Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу»

Виконав: студент 2 курсу, групи 2ІСТ-24м
спеціальності 126 «Інформаційні системи та технології»
_____Тарас СКРИННИК
Керівник: д.т.н., проф.
_____ Олександр МОКІН
«27» _____11_____ 2025 р.
Опонент: д.т.н., проф.
_____ Ярослав ІВАНЧУК
«03» _____12_____ 2025 р.

Допущено до захисту

Завідувач кафедри САІТ
_____ д.т.н., проф. Віталій МОКІН
«28» _____11_____ 2025 р.

Вінниця ВНТУ – 2025 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій
Рівень вищої освіти – другий (магістерський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 126 Інформаційні системи та технології
Освітньо-професійна програма – Інформаційні технології аналізу даних та зображень

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ
д.т.н., проф. Віталій МОКІН
«25» 09 2025 року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
Скриннику Тарасу Васильовичу

1. Тема роботи: «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу»
керівник роботи: Олександр МОКІН, д.т.н., проф
затверджені наказом ВНТУ від «24» 09 2025 року №313
2. Термін подання студентом роботи 28.11.2025 року
3. Вихідні дані до роботи:
Дані сервісу моніторингу якості повітря EcoCity.
4. Зміст текстової частини:
   1)Загальна характеристика об'єкту дослідження та аналіз інформаційних технологій;
   2) Підготовка та розвідувальний аналіз даних;
   3) Розробка інформаційної технології та прогнозування.
   4) Економічна частина.
5. Перелік ілюстративного матеріалу:
   1) Вигляд датасету;
   2) Діаграми розподілу даних;
   3) Блок-схема алгоритму інформаційної технології аналізу та прогнозування поширення пилу Сахари Україною;
   4) Результат діагностики оптимальної моделі ARIMA
   5) Виявлення різких стрибків концентрації |ΔPM1
   6) Результати прогнозування якості повітря та поширення пилу Сахари за допомогою найкращої моделі для станції ВНТУ,

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4	Олександр ЛЕСЬКО, к.е.н.,проф.каф. ЕПВМ		

7. Дата видачі завдання «25» 09 2025 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Загальна характеристика об'єкту дослідження	15.09.2025	25.09.2025	виконано
2	Підготовка та розвідувальний аналіз даних	25.09.2025	05.10.2025	виконано
3	Створення інформаційної технології прогнозування	05.10.2025	10.10.2025	виконано
4	Прогнозування поширення пилу	10.10.2025	05.11.2025	виконано
5	Економічна частина	05.11.2025	15.11.2025	виконано
6	Оформлення матеріалів до захисту МКР	15.11.2025	25.11.2025	виконано

Студент _____ Тарас СКРИННИК

Керівник роботи _____ Олександр МОКІН

АНОТАЦІЯ

УДК 004.09:504.054

Скринник Т.В. Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу. Магістерська кваліфікаційна робота зі спеціальності 126 – інформаційні системи та технології, освітньо-професійна програма – інформаційні технології аналізу даних та зображень. Вінниця: ВНТУ, 2025. 116 с.

На укр.мові. Бібліогр.: 33 назв; рис.: 63; табл.: 11.

Метою роботи є розробка інформаційної технології для прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу.

У роботі представлено розробку алгоритмів для збору, обробки та аналізу даних, а також створення моделей для прогнозування рівня забруднення повітря. Було розроблено технологію для обробки даних, яке включає в себе алгоритми аналізу та прогнозування, а також проведено тестування на реальних даних. Розроблена інформаційна технологія забезпечує точність і оперативність прогнозування забруднення повітря пилом Сахари, що дозволяє покращити якість моніторингу та своєчасне реагування на екологічні загрози. Отримані результати можуть бути впроваджені у системи екологічного спостереження.

Ілюстративна частина складається з 6 плакатів.

У розділі економічної частини розглянуто питання про доцільність розробки та впровадження інформаційної технології аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу

Ключові слова: якість повітря, прогнозування, машинне навчання, Prophet, інформаційна технологія, пил Сахари.

ABSTRACT

Skrynnyk T.V. Information technology for analyzing and forecasting the spread of Saharan dust in the air of Ukraine based on public monitoring data. Master's thesis in the specialty 126 – Information Systems and Technologies, educational and professional program – Information Technology Data and Image Analysis. Vinnytsia: VNTU, 2025. 116 p.

In Ukrainian. Bibliography: 33 references; figures: 63; tables: 11.

The aim of the work is to develop information technology for forecasting the spread of Saharan dust in the air in Ukraine based on public monitoring data.

The work presents the development of algorithms for collecting, processing, and analyzing data, as well as creating models for forecasting air pollution levels. A technology for data processing was developed, which includes algorithms for analysis and forecasting, and testing was carried out on real data. The developed information technology ensures the accuracy and timeliness of forecasting air pollution by Saharan dust, which improves the quality of monitoring and timely response to environmental threats. The results obtained can be implemented in environmental monitoring systems.

The illustrative part consists of 6 posters.

The economic section examines the feasibility of developing and implementing information technology for analyzing and forecasting the spread of Saharan dust in the air in Ukraine based on public monitoring data.

Keywords: air quality, forecasting, machine learning, Prophet, information technology, Saharan dust

ЗМІСТ

ВСТУП

Актуальність теми. У сучасних умовах якість атмосферного повітря є однією з ключових екологічних проблем, що безпосередньо впливає на здоров'я та добробут населення. Забруднення повітря може спричиняти серйозні захворювання, зокрема астму, серцево-судинні хвороби та рак легень. Тому вивчення якості атмосферного повітря, а також розробка ефективних методів його моніторингу та прогнозування рівня забруднення є критично важливими для забезпечення безпечного та здорового середовища проживання. В останні роки на якість повітря в Україні додатково впливає пил із Сахари, що спричинив значний дискомфорт серед населення.

Мета і задачі дослідження. Метою даної роботи є підвищення точності прогнозування поширення пилу Сахари в атмосферному повітрі України за даними громадського моніторингу шляхом створення інформаційної технології.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

– провести аналіз проблеми прогнозування якості повітря та розглянути існуючі сервіси моніторингу якості повітря;

– здійснити вибір оптимальних інформаційних технологій та моделей машинного навчання для прогнозування рівня забруднення повітря пилом Сахари, а також для збору та обробки даних;

– провести розвідувальний аналіз даних;

– розробити інформаційну технологію аналізу та прогнозування якості атмосферного повітря та поширення пилу Сахари територією України.

Об'єктом дослідження є процес розроблення інформаційної технології аналізу та прогнозування якості атмосферного повітря та поширення пилу Сахари Україною.

Предметом дослідження є методи машинного навчання та інформаційна технологія аналізу та прогнозування атмосферного якості повітря та поширення пилу Сахари Україною.

Новизна одержаних результатів. Подальшого розвитку набула інформаційна технологія аналізу та прогнозування поширення пилу Сахари в атмосферному повітрі України за даними громадського моніторингу, за рахунок оптимізації інтелектуальних моделей методами машинного навчання, що дозволяє підвищити точність прогнозу його концентрації в атмосферному повітрі м. Вінниці.

Апробація та публікації результатів магістерської кваліфікаційної роботи. Опубліковано тези на LV Всеукраїнській науково-технічній конференції Вінницького національного технічного університету (2025-2026) [1].

1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ОБ'ЄКТУ ДОСЛІДЖЕННЯ ТА АНАЛІЗ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

1.1 Аналіз предметної області

Якість повітря – це характеристика стану атмосферного повітря, яке становить середовище нашого дихання, і яка оцінюється за концентраціями шкідливих домішок (пилу, хімічних речовин, газів, диму тощо). Висока якість означає, що забруднювачів мало або їхня концентрація відповідає або перевищує нормативи безпеки, тоді як низька якість вказує на перевищення допустимих рівнів і підвищену загрозу для здоров'я [2].

Для систематичної оцінки якості повітря застосовують індекси якості повітря (Air Quality Indices, AQI) - інтегровані показники, що узагальнюють вплив декількох забруднювачів у єдиному числовому значенні. Наприклад, у Канаді використовується Air Quality Health Index (AQHI), шо зображена на рисунку 1.1 - шкала, яка безпосередньо пов'язує значення індексу з ризиком для здоров'я, а не просто з нормами викидів [3]. Згідно з офіційним джерелом, AQHI використовується на шкалі від 1 до 10+, де більші значення відповідають підвищеному ризику:

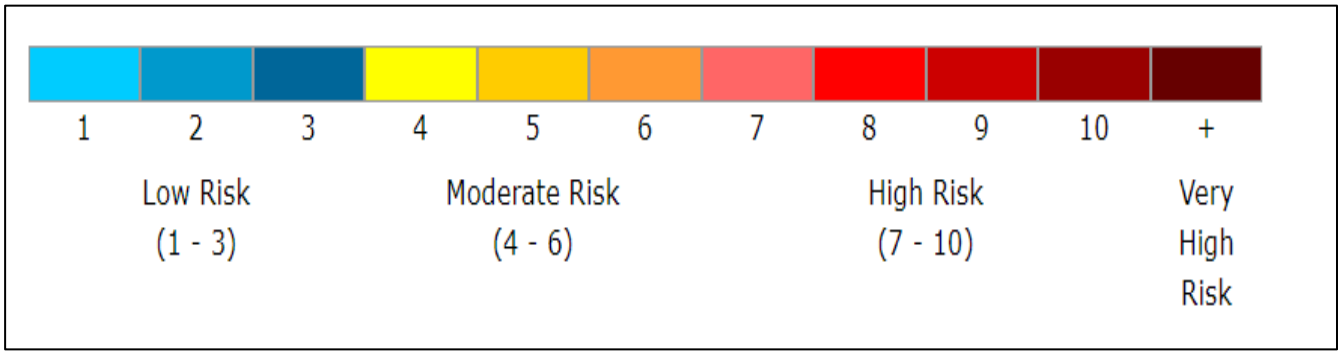

Рисунок 1.1 – Шкала вимірювання AQHI в Канаді

На рисунку 1.2 подано шкалу, що застосовується у Великій Британії та в багатьох інших країнах. Зелений колір означає, що якість повітря вважається

задовільною, а рівень забруднення не становить небезпеки або є практично відсутнім. Жовтий і помаранчевий кольори відповідають прийнятній якості повітря; однак окремі забруднювачі можуть викликати помірне занепокоєння щодо здоров'я в дуже невеликої кількості людей, які є надзвичайно чутливими до забруднення повітря. Відтінки червоного сигналізують про те, що кожен може почати відчувати вплив на здоров'я, тоді як у представників чутливих груп можливі більш серйозні наслідки. Чорний колір означає, що для всіх осіб можливі ще більш серйозні наслідки для здоров'я [4].

Band	Index	Ozone Running 8-hour mean ($\mu g\ m^{-3}$)	Nitrogen dioxide 1-hour mean ($\mu g\ m^{-3}$)	Sulphur dioxide 15-minute mean ($\mu g\ m^{-3}$)	$PM_{2.5}$ particles 24-hour mean ($\mu g\ m^{-3}$)	PM_{10} particles 24-hour mean ($\mu g\ m^{-3}$)
Low	1	0–26	0–66	0–88	0–11	0–16
	2	27–53	67–133	89–176	12–23	17–33
	3	54–80	134–200	177–265	24–35	34–50
Moderate	4	81–107	201–267	266–354	36–41	51–58
	5	108–134	268–334	355–442	42–46	59–66
	6	135–160	335–400	443–531	47–53	67–75
High	7	161–187	401–467	532–708	54–58	76–83
	8	188–213	468–534	709–886	59–64	84–91
	9	214–240	535–600	887–1063	65–70	92–100
Very High	10	241 or more	601 or more	1064 or more	71 or more	101 or more

Рисунок 1.2 –Вигляд шкали оцінювання атмосферного повітря у Великій Британії

В Україні для щоденної оцінки якості повітря використовується індекс Air Quality Index (AQI), який характеризує стан повітря із розбиттям на шість категорій:

- добра якість повітря;
- задовільна якість;
- якість несприятлива для чутливих груп;
- погана якість;
- дуже погана якість;
- надзвичайно погана якість [5].

Індекс базується на концентраціях п’яти основних забруднювачів: дрібнодисперсних частинок (PM_1, $PM_{2.5}$, PM_{10}), озону (O_3), діоксиду азоту (NO_2), оксиду вуглецю (CO) і діоксиду сірки (SO_2) [4]. Дані про їхнє збільшення надходять із мережі станцій моніторингу, що поширюється по всій країні.

1.2 Аналіз сервісів моніторингу якості повітря в Україні та аналіз проблеми пилу Сахари

Якість повітря, яким дихає людина, є критично важливим чинником для її здоров’я, добробуту та тривалості життя. За даними Всесвітньої організації охорони здоров’я (ВООЗ), забруднене повітря щороку спричиняє від 7 до 9 мільйонів передчасних смертей у світі [6]. Основними наслідками впливу забрудненого повітря є захворювання дихальної системи - бронхіт, астма, пневмонія, а також серцево-судинні хвороби, інсульт і рак легенів. Навіть короткочасне перебування в умовах перевищення рівнів пилових частинок може викликати головний біль, втому, подразнення очей і утруднене дихання (рисунок 1.3) [6]. Найбільше від забруднення повітря страждають діти, літні люди, вагітні жінки та особи з уже існуючими хронічними захворюваннями. У дітей тривале перебування в умовах забрудненого повітря може призводити до уповільнення розвитку легенів і зниження когнітивних функцій, а у вагітних жінок – до передчасних пологів і низької ваги новонароджених. Крім прямого впливу на здоров’я, забруднення повітря має значні економічні наслідки: втрата працездатності, зростання витрат на охорону здоров’я, зниження продуктивності праці.

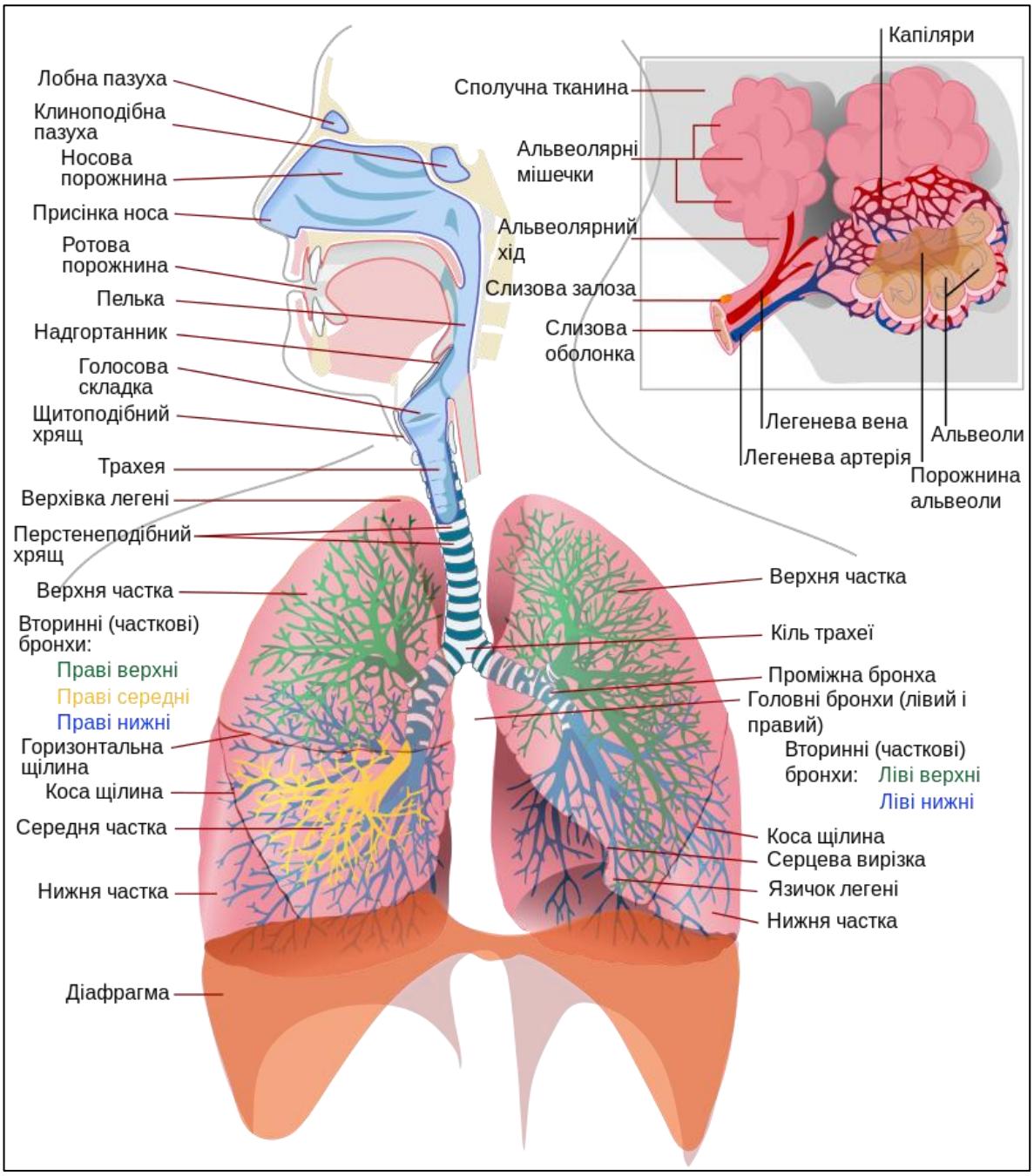

Рисунок 1.3 – Дихальна система людини

Отже, моніторинг якості повітря відіграє надзвичайно важливу роль у забезпеченні здоров’я та добробуту населення. Регулярне відстеження рівня забруднення дає змогу не лише своєчасно виявляти небезпечні концентрації шкідливих речовин, а й оперативно вживати ефективних заходів для їх зменшення. Нині існує значна кількість сервісів моніторингу якості повітря, що надають точну й актуальну інформацію про стан атмосферного повітря в режимі реального часу. Розглянемо деякі з них.

SaveEcoBot — це комплексна екологічна система, яка поєднує моніторинг довкілля, інструменти для протидії забрудненню та платформу для обміну даними [8].

Її можна уявити у вигляді інтерактивної карти України, на якій відображається поточний стан якості повітря у вашому місті, селі чи навіть на окремій вулиці. На рисунку 1.4 наведено приклад відображення такої інформації на сайті https://www.saveecobot.com/

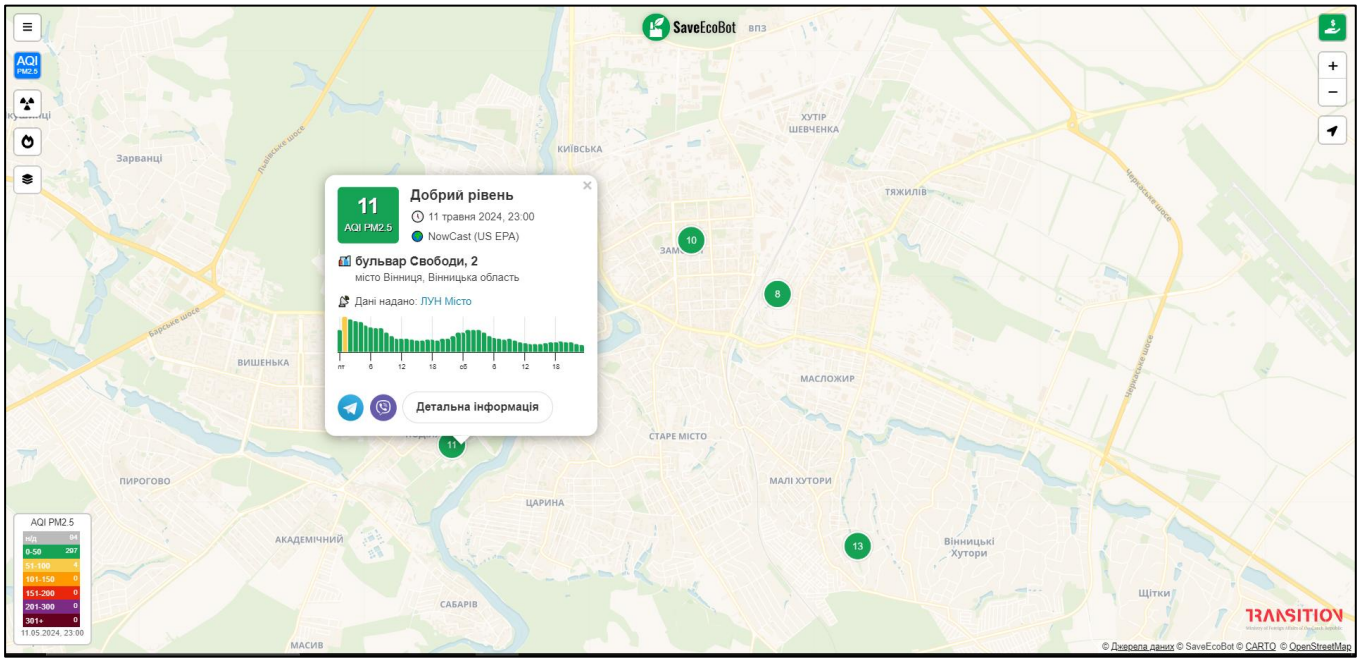

Рисунок 1.4 – Інформація, що наведена на сайті

SaveEcoBot збирає дані з різних джерел - державних систем, місцевих органів влади, громадських станцій та комерційних приладів - щоб сформувати максимально точну картину стану довкілля.

Окрім цього, сервіс відображає рівень радіаційного фону по всій території країни, оскільки це є важливим показником екологічної безпеки. За потреби проаналізувати динаміку змін якості повітря можна переглянути історичні дані за обраний період.

Також SaveEcoBot веде реєстр оцінок впливу на довкілля (ОВД), які є обов'язковими для окремих видів діяльності, що можуть чинити негативний вплив на навколишнє середовище.

reborn.eco-city.org.ua - це вебсайт проєкту громадського моніторингу якості повітря в Україні. Це некомерційна ініціатива, спрямована на надання українським громадянам точної та доступної інформації про стан повітря. Проєктом опікуються волонтери та ентузіасти, які дбають про захист довкілля [9].

