

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Навчально-науковий інститут комп'ютерних наук та штучного інтелекту
Кафедра комп'ютерних систем та робототехніки

«Затверджую»
в.о. завідуючого кафедри
комп'ютерних систем та робототехніки
_____ к. ф.-м. н., доцент Максим Хруслов
«___» червня 2025 р.

Пояснювальна записка

до кваліфікаційної роботи
бакалавра

на тему: «Комп'ютерна модель прогнозування споживання
електроенергії за допомогою методів аналізу часових рядів»

Спеціальність 123 – Комп'ютерна інженерія.
Галузь знань: 12 – Інформаційні технології.
Освітня програма «Комп'ютерна інженерія».

Захищено на засіданні
Екзаменаційної комісії № 44
протокол № __ від __.06.2025 р.
Оцінка ____ / ____
Голова Екзаменаційної комісії
_____ ЧУГАЙ А.М.

Виконав:
Студент групи КІ– 41
ХАВРОНЯ Артем Родіонович _____
(підпис)

Керівник: к. т. н., доцент,
доцент ЗВО кафедри комп'ютерних
систем та робототехніки
БАКУМЕНКО Ніна Станіславівна _____
Рецензент: к. т. н., с.н.с.,
доцент ЗВО кафедри кібербезпеки
інформаційних систем, мереж і технологій
СВАТОВСЬКИЙ Ігор Іванович _____

Харків – 2025

АНОТАЦІЯ

Кваліфікаційна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел і додатків. Загальний обсяг роботи становить 70 сторінки, з яких 49 сторінок основної частини з 22 рисунками та 1 таблицею, 3 сторінки використаних джерел із 23 найменувань, додатки на 15 сторінках.

Метою даної роботи є створення комп'ютерної моделі аналізу прогнозування споживання електроенергії за допомогою методів машинного навчання, зокрема, аналізу часових рядів.

Об'єктом дослідження є процес аналізу та прогнозування обсягів спожитої кількості електроенергії.

Предметом дослідження є моделі, методи та інформаційні технології аналізу часових рядів.

В кваліфікаційній роботі описано цілі та завдання проекту. Приведено теоретичні відомості, необхідні для чіткого формулювання знань в цій галузі. Обговорено питання розробки, тестування та аналізу комп'ютерної моделі за допомогою якої можна буде спрогнозувати споживання електроенергії.

Ключові слова: МОДЕЛЬ, ARIMA, SARIMA, SARIMAX, ARIMAX, БУСТИНГ, БЕГІНГ

ABSTRACT

The thesis consists of an introduction, three chapters, conclusions, a list of references, and appendices. The total volume of the work is 70 pages, including 49 pages of the main part with 22 figures and 1 table, 3 pages of references with 23 sources, and 15 pages of appendices.

The thesis aims of this thesis is to develop a model for forecasting electricity consumption using time series analysis methods.

The object of the research is the application and use of time series analysis methods.

The subject of the research is time series analysis methods.

The thesis outlines the goals and objectives of the project. It provides the theoretical background necessary for a clear understanding of the field. It discusses the development, testing, and analysis of a computer model capable of forecasting electricity consumption.

Keywords: MODEL, ARIMA, SARIMA, SARIMAX, ARIMAX, BOOSTING, BAGGING

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ І УМОВНИХ ПОЗНАЧЕНЬ	6
ВСТУП	7
РОЗДІЛ 1 МЕТОДИ РОЗВ’ЯЗКУ ЗАДАЧІ ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ.....	8
1.1 Опис предметної галузі	8
1.1.1 Значення електроенергії в сучасному світі.....	8
1.1.2 Вплив неточних прогнозів на енергосистему.....	9
1.1.3 Сезонність та добова варіативність електроспоживання	9
1.1.4 Види прогнозування (коротко-, середньо-, довгострокове).....	10
1.1.5 Методи прогнозування (традиційні та сучасні)	11
1.1.6 Економічні та екологічні переваги точного прогнозування	13
Висновки до розділу 1	14
РОЗДІЛ 2 МОДЕЛЮВАННЯ ЧАСОВОГО РЯДУ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ ЗА ДОПОМОГОЮ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ	16
2.1 Основні концепції часових рядів	16
2.1.1 Аналіз часових рядів	17
2.1.2 Прогнозування як процес передбачення	18
2.2 Аналіз методів машинного навчання для прогнозування	19
2.2.1 Теоретичні аспекти методів машинного навчання, що застосовуються для прогнозування	19

2.2.2 Використання методів машинного навчання для прогнозування часових рядів	20
2.3 Актуальність, новизна та можливе застосування.....	20
2.4 Метод ARIMA	21
2.5 Метод SARIMA.....	23
2.6 Метод ARIMAX	25
2.7 Метод SARIMAX.....	27
Висновок до розділу 2	29
РОЗДІЛ 3 ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕТОДІВ.....	30
3.1 Вибір датасету	30
3.2 Прогнозування за допомогою ARIMA	34
3.3 Прогнозування за допомогою SARIMA	37
3.4 Прогнозування за допомогою ARIMAX	39
3.5 Прогнозування за допомогою SARIMAX	41
3.6 Ансамблювання як спосіб покращити точність	43
3.7 Ансамблювання методом бустингу	45
3.8 Ансамблювання методом бегінгу	48
Висновок до розділу 3	50
ВИСНОВКИ	52
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	53
ДОДАТКИ	56

ПЕРЕЛІК СКОРОЧЕНЬ І УМОВНИХ ПОЗНАЧЕНЬ

ВДЕ - відновлювані джерела енергії;

ТЕС - теплова електростанція;

ККД - коефіцієнт корисної дії;

СЕС - сонячна електростанція.

ВСТУП

У сучасному світі електроенергія є невід'ємною складовою економіки та повсякденного життя, а ефективне управління її споживанням стає дедалі актуальнішим завданням. В умовах зростаючого навантаження на енергетичні системи, розвитку відновлюваних джерел енергії та нестабільності ринкових умов, точне прогнозування споживання електроенергії відіграє ключову роль у забезпеченні стабільності електромереж, зменшенні витрат та підвищенні енергоефективності.

Методи аналізу часових рядів є одними з найефективніших підходів до прогнозування енергоспоживання. Використання таких моделей, як ARIMA, SARIMA, експоненціальне згладжування, нейронні мережі та методи машинного навчання, дозволяє враховувати сезонність, тренди та випадкові коливання, що впливають на споживання електроенергії. Завдяки цим методам можна отримати високоточні прогнози, які сприяють оптимізації виробництва та розподілу електроенергії.

Прогнозування споживання електроенергії є важливим інструментом для енергетичних компаній, підприємств та державних органів. Воно дозволяє зменшити ризики перевантаження електромереж, оптимізувати генерацію електроенергії та забезпечити стабільну роботу критично важливих об'єктів. Особливо це актуально в умовах надзвичайних ситуацій, коли необхідно приймати швидкі та ефективні рішення щодо розподілу електроенергії.

Таким чином, розробка комп'ютерної моделі прогнозування споживання електроенергії за допомогою методів аналізу часових рядів є актуальним і важливим завданням, яке сприятиме підвищенню ефективності енергетичних систем та їх надійності.

РОЗДІЛ 1

МЕТОДИ РОЗВ'ЯЗКУ ЗАДАЧІ ПРОГНОЗУВАННЯ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ

1.1 Опис предметної галузі

1.1.1 Значення електроенергії в сучасному світі

Електроенергія є життєво важливим ресурсом для функціонування всіх сфер сучасного суспільства - від промисловості й транспорту до охорони здоров'я та освіти, а її стабільне постачання є основою економічної стабільності; водночас рівень електроспоживання виступає індикатором розвитку країни: у розвинених державах він високий через енергоємну економіку, тоді як у країнах, що розвиваються, зростання споживання свідчить про індустріалізацію та покращення якості життя.

Зі зростанням населення Землі, яке, за прогнозами ООН, досягне 9,7 мільярда осіб до 2050 року, а також із процесами урбанізації, глобальне споживання електроенергії щороку зростає. Це пов'язано не лише з демографічними факторами, а й зі збільшенням кількості електронних пристроїв, електрифікацією транспорту (електромобілі, електричні поїзди, метро), розвитком індустрії розваг та повсюдним переходом до цифрових технологій [1].

Згідно з аналітичним звітом Міжнародного енергетичного агентства (IEA), у 2023 році глобальне споживання електроенергії зросло на 4% - це найвищий темп приросту за останні 10 років. Найбільше зростання зафіксовано у країнах Азії, зокрема в Китаї та Індії, що пов'язано з бурхливим розвитком інфраструктури, промисловості та цифрової економіки. У Європі та США зростання було більш помірним, однак спостерігалось зростання споживання в таких секторах, як дата-центри, телекомунікації та "зелена" енергетика [2].

Розвиток електроенергетики є ключовим чинником сталого розвитку, оскільки електрифікація транспорту й опалення замість використання викопного палива сприяє зменшенню викидів парникових газів і зростанню попиту на електроенергію, що перетворює її на стратегічний ресурс, необхідний для досягнення кліматичних цілей, цифрової трансформації, інновацій і підвищення якості життя.

1.1.2 Вплив неточних прогнозів на енергосистему

Прогнозування електроспоживання є критично важливим для стабільного функціонування енергосистеми: неточності призводять до технічних збоїв і значних економічних втрат.

Завищений прогноз спричиняє перевиробництво й низьку ринкову вартість електроенергії, додаткові витрати на зберігання або відключення потужностей, особливо проблематично з нерегульованими відновлюваними джерелами.

Занижений прогноз викликає дефіцит енергії, перевантаження мережі, аварійні відключення та ризики для критичних галузей. В умовах лібералізованих ринків, де ціни формуються щогодини чи щоквартально, похибки (навіть 5 %) означають мільйонні збитки через штрафи та необхідність швидкого балансування зі сторонніми резервами, що додатково знижує ефективність системи та збільшує викиди шкідливих речовин.

1.1.3 Сезонність та добова варіативність електроспоживання

Непостійність споживання електроенергії проявляється у сезонних, тижневих і добових коливаннях, що формує динамічний попит і вимагає гнучкого планування та точного прогнозування; сезонна варіативність зумовлена кліматом: взимку зростає споживання через опалення та освітлення, а влітку - через активне використання кондиціонерів, що особливо навантажує систему в денні години.

Згідно з даними U.S. Energy Information Administration (EIA), сезонна амплітуда споживання в країнах з вираженою зміною пір року може досягати

20-30% від середнього річного рівня, що вимагає гнучкого управління генерацією та розподілом ресурсів [4].

Окрім сезонності, добова варіативність електроспоживання також є ключовим фактором і включає нічний спад (00:00-06:00), ранкове зростання (06:00-10:00) та вечірній пік (17:00-21:00); такі коливання, особливо за високої частки ВДЕ, ускладнюють балансування енергосистеми, тому точне прогнозування з урахуванням цих змін є критично важливим для її стабільної роботи.

У сучасних підходах до прогнозування використовуються часові ряди, що дозволяє враховувати закономірності споживання з різною періодичністю. Моделі типу SARIMA або LSTM дозволяють точно передбачати як короткострокові піки, так і довгострокові тренди, що доведено у дослідженні Wang et al. (2020), яке показало підвищення точності прогнозу на 15-20% при врахуванні сезонних компонент [5].

1.1.4 Види прогнозування (коротко-, середньо-, довгострокове)

Прогнозування електроспоживання в енергосистемах поділяється на три основні типи залежно від горизонту прогнозу: короткострокове (до 24-48 годин), середньострокове (від кількох днів до місяців) і довгострокове (від пів року до кількох років). Кожен тип має своє призначення та вимагає використання різних моделей і підходів.

Короткострокове прогнозування охоплює період від кількох хвилин до двох діб і є критично важливим для оперативного управління енергосистемою: балансування, керування генерацією, планування резервів і реагування на пікові навантаження. Воно враховує погодні умови в реальному часі, календарні чинники та останні значення навантаження.

Згідно з дослідженням Using the LSTM Network to Forecast the Demand for Electricity in Poland (2020), короткострокове прогнозування із застосуванням LSTM дало похибку менше 2% на горизонті до 24 годин [6].

Середньострокове прогнозування, що охоплює період від кількох днів до місяців, використовується для планування обслуговування обладнання, закупівлі палива, укладання контрактів і моделювання сезонних змін попиту. Воно базується на аналізі історичних даних, кліматичних умов, економічних показників і ринкових факторів, із застосуванням регресійних методів, машинного навчання та ієрархічних моделей.

Такі прогнози менш точні в абсолютному значенні, ніж короткострокові, проте вони важливі для стратегічного ресурсного планування та економічного моделювання. Згідно з аналізом Європейської мережі операторів системи передачі електроенергії (ENTSO-E), середньострокові прогнози можуть мати похибку 5-10% [7].

Довгострокове прогнозування (від кількох місяців до десятків років) є основою для інфраструктурного планування, розвитку ВДЕ, формування енергетичної політики та оцінки екологічного впливу; воно базується на макроекономічних, демографічних, технологічних і політичних сценаріях із використанням сценарного аналізу, системної динаміки та економетричних моделей, що, попри високу невизначеність, забезпечує стратегічне бачення розвитку енергетики.

1.1.5 Методи прогнозування (традиційні та сучасні)

Для ефективного управління енергосистемами застосовуються два основні підходи до прогнозування: традиційні статистичні методи, що аналізують закономірності в історичних даних, і сучасні методи штучного інтелекту та машинного навчання, які дозволяють виявляти складні нелінійні залежності у великих масивах інформації.

Почнемо розгляд з традиційних методів. Лінійна регресія - це один з найпростіших способів передбачення значення залежної змінної (наприклад, навантаження) на основі однієї або кількох незалежних змінних (наприклад, температура, день тижня, година доби) [8].

Модель має вигляд:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n + \varepsilon \quad (1.1)$$

Достоїнства - проста у реалізації, добре підходить для короткострокових прогнозів із чітко вираженою лінійною тенденцією.

Недоліки - не враховує складні нелінійні залежності, чутлива до мультиколінеарності між факторами.

Наступними йдуть методи експоненційного згладжування. Методи цієї групи дозволяють враховувати тренд та сезонність. Мають хорошу продуктивність у задачах короткострокового прогнозування, особливо за стабільних патернів навантаження.

Моделі типу ARIMA дозволяють будувати прогноз на основі автокореляцій попередніх значень. SARIMA додатково враховує сезонність. Наприклад, модель SARIMA(p,d,q)(P,D,Q)_s використовує: p, d, q - параметри авторегресії, інтегрування та ковзного середнього, P, D, Q, s - ті ж параметри, але для сезонної компоненти (s - період сезонності).

