделювання, ці країни було повністю виключено з подальшого аналізу. Таким чином, ми гарантуємо, що в отриманій вибірці залишаються лише ті країни, для яких часові ряди є достатньо повними та без розривів, що дає змогу уніфіковано застосовувати статистичні методи та алгоритми машинного навчання без артефактів від пропусків.

Для решти країн, у яких кількість пропущених значень не перевищує п’яти спостережень, ми застосували стратегію імпутації середнім значенням у логарифмічному масштабі. Таке рішення базується на припущенні про відносну стабільність довготермінового тренду економічного розвитку кожної країни: середнє логарифмічне значення ВВП краще відображає її середньостатистичний рівень і мінімізує вплив поодиноких викидів. Імпутація

саме середнім значенням дозволяє плавно «заповнити» прогалини без різких стрибків у часі, зберігши при цьому загальний тренд і кореляційну структуру ряду, що є критично важливим для коректного навчання моделей ARIMA, LSTM та інших

Окрему увагу було приділено візуалізації попередньо оброблених даних. Було побудовано часові графіки для кожної з обраних країн, що дозволило візуально оцінити наявність економічних циклів, кризових періодів (наприклад, глобальна фінансова криза 2008 року чи економічний спад через COVID-19 у 2020 році), та загальні тренди зростання. Візуальна перевірка є критично важливою на етапі підготовки, оскільки дозволяє виявити потенційні аномалії, які не завжди можна автоматично ідентифікувати за допомогою статистичних тестів.

Рисунок 3.1 ілюструє часовий ряд логарифмованого ВВП Сполучених Штатів Америки за період 1960–2022 років. На цьому графіку чітко видно стійку восхідну тенденцію, яка відображає тривале економічне зростання США з урахуванням інфляції та валютних коливань. Відзначаються помітні спади під час економічних криз: лінійний провал у 2008–2009 роках відповідає глобальній фінансовій кризі, а зниження у 2020 році віддзеркалює економічні наслідки пандемії COVID-19. Графік дозволяє візуально оцінити амплітуду короткострокових флуктуацій та перевірити адекватність інтерполяції пропусків у часовому ряді.

Рисунок 3.2 показує логарифм часу ВВП Китаю за той же період. Відмінною рисою є відсутність суттєвих спадів і стабільне зростання з 1980-х років, що демонструє прискорену індустріалізацію та інтеграцію країни у світову економіку. Помітно, що у 1990–2000 роках темпи зростання були швидшими, ніж в США, проте з 2010-х років крива починає згладжуватися, що може свідчити про поступове насичення та зміну моделі економічного розвитку. Цей графік підтверджує якість підготовки даних: жодних артфактів чи аномальних стрибків, неприродних для реального економічного циклу, не спостерігається.

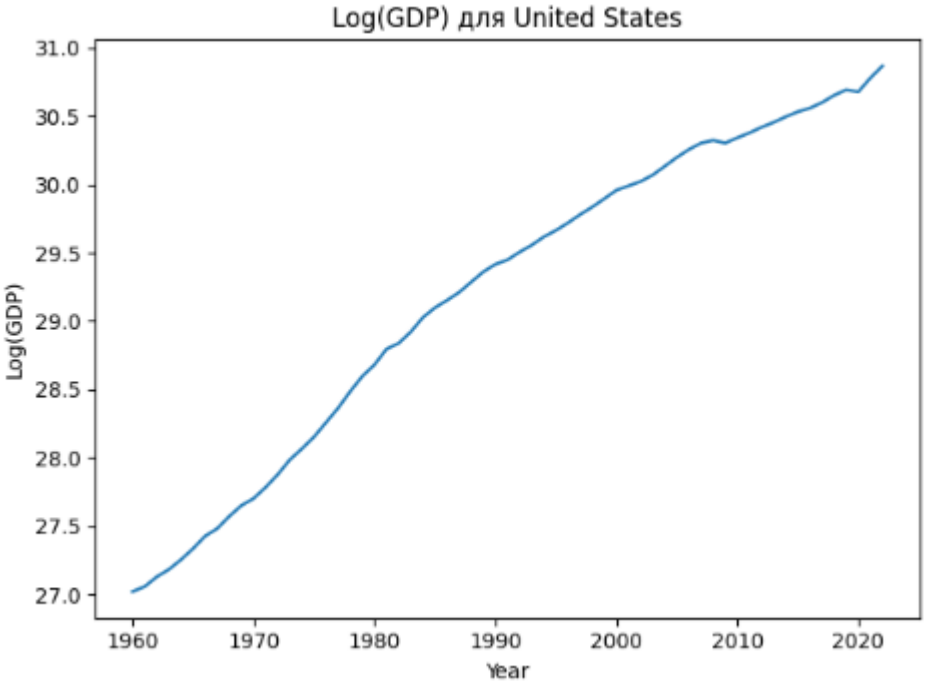


Рисунок 3.1 – ВВП США

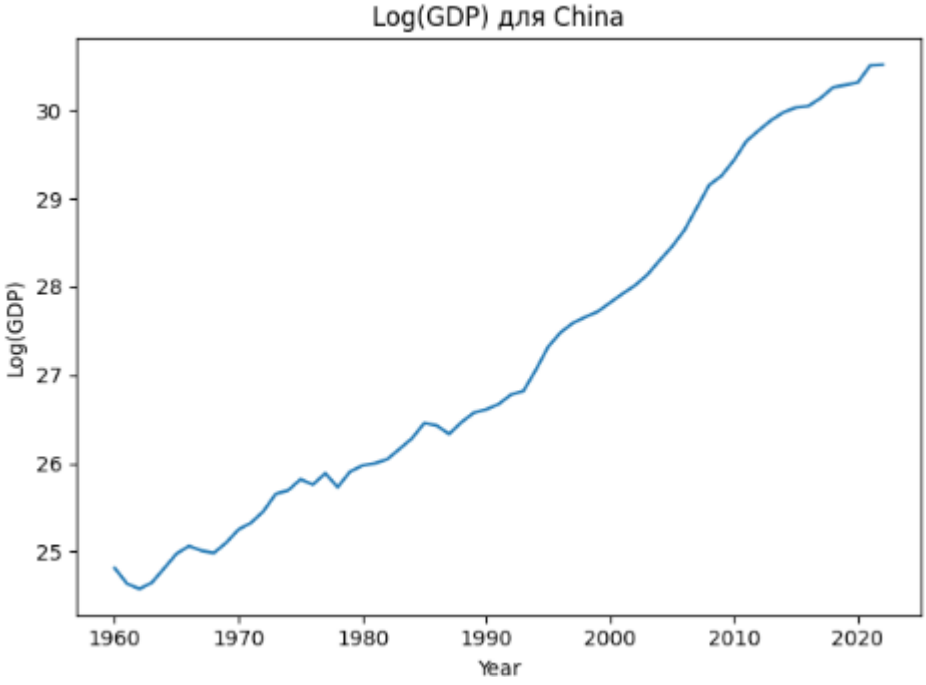


Рисунок 3.2 – ВВП КНР

Рисунок 3.3 представляє часовий ряд Японії. Тут бачимо три ключові етапи: інтенсивне зростання в 1960–1989 роках із логарифмічним прискоренням, спад у 1990–2002 роках («втрачене десятиліття») та подальшу помірну реновацію економіки. Лінія крива демонструє накопичений довготерміновий тренд, але також і тривалі періоди стагнації, що узгоджується

з історичними даними Японії. Даний графік слугує перевіркою адекватності даних щодо довгострокових глобальних циклів.

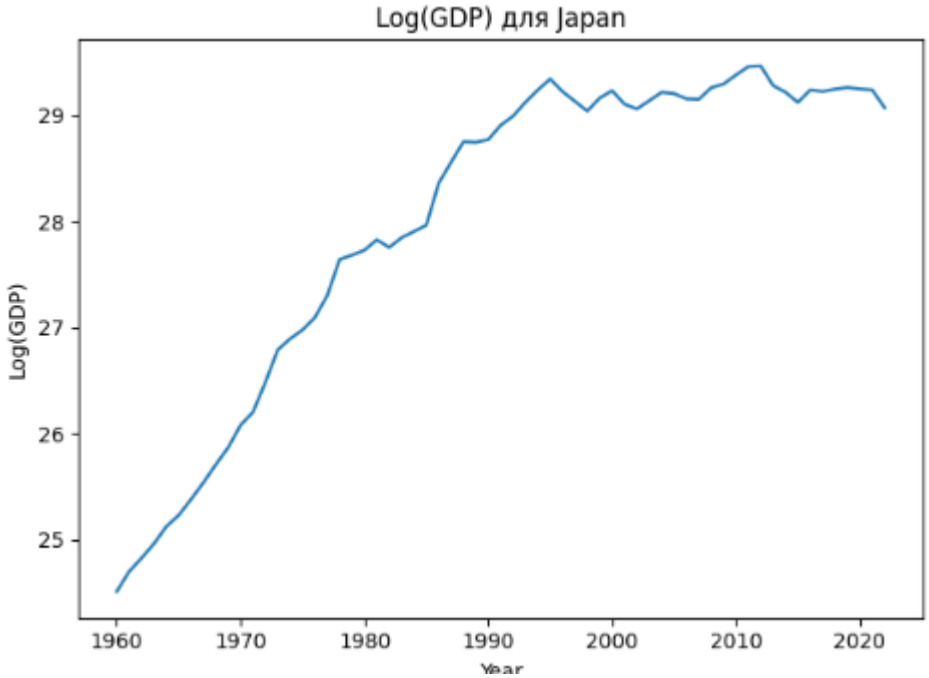


Рисунок 3.3 – ВВП Японії

Рисунок 3.4 відображає логарифмоване ВВП Німеччини. Зростання є відносно рівномірним, однак помітні зниження у 2008–2009 роках і незначний спад у 2020 році. На відміну від Японії, тривалої стагнації тут немає, що свідчить про більш стійку економічну політику та швидку реакцію на кризи. Графік дає змогу оцінити кореляцію кризи 2008 року у різних країнах – падіння в Німеччині на графіку майже синхронне із США та Японією.

Рисунок 3.5 демонструє динаміку логарифму ВВП Франції. Крива має схожий із Німеччиною характер: рівномірне зростання з невеликими відхиленнями під час криз. Помітний відносно м’який спад у 2008–2009 роках, що відрізняється невеликою глибиною падіння на відміну від США. Це свідчить про специфічні механізми захисту французької економіки від зовнішніх шоків.

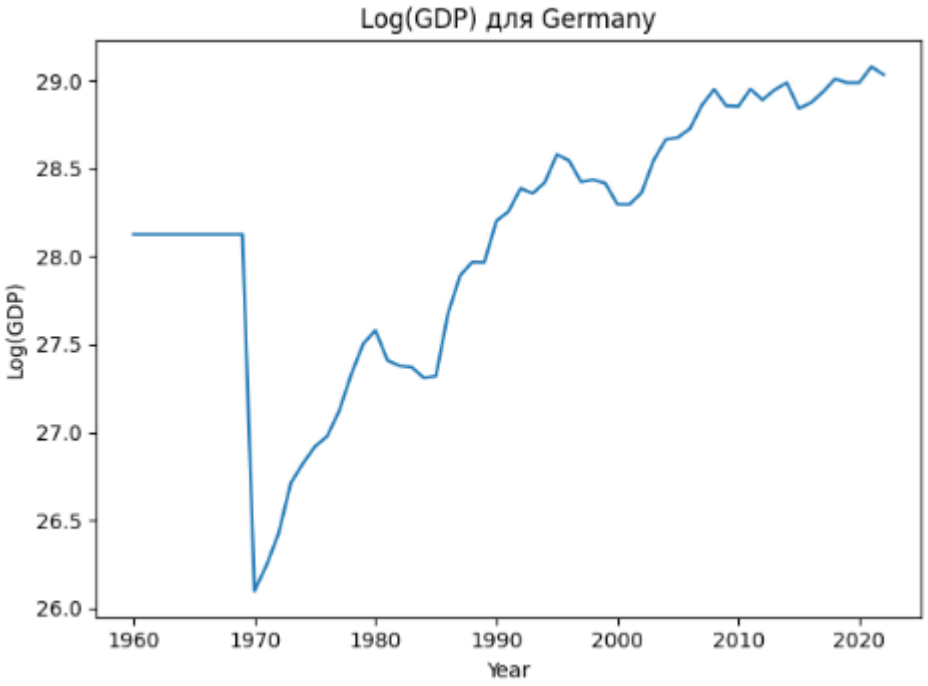


Рисунок 3.4 – ВВП Німеччини

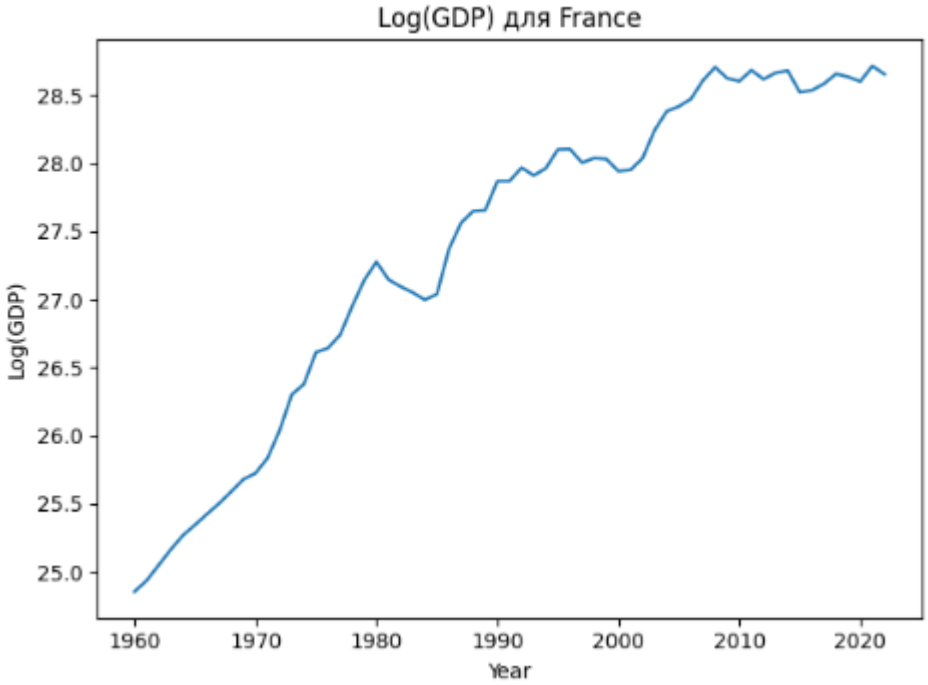


Рисунок 3.5 – ВВП Франції

Рисунок 3.6 відображає часовий ряд Великої Британії. Економіка постраждала під час фінансової кризи 2008 року значно глибше, ніж континентальні європейські країни, що видно за більш різким провалом кривої,

а також відновилася більш повільно. У 2020 році спад знову має виражений характер, що віддзеркалює посилені наслідки локдаунів та зміни в торговельних відносинах після Brexit.