Особливості Eco-City:

На сайті представлена інтерактивна мапа, що відображає поточні показники якості повітря для різних населених пунктів по всій території України. Карта використовує кольорове кодування для позначення рівня забруднення: зелений колір відповідає хорошій якості повітря, тоді як червоний вказує на погану [9]. Детальний вигляд шкали можна побачити на рисунках 1.5 та 1.6.

Категорія	Значення	Для уразливих та чутливих груп
Категорія якості повітря I	**Добра якість повітря**	Вміст у повітрі забруднюючих речовин у межах норми та не становить небезпеки. Плануйте діяльність, відпочинок або інші активності на відкритому повітрі без додаткових обмежень, застережень або рекомендацій.
Категорія якості повітря II	**Задовільна якість повітря**	Вміст у повітрі забруднюючих речовин у межах норми та не становить небезпеки. Плануйте діяльність, відпочинок або інші активності на відкритому повітрі без додаткових обмежень, застережень або рекомендацій. У рідких випадках дуже невеликої кількості населення, які надзвичайно чутливі до забруднення повітря, можуть виникнути дискомфорт та рефлекторні реакції – кашель, задишка тощо. Надзвичайно чутливим до забруднення повітря людям рекомендуємо зменшити тривалі або важкі фізичні навантаження на відкритому повітрі.
Категорія якості повітря III	**Якість повітря несприятлива**	Вміст у повітрі забруднюючих речовин у межах норми, проте уразливі та чутливі до забруднення категорії населення можуть мати наслідки для стану здоров'я при довготривалому впливі. Враховуйте спеціальні обмеження, застереження та рекомендації під час планування діяльності, відпочинку або інших активностей на відкритому повітрі. Рекомендуємо уразливим та чутливим до забруднення категоріям населення зменшити тривалі або важкі фізичні навантаження на відкритому повітрі. Якщо ви відчуваєте дискомфорт під час дихання та прояви інших рефлекторних реакцій – рекомендуємо дотримуватися звичайних порад та плану лікування від вашого лікаря. Люди хворі на астму, ХОЗЛ та респіратері захворювання можуть відчувати посилення звичних симптомів та рефлекторних реакцій. Рекомендуємо сумлінно дотримуватися свого плану терапії хронічних захворювань та прийому ліків, які призначив ваш лікар. Люди із хронічними захворюваннями серцево-судинної системи можуть відчувати додаткові та посилені рефлекторні реакції – підвищене серцебиття, задишка або незвичайна втома. У випадку довготривалого прояву цих реакцій рекомендуємо звернутися за консультацією до вашого лікаря та сумлінно виконувати його настанови.

Рисунок 1.5 – Шкала якості повітря, яка пропонується на сайті

Категорія якості повітря IV	**Погана якість повітря**	Вміст у повітрі забруднюючих речовин становить небезпеку при довготривалому впливі. Усі категорії населення можуть відчувати загострення рефлекторних реакцій та мати наслідки для здоров'я при довготривалому впливі. Діють спеціальні обмеження, застереження та рекомендації для тривалості діяльності, відпочинку або інших активностей на відкритому повітрі. Усім уразливим та чутливим до забруднення категоріям населення рекомендуємо виключити та перенести на інших час тривалі або важкі фізичні навантаження на відкритому повітрі. Чутливі до забруднення категорії населення, які мають хронічні захворювання, можуть відчувати значне посилення звичних симптомів та рефлекторних реакцій. Рекомендуємо сумлінно дотримуватися свого плану терапії хронічних захворювань та прийому ліків, які призначив ваш лікар. У випадку довготривалого прояву рефлекторних реакцій рекомендуємо звернутися за консультацією до вашого лікаря та сумлінно виконувати його настанови.
Категорія якості повітря V	**Дуже погана якість повітря**	Для уразливих та чутливих до забруднення категорій населення вміст у повітрі забруднюючих речовин становить небезпеку навіть при короткостроковому впливі. Усі категорії населення можуть відчувати сильне загострення рефлекторних реакцій та мати наслідки для здоров'я небезпеку навіть при короткостроковому впливі. Для уразливих та чутливих до забруднення категорій населення діють спеціальні обмеження, застереження та рекомендації для перебування на відкритому повітря для будь-якої діяльності. Рекомендуємо виключити або перенести на інший час будь-яку діяльність на відкритому повітрі. Якщо у вас виникає рефлекторна реакція на забруднене повітря – кашель, задишка, подразнення слизових оболонок носоглотки, біль в очах тощо – рекомендуємо використовувати засоби індивідуального захисту органів дихання. У випадку довготривалого прояву рефлекторних реакцій або інших відчутних наслідків для вашого здоров'я рекомендуємо звернутися за консультацією до вашого лікаря та сумлінно виконувати його настанови.
Категорія якості повітря VI	**Надзвичайно погана якість повітря**	Вміст у повітрі забруднюючих речовин становить небезпеку навіть при короткостроковому впливі. Усі категорії населення можуть відчувати сильне загострення рефлекторних реакцій та мати наслідки для здоров'я небезпеку навіть при короткостроковому впливі. Для всіх категорій населення діють спеціальні обмеження, застереження та рекомендації для перебування на відкритому повітря для будь-якої діяльності. Рекомендуємо виключити або перенести на інший час будь-яку діяльність на відкритому повітрі. Якщо у вас виникає рефлекторна реакція на забруднене повітря – кашель, задишка, подразнення слизових оболонок носоглотки, біль в очах тощо – рекомендуємо використовувати засоби індивідуального захисту органів дихання. У випадку довготривалого прояву рефлекторних реакцій або інших відчутних наслідків для вашого здоров'я рекомендуємо звернутися за консультацією до вашого лікаря та сумлінно виконувати його настанови.

Рисунок 1.6 – Шкала якості повітря, яка пропонується на сайті

На сайті подано детальні дані про якість повітря для кожної станції моніторингу, зокрема значення концентрацій різних забруднювальних речовин, таких як $PM_{2.5}$, PM_{10}, NO_2, SO_2 та O_3. На рисунку 1.7 наведено приклад відображення цих даних на сайті.

Крім того, там подаються рекомендації щодо збереження здоров’я залежно від поточного рівня якості повітря. Наприклад, людям із респіраторними захворюваннями можуть радити уникати інтенсивної фізичної активності на свіжому повітрі, якщо рівень забруднення повітря є високим.

Користувачі також можуть зареєструватися, щоб отримувати попередження електронною поштою або SMS у разі, коли показники якості повітря в їхньому регіоні перевищують установлені граничні значення.

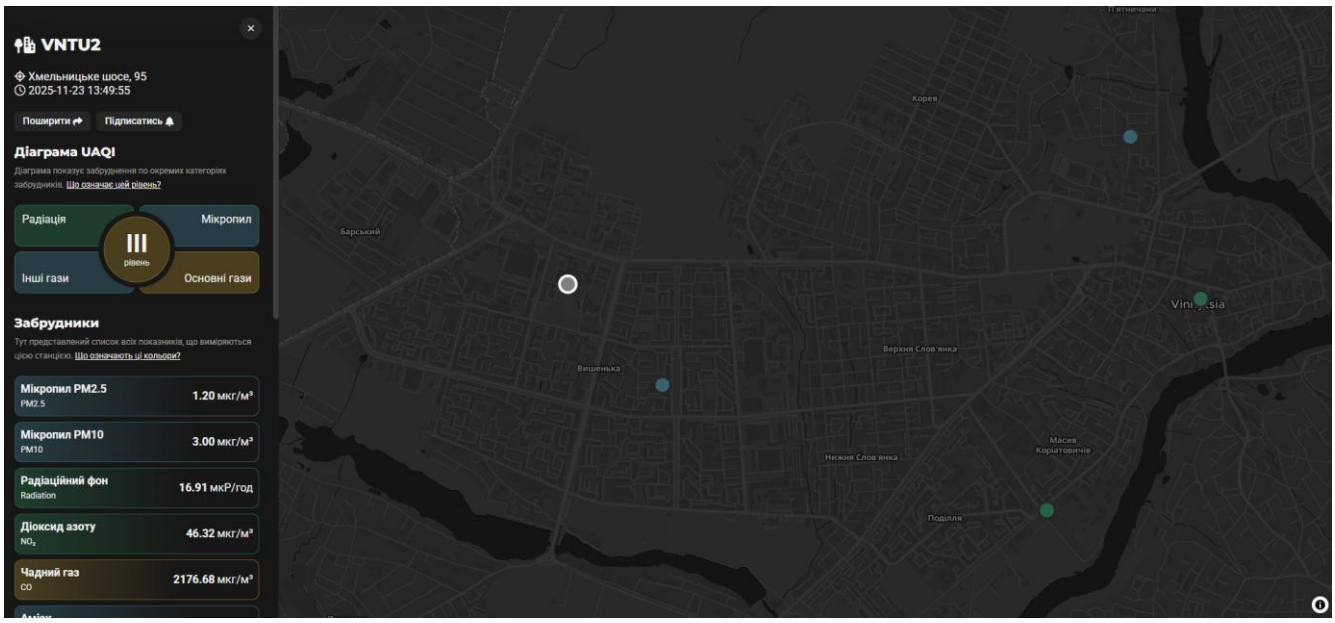

Рисунок 1.7 – Дані із сайту EcoCity

Дані проекту доступні громадськості безкоштовно для використання в дослідженнях, освіті та інших цілях.

Узагальнюючи, платформу EcoCity можна розглядати як важливе джерело інформації для всіх, хто прагне відстежувати стан якості атмосферного повітря в Україні. Орієнтація цього проєкту на відкритість та вільний доступ екологічних даних для громадськості робить його дієвим інструментом як для захисту довкілля, так і для активного залучення громадян до екологічних ініціатив.

Забруднення повітря може спричинятися як природними процесами, так і наслідками техногенної (людської) діяльності.

Джерела забруднення повітря поділяють на природні (вулканічний та космічний пил, пилові бурі, біогенні викиди) та антропогенні - викиди транспорту, промисловості й сільського господарства [10].

У межах цієї кваліфікаційної роботи основну увагу зосереджено на пилових бурях, зокрема на пилу із Сахари, який у 2024 році був чітко зафіксований на території України. Африканський пил здатний переноситися на великі відстані й досягати окремих регіонів Європи, Південної та Центральної Америки, Карибського басейну, а також Сполучених Штатів. Це явище не є новим для

євроатлантичного простору, адже подібні епізоди вже спостерігалися раніше. В Україні його також реєстрували у липні 2021 року [11].

Що ж таке пил із Сахари? Це мікроскопічні тверді частинки, що перебувають у повітрі над пустелею Сахара і переносяться на великі відстані під дією вітрів, циклонів та антициклонів. Наявність такого пилу в атмосфері змінює звичний колір неба, що проілюстровано на рисунку 1.8.

Рисунок 1.8 – Зміна забарвлення неба після впливу пилу з Сахари

Опади з пилом із Сахари осідають у вигляді грязьового дощу, або ж як його називають – кривавого дощу. На рисунку 1.9 показані наслідки таких опадів.

Підвищений вміст сахарського пилу в повітрі загострює захворювання дихальної системи (астму, хронічний бронхіт), провокує алергічні реакції, подразнення слизових оболонок очей і верхніх дихальних шляхів. Особливо вразливими є діти, люди похилого віку та особи із серцево-судинними захворюваннями.

Рисунок 1.9 – Наслідки опадів із пилом Сахари

Також може бути сухе осідання пилу, відбувається воно тоді, коли концентрація частинок висока й вони падають на поверхню землі під дією сили тяжіння.

У 2024 році сахарський пил двічі досяг території України: уперше - наприкінці березня та на початку квітня, вдруге — наприкінці квітня. Виникнення цих епізодів було зумовлене синоптичною ситуацією та перенесенням повітряних мас планетою. Наприкінці березня 2024 року спостерігався південний вітер зі швидкістю від 7 до 12 м/с, а в західних регіонах України його пориви місцями досягали близько 30 м/с. Наприкінці квітня 2024 року вітер переважно мав південно-східний напрямок зі швидкістю до 10 м/с, небо було вкрите хмарами, очікувалися опади. На рисунках 1.10 та 1.11 наведено супутникові знімки станом на 1 та 24 квітня 2024 року відповідно [12].

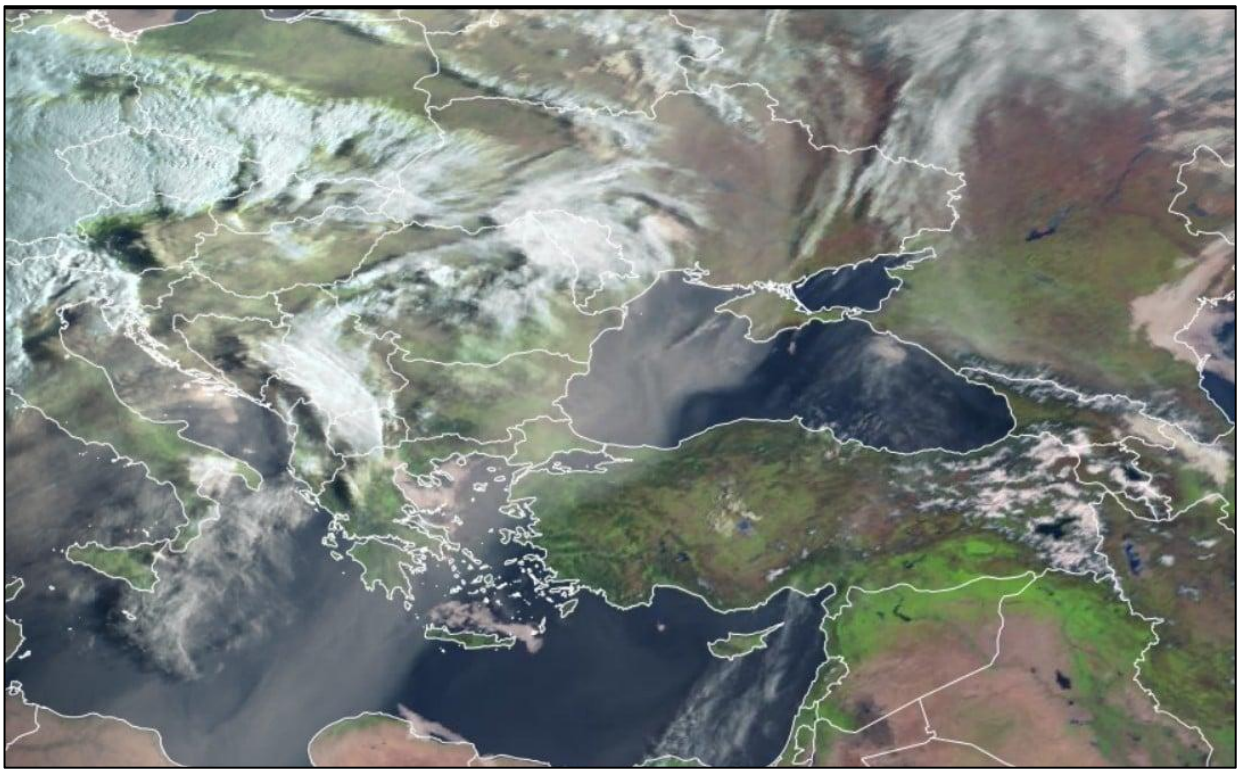

Рисунок 1.10 – Супутниковий знімок станом на 1 квітня 2024 року

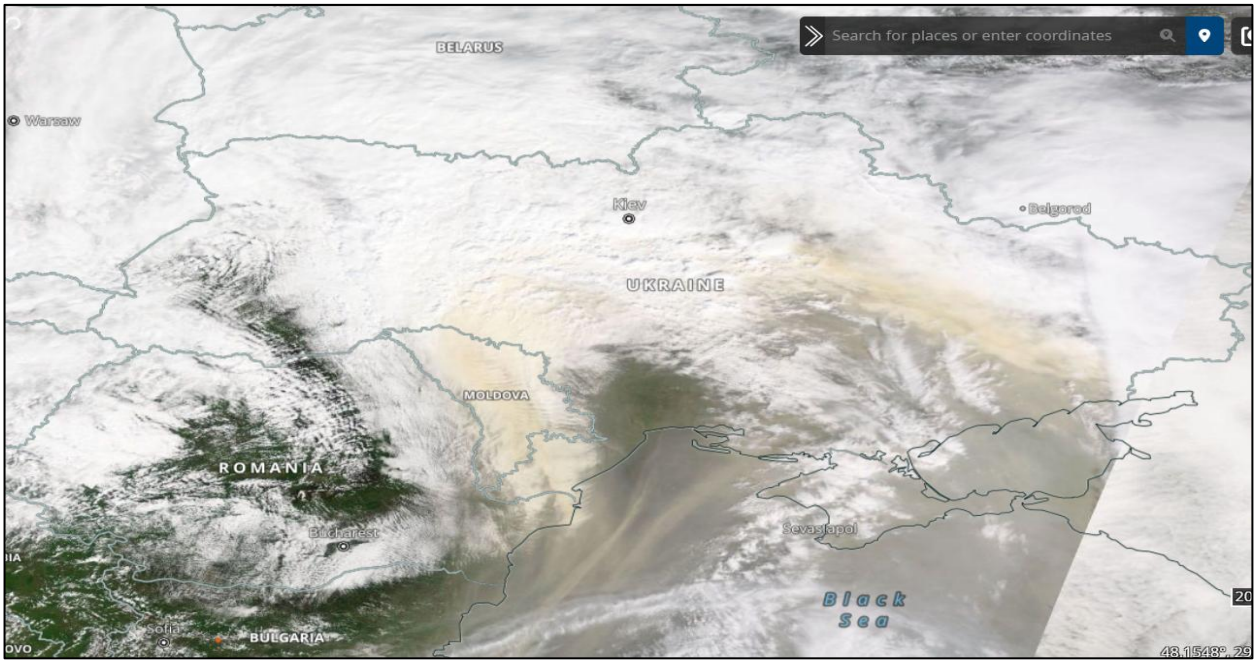

Рисунок 1.11 – Супутниковий знімок станом на 24 квітня 2024 року

На наведених супутникових знімках простежується переміщення повітряних мас у ці дні, а також чітко видно напрямок, з якого сахарський пил потрапляв на територію України.

1.3 Аналіз моделей та бібліотек Python для прогнозування даних

Мова Python є однією з найпоширеніших у сфері аналітики даних і машинного навчання. Її популярність зумовлена відкритою екосистемою, гнучкістю, простотою синтаксису та широкою спільнотою розробників. Створена у 1991 році Гвідо ван Россумом у Centrum Wiskunde & Informatica (Нідерланди), мова Python сьогодні є стандартом у галузі Data Science, Machine Learning, Forecasting та AI Engineering [13].

Основними перевагами Python є універсальність, висока читабельність, потужна стандартна бібліотека та тисячі відкритих пакетів для статистики, візуалізації, обчислень і моделювання часових рядів та іншого [14].

– Універсальність. Python можна застосовувати практично в будь-якій сфері: від створення простих програм і веб-застосунків до задач науки про дані, аналітики та машинного навчання.

– Читабельність. Код на Python зазвичай має зрозумілу структуру: логічні блоки виділяються відступами, що робить його легшим для сприйняття та супроводу.

– Широка екосистема бібліотек. У Python існує велика кількість готових бібліотек - це набори модулів і функцій, які значно спрощують розробку. Завдяки цьому немає потреби щоразу писати з нуля логіку та алгоритми, оскільки можна скористатися вже реалізованим функціоналом.

Ця мова програмування особливо популярна в галузі машинного навчання та роботи з даними. Основні причини - простий, зрозумілий синтаксис і наявність потужних спеціалізованих бібліотек, таких як NumPy, Pandas, Matplotlib, SciPy, Scikit-learn, Plotly.

NumPy [15] вважають однією з ключових та найпоширеніших Python-бібліотек для задач машинного навчання та штучного інтелекту. Це відкрита числова бібліотека, яку використовують для виконання різноманітних математичних операцій над масивами. Завдяки своїй популярності та

функціональності NumPy посідає важливе місце серед наукових бібліотек: багато фахівців активно застосовують її для аналізу даних.

Масиви NumPy займають значно менше пам’яті порівняно зі звичайними списками Python, а операції з ними виконуються швидше й у більш зручний спосіб. У підсумку NumPy є надзвичайно ефективним інструментом для підвищення продуктивності моделей машинного навчання без суттєвого ускладнення процесу розробки [15].

Pandas розширює можливості NumPy, додаючи об’єкти DataFrame і Series, що дозволяють ефективно обробляти табличні дані. Вона надає інструменти для очищення, трансформації, агрегації й аналізу часових рядів - включно з ресемплінгом, обчисленням ковзних середніх і заповненням пропущених значень [16].

Основні структури даних Pandas:

– DataFrame. Це базовий тип даних у Pandas - двовимірна таблиця з рядками та стовпцями, подібна до електронних таблиць. Кожен стовпець у такій таблиці може містити дані різних типів (числові значення, текст, категоріальні ознаки тощо).

– Series. Це одновимірна послідовність даних, яку можна порівняти з масивом NumPy, але кожне значення має власну мітку (індекс). Завдяки індексації стає значно простіше звертатися до окремих елементів і виконувати над ними різні операції [16].

Функціональні можливості Pandas:

– Очищення даних. Pandas надає засоби для обробки пропущених значень, виявлення та усунення дублікатів, зміни типів даних та інших перетворень, необхідних для підготовки набору даних до подальшого аналізу.

– Маніпуляція даними. За допомогою Pandas можна додавати, видаляти й перейменовувати рядки та стовпці, а також об’єднувати та розділяти таблиці. Це дає змогу гнучко змінювати структуру даних і адаптувати її під конкретні завдання аналізу.

– Індексація та фільтрація. Бібліотека підтримує різноманітні механізми вибірки даних на основі умов, значень індексу або діапазонів, що дозволяє швидко отримувати потрібні підмножини.

