

МІНІСТЕРСТВО ФІНАНСІВ УКРАЇНИ
ДЕРЖАВНИЙ ПОДАТКОВИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ФІНАНСІВ ТА ЦИФРОВИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І
СИСТЕМ

КВАЛІФІКАЦІЙНА РОБОТА
на тему:

«Розробка моделі машинного навчання для аналізу макроекономічних
показників України»

Здобувача вищої освіти першого (бакалаврського) рівня,
4 курсу, групи КБ-19-1, спеціальності 122 «Комп'ютерні науки»,
освітня програма «Інформаційні управляючі системи і технології (за галузями)»



(підпис)

Шпаковича Володимира
Івановича

Науковий керівник



(підпис)

Редич Олександр
Володимирович
(прізвище, ім'я, по батькові, науковий ступінь,
вчене звання)

Керівник освітньо-
професійної програми



(підпис)

Федотова - Півень Ірина
Миколаївна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь,
вчене звання)

Допущено до захисту за рішенням кафедри
протокол від «26» травня 2023 № 13

Завідувач кафедри



(підпис)

Філоненко Михайло
Миколайович, к.ф.-м.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Реєстрація № 15

05.06.2023

(дата)



(підпис відповідальної особи)

Результати перевірки

до захисту

(до захисту, на доопрацювання)

29.05.2023

(дата)



(підп

Результат захисту

5 (відмінно)

(оцінка)

15.06.2023

(дата)



(підп

Голова комісії:



(підпис)

В.М. Франчук

Члени комісії:



(підпис)

М.М. Філоненко



(підпис)

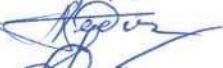
О.В. Редич



(підпис)

Є.В. Котух

БЛАНК ОЦІНЮВАННЯ ВИПУСКНОЇ РОБОТИ

Об'єкти контролю	Діапазон оцінки в балах	Фактична кількість балів	Примітки	Дата, П.І.Б.: керівника, голови комісії членів комісії, секретаря, (підписи)
ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ КОМІСІЄЮ ПО ЗАХИСТУ				
1. Зміст випускної роботи (якість теоретичного матеріалу, його зв'язок з практичним, вміння представляти матеріал в логічній послідовності, наявність висновків та пропозицій)	від 0 до 30	28		15.06.2023 Франчук В.М.  Філоненко М.М.  Редич О.В.  Котух Є.В.  Суховерхівка Л.П. 
2. Дотримання вимог щодо оформлення	від 0 до 10	7		
3. Вчасне виконання роботи на всіх етапах у встановлені кафедрою терміни	від 0 до 5	4		
4. Рецензент	від 0 до 5	4		
РАЗОМ БАЛІВ	від 0 до 50	43		

Оцінювання кваліфікаційної роботи науковим керівником - 47 балів.

ПЕРЕВЕДЕННЯ ОЦІНКИ ЗА 100-БАЛЬНОЮ ШКАЛОЮ ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою	Дата	П.І.Б.: голови, членів комісії, секретаря, (підпис)
90	A	5 (відмінно)	15.06.2023	Франчук В.М.  Філоненко М.М.  Редич О.В.  Котух Є.В.  Суховерхівка Л.П. 

ПРАВИЛА ПЕРЕВЕДЕННЯ ОЦІНОК ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою
90-100	A	5 (відмінно)
80-89	B	4 (добре)
70-79	C	
60-69	D	
50-59	E	3 (задовільно)
35-49	FX	2 (незадовільно)

Факультет фінансів та цифрових технологій

Кафедра комп'ютерних та інформаційних технологій і систем

Галузь знань: 12 «Інформаційні технології»

Спеціальність: 122 «Комп'ютерні науки»

Освітня програма: «Інформаційні управляючі системи та технології (за галузями)»

Затверджую



Завідувач кафедри

М.М. Філоненко

«20» 11 2022 р.

Завдання на кваліфікаційну роботу (проект)

здобувачу вищої освіти

Шпаковичу Володимиру Івановичу

1. Тема дипломної роботи

«Розробка моделі машинного навчання для аналізу макроекономічних показників України»

Затверджена наказом ректора від «15» листопада 2022 № 1205 (зі змінами наказ від «04» квітня 2023 № 361)

2. Строк здачі закінченої роботи: «20» травня 2023

3. Мета роботи: створення моделі машинного навчання для аналізу макроекономічних показників України

Об'єкт дослідження: інформаційні ресурси макроекономічних показників України

Предмет дослідження: бібліотеки машинного навчання мови Python, зокрема бібліотеки аналізу часових рядів

Орієнтовна база дослідження: відділи інформаційного забезпечення Міністерства фінансів України та інших центральних відомств

4. Вихідні дані до роботи: Інформаційні ресурси Мінфіну та Макроекономічні показники України

5. Перелік графічного матеріалу: рисунки та графіки, формули, що пояснюють суть застосованих методів та засобів. Структурні схеми та таблиці, графіки результатів моделювання.

6. Зміст випускної кваліфікаційної (дипломної) роботи (перелік питань за кожним розділом) :

РОЗДІЛ 1. ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ АСПЕКТИ ML

1.1. Аналіз існуючих технологій машинного навчання

1.2. Огляд інформаційних ресурсів макроекономічних показників

1.4. Огляд бібліотек мови Python для побудови моделей машинного навчання

РОЗДІЛ 2. РОЗРОБКА МОДЕЛІ ML

2.1. Статистичний аналіз макроекономічних даних

2.2. Вибір методу машинного навчання

2.3. Навчання моделі на навчальній вибірці

2.4. Тестування моделі та оцінка якості моделі

РОЗДІЛ 3. ПЕРСПЕКТИВИ РОЗВИТКУ МОДЕЛІ

3.1. Рекомендації щодо подальшого застосування моделі

7. Календарний план виконання роботи

№ п/п	Назва етапів випускної кваліфікаційної (дипломної) роботи	Строк виконання етапів роботи	
		За планом	Фактично
1	2	3	4
1	Вибір теми, підбір літературних джерел, формування структури ДР	До 15.11.2022 р.	« 15 » 11 2022
2	I розділ (розкрити сутність і стан дослідження наукової проблематики роботи, після чого навести її загальну характеристику; огляд літератури за темою і вибір конкретних напрямів досліджень)	26. 03. 2023 р.	« 26 » 03 2023
3	II розділ (аналіз експериментальної частини про і методики досліджень)	20. 04. 2023 р.	« 20 » 04 2023
4	III розділ (проведені теоретичні і (або) експериментальні дослідження та проектні рішення)	12. 05. 2023 р.	« 12 » 05 2023
5	Висновки	18. 05. 2023 р.	« 18 » 05 2023
6	Доопрацювання роботи з урахуванням зауважень наукового керівника	20.05.2023 р.	« 20 » 05 2023
7	Отримання відгуку керівника	03.06.2023 р.	« 03 » 06 2023
8	Отримання зовнішньої рецензії	05.06.2023р.	« 05 » 06 2023

Дата видачі завдання « 20 » листопада 2022 р.

Керівник дипломної роботи



О.В. Редич

(підпис)

Керівник освітньо-професійної програми



І.М. Федотова-Г

(підпис)

Завдання до виконання отримав
здобувач вищої освіти



В.І. Шпакович

(підпис)

АНОТАЦІЯ

Метою даного дипломного є створення моделі машинного навчання для аналізу макроекономічних показників України. Обробка даних та створення даних за допомогою бібліотек Python.

Дві основні цілі аналізу часових рядів - визначення характеру часового ряду та прогнозування (передбачення майбутніх значень часового ряду на основі поточних і минулих значень). Обидві ці цілі вимагають наявності моделі часового ряду, яка має бути ідентифікована та більш-менш формально описана. Після того, як модель визначена, її можна використовувати для інтерпретації представлених даних.

В рамках дипломного проекту, задля реалізації даного алгоритму та прогнозування нестационарних часових рядів було обрано мову програмування Python.

Використовуючи даний алгоритм аналізу часових рядів можливо аналізувати та спрогнозувати часові ряди з різних макроекономічних показників України.

Ключові слова: алгоритм, алгоритм аналізу часових рядів, часовий ряд, методи аналізу, машинне навчання, нестационарні часові ряди, стаціонарні часові ряди Python.

ANNOTATION

The purpose of this thesis is to create a machine learning model for the analysis of macroeconomic indicators of Ukraine. Data processing and creation of datasets using Python libraries.

The two main goals of time series analysis are determining the nature of the time series and forecasting (prediction of future values of the time series based on current and past values). Both of these goals require a time series model, which must be identified and more or less formally described. Once a model is defined, it can be used to interpret the data presented.

As part of the diploma project, the Python programming language was chosen to implement this algorithm and forecast non-stationary time series.

Using this time series analysis algorithm, it is possible to analyze and forecast time series from various macroeconomic indicators of Ukraine.

Keywords: algorithm, time series analysis algorithm, time series, analysis methods, machine learning, non-stationary time series, stationary Python time series.

Зміст

ВСТУП.....	3
1. ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ АСПЕКТИ ML.....	8
1.1. Аналіз існуючих технологій машинного навчання.....	8
1.2. Огляд інформаційних ресурсів макроекономічних показників.....	12
1.3. Огляд бібліотек мови Python для побудови моделей машинного навчання.....	15
2. РОЗРОБКА МОДЕЛІ ML.....	27
2.1. Статистичний аналіз макроекономічних даних.....	27
2.2. Вибір методу машинного навчання.....	31
2.3. Навчання моделі на навчальній вибірці.....	36
2.4. Тестування моделі та оцінка якості моделі.....	41
3. ПЕРСПЕКТИВИ РОЗВИТКУ МОДЕЛІ.....	44
3.1. Рекомендації щодо подальшого застосування моделі.....	44
ВИСНОВКИ.....	46
СПИСК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	47

ВСТУП

Актуальність теми. Сучасні дослідження макроекономічної динаміки, перехідних економічних процесів та фінансових ринків ґрунтуються на аналізі взаємозв'язків часових рядів соціально-економічних даних. Враховуючи часову структуру даних про реальні економічні процеси, можна правильно відобразити їх в економіці та фінансах. Врахування часової структури даних про реальні економічні процеси дозволяє адекватно відображати їх в економіко-математичних моделях. Визнання цього факту призвело до швидкого розвитку спеціальних методів аналізу таких даних. Знання цих методів та їх застосування для прогнозування соціально-економічних процесів.

Задача аналізу часових рядів не нова, однак з розвитком інформаційних технологій, з'явилися нові методи аналізу та відповідно задачі і алгоритми.

Дві основні цілі аналізу часових рядів - визначення характеру часового ряду та прогнозування (передбачення майбутніх значень часового ряду на основі поточних і минулих значень). Обидві ці цілі вимагають наявності моделі часового ряду, яка має бути ідентифікована та більш-менш формально описана. Після того, як модель визначена, її можна використовувати для інтерпретації представлених даних.

Метою дослідження є створення моделі машинного навчання для аналізу макроекономічних показників України. Обробка даних та створення дата сетів за допомогою бібліотек Python. Для досягнення поставленої мети були вирішені наступні завдання:

- аналітичний огляд реляційних та нереляційних баз даних;
- класифікація методів машинного навчання;
- вибір найбільш вагомих критеріїв та факторів, які впливають на макроекономічну складову України;
- прогнозування показників та аналіз;
- отримання точності на ключових метриках.

Об'єктом дослідження є бібліотеки мови Python, а саме:

- pandas (pd): для роботи з даними у форматі таблиці (DataFrame) та часовими рядами.
- google.colab: для завантаження файлів та роботи в середовищі Google Colab.
- sklearn.metrics: для розрахунку метрик оцінки моделей, таких як середня абсолютна помилка (mean_absolute_error), середня квадратична помилка (mean_squared_error) та коефіцієнт детермінації (r2_score).
- pandas.date_range: створення діапазону дат.
- pandas.to_datetime: для перетворення даних у формат дати та часу.
- numpy (np): для роботи з числовими даними та математичними операціями.
- statsmodels.api: для роботи з моделями часових рядів, включаючи ARIMA.
- statsmodels.tsa.arima_model: для роботи з моделлю ARIMA.
- statsmodels.tsa.seasonal: для сезонної декомпозиції часових рядів.
- matplotlib.pyplot (plt): для візуалізації даних.
- statsmodels.graphics.tsaplots: для побудови автокореляційної функції (ACF) та часткової автокореляційної функції (PACF) тимчасового ряду.
- statsmodels.tsa.stattools: для проведення тесту на стаціонарність часового ряду за допомогою функції adfuller.

Ці бібліотеки надають різні функції та інструменти для роботи з тимчасовими рядами, аналізу та прогнозування.

Предметом дослідження є макроекономічна статистика України.

База дослідження. НАБУ та МІНФІН - це організації, пов'язані з управлінням фінансовими ресурсами та боротьбою з корупцією в Україні.

НАБУ (Національне антикорупційне бюро України):

- НАБУ є незалежним органом, створеним для боротьби з корупцією у державному секторі України.

- Його завдання включають проведення розслідувань, запобігання та припинення корупційних злочинів, виявлення та притягнення до відповідальності винних осіб.
- НАБУ працює у тісній співпраці з іншими правоохоронними органами, судами та міжнародними організаціями для забезпечення ефективної протидії корупції.

МІНФІН (Міністерство фінансів України):

- МІНФІН України відповідає за управління фінансовою політикою держави та координацію фінансових операцій.
- У його обов'язки входять розробка та реалізація бюджетної політики, контроль та аналіз державних фінансів, управління державним боргом та фінансовими ресурсами, а також регулювання фінансових ринків.
- МІНФІН також відповідає за взаємодію з міжнародними фінансовими організаціями, такими як Міжнародний валютний фонд (МВФ) та Європейський банк реконструкції та розвитку (ЄБРР).

Обидві організації відіграють важливу роль у боротьбі з корупцією та управлінні фінансовими ресурсами в Україні. НАБУ займається розслідуваннями корупційних злочинів, а МІНФІН відповідає за розробку фінансової політики та управління державними фінансами.

Практична значимість. Створення моделі машинного навчання для аналізу та прогнозування макроекономічних показників України може мати кілька практичних цінностей:

- Прогнозування економічних показників: Модель може допомогти у прогнозуванні таких важливих показників, як ВВП, інфляція, безробіття та інші. Це може бути корисним для організацій, інвесторів та урядових органів, щоб приймати обґрунтовані рішення та планувати свої дії.
- Раннє виявлення економічних трендів і криз: Модель може допомогти в ранньому виявленні змін в економічній ситуації, таких як рецесії,

зростання або спад галузей, фінансові кризи і т. д. Це може допомогти вживати запобіжних заходів і адаптуватися до умов, що змінюються.