За дослідженням Hyndman & Athanasopoulos (2018), ARIMA-моделі можуть досягати точності до 90% у задачах добового прогнозування за стабільного профілю споживання [9].

Серед сучасних методів прогнозування особливо популярні нейронні мережі, зокрема багат шарові перцептрони, що моделюють складні нелінійні зв'язки на основі історичних даних, хоча потребують великих обсягів даних і важко піддаються інтерпретації. Ефективнішими для часових рядів є рекурентні нейронні мережі, зокрема LSTM, які спеціально адаптовані для роботи з послідовностями та добре підходять для прогнозування електроспоживання.

У дослідженні Comparative analysis of ARIMA and LSTM for predicting fluctuating time series data (2023), LSTM показала вищу точність порівняно з

традиційними моделями ARIMA, зменшуючи середню абсолютну похибку на 15% [10].

Далі йде градієнтний бустинг. Методи бустингу використовують ансамблі дерев рішень для побудови потужних моделей. Вони дають чудові результати в задачах прогнозування з великою кількістю вхідних параметрів.

Наступні це гібридні моделі. У багатьох сучасних дослідженнях застосовуються гібридні підходи, які поєднують сильні сторони різних методів. Це дозволяє моделювати як лінійні, так і нелінійні компоненти споживання електроенергії з високою точністю.

У роботі Time Series Forecasting using LSTM and ARIMA (2023) було показано, що поєднання LSTM з обробкою сигналів методом ЕМД (Empirical Mode Decomposition) забезпечує точність прогнозу до 95% при добовому горизонті [11].

1.1.6 Економічні та екологічні переваги точного прогнозування

Точне прогнозування електроспоживання має суттєві економічні переваги: воно дозволяє ефективніше планувати генерацію, знижуючи витрати на роботу резервних потужностей, і мінімізує витрати на закупівлю електроенергії на лібералізованих ринках, уникаючи дорогих балансуєвих операцій. Крім того, воно сприяє підвищенню ефективності енергоменеджменту, зокрема в промисловості та «розумних» будівлях, де автоматизовані системи адаптуються до очікуваного навантаження.

З екологічного боку точність прогнозів дозволяє уникнути перевиробництва енергії, що зменшує викиди шкідливих речовин, особливо при використанні резервних джерел з низьким ККД. Також це дає змогу енергетичним компаніям оптимізувати інвестиції в інфраструктуру, уникаючи непотрібного розширення мережі, що є особливо важливим для країн із обмеженими ресурсами на модернізацію.

Зниження викидів парникових газів. Точне прогнозування дозволяє мінімізувати використання неефективних, вуглецево-інтенсивних резервних

джерел енергії (наприклад, газотурбінних установок або вугільних ТЕС). Це призводить до зменшення обсягів CO₂, та інших шкідливих речовин, що потрапляють в атмосферу.

Більш ефективна інтеграція ВДЕ. Прогноз попиту з високою точністю дозволяє краще узгоджувати споживання з генерацією від сонця чи вітру, які є змінними. Це зменшує потребу в балансуєчих вугільних або газових станціях, підвищуючи загальну частку ВДЕ у виробництві та зменшуючи екологічне навантаження.

Підвищення ефективності «розумних мереж». У рамках концепції енергетичного переходу smart grids передбачають гнучке управління енергоспоживанням і виробництвом. Без точного прогнозування попиту функціонування таких мереж є неефективним. Точність прогнозу прямо впливає на інтеграцію децентралізованих джерел енергії, зокрема домашніх СЕС і мікро-ТЕС.

Підтримка кліматичних зобов'язань країн. У рамках Паризької кліматичної угоди держави зобов'язані знижувати викиди CO₂. Одним з важливих кроків до досягнення цих цілей є оптимізація енергоспоживання за допомогою прогнозування. Наприклад, у дослідженні, проведеному Lawrence Berkeley National Laboratory (США), зазначено, що точне керування попитом в масштабі одного міста здатне зменшити щорічні викиди CO₂ на десятки тисяч тонн [12].

Висновки до розділу 1

У сучасних умовах зростаючого енергоспоживання, урбанізації, цифровізації та переходу до електротранспорту прогнозування електроспоживання набуває особливої актуальності. Точне прогнозування дозволяє ефективно планувати ресурси, забезпечувати надійність мережі, зменшувати витрати на резервні потужності та уникати перевантажень. Вибір методів прогнозування залежить від якості даних, періоду прогнозу та особливостей енергосистеми, а сучасні підходи часто поєднують статистичні

методи з інструментами машинного навчання й глибокими нейронними мережами.

Помилки у прогнозах можуть мати серйозні технічні, економічні й екологічні наслідки - від збоїв у постачанні до мільйонних збитків і надлишкових викидів CO₂. Особливо важливо це в умовах інтеграції відновлюваних джерел енергії, які є нестабільними й потребують точного прогнозування для ефективної інтеграції. Прогнозування електроспоживання сьогодні - це не лише технічне завдання, а стратегічний інструмент сталого розвитку, який поєднує інновації, економічну вигоду та відповідальність за майбутнє енергетики.

РОЗДІЛ 2

МОДЕЛЮВАННЯ ЧАСОВОГО РЯДУ СПОЖИВАННЯ ЕЛЕКТРОЕНЕРГІЇ ЗА ДОПОМОГОЮ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

2.1 Основні концепції часових рядів

Часовий ряд - це послідовність спостережень певної змінної, впорядкованих за часом, де кожне значення відповідає конкретному моменту або інтервалу часу. Такі ряди широко використовуються для аналізу та прогнозування в різних сферах, включаючи економіку, фінанси, метеорологію та інженерію.

На відміну від звичайних наборів даних, де спостереження можуть бути незалежними та не мати природного порядку, часові ряди мають внутрішню структуру, обумовлену послідовністю в часі. Ця послідовність дозволяє виявляти часові залежності та закономірності, які неможливо визначити в статичних наборах даних.

Часові ряди зазвичай складаються з кількох основних компонентів:

Тренд(Trend) - це довгострокова зміна даних, що вказує на загальний напрямок розвитку. Сезонність(Seasonal) - регулярні коливання з фіксованою періодичністю, пов'язані з календарем чи погодою. Циклічність(Cyclical) - нерегулярні зміни, зумовлені економічними або іншими циклами. Випадкова складова(Irregular fluctuations) - це непередбачувані флуктуації, спричинені унікальними подіями або шумом.

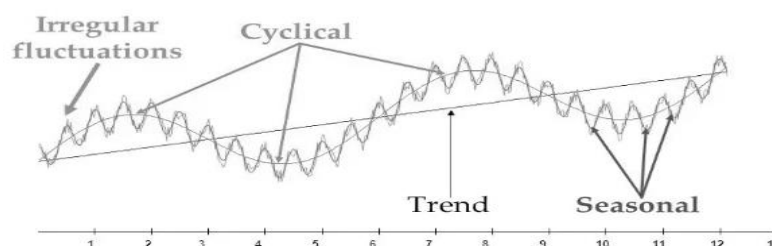


Рисунок 2.1 - компоненти часових рядів.

Існують різні методи для ідентифікації та аналізу зазначених компонентів:

Рухомі середні згладжують дані для виявлення тренду. Декомпозиція розділяє часовий ряд на тренд, сезонність і випадкову складову. Автокореляційна функція виявляє залежності між значеннями на різних лагах, що допомагає ідентифікувати сезонність. Часткова автокореляційна функція оцінює прямий вплив значень на певному лагу, корисна для визначення порядку AR-моделі [13].

2.1.1 Аналіз часових рядів

Аналіз часових рядів є методом дослідження послідовностей даних, організованих відповідно до часу їх виникнення. Основна мета цього аналізу полягає у виявленні закономірностей і характеристик, що можуть бути приховані в динаміці часових показників.

Однією з ключових задач аналізу часових рядів є прогнозування. Досліджуючи минулі дані та їх зміни в часі, можна розробити моделі, які дозволяють передбачити майбутні значення. Це є важливим інструментом для бізнесу та економіки, де здатність передбачати майбутні тенденції допомагає у ефективному управлінні ресурсами та прийнятті рішень.

Аналіз часових рядів застосовується в різних сферах, таких як фінанси, метеорологія, виробництво та інші. Він допомагає виявляти регулярності у даних, що є важливим для більш точного розуміння ситуації та прийняття обґрунтованих управлінських рішень.

Аналіз часових рядів можна поділити на дві основні категорії: методи, що працюють у часовому домені, і методи, що спираються на частотний домен. У першому випадку використовуються прямі спостереження за даними та їхнім розвитком, наприклад, аналіз графіків часових рядів для визначення тенденцій, коливань і циклічності. Статистичні методи, такі як ковзні середні,

застосовуються для згладжування короткострокових коливань і виявлення довгострокових трендів. Автокореляційний аналіз дозволяє досліджувати кореляції між послідовними спостереженнями на різних часових лагах. Методи частотного домену, такі як перетворення Фур'є, дозволяють розкласти часовий ряд на окремі циклічні компоненти.

Процес аналізу часових рядів включає кілька етапів:

Підготовка даних включає збирання, перевірку, обробку пропусків і оцінку стаціонарності. Дослідницький аналіз виявляє тренди, сезонність і кореляції. Далі здійснюється ідентифікація моделі (AR, MA, ARIMA тощо) і побудова прогнозу. Точність оцінюється за метриками, після чого модель застосовується на практиці для прогнозування попиту чи інших процесів [14].

2.1.2 Прогнозування як процес передбачення

Прогнозування - це процес оцінки майбутніх подій або стану системи, що ґрунтується на аналізі наявних даних та використанні спеціалізованих методів. Воно дозволяє приймати виважені рішення в умовах невизначеності та мінімізувати потенційні ризики. Основна мета прогнозування - виявлення закономірностей у даних та побудова моделей, здатних передбачати майбутні тенденції.

Процес прогнозування включає збір і обробку даних, вибір відповідного методу та його застосування в конкретній сфері - фінансах, медицині, економіці чи технологіях. Важливо враховувати можливі обмеження методів та використовувати сучасні цифрові інструменти для підвищення точності прогнозів.

Методи прогнозування можна поділити на кілька основних груп:

Методи прогнозування поділяють на статистичні, машинного навчання, експертні, оптимізаційні та комбіновані. Статистичні аналізують історичні дані для виявлення трендів, машинне навчання - ефективне при складних закономірностях. Експертні оцінки корисні за браку даних, а оптимізаційні - для вибору найкращих рішень. Комбіновані підходи забезпечують вищу

точність. Вибір методу залежить від завдання, доступних даних і потреб планування [15].

2.2 Аналіз методів машинного навчання для прогнозування

2.2.1 Теоретичні аспекти методів машинного навчання, що застосовуються для прогнозування

Обробка даних є ключовим етапом в процесі машинного навчання, що включає кілька важливих задач для забезпечення ефективної роботи моделей:

Обробка пропущених значень - на цьому етапі виявляються та коригуються пропущені дані. Це може включати заміщення пропусків середніми або медіанними значеннями, або видалення рядків чи стовпців з пропусками. Цей процес повинен здійснюватися обережно, щоб не втратити важливу інформацію.

Перед початком навчання моделі важливо підготувати дані. Категоріальні змінні слід перетворити на числові, наприклад, за допомогою one-hot кодування. Числові ознаки потрібно нормалізувати, щоб уникнути перекосів у навчанні. Також часто застосовується генерація ознак - створення нових змінних на основі існуючих, що дозволяє моделі краще виявляти закономірності.

Дані ділять на тренувальний і тестовий набори. На першому модель навчається, на другому - оцінюється її здатність до узагальнення. Ефективність вимірюється різними метриками: для класифікації - матриця плутанини, для регресії - графіки порівняння прогнозів і реальних значень.

Особливу увагу приділяють боротьбі з перенавчанням, коли модель занадто точно запам'ятовує тренувальні дані, втрачаючи здатність узагальнювати. Для запобігання цьому використовують регуляризацію, більші обсяги даних або простіші моделі.

Недонавчання відбувається, коли модель не встигає вивчити всі важливі закономірності в даних і, як наслідок, показує погані результати на тренувальних та тестових даних. Для боротьби з недонавчанням можна

використовувати складніші моделі, додавати нові ознаки або збільшувати час тренування моделі [16].

2.2.2 Використання методів машинного навчання для прогнозування часових рядів

Вибір методу прогнозування часових рядів залежить від структури даних: регулярності вимірювань, наявності трендів і сезонності. Для даних з фіксованим інтервалом підходять моделі типу ARIMA, а для сезонних змін - SARIMA або LSTM. Аналіз автокореляцій допомагає визначити, яка модель краще враховує внутрішні зв'язки в даних.

Великі обсяги даних доцільно аналізувати за допомогою глибоких нейронних мереж (наприклад, LSTM), які здатні виявляти складні закономірності. Проте їх використання вимагає значних обчислювальних ресурсів. У випадках з аномаліями або нестабільною поведінкою корисними є адаптивні моделі, здатні швидко підлаштовуватись до змін.

Швидкість прогнозування також є важливим чинником при виборі методу. Для реального часу або швидких реакцій на зміни даних підходять прості моделі, як-от ARIMA чи експоненційне згладжування, що мають низьку обчислювальну складність. Вони можуть генерувати прогнози швидко, але не завжди здатні моделювати складні залежності. Глибокі нейронні мережі, хоча й потужніші, потребують більше часу для тренування та прогнозування, а також значних обчислювальних ресурсів [17].

2.3 Актуальність, новизна та можливе застосування

У сучасному світі енергетична безпека стала одним із ключових аспектів національної та міжнародної безпеки. Політична нестабільність, природні катастрофи, терористичні загрози та інші непередбачувані події можуть серйозно порушити роботу енергетичних систем, що, у свою чергу, спричинить проблеми з постачанням електроенергії для населення та інфраструктури.

У цьому контексті стає надзвичайно важливим розвиток методів прогнозування та оптимізації розподілу електроенергії під час кризових ситуацій. Це питання набуває особливої актуальності в умовах сучасних конфліктів і геополітичних напружень, які можуть значно вплинути на енергетичні системи різних країн і регіонів.

Дослідження в цій сфері зосереджуються на розробці та впровадженні сучасних методів прогнозування, таких як ARIMA, ARIMAX, SARIMA, SARIMAX, для аналізу та управління енергопостачанням під час кризових подій. Практичне значення цього дослідження полягає в тому, що його результати можуть бути використані для ефективного управління енергетичними ресурсами та зменшення наслідків кризових ситуацій у галузі електропостачання.