– Агрегація даних. У Pandas легко виконувати агрегувальні операції (сума, середнє значення, мінімум, максимум тощо) як для окремих стовпців і рядків, так і для цілих таблиць, що допомагає сформувати стислий узагальнений огляд даних.

– Інтеграція з іншими бібліотеками. Важливою перевагою Pandas є її тісна інтеграція з іншими популярними Python-інструментами для аналізу даних, такими як NumPy, Matplotlib, Scikit-learn. У поєднанні вони утворюють потужний комплекс засобів для всебічної роботи з даними.

Matplotlib [17] - це відкрита бібліотека Python, призначена для побудови статичних, анімованих і інтерактивних графічних представлень даних. Вона надає широкий набір інструментів для створення різноманітних графіків, діаграм та інших елементів візуалізації, що дає змогу перетворювати числову інформацію на наочні та інформативні зображення. За допомогою Matplotlib можна будувати гістограми, лінійні графіки, діаграми розсіювання, секторні (кругові) діаграми, контурні графіки та багато інших типів візуалізацій. Користувач має можливість детально налаштовувати створені графіки, зокрема змінювати тип і стиль ліній, кольори, маркери, шрифти, параметри осей, масштабування та легенди [17].

Ця бібліотека є незмінним інструментом для візуалізації та аналізу даних, допомагаючи представити дані зрозуміло та зручно.

SciPy [18] це безкоштовна бібліотека з відкритим кодом, яка використовується для вирішення математичних, наукових, інженерних і технічних задач. Вона дозволяє користувачам маніпулювати даними та візуалізувати дані за допомогою широкого спектру високорівневих команд Python. SciPy побудовано на базі NumPy, тому їх зручно використовувати разом [18].

Scikit-learn [19] - це безкоштовна бібліотека машинного навчання на мові Python, яка надає широкий набір алгоритмів як навчання з учителем, так і без нього. Важливою її особливістю є те, що вона побудована на основі кількох поширених математичних бібліотек і забезпечує їхню зручну взаємодію між собою. Ще одна

суттєва перевага - велика спільнота користувачів та добре опрацьована документація, що значно полегшує освоєння й практичне застосування бібліотеки. Scikit-learn спеціалізується на алгоритмах машинного навчання для розв’язання задач навчання з учителем (класифікація та регресія) і без учителя (кластеризація, зменшення розмірності, виявлення аномалій) [19].

Таким чином, Scikit-learn є потужним інструментом машинного навчання в Python і часто обирається науковцями та розробниками, що працюють з аналізом даних, завдяки простоті використання, широким функціональним можливостям і високій надійності.

Plotly [20] - це бібліотека Python з відкритим кодом, призначена для створення інтерактивних веб-візуалізацій даних. На відміну від Matplotlib, який переважно орієнтований на статичні діаграми, Plotly дає змогу будувати графіки та діаграми, з якими користувач може взаємодіяти: масштабувати, пересувати область перегляду, наводити курсор для перегляду точних значень, вмикати й вимикати окремі серії даних тощо. Така інтерактивність робить Plotly особливо корисним інструментом для дослідницького аналізу даних, візуальної аналітики та побудови інформаційних панелей моніторингу.

Бібліотека підтримує великий набір типів візуалізацій: лінійні та стовпчикові діаграми, діаграми розсіювання, теплові карти, boxplot, 3D-графіки, карти та багато інших. Plotly добре інтегрується з Jupyter Notebook і веб-фреймворками, що дає можливість легко вбудовувати інтерактивні графіки у звіти, веб застосунки та дашборди для наочного подання результатів аналізу даних [20].

Для прогнозування було обрано такі моделі як Prophet, ARIMA.

Prophet - це бібліотека Python з відкритим вихідним кодом для прогнозування часових рядів, розроблена основною командою META, що займається аналізом даних [21].

До її ключових особливостей належать:

– Простота. Prophet має зрозумілий і лаконічний синтаксис, тому її відносно легко опанувати й застосовувати навіть фахівцям, які не мають великого досвіду роботи з часовими рядами чи аналізом даних загалом.

– Автоматизація. Бібліотека автоматизує значну частину етапів процесу прогнозування: вибір моделі, налаштування гіперпараметрів, врахування та параметризацію сезонності. Це зменшує потребу в ручному втручанні й спрощує організацію робочих процесів.

– Опрацювання сезонності. Prophet ефективно виявляє сезонні компоненти часових рядів (тижневі, місячні, річні тренди тощо) й дозволяє явно враховувати ці закономірності в моделі. Завдяки цьому прогнози стають точнішими, особливо для даних, що демонструють виражені сезонні коливання.

– Урахування впливу свят. У моделі можна додатково враховувати вплив державних свят чи інших аномальних подій, додаючи їх як регресори. Це дає змогу коригувати прогнози з огляду на можливі збої, стрибки або нетипову поведінку даних у такі періоди.

Додатковою перевагою бібліотеки Prophet є наявність вбудованих засобів візуального аналізу побудованої моделі. Після навчання моделі формується базовий графік прогнозу, на якому відображаються фактичні значення часового ряду (окремі точки), лінія прогнозу та довірчі інтервали. Такий графік дає змогу швидко оцінити, наскільки добре модель відтворює загальну динаміку процесу, чи потрапляють спостереження всередину довірчого інтервалу та в яких ділянках модель систематично недооцінює або переоцінює рівень показника. Приклад такого графіку зображено на рисунку 1.11.

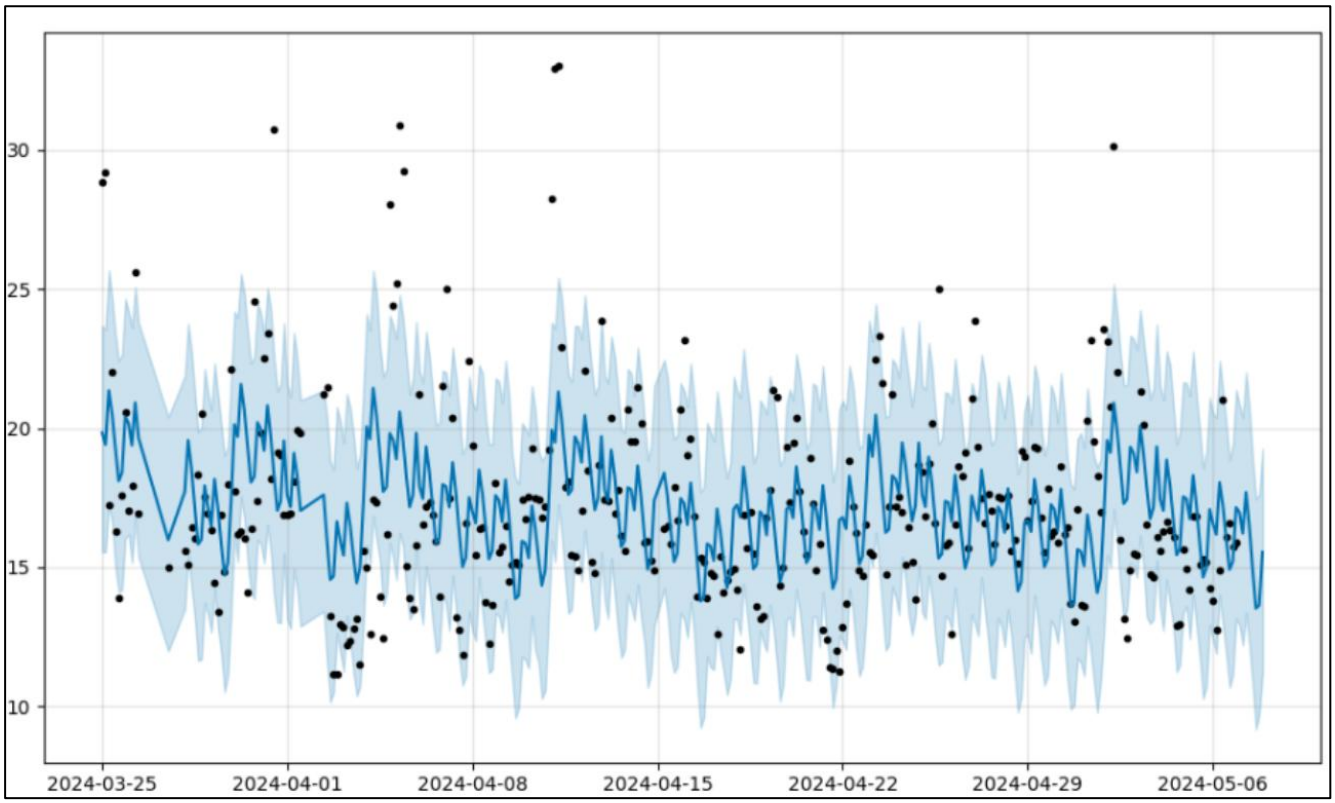

Рисунок 1.11 – Приклад графіку прогнозу

Окремою перевагою Prophet є можливість розкласти прогноз на інтерпретовані компоненти. На рисунку 1.12 зображено графік лінії тренду, який відображає зміну базового рівня часового ряду без урахування періодичних коливань. Такий графік дозволяє відокремити структурні зміни (наприклад, зростання або спад показника внаслідок зовнішніх факторів) від короткострокових флуктуацій.

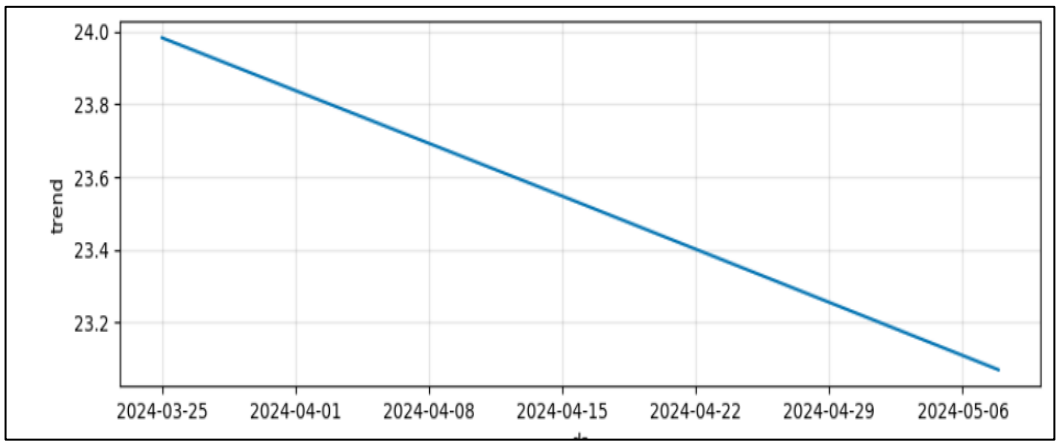

Рисунок 1.12 – Приклад графіку лінії тренду

На рисунку 1.13 подано добову сезонну компоненту, що показує типову зміну значення показника протягом доби; подібні криві часто інтерпретують через режим роботи системи, добові цикли попиту, навантаження тощо.

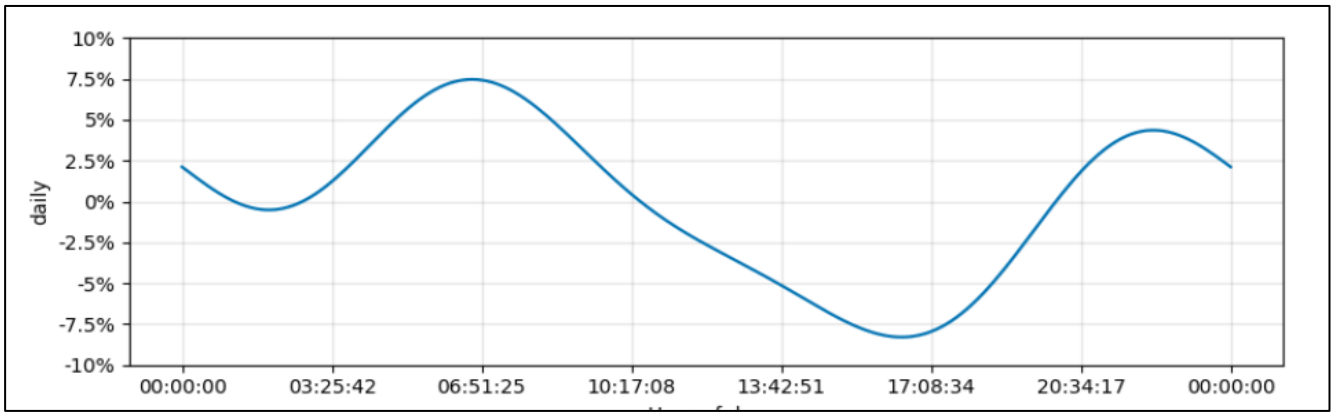

Рисунок 1.13 – Приклад графіку добової сезонності

Рисунок 1.14 ілюструє тижневу сезонність, яка відображає систематичні відмінності між днями тижня та дає змогу виявити, наприклад, характерні «робочі» й «вихідні» патерни.

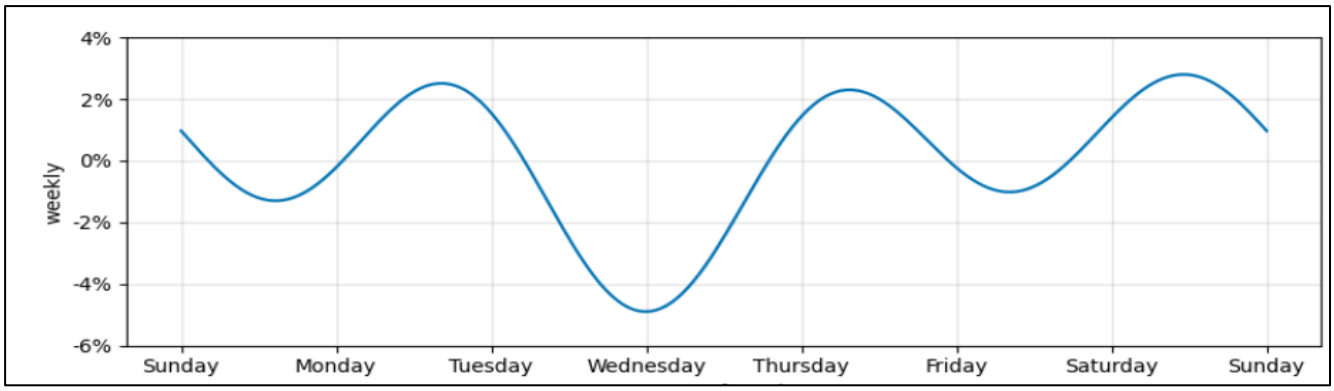

Рисунок 1.14 – Приклад графіку тижневої сезонності

Ще одна важлива складова моделі Prophet - компонент «holidays» або спеціальних подій, який показано на рисунку 1.15. У класичному випадку він використовується для моделювання впливу державних свят, вихідних, акцій, кампаній і т.п., що призводять до різких відхилень від типового режиму роботи часового ряду. Графік цієї компоненти дозволяє кількісно оцінити, наскільки і в

який бік такі дні зміщують очікуване значення показника відносно базового тренду та сезонності.

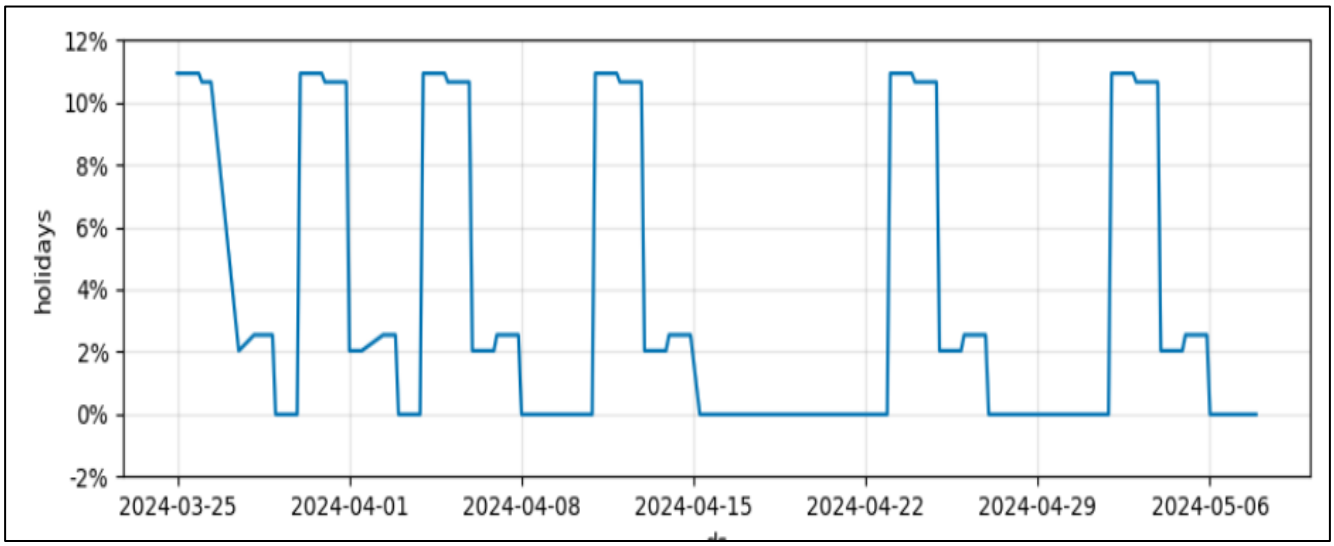

Рисунок 1.15 – Приклад графіку holidays-ефекту

Рисунок 1.16 демонструє алгоритм процесу прогнозування Prophet [22].

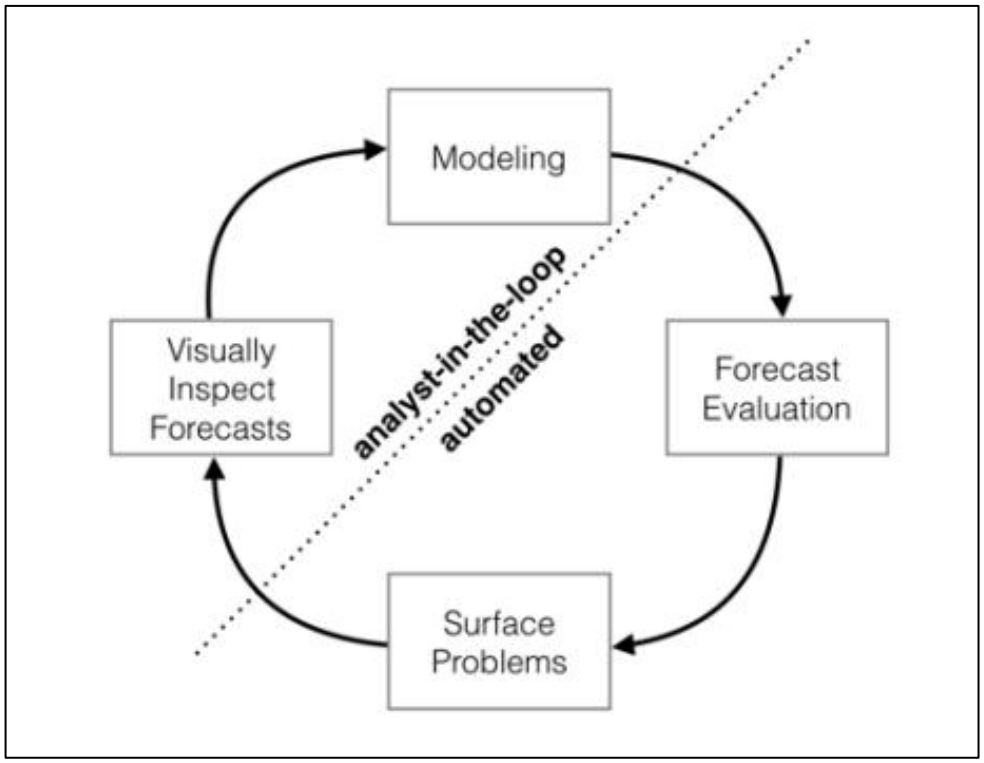

Рисунок 1.16 - Діаграма процесу прогнозування Prophet

Ця діаграма відображає цикл розроблення та вдосконалення моделі, який охоплює чотири ключові етапи:

– Create/Update Model (Створення/Оновлення моделі). На цьому кроці формується нова модель або вдосконалюється вже наявна. Сюди належать вибір алгоритму, налаштування гіперпараметрів, навчання моделі на тренувальних даних та інші пов’язані з цим процедури.

– Evaluate Model (Оцінювання моделі). Після побудови чи оновлення модель необхідно перевірити. Для цього використовують валідаційні або тестові дані, на яких оцінюють її продуктивність. Як показники якості можуть застосовуватися точність, коефіцієнт детермінації (R^2), середня абсолютна похибка (MAPE), корінь середньоквадратичної похибки (RMSE) тощо.

– Surface Problems (Виявлення проблем). На цьому етапі аналізують результати оцінювання, щоб виявити потенційні недоліки або слабкі місця моделі. Це може включати розбір помилок, пошук аномалій або визначення ділянок, у яких модель дає особливо неточні прогнози.

– Visually Inspect Model (Візуальна перевірка моделі). Візуалізація роботи моделі допомагає краще зрозуміти її поведінку й додатково ідентифікувати проблеми. Для цього будують графіки фактичних і прогнозованих значень, аналізують залишки, створюють діаграми важливості ознак та інші види візуального представлення результатів.

Модель ARIMA (Autoregressive Integrated Moving Average – авторегресійна інтегрована ковзного середнього) є однією з найпоширеніших і найефективніших статистичних моделей для прогнозування часових рядів. Вона належить до сімейства лінійних моделей і узагальнює три основні компоненти [23].

– AR (Autoregression - авторегресія) – компонент, що описує залежність поточного значення часового ряду від його власних попередніх (відстаючих) значень. Параметр p (порядок авторегресії) вказує, скільки попередніх спостережень включається до моделі [23].

– I (Integrated – інтегрована складова) - процедура диференціювання часового ряду, яка застосовується для усунення тренду та досягнення стаціонарності.