- **Аналіз впливу факторів:** Модель може допомогти у аналізі впливу різних факторів на макроекономічні показники. Наприклад, модель може показати, як зміна ставок відсотка, інвестицій, споживчих витрат або зовнішньоторговельних відносин може позначитися на економіці. Це може допомогти у плануванні політики та розробці стратегій.
- **Інформовані економічні рішення:** Модель може надати важливі інсайти для ухвалення поінформованих економічних рішень. Наприклад, на основі прогнозів моделі, організації можуть приймати рішення про розширення бізнесу, реорганізацію виробництва, залучення інвестицій або розробку заходів щодо стабілізації економіки.

Загалом створення моделі машинного навчання для аналізу та прогнозування макроекономічних показників України може допомогти покращити розуміння економічної ситуації, приймати більш обґрунтовані рішення та планувати майбутні дії з урахуванням впливу різних факторів на економіку.

Особистий внесок автора. Пропонується двоетапний алгоритм прогнозування, на першому етапі якого відстежуються властивості часового ряду, які зумовлюють його нестационарність, такі як тренд і сезонність, після чого зводять до стаціонарних. На другому етапі знаходимо автокореляцію та часткову автокореляцію, вони допомагають визначити наявність автокореляції у часовому ряду та допомагають визначити параметри моделі MA (Moving Average) у ARIMA. Також допомагає визначити наявність прямого зв'язку між поточним значенням і значенням певному лазі, і він використовується визначення параметрів моделі AR (Autoregressive) в ARIMA.

ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ АСПЕКТИ ML

1.1. Аналіз існуючих технологій машинного навчання

Машинне навчання (Machine learning, ML) – звід методів в області штучного інтелекту, набір алгоритмів, які застосовують, щоб створити машину, яка вчиться на власному досвіді. В якості навчання машина обробляє величезні масиви вхідних даних і знаходить у них закономірності.

Перша технологія машинного навчання, відома як перцептрон, була запропонована Френком Розенблаттом у 1957 році. Перцептрон був першим алгоритмом машинного навчання, здатним автоматично навчатися на основі вхідних даних та коригувати свої ваги для класифікації патернів. Він був натхненний роботою на польових теоріях нейронів та змодельовав процес навчання на основі нейронної активності.

Однак перші технології машинного навчання були обмежені у своїх можливостях та застосовувалися головним чином у вузькоспеціалізованих завданнях. Надалі розвиток машинного навчання відбувався поступово, з розробкою нових алгоритмів та методів, а також з розширенням обчислювальних потужностей комп'ютерів.

З появою потужніших обчислювальних систем та доступу до більших обсягів даних, технології машинного навчання почали розвиватися з вищою швидкістю. Сьогодні машинне навчання стало важливою та широко застосовуваною областю, здатною вирішувати складні завдання розпізнавання образів, обробки природної мови, аналізу даних та багато іншого.

В останні десятиліття технології машинного навчання продовжили розвиватися та застосовуватись у різних галузях. Ось деякі ключові моменти та досягнення в історії машинного навчання:

1980-і роки: У цей час дослідники активно займалися розробкою нових алгоритмів машинного навчання, таких як метод опорних векторів (Support Vector Machines) та вирішальні дерева (Decision Trees). Вони стали широко використовуваними методами завдання класифікації і регресії.

1990-і роки: В цей час почалося розширення застосування машинного навчання в різних галузях, включаючи комп'ютерний зір, розпізнавання мови та

обробку природної мови. Були розроблені методи глибокого навчання, такі як згорткові нейронні мережі (Convolutional Neural Networks) та рекурентні нейронні мережі (Recurrent Neural Networks).

2000-і роки: З розвитком Інтернету та доступом до великих обсягів даних, машинне навчання стало застосовуватися в галузі інтернет-пошуку, рекомендаційних систем, аналізу соціальних мереж та інших галузей. У цей період також розвивалися методи ансамблів (Ensemble Methods), які поєднують декілька моделей для підвищення точності передбачень.

2010-ті роки: Глибоке навчання стало центральним фокусом досліджень та застосування у машинному навчанні. Були розроблені більш глибокі та складніші архітектури нейронних мереж, такі як глибокі згорткові нейронні мережі (Deep Convolutional Neural Networks) та рекурентні нейронні мережі довгої короткострокової пам'яті (Long Short-Term Memory Networks). Також з'явилися фреймворки та бібліотеки, такі як TensorFlow, PyTorch та Keras, які спростили розробку та використання глибоких моделей.

Сьогодні машинне навчання продовжує розвиватися і застосовуватися в багатьох областях, включаючи медицину, фінанси, робототехніку, автоматизацію процесів та багато іншого. Зі зростанням доступності даних та покращенням алгоритмів, очікується, що машинне навчання продовжуватиме вносити значний внесок у різні аспекти нашого життя та сучасного суспільства.

Машинне навчання застосовується у багатьох галузях та технологіях. Ось кілька прикладів технологій, створених з використанням машинного навчання:

Рекомендаційні системи: Багато платформ та сервісів, таких як YouTube, Netflix, Amazon та Spotify, використовують машинне навчання для надання персональних рекомендацій контенту або продуктів користувачам на основі їх переваг та поведінки.

Обробка природної мови (Natural Language Processing, NLP): Технології NLP, такі як автоматичне розпізнавання мови, машинний переклад, аналіз тональності текстів та генерація тексту, ґрунтуються на алгоритмах машинного навчання.

Комп'ютерний зір: Машинне навчання використовується для обробки та аналізу зображень та відео. Це включає розпізнавання об'єктів, детектування осіб, сегментацію зображень, а також автоматичну анотацію та класифікацію вмісту.

Медична діагностика: Машинне навчання застосовується для аналізу медичних даних та зображень, допомагаючи у діагностиці різних захворювань, прогнозуванні їх розвитку та підборі найбільш ефективного лікування.

Автоматичне водіння: Технології самоврядних автомобілів та автопілотів

засновані на алгоритмах машинного навчання, які дозволяють автомобілю аналізувати довкілля, приймати рішення та керувати рухом.

Фінансовий аналіз: Машинне навчання застосовується у фінансових інструментах та системах для прогнозування цін на акції, визначення трендів ринку, автоматичної торгівлі та ризик-аналізу.

Класифікація та виявлення шахрайства: Машинне навчання використовується для визначення підозрілих активностей та виявлення шахрайських операцій у банківських системах та платіжних системах.

Це лише деякі приклади технологій, створених із використанням машинного навчання. Машинне навчання охоплює широкий спектр додатків і продовжує прогресувати, вносячи значні зміни до багатьох галузей та областей діяльності.

Технології машинного навчання охоплюють різні методи, алгоритми та інструменти, які дозволяють комп'ютерним системам отримувати знання з даних та автоматично навчатися на основі досвіду. Нижче наведено деякі основні технології, що використовуються в галузі машинного навчання:

- 1) Штучні нейронні мережі Це технологія, яка імітує роботу людського мозку і складається з мережі взаємопов'язаних штучних нейронів. Нейронні мережі широко використовуються в глибокому навчанні і здатні обробляти складні дані, розпізнавати образи, розпізнавати мовлення та багато іншого.
- 2) Вирішальні дерева: Вирішальні дерева є графічною структурою, що складається з гілок і вузлів, які приймають рішення на основі різних ознак і умов. Вони використовуються для класифікації завдань і регресії і можуть бути легко інтерпретовані.
- 3) Алгоритми кластеризації: Алгоритми кластеризації використовуються для групування об'єктів або даних у схожі кластери на основі їхньої подібності або близькості. Це дозволяє виявляти приховані структури та патерни у даних.
- 4) Методи оптимізації: Методи оптимізації використовуються для налаштування параметрів моделі для досягнення оптимальної продуктивності. Вони включають градієнтний спуск, стохастичний градієнтний спуск та інші методи, які допомагають знаходити локальні або глобальні оптимуми функцій втрат.
- 5) Методи регулювання: Методи регулювання використовуються для запобігання перенавчанню моделі шляхом додавання штрафів за складність моделі або обмежень на параметри моделі. Деякі з популярних

методів регулювання включають L1 і L2 регулювання.

- 6) Методи обробки природної мови (Natural Language Processing, NLP): Методи NLP дозволяють комп'ютерам обробляти та аналізувати текстові дані, розуміти природну мову та виконувати завдання, такі як розпізнавання мови, машинний переклад, сентимент-аналіз та багато іншого.
- 7) Рекомендаційні системи: Рекомендаційні системи використовуються для передбачення переваг користувачів та рекомендації персоналізованих товарів, послуг або контенту. Вони ґрунтуються на методах колаборативної фільтрації, контентної фільтрації чи гібридних підходах.

Висновок.

Це лише деякі з багатьох методів машинного навчання, які використовуються для вирішення різних завдань. Кожен метод має свої особливості та застосовність залежно від типу даних, завдання та вимог проекту.

1.2. Огляд інформаційних ресурсів макроекономічних показників.

Відповідно до міжнародних стандартів система національних рахунків включає низку взаємопов'язаних макроекономічних показників. На макrorівні найважливіші економічні показники об'єднуються, виявляються найважливіші зв'язки між ними і на цій основі проводяться аналітичні розрахунки для прогнозування змін макроекономічних пропорцій та визначення можливостей врахування та управління впливом різних факторів на соціально-економічні процеси.

Макроекономічні показники включають: валовий внутрішній продукт, національний дохід, валовий національний продукт, чистий національний продукт, національний дохід, особистий дохід і наявний дохід.

Валовий внутрішній продукт (ВВП) одна із ключових макроекономічних показників, який вимірює загальну вартість всіх кінцевих товарів та послуг, вироблених протягом певного часового періоду у країні.

Україна, розташована у Східній Європі, є однією з найбільших країн на континенті та має значний вплив на економіку регіону. ВВП України є важливим показником, що відображає економічний стан та рівень розвитку країни.

В останні роки Україна зіткнулася з низкою економічних викликів та періодами нестабільності. Однак, з покращенням внутрішньої політичної ситуації та реформами, країна поступово рухається до економічного зростання та розвитку.

Основні галузі української економіки, які роблять значний внесок у ВВП, включають промисловість, сільське господарство, будівництво, інформаційні технології та послуги. Україна також має значний потенціал у сферах експорту, особливо у галузі сільськогосподарських продуктів, металургії та IT-послуг.

Однак важливо відзначити, що Україна все ще стикається з деякими викликами, такими як корупція, нестабільність політичної ситуації, війна на сході країни та нестача інвестицій. Ці фактори впливають на економічний розвиток та потенціал зростання ВВП.

Для сталого економічного зростання та підвищення ВВП України країні необхідні подальші реформи, залучення інвестицій, поліпшення інфраструктури, розвиток людського капіталу та підтримка інновацій. Зусилля зі скорочення бюрократії, покращення ділового клімату та захисту прав власності також можуть сприяти економічному зростанню та підвищенню ВВП.

Статистику ВВП України можна знайти на таких офіційних сайтах, як:

- 1) Офіційний сайт Національного банку України: На офіційному сайті Національного банку України (<https://bank.gov.ua/>) ви можете знайти різні макроекономічні дані, включаючи дані щодо інфляції, валютного курсу, ставок відсотка та інших показників.
- 2) Державна служба статистики України: На сайті Державної служби статистики України (<http://www.ukrstat.gov.ua/>) ви знайдете офіційну статистичну інформацію щодо різних аспектів економіки України, включаючи ВВП, зайнятість, індустриальне виробництво та інші показники.
- 3) Міністерство економічного розвитку, торгівлі та сільського господарства України: На офіційному сайті Міністерства економічного розвитку, торгівлі та сільського господарства України (<https://www.me.gov.ua/>) можна знайти інформацію про макроекономічний розвиток країни, економічні прогнози та інші релевантні дані.
- 4) Міжнародний валютний фонд (МВФ): Міжнародний валютний фонд (<https://www.imf.org/>) надає дані та звіти про макроекономічну ситуацію в різних країнах, включаючи Україну. Можна знайти інформацію про фінансові показники, програму економічної підтримки та інші макроекономічні аспекти.
- 5) Національне антикорупційне бюро України: Сайт НАБУ(<https://nabu.ua/ua/vvp-2.html>)) є офіційним ресурсом, що надає доступ до інформації щодо боротьби з корупцією в Україні. Він призначений для громадян, журналістів, правоохоронних органів та інших зацікавлених сторін, які шукають інформацію про роботу НАБУ та проблему корупції в країні.
- 6) Міністерство фінансів України (<https://index.minfin.com.ua/>) є інформаційним ресурсом, який надає різні дані та статистику щодо фінансової ситуації в Україні. На цьому сайті можна знайти широкий спектр інформації, пов'язаної із фінансовою звітністю, бюджетом, податками, боргами та іншими фінансовими аспектами країни.

Для аналізу ВВП України будуть використовуватися сайти НаБу та Міністерства фінансів України. Ці два інформаційні джерела мають все що треба для побудови в майбутньому часових рядів та створенні на основі цих даних моделі для аналізу макроекономічних показників. Перелік особливостей сайту НаБу :

- а) ПРО НАБУ: Розділ містить загальну інформацію про бюро, його цілі, завдання, законодавчу основу та організаційну структуру.
- б) Новини: Тут ви знайдете актуальні новини та прес-релізи, пов'язані з діяльністю НАБУ та результатами проведених розслідувань.

- c) Діяльність: Розділ описує основні напрямки роботи НАБУ, включаючи процедури розслідування, проведення оперативно-розшукових заходів та співробітництво з іншими органами правопорядку.
- d) Звітність: Тут надаються звіти про виконані завдання, результати розслідувань та статистичні дані про корупційні злочини.
- e) Публікації: Розділ включає публікації, дослідження та аналітичні матеріали, пов'язані з корупцією та превентивними заходами щодо її протидії.
- f) Контакти: Тут надано контактну інформацію для зв'язку з НАБУ, включаючи адреси, телефони та електронні адреси.

Саме в розділі ПРО НАБУ можна знайти аналітику ВВП України за роками та скачати його в форматі .xls.