У кризових умовах, зокрема під час війни, прогнозування електроспоживання має критичне значення для стабільного енергозабезпечення населення та стратегічних об'єктів. Воно дозволяє оперативно реагувати на зміни попиту, оптимізувати розподіл ресурсів у разі пошкодження інфраструктури та ефективно керувати електропостачанням під час блекаутів, забезпечуючи пріоритетні потреби.

2.4 Метод ARIMA

ARIMA - це статистичний метод, призначений для аналізу та прогнозування часових рядів. Він об'єднує три основні компоненти: авторегресію (AR), інтегрування (I) та ковзні середні (MA).

Формальне визначення: якщо $W_t = \nabla^d Y_t$ є стаціонарним процесом ARMA(p,q) Y_t - це процес ARIMA(p,d,q)

Формула (2.1) для ARIMA(p,1,d):

$$Y_t - Y_{t-1} = \phi_1(Y_{t-1} - Y_{t-2}) + \dots + \phi_p(Y_{t-p} - Y_{t-p-1}) + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_p \varepsilon_{t-p} \quad (2.1)$$

Оперативна форма формула (2.2):

$$(1 - \sum_{i=1}^p \phi_i L^i)(1 - L)^d X_t = (1 - \sum_{j=1}^q \theta_j L^j) \quad (2.2)$$

де: L - оператор зсуву назад, d - порядок диференціювання.

Авторегресія (AR) полягає у тому, що значення в певний момент часу залежить від попередніх значень цього ж ряду. Це означає, що поточний результат може бути спрогнозований на основі його попередніх значень. Параметр AR (p) вказує на кількість попередніх значень, які враховуються для прогнозування.

Інтегрування (I) необхідне для того, щоб усунути тренди та сезонні коливання в ряді. Це досягається через диференціювання - віднімання попереднього значення від поточного, що дозволяє зробити ряд стаціонарним і підвищити точність прогнозів. Параметр d визначає кількість таких диференціювань, необхідних для досягнення стаціонарності.

Ковзні середні (MA) враховують помилки прогнозу, створюючи зв'язок між поточними значеннями і попередніми помилками прогнозування. Параметр MA (q) визначає кількість попередніх помилок, які беруться до уваги при прогнозуванні.

Модель ARIMA описується як ARIMA(p, d, q), де p , d , і q - це порядки авторегресії, інтегрування та ковзних середніх відповідно. Ці три компоненти разом дозволяють прогнозувати майбутні значення на основі вже наявних даних. Якщо в даних спостерігається сезонність, ARIMA може бути розширено до SARIMA (сезонна ARIMA), яка враховує сезонні коливання.

Перевірка стаціонарності є важливим кроком при використанні ARIMA. Для цього використовуються різні методи, включаючи візуальний аналіз графіків, де на основі середнього значення та дисперсії визначається, чи змінюються статистичні характеристики з часом. Також застосовуються

статистичні тести, такі як тест Дікі-Фуллера, для більш точного визначення стаціонарності.

Диференціювання - це процес, що дозволяє усунути сезонність та тренди в даних. Перший крок диференціювання передбачає обчислення різниці між кожним наступним і попереднім значенням ($X_t - X_{t-1}$). У випадку, коли цього недостатньо, застосовують друге диференціювання ($\text{diff}_t - \text{diff}_{t-1}$), щоб отримати стабільніші результати.

Побудова та оцінка моделі ARIMA включає кілька етапів. Спочатку визначають початкові значення параметрів (p, d, q), зазвичай за допомогою аналізу автокореляційних функцій (ACF) та часткових автокореляційних функцій (PACF). Після цього проводиться оцінка параметрів за допомогою методів, таких як метод максимальної правдоподібності.

Остаточна модель ARIMA (p, d, q) повинна пройти діагностику на адекватність, яка включає перевірку залишкових даних, щоб переконатися, що вони нормальні і не мають автокореляції.

Переваги методу ARIMA включають його здатність до прогнозування короткострокових та середньострокових змін у часі, що робить його корисним інструментом у фінансах, економіці та маркетингу. Однак, існують деякі обмеження, зокрема щодо складності вибору оптимальних параметрів моделі, а також можливі проблеми з лінійністю зв'язків у складних часових рядах [18].

2.5 Метод SARIMA

SARIMA - це розширена версія моделі ARIMA, яка включає додатковий компонент для моделювання сезонних коливань у часових рядах. SARIMA враховує не лише загальну динаміку часового ряду, але й сезонні впливи, що є важливим для аналізу даних з періодичними змінами, такими як продажі в різні пори року, зміни температури чи економічні цикли.

Модель SARIMA складається з чотирьох основних компонентів:

Авторегресія (AR) - подібно до ARIMA, авторегресія в SARIMA описує залежність поточного значення від попередніх значень часового ряду.

Інтегрування (I) - цей компонент допомагає зробити часовий ряд стаціонарним шляхом зняття тренду або сезонності через диференціювання.

Ковзні середні (MA) - ковзні середні дозволяють зменшити помилки прогнозування, пов'язані з неідеальними передбаченнями на основі попередніх значень.

Сезонні компоненти - у SARIMA до AR, I, MA додаються сезонні параметри. Сезонний авторегресивний компонент (SAR), сезонне інтегрування (SI), сезонне ковзне середнє (SMA), а також період сезонності (s), що вказує на кількість періодів у сезонному циклі.

Математичне представлення моделі SARIMA може бути записане як:

$$SARIMA(p, d, q)(P, D, Q)_s$$

Формула(2.3)

$$\phi(B^s)\phi(B)\nabla^d\nabla_s^D Y_t = \theta(B^s)\theta(B)\varepsilon_t \quad (2.3)$$

де: p - порядок авторегресії, d - порядок диференціювання, q - порядок ковзних середніх, P, D, Q - сезонні компоненти відповідно для авторегресії, інтегрування і ковзних середніх, s - період сезонності, B - оператор зсуву назад, сезонні та не сезонні компоненти: $\phi(B^s), \phi(B), \theta(B^s), \nabla_s = 1 - B^s$, сезонне різницювання.

Ключові етапи побудови SARIMA:

Перевірка на стаціонарність - перед використанням SARIMA необхідно перевірити, чи є часовий ряд стаціонарним. Якщо дані мають тренд або сезонні коливання, застосовуються відповідні диференціювання (звичайне або сезонне).

Вибір параметрів моделі - за допомогою аналізу автокореляції (ACF) і часткової автокореляції (PACF), а також діаграм сезонності, вибираються оптимальні значення параметрів для AR, MA, SAR і SMA.

Оцінка параметрів моделі - після визначення параметрів, проводиться оцінка моделі за допомогою методів максимальної правдоподібності чи мінімізації залишків.

Прогнозування - після налаштування моделі можна здійснити прогнозування майбутніх значень з урахуванням сезонних коливань.

Переваги SARIMA:

SARIMA дозволяє моделювати сезонні коливання, що робить її дуже корисною для обробки даних з чіткими сезонними патернами.

Модель добре працює для даних, що мають регулярні повторення в часі, наприклад, місячні чи квартальні дані.

Недоліки SARIMA:

Вибір правильних параметрів для сезонних компонентів може бути складним і вимагає детального аналізу.

SARIMA не дуже ефективна для прогнозування даних з дуже складними або непередбачуваними сезонними коливаннями.

Також існує проблема, коли сезонність сильно змінюється або є неможливою для чіткої ідентифікації, що може знизити точність прогнозів [19].

2.6 Метод ARIMAX

ARIMAX є вдосконаленою модифікацією моделі ARIMA, що надає можливість враховувати екзогенні змінні, що дозволяє точніше відображати складні взаємозв'язки між часовими рядами.

Традиційна ARIMA-модель є ефективним інструментом для аналізу й прогнозування часових рядів, і включає три ключові компоненти: авторегресію (AR), інтегрування (I) та ковзне середнє (MA).

Проте класична ARIMA не враховує вплив зовнішніх факторів, що може бути суттєвим обмеженням у випадках, коли аналізований ряд зазнає впливу сторонніх змінних.

Модель ARIMAX розширює можливості ARIMA шляхом додавання екзогенних змінних, що можуть допомогти краще описати або уточнити поведінку часових рядів. Такий підхід має широке застосування в галузях фінансів, економіки, маркетингу та наукових дослідженнях загалом.

$$Y_t = c + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \theta_1 \epsilon_t + \dots + \theta_q \epsilon_t + \beta_1 X_{t1} + \dots + \beta_k X_{tk} + \epsilon_t \quad (2.4)$$

де: Y_t - значення часового ряду в момент часу t , c - константа або зміщення, $\phi_1, \dots, \phi_p$ - параметри авторегресії, $\theta_1, \dots, \theta_q$ - значення часового ряду в попередньому моменті часу, ϵ_t - біле шумове, доданок з нульовим середнім та одиночною дисперсією ($N(0,1)$), $X_{t1}, \dots, X_{tk}$ - екзогенні змінні, $\beta_1, \dots, \beta_k$ - параметри, що відповідають екзогенним змінним, p - порядок авторегресії, q - порядок ковзного середнього.

ARIMAX - модель дозволяє враховувати не лише внутрішні закономірності в часовому ряді, а й вплив зовнішніх чинників. Завдяки цьому вона є ефективним інструментом для аналізу й прогнозування в умовах, де зовнішні впливи мають суттєве значення.

Оцінювання ARIMAX - моделі є ключовим кроком у процесі її побудови й практичного застосування. На цьому етапі здійснюється підбір параметрів моделі, які найкраще узгоджуються з вихідними даними. Зазвичай для цього використовують такі методи, як метод найменших квадратів або метод максимальної правдоподібності, що реалізуються за допомогою спеціалізованого програмного забезпечення, зокрема мовою Python з відповідними бібліотеками.

Ідентифікація та діагностика ARIMAX-моделі мають важливе значення для оцінки її придатності до конкретного завдання. Ідентифікація передбачає вибір відповідної структури моделі, включаючи компоненти AR, I, MA і екзогенні змінні. Після побудови моделі виконують аналіз залишків для перевірки її відповідності даним. Серед діагностичних перевірок - тестування на нормальність, автокореляцію, гомоскедастичність, а також оцінка стаціонарності та гетероскедастичності.

Процес прогнозування за допомогою ARIMAX включає врахування як внутрішніх характеристик часового ряду, так і впливу зовнішніх чинників. Спочатку здійснюється підготовка даних - часового ряду та відповідних екзогенних змінних. Потім будується модель із урахуванням компонентів AR, I, MA та зовнішніх впливів.

Після побудови проводиться оцінка точності прогнозу через діагностичні тести, зокрема перевірку розподілу залишків та автокореляції. На завершальному етапі здійснюється побудова прогнозу майбутніх значень, який може використовуватись для прийняття рішень у таких сферах, як фінанси, економіка, маркетинг тощо [14].

2.7 Метод SARIMAX

Метод SARIMAX є вдосконаленою версією моделі SARIMA, що дозволяє враховувати як внутрішні (ендогенні), так і зовнішні (екзогенні) фактори при моделюванні та прогнозуванні часових рядів.

Основна концепція SARIMAX полягає у поєднанні динамічних внутрішніх залежностей, характерних для SARIMA, з впливом зовнішніх змінних, таких як економічні індикатори, погодні умови тощо, що можуть суттєво впливати на поведінку ряду.

Завдяки можливості включення екзогенних факторів у модель, SARIMAX забезпечує точніше прогнозування у випадках, коли зміни у часовому ряді зумовлені як внутрішніми, так і зовнішніми впливами.

Для ефективного застосування SARIMAX важливо коректно задати параметри моделі - ступені авторегресії (AR), ковзного середнього (MA), сезонної компоненти (S), інтеграції (I) та інші, які підбираються на основі аналізу часових даних і відповідних зовнішніх чинників.

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q} - q + \epsilon_t - \sum_{j=1}^s \phi_j Y_{t-j} + \sum_{j=1}^s \theta_j \epsilon_{t-j} + \beta_1 X_{t1} + \beta_2 X_{t2} + \dots + \beta_k X_{tk} \quad (2.5)$$

де: Y_t - значення часового в момент часу t , c - константа або зміщення, $\phi_1, \phi_2, \dots, \phi_p$ - параметри авторегресії, $\theta_1, \theta_2, \dots, \theta_q$ -параметри ковзного середнього, ϵ_t - біле шумове, доданок з нульовим середнім та одиночною дисперсією($N(0,1)$), $X_{t1}, X_{t2}, \dots, X_{tk}$ -екзогенні змінні, $\beta_1, \beta_2, \dots, \beta_k$ - параметри авторегресії, p - порядок авторегресії, q -порядок ковзного середнього, s -період сезонності, $\phi_1, \phi_2, \dots, \phi_s$ -параметри сезонної авторегресії, $\theta_1, \theta_2, \dots, \theta_q$ - параметри сезонного ковзного середнього.

Після визначення параметрів моделі SARIMAX її можна реалізувати за допомогою спеціалізованого програмного забезпечення, наприклад, у Python з бібліотекою statsmodels.

Процес побудови та вдосконалення SARIMAX є ітеративним і вимагає глибокого аналізу даних для досягнення найкращих результатів прогнозування. Коли модель налаштована, її можна використовувати для передбачення майбутніх значень, враховуючи як внутрішні характеристики часового ряду, так і зовнішні впливи.

Прогнозування з використанням SARIMAX - ключовий етап у часовому аналізі. Воно базується на ARIMA-компонентах, що відображають внутрішню динаміку, та екзогенних змінних, що репрезентують зовнішні фактори. Для здійснення прогнозу потрібно визначити період передбачення та застосувати модель до наявних даних.

Прогнози, отримані з SARIMAX, можуть мати певну похибку, яка залежить від точності моделі, обсягу доступних даних та стабільності зовнішніх умов. Для оцінювання якості прогнозу використовують метрики, такі як sMASE, sMAPE [20].

Висновок до розділу 2

У другому розділі було розглянуто теоретичні та практичні аспекти моделювання часових рядів споживання електроенергії з використанням різних методів прогнозування. Спочатку були викладені основні концепції часових рядів, а також охарактеризовано процес прогнозування як ключовий етап у прийнятті рішень щодо енергоспоживання. Далі було проаналізовано сучасні методи машинного навчання, які застосовуються для прогнозування часових рядів, зокрема їхні теоретичні основи та практичне використання. Окрему увагу приділено актуальності та новизні застосування цих методів у сфері енергетики. У завершальній частині розділу здійснено огляд традиційних статистичних моделей ARIMA, SARIMA, а також їхніх розширень - ARIMAX та SARIMAX, що враховують зовнішні фактори. Представлені методи порівнюються за ефективністю і можливістю застосування для точного прогнозування споживання електроенергії.