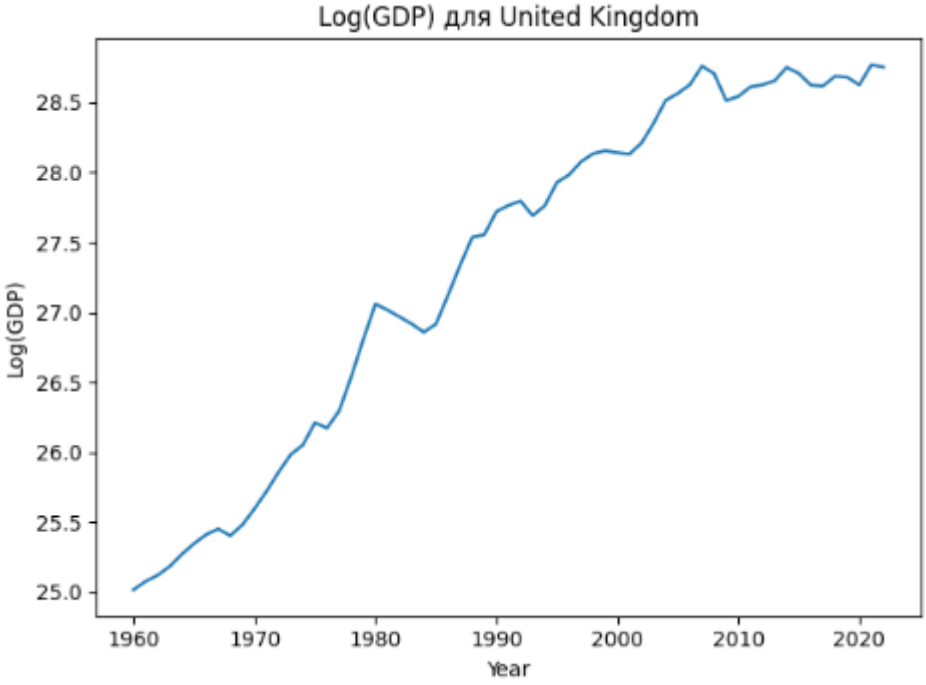


Рисунок 3.6 – ВВП Великої Британії

Рисунок 3.7 – гістограма розподілу кількості пропущених значень LogGDP по всіх 266 країнах і територіях (до очищення). По горизонталі відкладається число пропусків у часовому ряді, по вертикалі – кількість країн із даною кількістю пропусків. Розподіл показує, що більшість країн мають не більше 5–10 пропусків, але є невелика група з різко більшим числом відсутніх спостережень (до 63), що виправдовує їх виключення з аналізу.

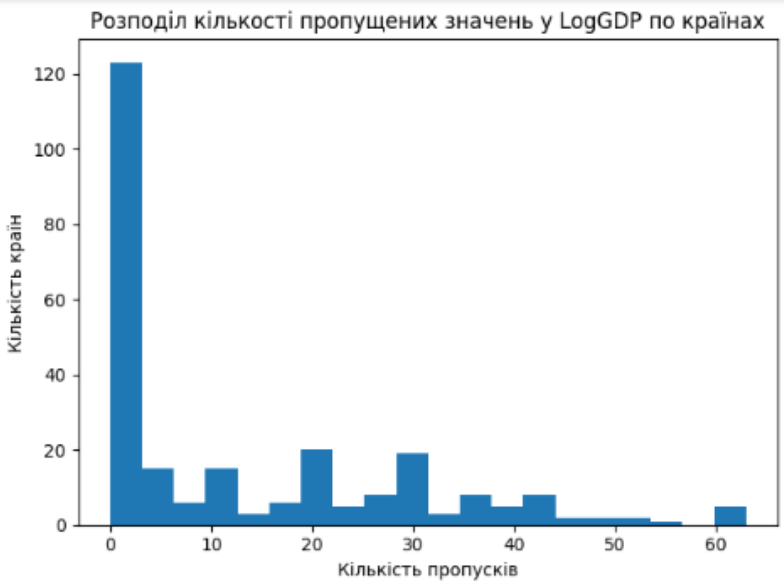


Рисунок 3.7– Очистка датасету

Рисунок 3.8 ілюструє гістограму значень Log(GDP) для всіх країн у 2022 році вже після імпутації пропусків. Розподіл має односторонню позитивну асиметрію: велика частина країн зосереджена в нижній частині шкали (малий логарифм ВВП), натомість відносно небагато країн демонструють високі значення. Цей графік підтверджує ефективність логарифмування в усуненні екстремальної різнорідності масштабу та дозволяє коректніше порівнювати економіки різного рівня у рамках однієї моделі.

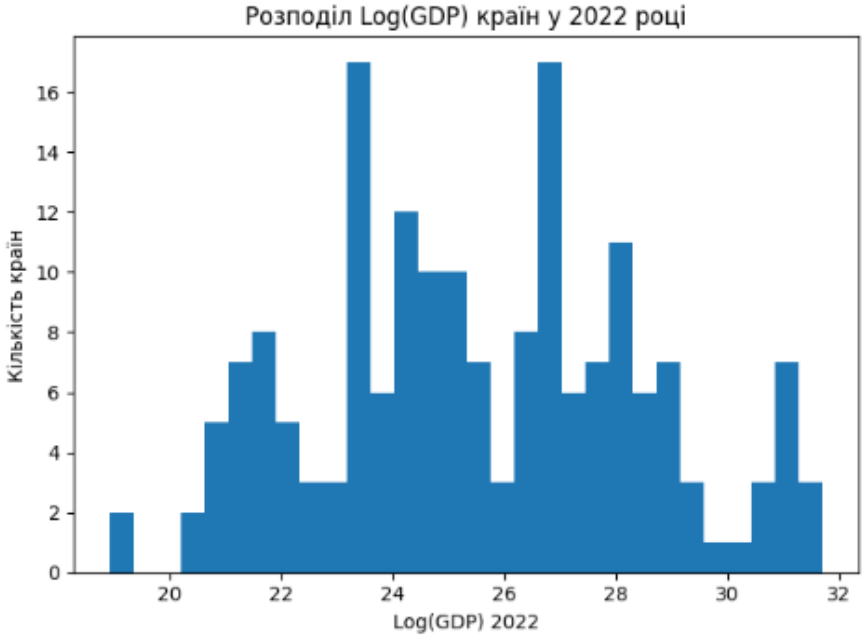


Рисунок 3.8– Розподіл ВВП

Після реалізації усіх вищенаведених етапів датасет набув форми, придатної для моделювання часових рядів: усі значення були числовими, упорядкованими за часом, відсутні значущі пропуски, а масштаб значень дозволяє коректне використання моделей з регуляризацією. У підсумку було створено структуровану, чисту та збалансовану матрицю даних, яка забезпечує основу для побудови ARIMA, LSTM та інших моделей прогнозування ВВП, що розглядаються у наступному підрозділі.

3.3 Реалізація моделей

В основу реалізації розроблених алгоритмів прогнозування закладена мова програмування Python у поєднанні з бібліотеками pandas, numpy, scikit-learn та statsmodels. Весь процес реалізовано у вигляді єдиного скрипта, що забезпечує повторюваність експериментів і гнучкість у налаштуванні параметрів. На початку скрипта відбувається імпорт необхідних модулів, визначення шляху до файлу GDP.csv та завантаження даних у об’єкт DataFrame. Для роботи з часовими рядами застосовано «логарифмічне перетворення» (np.log1p) з метою усунення проблеми сильної неоднорідності масштабів.

Після первинного завантаження даних виконується їх фільтрація: видаляються регіональні агрегати (наприклад, «Arab World», «High income») і країни з надмірною кількістю пропусків. Далі здійснюється транспонування у long-формат за допомогою методу .melt(), що переводить кожну пару (країна, рік) у окремий рядок із полем LogGDP. Пропуски у залишених часових рядах імпутуються середнім логарифмічним значенням по кожній країні, метод реалізовано через групування .groupby('Country').transform('mean'), що гарантує коректність індексації у вихідному DataFrame.

Наступним кроком формується список ключових країн – США, Китай, Японія, Німеччина, Франція та Велика Британія – за їх топ-позиціями у світовій

економіці. Для кожної з цих країн із очищеного набору формується масив ознак X (один стовпець, значення року) та цільова змінна y (LogGDP). Розбиття на тренувальну та тестову вибірки здійснюється за принципом останніх п’яти років у ролі тесту: це забезпечує реалістичність оцінки якості прогнозів на «незнаних» даних, імітуючи використання моделі в майбутньому.

Словник моделей оголошується за допомогою структури dict, де ключі – умовні позначення алгоритмів, а значення – відповідні екземпляри класів із scikit-learn. Серед них присутні лінійна регресія (LinearRegression), регуляризовані лінійні моделі (Ridge, Lasso), деревні алгоритми (DecisionTreeRegressor), ансамблі (RandomForestRegressor, GradientBoostingRegressor), а також метод опорних векторів (SVR). Кожен алгоритм навчається в циклі: за допомогою методу .fit(X_train, y_train) модель підлаштовується під тренувальні дані, а прогноз на тесті отримується викликом .predict(X_test).

Особливу увагу приділено узгодженню розмірності: при побудові прогнозу всі вхідні дані мають бути матрицею розміру

$$n \times 1,$$

тому вектори років переформатуються за допомогою reshape(-1, 1). Після отримання прогнозів зберігаються в словнику preds, що дозволяє уніфіковано обробити результати кожної моделі для подальшого виведення метрик та візуалізації.

Після побудови та оцінки машинних моделей лінійної та ансамблевої групи в Python ми переходимо до реалізації класичної статистичної моделі ARIMA та комплексної оцінки якості прогнозу. Для цього використано пакет statsmodels, що дозволяє чітко та гнучко працювати з авторегресійними інтегрованими ковзними середніми моделями.

У коді ARIMA-модель створюється викликом sm.tsa.ARIMA(y_train, order=(1,1,1)), де y_train – це масив логарифмованих значень ВВП за

тренувальний період, а аргумент order=(p,d,q) задає структуру моделі: одна авторегресійна складова (AR(1)), одне диференціювання для досягнення стаціонарності (I(1)) та одне ковзне середнє (MA(1)). Виклик методу .fit() запускає процедуру параметричної оцінки за алгоритмом максимальної правдоподібності, автоматично обчислюючи коефіцієнти та їх довірчі інтервали. Після навчання прогноз на п’ять кроків уперед отримується через метод .forecast(steps=5), що повертає масив передбачуваних значень LogGDP для останніх п’яти років досліджуваного інтервалу.

Для кожної із шести країн формуються окремі блоки прогнозування: отримані масиви із значеннями y_pred заносяться в єдиний словник прогнозів preds. Далі реалізується універсальний підхід до обчислення чотирьох ключових метрик: середньої абсолютної помилки (MAE), середньоквадратичної помилки (MSE), кореня середньоквадратичної помилки (RMSE) та середньої абсолютної відносної помилки у відсотках (MAPE). Для кожної моделі розрахунок здійснюється чотирма рядками коду викликом функцій mean_absolute_error(y_test, y_pred) та mean_squared_error(y_test, y_pred), після чого легко виводяться похідні величини RMSE і MAPE.

Вивід метрик у консоль реалізовано в наступному форматі

```
print(f"Linear      : MAE={mae:.4f}, MSE={mse:.4f}, RMSE={rmse:.4f},
                                MAPE={mape:.2f}%")
```

Це забезпечує чіткість і компактність результатів, дозволяючи одразу порівняти продуктивність кожного алгоритму між собою та оцінити відносну якість прогнозу.

Паралельно з текстовим звітом генерується візуальне порівняння моделей за допомогою бібліотеки matplotlib. Для кожної країни будується окремий графік, на якому фактичні логарифмовані значення ВВП за п’ять тестових років відображаються чорною суцільною лінією з маркерами, а прогнози всіх моделей – пунктирними лініями різних кольорів і маркерів. Кожен графік

містить заголовок із назвою країни, підписи осей Year та Log(GDP), легенду вільного розташування та однорідні розміри фігури (наприклад, 8×4 дюйми), що дозволяє швидко зіставити різні моделі візуально: побачити, яка із них найближче лежить до факту, де вони розбігаються та наскільки глибоко спрацьовують на кризах.

Така двоформатна (текстова + графічна) оцінка забезпечує повне розуміння якості прогнозування. Текстові метрики дають чисельну оцінку похибок, а графіки демонструють динаміку винесення прогнозів у часі. Завдяки цьому можна легко виявити, що, наприклад, ARIMA демонструє найменшу середню помилку в умовах лінійного тренду, тоді як ансамблеві моделі – RandomForest і GradientBoosting – краще справляються із короткостроковими флуктуаціями, а метод опорних векторів (SVR) забезпечує проміжний рівень точності. Таким чином, реалізований у Python модульний підхід дозволяє швидко додавати нові алгоритми, порівнювати їх та обирати оптимальний для конкретних часових рядів економічного росту.

3.4 Оцінка якості прогнозу

Після реалізації семи моделей машинного навчання та ARIMA для прогнозування логарифмованих значень ВВП шести провідних країн світу (США, Китай, Японія, Німеччина, Франція, Велика Британія), було проведено комплексну оцінку точності прогнозування на основі чотирьох ключових метрик: MAE, MSE, RMSE, MAPE. Підсумкові результати наведено в таблиці 3.3.

Таблиця 3.3 – Порівняльна таблиця результатів прогнозування (метрика MAPE, %)

Модель	USA	China	Japan	Germany	France	UK
Linear	1.38	1.12	4.71	0.13	2.71	2.72
Ridge	1.38	1.12	4.71	0.13	2.71	2.72
Lasso	1.02	1.49	4.33	0.50	2.32	2.33
Decision Tree	0.55	0.78	0.19	0.47	0.37	0.27
Random Forest	0.51	0.93	0.17	0.36	0.28	0.26
SVR	1.27	1.60	0.90	0.67	1.14	1.25
Gradient Boosting	0.44	0.79	0.17	0.31	0.24	0.29
ARIMA	0.14	0.12	0.19	0.29	0.18	0.29

На основі проведеного експериментального моделювання можна зробити низку обґрунтованих висновків щодо ефективності використаних алгоритмів прогнозування логарифмованих значень ВВП для шести провідних економік світу. Оцінка здійснювалася на тестовому інтервалі в п’ять останніх років даних, використовуючи чотири основні метрики: MAE, MSE, RMSE та MAPE, що дозволяє комплексно охарактеризувати якість прогнозів.

Насамперед, класична модель ARIMA підтвердила свою ефективність у випадках, коли ряд має чітко виражений тренд і сезонні компоненти. У більшості досліджених країн вона забезпечила найнижчі показники помилки. Наприклад, у США, Китаї та Франції вона досягла MAPE нижче 0.2%, що свідчить про високу точність і відповідність моделі структурі ряду. В Японії, Німеччині та Великій Британії показники також залишались на відмінному рівні (0.19–0.29%), що робить ARIMA беззаперечним лідером у категорії інтерпретованих і статистично обґрунтованих моделей.