Сайт НБУ також є не мало важливим та може надавати чимало корисної інформації. Цей ресурс дає доступ до різних розділів та сервісів, включаючи:

- a) Офіційні публікації: офіційні документи, звіти, рішення правління та іншу регуляторну інформацію, яку надає Національний банк України.
- b) Статистика: Сайт надає широкий спектр статистичних даних, включаючи дані про грошовий обіг, банківську систему, інфляцію, валютні резерви та інші ключові показники економіки України.
- c) Монетарна політика: Тут можна отримати інформацію про монетарну політику НБУ, включаючи ставки відсотка, операції на відкритому ринку та макроекономічні прогнози.
- d) Регулювання та нагляд: Сайт містить інформацію про банківське регулювання, нагляд, правила та нормативні акти, що регулюють діяльність банків та інших фінансових установ в Україні.
- e) Валютний ринок: Тут можна отримати дані про валютний курс, валютні операції та іншу інформацію, пов'язану з валютним ринком України.

Висновок.

На закінчення ВВП України є ключовим економічним показником, що відображає загальну вартість виробництва товарів і послуг у країні. Незважаючи на існуючі виклики, Україна прагне економічного зростання та розвитку, і реалізація необхідних реформ може сприяти сталому розвитку та збільшенню ВВП у майбутньому. Ресурс НаБу та НБУ є достовірним джерелом інформації, на основі яких можна побудувати точну модель аналізу макроекономічних показників.

1.3. Огляд бібліотек мови Python для побудови моделей машинного навчання

У Python існує багато бібліотек, які надають інструменти для побудови моделей машинного навчання. Деякі з найбільш популярних і широко використовуваних бібліотек включають:

- 1) **scikit-learn**: scikit-learn є однією з найпопулярніших бібліотек машинного навчання у Python. Вона пропонує широкий набір алгоритмів машинного навчання, включаючи класифікацію, регресію, кластеризацію, зниження розмірності та багато іншого. Scikit-learn також надає зручні інструменти для попередньої обробки даних, вибору моделей та оцінки їхньої продуктивності.
- 2) **TensorFlow**: TensorFlow є бібліотекою з відкритим кодом, розробленою Google, для побудови та навчання моделей глибокого навчання. Він надає гнучкий набір інструментів для створення різних архітектур нейронних мереж, включаючи нейронні згорткові мережі (CNN), рекурентні нейронні мережі (RNN) і генеративно-змагальні мережі (GAN).
- 3) **Keras**: Keras є високорівневим інтерфейсом для створення нейронних мереж, що працює поверх бібліотеки TensorFlow. Він має простий та інтуїтивно зрозумілий API, що робить процес створення моделей глибокого навчання більш простим. Keras також підтримує різні типи моделей та шарів нейронних мереж.
- 4) **PyTorch**: PyTorch є ще однією популярною бібліотекою для глибокого навчання. Вона пропонує гнучку та інтуїтивно зрозумілу платформу для побудови та навчання нейронних мереж. PyTorch дозволяє використовувати динамічні обчислення і має велику підтримку для роботи з графічними процесорами (GPU).
- 5) **XGBoost**: XGBoost (Extreme Gradient Boosting) – це бібліотека, призначена для градієнтного бустингу, що пропонує високу продуктивність та точність. Вона ефективно працює з великими наборами даних і має потужні можливості для налаштування параметрів моделі.
- 6) **LightGBM**: LightGBM – це ще одна бібліотека для градієнтного бустингу, розроблена Microsoft. Вона також відома своєю високою продуктивністю та ефективністю при роботі з великими наборами даних. LightGBM пропонує швидку реалізацію алгоритму градієнтного бустингу з використанням особливостей, таких як листове ростове розбиття та гістограмне наближення.
- 7) **Caffe**: Caffe (Convolutional Architecture for Fast Feature Embedding) – це бібліотека глибокого навчання, що спеціалізується на обробці зображень

та відео. Вона надає ефективні інструменти для створення та навчання згорткових нейронних мереж.

- 8) Theano: Theano - це бібліотека для оптимізації та обчислень на GPU. Вона дозволяє створювати та навчати різні типи нейронних мереж, включаючи згорткові нейронні мережі та рекурентні нейронні мережі.
- 9) H2O: H2O - це платформа машинного навчання з відкритим вихідним кодом, що надає простий та зручний інтерфейс для створення моделей машинного навчання. Вона підтримує різні алгоритми, включаючи градієнтний бустинг, випадковий ліс та нейронні мережі.
- 10) MXNet: MXNet - це бібліотека глибокого навчання з відкритим вихідним кодом, розроблена для роботи з великими наборами даних та розподіленими обчисленнями. Вона пропонує гнучкі можливості для створення та навчання моделей глибокого навчання.

Кожна з цих бібліотек має свої особливості та сфери застосування. Вибір бібліотеки залежить від ваших потреб, завдань, даних та рівня досвіду. Важливо вибирати відповідну бібліотеку, яка відповідає вимогам вашого проекту та забезпечує необхідну функціональність та продуктивність. Наприклад бібліотека:

1) scikit-learn.

Бібліотека scikit-learn (також відома як sklearn) є однією з найбільш популярних і широко використовуваних бібліотек машинного навчання мовою Python. Вона надає безліч інструментів та алгоритмів для задач класифікації, регресії, кластеризації, зниження розмірності, обробки даних та вибору моделей.

Основні переваги та особливості scikit-learn:

- Простота використання: scikit-learn розроблений з упором на простоту використання та прямолінійність API. Він забезпечує інтуїтивно зрозумілий та послідовний інтерфейс для створення та навчання моделей машинного навчання.
- Широкий вибір алгоритмів: scikit-learn надає широкий вибір алгоритмів машинного навчання, включаючи лінійну та логістичну регресію, вирішальні дерева, метод опорних векторів (SVM), наївний класифікатор Байєса, випадкові ліси, градієнтний бустинг і багато іншого. Це дозволяє дослідникам та розробникам вибирати найбільш підходящий алгоритм для своїх завдань.
- Підтримка попередньої обробки даних: scikit-learn надає інструменти попередньої обробки даних, такі як масштабування, перетворення категоріальних ознак, обробка пропущених значень і викидів. Це допомагає у підготовці даних перед навчанням моделі.

- Вибір моделей та оцінка продуктивності: scikit-learn пропонує методи для вибору найкращої моделі, використовуючи крос-валідацію та оптимізацію гіперпараметрів. Він також надає метрики для оцінки продуктивності моделей, такі як точність, повнота, F1 міра, середньоквадратична помилка та інші.
- Інтеграція з іншими бібліотеками: scikit-learn добре інтегрується з іншими популярними бібліотеками Python, такими як NumPy, SciPy та Pandas. Це дозволяє використовувати потужні інструменти для роботи з даними та виконання наукових обчислень разом із функціональністю scikit-learn.
- Навчання з учителем та без вчителя: scikit-learn підтримує як завдання навчання з учителем (наприклад, класифікація та регресія), так і завдання без вчителя (наприклад, кластеризація та зниження розмірності). Це робить його корисним для широкого спектру задач машинного навчання.
- Обширна документація та активна спільнота: scikit-learn має докладну документацію та великі приклади використання, що робить його привабливим для початківців та досвідчених користувачів. Крім того, він має активну спільноту користувачів і розробників, які підтримують і розширюють бібліотеку.

Бібліотека scikit-learn є потужним інструментом розробки моделей машинного навчання мовою Python. Вона дозволяє легко та ефективно створювати, навчати та оцінювати моделі для різних завдань.

2) TensorFlow

TensorFlow - це бібліотека з відкритим кодом для чисельних обчислень, що спеціалізується на побудові та навчанні моделей глибокого навчання. Вона була розроблена командою штучного інтелекту Google і є однією з найпопулярніших бібліотек для машинного навчання.

Основні переваги та особливості TensorFlow:

- Графове уявлення: TensorFlow використовує графове уявлення обчислень. Він дозволяє визначити та організувати обчислення у вигляді направленого графа, де вузли представляють операції, а ребра – потоки даних. Це забезпечує гнучкість та ефективність при роботі з великими обчислювальними графами.
- Підтримка глибокого навчання: TensorFlow надає широкий спектр інструментів та можливостей для розробки та навчання моделей глибокого навчання. Він пропонує різні типи шарів та активацій, оптимізатори, функції втрат, а також вбудовану підтримку для автоматичного диференціювання.
- Обчислення на GPU та TPU: TensorFlow забезпечує можливість виконання обчислень на графічних процесорах (GPU) та тензорних

процесорах (TPU), що значно прискорює навчання моделей глибокого навчання та підвищує їх продуктивність.

- Масштабованість: TensorFlow пропонує інструменти для розподіленого навчання та виконання обчислень на кількох вузлах та пристроях. Це дозволяє масштабувати навчання моделей та керувати великими обсягами даних.
- Обширна підтримка та спільнота: TensorFlow має велику документацію, багаті приклади використання та активну спільноту розробників та користувачів. Це забезпечує доступ до різних ресурсів та допомогу при використанні та розробці з використанням TensorFlow.
- Гнучкість та переносимість: TensorFlow може бути використаний на різних платформах та операційних системах. Він також підтримує різні мови програмування, включаючи Python, C++, Java та інші.

TensorFlow надає широкий спектр інструментів та можливостей для розробки та навчання моделей глибокого навчання. Він широко застосовується в академічних та промислових сферах, а також у різних галузях, таких як комп'ютерний зір, обробка природної мови, рекомендаційні системи та інші.

3) Keras

Keras – це високорівневий інтерфейс над бібліотекою TensorFlow (а також над іншими бекендами, такими як Theano та Microsoft Cognitive Toolkit), який спрощує розробку та навчання моделей глибокого навчання. Він був розроблений з акцентом на простоту використання, гнучкість та швидкість розробки.

Основні переваги та особливості Keras:

- Простота використання: Keras пропонує простий та інтуїтивно зрозумілий інтерфейс, що дозволяє швидко створювати моделі глибокого навчання. Він забезпечує високий рівень абстракції та приховує безліч деталей реалізації, що спрощує процес розробки.
- Модульність: Keras пропонує модульну структуру, що дозволяє легко створювати та комбінувати різні шари, активації, оптимізатори та функції втрат. Це дозволяє створювати складні моделі із мінімальним зусиллям.
- Підтримка різних бекендів: Keras може використовувати різні бекенди для виконання обчислень, включаючи TensorFlow, Theano та Microsoft Cognitive Toolkit. Це дає можливість вибрати найбільш підходящий бекенд залежно від потреб та переваг.
- Широка підтримка шарів та архітектур: Keras надає широкий вибір певних шарів, таких як повнозв'язкові шари, згорткові шари, рекурентні шари та інші. Він також підтримує створення власних шарів і моделей з архітектурою користувача.

- Підтримка різних типів завдань: Keras підтримує різні типи завдань машинного навчання, включаючи класифікацію, регресію, сегментацію, генерацію тексту та інші. Він пропонує спеціалізовані шари та функції втрат для кожного типу завдання.
- Вбудована підтримка обробки даних: Keras надає зручні інструменти для попередньої обробки даних, такі як масштабування, перетворення категоріальних ознак і обробка пропущених значень. Це полегшує підготовку даних перед навчанням моделі.
- Обширна документація та активна спільнота: Keras має докладну документацію, багаті приклади використання та активну спільноту розробників та користувачів. Це забезпечує доступ до ресурсів, допомогу та обмін досвідом.

Keras є потужним та гнучким інструментом для розробки моделей глибокого навчання. Він поєднує простоту використання з високою ефективністю та широкими можливостями, що робить його популярним вибором для розробників та дослідників у галузі машинного навчання.

4) PyTorch

PyTorch - це бібліотека глибокого навчання з відкритим кодом, розроблена Facebook. Вона пропонує гнучкі інструменти для розробки та навчання моделей глибокого навчання, а також виконання обчислень на графічних процесорах (GPU).

Основні переваги та особливості PyTorch:

- Динамічний обчислювальний граф: PyTorch використовує динамічний обчислювальний граф, який дозволяє визначати та змінювати граф обчислень під час виконання. Це робить PyTorch гнучким інструментом для дослідження та експериментування з новими моделями та ідеями.
- Простота використання: PyTorch має простий і інтуїтивно зрозумілий API, що робить його доступним для користувачів-початківців. Він пропонує ясний та зрозумілий код, який легко читати та налагоджувати.
- Нативна підтримка GPU: PyTorch забезпечує нативну підтримку виконання обчислень на графічних процесорах (GPU), що дозволяє прискорити навчання моделей глибокого навчання та підвищити їхню продуктивність.
- Гнучкий набір інструментів: PyTorch пропонує широкий спектр інструментів розробки моделей глибокого навчання, включаючи різні типи шарів, активації, оптимізатори, функції втрат та інші компоненти. Це дозволяє створювати та налаштовувати складні моделі з високою гнучкістю.

- Розширюваність: PyTorch володіє відкритою структурою та розширюваним API, що дозволяє розробникам легко розширювати функціональність бібліотеки та вносити власні зміни та покращення.
- Наукова спільнота: PyTorch має активну наукову спільноту, яка активно розвиває та підтримує бібліотеку. Це забезпечує доступ до безлічі ресурсів, включаючи документацію, приклади коду та обговорення, а також можливість отримати підтримку та поради від експертів у галузі глибокого навчання.

PyTorch є потужним інструментом для розробки моделей глибокого навчання, пропонуючи гнучкий та інтуїтивно зрозумілий підхід до розробки та навчання моделей. Він широко використовується як в академічному середовищі, так і в промисловості для вирішення різних завдань машинного навчання та дослідницьких проектів.

5) XGBoost

XGBoost (eXtreme Gradient Boosting) – це бібліотека машинного навчання, яка надає реалізацію градієнтного бустингу на деревах рішень. Вона була розроблена з метою досягнення високої продуктивності, точності та масштабованості при обробці структурованих даних.

Основні переваги та особливості XGBoost:

- Градієнтний бустинг на деревах рішень: XGBoost є реалізацією градієнтного бустингу, методу ансамблювання дерев рішень. Він навчає модель шляхом послідовного додавання слабких дерев, кожне з яких виправляє помилки попередніх дерев, що дозволяє досягти високої точності передбачень.
- Оптимізована реалізація: XGBoost має оптимізовану реалізацію, яка використовує ефективні алгоритми для прискорення навчання моделі та передбачень. Він підтримує паралельне виконання багатопроцесорних системах і може використовувати графічні процесори (GPU) для прискорення обчислень.
- Висока продуктивність: XGBoost призначений для роботи з великими обсягами даних та забезпечує високу продуктивність. Він має механізми для управління пам'яттю, стиснення даних та оптимізації обчислень, що дозволяє ефективно обробляти великі набори даних.
- Регулювання та контроль перенавчання: XGBoost пропонує механізми регуляризації, такі як обмеження глибини дерев, штрафи на складність моделі та відбір ознак. Це допомагає контролювати перенавчання моделі та підвищує її узагальнюючу здатність.
- Підтримка різних завдань машинного навчання: XGBoost підходить для вирішення різних задач машинного навчання, включаючи класифікацію,

регресію та ранжування. Він також надає спеціальні функції втрат та метрики для кожного типу завдання.