РОЗДІЛ 3

ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕТОДІВ

3.1 Вибір датасету

Для прогнозування часового ряду споживання електроенергії для початку треба було знайти датасет. Для цього було обрано сайт [kaggle.com](https://www.kaggle.com), на ньому багато різних датасетів і було обрано датасет споживання електроенергії в Індії. І перше що було зроблено це побудував графік споживання електроенергії рисунок 3.1 [21].



Рисунок 3.1 - Візуалізація датасету.

Перш ніж продовжити роботу з даними, необхідно детальніше ознайомитися з самим датасетом. Один із найкращих способів його дослідження - це розкладання часового ряду на окремі компоненти: тренд, сезонність і залишкову складову.

Трендова компонента відображає довгострокову зміну середнього значення часового ряду з часом. Вона дозволяє побачити загальний напрям розвитку показника. Сезонна складова демонструє повторювані закономірності у даних, які виникають з певною періодичністю. Наприклад, у випадку щоденного обліку споживання електроенергії, сезонність може проявлятися як регулярні піки у певні години доби.

Залишкова (резидуальна) компонента охоплює всі випадкові та нерегулярні коливання, які не пояснюються ні трендом, ні сезонністю. Візуалізація цих трьох складових допомагає краще зрозуміти структуру часового ряду та виявити приховані закономірності, що важливо для подальшого аналізу та побудови прогнозів. Результати такого розкладу наведені на рисунку 3.2.

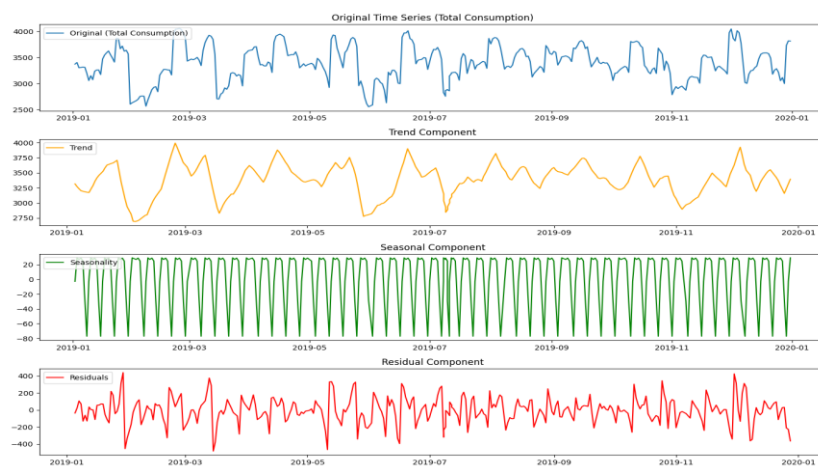


Рисунок 3.2 - Розклад часового ряду на складові.

Тест Дікі-Фуллера - це статистичний метод, який застосовується для визначення, чи є часовий ряд стаціонарним. Стаціонарність є важливою характеристикою, оскільки багато методів аналізу та побудови моделей ґрунтуються на припущенні, що дані не змінюють своїх статистичних властивостей з часом.

Стаціонарний часовий ряд має постійні характеристики, зокрема середнє значення та дисперсію, упродовж усього періоду спостережень. Урахування цієї властивості є критично важливим для побудови точних та надійних прогнозів. Якщо ж часовий ряд не є стаціонарним, це може призвести до хибних висновків та некоректного прогнозування. На рисунку 3.3 представлено результати тесту Дікі-Фуллера [22].

```

Augmented Dickey-Fuller Test Results:
Statistic: -6.250817323626135
p-value: 4.458769321447558e-08
Lags used: 1
Number of observations: 357
Critical Values:
  1%: -3.44880082033912
  5%: -2.869670179576637
 10%: -2.5711012838861036
Часовий ряд стаціонарний (відкидаємо нульову гіпотезу).

```

Рисунок 3.3 - Результати тесту Дікі-Фуллера.

Тестова статистика має від'ємне значення, що свідчить про стаціонарність ряду. Це також підтверджується дуже малим значенням p -value, яке наближене до нуля.

Низьке значення p -value вказує на відхилення нульової гіпотези щодо нестационарності ряду. Це свідчить про те, що досліджуваний часовий ряд є стаціонарним.

Кількість затримок, що використовувалися, та обсяг спостережень, які бралися до уваги, демонструють, скільки лагів застосовано в тесті та скільки даних було використано для розрахунку статистики.

Також подано критичні значення, які дозволяють оцінити, наскільки близько тестова статистика знаходиться до порогових значень на різних рівнях значущості. У цьому випадку тестова статистика є набагато меншою за критичні значення, що знову підтверджує стаціонарність ряду.

Отже, результати свідчать про те, що часовий ряд є стаціонарним, тобто не містить виражених трендів або сезонних коливань, які необхідно моделювати у рамках ARIMA. У зв'язку з цим додаткове диференціювання не буде використовуватися, а сезонність можна врахувати в інших моделях, таких як SARIMA чи ARIMAX.

Наступний крок це побудова графік автокореляції і часткової автокореляції. Графіки ACF(Autocorrelation Function) та PACF(Partial Autocorrelation Function) будуються для аналізу автокореляційних зв'язків у часовому ряді. Це показано на рисунках 3.4 і 3.5.

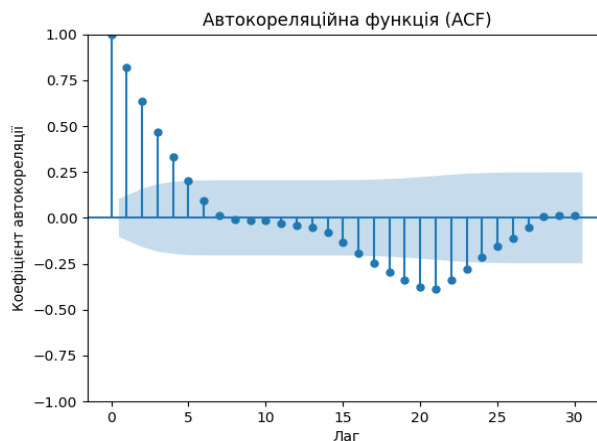


Рисунок 3.4. - Графік автокореляції.

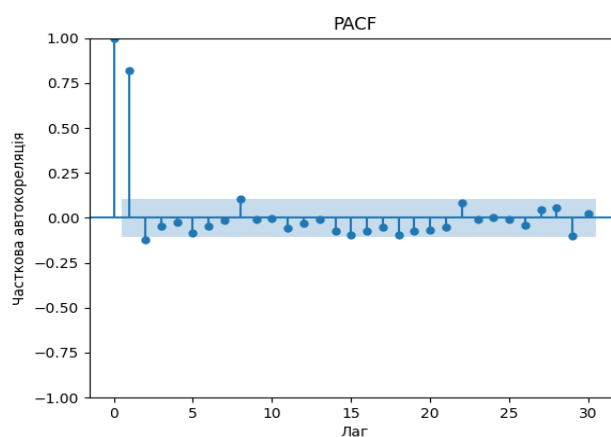


Рисунок 3.5. - Графік часткової автокореляційної функції.

Основна мета побудови графіків полягає у виявленні кореляцій між значеннями часового ряду на різних часових затримках. Графік ACF показує кореляцію між поточними значеннями ряду та його затриманими копіями на різних проміжках часу. Кожна точка на графіку відображає кореляцію між значеннями ряду в даний момент та значеннями на певній кількості попередніх моментів часу. Графік ACF допомагає виявити сезонні патерни та кореляційні структури в ряді. Графік PACF показує часткову кореляцію між значеннями часового ряду та його відстроченими копіями, при цьому контролюється вплив значень на всіх коротших затримках. Іншими словами, PACF відображає кореляцію між значеннями ряду на конкретній затримці, враховуючи вплив

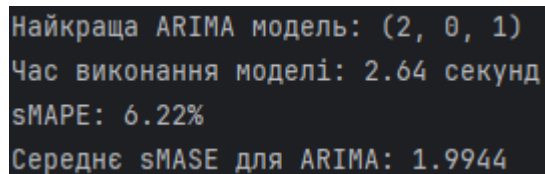
інших затримок. Цей графік допомагає визначити лаги, які значно впливають на поточні значення ряду та сприяє вибору моделі для ARIMA.

3.2 Прогнозування за допомогою ARIMA

Для вибору найкращої моделі ARIMA для прогнозування часового ряду необхідно визначити оптимальні параметри моделі.

Метою є знаходження такої моделі, яка найточніше відображає поведінку конкретного часового ряду. ARIMA-модель характеризується трьома параметрами (p , d , q), які задають її структуру. Підбір найбільш відповідних значень цих параметрів дозволяє підвищити точність прогнозування.

Оптимізація параметрів допомагає уникнути проблем перенавчання або недонавчання, забезпечуючи баланс між складністю моделі та її узагальнювальною здатністю. Правильний вибір параметрів робить модель більш адаптованою до структури часового ряду, що сприяє побудові точніших прогнозів. Застосування оптимальних параметрів дає змогу побудувати модель, яка ефективно використовує доступні дані та ресурси для передбачення майбутніх значень ряду. Результат запуску моделі рисунок 3.6 і 3.7.



```
Найкраща ARIMA модель: (2, 0, 1)
Час виконання моделі: 2.64 секунд
sMAPE: 6.22%
Середнє sMASE для ARIMA: 1.9944
```

Рисунок 3.6 - Найкращі параметри.



Рисунок 3.7. - Результат прогнозування ARIMA.

Таким чином, налаштування параметрів моделі ARIMA є ключовим етапом у побудові ефективних прогностичних моделей для аналізу та передбачення часових рядів.

На початковому етапі дані поділяються на навчальну та тестову вибірки. Далі за допомогою циклічного перебору виконуються варіації комбінацій параметрів моделі ARIMA. Для кожної з них модель навчається на навчальному наборі, після чого здійснюється прогнозування на тестовому.

Оцінка якості прогнозу проводиться за допомогою показника symmetric mean absolute percentage error (sMAPE). На основі значень цієї метрики обирається найефективніша модель. Формула, яка використовується для обчислення метрики, наведена нижче.

$$sMAPE = n * \sum t = \ln(|At| + |Ft|) / 2|Ft - At| \quad (3.1)$$

де: At - фактичне значення в часовій точці t , Ft - прогнозоване значення в часовій точці t , n - кількість точок у тестовому наборі.

Оптимальні параметри, визначені в процесі підбору, застосовуються для побудови остаточної моделі ARIMA, яка навчається на повному обсязі даних (включаючи як навчальний, так і тестовий періоди) та використовується для прогнозування значень у межах тестового інтервалу.

У результаті формується найточніша модель ARIMA для заданого часового ряду, разом з її характеристиками, такими як коефіцієнти та статистичні оцінки.

Згідно з проведеною оптимізацією, найкраща модель має параметри $(p, d, q) = (2, 0, 1)$. Проведений аналіз дозволяє зробити висновок, що модель ARIMA з параметрами $(2, 0, 1)$ є найбільш ефективною серед протестованих варіантів для даного часового ряду.

Побудова гістограми розподілу значень метрики sMASE для кожного часового ряду є ключовим етапом у аналізі якості прогностичної моделі.

Положення найвищої точки гістограми часто відображає середнє значення sMASE для вибірки. Чим вище пік, тим більше часових рядів мають значення sMASE, близьке до цього середнього.

Форма та ширина гістограми дозволяють оцінити варіативність значень sMASE. Широкий розподіл із великою дисперсією свідчить про те, що точність моделі суттєво змінюється між різними часовими рядами.

Симетричність чи асиметричність розподілу на графіку вказує, наскільки однорідними є результати прогнозування: симетрична форма означає приблизно однакову точність для більшості рядів, тоді як асиметрія може свідчити про наявність рядів із суттєво кращими або гіршими результатами.

Висота кожного стовпчика на гістограмі демонструє кількість часових рядів, що мають sMASE у відповідному діапазоні: чим вищий стовпець, тим більше таких рядів. Гістограма наведена на рисунку 3.8.

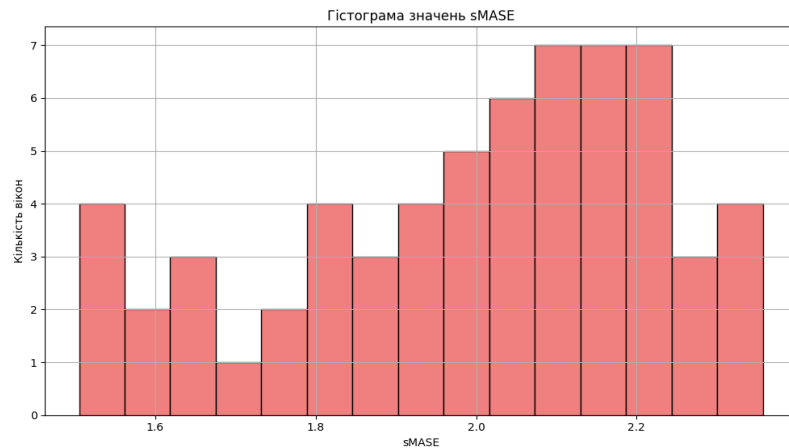


Рисунок 3.8 - Гістограма значень sMAPE.

У цілому, результати дослідження суперечать теоретичним міркуванням. Бо за проведеними вище тестами за датасетом ARIMA мала б краще працювати. Це відбувається через те що в датасеті присутня сезонність.

3.3 Прогнозування за допомогою SARIMA

У моделі SARIMA було застосовано фіксовані параметри - SARIMA(1,1,1)(0,1,1)[24]. Такий вибір обумовлений тим, що подібна конфігурація зазвичай забезпечує прийнятну точність для багатьох типів часових рядів. Хоча підбір оптимальних параметрів може підвищити якість прогнозу, він потребує значних обчислювальних ресурсів і значно збільшує час навчання моделі, результат показано на рисунку 3.9.

```
SARIMA(1, 1, 1)x(1, 1, 1, 24) | AIC = 1716.88
Час підбору параметрів SARIMA: 16.14 секунд

Найкращі параметри SARIMA:
order=(1, 1, 1), seasonal_order=(0, 1, 1, 24), AIC=1715.14
sMAPE для SARIMA: 6.22%
Середнє sMAPE для SARIMA: 0.3918
```

Рисунок 3.9 - Найкращі параметри.