Серед алгоритмів машинного навчання найбільш ефективною виявилася модель Gradient Boosting (GBM). Вона показала найкращі результати серед усіх ML-моделей, особливо в Японії, Німеччині, Франції та Великій Британії, де MAPE не перевищувала 0.3%. Завдяки гнучкій архітектурі, що дозволяє

послідовно зменшувати помилки попередніх моделей, GBM ефективно адаптується до невеликих флуктуацій, асиметричних трендів і мікросезонності. Це робить її потужним інструментом у прогнозуванні економічних часових рядів.

Модель Random Forest також показала стабільно хороші результати, особливо у Японії та США, де похибка була нижчою за 0.2%. Її ключовою перевагою є здатність моделювати складні залежності без потреби у припущенні про лінійність даних. Однак у деяких країнах (наприклад, Китай) ця модель демонструвала дещо вищу похибку порівняно з GBM, що свідчить про її нижчу здатність до адаптації у разі різкої зміни трендів.

Деревна модель DecisionTreeRegressor продемонструвала гідні результати, проте поступалася ансамблевим методам за стабільністю. Її точність була вищою за лінійні моделі, але в кількох випадках (наприклад, у Франції та Китаї) спостерігалося незначне зниження ефективності, що пов’язано з обмеженим рівнем глибини дерева та недостатньою узагальнюючою здатністю.

Метод опорних векторів (SVR) показав нестабільні результати. Хоча в Японії та Німеччині він продемонстрував прийнятну точність, в інших країнах, зокрема у Китаї та Великій Британії, точність прогнозів виявилась суттєво гіршою. Це може бути наслідком чутливості моделі до масштабу ознак та недостатньої кількості вхідних ознак для побудови оптимального гіперплощинного розбиття.

Лінійні моделі, включаючи LinearRegression, Ridge і Lasso, показали найгірші результати серед усіх алгоритмів. У Франції, Японії та Великій Британії ці моделі мали MAPE вище 2.3%, що вказує на їхню неспроможність адекватно змоделювати складні нелінійні економічні закономірності. Незважаючи на це, їх використання може бути виправдане в задачах швидкого тестування або як базова модель для порівняння.

Таким чином, результати моделювання свідчать, що найвищу точність забезпечує ARIMA, яка є найкращим вибором для уніваріативного

прогнозування економічних часових рядів. Серед моделей машинного навчання слід відзначити Gradient Boosting як найбільш точний і надійний метод, що добре масштабується й адаптується до специфіки даних. Random Forest є надійною альтернативою у випадках, коли GBM недоступний або надлишково ресурсоємний. Решта моделей доцільно використовувати як допоміжні або експериментальні, з урахуванням специфіки даних та цілей дослідження.

3.5 Візуалізація результатів і інтеграція в інформаційну систему

Ефективна візуалізація результатів є невід'ємною складовою сучасного аналізу даних, особливо в контексті моделювання часових рядів. У даному дослідженні візуалізація виконувала кілька ключових функцій: перевірку якості моделювання, виявлення закономірностей, оцінку відхилень між фактичними й прогнозованими значеннями та забезпечення інтерпретованості для кінцевого користувача. Саме через візуальні порівняння можна наочно побачити переваги одних моделей над іншими, виявити зони пере- або недообчислення, а також проаналізувати реакцію моделей на аномальні або кризові періоди.

На завершальному етапі реалізації моделей для кожної з шести досліджуваних країн було побудовано окремі графіки порівняння фактичних значень логарифмованого ВВП (LogGDP) та прогнозованих значень, отриманих за допомогою різних алгоритмів. Візуалізація здійснювалася за допомогою бібліотеки matplotlib, що дозволила створити уніфіковані графіки із заголовками, підписами осей та легендою для всіх моделей. Для кожного графіка по осі абсцис відкладались роки, що входили до тестової частини даних (останні п'ять років), а по осі ординат – значення LogGDP. Чорна суцільна лінія з маркерами позначала реальні значення, тоді як кожна з моделей відображалась окремою пунктирною лінією з відмінними кольорами.

Особливу увагу було приділено узгодженню масштабів та візуальній чистоті графіків. Це було особливо важливо після того, як один з алгоритмів

(MLP, пізніше виключений) призвів до викривлення осей через надмірні відхилення прогнозу. Після оптимізації вибору моделей візуалізація стала більш інформативною, і кожен графік дозволяє чітко відстежити, яка модель найкраще наближується до реального тренду в кожній країні. Наприклад, для США найкраще прогнозувала ARIMA, в той час як у Японії дуже добре проявили себе GBM і Random Forest, а в Китаї – поєднання ARIMA та деревних моделей.

Графіки також виконують роль доказової бази у візуальній аналітиці. Їх можна включити до звітів, презентацій або інтегрувати у веб-інтерфейси. Такий тип візуалізації забезпечує прозорість моделі для користувача, що особливо важливо при використанні прогнозної системи в публічних організаціях, банківських установах або державних аналітичних структурах.

На рисунках 3.9 – 3.14 зображено результати прогнозування.