- Інтерпретованість та важливість ознак: XGBoost надає можливості для інтерпретації моделі та оцінки важливості ознак. Він дозволяє аналізувати внесок кожної ознаки прийняття рішень моделлю, що допомагає зрозуміти, як модель робить передбачення.

XGBoost є однією з найпопулярніших бібліотек машинного навчання для градієнтного бустингу на деревах рішень. Його висока продуктивність, точність та гнучкість роблять його популярним вибором для вирішення різних завдань прогнозування та класифікації.

6) LightGBM

LightGBM (Light Gradient Boosting Machine) – це бібліотека градієнтного бустингу на деревах рішень, розроблена Microsoft. Вона надає високу продуктивність та ефективність при обробці великих обсягів даних.

Основні переваги та особливості LightGBM:

- Висока швидкість навчання та передбачення: LightGBM використовує алгоритми стиснення даних та ефективну структуру даних для прискорення навчання моделі та передбачень. Він може обробляти великі обсяги даних із високою швидкістю.
- Оптимізована пам'ять та масштабованість: LightGBM використовує оптимізовані алгоритми для управління пам'яттю та оптимізації обчислень. Це дозволяє обробляти великі датасети, які не поміщаються в оперативну пам'ять, та ефективно використовувати ресурси комп'ютера.
- Гнучкість у налаштуванні моделі: LightGBM пропонує широкий набір гіперпараметрів, які можна налаштувати для оптимізації моделі під конкретне завдання. Це включає параметри, пов'язані із структурою дерев, швидкістю навчання, регуляризацією та іншими.
- Підтримка категоріальних ознак: LightGBM має вбудовану підтримку категоріальних ознак, що дозволяє використовувати їх без попереднього кодування. Це полегшує роботу з даними, що містять категоріальні змінні.
- Інтерпретованість моделі: LightGBM надає можливість оцінки важливості ознак та інтерпретації моделі. Ви можете дізнатися, які ознаки мають найбільший вплив на прогнози моделі.
- Підтримка паралельної обробки та GPU: LightGBM може використовувати кілька ядер процесора для паралельного навчання та передбачень. Він також підтримує використання графічних процесорів (GPU) для прискорення обчислень.

LightGBM є потужною бібліотекою градієнтного бустингу, що пропонує високу швидкість та ефективність роботи з великими обсягами даних. Вона широко використовується в задачах машинного навчання, включаючи класифікацію, регресію та ранжування.

7) Caffe

Caffe (Convolutional Architecture for Fast Feature Embedding) – це бібліотека глибокого навчання, розроблена командою дослідників у Berkeley Vision and Learning Center (BVLC). Caffe спочатку був створений для завдань комп'ютерного зору, але він також може бути використаний в інших областях, де потрібна обробка та аналіз даних з використанням глибоких нейронних мереж.

Основні переваги та особливості Caffe:

- Простота використання: Caffe надає простий та інтуїтивно зрозумілий інтерфейс для створення, навчання та використання глибоких нейронних мереж. Він пропонує декларативну мову конфігурації, яка дозволяє визначити архітектуру моделі та параметри навчання.
- Підтримка різних типів архітектур нейронних мереж: Caffe підтримує широкий спектр архітектур нейронних мереж, включаючи нейронні згорткові мережі (CNN), рекурентні нейронні мережі (RNN), пов'язані нейронні мережі та інші. Це дозволяє вирішувати різні завдання, включаючи класифікацію зображень, виявлення об'єктів, сегментацію та інші.
- Ефективність роботи з великими наборами даних: Caffe оптимізовано для обробки великих обсягів даних і може використовувати обчислювальні ресурси, такі як багатоядерні процесори та графічні процесори (GPU), для прискорення навчання та застосування моделі.
- Попередньо навчені моделі: Caffe надає колекцію попередньо навчених моделей, які можна використовувати як відправну точку для своїх завдань. Це дозволяє швидко розпочати роботу та отримати вигоду з вже навчених моделей на великих наборах даних.
- Активна спільнота та розвиток: Caffe має активну спільноту користувачів та розробників, які роблять внесок у її розвиток, пропонують нові функціональні можливості та забезпечують підтримку. Це робить Caffe надійною та підтримуваною бібліотекою для розробки глибоких нейронних мереж.

Caffe є потужною та гнучкою бібліотекою для розробки та навчання глибоких нейронних мереж. Він знаходить застосування в різних галузях, включаючи комп'ютерний зір, розпізнавання мови, обробку природної мови та багато інших.

8) Theano

Theano – це бібліотека для глибокого навчання та наукових обчислень на мові Python. Вона була розроблена з метою спростити та прискорити розробку та виконання обчислювальних завдань, пов'язаних із глибоким навчанням.

Основні переваги та особливості Theano:

- Ефективні обчислення на графах: Theano використовує символічні обчислення для оптимізації обчислювальних завдань. Вона дозволяє описувати математичні операції у термінах символічних змінних та графів, що забезпечує оптимізацію та прискорення обчислень.
- Гнучкість та зручність використання: Theano надає простий та інтуїтивно зрозумілий інтерфейс для створення та оптимізації моделей глибокого навчання. Вона дозволяє визначати шари, функції активації, функції втрат та інші компоненти моделі, що робить процес розробки та експериментування зручнішим.
- Підтримка обчислень GPU: Theano може використовувати графічні процесори (GPU) для прискорення обчислень. Це особливо корисно для обробки великих обсягів даних і виконання складних операцій глибокого навчання.
- Підтримка автоматичного диференціювання: Theano автоматично обчислює градієнти функцій, що полегшує завдання навчання моделей глибокого навчання з використанням методів оптимізації, таких як стохастичний градієнтний спуск.
- Інтеграція з іншими бібліотеками: Theano може взаємодіяти з іншими популярними бібліотеками машинного навчання та наукових обчислень у Python, таких як NumPy та SciPy. Це забезпечує більшу гнучкість та можливості інтеграції з вже існуючими інструментами та кодом.
- Активна спільнота та підтримка: Theano має активну спільноту користувачів та розробників, які роблять внесок у її розвиток, пропонують нові можливості та забезпечують підтримку. Це робить Theano надійною та підтримуваною бібліотекою для глибокого навчання.

Theano надає потужні інструменти для розробки та виконання завдань глибокого навчання та наукових обчислень. Вона широко використовується в академічних та промислових дослідженнях для вирішення різних завдань машинного навчання та аналізу даних.

9) H2O

H2O – це відкрита платформа для машинного навчання та аналізу даних, розроблена компанією H2O.ai. Вона надає простий та потужний інструментарій для виконання широкого спектру задач машинного навчання та аналізу даних.

Основні переваги та особливості H2O:

- Масштабованість та продуктивність: H2O дозволяє обробляти великі обсяги даних та працювати на кластерах з кількома вузлами. Це забезпечує високу продуктивність та можливість масштабування залежно від потреб проекту.
- Широкий набір алгоритмів машинного навчання: H2O надає широкий набір алгоритмів машинного навчання, включаючи лінійні моделі, дерева рішень, довільні ліси, градієнтний бустинг та інші. Це дозволяє вирішувати різні завдання, включаючи класифікацію, регресію, кластеризацію та прогнозування.
- Автоматична обробка ознак: H2O пропонує можливості автоматичної обробки ознак, такі як обробка категоріальних змінних, масштабування та нормалізація даних. Це дозволяє спростити процес підготовки даних та покращити продуктивність моделей.
- Інтерпретованість моделей: H2O надає інструменти для інтерпретації та розуміння моделей. Ви можете отримати інформацію про важливість ознак, інтерпретувати передбачення моделі та проводити аналіз результатів.
- Підтримка різних мов програмування: H2O має інтерфейси для роботи з різними мовами програмування, включаючи Python, R, Java та Scala. Це забезпечує зручність та гнучкість у виборі мови для розробки та інтеграції моделей.
- Інтеграція з екосистемою Big Data: H2O може інтегруватися з екосистемою Big Data, включаючи Apache Hadoop та Apache Spark. Це дозволяє виконувати аналіз даних у розподіленому середовищі та використовувати переваги великих даних.

H2O є потужною та гнучкою платформою для машинного навчання та аналізу даних, яка дозволяє вирішувати складні завдання з використанням різноманітних алгоритмів та обробляти великі обсяги даних.

10) MXNet

MXNet (Мікснет) - це гнучка та ефективна бібліотека глибокого навчання та машинного навчання, розроблена для роботи з великими обсягами даних та розподіленими обчисленнями. Вона надає простий та ефективний спосіб створення, навчання та розгортання моделей глибокого навчання.

Основні переваги та особливості MXNet:

- Гнучкість та масштабованість: MXNet підтримує гнучку модель програмування, що дозволяє створювати моделі глибокого навчання як у статичному, так і динамічному режимі. Вона може працювати з різними

типами моделей, включаючи згорткові нейронні мережі (CNN), рекурентні нейронні мережі (RNN) та комбіновані моделі. MXNet також має високий рівень масштабованості, дозволяючи працювати з великими обсягами даних і використовувати розподілені обчислення.

- Підтримка різних мов програмування: MXNet надає інтерфейси для роботи з різними мовами програмування, включаючи Python, R, Scala, Julia та C++. Це забезпечує зручність та гнучкість у виборі мови для розробки моделей та інтеграції з іншими інструментами та фреймворками.
- Ефективність та продуктивність: MXNet оптимізовано для високої продуктивності та ефективності роботи з графічними процесорами (GPU) та багатоядерними системами. Він використовує розподілені обчислення та оптимізовані алгоритми, що дозволяє прискорити навчання моделей та виконання передбачень.
- Гнучка система графів: MXNet заснований на графовій моделі обчислень, яка представляє модель глибокого навчання як спрямований ациклічний граф операцій. Це дозволяє легко визначити та налаштовувати складні моделі, а також спрощує процес оптимізації та автоматичного диференціювання.
- Багатий набір функцій та інструментів: MXNet пропонує багатий набір функцій та інструментів для розробки моделей глибокого навчання. Він включає різні типи шарів, функцій активації, функцій втрат, оптимізаторів та інші компоненти, які допомагають спростити процес створення і налаштування моделей.
- Активна спільнота та підтримка: MXNet має активну спільноту розробників та користувачів, які активно роблять внесок у її розвиток та забезпечують підтримку. Це забезпечує доступність документації, навчальних матеріалів та відповіді на питання, що допомагає розробникам успішно використовувати бібліотеку.

MXNet є потужним інструментом для розробки моделей глибокого навчання та машинного навчання, що надає гнучкість, ефективність та високу продуктивність при роботі з великими обсягами даних. Він широко використовується як в академічному, так і в промисловому середовищі для вирішення складних завдань аналізу даних та побудови інтелектуальних систем.

Висновок.

Python має достатньо бібліотек, щоб задовольнити потреби користувача. Для вибору набору інструментів має бути чітке розуміння моделі прогнозування, яку бажає створити кодер. З часом створюються все більш потужні бібліотеки для точніших моделей та більших запитів.

РОЗРОБКА МОДЕЛІ ML

2.1. Статистичний аналіз макроекономічних даних

Для вимірювання економічного розвитку використовуються різні показники. Статистичний аналіз дозволяє оцінити напрямок і тенденції розвитку, а також визначити фактори, що на нього впливають. На макрорівні основний показник економічного розвитку країни - валовий внутрішній продукт (ВВП) є важливим показником для оцінки економічного розвитку країни. Він показує ринкову вартість усіх товарів і послуг, вироблених в країні протягом року для споживання, експорту та нагромадження. Він не включає вартість проміжних товарів і послуг, тобто сировини, палива та енергії. Аналіз номінального ВВП України за останні 25 років свідчить про позитивну тенденцію до зростання протягом усього періоду, за винятком 2008-2009 років, року світової економічної кризи. Тенденція до зростання стосується також номінального ВВП на душу населення(Рис. 2.1)[1].

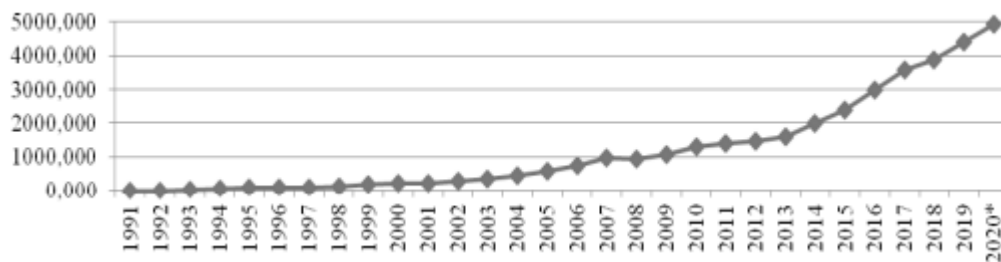


Рис. 2.1. Динаміка номінального ВВП України за 1991-2020 рр., млрд. грн.

У рейтингу країн світу за номінальним ВВП, розрахованим за методикою Світового банку, Україна в 2018 році посідає 60-е місце серед 190 країн світу. Цікаво, що номінальний ВВП в Україні зростає динамічніше, ніж в країнах-сусідах. Однак показник зростання ВВП в 20% – не дає повної картини, оскільки реальний ВВП зріс всього на 2,5%, якщо врахувати інфляцію. За цим показником Україна – беззаперечний лідер і серед країн-сусідів, і на пострадянському просторі – 22,1%. Схожі показники зростання цін в Узбекистані (19%) і Азербайджані (16%). Водночас рівень інфляції в 2018 році в нашій країні був майже в три рази більше, ніж в Білорусі, практично в чотири рази перевищував показник інфляції в Молдові [2].

Номінальні показники не дозволяють проводити порівняння між країнами або порівнювати рівень економічного розвитку однієї і тієї ж країни за різні періоди часу.

Такі порівняння можна робити тільки за допомогою реальних показників, які виражені в незмінних цінах. У реальному ВВП враховується, в якій мірі зростання ВВП визначається реальним зростанням виробництва, а не зростанням цін (Рис. 2.2).