На наступному графіку наведено порівняння фактичних даних із прогнозом, побудованим за моделлю SARIMA. Синя лінія відображає реальні значення, тоді як червона - прогноз моделі.. Аналізуючи цей графік разом із відповідними графіком моделі ARIMA, можна оцінити, наскільки ефективно

модель SARIMA відтворює реальні дані у порівнянні з альтернативним підходам. Графічне зображення подано на рисунку 3.10.

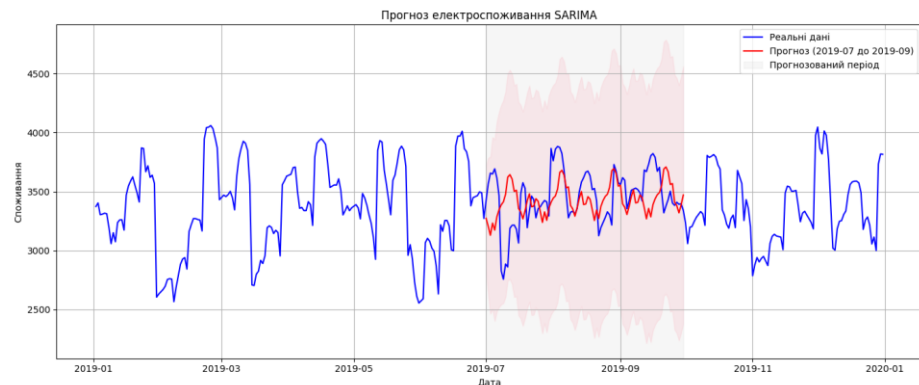


Рисунок 3.10 - Графік прогнозування моделі SARIMA.

Якщо червона лінія, що відповідає прогнозу SARIMA, тісно повторює синю лінію фактичних даних, це свідчить про хорошу відповідність моделі реальному часовому ряду та її потенційну ефективність у прогнозуванні.

Досить близьке накладання фактичних (синя лінія) та прогнозованих (червона лінія) значень свідчить про те, що модель добре захоплює динаміку та тенденції даних. У такому випадку модель SARIMA показує високу здатність до відтворення характеристик досліджуваного часового ряду, що робить її доцільною для подальшого використання в прогнозуванні.

Наступним етапом аналізу є розгляд гістограми розподілу метрики sMASE. Порівнюючи гістограму для SARIMA з відповідними гістограмами для ARIMA, можна отримати уявлення про якість прогнозу кожної з моделей.

Гістограма sMASE відображає розподіл відносних помилок прогнозу для кожного часового ряду в датасеті. Це дозволяє оцінити, наскільки точно модель SARIMA працює порівняно з іншими підходами. Якщо форма або характер розподілу гістограми для SARIMA подібні до ARIMA, це свідчитиме про подібний рівень точності. Водночас, відмінності в розподілах можуть вказувати як на перевагу, так і на недоліки SARIMA у порівнянні з

альтернативами. В нашому випадку SARIMA показала себе значно краще за ARIMA і це підтверджується гістограмами. Гістограма представлена на рисунку 3.11.

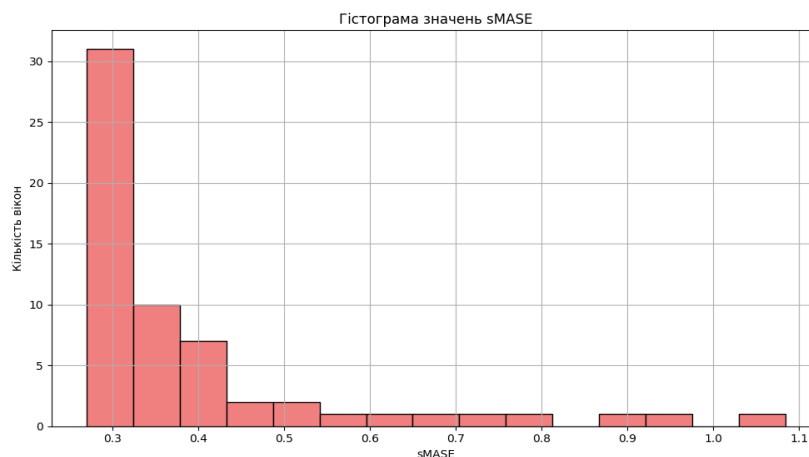


Рисунок 3.11 - Гістограма значень sMESA.

3.4 Прогнозування за допомогою ARIMAX

Виконуємо такі самі дії як і для моделі ARIMA. Щоб порівняти результати ARIMAX з ARIMA обираємо найкращу модель. У нашому випадку найкраща модель ARIMAX(1,0,2). Для порівняння в ARIMA(2,0,1).

Параметри ARIMAX включають коефіцієнти для днів тижня, що вказує на використання екзогенних змінних у моделі, а в ARIMA таких змінних не враховувалось. Це може вказувати на те що дні в тижні мають велике значення для прогнозування енергоспоживання, результат показано рисунку 3.12.

```
Час виконання пошуку найкращих параметрів ARIMAX: 3.08 секунд
Найкраща ARIMAX модель: (1, 0, 2)
sMAPE: 6.78%
Середнє sMAPE для ARIMAX: 2.0634
```

Рисунок 3.12 - Найкраща модель ARIMAX.

Наступний крок побудувати графік що показує фактичні та прогнозовані значення часового ряду для моделі ARIMAX. Тут вже з першого погляду можна сказати що ця модель краще спрогнозувала ніж ARIMA. Це відбувається через те що модель враховує екзогенні змінні, що робить прогнози більш адаптивними до змін у даних і це в свою чергу робить моделювання більш складних залежностей у часовому ряді більш легким. Синя лінія відповідно це реальні дані, червона прогнозовані, рисунок 3.13.



Рисунок 3.13 - Графік прогнозування моделі ARIMAX.

Гістограма розподілу значень метрики sMASE для ARIMAX відображає частоту з якою різні значення sMASE зустрічаються серед прогнозів моделі ARIMAX для кожного часового ряду.

Вісь x гістограми показує значення метрики яка є показником точності прогнозів. Чим менше значення тим краще прогнози моделі. Вісь y показує кількість разів коли значення метрики потрапило в певний діапазон. Чим вищий стовпець на гістограмі тим більше прогнозів отримали значення що відповідає цьому діапазону. Це можна побачити на рисунку 3.14.

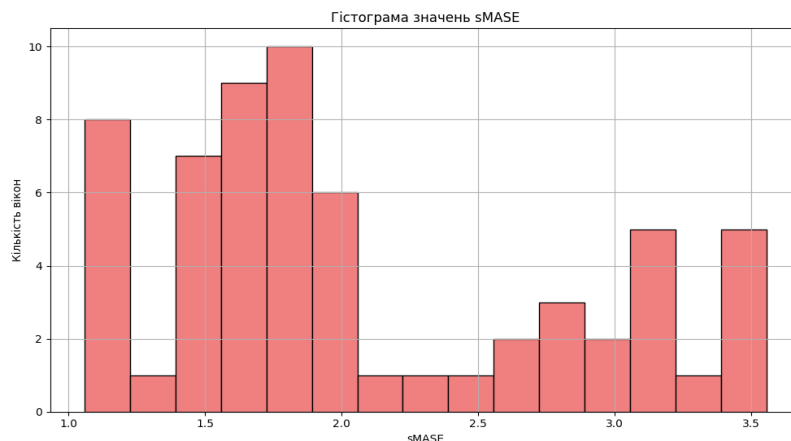


Рисунок 3.14 - Гістограма значень sMASE.

Відмінність між гістограмами для моделей ARIMAX і ARIMA може проявлятися в характері розподілу значень метрики sMASE. Оскільки ARIMAX враховує зовнішні екзогенні чинники, наприклад, погодні умови чи святкові дні, можна припустити, що це вплине на розподіл sMASE. Якщо ARIMAX краще адаптується до таких факторів, ймовірно, її значення sMASE будуть ближчими до нуля, що призведе до концентрації значень у нижній частині гістограми, на відміну від ARIMA.

3.5 Прогнозування за допомогою SARIMAX

Наступний графік ілюструє як реальні значення, так і прогноз, побудований за допомогою моделі SARIMAX(1,1,1)(0,1,1)(24).

Порівнюючи цей графік із відповідними візуалізаціями для моделей ARIMA, ARIMAX і SARIMA, можна помітити як схожі риси, так і розбіжності у прогностичних значеннях. Зокрема, червона лінія, що відображає прогноз SARIMAX, може добре наближатися до фактичних даних у разі вдалого моделювання. Це дає змогу оцінити, чи покращує SARIMAX точність прогнозу порівняно з іншими моделями.

Такий результат може свідчити про високу ефективність моделі SARIMAX для передбачення майбутніх значень часового ряду. Найкращу модель наведено на рисунку 3.15 графік наведено на рисунку 3.16.

```

SARIMAX(1, 1, 1)x(1, 1, 0, 24) | AIC = 1685.04
SARIMAX(1, 1, 1)x(1, 1, 1, 24) | AIC = 1651.16

Найкращі параметри SARIMAX:
order=(1, 1, 1), seasonal_order=(0, 1, 1, 24), AIC=1649.16
Час виконання підбору параметрів і тренування моделі: 26.11 секунд
sMAPE для SARIMAX: 5.78%
Середнє sMAPE для SARIMAX: 0.7823

```

Рисунок 3.15 - Найкраща модель SARIMAX.

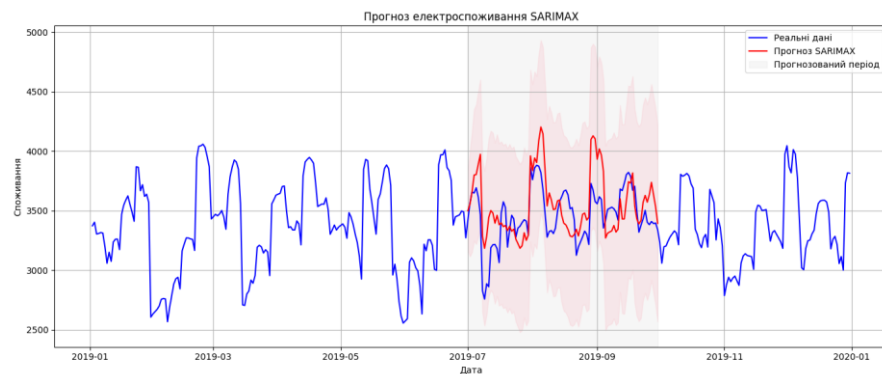


Рисунок 3.16 - Прогнозування моделі SARIMAX.

Гістограма значень метрики sMAPE для моделі SARIMAX дає змогу оцінити точність її прогнозів у порівнянні з іншими розглянутими моделями.

Якщо на гістограмі спостерігається пік поблизу нуля або зосередженість значень біля низьких значень метрики, це свідчить про високу точність більшості прогнозів і незначні помилки, що є позитивною ознакою ефективності моделі SARIMAX у задачах прогнозування.

Навпаки, широкий пік або висока щільність значень у зоні великих значень sMAPE вказує на значні похибки в прогнозах, що може свідчити про недостатню ефективність моделі або про те, що вона не враховує важливі характеристики даних. Візуалізацію представлено на рисунку 3.17.

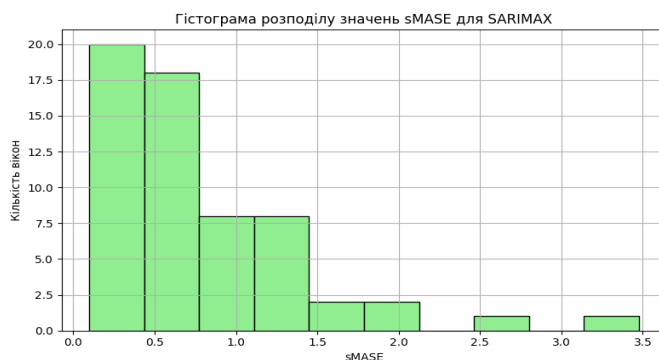


Рисунок 3.17 - Гістограма sMASE.

3.6 Ансамблювання як спосіб покращити точність

Під час дослідження було виявлено, що в розглянутих моделях є свої переваги та недоліки при використанні їх для обраного датасету. Провівши дослідження можна підбити підсумки про ефективність кожної з розглянутих моделей.

Таблиця 3.1.

Результати прогнозування для кожної моделі

Метод	sMAPE	sMASE
SARIMA	6.22%	0.3918
SARIMAX	5.78%	0.7823
ARIMA	6.22%	1.9944
ARIMAX	6.78%	2.0634

Виходячі з таблиці 3.1 найкращий результат показали моделі SARIMA і SARIMAX. Ці моделі можна назвати класичними, вони вже довели свою ефективність в прогнозуванні часових рядів. Дані моделі добре пристосовані для моделювання та прогнозування часових рядів з різними сезонними та

трендовими компонентами. Обидва методи мають достатній широкий спектр застосування і можуть бути успішно використані у багатьох ситуаціях.

Таким чином, враховуючи успішність цих методів в окремоті обираємо їх для подальшого дослідження з ансамблями, оскільки можна очікувати що їх комбінація може призвести до покращення точності та надійності прогнозів. Крім того моделі мають свої особливості які можуть доповнити одна одну в ансамблі, що робить їх привабливим вибором для дослідження.

SARIMA може добре працювати з регулярними сезонними змінами в той час як SARIMAX може краще враховувати вплив екзогенних змінних на часовий ряд. Однак в кожній моделі може бути деяка ступінь невідомості щодо їхньої точності або ефективності в конкретній ситуації.

Основна мета ансамблювання цих моделей полягає в використанні їх комбінованої сили для отримання більш точних та надійних прогнозів. Шляхом комбінування результатів кількох моделей можна спробувати зменшити можливі помилки та недоліки кожної окремої моделі, забезпечивши таким чином більш надійні прогнози на основі колективних результатів моделей.

Ситуація з прогнозуванням часових рядів електроенергії по штатах Індії - ідея ансамблювання моделей має кілька суттєвих переваг.

Перше, це зменшення дисперсії. Кожна модель як вже говорилося вище має свої сильні і слабкі сторони тому ансамблювання дозволяє скористатись перевагами кожної сильної сторони а також зменшити вплив недоліків, що можуть виникати при використанні лише однієї моделі. Такий підхід може дати більш стабільні і надійні прогнози.

Друге, покращення точності. Досить часто різні моделі можуть дати різні прогнози і деякі з них можуть бути ближчими до правдивих значень ніж інші. Ансамблювання дозволяє об'єднати ці різні прогнози та отримати більш точний результат оскільки воно враховує різноманітність підходів.