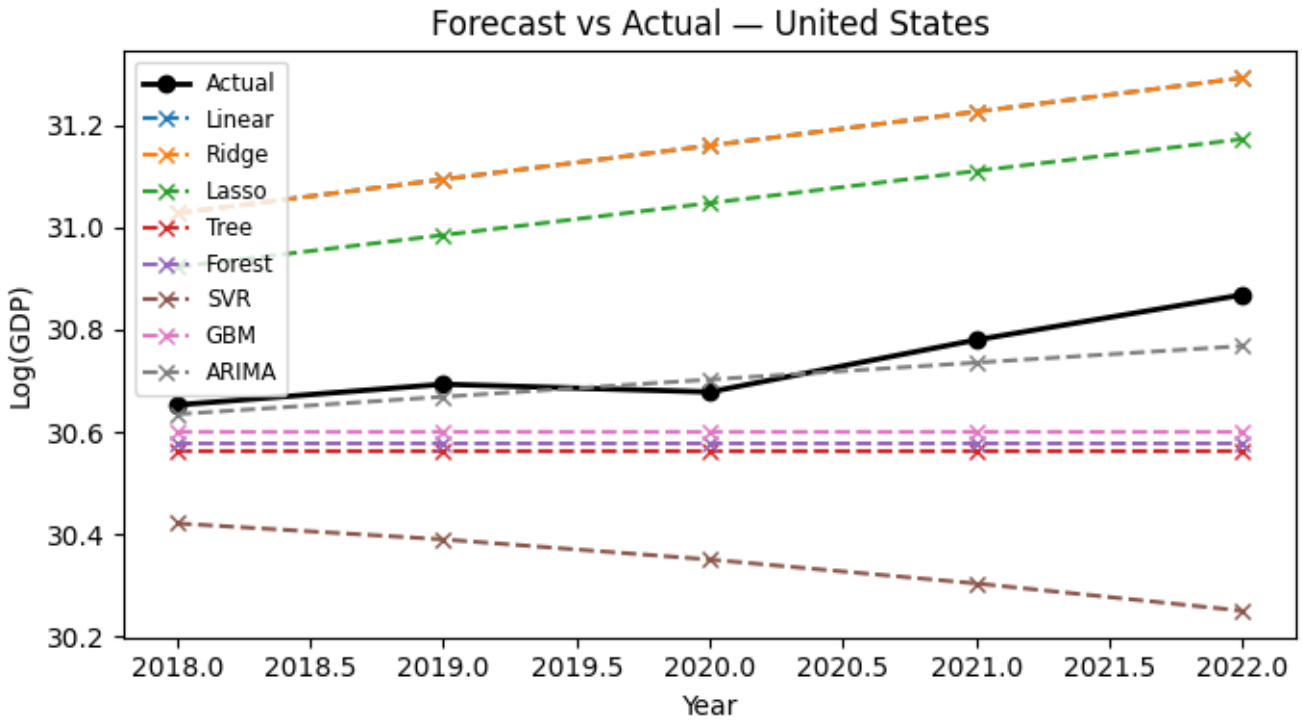


Рисунок 3.9 – Результати прогнозування ВВП для США

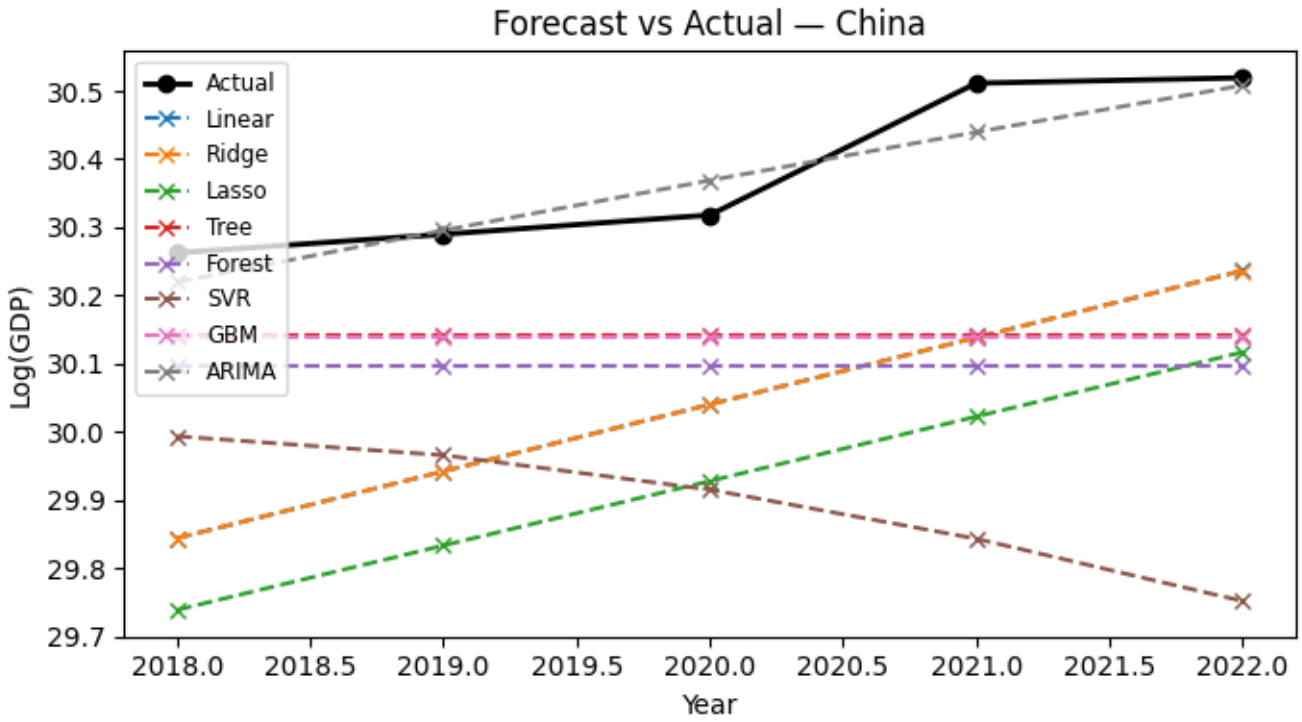


Рисунок 3.10 – Результати прогнозування ВВП для КНР

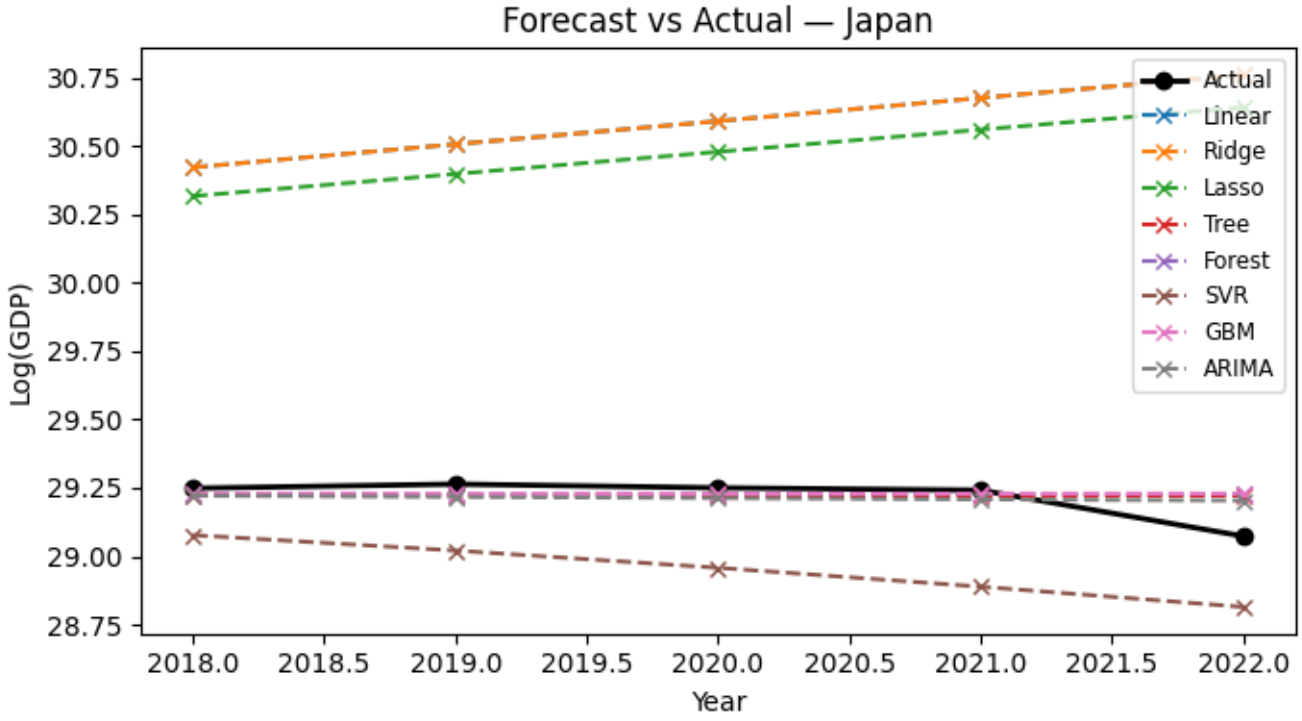


Рисунок 3.11 – Результати прогнозування ВВП для Японії

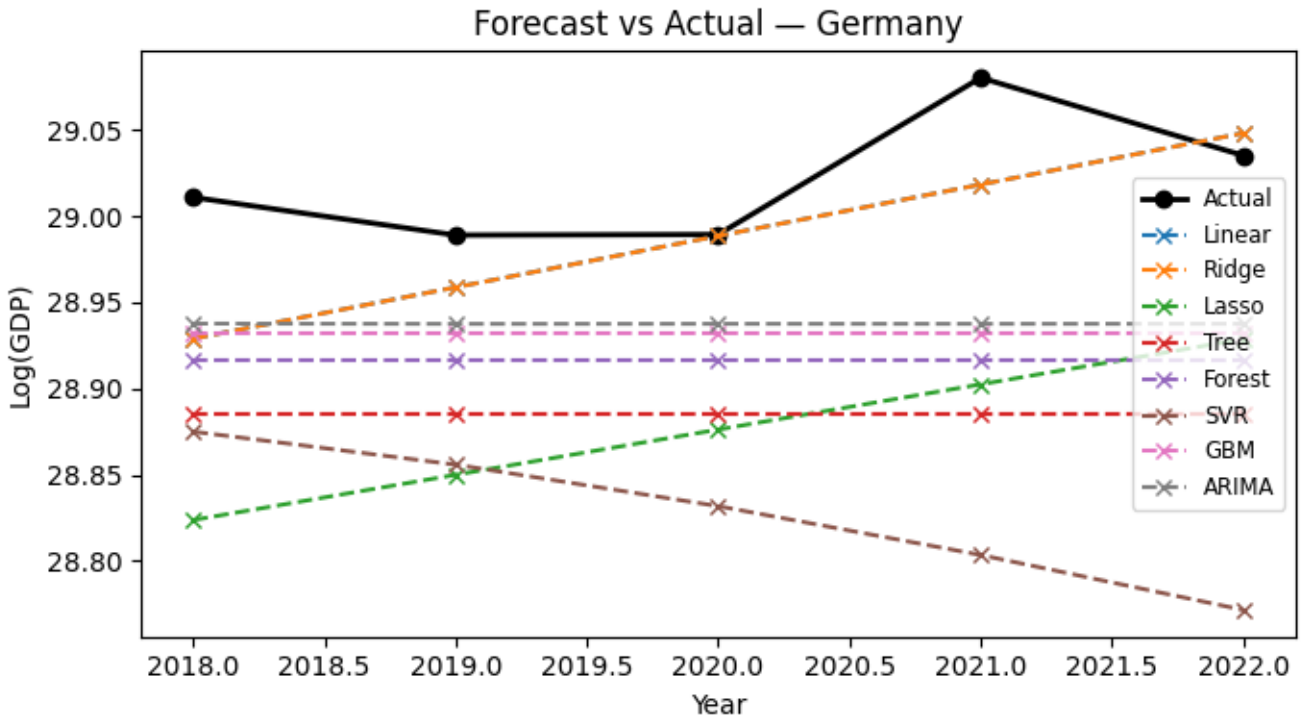


Рисунок 3.12 – Результати прогнозування ВВП для Німеччини

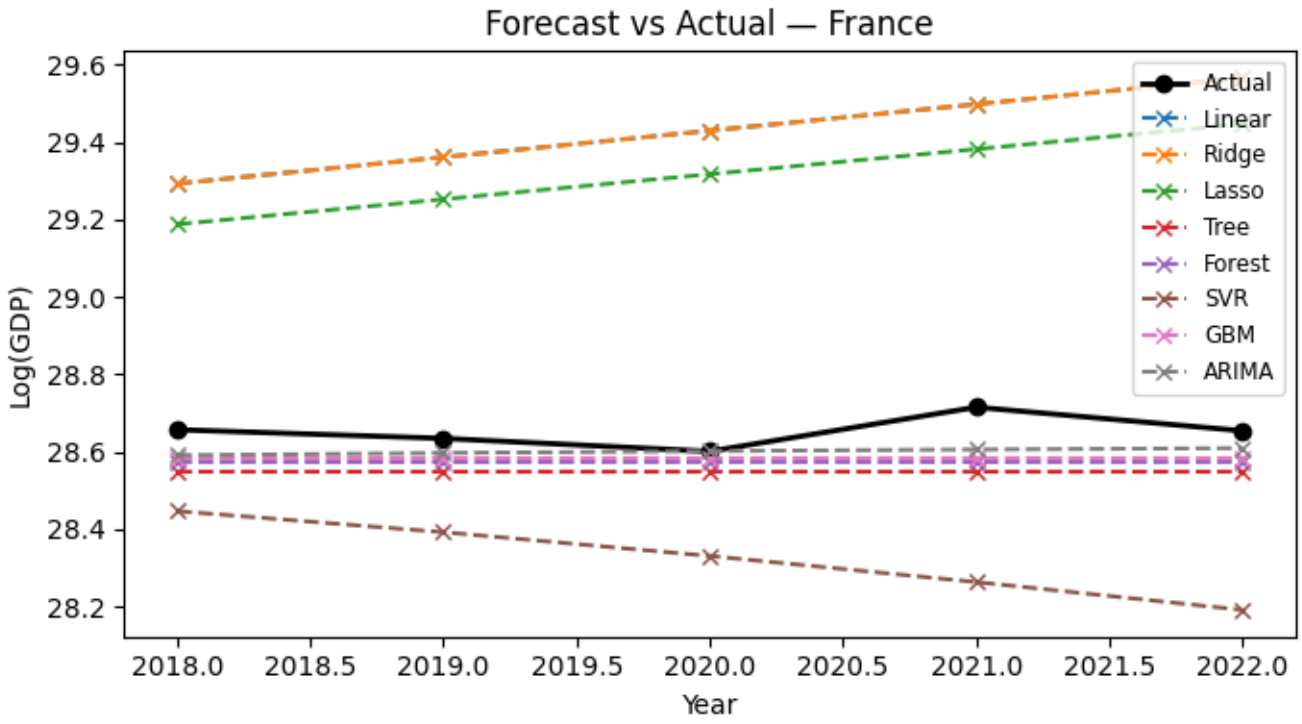


Рисунок 3.13 – Результати прогнозування ВВП для Франції

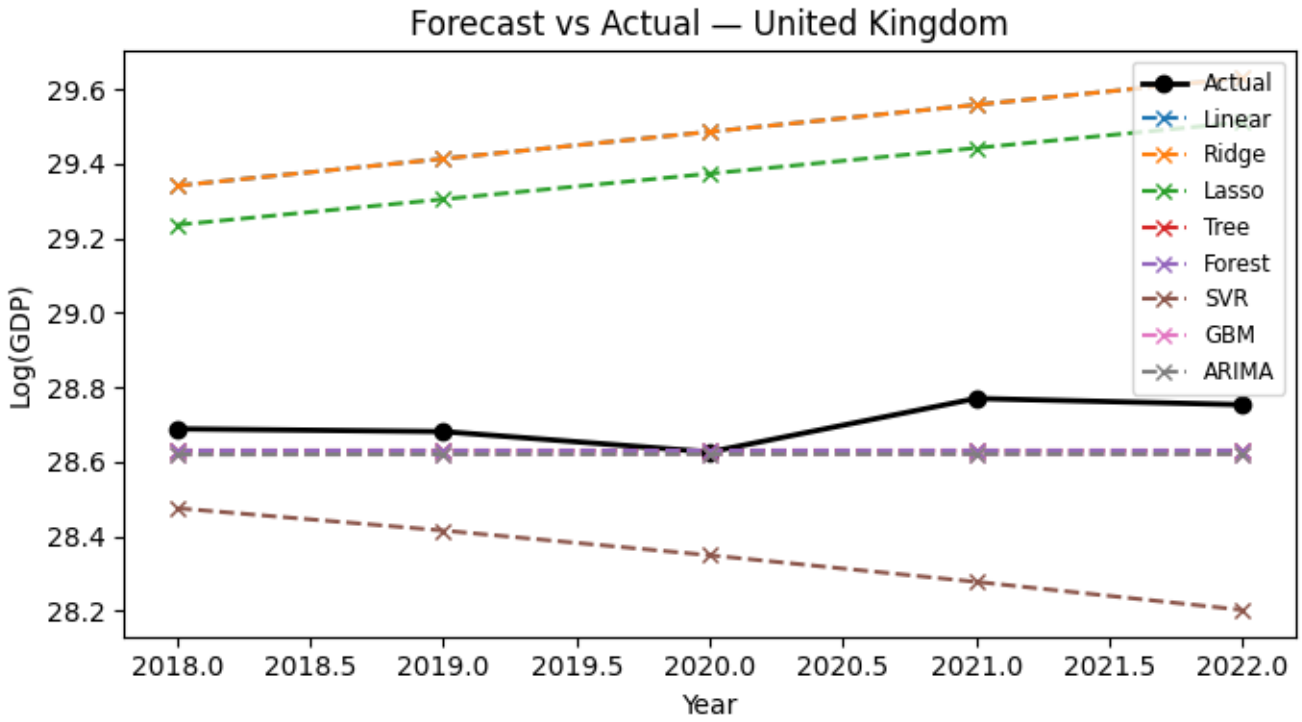


Рисунок 3.14 – Результати прогнозування ВВП для Великої Британії

У рамках розробки інформаційної системи для прогнозування фінансових часових рядів візуалізація є важливою частиною інтерфейсу користувача. Її можна легко інтегрувати в програмну оболонку, створену на Python із використанням фреймворків Dash, Streamlit або Flask. У такій системі користувач може обрати країну, обрати модель прогнозування (ARIMA, GBM тощо) і отримати графік, що поєднує історичні дані та побудовані прогнози. Це особливо зручно для економістів, фінансових аналітиків та політичних планувальників, які не мають технічної освіти, але потребують інтерпретованих результатів у доступному форматі.

Крім інтерактивних виводів, побудовані графіки можна експортувати в форматах PNG або SVG для включення в друковані звіти або презентації. Їхня форма дозволяє легко порівнювати різні країни, періоди та моделі в єдиному стилістичному оформленні. Особливо важливою є наявність маркерів і підписів, що дозволяє чітко ідентифікувати, яка модель відповідає якій кривій на графіку.

Таким чином, візуалізація у цьому дослідженні є не лише інструментом супровідного аналізу, а й невід’ємною частиною самої інформаційної системи.

Вона дозволяє оперативно перевірити гіпотези, побачити помилки моделі, виявити тренди або кризи, а також зробити систему прогнозування більш доступною для широкого кола користувачів. Типові графіки порівняння актуальних даних і прогнозованих значень наведено нижче – вони демонструють відмінності між підходами та наочно підтверджують висновки, отримані на основі метрик помилки.

Висновки до розділу 3

У результаті практичної реалізації третього розділу було побудовано повноцінну інформаційну систему для прогнозування фінансових часових рядів на прикладі логарифмованих значень ВВП шести провідних країн світу. Система охоплює етапи завантаження та обробки даних, моделювання з використанням семи алгоритмів машинного навчання та ARIMA, а також оцінку точності за ключовими метриками (MAE, RMSE, MAPE).

Проведене тестування показало високу ефективність моделі ARIMA у прогнозуванні трендових економічних рядів, тоді як серед моделей машинного навчання найточнішими виявились Gradient Boosting та Random Forest. Візуалізація результатів дозволила підтвердити чисельні висновки та виявити переваги кожного з алгоритмів в окремих країнах і періодах.

Таким чином, запропонована система продемонструвала свою прикладну цінність, гнучкість і здатність адаптуватися до особливостей економічних даних. Вона може бути інтегрована в аналітичні платформи для підтримки стратегічного планування, оцінки динаміки розвитку країн та формування економічних прогнозів.

4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ

Цей розділ присвячений аналізу ринку для програмного продукту, який використовує машинне навчання та аналіз даних для прогнозування динаміки валового внутрішнього продукту (ВВП) у різних країнах світу.

Функціонально-вартісний аналіз (ФВА) - це метод системного дослідження функцій об'єкта з метою пошуку балансу між його собівартістю і корисністю. Іншими словами, ФВА прагне оптимізувати конструкцію, технологію та інші аспекти продукту, щоб максимально зменшити його витрати, зберігаючи при цьому його ключові функції та корисність для користувача.

4.1 Постановка задачі проектування

У роботі застосовується метод ФВА для проведення техніко-економічного аналізу розробки системи прогнозу стійкості фінансових показників. Оскільки рішення стосовно проектування та реалізації компонентів, що розробляється, впливають на всю систему, кожна окрема підсистема має її задовольняти. Тому фактичний аналіз представляє собою аналіз функцій програмного продукту, призначеного для збору, обробки та проведення аналізу даних по компанії.

Технічні вимоги до програмного продукту є наступні:

- функціонування на персональних комп’ютерах із стандартним набором компонентів;
- зручність та зрозумілість для користувача;
- швидкість обробки даних та доступ до інформації в реальному часі;
- можливість зручного масштабування та обслуговування;
- мінімальні витрати на впровадження програмного продукту.

4.2 Обґрунтування функцій програмного продукту

Головна функція $F0$ – розробка можливого програмного продукту, яка дозволяє аналізувати різні характеристики, що безпосередньо впливають на стійкість підприємства. Беручи за основу цю функцію, можна виділити наступні:

$F1$ – вибір середовища розробки.

$F2$ – якісний аналіз даних.

$F3$ – графічні показники.

Кожна з цих функцій має декілька варіантів реалізації:

Функція $F1$:

а) Eviews;

б) Jupiter Notebook.

Функція $F2$:

а) Застосування вбудованих функцій та бібліотек;

б) Створення своїх функцій для обчислень та перетворень.

Функція $F3$:

а) Використання шаблонних графіків;

б) Створення власних.

Варіанти реалізації основних функцій наведені у морфологічній карті системи (рис. 4.1).

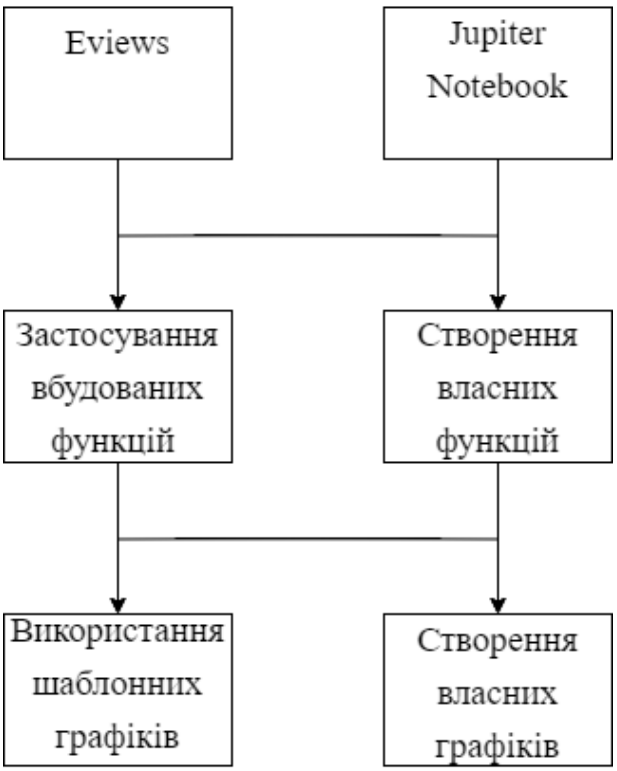


Рисунок 4.1 – Морфологічна карта

Морфологічна карта відображає множину всіх можливих варіантів основних функцій. Позитивно-негативна матриця показана в таблиці 4.1.

На основі аналізу позитивно-негативної матриці робимо висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто відкинути, тому, що вони не відповідають поставленим перед програмним продуктом задачам. Ці варіанти відзначені у морфологічній карті.