Параметр d визначає кількість разів, скільки потрібно продиференціювати ряд (ступінь диференціювання) [23].

– MA (Moving Average – ковзне середнє) – компонент, що моделює залежність поточного значення від залишкових помилок попередніх періодів моделі ковзного середнього. Параметр q визначає розмір вікна ковзного середнього (порядок ковзного середнього) [23].

Таким чином, повна модель позначається як ARIMA(p, d, q), де:

p - порядок авторегресійної складової (кількість відстаючих спостережень, що включаються);

d - ступінь диференціювання, необхідний для перетворення нестаціонарного ряду на стаціонарний;

q - порядок складової ковзного середнього (кількість залишкових помилок, що враховуються).

1.4 Висновки

У першому розділі проведено аналіз предметної області, сучасних підходів до оцінювання якості атмосферного повітря та впливу пилу Сахари на екологічний стан України. Розглянуто міжнародні й національні індекси якості повітря, особливості роботи сервісів громадського моніторингу SaveEcoBot та EcoCity, а також проаналізовано зафіксовані у 2024 році пилові епізоди, пов’язані з перенесенням частинок із Сахари на територію України. Окрему увагу приділено огляду мовних засобів і бібліотек Python для аналізу часових рядів та побудови прогнозних моделей, зокрема бібліотек NumPy, Pandas, Matplotlib, Plotly, Scikit-learn, а також моделей ARIMA та Prophet. На основі проведеного аналізу обґрунтовано вибір сервісу EcoCity як джерела вхідних даних і моделей ARIMA та Prophet як основних інструментів прогнозування, що стало теоретичною основою для подальшої підготовки даних та розробки інформаційної технології.

2 ПІДГОТОВКА ТА РОЗВІДУВАЛЬНИЙ АНАЛІЗ ДАНИХ

2.1 Імпорт даних та початкова візуалізація

Для розв’язання поставленої задачі було зібрано дані з сервісу EcoCity та на їх основі сформовано відповідний набір даних Air Quality Monitoring from EcoCity [24], а також підготовлено ноутбуки в середовищі Kaggle [25, 26]. Датасет містить такі поля:

- id_station – ідентифікаційний код станції на сайті EcoCity;
- location – координати відповідної станції моніторингу якості повітря;
- date – календарна дата, на яку було зафіксовано вимірювання;
- time_point - індекс або множник, що дає змогу обчислити точний час усередині конкретного дня;
- indicator – назва показника концентрації забруднення;
- dimension – одиниця вимірювання рівня забруднення;
- value – числове значення рівня забруднення..

Було сформовано окремий DataFrame з інформацією про кожну станцію Вінницької області. Цей набір даних використовується для зіставлення станцій із їхніми поштовими адресами, що забезпечує більш однозначне розуміння того, яка саме станція аналізується в подальших дослідженнях.

Після цього здійснюється зчитування даних із CSV-файлів та доповнення наявних записів інформацією про адреси станцій (рис. 2.1). Спочатку завантажується основний файл, який містить часові ряди вимірювань якості повітря з усіх станцій моніторингу. Далі зчитується допоміжний файл із довідковою інформацією про станції, де для кожного ідентифікатора станції вказана точна адреса розташування, а також інші метадані.

На наступному етапі обидві таблиці об’єднуються за спільним ідентифікатором станції таким чином, щоб до кожного запису з вимірюваннями автоматично додавалася відповідна адреса. Об’єднання виконується з збереженням усіх рядків основного набору даних: якщо для якоїсь станції адреса в довідковій

таблиці відсутня, відповідне поле просто залишається порожнім. Після об’єднання зайвий службовий стовпець, що виник у процесі злиття, видаляється.

У результаті формується єдиний набір даних, у якому кожне вимірювання якості повітря доповнене чіткою локацією станції, де воно було зафіксоване. Це дає змогу надалі проводити аналіз і візуалізацію результатів з урахуванням географічного розташування станцій, групувати дані за районами чи вулицями, а також будувати карти забруднення повітря.

```
column_names = ['id_station', 'location', 'date', 'time_point', 'Column6', 'indicator', 'dimension', 'value']
stations = pd.read_csv('/kaggle/input/air-quality-monitoring-from-ecocity/ECOCITY_Archive_651_561_2024-03-25_2024-05-10.csv', sep=',', names=column_names)

# Визначення назв колонок для about_station
column_names_a = ["id_saveecobot", "id_ecocity", "network", "locality", "address", "start_date", "lat", "lng", "notes", "source"]
about_stations = pd.read_csv('/kaggle/input/eco-city-stations-about/ecocity_about_stations_2024.csv', header=None, names=column_names_a)

stations['id_station'] = pd.to_numeric(stations['id_station'], errors='coerce').dropna().astype(int)
about_stations['id_ecocity'] = pd.to_numeric(about_stations['id_ecocity'], errors='coerce').dropna().astype(int)
# Перевірка зчитаних даних
print(stations)
stations.info()

result = pd.merge(stations, about_stations[['id_ecocity', 'address']], left_on='id_station', right_on='id_ecocity', how='left')

result = result.drop(columns=['id_ecocity'])

result
```

Рисунок 2.1 – Блок коду, який відповідає за зчитування даних

Переглянемо наявні дані на рисунку 2.2 та інформацію про них на рисунку 2.3.

	id_station	location	date	time_point	Column6	\
0	1315	49.23327 28.409161	2024-03-25	1	18	
1	1315	49.23327 28.409161	2024-03-25	2	20	
2	1315	49.23327 28.409161	2024-03-25	3	18	
3	1315	49.23327 28.409161	2024-03-25	4	20	
4	1315	49.23327 28.409161	2024-03-25	5	20	
...	...	...	...	...	...	
19642	1876	49.2048561 28.5288355	2024-05-10	68	20	
19643	1876	49.2048561 28.5288355	2024-05-10	69	20	
19644	1876	49.2048561 28.5288355	2024-05-10	70	20	
19645	1876	49.2048561 28.5288355	2024-05-10	71	20	
19646	1876	49.2048561 28.5288355	2024-05-10	72	20	

	indicator	dimension	value
0	PM1.0	ug/m3	26.8422
1	PM1.0	ug/m3	25.4215
2	PM1.0	ug/m3	24.4250
3	PM1.0	ug/m3	22.8410
4	PM1.0	ug/m3	22.6225
...	...	...	...
19642	PM1.0	ug/m3	6.3255
19643	PM1.0	ug/m3	5.8295
19644	PM1.0	ug/m3	3.7260
19645	PM1.0	ug/m3	3.9925
19646	PM1.0	ug/m3	4.0595

Рисунок 2.2 – Наявні дані

```
[19647 rows x 8 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 19647 entries, 0 to 19646
Data columns (total 8 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   id_station  19647 non-null  int64
 1   location    19647 non-null  object
 2   date        19647 non-null  object
 3   time_point  19647 non-null  int64
 4   Column6     19647 non-null  int64
 5   indicator   19647 non-null  object
 6   dimension   19647 non-null  object
 7   value       19647 non-null  float64
dtypes: float64(1), int64(3), object(4)
memory usage: 1.2+ MB
```

Рисунок 2.3 – Інформація про дані

Згідно інформації про дані, пропущених записів немає.

В результаті вищезгаданих маніпуляцій з даними отримуємо таблицю зображену на рисунку 2.4.

[27]:

	id_station	location	date	time_point	Column6	indicator	dimension	value	address
0	1315	49.23327 28.409161	2024-03-25	1	18	PM1.0	ug/m3	26.8422	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
1	1315	49.23327 28.409161	2024-03-25	2	20	PM1.0	ug/m3	25.4215	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
2	1315	49.23327 28.409161	2024-03-25	3	18	PM1.0	ug/m3	24.4250	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
3	1315	49.23327 28.409161	2024-03-25	4	20	PM1.0	ug/m3	22.8410	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
4	1315	49.23327 28.409161	2024-03-25	5	20	PM1.0	ug/m3	22.6225	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
...	...	...	...	...	...	...	...	...	...
19642	1876	49.2048561 28.5288355	2024-05-10	68	20	PM1.0	ug/m3	6.3255	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19643	1876	49.2048561 28.5288355	2024-05-10	69	20	PM1.0	ug/m3	5.8295	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19644	1876	49.2048561 28.5288355	2024-05-10	70	20	PM1.0	ug/m3	3.7260	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19645	1876	49.2048561 28.5288355	2024-05-10	71	20	PM1.0	ug/m3	3.9925	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19646	1876	49.2048561 28.5288355	2024-05-10	72	20	PM1.0	ug/m3	4.0595	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru

19647 rows × 9 columns

Рисунок 2.4 – Вигляд таблиці із адресою

На наступному етапі проводиться перевірка унікальності станцій моніторингу, наявних у наборі даних. Для цього визначається кількість унікальних значень у стовпці id_station, а також формується перелік усіх ідентифікаторів станцій, що зустрічаються в таблиці.

Отриманий результат, що зображений на рисунку 2.5, показує, що в оброблюваному наборі даних містяться вимірювання лише з шести станцій моніторингу якості повітря, а саме зі станцій із ідентифікаторами 1315, 1612, 1769, 1864, 1872 та 1876.

Це дозволяє переконатися, що подальший аналіз і прогнозування будуть базуватися на даних обмеженої, але чітко визначеної кількості пунктів спостереження, а також дає змогу в разі потреби перевірити відповідність цих ідентифікаторів реальним станціям у довідковій таблиці або на карті міста.

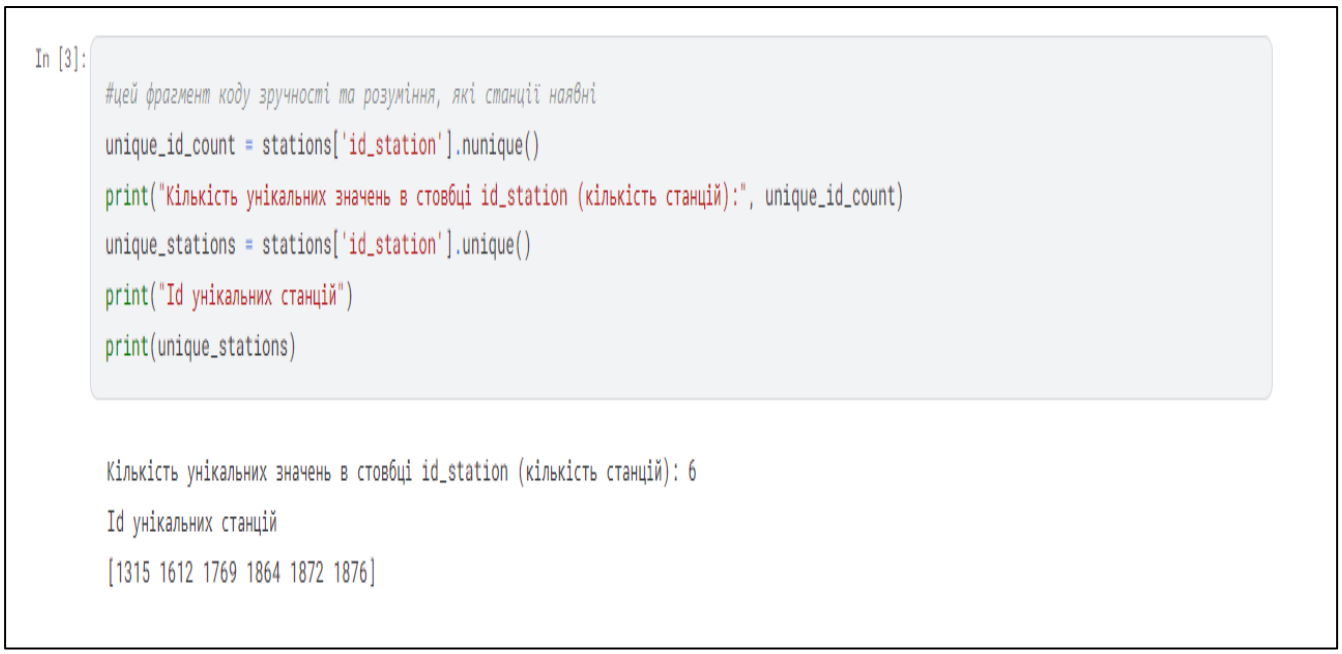

Рисунок 2.5 – Визначення унікальних станцій

Для подальшого аналізу реалізуємо функцію get_data, яка виконує вибірку та попередню обробку даних для заданої станції моніторингу. На основі полів date та time_point формується єдина часова мітка datetime (дата та час вимірювання з кроком 20 хвилин), після чого дані сортуються за часом і подаються у вигляді часової послідовності значень PM_1. Для кожної станції будується графік зміни концентрації PM_1 у часі, що дозволяє візуально оцінити наявність пікових епізодів забруднення.

Для автоматичного виявлення аномалій застосуємо квантильний підхід [27]. З усіх виміряних значень PM_1 для обраної станції формується емпіричний розподіл,

після чого як порогове значення T_q береться q-квантиль цього розподілу. У даній роботі використано рівень q=0,98, що відповідає верхнім 2% найбільших концентрацій. Вимірювання, для яких виконується нерівність $PM_1(t)>T_q$, розглядаються як кандидати в аномальні значення, пов'язані з потенційними епізодами перенесення пилу Сахари.

Щоб уникнути множинного спрацьовування всередині одного пилового епізоду, кандидати в аномалії переглядаються в хронологічному порядку з використанням «вікна охолодження» тривалістю 96 годин. Перша точка, що перевищила поріг, фіксується як початок епізоду, а всі подальші перевищення впродовж наступних чотирьох діб до нового епізоду не додаються до переліку. Таким чином, для кожної станції формується компактний список дат початку аномальних подій. Надалі ці дати використовуються для аналізу пилових епізодів та налаштування моделей прогнозування.

За допомогою реалізованої функції get_data виконаємо автоматичне виявлення аномальних значень концентрацій PM_1 для станцій наявних у датаесті. На рисунку 2.6 зображено виклик функції для станції 1315. Зафіксовано сім унікальних аномальних подій (25.03, 30.03, 04.04, 08.04, 25.04, 02.05, 10.05.2024 р.).

Перші два піки - 25 та 30 березня - чітко збігаються з першим Сахарським епізодом. Пік 25 квітня відповідає другому епізоду перенесення пилу Сахари наприкінці квітня. Аномалії на початку квітня (04 та 08.04) можна інтерпретувати як підвищений фон запиленості після першого епізоду або вплив локальних джерел (транспорт, спалювання рослинності). Піки 02 та 10 травня мають, імовірно, суто локальну природу і відображають короткочасні епізоди зростання концентрацій у міській забудові.

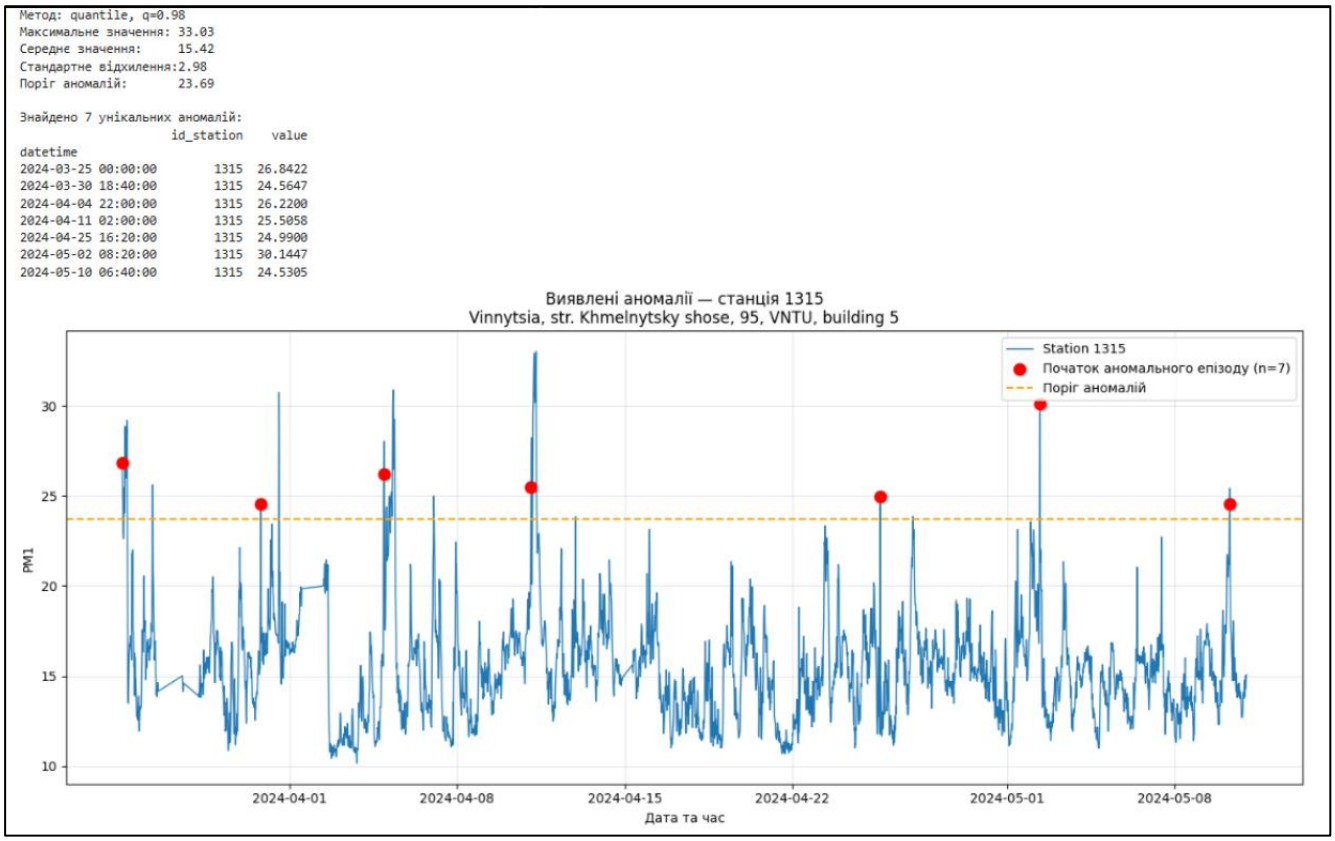

Рисунок 2.6 – Графік виявлених аномалій для станції 1315

Для станції 1612 також виявлено сім аномалій (25.03, 29.03, 05.04, 19.04, 25.04, 29.04, 06.05.2024 р.), що зображено на рисунку 2.7.

Піки 25 та 29 березня однозначно пов’язані з першим епізодом Сахарського пилу. Аномалія 25 квітня збігається за часом із другим пиловим епізодом і свідчить про його прояв над даною точкою спостереження. Підвищення 05.04 та 19.04 можна розглядати як наслідок локальних процесів (накопичення пилу у приземному шарі, транспортні викиди), а також як можливі «хвости» першого епізоду. Події 29.04 та 06.05 не співпадають із відомими датами перенесення і, ймовірно, зумовлені локальними умовами забруднення.

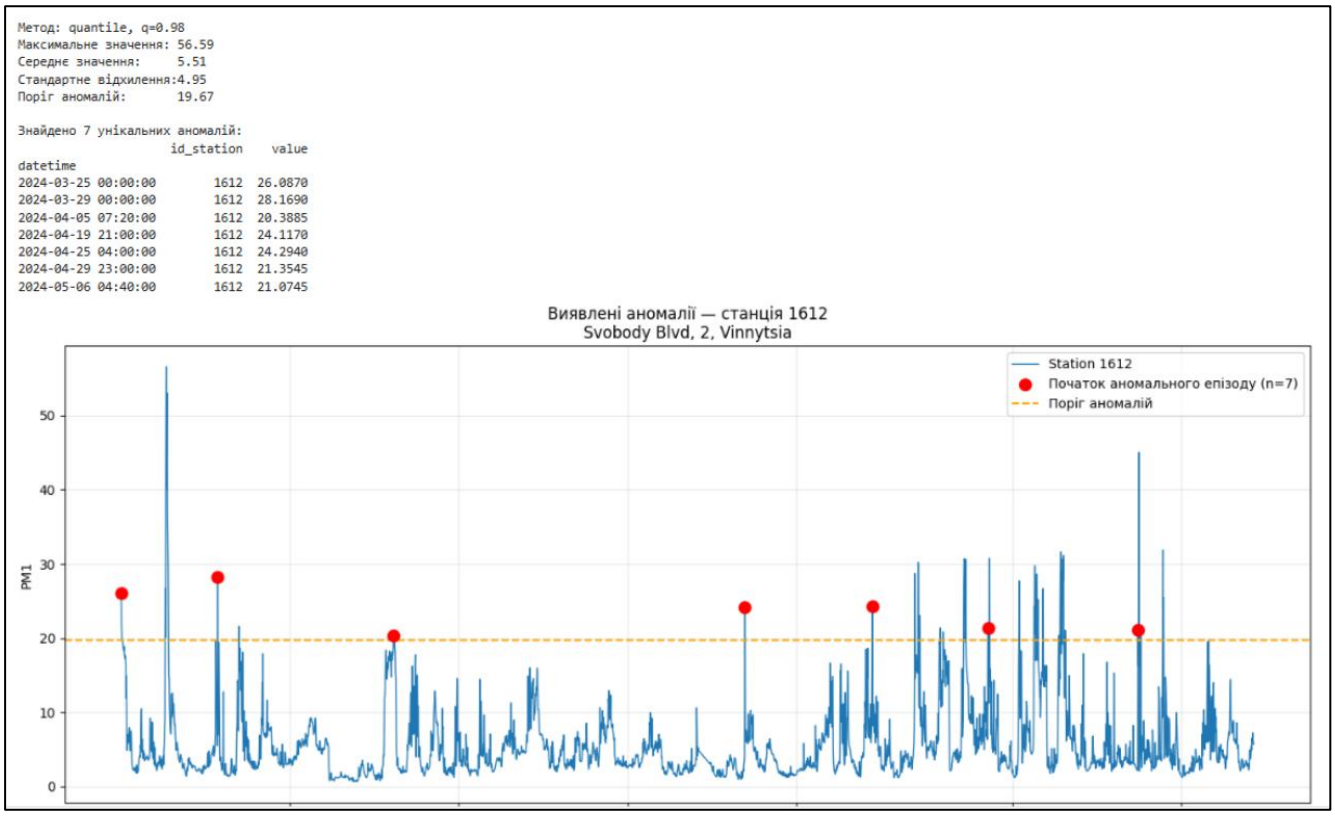

Рисунок 2.7 – Графік виявлених аномалій для станції 1612

На станції 1769 зафіксовано сім аномальних подій (25.03, 29.03, 05.04, 10.04, 24.04, 30.04, 10.05.2024 р.), що зображено на рисунку 2.8.