Третє, зниження ризику перенавчання. Коли ми користуємось тільки однією моделлю для прогнозування то існує ризик перенавчання на конкретних особливостях тренувальних даних що може призвести до поганої узагальнювальної здатності моделі на нових даних. Ансамблювання в свою чергу допомагає зменшити такий ризик, тому що воно використовує кілька моделей з різними підходами що може покращити узагальненість прогнозів.

Четверте і останнє, робота зі змінною природою даних. Для складання часових рядів таких як використання електроенергії, характеристики можуть змінюватись з часом або бути сезонними. Різні моделі можуть краще адаптуватись до цих змін. Ансамблювання дозволяє комбінувати ці різні адаптивності моделей для отримання більш точних прогнозів навіть у змінних умовах.

Підбиваючи все вище сказане, ансамблювання для прогнозування часових рядів може бути ефективним способом отримання точних та стабільних прогнозів у ситуаціях де одна модель може бути недостатньою в конкретному контексті прогнозування часових рядів електроенергії [23].

3.7 Ансамблювання методом бустингу

Ансамблювання методом бустингу є популярним підходом у машинному навчанні для покращення точності прогнозів. Цей метод полягає у поетапному покращенні прогнозу за допомогою кількох простих моделей, де кожна наступна модель намагається виправити помилки попередньої. У контексті прогнозування часових рядів, бустинг може бути використаний для об'єднання прогнозів різних моделей, таких як SARIMA та SARIMAX, щоб досягти більш точного результату.

Бустинг - це ансамблевий метод машинного навчання, який використовує слабкі моделі для створення потужної прогнозуючої системи. Слабка модель - це модель, що тільки трохи краща за випадкове вгадування. Бустинг покращує її точність, зменшуючи помилки через повторне навчання на основі попередніх помилок.

В бустингу моделі створюються послідовно, при цьому кожна нова модель намагається виправити помилки, зроблені попередніми моделями. Найвідоміші алгоритми бустингу включають: AdaBoost (Adaptive Boosting), Gradient Boosting, XGBoost

Було прийнято рішення використовувати XGBoost, що є однією з найпопулярніших реалізацій бустингу, зокрема для регресії та класифікації. XGBoost використовує градієнтний бустинг, де кожен новий класифікатор намагається мінімізувати похибку попередніх класифікаторів.

У задачі прогнозування споживання електроенергії, де використовуються моделі SARIMA та SARIMAX, бустинг дозволяє комбінувати прогнози, отримані з цих моделей, для покращення точності.

Використання бустингу для ансамблювання цих двох моделей дозволяє отримати більш точний прогноз, оскільки він коригує похибки окремих моделей, використовуючи додаткові ознаки та виправлення помилок.

В нашому випадку XGBoost для ансамблювання прогнозів SARIMA та SARIMAX. Основні етапи цього процесу:

Підготовка даних - об'єднується прогнози з SARIMA та SARIMAX у єдиний DataFrame, створюючи ознаки для моделі бустингу.

Навчання моделі - модель XGBoost навчається на цих ознаках (прогнозах), щоб знайти найкращу комбінацію для поліпшення точності прогнозу. Прогнозування - модель XGBoost генерує фінальний прогноз на основі отриманих ознак.

Ансамблювання методом бустингу є потужним підходом для покращення точності прогнозів у задачах, де використовуються кілька моделей для прогнозування часових рядів. Це дозволяє комбінувати прогнози з різних моделей, виправляючи їх помилки і забезпечуючи більш точний результат. В нашому випадку, використання бустингу для поєднання прогнозів моделей SARIMA та SARIMAX дозволяє досягти більш надійних і точних прогнозів для споживання електроенергії.

Цей метод є важливим інструментом у прогнозуванні та застосовується в різноманітних галузях, де точність є критично важливою, наприклад, в енергетиці, фінансах та маркетингу, результати на рисунках 3.18 і 3.19 [23].

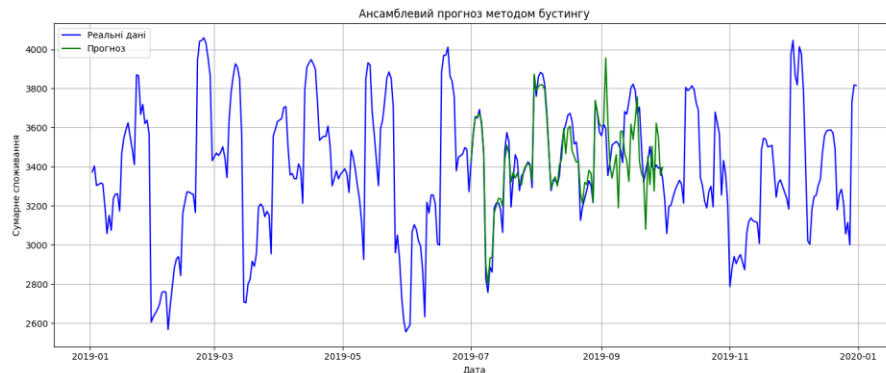


Рисунок 3.18 - Результат ансамблювання.

```
sMAPE Train: 1.31%
sMAPE Test: 4.74%
sMASE Train: 0.3960
sMASE Test: 2.3880
```

Рисунок 3.19 - Похибки ансамблювання.

Якість прогнозу на тренувальній вибірці є високою. Згідно з показником $sMAPE\ Train = 1.31\%$, модель дуже точно описує тренувальні дані. Це свідчить про те, що ансамбль методів бустингу добре навчився на наданих даних. Прогноз на тестовій вибірці також доволі точний. $sMAPE\ Test = 4.74\%$ - це прийнятне значення для задач прогнозування часового ряду, що свідчить про здатність моделі адекватно узагальнювати нові, ще не бачені дані. Абсолютна похибка на тренувальних та тестових даних демонструє стабільну продуктивність моделі. Значення $sMASE\ Train = 0.3960$ та $sMASE\ Test = 2.3880$ вказують, що хоча точність зменшується на тестовій вибірці, модель усе ще забезпечує значно кращий результат, ніж наївні методи прогнозу.

Метод ансамблювання з бустингом продемонстрував ефективність. Результати свідчать, що використання ансамблю дозволяє значно підвищити точність прогнозу у порівнянні з базовими моделями, зберігаючи баланс між точністю на тренуванні та узагальненням на тесті.

3.8 Ансамблювання методом бегінгу

Метод бегінгу полягає у побудові кількох прогнозних моделей і подальшому агрегуванні їх результатів. У випадку регресійних задач, як-от прогнозування електроспоживання, агрегування зазвичай здійснюється шляхом усереднення числових прогнозів кожної з моделей:

$$y_e(t) = 1/M \sum_{i=1}^M y_i(t) \quad (3.2)$$

де: $y_i(t)$ -прогноз у момент часу t отриманий i -ю моделлю, M -кількість моделей на ансамблі, $y_e(t)$ -агрегований ансамблевий прогноз.

Цей підхід добре працює, якщо моделі є достатньо незалежними і помилки мають різну природу, що дозволяє взаємно компенсувати похибки.

Переваги бегінгу: зменшення дисперсії прогнозу, стабільніші результати, особливо при наявності шуму в даних, покращення точності у порівнянні з окремими моделями.

Прогнози з кожної моделі зчитуються з CSV-файлів та поелементно усереднюються для отримання єдиного ансамблевого прогнозу.

Для оцінки якості прогнозу ансамблю використано дві метрики:

sMAPE - симетрична середня абсолютна відносна помилка:

$$sMAPE = 100\%/n \sum_{t=1}^n (2 * |\hat{y}_t - y_t|) / (|\hat{y}_t| + |y_t|) \quad (3.3)$$

sMASE - відношення середньої абсолютної помилки моделі до помилки наївного прогнозу:

$$sMASE = MAE_{\text{модель}} / MAE_{\text{наївна}} \quad (3.4)$$

Ці метрики дають змогу порівнювати продуктивність моделі незалежно від масштабів вихідного ряду. Після побудови ансамблевого прогнозу здійснюється його візуалізація разом із реальними даними та виділенням прогнозованого періоду. Таке представлення дозволяє наочно оцінити якість передбачення та відповідність моделі трендам і сезонним коливанням у часовому ряді, результати на рисунках 3.20 і 3.21 [23].

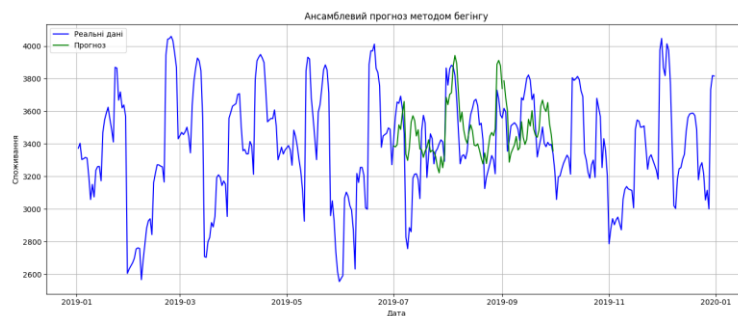


Рисунок 3.20 - Результат ансамблювання.

```
sMAPE (Train): 5.65%
sMAPE (Test): 4.66%
sMASE (Train): 1.7187
sMASE (Test): 2.3678
```

Рисунок 3.21 - Похибки ансамблювання.

Точність прогнозу на тренувальній вибірці є нижчою порівняно з бустингом. $sMAPE_{\text{Train}} = 5.65\%$ вказує на те, що модель не настільки точно апроксимує тренувальні дані. Це може свідчити про меншу гнучкість моделі бегінгу або про його менш виражену здатність до адаптації до специфіки тренувальної вибірки.

Якість прогнозу на тестовій вибірці є прийнятною. $sMAPE\ Test = 4.66\%$ залишається на хорошому рівні, близькому до результату бустингу (4.74%), що свідчить про певну стабільність моделі та здатність до узагальнення. Абсолютна похибка вища порівняно з бустингом. $sMASE\ Train = 1.7187$ та $sMASE\ Test = 2.3678$ свідчать про те, що абсолютна помилка як на тренувальній, так і на тестовій вибірці вища, ніж у попередньому методі. Це свідчить про меншу ефективність бегінгу в даній задачі.

Модель бегінгу має кращу генералізацію, але гіршу апроксимацію. Можна зробити висновок, що бегінг забезпечує дещо кращу узагальнюваність, адже його $sMAPE$ на тесті навіть трохи нижчий, ніж на тренуванні. Водночас висока помилка на тренуванні свідчить про недостатню адаптацію до навчальних даних.

Загалом, метод бегінгу показує стабільність, але поступається бустингу за точністю. Попри непогані результати на тесті, метод поступається бустингу як за відносною ($sMAPE$), так і абсолютною ($sMASE$) точністю. Це робить його менш бажаним вибором для задачі з високими вимогами до точності прогнозу.

Висновок до розділу 3

Метод бустингу продемонстрував вищу точність прогнозування. На основі показників похибок видно, що бустинг досяг значно нижчої відносної помилки на тренувальній вибірці ($sMAPE\ Train = 1.31\%$) порівняно з бегінгом ($sMAPE\ Train = 5.65\%$). Це свідчить про кращу здатність моделі бустингу до точного відтворення тренувальних даних. Тестові результати обох моделей є подібними за $sMAPE$, але різняться за $sMASE$: $sMAPE\ Test$ у бустингу - 4.74% , $sMAPE\ Test$ у бегінгу - 4.66% . Хоча ці значення близькі, абсолютна помилка ($sMASE$) на тесті у бустингу (2.3880) вища, ніж у бегінгу (2.3678), однак

різниця не є суттєвою. Це означає, що обидві моделі мають порівняну якість генералізації на нові дані, хоча бустинг трохи точніший у відносному вимірі.

Бустинг краще відтворює складні коливання у часових рядах. З графіків видно, що прогнозна лінія для бустингу краще повторює форму реального часового ряду, включаючи піки та падіння. Бегінг має тенденцію до більш згладженого прогнозу, що може знижувати точність у нестабільних ділянках ряду.

Бегінг виявився менш схильним до перенавчання. На відміну від бустингу, де помилка на тренуванні дуже мала (можливе перенавчання), бегінг має приблизно однакові значення sMAPE для train і test. Це свідчить про хорошу здатність моделі бегінгу узагальнювати, хоча й за рахунок меншої точності.

Обидва методи показали ефективність, але з різними акцентами. Бустинг - кращий вибір для задач, де критично важлива точність прогнозу. Бегінг - доцільний у задачах, де важлива стабільність моделі та зменшення ризику перенавчання.

ВИСНОВКИ

За результатами виконаного дослідження проведено аналіз і порівняння методів прогнозування часових рядів в контексті споживання електроенергії. Методи ARIMA, SARIMA, ARIMAX, SARIMAX та ансамблюючі моделі SARIMA, SARIMAX були застосовані для задачі прогнозування споживання електроенергії. Сформульовано переваги та недоліки розглянутих методів в залежності від конкретних умов і характеристик досліджуваного часового ряду. Як приклад ARIMA, ARIMAX продемонстрували низьку точність у прогнозуванні часового ряду споживання електроенергії, а використання SARIMA, SARIMAX ефективно при наявності сезонності та трендів.