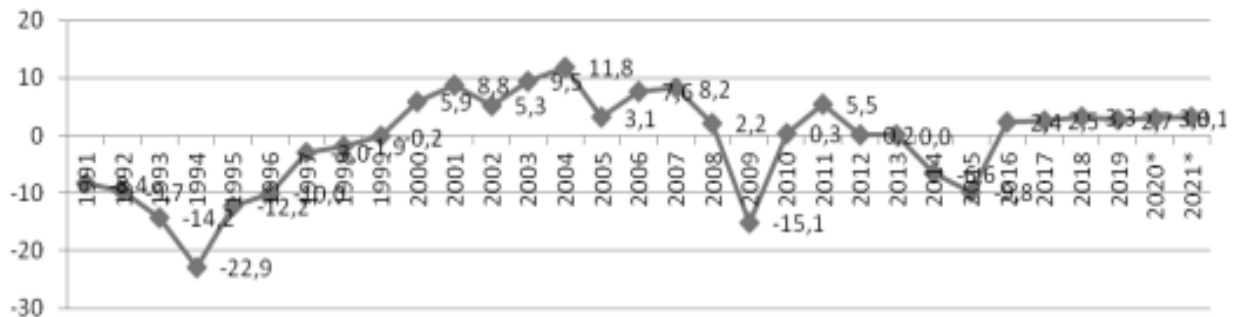


Рис. 2.2. Динаміка зміни реального ВВП України за 1991–2021 рр., %

Якщо розглянути динаміку зміни реального ВВП України з 1991 по 2020 роки, то ситуація докорінно протилежна і тенденція найчастіше має негативний характер. Можна побачити три глибокі кризи, які наша економіка пережила за період незалежності [1].

Перша пов'язана з кардинальною трансформацією економіки, зумовленою переходом до ринкових відносин, лібералізацією зовнішньої торгівлі, розривом економічних зв'язків у межах колишнього СРСР. Після потрясінь відбулося часткове відновлення завдяки зростанню світових цін на сировину, насамперед продукцію металургії. Поступальний розвиток тривав до 2008 року, однак унаслідок порушення основних макроекономічних пропорцій, зокрема різкого зростання рівня споживання, збільшення залежності від зовнішніх ринків, скорочення зовнішнього попиту, дефіциту зовнішніх запозичень та курсової політики, країна виявилась надто вразливою до погіршення зовнішньоекономічної кон'юнктури. Саме тому під час світової фінансової кризи у 2009 році ВВП України знизився на 15,1%, що було найгіршим показником серед країн СНД. Найбільших втрат у цей період зазнали будівництво, фінанси, переробна промисловість і торгівля. Часткове відновлення економіки протягом 2010–2012 рр. не дало змогу досягнути передкризового рівня 2008 р. Третій удар по економіці у 2013 р. пов'язаний з війною на сході України. Відновлювальне зростання почалося в 2016 році, коли ВВП виріс на 2,4% [3, с. 20].

Впродовж 2016-2018 рр. зростання ВВП триває. В розрізі видів діяльності істотний вплив на зростання ВВП в 2018 році мало виробництво аграрної продукції в результаті рекордного врожаю зернових і олійних культур. Завдяки стійкому споживчому попиту істотний вплив на зростання ВВП мали також

сектора торгівлі та послуг. Високими темпами зростало будівництво і фінансовий сектор. Також основними драйверами зростання ВВП в 2018 році були внутрішній і інвестиційний попит. Кінцеві споживчі витрати домогосподарств зросли на 8,9% в річному вимірі, причини: значні темпи зростання заробітних плат, пенсій та переказів трудових мігрантів в Україну. Також зросло валове нагромадження основного капіталу на 14,3% в річному вимірі, зокрема, за рахунок потреби бізнесу в модернізації виробництва та оновлення основних фондів [4, с. 62].

Але знизилися показники переробної промисловості та транспорту. Причиною такого ослаблення були логістичні проблеми і проведення ремонтів на металургійних заводах. Фізичні обсяги експорту товарів і послуг в 2018 році знизилися на 1,6% після зростання на 3,8% в 2017 році. Причиною є погіршення зовнішніх умов протягом року, зокрема, через поширення протекціоністських заходів у світовій торгівлі та зниження врожаю ранніх зернових культур. Істотно сповільнилися також темпи зростання імпорту – з 12,6% у 2017 році до 3,2% в 2018 році, в тому числі через зменшення обсягів закупівлі природного газу на 25% [5, с. 102].

Разом з показником загального ВВП використовують показник ВВП у розрахунку на душу населення. Він не тільки відображає рівень розвитку економіки країни, але й використовується як індикатор рівня добробуту населення. Його також часто використовують під час зіставлення економічного розвитку країни з іншими країнами. Крім цього, для порівняння країн між собою, використовують показники ВВП за паритетом купівельної спроможності населення (ВВП по ПКС) (Рис. 2.3). Цей показник є мірою випуску продукції чи послуг на душу населення в доларах постійної купівельної спроможності. При цьому нівелюється вплив інфляційних процесів. Динаміка ВВП по ПКС значно відрізняється від динаміки ВВП України, але в історичній перспективі так само повторює всі злети і падіння [1; 2].

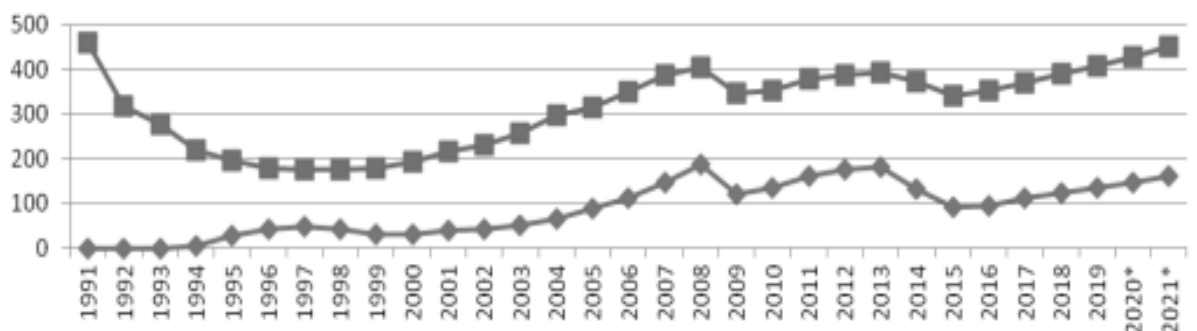


Рис. 2.3. Динаміка ВВП України за паритетом купівельної спроможності відносно інших країн 1991-2021 рр., млрд. дол. Економіка України 49-а в світі

за розміром ВВП по ПКС – 390 млрд. дол. За ВВП по ПКС на душу населення Україна в 2018 році зайняла передостаннє місце серед країн Європи, випереджаючи лише Молдову, і 136-е місце в світі (рис. 2.4) [2].

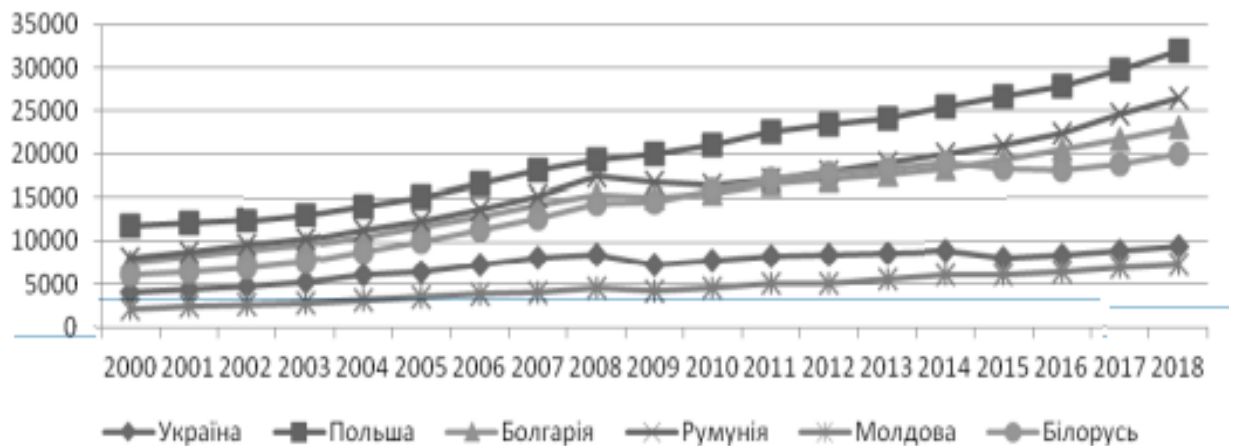


Рис. 2.4. ВВП по ПКС на душу населення 1991-2018 рр., дол.

Якщо поглянути на найближчих сусідів, то зростання України є незначним і відстає від темпів зростання інших країн. Повільні темпи економічного розвитку в Україні мають переважно внутрішню причину - високий рівень корупції, події на сході країни, зростання інфляції, підвищення облікової ставки НБУ, низька інвестиційна привабливість вітчизняних компаній, зростання зовнішнього боргу, зношеність транспортної інфраструктури та слабка економічна динаміка.

У 2019 році економічне зростання в Україні продовжилося, але повільнішими темпами (+2,5%). Фактори, що обмежували зростання, були тимчасовими, наприклад, жорстка монетарна політика, спрямована на утримання інфляції на рівні 5%, а також стриманість фіскальної політики через значні виплати за зовнішнім боргом та сповільнення світової економіки. Всі ці заходи знижують інвестиційну привабливість країни, підвищують облікову ставку, знижують вартість цінних паперів і збільшують податкове навантаження на компанії.

Внутрішній попит, як споживчий, так і інвестиційний, залишиться основним драйвером зростання ВВП в Україні. Також можуть бути меншими темпи зростання в будівельній галузі через зниження динаміки попиту з боку населення. Крім того, тенденція до відтоку робочої сили в інші країни зберігається, що може привести до зниження споживання [6, с. 215].

Також необхідно згадати, що у 2015 році Україна провела реструктуризацію боргів. Одним із пунктів угоди став випуск 20-річних інструментів відновлення вартості або ВВП-варрантів. Суми виплати за ними

залежать від темпів зростання української економіки. При темпах зростання до 3% на рік – виплати є нульовими, 3%–4% – 15% від приросту ВВП, більше 4% – 40% вартості від кожного відсотка зростання.

При цьому виплати мають здійснюватися лише після того, як український ВВП сягне 125,4 млрд. дол. За даними Світового банку, у 2018 році український ВВП якраз сягнув 125,5 млрд. дол. При зростанні у 5% за цими паперами потрібно буде віддавати близько 800 млн. дол. щороку, тобто, потенціал зростання країни весь час буде обмежений борговою складовою.

З іншого боку, зростати меншими темпами, ніж на 7% в рік, Україні немає сенсу. Тільки в цьому випадку можна в межах 10 років вийти на показники 2013 року (більш 180 млрд. дол. або 4,5–5 тис. дол. на душу населення в рік) [7].

Якщо ж наша економіка буде зростати в межах 2–3%, то вже через п'ять років, з урахуванням девальвації національної валюти, виникає ризик падіння за точку неповернення, коли реального ВВП не буде вистачати навіть на просте відтворення зношених основних фондів і базової інфраструктури.

Висновок.

Динамічному зростанню України сприяють як ендегенні, так і екзогенні фактори. Так, українська економіка зростає синхронно зі світовим товарним циклом, а потім різко падає разом з ним. Для того, щоб екзогенний фактор зростання світових цін на сировинні товари трансформувався у 7% річного зростання ВВП, ціни на нафту мають зрости щонайменше до 120-140 доларів США за барель і утримуватися на цьому рівні протягом щонайменше п'яти років. Однак головним індикатором цього може стати закінчення війни.

1.4. Вибір методу машинного навчання

У найзагальнішому випадку існує два методи машинного навчання: "навчання на основі прецедентів" (індуктивне навчання) та "дедуктивне навчання". Оскільки останнє зазвичай належить до сфери експертних систем, "машинне навчання" і "навчання на основі прецедентів" можна вважати синонімами. Оскільки останнє прийнято відносити до області експертних систем, то терміни «машинне навчання» і «навчання по прецедентах» можна вважати синонімами.

Навчання по прецедентах, в свою чергу, поділяють на три основних методи:

- Контрольоване навчання, або навчання з учителем (supervised learning);
- Неконтрольоване навчання (unsupervised learning), або навчання без учителя;
- Навчання з підкріпленням (reinforcement learning) ;

Останні роки також почали все більше розвиватися інші технології, такі як:

- 1) Глибоке навчання (Deep Learning) - є підмножиною машинного навчання і засновано на штучних нейронних мережах. Воно імітує роботу людського мозку, що складається з безлічі зв'язків нейронів. Глибокі нейронні мережі складаються з безлічі шарів, кожен з яких виконує певні обчислення і передає результати наступного шару.
- 2) Online-навчання (Online Learning) - являє собою метод навчання моделі даних, які проходять потоком або за їх поступом, у відмінності від традиційного пакетного навчання. Це особливо корисно в ситуаціях, де дані змінюються з часом або нові дані доступні поступово. Онлайн-навчання дозволяє моделі швидко адаптуватися до змін і постійно покращуватися з протягом часу.
- 3) Мета-навчання (Meta Learning) - це область машинного навчання, яка фокусується на навчанні моделей навчатися самим собою. Вона досліджує стратегії та алгоритми, які дозволяють моделям ефективно адаптуватися до нових завдань і навчатися на невеликій кількості даних. Мета-навчання може бути особливо корисним у випадках, коли доступна обмежена кількість розмічених даних або коли потрібне швидке навчання для нових завдань.
- 4) Навчання на великих даних (Big Data Learning) - відноситься до розробки алгоритмів і техніки машинного навчання, які здатні працювати з величезними обсягами даних. У сучасному світі обсяги даних постійно розходяться, а обробка та аналіз таких даних вимагає спеціальних підходів та інструментів. Технології машинного навчання на великих даних включають розподілене навчання, паралельні обчислення та використання спеціалізованих платформ та інфраструктуру для обробки даних.
- 5) Автоматизоване машинне навчання (Automated Machine Learning) - це область, яка прагне автоматизувати процес розробки моделей машинного навчання. Вона пропонує методи та інструменти, які автоматично вибирають і налаштовують моделі, оптимізують гіперпараметри і виконують інші завдання, які раніше вимагали більшої кількості експертної роботи. Автоматизоване машинне навчання може значно спростити і прискорити процес розробки моделей, дозволяючи навіть неспеціалістам в області машинного навчання використовувати свої переваги.