Піки 25 та 29 березня узгоджуються з першим Сахарським епізодом. Аномалія 24 квітня припадає майже точно на другий епізод перенесення пилу Сахари (24 - 25.04) і відображає його прояв у даному мікрорайоні.

На рисунку 2.9 зображено графічний аналіз для станції 1864. Виявлено вісім аномальних епізодів (25.03, 29.03, 05.04, 08.04, 21.04, 24.04, 01.05, 10.05.2024 р.).

Перші два піки - 25 та 29 березня - відповідають вищезгаданому епізоду перенесення пилу. Аномалія 24 квітня співпадає з другим пиловим епізодом і може розглядатися як його основний прояв для цієї станції.

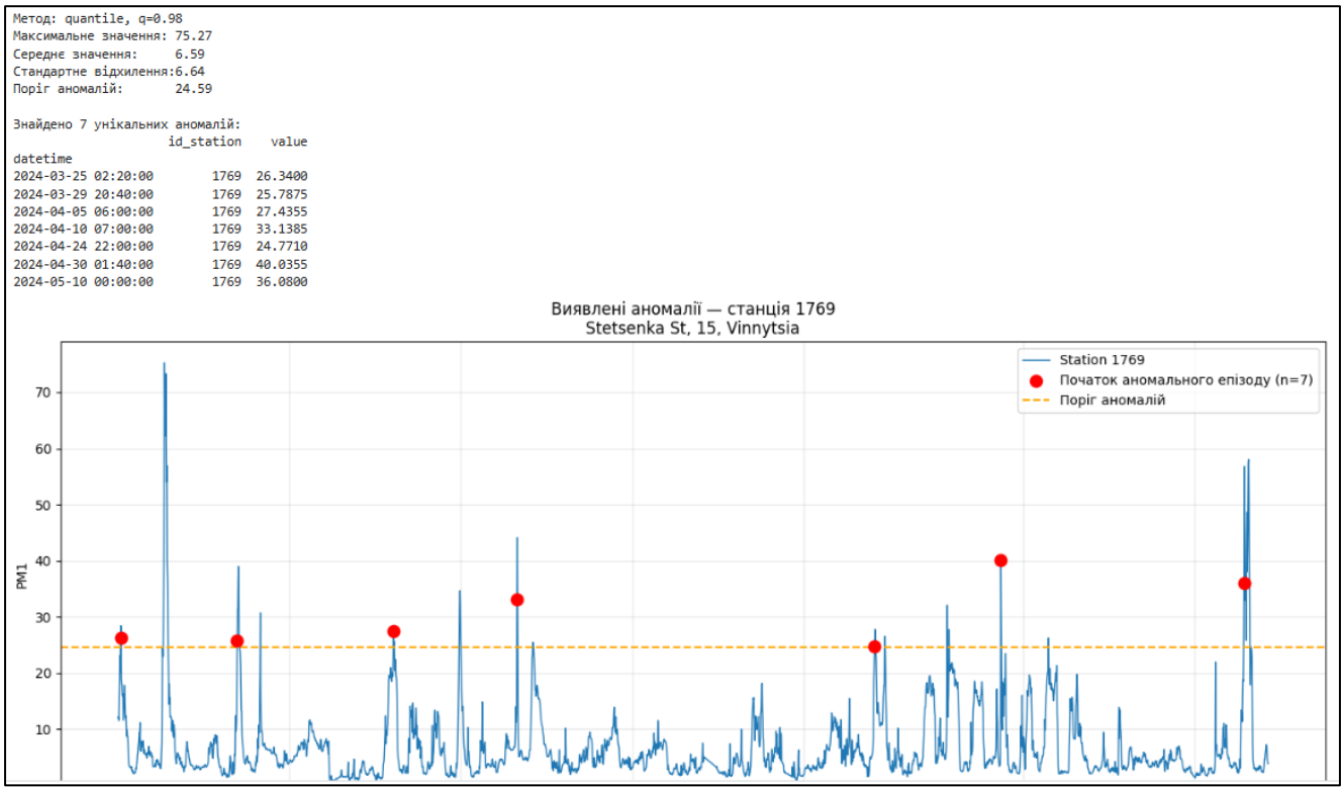

Рисунок 2.8 – Графік даних зі списком дат аномалій станції 1769

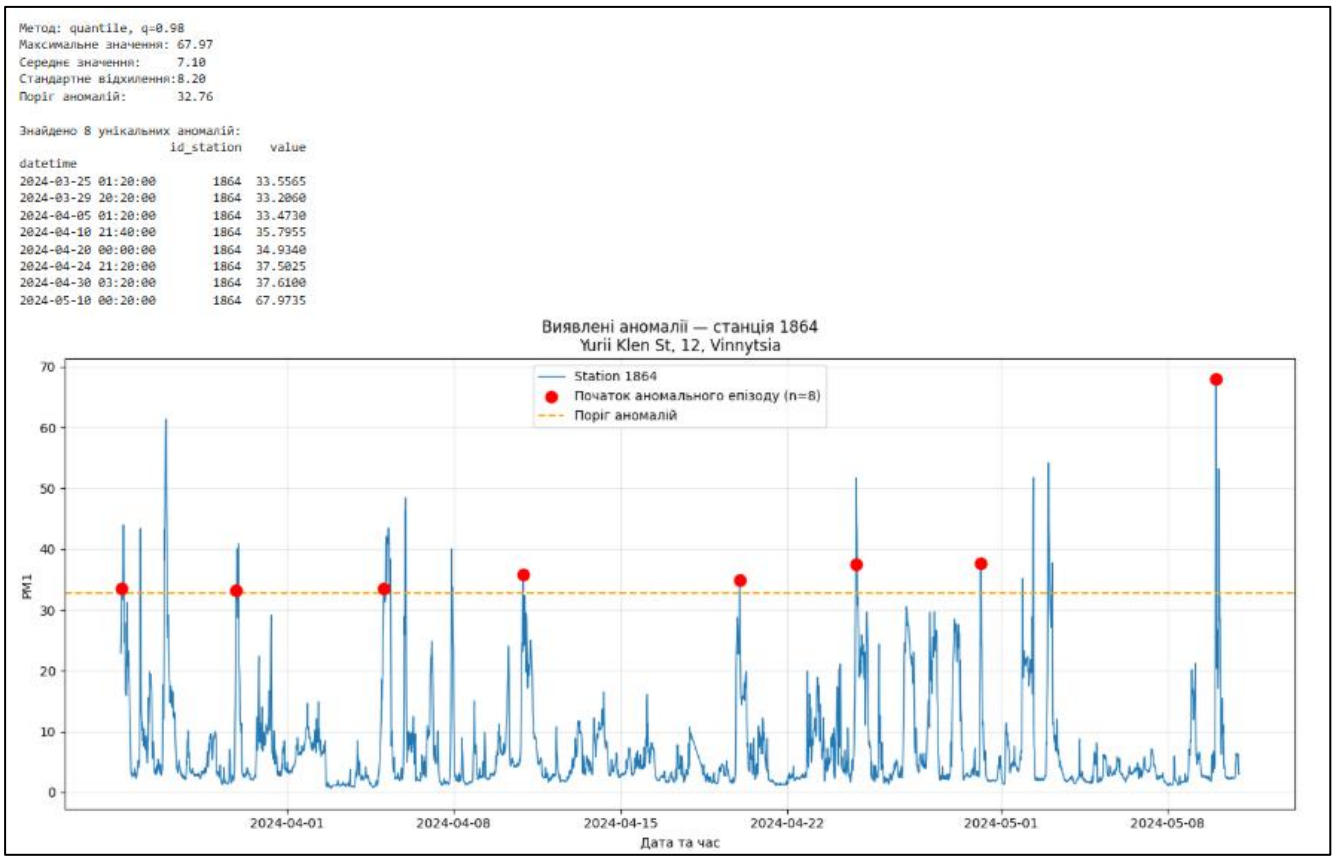

Рисунок 2.9 – Графік даних зі списком дат аномалій станції 1864

Аналізи для станцій, 1872 та 1876, які зображені на рисунках 2.10 та 2.11 відповідно, показують схожу тенденцію, що й усі інші.

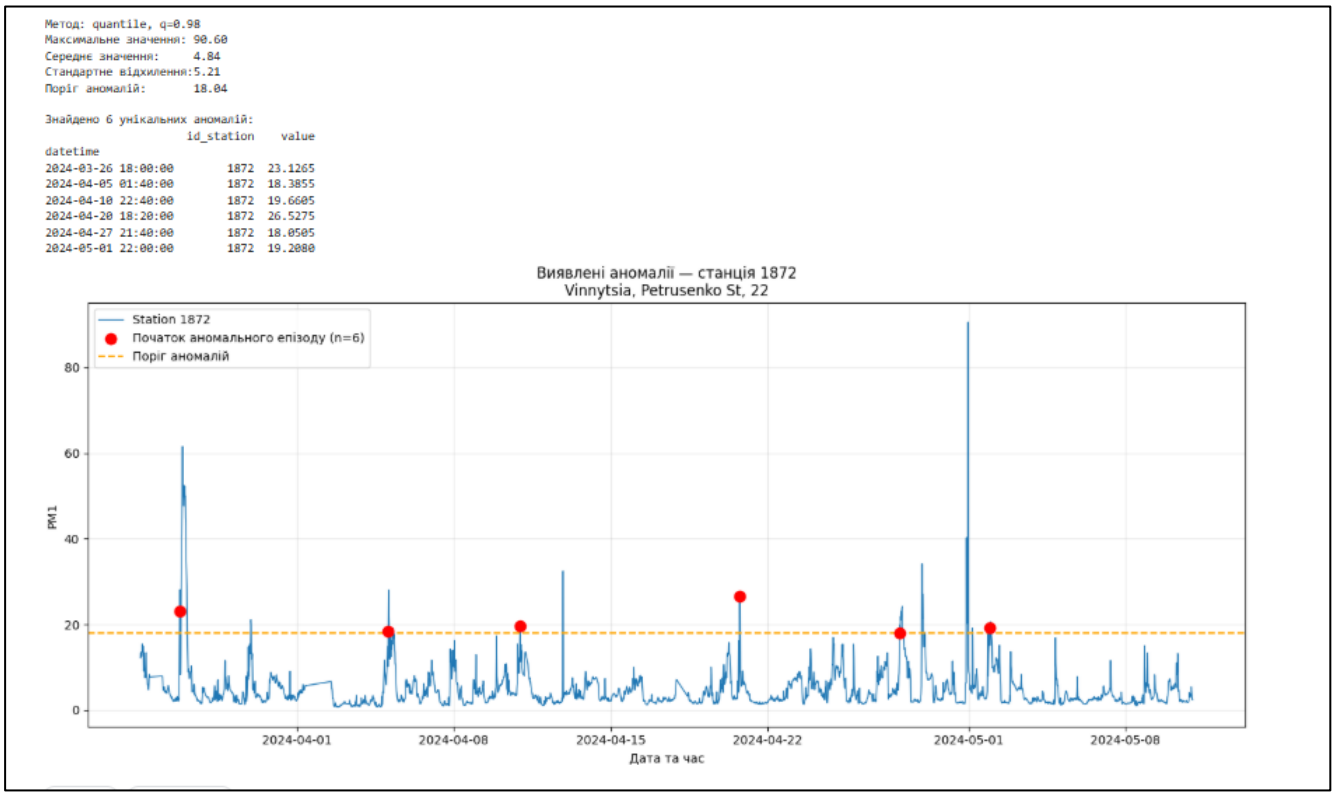

Рисунок 2.10 – Графік даних зі списком дат аномалій станції 1872

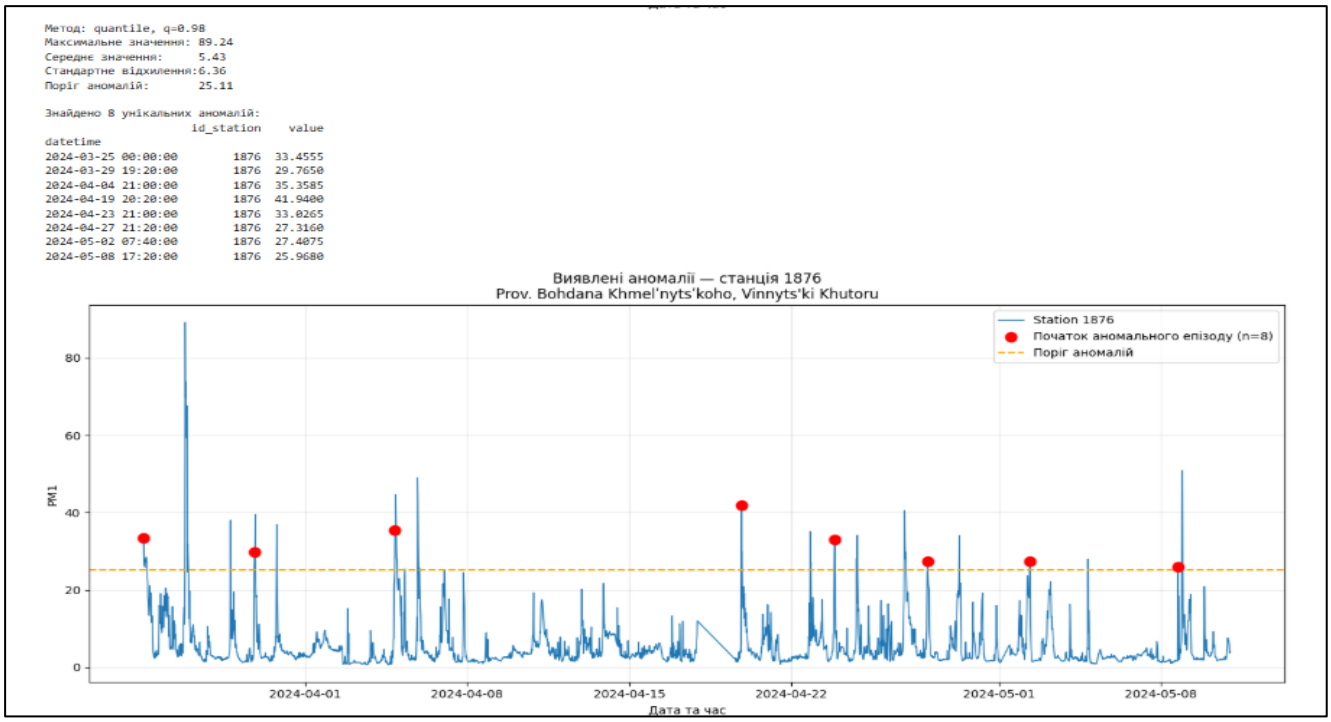

Рисунок 2.11 – Графік даних зі списком дат аномалій станції 1876

Порівняння часових рядів для всіх станцій показало узгодженість аномалій за часом. Більшість станцій міста Вінниці зафіксували різке зростання концентрацій PM_1 у вищезгадані дати, що підтверджує вплив двох епізодів перенесення пилу Сахари. Інші виявлені аномалії, які не збігаються з цими періодами - інтерпретуються як наслідок локальних джерел забруднення та специфічних метеорологічних умов.

2.2 Просторовий аналіз

Було розроблено ноутбук Sahara's Dust in the Region - 2D Analysis [28], призначений для візуалізації даних за конкретно обрані дату та годину. Для побудови графіків у ньому використовуються дані з набору Air Quality Monitoring from EcoCity [24].

Як видно з рисунка 2.2, у вихідному датасеті координати не подано окремими стовпцями, а містяться в одному полі. Тому було створено спеціальну функцію для виокремлення широти (lat) та довготи (lng); фрагмент відповідного коду наведено на рисунку 2.12.

```
def extract_location_data(df, id_station=None):
    if id_station is not None:
        station_data = df[df['id_station'] == id_station].copy()
        if not station_data.empty:
            # Об'єднання колонок 'date' і 'time_point' у нову колонку 'datetime'
            station_data['datetime'] = pd.to_datetime(station_data['date']) + pd.to_timedelta((sta
tion_data['time_point'] - 1) * 20, unit='m')
            station_data[['lat', 'lng']] = station_data['location'].str.split(expand=True)
            # Видалення колонок 'location', 'date' та 'time_point'
            return station_data.drop(columns=['location', 'date', 'time_point'])
        else:
            print(f"Station ID {id_station} does not exist in the dataset.")
            return pd.DataFrame()
    else:
        df[['lat', 'lng']] = df['location'].str.split(expand=True)
        # Об'єднання колонок 'date' і 'time_point' у нову колонку 'datetime'
        df['datetime'] = pd.to_datetime(df['date']) + pd.to_timedelta((df['time_point'] - 1) * 20,
unit='m')
        # Видалення колонок 'location', 'date' та 'time_point'
        return df.drop(columns=['location', 'date', 'time_point'])
```

Рисунок 2.12 – Функція для розділення координат

Сформуємо нову вибірку для подальших маніпуляцій з нею в межах цього ноутбуку, виділивши лише потрібні поля для просторового аналізу (рис. 2.13)

```
df = data[['id_station', 'datetime', 'value', 'lat', 'lng']]
df.columns = ['id_station', 'ds', 'value', 'lat', 'lng']
df['network'] = 'Eco-City'
df
```

	id_station	ds	value	lat	lng	network
0	1315	2024-03-25 00:00:00	26.8422	49.23327	28.409161	Eco-City
1	1315	2024-03-25 00:20:00	25.4215	49.23327	28.409161	Eco-City
2	1315	2024-03-25 00:40:00	24.4250	49.23327	28.409161	Eco-City
3	1315	2024-03-25 01:00:00	22.8410	49.23327	28.409161	Eco-City
4	1315	2024-03-25 01:20:00	22.6225	49.23327	28.409161	Eco-City
...	...	...	...	...	...	...
19642	1876	2024-05-10 22:20:00	6.3255	49.2048561	28.5288355	Eco-City
19643	1876	2024-05-10 22:40:00	5.8295	49.2048561	28.5288355	Eco-City
19644	1876	2024-05-10 23:00:00	3.7260	49.2048561	28.5288355	Eco-City
19645	1876	2024-05-10 23:20:00	3.9925	49.2048561	28.5288355	Eco-City
19646	1876	2024-05-10 23:40:00	4.0595	49.2048561	28.5288355	Eco-City

Рисунок 2.13 – Створення нового датафрейму

Для того, щоб задати час візуалізації використаємо стрічку коду datetime_analysis =’ ‘

По заданому часу ми можемо виокремити показники PM_1 з кожної станції (рис 2.14).

```
# Selection data for interpolation
data = df[df['ds']==datetime.datetime.fromisoformat(datetime_analysis)].reset_index(drop=True)
x = data.lng.values
y = data.lat.values
z = data.value.values
fig = plt.figure()
plt.scatter(x, y)
plt.title(f'Stations in Vinnytsia region with data for {indicator_name} in {datetime_analysis}')
display(data)
plt.show()
```

	id_station	ds	value	lat	lng	network
0	1315	2024-03-31	16.8568	49.23327	28.409161	Eco-City
1	1612	2024-03-31	6.5630	49.2177336	28.4497946	Eco-City
2	1769	2024-03-31	6.7130	49.2436876	28.4964322	Eco-City
3	1864	2024-03-31	8.3815	49.2368871	28.5133055	Eco-City
4	1872	2024-03-31	6.4255	49.2419269	28.4620209	Eco-City
5	1876	2024-03-31	4.5600	49.2048561	28.5288355	Eco-City

Рисунок 2.14 – Датафрейм в точно заданий час

Наступним етапом є інтерполяція даних. Під інтерполяцією в просторовому аналізі розуміють процес оцінювання значень у тих точках простору, де безпосередні вимірювання не проводилися, на основі вже відомих значень у сусідніх точках. Завдяки цьому можна формувати безперервні карти та поверхні, які відображають просторовий розподіл обраної характеристики - у нашому випадку рівня забруднення повітря. На рисунку 2.15 наведено фрагмент коду, що реалізує інтерполяцію даних.

```
data['lng'] = data['lng'].astype(float)
data['lat'] = data['lat'].astype(float)
```

```
# Data interpolation
x = x.astype(float)
y = y.astype(float)
f = interp2d(x, y, z, kind='linear')  # 'linear', 'cubic', 'quintic'
f
```

```
<scipy.interpolate._interpolate.interp2d at 0x790156f17700>
```

```
# Calculation of values for a regular network of points
X = np.linspace(data.lng.min()*0.99995, data.lng.max()*1.00005, 100)
Y = np.linspace(data.lat.min()*0.99995, data.lat.max()*1.00005, 100)
Z = f(X, Y)
Z[0]
```

Рисунок 2.15 – Інтерполяція даних

За побудову графіків відповідає фрагмент коду наведений на рисунку 2.16.

```
# Coordinates of stations - from EcoCity or no
xeco = data['lng'].values
yeco = data['lat'].values
numeco = data['id_station'].astype('str').values
```

```
# Visualization
fig = plt.figure(figsize=(12,10))
plt.contourf(X, Y, Z)

plt.scatter(xeco, yeco, c='k', s=100, label='EcoCity')
for i in range(len(xeco)):
    plt.annotate("  "+numeco[i], xy=(xeco[i], yeco[i]), textcoords='data')

plt.axis()
plt.title(f'Stations in Vinnytsia region with hourly average data for {indicator_name} in {datetim
e_analysis} (maximum value = {round(data.value.max(),2)})')
plt.colorbar()
plt.legend(loc='best')
plt.grid()
plt.show()
```

Рисунок 2.16 – Фрагмент коду, який відповідає за побудову графіків

Прикладами роботи ноутбуку є графіки на рисунках 2.17 та 2.18.