Таблиця 4.1 – Позитивно-негативна матриця

Основні функції	Варіанти реалізації	Переваги	Недоліки
F1	А	Має зручний графічний інтерфейс. Простий у використанні для початківців.	Менш гнучкий, ніж інші програмні пакети.
	Б	Дозволяє створювати власні коди та візуалізації.	Потребує більше ресурсів
F2	А	Знижує необхідність писати власний код. Забезпечує надійність та точність результатів.	Може бути менш гнучким, ніж створення власних функцій.
	Б	Повна гнучкість та контроль над процесом. Можливість створити функції, які відповідають вашим конкретним потребам.	Може бути більш часозатратним та трудомістким.
F3	А	Швидкий та простий спосіб створити графіки.	Може бути менш гнучким, ніж створення власних графіків. Не завжди доступні всі необхідні типи графіків.
	Б	Повна гнучкість та контроль над зовнішнім виглядом графіка.	Може бути більш часозатратним та трудомістким.

Функція $F1$: Перевагу даємо гнучкості та кількості вбудованих бібліотек. Для зручного коду під конкретні задачі варіант А має бути відкинутий.

Функція $F2$: Програма допускає обрання обох варіантів. Можливо використати варіанти А чи Б.

Функція $F3$: Реалізація першого варіанту є сприйнятливою для програми. Це варіант А. Таким чином, будемо розглядати такий варіанти реалізації ПП:

$$F1б – F2а – F3а$$
$$F1б – F2б – F3а$$

4.3 Обґрунтування системи параметрів програмного продукту

На основі даних, які були розглянуті раніше, ми визначаємо основні параметри вибору, які будуть використовуватися для розрахунку коефіцієнта технічного рівня програмного продукту. Для характеристики програмного продукту ми використовуємо наступні параметри:

X1 – Швидкодія мови програмування;

X2 – Обсяг пам'яті для обчислень та збереження даних;

X3 – Час навчання даних;

X4 – Потенційний обсяг програмного коду.

Гірші, середні і кращі значення цих параметрів вибираються на основі вимог замовника та умов, що характеризують експлуатацію програмного продукту, як показано у таблиці 4.2.

Таблиця 4.2 – Гірші, середні і кращі значення параметрів

Назва параметра	Умовні позначення	Одиниці вимірювання	Значення параметра		
			Гірші	Середні	Кращі
Швидкодія мови програмування	X1	оп/мс	100	1000	10000
Обсяг пам'яті для обчислень та збереження даних	X2	Мб	100	50	10
Час навчання даних	X3	мс	100000	50000	25000
Потенційний обсяг програмного коду	X4	кількість рядків коду	1200	500	300

За даними таблиці 4.2 будуються графічні характеристики параметрів – рис. 4.2 – рис. 4.5.

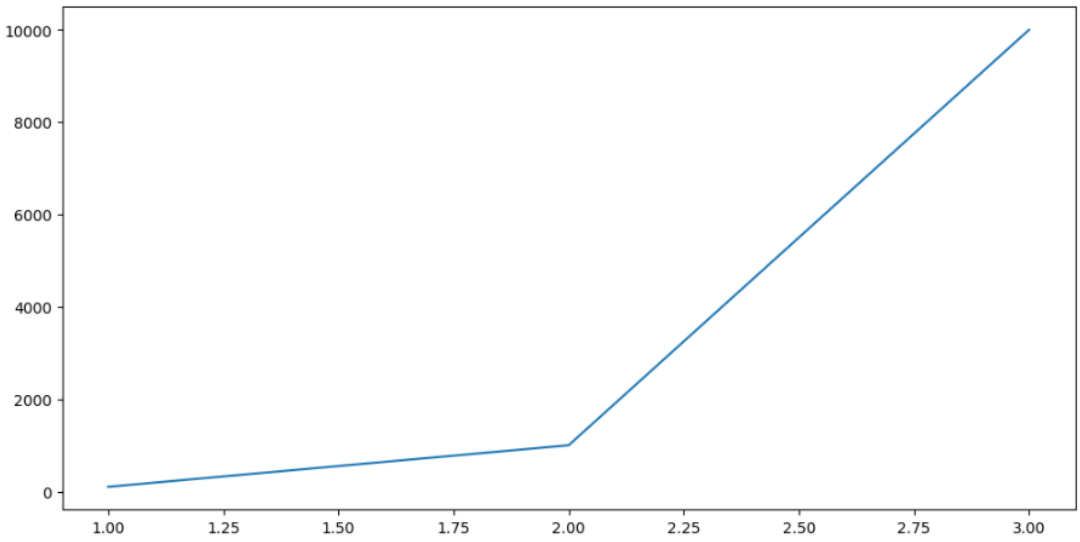


Рисунок 4.2 – Швидкодія мови програмування (оп/мс), X1

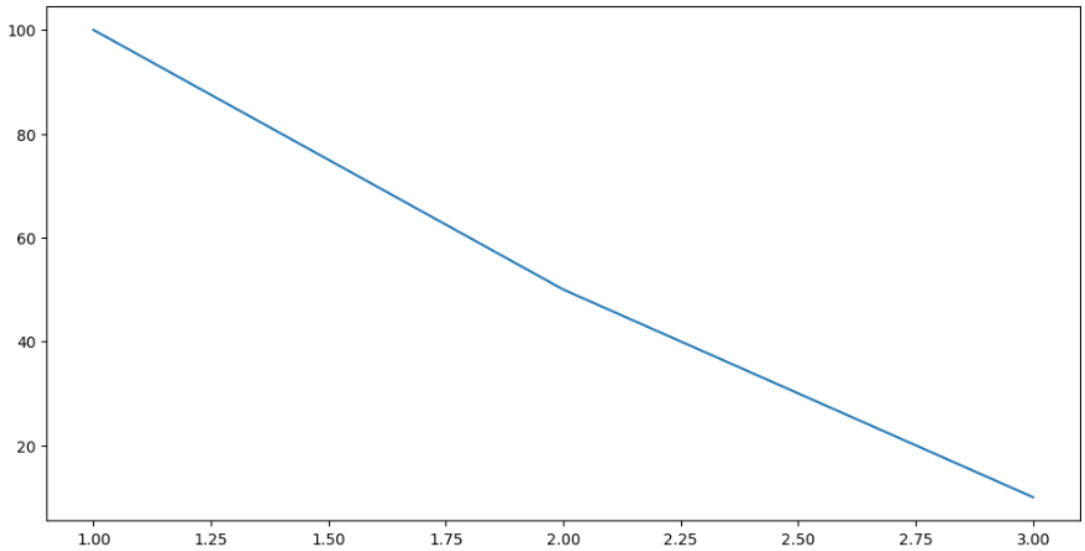


Рисунок 4.3 – Об’єм обчислювальних ресурсів (мб), X2

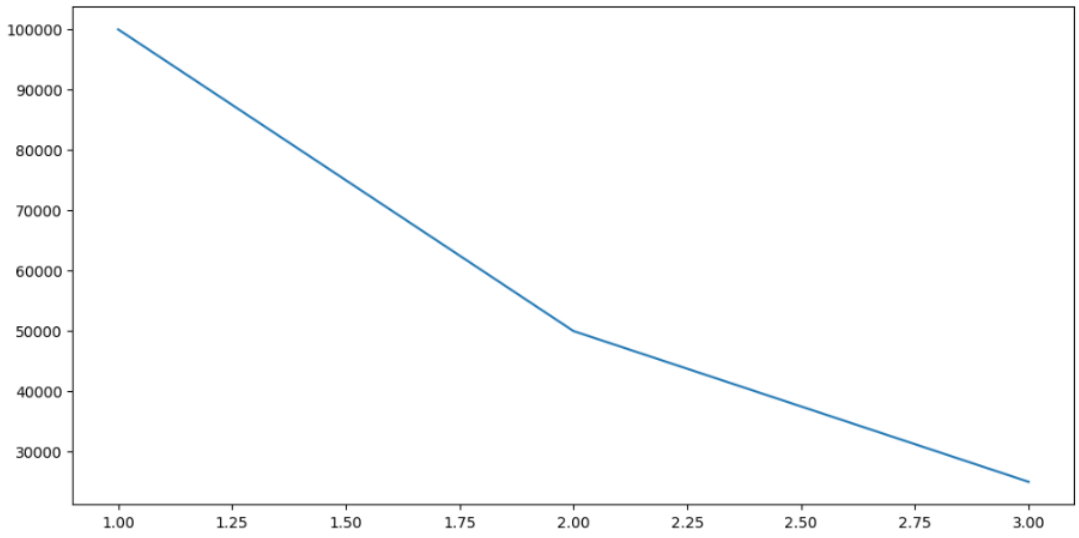


Рисунок 4.4 – Час навчання моделі (хв), X3

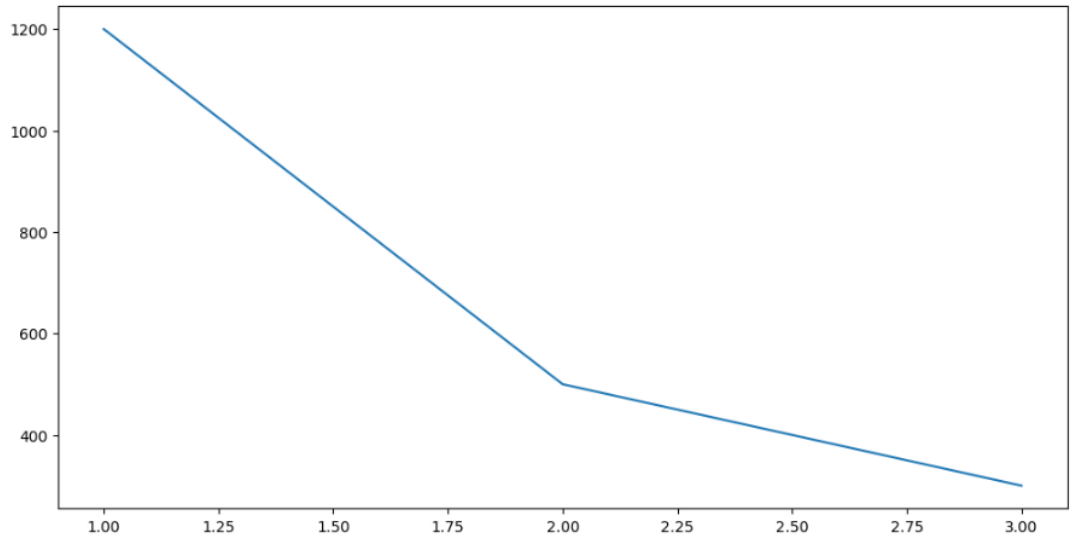


Рисунок 4.5 – Потенційний об’єм програмного коду, X4

4.4 Аналіз експертного оцінювання параметрів

Після детального обговорення й аналізу кожний експерт оцінює ступінь важливості кожного параметру для конкретно поставленої цілі – розробка програмного продукту, який дає найбільш точні результати при знаходженні параметрів моделей адаптивного прогнозування і обчислення прогнозних значень.

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із 7 людей. Визначення коефіцієнтів значимості передбачає:

− визначення рівня значимості параметра шляхом присвоєння різних рангів;

− перевірку придатності експертних оцінок для подальшого використання;

− визначення оцінки попарного пріоритету параметрів;

− обробку результатів та визначення коефіцієнту значимості.

Таблиця 4.3 – Результати експертного ранжування

Назва параметра	Ранг параметра за оцінкою експерта							Сума рангів,R_i	Відхи-лення, Δ_i	Δ_i^2
	1	2	3	4	5	6	7			
X1	1	2	2	3	2	2	2	13	-4.5	20.25
X2	3	3	4	2	3	3	1	19	1.5	2.25
X3	2	1	1	1	1	1	3	10	-7.5	56.25
X4	4	4	3	4	4	4	4	27	9.5	90.25
Разом	10	10	10	10	10	10	10	70	0	

Для перевірки степені достовірності експертних оцінок, визначимо наступні параметри:

а) сума рангів кожного з параметрів і загальна сума рангів:

$$R_i = \sum_{j=1}^{N} r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 70$$

де N – число експертів,

n – кількість параметрів.

b) Середня сума рангів:

$$T = \frac{1}{n} R_{ij} = 17.5$$

c) Відхилення суми рангів кожного параметра від середньої суми рангів:

$$\Delta_i = R_i - T$$

Сума відхилень по всім параметрам повинна дорівнювати 0

d) Загальна сума квадратів відхилення:

$$S = \sum_{i=1}^{N} \Delta_i^2 = 169$$

Порахуємо коефіцієнт узгодженості:

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 * 169}{7^2(4^3 - 4)} = 0.69 > W_k = 0.67$$

Ранжування можна вважати достовірним, тому що знайдений коефіцієнт узгодженості перевищує нормативний, котрий дорівнює 0.67.

Скориставшись результатами ранжирування, проведемо попарне порівняння всіх параметрів і результати занесемо у таблицю 4.4

Таблиця 4.4 – Попарне порівняння всіх параметрів

Параметри	Експерти							Підсумкова оцінка	Числове значення коефіцієнтів переваги (x_{ij})
	1	2	3	4	5	6	7		
x_1 та x_2	>	>	>	<	>	>	<	>	1,5
x_1 та x_3	>	<	<	<	<	<	>	<	0,5
x_1 та x_4	>	>	>	>	>	>	>	>	1,5
x_2 та x_3	<	<	<	<	<	<	>	<	0,5
x_2 та x_4	>	>	<	>	>	>	>	>	1,5
x_3 та x_4	>	>	>	>	>	>	>	>	1,5

Числове значення, що визначає ступінь переваги і–го параметра над j–тим, аіj визначається по формулі

$$a_{ij} = \begin{cases} 1.5, X_i > X_j \\ 1.0, X_i = X_j \\ 0.5, X_i < X_j \end{cases}$$

З отриманих числових оцінок переваги складемо матрицю А=‖ aij ‖.

Для кожного параметра зробимо розрахунок вагомості Кві за наступними формулами:

$$K_{ві} = \frac{b_i}{\sum_{i=1}^{n} b_i}$$

$$b_i = \sum_{i=1}^{n} a_{ij}$$

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються за наступними формулами

$$K_{ві} = \frac{b_i^{'}}{\sum_{i=1}^{n} b_i^{'}}$$

$$b_i^{'} = \sum_{i=1}^{n} a_{ij} b_j$$

Як видно з таблиці 4.5, різниця значень коефіцієнтів вагомості не перевищує 5%, тому більшої кількості ітерацій не потрібно.

Таблиця 4.5 – Різниця значень коефіцієнтів

X_i	Параметри Xj,				Перша ітерація		Друга ітерація	
	X_1	X_2	X_3	X_4	$в_i$	$К_{ві}$	$в'_i$	$К'_{ві}$
X_1	1,0	1,5	0,5	1,5	4,5	0,28125	16,25	0,2754
X_2	0,5	1,0	0,5	1,5	3,5	0,21875	12,25	0,2076
X_3	1,5	1,5	1,0	1,5	5,5	0,34375	21,25	0,35
X_4	0,5	0,5	0,5	1,0	2,5	0,15625	9,25	0,156
Всього					16,0	1,0	59	1,0

4.5 Аналіз рівня якості варіантів реалізації функцій

Визначаємо рівень якості кожного варіанту виконання основних функцій окремо. Абсолютні значення параметрів X2 (об’єм пам’яті), X3 (час навчання) та X4 (потенційний об’єм програмного коду) відповідають технічним вимогам умов функціонування даного ПП. Абсолютне значення параметра X1

(швидкість роботи мови програмування) обрано не найгіршим. Коефіцієнт технічного рівня для кожного варіанта реалізації ПП розраховується так (таблиця 4.6):

$$K_K(j) = \sum_{i=1}^{n} K_{ві} B_{ij},$$

де n – кількість параметрів;

*К*в*і* – коефіцієнт вагомості і–го параметра;

Ві – оцінка і–го параметра в балах.