Це лише деякі з безлічі технологій машинного навчання, які існують сьогодні. Усі ці технології мають свої переваги та обмеження та можуть бути застосовані в різних сценаріях. З постійним розвитком області машинного навчання, очікується поява нових і поліпшення існуючих, що відкриває ще

більший потенціал для застосування в різних сферах і задачах.

Контрольоване навчання.

У керованому алгоритмі ML результат вже відомий. Існує відображення вхідних даних з вихідними. Отже, для створення моделі машина подається з великою кількістю навчальних вхідних даних (з відомими вхідними та відповідними вихідними даними)(Рис. 2.5).



Рис. 2.5. Створення моделі машинного навчання.

Алгоритми навчання під наглядом є найпоширенішим типом алгоритму машинного навчання. Ці алгоритми використовують позначені навчальні дані для вивчення функції, яка відображає вхідні змінні (x) та вихідні змінні (y).

Цей алгоритм часто використовується для прогнозування майбутніх результатів і класифікації даних.

Кілька прикладів алгоритмів керованого навчання — це лінійна регресія, дерева рішень, випадкові ліси та опорні векторні машини.

Дані тренінгу допомагають досягти рівня точності для створеної моделі даних. Вбудована модель тепер готова до отримання нових вхідних даних та прогнозування результатів(Рис.2.6).

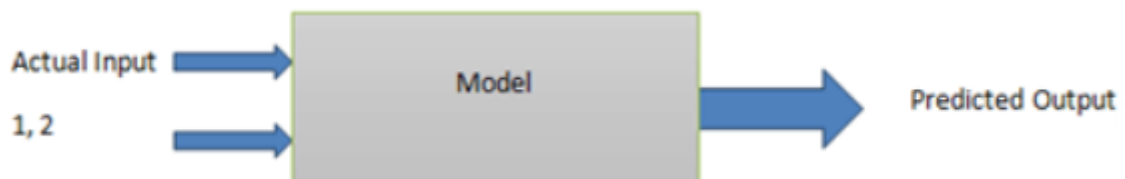


Рис. 2.6. Використання побудованої моделі для прогнозування результатів.

Методи контрольованого навчання використовуються для таких додатків, як визначення рівня фінансового ризику, пов'язаного з окремими особами та компаніями, на основі попередньої інформації про їх діяльність. Крім того, використовуючи аналіз минулих моделей поведінки, вони можуть надати корисну інформацію про те, як клієнти, швидше за все, будуть вести себе і які їхні переваги.

Неконтрольоване навчання.

Алгоритми для неконтрольованого машинного навчання використовуються для пошуку шаблонів і груп даних без міток.

Цей алгоритм зазвичай використовується для кластеризації, зменшення

розмірів і виявлення аномалій.

Алгоритми неконтрольованого навчання включають кластеризацію k-середніх, ієрархічну кластеризацію та виявлення аномалій.

Оскільки немає відомих вихідних значень між входами і виходами, які можна було б використати для побудови логічних моделей, використовується кілька методів для вилучення правил, шаблонів і груп подібних типів даних. Ці групи допомагають кінцевому користувачеві краще зрозуміти дані і знайти значущі результати.

Вхідні дані не мають такої ж структури, як навчальні дані (у керованому навчанні). Вони можуть містити викиди, зашумлені дані тощо. Ці вхідні дані подаються в систему разом. Під час навчання моделі вхідні дані організовуються у кластери(Рис. 2.7).

Проходячи процес виявлення закономірностей у даних, модель сама коригує свої параметри, отже, її також називають самоорганізуючою. Кластери будуть сформовані шляхом з'ясування подібності між вхідними даними.

Основна перевага неконтрольованого навчання полягає в тому, що воно не потребує міток або великої кількості навчальних даних, яких вимагає навчання під контролем. Однак результати можуть бути складними для інтерпретації та схильними до переобладнання.

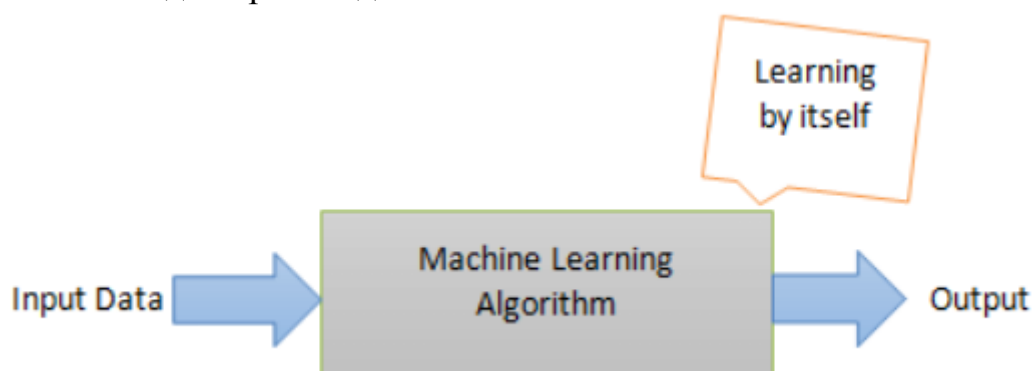


Рис. 2.7. Алгоритм неконтрольованого навчання

Навчання з підкріпленням.

При цьому типі навчання алгоритм навчається за допомогою механізму зворотного зв'язку та минулого досвіду. Завжди бажано, щоб кожен крок в алгоритмі робився для досягнення мети.

Таке навчання є окремим випадком керованого навчання, в якому вчитель є "середовищем". Машина (яку в цьому контексті часто називають "агентом") не має попередніх знань про середовище, але може виконувати довільні дії в середовищі. Середовище реагує на ці дії, тим самим надаючи агенту дані для реакції та навчання. Фактично, агент і середовище утворюють систему

зворотного зв'язку.

Отже, щоразу, коли потрібно зробити наступний крок, він отримує зворотний зв'язок із попереднього кроку, разом із досвідом досвіду, щоб передбачити, що може бути наступним найкращим кроком. Цей процес також називають процесом спроб і помилок для досягнення мети.

Навчання з підкріпленням - це довготривалий ітеративний процес. Чим більше зворотного зв'язку, тим точнішою стає система. Базове навчання з підкріпленням також називають Марковським прийняттям рішень.

Механізм зворотного зв'язку підкріпленого навчання (Рис. 2.8).

- 1) Вхідні дані спостерігає агент, який є елементом ШІ.
- 2) Цей агент ШІ діє на навколишнє середовище відповідно до прийнятого рішення.
- 3) Реакція середовища надсилається ШІ у вигляді винагороди у відповідь.
- 4) Також зберігаються стан і дії, що здійснюються на навколишнє середовище.



Рис.2.8. Механізм зворотного зв'язку.

Висновок.

Технології машинного навчання являють собою широкий спектр методів і алгоритмів, які забезпечують можливість комп'ютерним системам навчатися і робити прогнози на основі даних. Вони знайшли застосування в багатьох сферах і продовжують розвиватися з появою нових методів і підходів. Вибір конкретної технології залежить від типу задач, доступних даних, необхідної точності та інших факторів. Проте всі ці технології мають загальну ціль - отримання корисної інформації та знань із даних для прийняття рішень, оптимізації процесів та досягнення нових висот у штучному інтелекті та автоматизації. Для побудови моделі машинного навчання на основі існуючих даних краще всього обрати метод контролювано навчання, або навчання з

учителем (supervised learning);

2.3. Навчання моделі на навчальній вибірці

Модель прогнозування часових рядів ARIMA (в даному випадку використовується SARIMAX для поквартально прогнозування даних), потребує самих даних. Для початку формуємо файл .csv з даними в форматі часових рядів. В самому файлі в двох колонках вказуємо наші дані, це будуть дата та саме ВВП (в доларах.)(Рис.2.9). Як можемо спостерігати дати розписані щоквартально, що дасть змогу моделі навчитися на більшій кількості даних та точніше спрогнозувати дані.

A	B	C
DataK	N	GDP\$
4/1/2002	1	8292215479
7/1/2002	2	9404931692
10/1/2002	3	12208837602
1/1/2003	4	12469807216
4/1/2003	5	9857156247
7/1/2003	6	11400765077
10/1/2003	7	14219638001
1/1/2004	8	14658351308
4/1/2004	9	12568442385
7/1/2004	10	14773807958
10/1/2004	11	18728451118
1/1/2005	12	18871338636
4/1/2005	13	15962354189
7/1/2005	14	19287636004
10/1/2005	15	23642178218
1/1/2006	16	24466336634
4/1/2006	17	21059009901
7/1/2006	18	25013663366
10/1/2006	19	30179405941
1/1/2007	20	31500990099
4/1/2007	21	27612673267
7/1/2007	22	33043366337
10/1/2007	23	39511881188
1/1/2008	24	42551089109
4/1/2008	25	37912673267
7/1/2008	26	48677638227
10/1/2008	27	56871219914
1/1/2009	28	31702987013
4/1/2009	29	24549090909

Рис. 2.9. Дані ВВП України з 2002-2020 роки

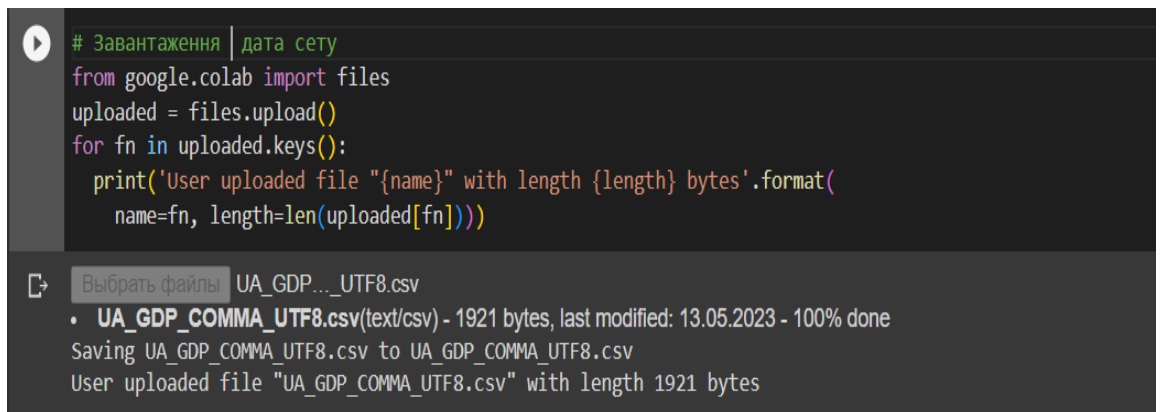
Наступним кроком буде встановлення всіх необхідних бібліотек(Рис. 2.10).

```
import pandas as pd
from google.colab import files
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
from pandas import date_range, to_datetime
import numpy as np
import statsmodels.api as sm
from statsmodels.tsa.arima_model import ARIMA
from statsmodels.tsa.seasonal import seasonal_decompose
import matplotlib.pyplot as plt
from flask import Flask, send_file
```

Рис. 2.10. Імпорт необхідних бібліотек

Імпортовані бібліотеки допоможуть більш точно візуалізувати дані та довести їх до кінцевого вигляду.

Наступним кроком є завантаження заздалегідь підготовленого дата сету(Рис. 2.11).



```
# Завантаження | дата сету
from google.colab import files
uploaded = files.upload()
for fn in uploaded.keys():
    print('User uploaded file "{name}" with length {length} bytes'.format(
        name=fn, length=len(uploaded[fn])))
```

Выбрать файлы UA_GDP..._UTF8.csv

- UA_GDP_COMMA_UTF8.csv(text/csv) - 1921 bytes, last modified: 13.05.2023 - 100% done

Saving UA_GDP_COMMA_UTF8.csv to UA_GDP_COMMA_UTF8.csv
User uploaded file "UA_GDP_COMMA_UTF8.csv" with length 1921 bytes

Рис.2.11. Завантаження дата сету.

Завантаживши дані потрібно оцінити стаціонарність ряду та побудувати графіки, якщо ряд не є стаціонарним треба провести диференціювання ряду (Рис. 2.12).

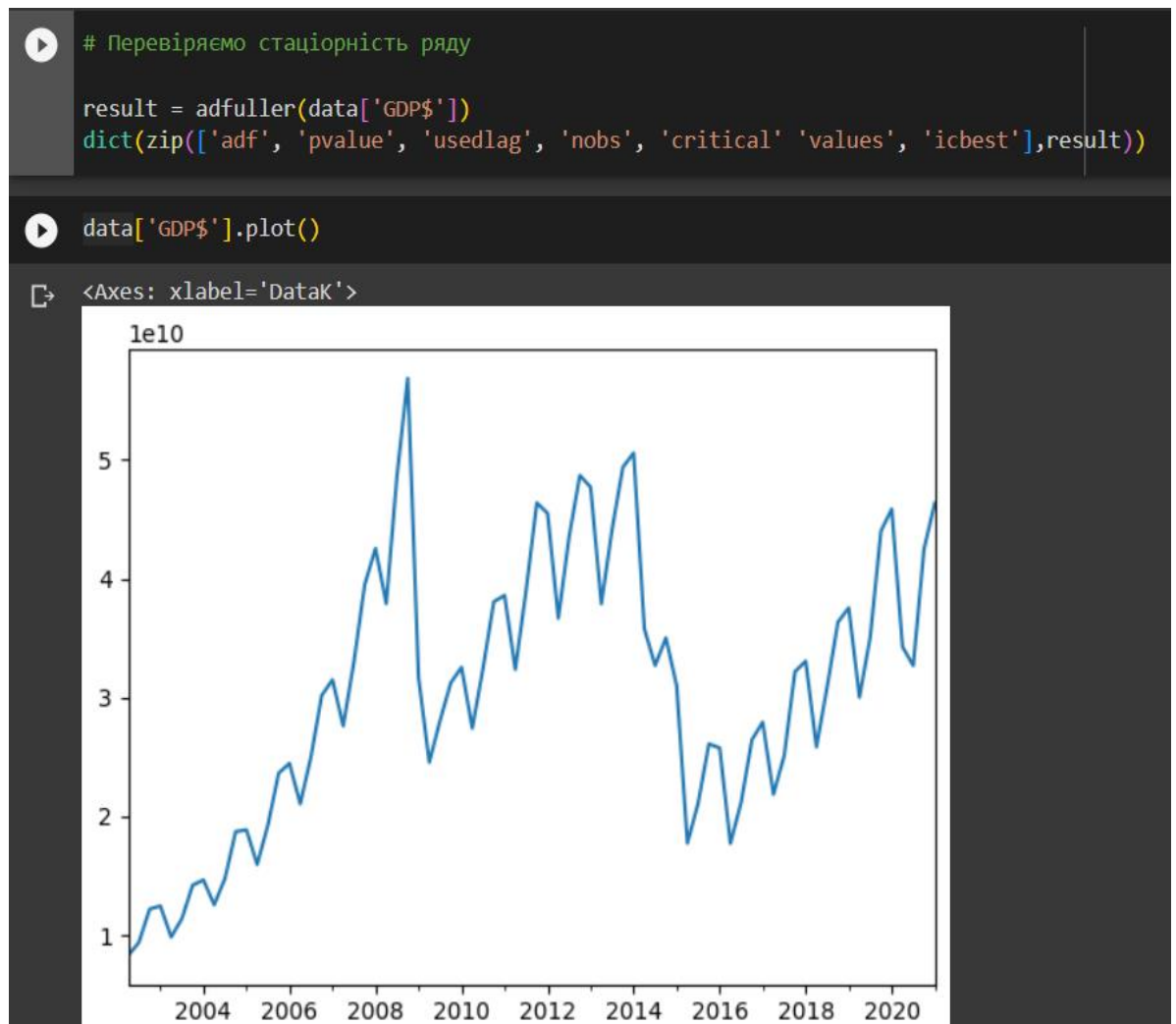


Рис. 2.12. Перевірка стаціонарності ряду та побудова графіку стаціонарності

Стаціонарність учасових рядів означає, що статистичні властивості (такі як середнє значення, дисперсія і автокореляція) не змінюються з часом. Тобто стаціонарний ряд є стабільним та передбачуваним, його поведінка не залежить від конкретного часового періоду.