На графіку рівні показника якості повітря (PM_1) відображено у вигляді зон, зафарбованих у різні кольори. Кожен колір відповідає певному діапазону значень цього показника.

Контурні лінії, що проходять між кольоровими зонами, позначають межі переходу між різними рівнями концентрацій.

Такий графік дає змогу наочно оцінити просторовий розподіл показника PM_1 на території Вінницької області в обраний момент часу, виявити потенційно проблемні ділянки та слугувати основою для подальшого аналітичного опрацювання.

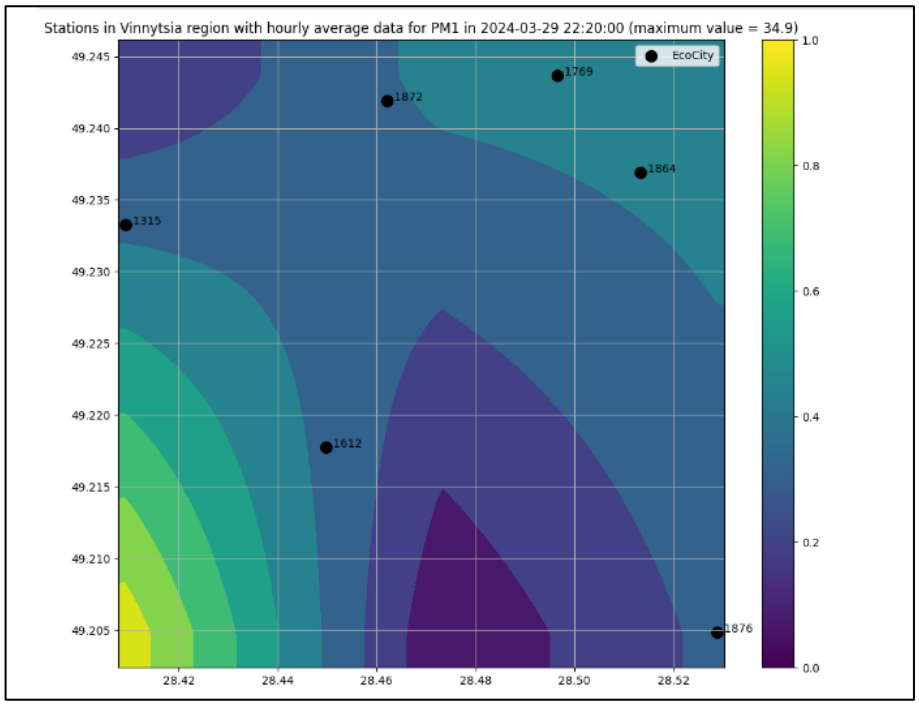

Рисунок 2.17 – Просторовий аналіз даних станом на 22:20 29-03-2024

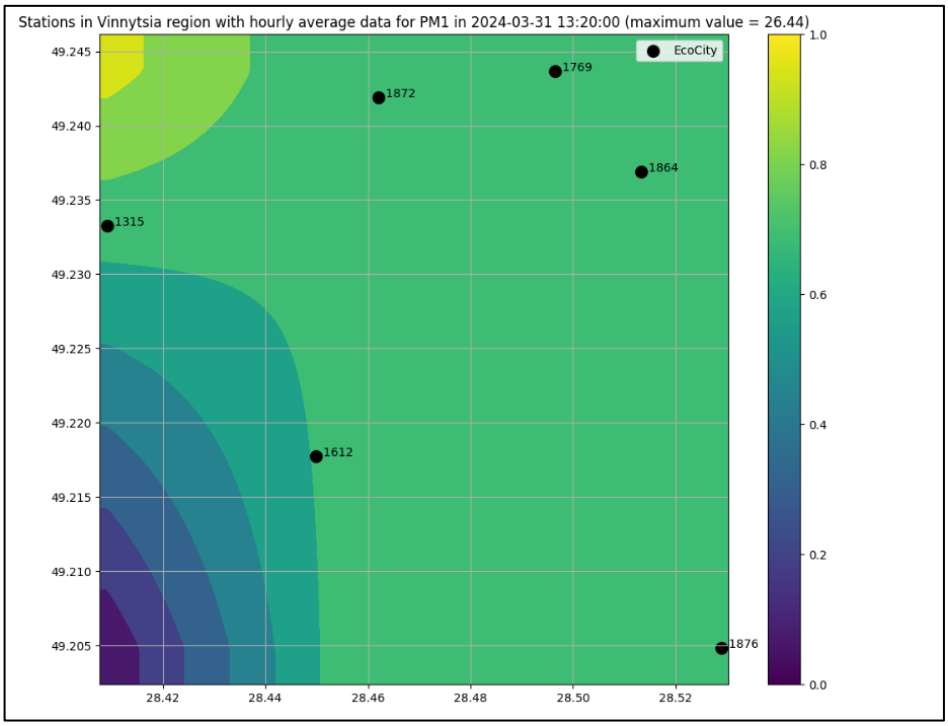

Рисунок 2.18 – Просторовий аналіз даних станом на 13:20 31-03-2024

2.3 Розвідувальний аналіз даних

Для станції 1315 було проведено детальний розвідувальний аналіз даних, результати якого представлені у вигляді кількох візуалізацій.

На гістограмі розподілу значень PM_1, що зображено на рисунку 2.19 видно, що дані мають асиметричний розподіл, зміщений вправо. Більшість спостережень зосереджена у діапазоні від 13 до 17 мкг/м^3, що відповідає періодам відносно чистого повітря. Водночас, невелика кількість спостережень перевищує 25–30 мкг/м^3, що може вказувати на поодинокі епізоди різкого підвищення концентрації дрібнодисперсного пилу, ймовірно спричинені пиловими бурями або локальними джерелами забруднення.

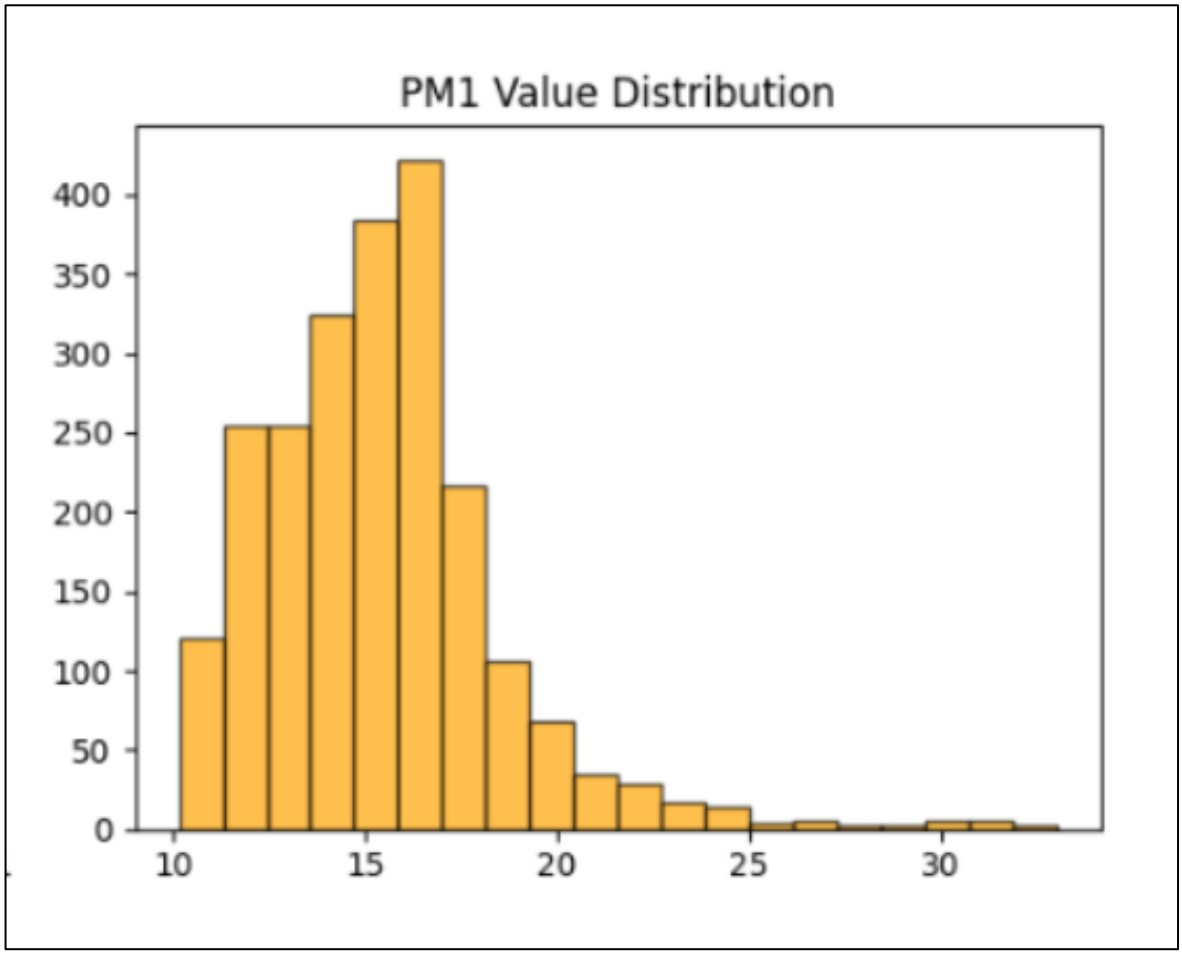

Рисунок 2.19 - Гістограма розподілу значень PM1

Боксплот, що зображено на рисунку 2.20, підтверджує наявність таких аномальних значень - значна кількість точок розташована праворуч за межами «вусів», що свідчить про наявність викидів у даних. Основна частина спостережень зосереджена навколо медіани близько 15 мкг/м^3, тоді як максимальні спостереження перевищують 30 мкг/м^3. Це характерно для екологічних даних, де періоди спокійного стану повітря змінюються короткочасними епізодами сильного забруднення.

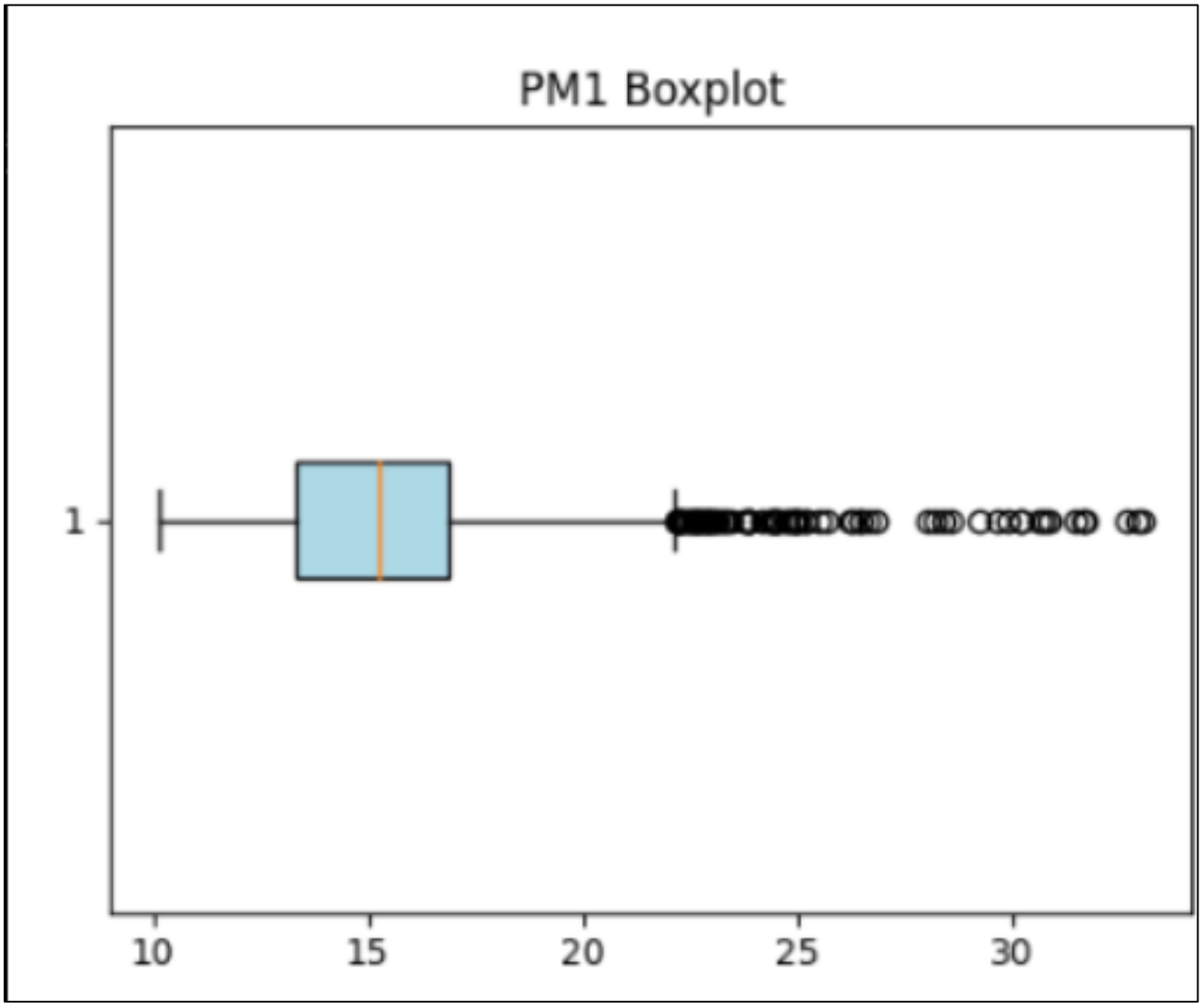

Рисунок 2.20 – Боксплот розподілу значень PM1

Графік автокореляційної функції, що зображено на рисунку 2.21, показує поступове спадання коефіцієнтів автокореляції зі зростанням лага, що є типовою

ознакою наявності інерційності процесу. Це означає, що поточні значення концентрації пилу значною мірою залежать від попередніх, тобто процес має пам’ять і не є повністю випадковим.

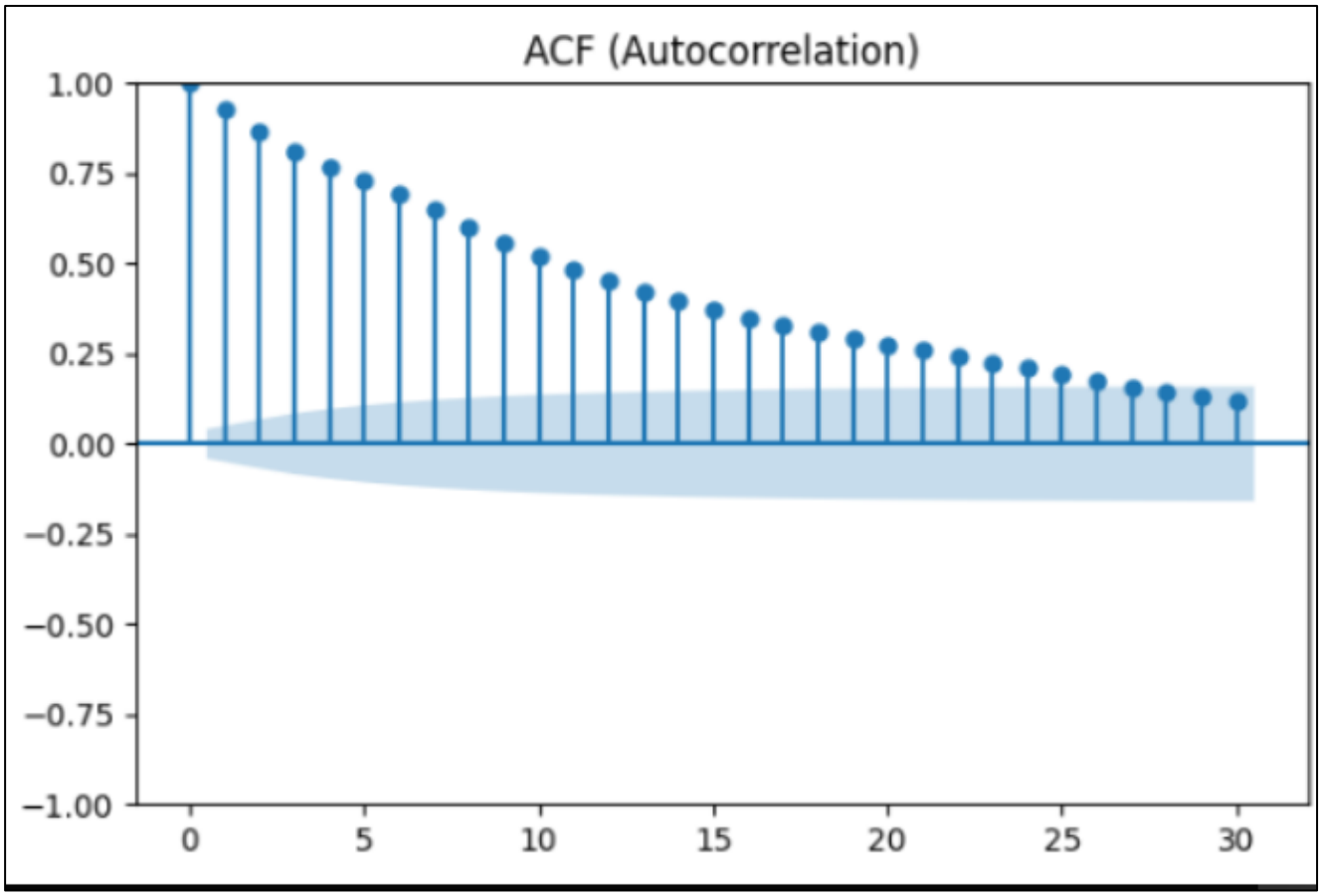

Рисунок 2.21 - Графік автокореляційної функції

Натомість часткова автокореляційна функція (PACF), що зображена на рисунку 2.22, демонструє сильну кореляцію лише на першому лагу, після чого значення різко зменшуються до нуля.

На тепловій карті кореляцій між лагами (рисунок 2.23) спостерігається висока взаємозалежність між найближчими лагами - значення коефіцієнтів перевищують 0.9 для перших трьох лагів і поступово знижуються до близько 0.5 на десятому.

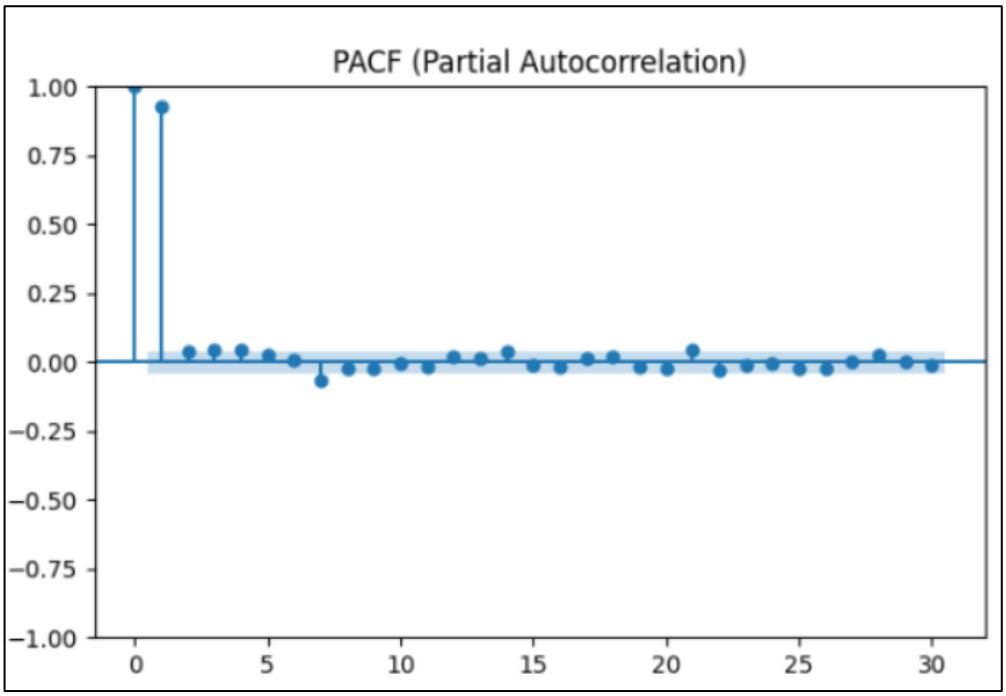

Рисунок 2.22 -Графік часткової автокореляційної функції

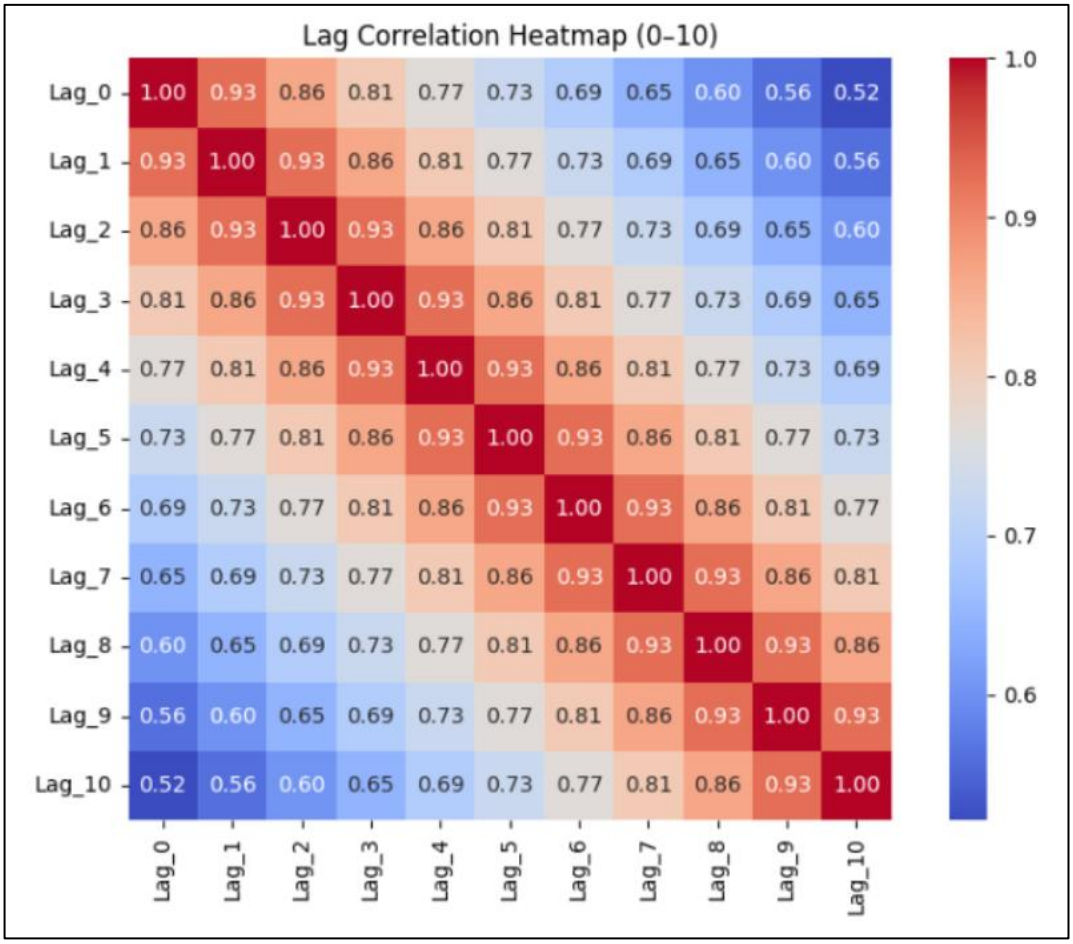

Рисунок 2.23 – Графік теплової карти кореляцій між лагами

Загалом, результати аналізу для станції 1315 свідчать про те, що дані мають стаціонарну природу, містять окремі викиди та виражену автокореляцію. Це робить їх придатними для подальшого моделювання умовної волатильності з метою прогнозування коливань концентрацій PM_1 у повітрі.