Таблиця 4.6

Основна функція,	Варіант реалізації функції	Параметри, які беруть участь у реалізації функцій	Абсолютне значення параметра	Оцінка параметра в балах, Ві	Коефіцієнт вагомості,$К_{ві}$	Показник технічного рівня, $К_{тр}$ [F_i]
F1	б	X_1	900	4	0,28	1,25
F2	а	X_2	35	8	0,21	1,68
	б	X3	25000	10	0,35	3,5
Fi	а	X4	450	5	0,16	0,8

Показник рівня якості *k*-го варіанта реалізації основних функцій виробу

$$K_к = K_{тр}\,[F_{1k}] + K_{тр}[F_{2k}] + \ldots + K_{тр}\,[F_{zk}]\,,$$

де $K_{тр}\,[F_{1k}]$ - показник технічного рівня першої функції *К*-го варіанта реалізації основних функцій виробу.

Визначаємо рівень якості кожного з варіантів:

$$K_{K1} = 1,25 + 1,68 + 0,8 = 3,73$$
$$K_{K2} = 1,25 + 3,5 + 0,8 = 5,55$$

Як видно з розрахунків, кращим є варіант 2, для якого коефіцієнт технічного рівня має найбільше значення.

4.6 Економічний аналіз варіантів розробки ПП

Для визначення вартості розробки ПП спочатку проведемо розрахунок трудомісткості. Всі варіанти включають в себе два окремих завдання:

1. Розробка проекту програмного продукту;

2. Розробка програмної оболонки.

Завдання 1 за ступенем новизни відноситься до групи А, завдання 2 – до групи Б. За складністю алгоритми, які використовуються в завданні 1 належать до групи 1; а в завданні 2 – до групи 3.

Для реалізації завдання 1 використовується довідкова інформація, а завдання 2 використовує інформацію у вигляді даних. Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань. Загальна трудомісткість обчислюється як:

$$T_o = T_p \bullet K_n \bullet K_{ck} \bullet K_M \bullet K_{ct} \bullet K_{ct.M}$$

де ТР – трудомісткість розробки ПП;

КП – поправочний коефіцієнт;

КСК – коефіцієнт на складність вхідної інформації;

КМ – коефіцієнт рівня мови програмування;

КСТ – коефіцієнт використання стандартних модулів і прикладних програм;

КСТ.М – коефіцієнт стандартного математичного забезпечення

Для першого завдання, виходячи із норм часу для завдань розрахункового характеру степеню новизни А та групи складності алгоритму 1, трудомісткість дорівнює: ТР = 92 людино-днів. Поправочний коефіцієнт, який враховує вид нормативно-довідкової інформації для першого завдання: КП = 1,45. Поправочний коефіцієнт, який враховує складність контролю вхідної та вихідної інформації для всіх завдань рівний 1: КСК = 1. Оскільки при розробці першого завдання використовуються стандартні модулі, врахуємо це за допомогою коефіцієнта КСТ = 0.8. Тоді загальна трудомісткість програмування першого завдання дорівнює:

$$T_1 = 92 * 1.45 * 0.8 = 106,7$$

Проведемо аналогічні розрахунки для подальших дій. Для другого завдання (використовується алгоритм третьої групи складності, степінь новизни Б), тобто ТР =69 людино-днів, КП =0,72, КСК = 1, КСТ =0.9:

$$T_2 = 69 * 0.72 * 0.9 = 44,7$$

Складаємо трудомісткість відповідних завдань щоб отримати їх трудомісткість:

$$T_I = (106,7 + 44,7 + 44,7) * 8 = 1568,8$$

$$T_{II} = (106,7 + 106,7 + 44,7) * 8 = 2064,8$$

В розробці беруть участь два програмісти з окладом 30000 грн., один аналітик в області даних з окладом 35000. Визначимо середню зарплату за годину за формулою:

$$C_ч = \frac{M}{T_m t} грн.,$$

де М – місячний оклад працівників;

Tm – кількість робочих днів тиждень;

t – кількість робочих годин в день.

$$C_ч = \frac{30000 + 30000 + 35000}{3 * 21 * 8} = 188,49 грн.$$

Тоді, розрахуємо заробітну плату за формулою:

$$C_{ЗП} = C_ч T_i K_Д ,$$

де СЧ– величина погодинної оплати праці програміста;

Тi – трудомісткість відповідного завдання;

КД – норматив, який враховує додаткову заробітну плату.

Зарплата розробників за варіантами становить:

$$I.C_{ЗП} = 188.49 * 1568.8 * 1.2 = 354843.7$$

$$II.C_{ЗП} = 188.49 * 2064.8 * 1.2 = 467032.98$$

Відрахування на єдиний соціальний внесок становить 22%:

$$I.C_{ВІД} = C_{ЗП} * 0,22 = 354843.7 * 0,22 = 78065,6$$

$$II.C_{ВІД} = C_{ЗП} * 0,22 = 467032.98 * 0,22 = 102747,26$$

Тепер визначимо витрати на оплату однієї машино-години. ($С_М$)

Так як одна ЕОМ обслуговує одного програміста з окладом 30000 грн., з коефіцієнтом зайнятості 0,2 то для однієї машини отримаємо:

$$С_Г = 12 \cdot М \cdot К_З = 12 \cdot 30000 \cdot 0,2 = 72000 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$С_{ЗП} = С_Г \cdot (1 + К_З) = 72000 \cdot (1 + 0.2) = 86400 \text{ грн.}$$

Відрахування на соціальний внесок:

$$С_{ВІД} = С_{ЗП} \cdot 0.22 = 48960 \cdot 0,22 = 19008 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 10000 грн.

$$С_А = К_{ТМ} \cdot К_А \cdot Ц_{ПР} = 1.4 \cdot 0.25 \cdot 10000 = 3500 \text{ грн.,}$$

де $К_{ТМ}$– коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача;

$К_А$– річна норма амортизації;

$Ц_{ПР}$– договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$$С_Р = К_{ТМ} \cdot Ц_{ПР} \cdot К_Р = 1.4 \cdot 10000 \cdot 0.08 = 1120 \text{ грн.,}$$

де $К_Р$– відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$Т_{ЕФ} = (Д_К – Д_В – Д_С – Д_Р) \cdot t_З \cdot К_В = (365 – 104 – 12 – 16) \cdot 8 \cdot 0.35 =$$

$$= 627,2 \text{ години,}$$

де $Д_К$ – календарна кількість днів у році;

$Д_В$, $Д_С$ – відповідно кількість вихідних та святкових днів;

$Д_Р$ – кількість днів планових ремонтів устаткування;

t –кількість робочих годин в день;

$К_В$– коефіцієнт використання приладу у часі протягом зміни.

Витрати на оплату електроенергії розраховуємо за формулою:

$$С_{ЕЛ} = Т_{ЕФ} \cdot N_С \cdot К_З \cdot Ц_{ЕН} = 627,2 \cdot 0,2 \cdot 0,3 \cdot 9,43 = 354,87 \text{ грн.},$$

де $N_С$ – середньо-споживча потужність приладу;

$К_З$– коефіцієнтом зайнятості приладу;

$Ц_{ЕН}$ – тариф за 1 КВт-годин електроенергії.

Накладні витрати розраховуємо за формулою:

$$С_Н = Ц_{ПР} \cdot 0.67 = 10000 \cdot 0,67 = 6700 \text{ грн.}$$

Тоді, річні експлуатаційні витрати будуть:

$$С_{ЕКС} = С_{ЗП} + С_{ВІД} + С_А + С_Р + С_{ЕЛ} + С_Н$$

$С_{ЕКС} = 86400 + 19008 + 3500 + 1120 + 354,87 + 6700 = 117082,87$ грн.

Собівартість однієї машино-години ЕОМ дорівнюватиме:

$$С_{М\text{-}Г} = С_{ЕКС} / Т_{ЕФ} = 117082,87 / 627,2 = 186,68 \text{ грн/год.}$$

Оскільки в даному випадку всі роботи, які пов‘язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$С_М = С_{М-Г} \cdot T, \#(4.18)$$

I. $C_М = 186,68 \cdot 1568,8 = 292863,58$ грн.

II. $C_М = 186,68 \cdot 2064,8 = 385456,86$ грн.

Накладні витрати складають 67% від заробітної плати:

$$C_Н = C_{ЗП} \cdot 0,67, \#(4.19)$$

I. $C_Н = 354843.7 \cdot 0,67 = 237745,28$ грн.

II. $C_Н = 467032.98 \cdot 0,67 = 312912,1$ грн.

Отже, вартість розробки ПП за варіантами становить:

$$C_{ПП} = C_{ЗП} + C_{ВІД} + C_М + C_Н$$

I. $C_{ПП} = 354843.7 + 78065,6 + 292863,58 + 237745,28 = 963518,16$ грн.

II. $C_{ПП} = 467032.98 + 102747,26 + 385456,86 + 312912,1 = 1268149,2$ грн.

4.7 Вибір кращого варіанту ПП техніко-економічного рівня

Розрахуємо коефіцієнт техніко-економічного рівня за формулою:

$$К_{ТЕР j} = К_{К j} / С_{Ф j},$$

$$К_{ТЕР1} = 3,73 / 963518,16 = 3,871 \cdot 10^{-6},$$

$$К_{ТЕР2} = 5,55 / 1268149,2 = 4,376 \cdot 10^{-6}.$$

Як бачимо, найбільш ефективним є другий варіант реалізації програми з коефіцієнтом техніко-економічного рівня $К_{ТЕР1} = 4,376 \cdot 10^{-6}$.

Після виконання функціонально-вартісного аналізу програмного

комплексу що розроблюється, можна зробити висновок, що з альтернатив, що залишились після першого відбору двох варіантів виконання програмного комплексу оптимальним є другий варіант реалізації програмного продукту. У нього виявився найкращий показник техніко-економічного рівня якості $К_{ТЕР} = 4{,}376 \cdot 10^{-6}$.

Цей варіант реалізації програмного продукту має такі параметри:

- Вибір середовища розробки – Jupiter notebook;
- Якісний аналіз даних за допомогою власних створених функцій;
- Використання шаблонних варіантів для побудови графіків.

Даний варіант виконання програмного комплексу дає користувачу зручний інтерфейс, швидку реалізацію програми та доступний функціонал для роботи.

Висновки до розділу 4

В даному розділі було проведено повний функціонально-вартісний аналіз програмного продукту. Також було знайдено оцінку основних функцій програмного продукту. Також в даному дослідженні показано різні варіанти реалізації для забезпечення найбільш коректної та оптимальної стратегії вибору, що має вплив на економічні фактори та сумісність з майбутнім програмним продуктом. В результаті виконання функціонально-вартісного аналізу програмного комплексу що розроблюється, було визначено та проведено оцінку основних функцій програмного продукту, а також знайдено параметри, які його характеризують. На основі аналізу вибрано варіант реалізації програмного продукту.

ВИСНОВКИ

У результаті виконання повного дослідницького циклу, охопленого в цій науковій роботі, було досягнуто поставленої мети – створено, протестовано та проаналізовано інформаційну систему прогнозування фінансових часових рядів на прикладі динаміки ВВП різних країн світу в логарифмічному представленні. Побудована система ґрунтується на інтеграції математичної статистики, класичних моделей часових рядів та сучасних алгоритмів машинного навчання в єдину аналітичну архітектуру. Вона реалізована за допомогою мови програмування Python з використанням потужного інструментарію – бібліотек pandas, numpy, matplotlib, statsmodels, scikit-learn тощо.

Одним із важливих етапів дослідження була ретельна підготовка даних. Було використано відкритий датасет із платформи Kaggle, що містив історичні значення ВВП у поточних доларах США для 266 країн світу за період з 1960 по 2022 рік. Спочатку дані було очищено від записів, що не стосуються окремих держав (наприклад, «World», «Arab World», «High Income» тощо), а також реалізовано перетворення з wide-формату в long-формат, придатний для побудови та аналізу часових рядів. Подальша логарифмічна трансформація значень ВВП (log1p) дозволила зменшити вплив масштабних диспропорцій між країнами. Важливою частиною була імпутація відсутніх значень – реалізована через заповнення пропусків середнім по кожній країні (із заздалегідь заданим порогом допустимих пропусків). Це дозволило уникнути значних перекручень трендів та зберегти цілісність рядів для аналізу.

На основі підготовленого датасету було здійснено побудову моделей прогнозування логарифмованого ВВП для шести провідних економік світу: США, Китаю, Японії, Німеччини, Франції та Великої Британії. Було обрано 8 моделей: сім із машинного навчання (Linear Regression, Ridge, Lasso, Decision Tree, Random Forest, SVR, Gradient Boosting) та одну статистичну модель ARIMA. Для кожної моделі було проведено навчання на історичних даних, розділених на тренувальну (до 2017 року) та тестову частини (2018–2022), що

відповідає класичному підходу rolling window оцінки прогнозу на майбутнє.

Результати оцінювання моделей продемонстрували переконливу перевагу ARIMA, яка забезпечила найнижчі значення середньої абсолютної помилки (MAE) та середньої абсолютної відносної похибки (MAPE) у більшості випадків. Зокрема, у США ARIMA досягла MAPE = 0.14%, у Китаї – 0.12%, у Японії – 0.19%, у Франції – 0.18%, у Великій Британії – 0.29%. Навіть у Німеччині, де точність інших моделей також була високою, ARIMA продемонструвала MAPE = 0.29%, лише трохи поступаючись GBM (0.31%). Таким чином, ARIMA довела свою ефективність для стабільних трендових рядів з одновимірною структурою.

Серед моделей машинного навчання найвищу точність показали Gradient Boosting та Random Forest. Gradient Boosting досяг MAPE нижче 0.5% у США, Японії, Німеччині та Франції, продемонструвавши гнучкість у адаптації до слабких флуктуацій. Random Forest виявився надійною альтернативою, забезпечуючи стабільно хороші результати у всіх країнах, особливо в Японії (MAPE = 0.17%) та Великій Британії (0.26%). Інші моделі, зокрема Linear, Ridge, Lasso та SVR, показали гірші результати, особливо в країнах зі складною економічною динамікою – Японії та Франції, де похибка перевищувала 2–4% у лінійних моделей. Це свідчить про обмеження цих методів у задачах, де присутні нелінійні тренди або структурні зміни.

Візуалізація прогнозів була реалізована у вигляді графіків, на яких порівнювалися реальні та прогнозовані значення для кожної країни. Завдяки виключенню моделі MLP, яка призводила до викривлення масштабу, всі графіки стали інформативними та наочними. На рисунках чітко видно, як ARIMA та GBM відтворюють динаміку ряду із високою точністю, у той час як лінійні моделі відхиляються від фактичного тренду. Такі графіки є не лише доказовою частиною дослідження, а й інструментом комунікації результатів у реальній інформаційній системі.