Стаціонарність важлива при аналізі часових рядів та побудові моделей прогнозування, оскільки багато статистичних методів і моделей припускають стаціонарність ряду. Якщо ряд не є стаціонарним, то це може призвести до невірних висновків та некоректних прогнозів.

Стаціонарний ряд зазвичай має такі властивості:

- Константне середнє значення (очікуване значення ряду не змінюється з часом).
- Константна дисперсія (розкид значень ряду не змінюється з часом).
- Константна автокореляція (кореляція між значеннями ряду у різних часових відрізках не змінюється з часом).

Для перевірки стаціонарності ряду можна використовувати різні статистичні тести, такі як тест Дікі-Фуллера (ADF тест), тест KPSS та інші. Також стаціонарність може бути досягнута шляхом застосування перетвореного ряду, наприклад, диференціювання або сезонної декомпозиції.

Диференціювання часових рядів використовується для приведення нестаціонарного ряду до стаціонарного стану. Диференціювання дозволяє видалити тренд та сезонність, що спрощує аналіз та прогнозування часових рядів.

Основна мета диференціювання – усунути зміну середнього значення ряду та привести його до постійного рівня. Для цього береться різниця між поточним та попереднім значеннями ряду. Якщо ряд містить тренд, диференціювання дозволяє його видалити і перетворити ряд на стаціонарний.

Диференціювання може бути застосоване один або кілька разів, залежно від ступеня нестаціонарності. Часто використовуються перші різниці (різниця між сусідніми значеннями) або сезонні різниці (різниця між значеннями на тому самому часі в різні роки).

Переваги диференціювання включають:

- Усунення тренду та сезонності для покращення стаціонарності ряду.
- Простіший аналіз та прогнозування стаціонарного ряду.
- Найкраща інтерпретованість статистичних властивостей ряду після диференціювання.

Однак слід пам'ятати, що диференціювання може призвести до втрати інформації, особливо при використанні високих порядків диференціювання. Тому важливо збалансувати ступінь диференціювання задля досягнення стаціонарності без втрати істотних тимчасових властивостей ряду.

Отримавши результат проводимо операцію диференціювання (Рис. 2.13).

```
# Перетворюємо дані в стаціонарні, щоб оцінити параметри d і D, які ми будемо використовувати в моделі. Для цього проводимо операцію диференціювання
data['1difference']=data['GDP$']-data['GDP$'].shift(1)
data['1difference'].plot()
```

Рис. 2.13. Операція диференціювання

Далі знову перевіряємо ряд на стаціонарність, якщо дані стаціонарні створюємо графіки автокореляції та часткової автокореляції (Рис. 2.14).

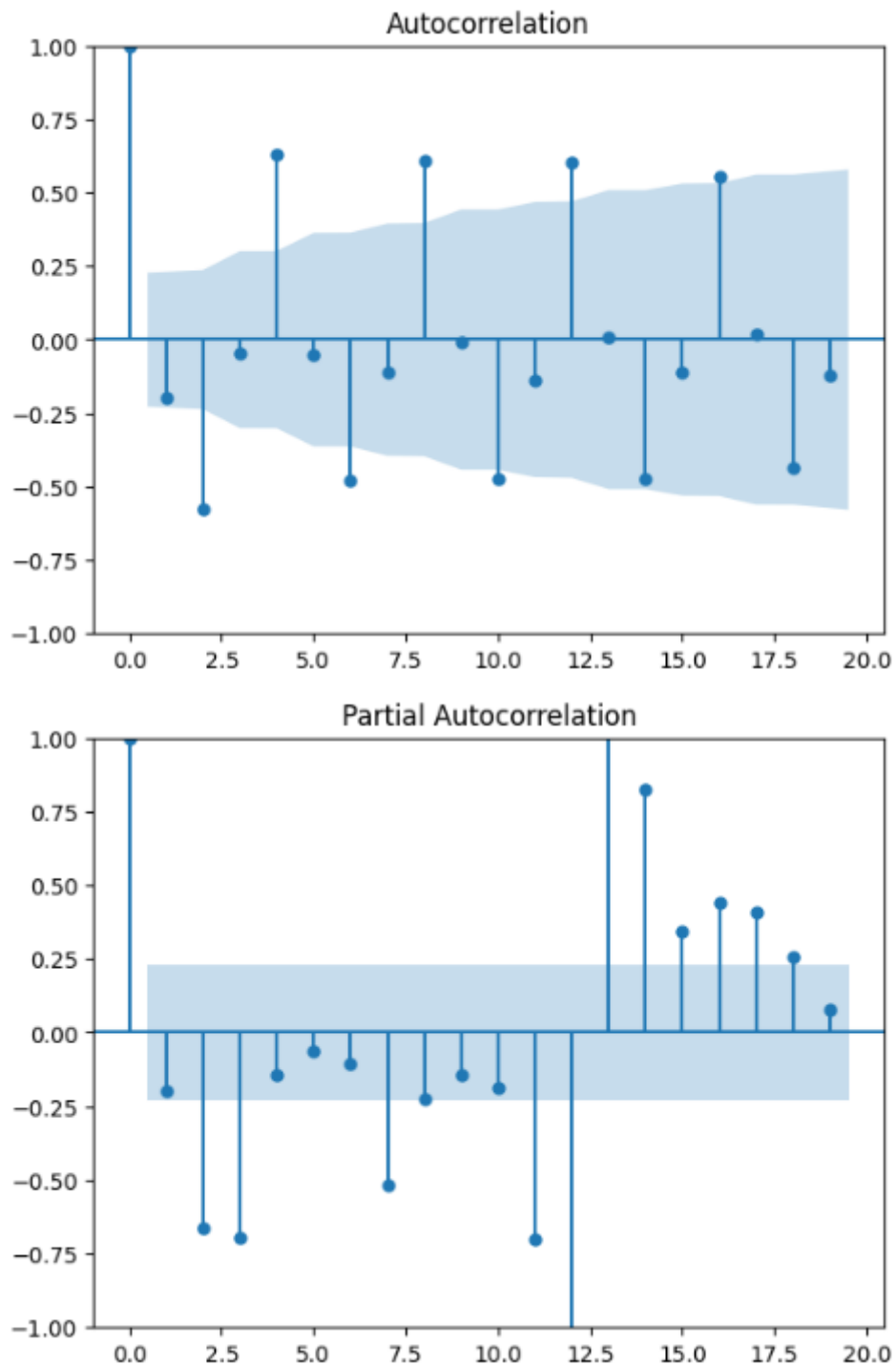


Рис. 2.14. Створення графіків автокореляції

Автокореляція (autocorrelation) в контексті часових рядів є статистичним заходом, який вимірює ступінь залежності між значеннями ряду та його лагованими (відстають) версіями.

Формально, автокореляція виражає кореляцію між значенням низки момент часу t і його значенням момент часу $t-k$, де k - це лаг (відставання). Автокореляція може набувати значень від -1 до 1, де:

- Значення 1 означає позитивну лінійну залежність між значеннями ряду та його лагованими версіями на даному лазі.
- Значення -1 означає негативну лінійну залежність між значеннями ряду та його лагованими версіями на даному лазі.
- Значення 0 означає відсутність лінійної залежності між значеннями ряду та його лагованими версіями на даному лазі.

Автокореляційна функція (ACF) є графічним представленням автокореляції для різних лагів. Графік ACF дозволяє досліджувати залежність між значеннями ряду на різних лагах та допомагає визначити наявність сезонності або інших тимчасових патернів у ряді.

Автокореляція має важливе значення при аналізі та моделюванні часових рядів. Наявність автокореляції може вказувати на наявність залежності між значеннями ряду на різних часових відрізках, що можна використовувати для прогнозування майбутніх значень ряду. При моделюванні часових рядів автокореляція може бути використана для вибору оптимальних параметрів моделі, таких як порядки авторегресії (AR) та ковзного середнього (MA) у моделі ARMA або SARIMA.

Після отримання даних про автокореляцію, можна створювати модель ARIMA для посезонного передбачення даних на 12 місяців.(Рис. 2.15).

```
[327] # Створюємо модель ARIMA з посезонним передбаченням даних
      model = sm.tsa.statespace.SARIMAX(data['GDP$'], order=(1, 2, 2),seasonal_order=(1, 2, 3, 4)).fit()
```

Рис. 2.15. Створення моделі SARIMAX

Висновок.

SARIMA: Ця модель є розширенням моделі ARIMA для врахування сезонності. Вона враховує як сезонні, і несезонні компоненти часового ряду. SARIMA модель має параметри, такі як порядки авторегресії (p), різниці (d) і ковзного середнього (q), а також порядки сезонної авторегресії (P), сезонної різниці (D) і сезонного ковзного середнього (Q). Модель SARIMA може бути ефективною для прогнозування квартальних часових рядів із явною сезонністю.

2.4. Тестування моделі та оцінка якості моделі

Зроблена модель має прогнозувати ВВП України на 2021 рік(Рис.2.16), на основі даних про ВВП з 2002-2020 рр., така модель має не бути дуже чіткою, адже тут не враховуються додаткові фактори такі, як криза війна і т.д.

```
print(predicted_values)
```

2021-04-01	3.728712e+10
2021-07-01	3.881798e+10
2021-10-01	4.825691e+10
2022-01-01	5.161295e+10

Рис. 2.16. Передбачені значення

Оцінити дану модель можна за допомогою метрик MAE, MSE, R^2 (Рис. 2.17).

```
# Вивід метрик оцінки
print('MAE:', mae)
print('MSE:', mse)
print('R^2:', r2)
```

```
MAE: 6619216290.661238
MSE: 6.577505085947504e+19
R^2: 0.47091531198258474
```

Рис. 2.17. Метрики оцінки

MAE (Mean Absolute Error) означає середню абсолютну помилку. Це метрика, яка вимірює середнє абсолютне відхилення між прогнозованими значеннями та фактичними значеннями. В даному випадку значення MAE дорівнює 6,619,216,290.66.

MSE (Mean Squared Error) означає середньоквадратичну помилку. Це метрика, яка вимірює середнє значення квадрата відхилення між прогнозованими значеннями та фактичними значеннями. У даному випадку значення MSE дорівнює $6.577505085947504e+19$, що представлено в науковій нотації.

R^2 (Coefficient of Determination) означає коефіцієнт детермінації. Він є мірою, яка оцінює пояснену варіацію моделі щодо загальної варіації даних. Значення R^2 знаходиться в діапазоні від 0 до 1, де 1 означає ідеальне відповідність моделі даним. У цьому випадку значення R^2 дорівнює 0.47091531198258474, що вказує на те, що модель пояснює приблизно 47.09% варіації даних.

Ці метрики використовуються з метою оцінки точності прогнозних моделей. Найменші значення MAE та MSE і ближче до 1 значення R^2 вказують на більш точну модель прогнозування.

Дані метрики як і графіки(Рис. 2.18) значень показують, що дана модель може пояснити лише 47.09% даних.

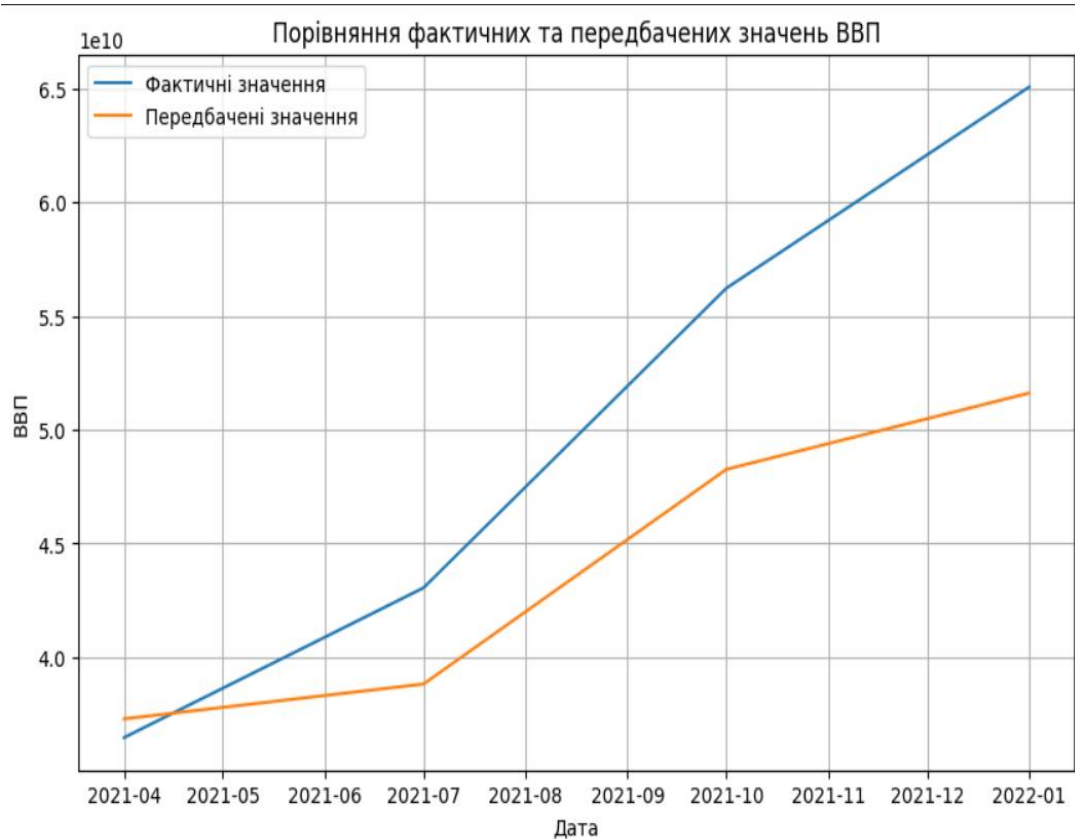


Рис. 2.18. Порівняння значень сезонності протягом 12 місяців

Висновок.