Для перевірки висунутих гіпотез було здійснено тестування часового ряду на стаціонарність. Під стаціонарністю розуміють таку властивість ряду, за якої його статистичні характеристики не змінюються з часом [30].

Стаціонарні часові ряди, як правило, дають змогу будувати точніші прогнози, оскільки їхня динаміка є більш передбачуваною. Натомість нестаціонарні ряди можуть містити виражені тренди, сезонні коливання чи інші компоненти, що ускладнюють процес прогнозування.

На рисунку 2.24 зображений фрагмент коду перевірки даних на стаціонарність.

[47]:

```
def check_stationarity(series):
    # Thanks to https://machinelearningmastery.com/time-series-data-stationary-python/

    result = adfuller(series.values)

    print('ADF Statistic: %f' % result[0])
    print('p-value: %f' % result[1])
    print('Critical Values:')
    for key, value in result[4].items():
        print('\t%s: %.3f' % (key, value))

    if (result[1] <= 0.05) & (result[4]['5%'] > result[0]):
        print("\u001b[32mStationary\u001b[0m")
    else:
        print("\x1b[31mNon-stationary\x1b[0m")
```

Рисунок 2.24 – Фрагмент коду для перевірки ряду на стаціонарність

Результати перевірки стаціонарності, що зображені на рисунку 2.25, для станції 1315 свідчать, що часовий ряд концентрацій PM1 є стаціонарним як у вихідному вигляді, так і після диференціювання.

У першому випадку статистика ADF дорівнює –5.268, що значно менше за критичні значення на рівнях 1%, 5% і 10%. При цьому значення p-value дорівнює

0.000006, тобто воно істотно нижче порогового рівня 0.05. Це означає, що нульова гіпотеза про наявність одиничного кореня відхиляється, а ряд можна вважати стаціонарним.

Для перевірки стійкості результату було обчислено першу різницю ряду. Повторний тест дав ADF-статистику –7.863 при p-value = 0.000000. Це підтверджує, що навіть після диференціювання часовий ряд залишається стаціонарним і не має чіткої трендової складової.

Отже, дані станції 1315 не містять систематичних змін середнього чи дисперсії з часом, а їхні статистичні властивості є стабільними.

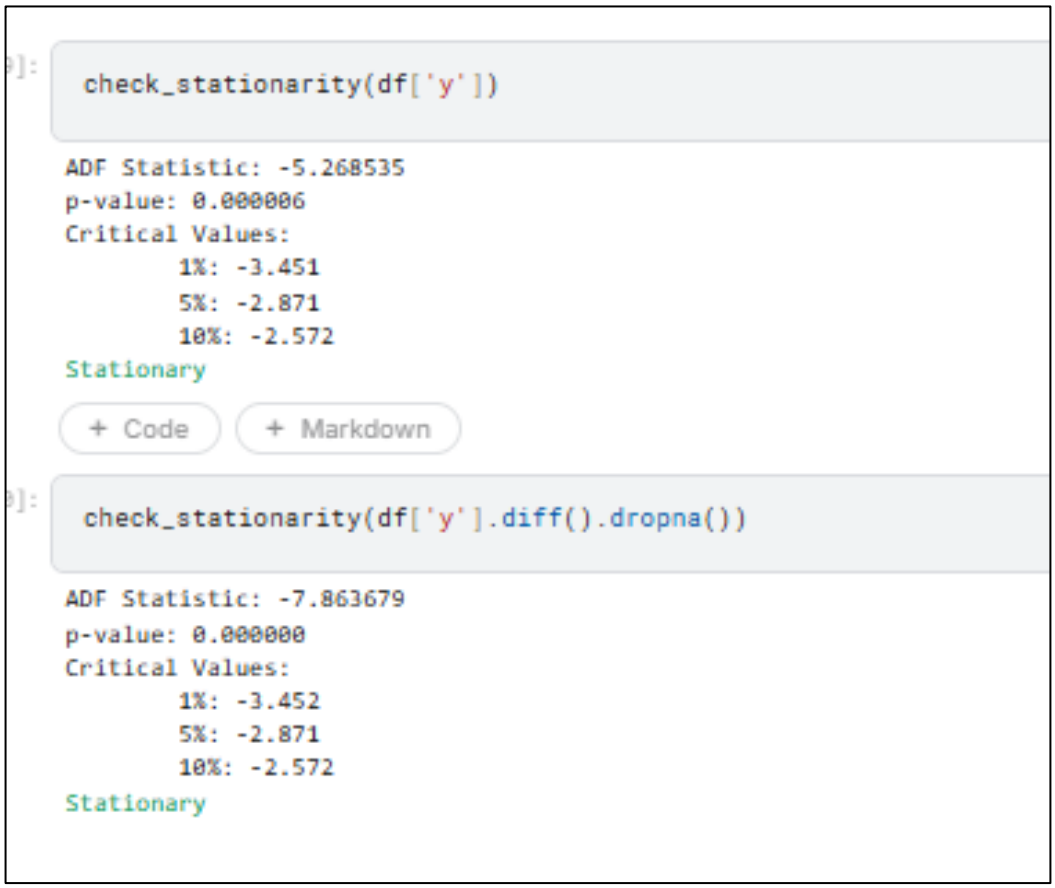

Рисунок 2.25 - результати перевірки часового ряду станції ВНТУ 1315 на стаціонарність.

2.4 Висновки

У цьому розділі проведено підготовку та розвідувальний аналіз даних, отриманих із громадського моніторингу якості повітря EcoCity. Виконано імпорт, очищення, структурування й інтеграцію даних із різних станцій спостереження. Здійснено просторовий аналіз концентрацій PM_1 для виявлення локальних зон підвищеного забруднення та підтверджено наявність пилових епізодів, пов’язаних із перенесенням частинок Сахари.

Проведено розвідувальний аналіз часових рядів, який виявив наявність аномалій.

Досліджено стаціонарність даних і підтверджено, що часові ряди концентрацій PM_1 є придатними для моделювання та прогнозування. Результати показали, що більшість станцій мають подібну динаміку з вираженими піковими значеннями у періоди активного впливу пилу Сахари.

Отримані результати стали основою для побудови моделей прогнозування та розробки інформаційної технології аналізу і прогнозування якості повітря, представлених у наступному розділі.

3. РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ

3.1 Розроблення інформаційної технології

Запропонований алгоритм розробки інформаційної технології прогнозування концентрації сахарського пилу в атмосфері України (рис. 3.1) складається з восьми послідовних етапів:

1. Визначення цільової ознаки, що підлягає прогнозуванню;
2. Попередній аналіз часових рядів та метеорологічних даних з метою виявлення сезонних закономірностей, характерних періодів надходження пилу із Сахари та основних факторів, що впливають на інтенсивність епізодів;
3. Виявлення та обробка аномальних і пропущених значень, усунення викидів, які можуть спотворити навчання моделі;
4. Поділ очищеного набору даних на тренувальну, валідаційну та тестову вибірки з урахуванням часової послідовності (без випадкового перемішування), щоб уникнути витоку інформації з майбутнього в минуле;
5. Налаштування гіперпараметрів та архітектури моделей-кандидатів;
6. Навчання моделей на тренувальній вибірці з подальшою оцінкою їхньої продуктивності на валідаційній вибірці за допомогою метрик, RMSE, MAPE.
7. Виконання фінального прогнозування на тестовій вибірці, яка хронологічно слідує за тренувальними та валідаційними даними, та розрахунок остаточних показників точності.
8. Порівняльний аналіз результатів усіх моделей, вибір оптимальної за критерієм найкращої точності за прийнятного часу обчислень. У разі незадовільних значень метрик здійснюється повернення до етапів 5–6 для додаткового тюнінгу або залучення нових ознак та повторне навчання.

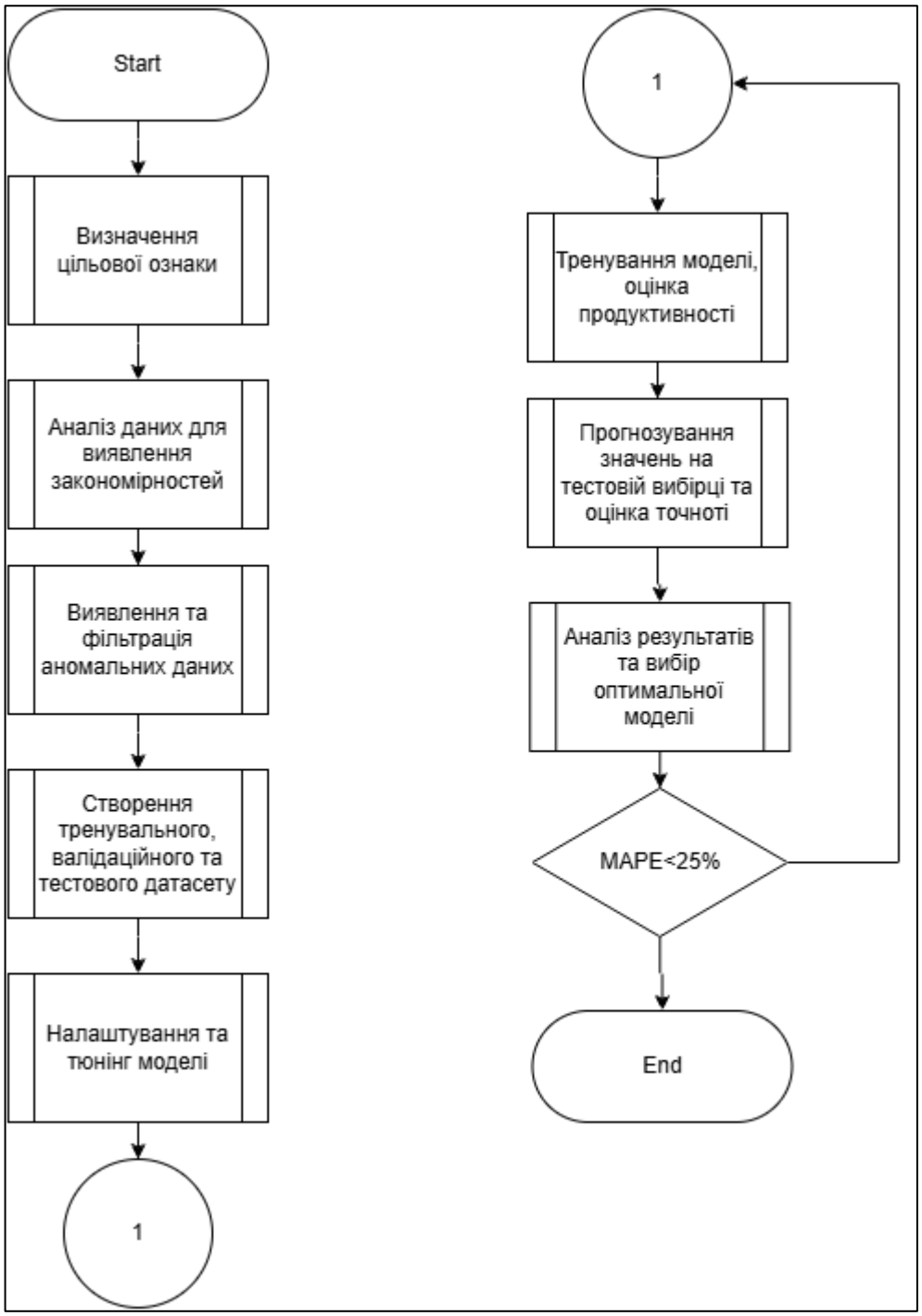

Рисунок 3.1 – Блок-схема алгоритму інформаційної технології аналізу та прогнозування поширення пилу Сахари Україною

3.2 Використання UML-діаграм для опису інтеграції інформаційної технології

Для формалізованого опису структури та поведінки розробленої інформаційної технології аналізу й прогнозування поширення пилу Сахари доцільно застосувати нотацію UML (Unified Modeling Language). UML є стандартизованою мовою візуального моделювання програмних систем, яка забезпечує єдиний набір графічних засобів для опису архітектури, компонентів та їхніх взаємодій на різних рівнях абстракції [29]. Використання UML-діаграм дозволяє узгоджено уявити запропоновану технологію з погляду її інтеграції в існуючі інформаційні системи моніторингу, а також полегшує подальший супровід і розвиток рішення.

На рисунку 3.2 подано UML-діаграму розгортання, яка демонструє можливий варіант інтеграції розробленої інформаційної технології в інформаційну систему моніторингу якості атмосферного повітря. На діаграмі зображено користувача, який через робочу станцію з веб-браузером взаємодіє з веб-сервером, де розгорнуто презентаційний шар (web-інтерфейс) та окремий компонент «Sahara Dust Forecasting Technology». Усередині цього компонента виділено модуль виявлення аномалій та модуль візуалізації, що реалізують відповідні етапи алгоритму, описаного у розділі 3.1. Веб-сервер пов’язаний із сервером баз даних, у якому розміщено сховище «Air quality DB», а також із зовнішнім сервером моніторингу EcoCity API, через який здійснюється отримання даних громадського моніторингу. Окремим елементом показано файл журналу («log file»), до якого записуються службові повідомлення про роботу технології. Така діаграма дозволяє наочно відобразити, в якому апаратно-програмному середовищі може бути розгорнута запропонована технологія, та які зовнішні сервіси вона використовує.

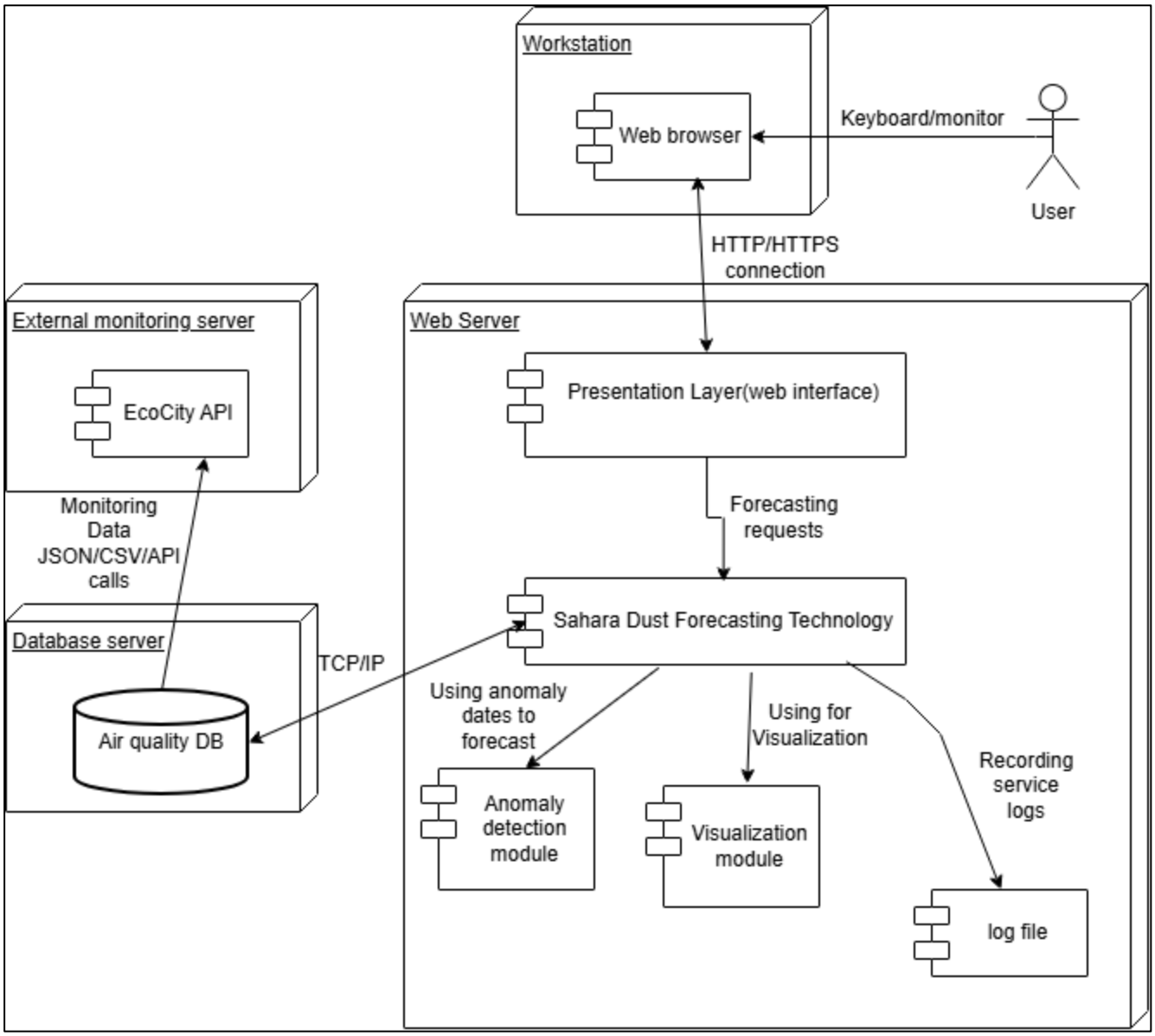

Рисунок 3.2 - UML-діаграма розгортання

Для повнішого розуміння функціональних можливостей системи та ролі користувача доцільно доповнити опис ще двома UML-діаграмами: діаграмою варіантів використання (use case diagram) та діаграмою послідовності (sequence diagram) [29]. Діаграма варіантів використання узагальнює, які задачі може виконувати користувач за допомогою інформаційної системи, а діаграма послідовності деталізує порядок обміну повідомленнями між основними компонентами під час реалізації одного з ключових сценаріїв роботи. У поєднанні з діаграмою розгортання це забезпечує комплексне подання як структурних, так і поведінкових аспектів системи.

На рисунку 3.3 доцільно подати діаграму варіантів використання для розробленої інформаційної технології аналізу та прогнозування поширення пилу Сахари в складі системи.

Діаграма відображає основні сценарії взаємодії користувачів із розробленою інформаційною технологією та її зв’язок із зовнішнім сервісом моніторингу EcoCity API. Система подана у вигляді межі, всередині якої розміщено варіанти використання, а зовні - актори, що ініціюють відповідні сценарії.

Основним актором є User, який представляє аналітика або дослідника, що безпосередньо працює з даними моніторингу. Він може виконувати такі варіанти використання:

1. Import Data - імпорт даних моніторингу в систему. Для цього система взаємодіє з зовнішнім сервісом EcoCity API, який надає дані для подальшого аналізу;
2. EDA - розвідувальний аналіз даних, під час якого користувач отримує початкові узагальнення та виявляє типові закономірності у часових та просторових розподілах;
3. Anomaly detection - виявлення аномальних епізодів (піків забруднення), що розширює базові можливості системи з аналізу даних;
4. Forecasting for selected station - побудова прогнозу для обраної станції моніторингу з урахуванням попередньо виявлених аномалій;
5. View graphical results - перегляд графічних результатів аналізу та прогнозування.

У діаграмі використано відношення «include» та «extend», що відображають залежності між окремими варіантами використання. Варіант Forecasting for selected station включає сценарій Setting parameters, оскільки перед запуском прогнозування адміністратор або досвідчений користувач повинен налаштувати параметри моделі (горизонт прогнозу, тип моделі, порогові значення тощо). Водночас варіанти EDA, Anomaly detection та Forecasting for selected station розширюють варіант View graphical results, оскільки після їх виконання користувач, за потреби, може перейти до детального перегляду побудованих графіків, карт та інших візуалізацій. Таким

чином, перегляд результатів не є обов’язковою частиною кожного сценарію, але логічно продовжує їх.

Другим актором є Admin, який відповідає за конфігурацію системи та має доступ до варіанта використання Setting parameters. Адміністратор може змінювати параметри алгоритмів, режими роботи модулів та інші налаштування, від яких залежить якість аналізу й прогнозування.