Дослідження не лише продемонструвало ефективність конкретних алгоритмів прогнозування, а й підтвердило важливість системного підходу до

побудови інформаційної системи як складного багаторівневого інструменту. У межах реалізації роботи було побудовано архітектуру, що охоплює всі ключові компоненти сучасної аналітики часових рядів: підготовку даних, побудову та навчання моделей, їхню оцінку, візуалізацію результатів і потенційну інтеграцію в користувацький інтерфейс. Такий цілісний підхід забезпечує надійність системи в умовах різної якості вхідних даних, що особливо актуально при роботі з реальними економічними індикаторами, які часто мають пропуски, аномалії та складну динаміку.

Інформаційна система, побудована в межах цього дослідження, володіє кількома ключовими властивостями. По-перше, вона є адаптивною: користувач має можливість змінювати параметри моделей, вибирати країни, періоди, методи імпутації або глибину тренувальних вікон. По-друге, система є відкритою для масштабування – до неї легко можна додати інші фінансові показники, наприклад, інфляцію, рівень безробіття або обсяг експорту. По-третє, вона інтерпретована: кожен етап моделювання можна проконтролювати, кожен прогноз супроводжується числовими метриками і графіком. Це дозволяє використовувати систему як у наукових дослідженнях, так і в реальній практиці – зокрема, в економічному плануванні, фінансовому консалтингу або державній статистиці.

Окремо слід наголосити на цінності реалізованої візуалізації. Графіки, які наведено в розділі 3.5, дозволяють здійснювати швидке та наочне порівняння між фактичними значеннями й прогнозами моделей, а також між самими моделями. Це є надзвичайно важливим з точки зору прийняття рішень, оскільки дає змогу фахівцю з економіки чи фінансів одразу оцінити, яка модель найбільш точно відображає ситуацію в обраній країні. Більше того, візуальні інструменти можуть бути безпосередньо інтегровані в веб-інтерфейс системи або вбудовані в програмні модулі для автоматизованої аналітики.

Отже можна стверджувати, що створена система відповідає сучасним вимогам до інтелектуальної обробки фінансової інформації. Вона поєднує в собі точність, гнучкість і аналітичну глибину. Проведена робота відкриває

перспективи подальшого розвитку: можлива інтеграція багатовимірних моделей (VAR, Prophet, LSTM), автоматизація підбору гіперпараметрів, поєднання з геоекономічними показниками або новинними потоками. Практична значущість системи особливо актуальна в умовах постпандемічного та посткризового економічного відновлення, де швидкий доступ до точного прогнозу може мати вирішальне значення для розробки державної політики, формування інвестиційних стратегій та оцінки макроекономічних ризиків.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Мнишенко В. С. Теорія випадкових процесів та часових рядів : навч. посібник. Київ : КНЕУ. 2019. 276 с.

2. Box G. E. P. Jenkins G. M. Reinsel G. C. Ljung G. M. Time Series Analysis : Forecasting and Control. 5th ed. Wiley. 2015.

3. Engle R. Autoregressive Conditional Heteroskedasticity with Estimates of the Variance of United Kingdom Inflation Econometrica. 1982. Vol. 50. No. 4. P. 987–1007.

4. Hochreiter S. Schmidhuber. J. Long Short-Term Memory Neural Computation. 1997. Vol. 9 No. 8. P. 1735–1780.

5. Lütkepohl H. New Introduction to Multiple Time Series Analysis – Springer. 2005.

6. Lim B. Loeff N. Pfister T. Temporal Fusion Transformers for Interpretable Multi-Horizon Time Series Forecasting International Journal of Forecasting. 2021. Vol. 37. No. 4. P. 1748–1764.

7. Zhang G. Eddy Patuwo B. Hu. M. Y. Forecasting with Artificial Neural Networks: The State of the Art International Journal of Forecasting. 1998. Vol. 14. No. 1. P. 35–62.

8. Taylor S. J. Letham B. Forecasting at Scale The American Statistician. 2018. Vol. 72. No. 1. P. 37–45.

9. Tavenard R. et al. Darts: User-Friendly Modern Machine Learning for Time Series Journal of Machine Learning Research. 2020. Vol. 23. No. 124. P. 1–6.

10. Dietterich T. G. Ensemble Methods in Machine Learning International Workshop on Multiple Classifier Systems. 2022. P. 1–15.

11. Шапочкіна О. В. Інформаційні системи у фінансово-економічній діяльності : навч. посібник. Київ : Центр учбової літератури. 2020. 308 с.

12. Brockwell P. J. Davis R. A. Introduction to Time Series and Forecasting. – Springer. 2016.

13. IBM SPSS Forecasting Documentation [Електрон. ресурс]. – IBM

Knowledge Center. – URL: https://www.ibm.com/docs. – Дата доступу: 01.06.2025.

14. Salinas D. Flunkert. V. Gasthaus. J. Januschowski. T. DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks International Journal of Forecasting. – 2020. Vol. 36. No. 3. P. 1181–1191.

15. Bloomberg Terminal Overview [Електрон. ресурс] Bloomberg Professional Services. – URL: https://www.bloomberg.com/professional/solution/bloomberg-terminal/ – Дата доступу: 01.06.2025.

16. Hyndman R. J. Athanasopoulos. G. Forecasting: Principles and Practice. – 3rd ed. OTexts. 2018.

17. Zhang Q. Yang L. Zhang D. A Survey on Deep Learning in Time-Series Forecasting IEEE Transactions on Neural Networks and Learning Systems. 2022. Vol. 33. No. 10. P. 5321–5343.

18. Makridakis S. Spiliotis E. Assimakopoulos V. Statistical and Machine Learning Forecasting Methods: Concerns and Ways Forward International Journal of Forecasting. 2020. Vol. 36. No. 1. P. 186–213.

19. Геєць В. М. Гриньова В. М. Олексенко О. С. Прогнозування економічного розвитку : методи та моделі. Харків : Видавництво «ІНЖЕК». 2021. 352 с.

20. Chollet F. Deep Learning with Python. 2nd ed. Manning Publications. 2021.

21. Rudin C. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead Nature Machine Intelligence. 2019. Vol. 1. No. 5. P. 206–215.

22. World Bank Data. World Development Indicators [Електрон. ресурс]. – URL: https://data.worldbank.org – Дата доступу: 01.06.2025.

23. Hamilton J. D. Time Series Analysis. – Princeton : Princeton University Press. 1994.

24. OECD. OECD Economic Outlook. – Paris: OECD Publishing. 2025.

25. Chatfield C. The Analysis of Time Series: An Introduction. – 6th ed. Boca

Raton : Chapman & Hall/CRC. 2003.

26. Wei W. W. S. Time Series Analysis: Univariate and Multivariate Methods. – 2nd ed. Pearson. 2014.

27. Shumway R. H. Stoffer. D. S. Time Series Analysis and Its Applications: With R Examples. – 4th ed. Springer. 2017.

28. Козленко А. П. Тарасюк. А. П. Основи аналізу часових рядів : навч. посібник. Київ : КНЕУ. 2020. 212 с.

29. Durbin J. Koopman S. J. Time Series Analysis by State Space Methods. – 2nd ed. Oxford : Oxford University Press. 2012.

30. Priestley M. B. Spectral Analysis and Time Series. – Academic Press. 1981.

31. Oord A. van den. Li. Y. Vinyals. O. WaveNet: A Generative Model for Raw Audio arXiv:1609.03499 [preprint]. 2016.

32. Кузьменко О. В. Савченко Т. І. Економетрія : підручник. – Київ : Центр учбової літератури. 2021. 496 с.

33. Hyndman R. J. Athanasopoulos. G. Forecasting: Principles and Practice. – 2nd ed. OTexts. 2018.

34. Box G. E. P. Jenkins G. M. Reinse. G. C. Ljung G. M. Time Series Analysis : Forecasting and Control. – 4th ed. Wiley. 2008.

35. Tsay R. S. Multivariate Time Series Analysis: With R and Financial Applications. – 2nd ed. Wiley. 2014.

36. Ковальов В. В. Економетрія. – Київ : Центр учбової літературі. 2020. 512 с.

37. Pfaff B. Analysis of Integrated and Cointegrated Time Series with R. – 2nd ed. Springer. 2008.

38. Akaike H. A New Look at the Statistical Model Identification IEEE Transactions on Automatic Control. 1974. Vol. 19. No. 6. P. 716–723.

39. Ljung G. M. Box G. E. P. On a Measure of Lack of Fit in Time Series Models Biometrika. 1978. Vol. 65. No. 2. P. 297–303.

40. Shapiro S. S. Wilk. M. B. An Analysis of Variance Test for Normality

(Complete Samples) Biometrika. 1965. Vol. 52. No. 3/4. P. 591–611.

41. Priestley M. B. Nonlinear and Nonstationary Time Series Analysis. – Academic Press. 1999.

42. Bollerslev T. Generalized Autoregressive Conditional Heteroskedasticity Journal of Econometrics. 1986. Vol. 31. No. 3. P. 307–327.

43. Sims C. A. Macroeconomics and Reality Econometrica. – 1980. Vol. 48. No. 1. . 1–48.

44. Bandara K. LSTM-MSNet : Leveraging Forecasts on Sets of Time Series Using a Multi-Scale LSTM-Based Architecture Neurocomputing. 2020. Vol. 396. P. 197–213.

45 Stoc J. H. Watson M. W. Introduction to Econometrics. – 3rd ed. Pearson. 2015.

46. Friedman J. H. Greedy Function Approximation: A Gradient Boosting Machine Annals of Statistics. 2001. Vol. 29. No. 5. P. 1189–1232.

47. Breiman L. Random Forests Machine Learning. 2001. Vol. 45. No. 1. P. 5–32.

48. Tibshirani R. Regression Shrinkage and Selection via the Lasso Journal of the Royal Statistical Society: Series B. 1996. Vol. 58. No. 1. – P. 267–288.

49. Zou H. Hastie T. Regularization and Variable Selection via the Elastic Net Journal of the Royal Statistical Society: Series B. 2005. Vol. 67. No. 2. – P. 301–320.

50. Prokhorenkova L. Gusev G. Vorobev A. Dorogush A. V. Gulin. A. CatBoost : Unbiased Boosting with Categorical Features In: Proceedings of NeurIPS. 2018.

51. Feurer M. Klein A. Eggensperger K. Springenberg J. Blum M. Hutter F. Efficient and Robust Automated Machine Learning In: Proceedings of NeurIPS. – 2015.

52. Molnar C. Interpretable Machine Learning – 2nd ed. [Електрон. ресурс]. URL: https://christophm.github.io/interpretable-ml-book/ – Дата доступу: 01.06.2025.

53. Lundberg S. M. Erion. G. G. Lee. S.-I. From Local Explanations to Global

Understanding with Explainable AI for Trees Nature Machine Intelligence. – 2020. Vol. 2. No. 1. P. 56–67.

54. Sezer O. B. Gudelek M. U. Ozbayoglu A. M. Financial Time Series Forecasting with Deep Learning: A Systematic Literature Review: 2005–2019 Applied Soft Computing. 2020. Vol. 90. – Article ID: 106181.

55. Cho K. Van Merriërboer. B. Gulcehre C. Bahdanau D. Bougares F. Schwenk. H. Bengio. Y. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation EMNLP. 2014. P. 1724–1734.

56. Graves A. Speech Recognition with Deep Recurrent Neural Networks ICASSP. 2013. P. 6645–6649.

57. Bengio Y. Simard P. Frasconi P. Learning Long-Term Dependencies with Gradient Descent is Difficult IEEE Transactions on Neural Networks. 1994. Vol. 5. No. 2. P. 157–166.

58. Srivastava N. Hinton G. Krizhevsky A. Sutskever I. Salakhutdinov R. Dropout : A Simple Way to Prevent Neural Networks from Overfitting Journal of Machine Learning Research. 2014. Vol. 15. P. 1929–1958.

59. Bahdanau D. Cho K. Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate ICLR. 2015.

Lipton Z. C. Berkowitz J. Elkan C. A Critical Review of Recurrent Neural Networks for Sequence Learning arXiv:1506.00019 [preprint]. 2015.

60. Brownlee J. Deep Learning for Time Series Forecasting. – Machine Learning Mastery. 2017. [Електрон. ресурс]. URL: https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/ – Дата доступу: 01.06.2025.

61. Vaswani A. Shazeer N. Parmar N. Uszkoreit J. Jones L. Gomez A. N. Kaiser. Ł. Polosukhin. I. Attention is All You Need NeurIPS. 2017. P. 5998–6008.

63. Zhou H. Zhang S. Peng J. Zhang S. Li J. Zhang X. Li. H. Lu. H. Informer : Beyond Efficient Transformer for Long Sequence Time-Series Forecasting AAAI. 2021. P. 11106–11115.

64. Wu H. Xu J. Wang J. Long. M. Autoformer : Decomposition Transformers

with Auto-Correlation for Long-Term Series Forecasting NeurIPS. 2021.

65. Zhou T. Ma Z. Zhang. Q. Gao. Z. Wu. C. Yu. A. Wei. S. Tao. D. Chen. Z. FEDformer : Frequency Enhanced Decomposed Transformer for Long-Term Series Forecasting ICML. 2021.

66. Tsay. R. S. Analysis of Financial Time Series. – 3rd ed. Wiley. 2010.

67. Li S. Jin X. Xuan L. Zhou. X. Chen. W. Wang. Y. Yan. X. Feng. L. Enhancing the Locality and Breaking the Memory Bottleneck of Transformer on Time Series Forecasting NeurIPS. 2020.

68. Shumway R. H. Stoffer D. S. Time Series Analysis and Its Applications : With R Examples. – 4th ed. Springer. 2017.

69. Sezer O. B. Gudelek. M. U. Ozbayoglu. A. M. Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review : 2005–2019 Applied Soft Computing. 2020. Vol. 90. Article ID 106181.