Розглядаючи моделі SARIMA, SARIMAX в контексті прогнозування споживання електроенергії і використовуючи ансамблювання можна значно покращити результати прогнозування шляхом комбінування сильних сторін окремих моделей та зменшення впливу їхніх недоліків.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. The United Nations publishes new world population projections [Електронний ресурс] Режим доступу:
https://www.ined.fr/en/everything_about_population/demographic-facts-sheets/focus-on/2024-les-nations-unies-publient-de-nouvelles-projections-de-population-mondiale. Дата звернення: 27.12.2024
2. Executive summery [Електронний ресурс] Режим доступу:
<https://www.iea.org/reports/electricity-2025/executive-summary>. Дата звернення: 2.01.2025
3. The importance of accurate load forecasting [Електронний ресурс] Режим доступу:
<https://www.net2grid.com/post/the-importance-of-accurate-load-forecasting>. Дата звернення: 9.01.2025
4. Federal Financial Interventions and Subsidies in Energy Markets 2007 28-35 с. Дата звернення: 14.01.2025
5. Peak Electrical Energy Consumption Prediction by ARIMA, LSTM, GRU, ARIMA-LSTM and ARIMA-GRU Approaches [Електронний ресурс] Режим доступу:
<https://www.mdpi.com/1996-1073/16/12/4739>. Дата звернення: 21.01.2025
6. Using the LSTM Network to Forecast the Demand for Electricity in Poland [Електронний ресурс] Режим доступу: <https://www.mdpi.com/2076-3417/10/23/8455>. Дата звернення: 28.01.2025
7. Winter Outlook 2024-2-25 2025 7-18 с. Дата звернення: 29.01.2025
8. Загальна лінійна модель [Електронний ресурс] Режим доступу:
https://uk.wikipedia.org/wiki/%D0%97%D0%B0%D0%B3%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0_%D0%BB%D1%96%D0%BD%D1%96%D0%B9%D0%BD%D0%B0_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C.
 Дата звернення: 5.02.2025
9. Rob J Hyndman, George Athanasopoulos Forecasting Principles and Practic 2016 180-220 с. Дата звернення: 7.02.2025

10. Comparative analysis of ARIMA and LSTM for predicting fluctuating time series data [Электронный ресурс] Режим доступа: https://www.researchgate.net/publication/381072094_Comparative_analysis_of_ARIMA_and_LSTM_for_predicting_fluctuating_time_series_data. Дата звернення: 11.02.2025
11. Time Series Forecasting using LSTM and ARIMA [Электронный ресурс] Режим доступа: https://www.researchgate.net/publication/367986282_Time_Series_Forecasting_using_LSTM_and_ARIMA. Дата звернення: 15.02.2025
12. Demand response complements energy efficiency in lowering costs & emissions, Berkeley Lab and NREL find [Электронный ресурс] Режим доступа: <https://emp.lbl.gov/news/demand-response-complements-energy>. Дата звернення: 22.02.2025
13. The Journey into Time Series: Dancing with Trends and Seasons [Электронный ресурс] Режим доступа: <https://medium.com/donato-story/the-journey-into-time-series-dancing-with-trends-and-seasons-d8b635f8fa3c>. Дата звернення: 2.03.2025
14. Shumway, Robert H., Davis S., Stoffer, Time Series Analysis and Its Applications: With R Examples 2017 17-180 с. Дата звернення: 10.03.2025
15. Michael Gilliland, Len Tashman, Udo Sglavo, Business Forecasting: Practical Problems and Solutions 2010 3-14 с. Дата звернення: 20.03.2025
16. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction 2009 9-99 с. Дата звернення: 4.04.2025
17. Michael Gilliland, Len Tashman, Udo Sglavo, Machine Learning for Time Series Forecasting with Python 2020 1-42 с. Дата звернення: 6.04.2025
18. Jonathan D. Cryer, Kung-Sik Chan, Time Series Analysis With Applications in R Second Edition 2008 92-97 с. Дата звернення: 9.04.2025
19. Forecasting: Principles and Practice [Электронный ресурс] <https://otexts.com/fpp3/what-can-be-forecast.html> Дата звернення: 11.04.2025

20. Aileen Nielsen, Practical Time Series Analysis: Prediction with Statistics and Machine Learning 2019 131-160 с. Дата звернення: 15.04.2025
21. Kaggle.com [Електронний ресурс] <https://www.kaggle.com> Дата звернення: 12.04.2025
22. Paul S.P. Cowpertwait, Andrew V. Metcalfe, Practical Time Series Analysis: Introductory Time Series with R 2019 77-92 с. Дата звернення: 17.04.2025
23. Zhi-Hua Zhou, Ensemble Methods: Foundations and Algorithms 2012 13-80 с. Дата звернення: 19.04.2025

ДОДАТКИ

Додаток А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет імені В. Н. Каразіна

Навчально-науковий інститут комп'ютерних наук та штучного інтелекту
Кафедра комп'ютерних систем та робототехніки
Рівень вищої освіти (освітньо-кваліфікаційний рівень) **Бакалавр**
Галузь знань: 12 – Інформаційні технології
Спеціальність: 123 «Комп'ютерна інженерія»
Освітня програма «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри комп'ютерних
систем та робототехніки
к. ф.-м. н., доц. ХРУСЛОВ М. М.
«02» жовтня 2024 року

З А В Д А Н Н Я НА КВАЛІФІКАЦІЙНУ РОБОТУ

Хавроня Артем Родіонович
(прізвище, ім'я, по батькові студента)

1. Тема роботи «Комп'ютерна модель прогнозування
споживання електроенергії за допомогою методів аналізу часових рядів»

керівник роботи кандидат технічних наук, доцент Бакуменко Ніна Станіславівна
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від **16.04.2025 року № 4101-5/962**

2. Строк подання студентом роботи **30 травня 2025 року**

3. Перелік питань, які потрібно розробити)

1. Аналіз основних методів аналізу часових рядів для прогнозування споживання електроенергії.
2. Огляд існуючих підходів до прогнозування електроспоживання (класичні статистичні методи, машинне навчання, гібридні моделі).
3. Дослідження впливу сезонності, трендів та випадкових факторів на електроспоживання.
4. Аналіз популярних алгоритмів прогнозування: ARIMA, SARIMA, ARIMAX, SARIMAX, ансамблювання та їх застосування.
5. Вибір оптимального методу прогнозування для конкретного набору даних.
6. Розробка комп'ютерної моделі прогнозування споживання електроенергії.
7. Реалізація моделі в середовищі програмування Python.
8. Тестування та оцінка точності прогнозу за допомогою показників sMAPE, sMAE.
9. Оптимізація моделі та порівняння її ефективності з існуючими методами.
10. Аналіз програмних рішень та бібліотек для прогнозування часових рядів.

4. План роботи

№ з/п	Назви етапів роботи	Термін виконання етапів роботи
1	Затвердження теми роботи	05.09.2024 – 02.10.2024
2	Аналіз та пошук методичної літератури щодо прогнозування споживання електроенергії	02.10.2024 – 24.10.2024
3	Огляд та аналіз методів аналізу часових рядів	25.10.2024 – 09.11.2024
4	Вибір алгоритмів прогнозування електроспоживання	10.11.2024 – 24.11.2024
5	Розробка математичної моделі прогнозування	25.11.2024 – 08.12.2024
6	Реалізація моделі на основі вибраного алгоритму	09.12.2024 – 29.12.2024
7	Тестування моделі, оцінка точності прогнозування	30.12.2024 – 15.01.2025
8	Оптимізація параметрів моделі, порівняльний аналіз	16.01.2025 – 01.02.2025
9	Аналіз програмних рішень для прогнозування часових рядів	02.02.2025 – 20.02.2025
10	Підготовка та оформлення звітних матеріалів	21.02.2025 – 30.04.2025
11	Оформлення пояснювальної записки	01.05.2025 – 30.05.2025
12	Захист кваліфікаційної роботи	Червень 2025

5. Дата видачі завдання *02 жовтня 2024 року.*

Студент

Хавроня А. Р.
ініціали, прізвище


(підпис)

Керівник роботи

Бакуменко Н. С.
ініціали, прізвище


підпис

Додаток Б

Технічне завдання
на розробку програмного виробу

Назва розділу	Назва і зміст підрозділу
1. Введення	1.1. Назва програмного виробу «Комп'ютерна модель прогнозування споживання електроенергії за допомогою методів аналізу часових рядів» 1.2. Галузь застосування: електроенергетика
2. Підстава для розробки	2.1. Навчальний план за спеціальністю 123 – «Комп'ютерна інженерія». 2.2. Завдання на кваліфікаційну роботу бакалавра затверджено наказом № <u>4101-5/962</u> _____;
3. Призначення розробки	3.1. Мета розробки програмного виробу є подальше використання цього виробу у енергетичній сфері для покращення та полегшення процесу прогнозування споживання електроенергії. 3.2. Призначення програмного виробу: для використання в енергетиці. 3.3. Вихідні дані для розробки: графік споживання електроенергії.
4. Технічні вимоги програмного виробу	Програма повинна: 1) Представляти з себе комп'ютерну реалізацію моделі прогнозування споживання електроенергії; 2) Забезпечити можливість завантаження нових даних, для процесу прогнозування; 3) Забезпечувати можливість виведення результату на екран за допомогою інтуїтивно зрозумілого інтерфейсу; 5) Для виконання програми необхідний ПК з ОС «Windows» або «Linux», а також програмне середовище PyCharm; 6) Вимоги до інформаційної та програмної сумісності: «Windows» будь-яка версія, залежно від версії PyCharm, «Linux» (будь-який дистрибутив); 7) Вимоги до маркування та упаковки (не висуваються); 8) Вимоги до транспортування і зберігання (не висуваються); 9) Спеціальні вимоги (не пред'являються).
5. Вимоги до програмної документації.	Програмою документацією до виробу вважати: 1) Справжнє Технічне завдання на розробку програмного виробу. 2) Програму і методику випробувань розробленого програмного

	виробу. 3) Опис програмного виробу в розділі 3 кваліфікаційної роботи. 4) Текст програми в додатку Г кваліфікаційної роботи.																										
6. Техніко-економічні показники	Орієнтовна оцінка ефективності виконуваної роботи: за отриманими під час виконання роботи даними, побудована модель класифікації є коректною та показує дуже добрі результати.																										
7. Стадії і етапи розробки	<table> <tr> <th>Дата</th><th>Назва етапу</th></tr> <tr> <td>05.09.2024 – 02.10.2024</td><td>Затвердження теми роботи</td></tr> <tr> <td>02.10.2024 – 24.10.2024</td><td>Аналіз та пошук методичної літератури щодо прогнозування споживання електроенергії</td></tr> <tr> <td>25.10.2024 – 09.11.2024</td><td>Огляд та аналіз методів аналізу часових рядів</td></tr> <tr> <td>10.11.2024 – 24.11.2024</td><td>Вибір алгоритмів прогнозування електроспоживання</td></tr> <tr> <td>25.11.2024 – 08.12.2024</td><td>Розробка математичної моделі прогнозування</td></tr> <tr> <td>09.12.2024 – 29.12.2024</td><td>Реалізація моделі на основі вибраного алгоритму</td></tr> <tr> <td>30.12.2024 – 15.01.2025</td><td>Тестування моделі, оцінка точності прогнозування</td></tr> <tr> <td>16.01.2025 – 01.02.2025</td><td>Оптимізація параметрів моделі, порівняльний аналіз</td></tr> <tr> <td>02.02.2025 – 20.02.2025</td><td>Аналіз програмних рішень для прогнозування часових рядів</td></tr> <tr> <td>21.02.2025 – 30.04.2025</td><td>Підготовка та оформлення звітних матеріалів</td></tr> <tr> <td>01.05.2025 – 30.05.2025</td><td>Оформлення пояснювальної записки</td></tr> <tr> <td>Червень 2025</td><td>Захист кваліфікаційної роботи</td></tr> </table>	Дата	Назва етапу	05.09.2024 – 02.10.2024	Затвердження теми роботи	02.10.2024 – 24.10.2024	Аналіз та пошук методичної літератури щодо прогнозування споживання електроенергії	25.10.2024 – 09.11.2024	Огляд та аналіз методів аналізу часових рядів	10.11.2024 – 24.11.2024	Вибір алгоритмів прогнозування електроспоживання	25.11.2024 – 08.12.2024	Розробка математичної моделі прогнозування	09.12.2024 – 29.12.2024	Реалізація моделі на основі вибраного алгоритму	30.12.2024 – 15.01.2025	Тестування моделі, оцінка точності прогнозування	16.01.2025 – 01.02.2025	Оптимізація параметрів моделі, порівняльний аналіз	02.02.2025 – 20.02.2025	Аналіз програмних рішень для прогнозування часових рядів	21.02.2025 – 30.04.2025	Підготовка та оформлення звітних матеріалів	01.05.2025 – 30.05.2025	Оформлення пояснювальної записки	Червень 2025	Захист кваліфікаційної роботи
Дата	Назва етапу																										
05.09.2024 – 02.10.2024	Затвердження теми роботи																										
02.10.2024 – 24.10.2024	Аналіз та пошук методичної літератури щодо прогнозування споживання електроенергії																										
25.10.2024 – 09.11.2024	Огляд та аналіз методів аналізу часових рядів																										
10.11.2024 – 24.11.2024	Вибір алгоритмів прогнозування електроспоживання																										
25.11.2024 – 08.12.2024	Розробка математичної моделі прогнозування																										
09.12.2024 – 29.12.2024	Реалізація моделі на основі вибраного алгоритму																										
30.12.2024 – 15.01.2025	Тестування моделі, оцінка точності прогнозування																										
16.01.2025 – 01.02.2025	Оптимізація параметрів моделі, порівняльний аналіз																										
02.02.2025 – 20.02.2025	Аналіз програмних рішень для прогнозування часових рядів																										
21.02.2025 – 30.04.2025	Підготовка та оформлення звітних матеріалів																										
01.05.2025 – 30.05.2025	Оформлення пояснювальної записки																										
Червень 2025	Захист кваліфікаційної роботи																										

8. Порядок контролю і приймання	1) Перевірку ходу розробки програмного виробу Керівнику робіт виконувати 1 раз в 3 тижні. 2) Випробування програмного виробу відповідно до програми і методики випробувань провести на базі комп'ютерного класу. 3) Захист розробленого програмного виробу провести на засіданні Атестаційної комісії. 4) Пояснювальну записку подати в електронному вигляді в примірнику.
---------------------------------	---

Виконавець

студент групи КІ-41

Хавроня А.Р.

_____  _____
 (підпис)

Замовник

канд. техн. наук, доц.

Бакуменко Н.С.

_____  _____

Додаток В**Програма і методика випробувань програмного виробу****“Комп’ютерна модель прогнозування споживання електроенергії за допомогою методів аналізу часових рядів”****1. Об’єкт випробувань**

1. Назва програмного виробу: “ Комп’ютерна модель прогнозування споживання електроенергії за допомогою методів аналізу часових рядів ”

2. Галузь застосування: електроенергетика

2. Мета випробувань

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

3. Загальні положення**1. Підстави для проведення випробувань**

Проведення випробувань здійснюється на підставі наказу про створення атестаційної комісії.

2. Місце та термін проведення випробувань

Приймально-здавальні випробування відбуваються в комп’ютерному класі кафедри під час роботи атестаційної комісії.

3. Обсяг випробувань

Випробування програмного продукту виконуються у межах, визначених відповідною програмою та методикою їх проведення.

4. Учасники випробувань

До приймальних випробувань залучається атестаційна комісія, яка проводить їх напередодні або під час засідання за участю Замовника, Розробника та інших зацікавлених осіб.