Загалом модель машинного навчання може значно покращити аналіз та прогнозування макроекономічних показників, створена модель змогла спрогнозувати ВВП України поквартально на 2021 рік з точністю 47.09% за метрикою R^2 (Coefficient of Determination) коефіцієнт детермінації, коефіцієнт кореляції складає 0.68, що є доволі гарним показником. В економіці точною моделлю можна назвати від 50-60%. Тож дана модель може бути доопрацьована, але на даній стадії є досить точною. Порівнюючи передбачені та наявні дані, а саме суму ВВП України на 2021 рік (спрогнозовані дані) – 175 974 960 000 \$ і фактичні -200 801 102 885\$, отримуємо похибку прогнозу на рік 12.53%.

2. ПЕРСПЕКТИВИ РОЗВИТКУ МОДЕЛІ

3.1.Рекомендації щодо подальшого застосування моделі

Основною рекомендацією є покращення моделі. Для покращення моделі SARIMA можна спробувати такі підходи:

- 1) Вибір оптимальних параметрів моделі: Параметри моделі SARIMA, такі як порядки авторегресії (p), порядки різниць (d) та порядки ковзного середнього (q), можуть бути налаштовані для досягнення кращої продуктивності моделі. Можна використовувати методи вибору параметрів, такі як пошук сітків або метод найменших квадратів, щоб визначити оптимальні значення параметрів моделі.
- 2) Додавання сезонної складової: Якщо в даних є сезонність, можна розглянути модель SARIMA з сезонними параметрами (P, D, Q, s), де P, D, Q - порядки сезонних компонентів, а s - період сезонності. Додавання сезонної складової може покращити здатність моделі вловлювати сезонні патерни та покращити точність прогнозів.
- 3) Облік зовнішніх факторів: Якщо у вас є додаткові дані про зовнішні фактори, які можуть впливати на тимчасовий ряд, можна розглянути включення цих факторів у модель SARIMA. Наприклад, економічні дані, свята, події тощо. Модель SARIMAX дозволяє врахувати такі зовнішні фактори прогнозування.
- 4) Аналіз та усунення викидів: Аналіз даних на наявність викидів та аномалій, а також їх видалення чи корекція може допомогти покращити модель. Викиди можуть спотворити навчання моделі та призвести до неточних прогнозів.
- 5) Використання альтернативних моделей: Якщо модель SARIMA не дає достатньо хороших результатів, можна розглянути альтернативні моделі прогнозування часових рядів, такі як ARIMA, VAR, LSTM та інші. Різні моделі можуть краще підходити для конкретних типів даних та патернів.
- 6) Оцінка моделі та періодичне оновлення: Після побудови моделі SARIMA та отримання прогнозів важливо оцінити її продуктивність та провести аналіз помилок. Якщо модель не є достатньо точною, можна спробувати внести коригування або оновити модель з новими даними.
- 7) Використання крос-валідації: Крос-валідація дозволяє оцінити продуктивність моделі на різних часових періодах та допомагає уникнути перенавчання. Можна розділити дані на навчальний та тестовий набори, застосувати модель SARIMA до навчального набору, а потім оцінити її продуктивність на тестовому наборі. Це допоможе отримати більш об'єктивну оцінку точності моделі.

- 8) Використання ансамблевих моделей: Замість використання окремої моделі SARIMA можна спробувати застосувати ансамбль моделей, таких як модель середніх або модель Бегінга. Це дозволяє комбінувати прогнози кількох моделей SARIMA для отримання більш точних та надійних прогнозів.
- 9) Оновлення моделі: Тимчасові ряди можуть змінюватися з часом. Тому важливо регулярно оновлювати модель, перенавчати її на нових даних та проводити аналіз помилок. Якщо з'являються нові тренди, сезонні патерни або інші зміни даних, модель може вимагати переналаштування або зміни параметрів.

Після доопрацювання моделі, вона може використовуватися в таких областях:

- 1) Фінансові ринки: Прогнозування цін акцій, курсів валют, процентних ставок та інших фінансових показників з урахуванням економічних чинників та сезонності.
- 2) Економічний аналіз: Прогнозування ВВП, інфляції, безробіття та інших економічних показників з урахуванням макроекономічних чинників.
- 3) Продаж та попит: Прогнозування продажів, споживчого попиту та прогнозів ринкового попиту з урахуванням сезонності, маркетингових акцій та інших факторів.
- 4) Демографічний аналіз: Прогнозування населення, міграції, народжуваності та смертності з урахуванням демографічних трендів.
- 5) Планування виробництва та операцій: Прогнозування попиту на товари та послуги для планування виробництва, управління запасами та оптимізації операцій.
- 6) Кліматичні та метеорологічні прогнози: Прогнозування погодних умов, температури, опадів та інших кліматичних показників з урахуванням сезонності та метеорологічних факторів.
- 7) Залізничні та транспортні системи: Прогнозування пасажиропотоку, вантажних перевезень, затримок та інших показників у залізничних та транспортних системах.

Висновок.

Дана модель потребує доопрацювання. Загалом, модель SARIMAX може бути корисною для прогнозування часових рядів у будь-якій галузі, де є вплив зовнішніх факторів та сезонності.

ВИСНОВКИ

В даній роботі було вирішено науково-практичну задачу аналізу та прогнозування макроекономічних показників України, посезонно на 12 місяців, методами машинного навчання.

- 1) Прогнозування: Моделі машинного навчання можна використовувати для прогнозування макроекономічних показників, як-от ВВП, інфляція, безробіття тощо. Це дозволяє покращити планування та прийняття рішень в економіці, а також допомагає передбачити майбутні тренди та зміни.
- 2) Аналіз впливу факторів: Моделі машинного навчання дозволяють аналізувати вплив різних факторів на макроекономічні показники. Це дозволяє виявляти ключові драйвери економічного зростання чи спаду, ідентифікувати важливі чинники, що впливають на інфляцію чи безробіття, і оцінювати їх вплив.
- 3) Виявлення неявних зв'язків: Моделі машинного навчання можуть допомогти виявити неявні зв'язки та закономірності між різними макроекономічними показниками. Наприклад, вони можуть допомогти виявити взаємозв'язок між інфляцією та ВВП, вплив зовнішніх факторів на економічну активність тощо.
- 4) Детекція аномалій та ризиків: Моделі машинного навчання можуть допомогти у виявленні аномалій та ризиків у макроекономічних даних. Вони можуть допомогти у виявленні економічних криз, фінансових бульбашок, зрушень в економічних трендах і т.д., що дозволяє вживати запобіжних заходів та мінімізувати негативні наслідки.

Загалом модель машинного навчання може значно покращити аналіз та прогнозування макроекономічних показників, створена модель змогла спрогнозувати ВВП України поквартально на 2021 рік з точністю 47.09% за метрикою R^2 (Coefficient of Determination) коефіцієнт детермінації. В економіці точною моделлю можна назвати від 50-60%. Тож дана модель потребує доопрацювання, але на даній стадії є досить точною. Також коефіцієнт кореляції складає 0.68, що є дуже доволі гарним показником. Порівнюючи передбачені та наявні дані. А саме сума ВВП України на 2021 рік спрогнозовані дані) – 17 597 960 000 \$ Фактичні -200801102885 \$ отримуємо процент похибки 12.53%.

СПИСКИ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Державна служба статистики України : сайт. URL: <http://www.ukrstat.gov.ua> (дата звернення: 10.03.2023).
2. Світовий банк : сайт. URL: <https://www.worldbank.org/uk/country/ukraine> (дата звернення: 12.03.2023).
3. Зверяков М. І. Теоретична парадигма сталого розвитку та українські реалії. Економіка України. 2018. №10. С. 10–31.
4. Пелех О. П. Аналіз динаміки ВВП України в контексті Європейського розвитку. Науковий вісник Ужгородського національного університету. 2019. №2. Ч. 2. С. 60–65.
5. Інтеграція України в глобальний соціально-економічний простір : монографія / М. І. Зверяков, А. А. Гриценко, О. Г. Білорус та ін. Київ, 2019. 287 с.
6. Національні та глобальні детермінанти економічного зростання України : наукова доповідь / За ред. І. Бобух. Київ : НАН України, ДУ «Ін-т екон. та прогнозув. НАН України», 2018. 390 с.
7. У Кабміні назвали п'ять чинників для щорічного росту ВВП на 5–7% / Укрінформ: мультимедійна платформа іномовлення України. 2019. URL: <https://www.ukrinform.ua> (дата звернення: 08.04.2023).
8. scikit-learn (офіційна документація). URL: <https://scikit-learn.org/stable/> (дата звернення: 08.05.2023).
9. Аурелієн Герон . "Машинне навчання з використанням Python та Scikit-Learn" Аурелієн Герон (перекладена версія книги "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow").
10. Петра Флоринський . "Машинне навчання".
11. Олександра Голдберга, Олександра Левіна та Ігоря Яремчука. "Задачі машинного навчання".
12. Sebastian Raschka и Vahid Mirjalili . "Python Machine Learning".
13. "Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) Plots" от Machine Learning Mastery URL: <https://machinelearningmastery.com/gentle-introduction-autocorrelation-partial-autocorrelation/> (дата звернення: 17.05.2023).
14. ARIMA. URL: <https://www.statsmodels.org/stable/generated/statsmodels.tsa.arima.model.ARIMA.html> (дата звернення: 10.05.2023).

15.SARIMAX.URL:

<https://www.statsmodels.org/stable/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html>

(дата звернення: 12.05.2023).

16."Forecasting: Principles and Practice" авторів Rob J. Hyndman и George Athanasopoulos. URL: <https://otexts.com/fpp2/seasonal-arima.html>

(дата звернення: 21.05.2023).

17.Wes McKinney . "Python for Data Analysis"

18.Офіційна документація pandas. URL: <https://pandas.pydata.org/docs/>

(дата звернення: 04.04.2023)

19.В.В. Буда, І.С. Гребенькова, О.С. Греченюк, І.А. Пшеничний. "Часові ряди: теорія та практика".

20.І.П. Лебедєв, О.О. Гуменюк, М.П. Гнатюк, О.О. Чигир."Аналіз часових рядів у системах Економетрика та Мінінформіка".

21. А.А. Панченко, Ю.В. Воробйов . "Часові ряди: теорія, методи, моделі".

22.Rob J. Hyndman and George Athanasopoulos . "Forecasting: Principles and Practice".

23."Стационарність в аналізі часових рядів" - <http://infomine.kiev.ua/statistika/9-7-3-stationarity-v-analizi-chasovih-ryadiv.html>

(дата звернення: : 18.04.2023)

24."Introduction to Time Series Analysis in Python" - <https://machinelearningmastery.com/time-series-data-stationary-python/>
(дата звернення: 18.04.2023)

25.В. В. Гришаєв, О. В. Микитин . "Макроекономіка: Теорія та практика"

26.William M. Scarth ."Macroeconomics: An Introduction".

27.Бібліотека для статистичного моделювання, яка надає моделі прогнозування, такі як ARIMA, SARIMA, VAR, ARCH, GARCH та інші.Statsmodels. URL: (<https://www.statsmodels.org/stable/index.html>).

(дата звернення: 20.03.2023)

28.Бібліотека машинного навчання з широким спектром моделей прогнозування, включаючи лінійну регресію, вирішальні дерева, випадкові ліси, градієнтний бустинг та інші.Scikit-learn. URL: (<https://scikit-learn.org/stable/>).

(дата звернення: 25.03.2023)

- 29.Посібник з тимчасових рядів у документації statsmodels: У документації statsmodels є розділ, присвячений аналізу стаціонарності часових рядів, включаючи використання тесту Дікі-Фуллера.URL: <https://www.statsmodels.org/stable/tsa.html#stationarity-tests-and-statistical-tests-for-randomness>
(дата звернення: 26.04.2023)
- 30.О.В. Осауленко, О.С. Рафальська. "Макроекономічний аналіз України: проблеми, тенденції, перспективи" (Macroeconomichnyi analiz Ukrainy: problemy, tendentsii, perspektyvy).
- 31.Padma Desai and Oleh Havrylyshyn . "The Ukrainian Economy: Achievements, Problems, Challenges".
- 32.Kevin P. Murphy ."Machine Learning: A Probabilistic Perspective".
- 33.Christopher M. Bishop ."Pattern Recognition and Machine Learning".
- 34.Wes McKinney ."Python for Data Analysis" (2nd Edition).
- 35.Офіційна документація Matplotlib. URL: <https://matplotlib.org/>
(дата звернення: 25.05.2023)
- 36.Max Kuhn and Kjell Johnson. "Applied Predictive Modeling".
- 37.NumPy Quickstart Tutorial by Real Python.URL: <https://realpython.com/numpy-tutorial-getting-started/>
(Дата звернення: 26.02.2023)
- 38.NumPy Tutorial by W3Schools. URL: https://www.w3schools.com/python/numpy_intro.asp
(дата звернення: 27.03.2023)
- 39.Офіційна документація NumPy. URL: <https://numpy.org/doc/>
(дата звернення: (20.02.2023)
- 40.Qiang Zhan, Ben De Wang, Bin He, Yong Peng, Ming Lei Ren. Singular Spectrum Analysis and ARIMA Hybrid Model for Annual Runnoff Forecasting // Water Resource Manag. 2011