ДОДАТОК А

Лістинг програми

```
import pandas as pd
import numpy as np

# Спроба зчитування CSV із робочого каталогу
try:
    df = pd.read_csv('GDP.csv')

    # Видалення регіональних об'єктів
    regions = [
        "Arab World", "Sub-Saharan Africa", "Europe & Central Asia", "High income",
        "East Asia & Pacific", "World", "Europe & Central Asia (excluding high income)",
        "East Asia & Pacific (excluding high income)", "Early-demographic dividend",
        "Fragile and conflict affected situations"
    ]
    df = df[~df['Country'].isin(regions)]

    # Перетворення стовпців з роками на числові
    years = [str(year) for year in range(1960, 2023)]
    df[years] = df[years].apply(pd.to_numeric, errors='coerce')

    # Логарифмічне перетворення
    df_log = df.copy()
    df_log[years] = np.log1p(df_log[years])
```

```
        # Транспонування у формат long
        df_melted = df_log.melt(
            id_vars=['Country', 'Country Code'],
            value_vars=years,
            var_name='Year',
            value_name='LogGDP'
        )
        df_melted['Year'] = df_melted['Year'].astype(int)

        # Виведення перших 10 рядків результату
        print(df_melted.head(100))
    except FileNotFoundError:
        print("Файл 'GDP.csv' не знайдено в робочому каталозі. Будь ласка, завантажте його в поточний каталог.")

    # Параметр: максимальна кількість пропусків для однієї країни
    threshold_missing = 20

    # Обчислюємо кількість пропусків та загальну кількість записів на країну
    missing_counts = df_melted.groupby('Country')['LogGDP'].apply(lambda x: x.isna().sum())
    total_counts = df_melted.groupby('Country')['LogGDP'].count() + missing_counts

    # Визначаємо країни для видалення та для імпутації
    countries_to_drop = missing_counts[missing_counts > threshold_missing].index.tolist()
    countries_to_impute = missing_counts[(missing_counts > 0) & (missing_counts <= threshold_missing)].index.tolist()
```

```
# Видаляємо країни з надто великою кількістю пропусків
df_filtered = df_melted[~df_melted['Country'].isin(countries_to_drop)].copy()

# Імпутація пропусків середнім значенням для кожної країни
country_means = df_filtered.groupby('Country')['LogGDP'].transform('mean')
df_filtered['LogGDP'] = df_filtered['LogGDP'].fillna(country_means)

# Виведемо результати
print(f"Країн із > {threshold_missing} пропусків видалено: {len(countries_to_drop)}")
print(f"Країн із <= {threshold_missing} пропусків імпутовано: {len(countries_to_impute)}")
print("\nПерші 10 рядків після обробки:")
print(df_filtered.head(10))

import matplotlib.pyplot as plt

# Вибір топ-6 країн за середнім LogGDP (після очищення та імпутації)
top_countries = df_filtered.groupby('Country')['LogGDP'].mean().nlargest(6).index.tolist()

# Графіки часових рядів для топ-6 країн
for country in top_countries:
    country_data = df_filtered[df_filtered['Country'] == country]
    plt.figure()
    plt.plot(country_data['Year'], country_data['LogGDP'])
    plt.title(f"Часовий ряд Log(GDP) для {country}")
    plt.xlabel('Year')
    plt.ylabel('Log(GDP)')
```

```
    plt.tight_layout()

    # Гістограма розподілу кількості пропусків по країнах (до очищення)
    original_missing = df_melted.groupby('Country')['LogGDP'].apply(lambda x: x.isna().sum())
    plt.figure()
    plt.hist(original_missing, bins=20)
    plt.title('Розподіл кількості пропущених значень у LogGDP по країнах')
    plt.xlabel('Кількість пропусків')
    plt.ylabel('Кількість країн')
    plt.tight_layout()

    # Гістограма розподілу LogGDP країн у 2022 році (після імпутації)
    data_2022 = df_filtered[df_filtered['Year'] == 2022]['LogGDP'].dropna()
    plt.figure()
    plt.hist(data_2022, bins=30)
    plt.title('Розподіл Log(GDP) країн у 2022 році')
    plt.xlabel('Log(GDP) 2022')
    plt.ylabel('Кількість країн')
    plt.tight_layout()

    plt.show()



    import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt

    from sklearn.linear_model import LinearRegression, Ridge, Lasso
    from sklearn.tree import DecisionTreeRegressor
```

```
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.svm import SVR
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error

import statsmodels.api as sm

# 1. Завантаження та первинна обробка
df = pd.read_csv('GDP.csv')
years = [str(year) for year in range(1960, 2023)]
df[years] = df[years].apply(pd.to_numeric, errors='coerce')

# Видалення групових регіональних записів
to_remove = [
    "Arab World", "Sub-Saharan Africa", "Europe & Central Asia", "High income",
    "East Asia & Pacific", "World",
    "Europe & Central Asia (excluding high income)",
    "East Asia & Pacific (excluding high income)",
    "Early-demographic dividend", "Fragile and conflict affected situations"
]
df = df[~df['Country'].isin(to_remove)]

# Логарифмування
df_log = df.copy()
df_log[years] = np.log1p(df_log[years])

# Перехід в довгий формат
df_long = df_log.melt(
```

```
        id_vars=['Country', 'Country Code'],
        value_vars=years,
        var_name='Year',
        value_name='LogGDP'
    )
    df_long['Year'] = df_long['Year'].astype(int)

    # 2. Імпутація пропусків
    threshold = 20
    missing_counts = df_long.groupby('Country')['LogGDP'].apply(lambda x:
x.isna().sum())
    to_drop = missing_counts[missing_counts > threshold].index
    df_filtered = df_long[~df_long['Country'].isin(to_drop)].copy()

    # Заповнення NaN середнім по країні
    df_filtered['LogGDP'] = df_filtered.groupby('Country')['LogGDP']\
        .transform(lambda x: x.fillna(x.mean()))

    # Переконаємося, що NaN більше немає
    df_filtered = df_filtered.dropna(subset=['LogGDP'])

    # 3. Налаштування моделей та країн
    countries = ['USA','CHN','JPN','DEU','FRA','GBR']
    names = {
        'USA':'United States','CHN':'China','JPN':'Japan',
        'DEU':'Germany','FRA':'France','GBR':'United Kingdom'
    }
    models = {
        'Linear': LinearRegression(),
        'Ridge': Ridge(),
```

```
        'Lasso': Lasso(),
        'Tree': DecisionTreeRegressor(max_depth=5, random_state=0),
        'Forest': RandomForestRegressor(n_estimators=100, random_state=0),
        'SVR': SVR(),
        'GBM': GradientBoostingRegressor(n_estimators=100, random_state=0)
    }

    # 4. Навчання, оцінка, візуалізація
    for code in countries:
        df_c = df_filtered[df_filtered['Country Code']==code].sort_values('Year')
        X = df_c['Year'].values.reshape(-1,1)
        y = df_c['LogGDP'].values
        # split
        X_train, X_test = X[:-5], X[-5:]
        y_train, y_test = y[:-5], y[-5:]

        preds = {}
        print(f"\n=== {names[code]} ===")
        # ML моделі
        for m_name, m in models.items():
            m.fit(X_train, y_train)
            y_pred = m.predict(X_test)
            mae = mean_absolute_error(y_test, y_pred)
            mse = mean_squared_error(y_test, y_pred)
            rmse = np.sqrt(mse)
            mape = np.mean(np.abs((y_test - y_pred)/y_test))*100
                        print(f"{m_name:8}:     MAE={mae:.3f},     MSE={mse:.3f},
RMSE={rmse:.3f}, MAPE={mape:.2f}%")
            preds[m_name] = y_pred
```

```
        # ARIMA
        ar = sm.tsa.ARIMA(y_train, order=(1,1,1)).fit()
        y_ar = ar.forecast(steps=5)
        mae = mean_absolute_error(y_test, y_ar)
        mse = mean_squared_error(y_test, y_ar)
        rmse = np.sqrt(mse)
        mape = np.mean(np.abs((y_test - y_ar)/y_test))*100
         print(f"ARIMA    : MAE={mae:.3f}, MSE={mse:.3f}, RMSE={rmse:.3f},
MAPE={mape:.2f}%")
        preds['ARIMA'] = y_ar

        # Графік порівняння
        plt.figure(figsize=(7,4))
        plt.plot(X_test.flatten(), y_test, 'ko-', label='Actual', linewidth=2)
        for m_name, y_pred in preds.items():
           plt.plot(X_test.flatten(), y_pred, '--x', label=m_name)
        plt.title(f"Forecast vs Actual – {names[code]}")
        plt.xlabel('Year')
        plt.ylabel('Log(GDP)')
        plt.legend(fontsize='small')
        plt.tight_layout()
        plt.show()
```

ДОДАТОК Б

Презентаційні матеріали

ДИПЛОМНА РОБОТА НА ТЕМУ:

Інформаційна ситстема «Прогнозування фінансових часових рядів»

Виконав: Студент IV курсу, групи КА-12
Шинкарчук Денис Дмитрович
Керівник: Доцент, к.т.н.
Селін Юрій Миколайович

1

Об’єкт та предмет дослідження

Об'єкт дослідження – Фінансові часові ряди, зокрема історичні дані валового внутрішнього продукту (ВВП) країн світу з 1960 по 2022 рік.

Предмет дослідження – Інформаційні моделі прогнозування часових рядів ВВП та засоби їх реалізації.

2

Мета роботи

Мета роботи – поліпшити прогнозні оцінки часових рядів економічної природи.

Актуальність роботи полягає у потребі в адаптивних моделеях прогнозування. Їх реалізаця дозволяє органам державного управління, інвесторам і підприємствам приймати обґрунтовані рішення в нестабільний час.

3

ПЕРЕЛІК ЗАВДАНЬ:

- Провести аналіз предметної області та наявних підходів до прогнозування фінансових часових рядів;
- Розробити інформаційну систему для реалізації моделей аналізу та прогнозування та провести її експериментальну оцінку;
- Визначити класи моделей, що найкраще підходять для прогнозування ввп;
- Здійснити візуалізацію результатів.

4

ОСНОВНІ ПОНЯТТЯ ТЕОРІЇ ЧАСОВИХ РЯДІВ

01
Стаціонарність

02
Тренд

03
Сезонність

04
Автокореляція

05
Стохастичний процес

06
Спектральний аналіз

07
Інтегрованність

08
Білий шум

5

Класифікація методів прогнозування

Класичні статистичні методи

AR, MA, ARIMA

Методи машинного навчання

Лінійна регресія, дерева рішень

Гібридні та інтегровані моделі

LSTM, Transformer

6

Огляд інформаційних систем

Корпоративні

SAS Forecast Server,
IBM SPSS Forecasting

Мови програмування

Python, C++, Julia

Хмарні сервіси

Google Cloud AI Platform,
Amazon Forecast

7

Вибір мови програмування

Було проведено аналіз мов, таких як Python, R, Matlab, Julia та C++. В результаті виявлено, що **Python** посідає провідні позиції за всіма ключовими критеріями, такими як: наявність бібліотек для статистики та машинного навчання, підтримка нейронних мереж, зручність візуалізації та швидкодія.

8

Вибір середовища розробки

Як основне середовище розробки обрано **Jupyter Notebook** завдяки його інтерактивності, зручності візуалізації, підтримці блокової структури, можливості поєднання коду і тексту, а також легкості інтеграції з сучасними бібліотеками машинного навчання і глибокого навчання.

9

ВИБІР СТРУКТУРИ ДАНИХ

Датасет «World GDP by Country: 1960–2022», платформа Kaggle

Column name	Description
Country	Name of the Country
Country Code	3 Letter Country Code
1960	GDP in 1960
...	
2022	GDP in 2022

10

Отримані дані

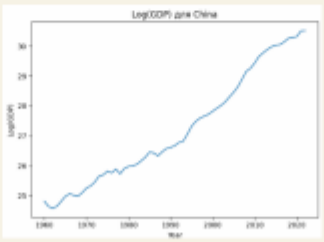
Після аналізу датасету відносно топ 6 країн за середнім значенням логарифму ВВП можна виділити такі моменти:

1. У всіх країн буде загальне зростання LogGDP з 1960-х до 2020-х (експоненційне зростання ВВП).
2. Помітні просідання на кризових періодах: 2008 (фінансова криза), 2020 (COVID-19).
3. Китай показує різкий стрибок після 1990-х — ефект економічних реформ.
4. Після стрімкого зростання до 1990-х — уповільнення через економічну "втрату десятиліття".

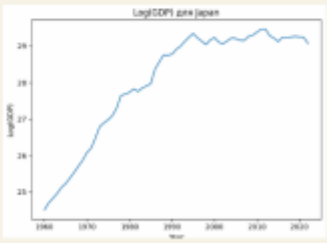
11

ВІЗУАЛЬНА ПЕРЕВІРКА

GDP USA



GDP CHINA



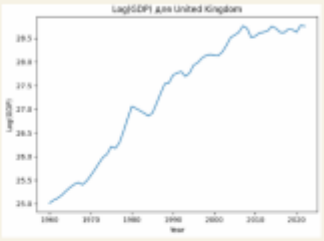
GDP JAPAN

12

ВІЗУАЛЬНА ПЕРЕВІРКА

GDP GERMANY

GDP FRANCE



GDP UK

13

Структура інформаційної системи

01 Первинне завантаження даних

02 Робота над датасетом та формування вибірки

03 Побудова та оцінка машинних моделей та ARIMA

04 Обчислення метрик для оцінки якості прогнозу

05 Виведення результатів

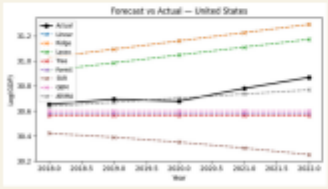
14

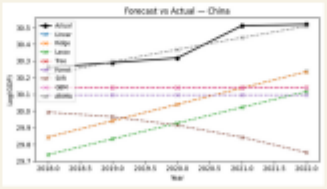
Оцінка якості прогнозу, MAPE

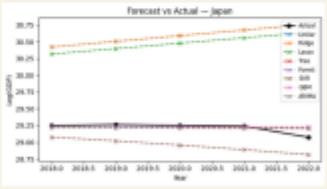
Модель	USA	China	Japan	Germany	France	UK
Linear	1.38	1.12	4.71	0.13	2.71	2.72
Ridge	1.38	1.12	4.71	0.13	2.71	2.72
Lasso	1.02	1.49	4.33	0.1	2.32	2.33
Decision Tree	0.55	0.78	0.19	0.07	0.27	0.27
Random Forest	0.51	0.93	0.17	0.06	0.28	0.26
SVR	1.27	1.6	0.9	0.07	1.16	1.25
Gradient Boosting	0.64	0.79	0.17	0.11	0.26	0.29
ARIMA	0.14	0.12	0.19	0.29	0.18	0.29

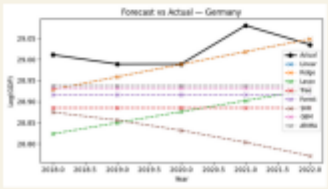
15

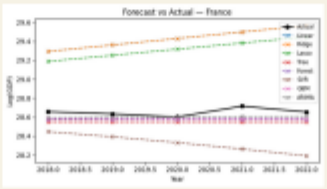
Візуалізація результатів

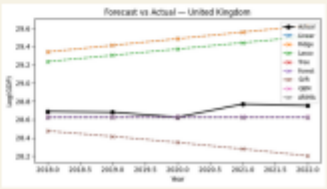












16

Висновки

У результаті виконання повного дослідницького циклу було створено, протестовано та проаналізовано інформаційну систему прогнозування фінансових часових рядів на прикладі динаміки ВВП різних країн світу в логарифмічному представленні. Побудована система ґрунтується на інтеграції математичної статистики, класичної моделі часових рядів ARIMA та 7 сучасних алгоритмів машинного навчання в єдину аналітичну архітектуру. Також була реалізована візуалізація прогнозів у вигляді графіків. Проведена робота відкриває перспективи подальшого розвитку: можлива інтеграція багатовимірних моделей, поєднання з геоекономічними показниками або новинними потоками.

17

ДЯКУЮ ЗА УВАГУ

18