4. Вимоги до програми або програмного продукту

Розроблена модель повинна відповідати таким вимогам:

1. забезпечувати працездатність на основних операційних системах: Windows, Linux;
2. мати можливість легкого розширення функціональності;

3. мати модульну структуру, яка дозволяє ізолювати окремі компоненти з метою мінімізації впливу змін у коді на загальну роботу програми;
4. відповідати вимогам щодо складу та технічних характеристик необхідного обладнання;
5. вимоги до маркування та упаковки – не встановлюються;
6. вимоги до транспортування та зберігання – не встановлюються;
7. спеціальні вимоги – відсутні.

5. Вимоги до програмної документації

До складу програмної документації на програмний продукт «Комп'ютерна модель прогнозування споживання електроенергії за допомогою методів аналізу часових рядів» належать:

1. технічне завдання на створення програми (включається як Додаток Б до пояснювальної записки кваліфікаційної роботи);
2. програма та методика проведення випробувань розробленого програмного забезпечення (оформлюється як Додаток В до пояснювальної записки);
3. рекомендації щодо впровадження та стандартизації програмного продукту в подібних проєктах (викладаються у Розділі 3 пояснювальної записки);
4. вихідний текст програми (надається як Додаток Г до пояснювальної записки).

6. Засоби і порядок випробувань

6.1 Засоби вапробувань

Для проведення випробувань необхідна середа розробки PyCharm, відповідний файл з програмою лістинги в Додатку Г, операційна система, Windows, Linux.

6.2 Порядок проведення випробувань

Випробування програмного забезпечення для побудови моделей прогнозування електроспоживання проводились з метою перевірки коректності завантаження вхідних даних, обробки пропущених значень, налаштування параметрів моделей та візуалізації результатів прогнозування.

Випробування виконувались у такій послідовності:

1. Завантаження вхідного датасету (часовий ряд електроспоживання).

2. Автоматична перевірка даних на наявність пропущених значень та приведення їх до придатного для моделювання формату.
3. Побудова відповідної моделі (ARIMA, ARIMAX, SARIMA, SARIMAX).
4. Проведення прогнозування на заданий період.
5. Побудова графічного відображення результатів, що включає:
 1. вихідний часовий ряд;
 2. прогнозовані значення;
 3. інформацію про модель та її параметри.

Процес використання програмного забезпечення не потребує ручного введення даних через консоль або інші інтерфейси - всі дії ініціюються натисканням однієї кнопки запуску. Вся необхідна інформація представлена на графіку з відповідними підписами, що дозволяє фахівцю швидко інтерпретувати результати.

Критеріями успішності випробувань є:

1. коректна робота програми без помилок;
2. адекватна побудова моделей;
3. наочна візуалізація з достатнім рівнем інформативності;
4. відповідність результатів очікуванню (підтверджено в Розділі 3).

Висновки: як видно з розділу 3 всі вище вказані пункти виконано. Випробування пройшло успішно.

Виконавець: студент групи KI-41, Хавроня А.Р. _____


(підпис)

Модель ARIMA

```
warnings.filterwarnings("ignore")

# === 3. Перевірка на стаціонарність ===
def check_stationarity(data):
    result = adfuller(data)
    p_value = result[1]
    print(f"ADF p-value: {p_value:.5f}")
    if p_value > 0.05:
        print("→ Дані нестационарні. Виконуємо різницювання.")
        return data.diff().dropna(), 1
    print("→ Дані стаціонарні.")
    return data, 0

train_stationary, d_order = check_stationarity(train)

# === 4. Пошук найкращих параметрів ARIMA ===
def find_best_arima_params(data, p_range, d, q_range):
    best_aic = np.inf
    best_order = None
    best_model = None

    for p, q in itertools.product(p_range, q_range):
        try:
            model = ARIMA(data, order=(p, d, q))
            fitted = model.fit()
            print(f"ARIMA({p},{d},{q}) - AIC: {fitted.aic:.2f}")
            if fitted.aic < best_aic:
                best_aic = fitted.aic
                best_order = (p, d, q)
                best_model = fitted
        except:
            continue
    return best_order, best_model

# Визначаємо найкращі параметри
p_range = range(0, 5)
q_range = range(0, 5)
start_time = time.time()
best_order, best_model =
find_best_arima_params(train_stationary, p_range, d_order,
q_range)
end_time = time.time()
print(f"\nНайкраща ARIMA модель: {best_order}")
print(f"Час виконання моделі: {end_time - start_time:.2f}
секунд")
# === 5. Прогноз ===
forecast = best_model.forecast(steps=forecast_days)
```

```

forecast_index = pd.date_range(start=test_start,
periods=forecast_days, freq='D')

# === 6. SMAPE метрика ===
def smape(y_true, y_pred):
    y_true = np.array(y_true)
    y_pred = np.array(y_pred)
    denominator = np.abs(y_true) + np.abs(y_pred)
    mask = denominator != 0 # щоб уникнути ділення на нуль
    return 100 / len(y_true[mask]) * np.sum(2 *
np.abs(y_pred[mask] - y_true[mask]) / denominator[mask])

# Вирівнюємо довжини перед викликом smape
forecast = forecast[:len(test)]
test = test[:len(forecast)] # Додано на випадок розбіжностей
довжин

smape_score = smape(test, forecast)
print(f"SMAPE: {smape_score:.2f}%")

# === 7. Візуалізація ===
plt.figure(figsize=(14,6))
plt.plot(total_consumption, label='Історичні дані',
color='blue')
plt.plot(forecast_index[:len(forecast)], forecast,
label=f'Прогноз ARIMA{best_order}', color='red')
plt.axvspan(pd.to_datetime(test_start),
forecast_index[min(len(forecast_index)-1, len(forecast)-1)],
color='lightgrey', alpha=0.3, label='Прогнозований період')
plt.title(f"Прогноз електроспоживання на {forecast_days} днів
(sMAPE: {smape_score:.2f}%")
plt.xlabel("Дата")
plt.ylabel("Споживання (МВт)")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

МОДЕЛЬ SARIMA

```

train_data = df.loc[:'2019-06-30', "Total_Consumption"]

# Параметри SARIMA
p = d = q = range(0, 2)
P = D = Q = range(0, 2)
m = 24 # річна сезонність для щоденних даних

best_aic = np.inf
best_params = None
best_model = None

```

```

print("Початок підбору параметрів...")
start_time = time.time()
for param in itertools.product(p, d, q):
    for seasonal_param in itertools.product(P, D, Q):
        try:
            model = SARIMAX(train_data,
                            order=param,
                            seasonal_order=seasonal_param + (m,),
                            enforce_stationarity=False,
                            enforce_invertibility=False)
            result = model.fit(dispatch=False)
            if result.aic < best_aic:
                best_aic = result.aic
                best_params = (param, seasonal_param)
                best_model = result
            print(f"SARIMA{param}x{seasonal_param + (m,)} | AIC = {result.aic:.2f}")
        except:
            continue
param_search_time = time.time() - start_time
print(f"\nЧас підбору параметрів SARIMA: {param_search_time:.2f} секунд")
print("\nНайкращі параметри SARIMA:")
print(f"order={best_params[0]}, seasonal_order={best_params[1] + (m,)}, AIC={best_aic:.2f}")

# Графік
plt.figure(figsize=(14, 6))
plt.plot(df["Total_Consumption"], label="Реальні дані", color='blue')
plt.plot(forecast_index, forecast_mean, label="Прогноз (2019-07 до 2019-09)", color='red')
plt.fill_between(forecast_index, conf_int.iloc[:, 0], conf_int.iloc[:, 1], color='pink', alpha=0.3)
plt.axvspan(pd.to_datetime('2019-07-01'), pd.to_datetime('2019-09-30'), color='lightgrey', alpha=0.2, label="Прогнозований період")
plt.title("Прогноз електроспоживання SARIMA")
plt.xlabel("Дата")
plt.ylabel("Споживання")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

МОДЕЛЬ ARIMAX

```

# === 3. Перевірка на стаціонарність ===
def check_stationarity(data):
    result = adfuller(data)
    p_value = result[1]
    print(f"ADF p-value: {p_value:.5f}")

```

```

    if p_value > 0.05:
        print("→ Дані нестационарні. Виконуємо різницювання.")
        return data.diff().dropna(), 1
    print("→ Дані стаціонарні.")
    return data, 0

train_stationary, d_order = check_stationarity(train)
exog_train = exog_train.loc[train_stationary.index]

# === 4. Пошук найкращих параметрів ARIMAX ===
def find_best_arimax_params(endog, exog, p_range, d, q_range):
    best_aic = np.inf
    best_order = None
    best_model = None

    start_time = time.time()

    for p, q in itertools.product(p_range, q_range):
        try:
            model = SARIMAX(endog, exog=exog, order=(p, d, q),
enforce_stationarity=False, enforce_invertibility=False)
            fitted = model.fit(dispatch=False)
            print(f"ARIMAX({p},{d},{q}) - AIC: {fitted.aic:.2f}")
            if fitted.aic < best_aic:
                best_aic = fitted.aic
                best_order = (p, d, q)
                best_model = fitted
        except:
            continue

    end_time = time.time() # Кінець відліку часу
    elapsed_time = end_time - start_time
    print(f"\nЧас виконання пошуку найкращих параметрів ARIMAX:
{elapsed_time:.2f} секунд")
    return best_order, best_model

p_range = range(0, 5)
q_range = range(0, 5)
best_order, best_model =
find_best_arimax_params(train_stationary, exog_train, p_range,
d_order, q_range)

print(f"\nНайкраща ARIMAX модель: {best_order}")

# === 6. Візуалізація ===
plt.figure(figsize=(14,6))
plt.plot(total_consumption, label='Історичні дані',
color='blue')
plt.plot(forecast_index[:len(forecast)], forecast,
label=f'Прогноз ARIMAX{best_order}', color='red')

```

```

plt.axvspan(pd.to_datetime(test_start),
forecast_index[min(len(forecast_index)-1, len(forecast)-1)],
color='lightgrey', alpha=0.3, label='Прогнозований період')
plt.title(f"Прогноз електроспоживання на {forecast_days} днів")
plt.xlabel("Дата")
plt.ylabel("Споживання (МВт)")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

МОДЕЛЬ SARIMAX

```

# Параметри SARIMAX
p = d = q = range(0, 2)
P = D = Q = range(0, 2)
m = 24 # сезонність (наприклад, добова для погодинних або річна
для щоденних)

best_aic = np.inf
best_params = None
best_model = None

print("Початок підбору параметрів SARIMAX...")

start_time = time.time()

for param in itertools.product(p, d, q):
    for seasonal_param in itertools.product(P, D, Q):
        try:
            model = SARIMAX(train_data,
                            exog=exog_train,
                            order=param,
                            seasonal_order=seasonal_param + (m,),
                            enforce_stationarity=False,
                            enforce_invertibility=False)
            result = model.fit(dispatch=False)
            if result.aic < best_aic:
                best_aic = result.aic
                best_params = (param, seasonal_param)
                best_model = result
            print(f"SARIMAX{param}x{seasonal_param + (m,)} | AIC
= {result.aic:.2f}")
        except:
            continue

end_time = time.time() # кінець відліку часу
elapsed_time = end_time - start_time
print("\nНайкращі параметри SARIMAX:")

```

```

print(f"order={best_params[0]}, seasonal_order={best_params[1] +
(m,)}, AIC={best_aic:.2f}")
print(f"Час виконання підбору параметрів і тренування моделі:
{elapsed_time:.2f} секунд")

# Прогноз
forecast_steps = len(test_data)
forecast_index = test_data.index

forecast = best_model.get_forecast(steps=forecast_steps,
exog=exog_test)
forecast_mean = forecast.predicted_mean
conf_int = forecast.conf_int()

# Побудова графіку
plt.figure(figsize=(14, 6))
plt.plot(df["Total_Consumption"], label="Реальні дані",
color='blue')
plt.plot(forecast_index, forecast_mean, label="Прогноз SARIMAX",
color='red')
plt.fill_between(forecast_index, conf_int.iloc[:, 0],
conf_int.iloc[:, 1], color='pink', alpha=0.3)
plt.axvspan(pd.to_datetime('2019-07-01'), pd.to_datetime('2019-
09-30'), color='lightgrey', alpha=0.2, label="Прогнозований
період")
plt.title("Прогноз електроспоживання SARIMAX")
plt.xlabel("Дата")
plt.ylabel("Споживання")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

АНСАМБЛЮВАННЯ БУСТИНГОМ

```

# Формування DataFrame з прогнозами SARIMA/SARIMAX
ensemble_data = pd.DataFrame({
    'SARIMA': sarima_forecast.iloc[:len(y_true), 0].values,
    'SARIMAX': sarimax_forecast.iloc[:len(y_true), 0].values,
}, index=y_true.index)

ensemble_target = y_true.values

plt.figure(figsize=(14, 6))

# Побудова всіх реальних даних з 2019-01-01 по 2020-01-01
plot_range = df.loc["2019-01-01":"2020-01-01"]
plt.plot(plot_range.index, plot_range["Total_Consumption"],
label="Реальні дані", color="blue")

```

```

# Прогнози на навчальній частині
plt.plot(X_train.index, train_pred, color="green")

# Прогнози на тестовій частині
plt.plot(X_test.index, test_pred, label="Прогноз",
color="green")

plt.title("Ансамблевий прогноз методом бустингу")
plt.xlabel("Дата")
plt.ylabel("Сумарне споживання")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

АНСАМБЛЮВАННЯ БЕГІНГОМ

```

# Розбиваємо період на train/test 70%/30%
split_idx = int(len(data_period) * 0.7)
train_true = data_period.iloc[:split_idx]
test_true = data_period.iloc[split_idx:]

train_pred = ensemble_forecast["Forecast"].iloc[:split_idx]
test_pred = ensemble_forecast["Forecast"].iloc[split_idx:]

# Наївні прогнози (зсув на 1 день) для sMASE
train_naive = train_true.shift(1).fillna(method='bfill')
test_naive = test_true.shift(1).fillna(method='bfill')

# Графік
plt.figure(figsize=(14, 6))
plt.plot(df["Total_Consumption"], label="Реальні дані",
color='blue')
plt.plot(train_true.index, train_pred, color='green')
plt.plot(test_true.index, test_pred, label="Прогноз",
color='green')
#plt.axvline(x=train_true.index[-1], color='gray', linestyle='--',
', label="Розділ Train/Test")
plt.title("Ансамблевий прогноз методом бегінгу")
plt.xlabel("Дата")
plt.ylabel("Споживання")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# Збереження результату
ensemble_forecast.to_csv("ensemble_bagging_forecast.csv")

```