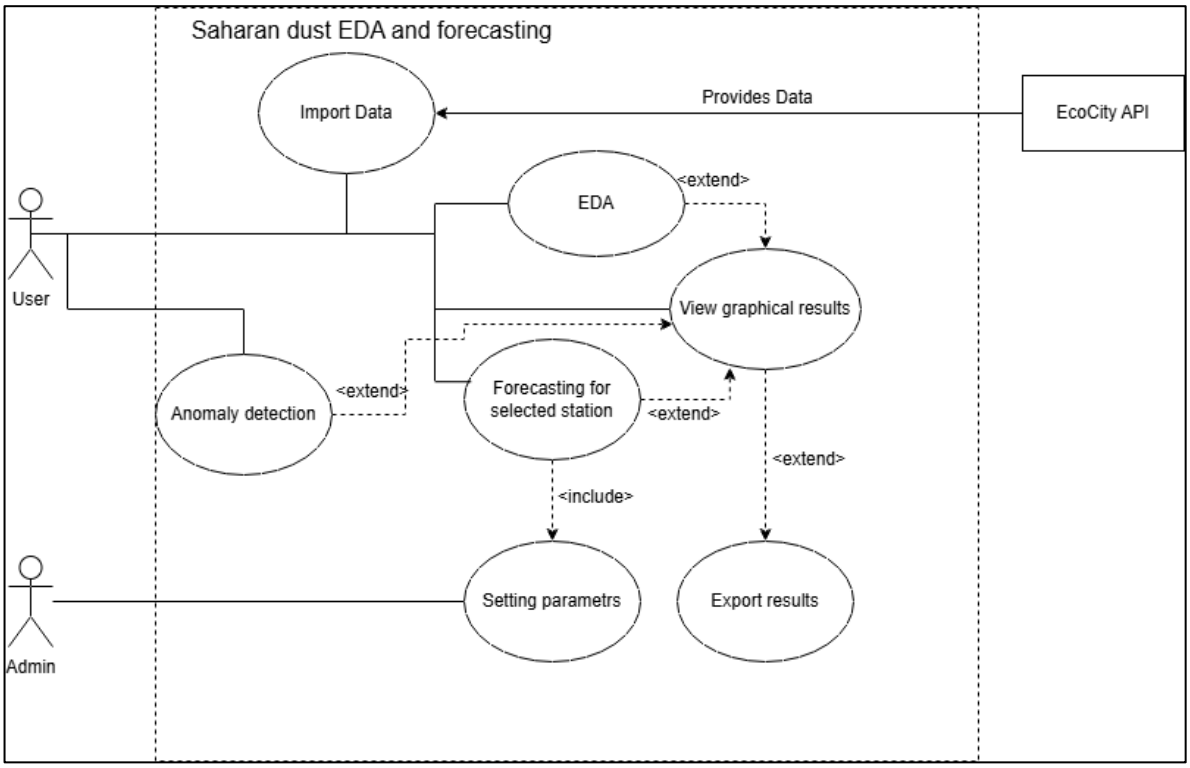

Рисунок 3.3 - Use-case діаграма

На рисунку 3.4 наведено UML-діаграму послідовності, яка деталізує сценарій «Побудова прогнозу для обраної станції»

Уздовж горизонтальної осі розташовано учасників взаємодії: актор User, клієнтський Web Browser, Presentation layer, ядро системи Saharan dust forecasting technology, внутрішні модулі Anomaly detection module та Visualisation module, а також сховище Air quality DB. Вертикальні «життєві лінії» відображають час існування об’єктів, а стрілки між ними - виклики операцій і повернення результатів.

Послідовність взаємодій розпочинається з того, що користувач формує запит на прогнозування (крок 1), який через веб-браузер надсилається на веб-сервер у

вигляді HTTPS-запиту (крок 2). Презентаційний шар викликає операцію Forecast() ядра технології (крок 3). Далі система звертається до бази даних Air quality DB для отримання історичних вимірювань (кроки 4–5). Після цього Anomaly detection module виконує виявлення аномальних епізодів у часовому ряді (крок 6) і повертає їх до ядра технології (крок 7). Отримані дані передаються до Visualisation module для побудови початкових графіків та аналітичних візуалізацій (кроки 8–9), після чого запускається власне процедура прогнозування (крок 10) та візуалізація результатів прогнозу (крок 11). Зведені прогнозні дані повертаються до презентаційного шару (крок 12), який формує HTML-відповідь (крок 13); у підсумку користувач отримує на екрані інтерактивне відображення прогнозу (крок 14).

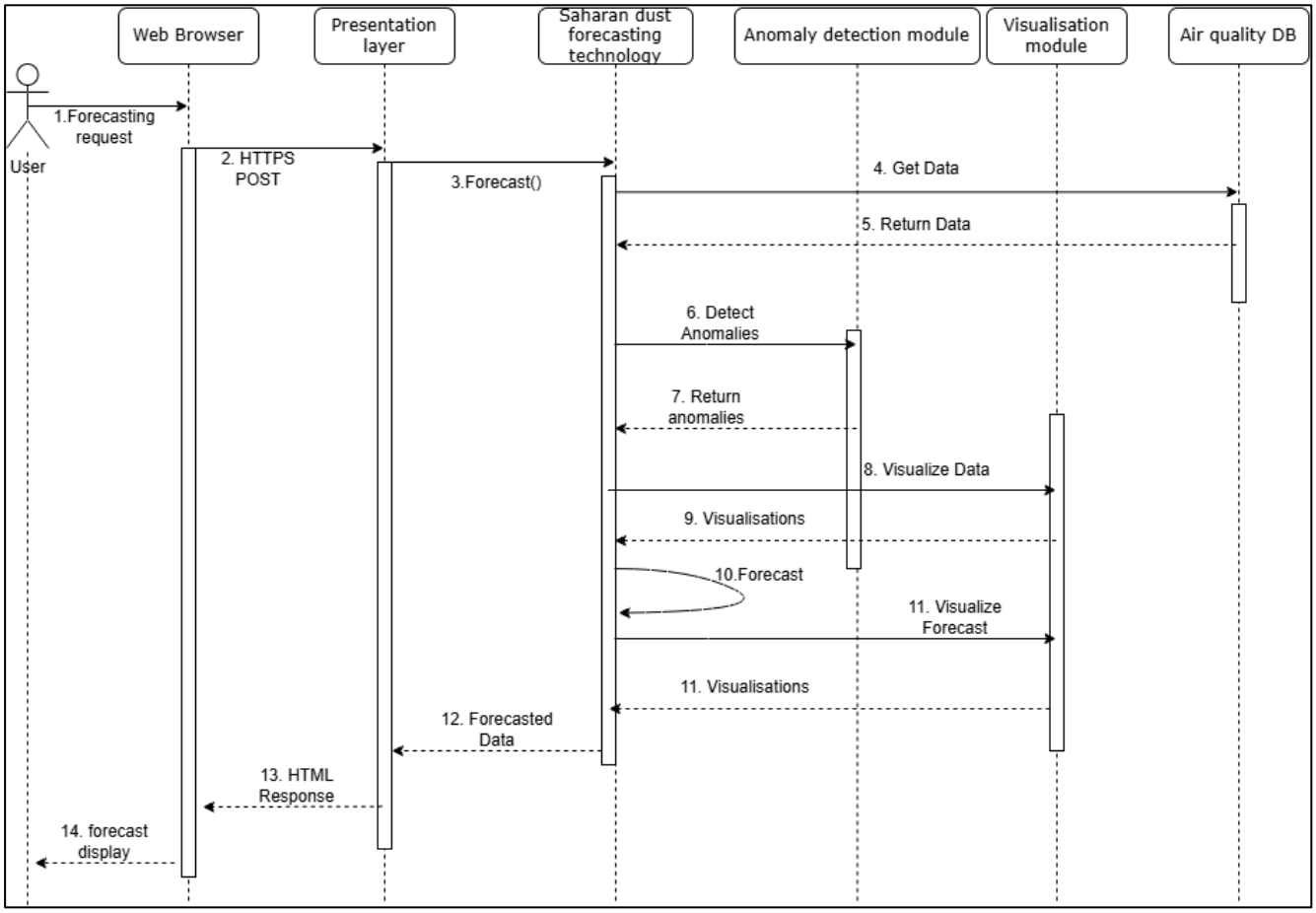

Рисунок 3.4 – Діаграма послідовності

Таким чином, включення до роботи комплексу UML-діаграм - діаграми розгортання (рисунок 3.2), діаграми варіантів використання (рисунок 3.3) та діаграми послідовності (рисунок 3.4) - забезпечує формалізоване подання інтеграції, функцій та динаміки роботи запропонованої інформаційної технології.

3.3 Прогнозування поширення пилу Сахари

Виконаємо прогнозування по даним станції ВНТУ 1315 за допомогою моделей описаних у першому розділі, а саме ARIMA та Prophet.

Для прогнозування часових рядів використаємо моделі ARIMA та Prophet.

Прогнозування здійснюємо на горизонт 24 години вперед.

Задаємо початкову конфігурацію, назва набору даних “PM1”, цільову зміну y, та період для прогнозування (фрагмент коду рис 3.5).

```
# Set main parameters
series_name = 'PM1'
target = 'y'
forecasting_points = 8 # 3H resampling means 1 day = 24/3 = 8 points, daily resampling - 1 = 1 day
```

Рисунок 3.5 – Початкові налаштування

Оскільки вимірювання якості повітря виконуються з інтервалом у 20 хвилин, часові ряди зазнають значного впливу швидкозмінних факторів: добових коливань температури, вологості, швидкості та напрямку вітру, інтенсивності дорожнього руху, роботи промислових об’єктів, а також епізодичних викидів. Усе це призводить до високої зашумленості даних, що суттєво ускладнює роботу класичних моделей прогнозування, зокрема Prophet та ARIMA.

Під зашумленістю даних розуміють наявність випадкових відхилень, викидів та сторонніх коливань, які не відображають реальних закономірностей

досліджуваного процесу, а є результатом вимірювальних похибок, короткочасних локальних впливів або неврахованих зовнішніх факторів [29]. Такий шум суттєво погіршує здатність лінійних і сезонних моделей коректно ідентифікувати тренди та сезонні складові, що проявляється в:

– зростанні помилки прогнозування (завищені значення RMSE та MAPE);

– нестабільності оцінок параметрів моделей;

– появі хибних або надмірно складних сезонних патернів, які насправді є артефактами шуму.

Для того, щоб позбутись зашумленості прийнято рішення агрегувати дані з кроком в три години по максимальному значенні, адже саме за такої конфігурації ми не втрачаємо різкі сплески забруднення чи піків Сахари. На рисунку 3.6 зображено тимчасовий ряд максимальних 3-годинних концентрацій PM1 після попередньої обробки даних. Фонові значення концентрації коливаються переважно в межах 10–20 мкг/м^3. Водночас чітко простежуються періодичні різкі підвищення до 25–35 мкг/м^3 і вище, які відповідають епізодам перенесення сахарського пилу навесні 2024 року. Кожен такий епізод триває 1-4 доби, причому максимальна інтенсивність спостерігається протягом першої доби. Саме ці короткочасні, але значні сплески є основною причиною перевищень нормативів якості повітря та становлять найбільшу небезпеку для здоров’я населення. Тому подальше моделювання та прогнозування виконується саме на цьому часовому ряді з агрегацією за максимальним значенням у 3-годинних інтервалах.

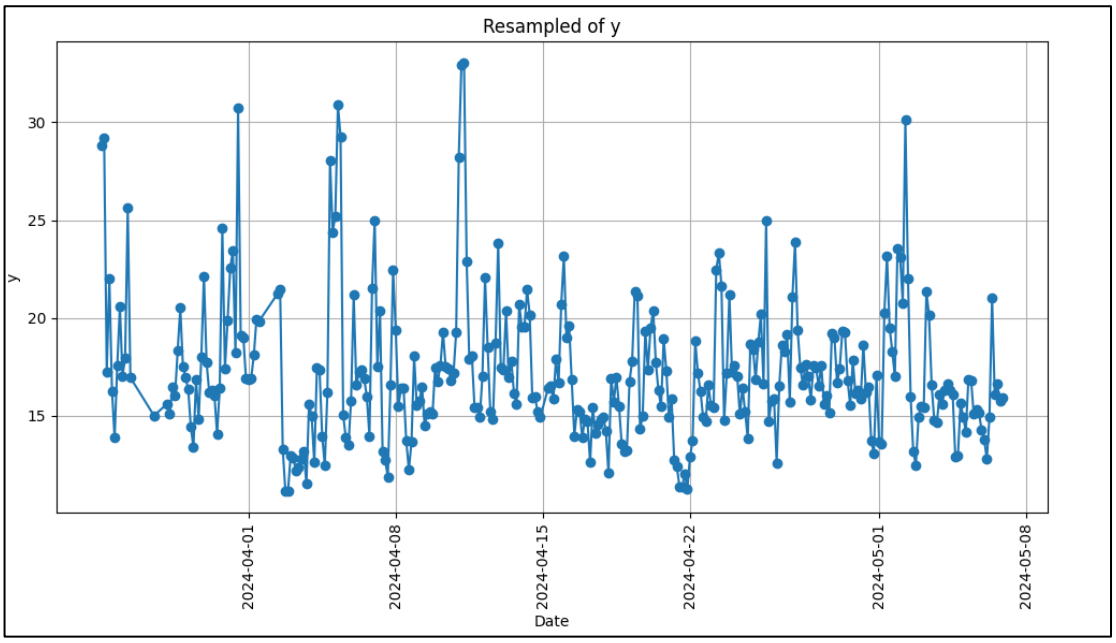

Рисунок 3.6 – Динаміка максимальних 3-годинних концентрацій PM1 на станції ВНТУ-1315

Виявлення аномалій є важливим кроком у підготовці даних для прогнозування часових рядів. У даній роботі використано бібліотеку Prophet для прогнозування, а також методи для виявлення аномальних значень у даних. Ці аномалії можуть суттєво впливати на якість прогнозу, тому їх ідентифікація та обробка є критично важливими.

На рисунку 3.7 наведено результат пошуку моментів часу, коли максимальна 3-годинна концентрація PM1 перевищувала поріг 29 мкг/м^3. Цей поріг визначено під час розвідувального аналізу як значення, яке вище типового фонового рівня [31]. Усього виявлено 7 таких точок, які в подальшому використовуються як кандидати на дати початку епізодів перенесення сахарського пилу.

```
if is_anomalies:
    anomalous_by_level = df[df['y'] > 29]['ds'].copy()

    print(f"   Found {len(anomalous_by_level)} timestamps with PM1 > 29 µg/m³")
    display(anomalous_by_level.dt.strftime('%Y-%m-%d %H:%M').unique())
```

```
   Found 7 timestamps with PM1 > 29 µg/m³
array(['2024-03-25 03:00', '2024-03-31 12:00', '2024-04-05 06:00',
       '2024-04-05 09:00', '2024-04-11 03:00', '2024-04-11 06:00',
       '2024-05-02 06:00'], dtype=object)
```

Рисунок 3.7 – Виявлення аномальних дат за абсолютним порогом

На рисунку 3.8 зображено першу різницю ряду (y(t) – y(t–1)). Такий графік дозволяє візуально виявити моменти різкого зростання концентрації, що характерно саме для початку сахарських інвазій, коли за кілька годин концентрація може зрости на 10–20 мкг/м^3 і більше.

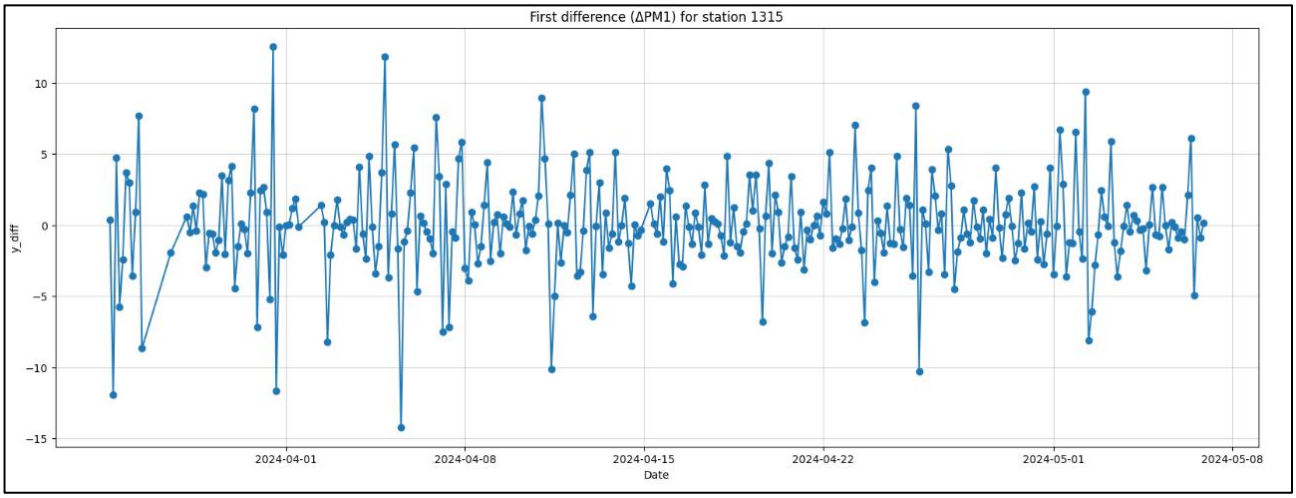

Рисунок 3.8 – Графік першої різниці часового ряду ΔPM_1

На рисунку 3.9 червоними крапками позначено моменти часу, коли абсолютна зміна концентрації за один 3-годинний крок перевищила 7 мкг/м^3. Усього виявлено 20 таких стрибків. Цей метод є більш чутливим до початку епізоду, ніж простий поріг по рівню.

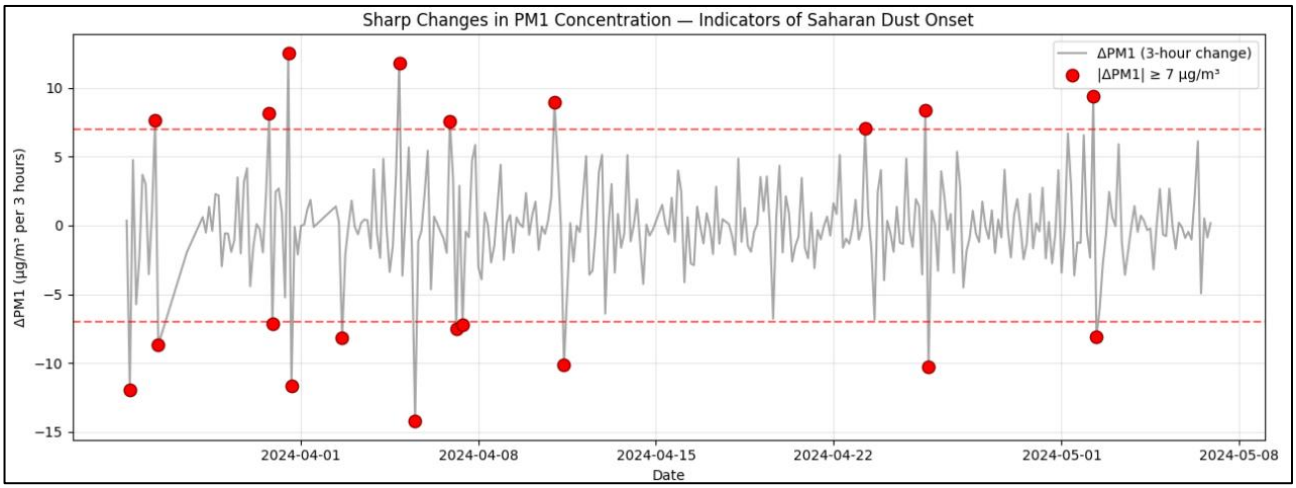

Рисунок 3.9 – Виявлення різких стрибків концентрації $|\Delta PM_1| \geq 7$ мкг/м^3

На рисунку 3.10 продемонстровано процес злиття двох списків кандидатів (за рівнем і за стрибком) з подальшим видаленням сусідніх дат. Оскільки один епізод сахарського пилу триває 1–4 доби, залишаємо лише першу дату кожного кластера, якщо між сусідніми подіями минає менше 4 діб. У результаті отримано 6 чистих епізодів.

```
# Combine both methods, remove duplicates, and ensure proper datetime type
all_candidates = pd.concat([anomalous_by_level, anomalous_by_jump]).drop_duplicates()
candidates = pd.DatetimeIndex(all_candidates).sort_values()

# Filter: keep only the first date of each cluster (Saharan events last 2–4 days)
sahara_start_dates = [candidates[0]]  # first event is always kept
min_gap_between_episodes = pd.Timedelta(days=4)

for current_date in candidates[1:]:
    if current_date - sahara_start_dates[-1] >= min_gap_between_episodes:
        sahara_start_dates.append(current_date)
    # else: skip — belongs to the same ongoing episode

sahara_start_dates = pd.DatetimeIndex(sahara_start_dates)

print(f"   Final number of clean Saharan dust episodes: {len(sahara_start_dates)}")
print("   Episode start dates:")
for d in sahara_start_dates.strftime('    • %Y-%m-%d'):
    print(d)
```

```
Final number of clean Saharan dust episodes: 6
Episode start dates:
• 2024-03-25
• 2024-03-30
• 2024-04-04
• 2024-04-11
• 2024-04-23
• 2024-05-02
```

Рисунок 3.10 – Об’єднання та очищення кандидатів від сусідніх дат

На рисунку 3.11 наведено остаточний датафрейм holidays_df, що містить лише 6 дат початку аномальних епізодів. Для кожної дати встановлено upper_window = 3 (ефект поширюється на наступні 3 дні) та prior_scale = 80. Така структура дозволяє моделі Prophet адекватно враховувати вплив рідкісних, але надзвичайно сильних зовнішніх подій без дублювання та гіперболізації їхньої амплітуди.

	holiday	ds	lower_window	upper_window	prior_scale
0	sahara_event	2024-03-25 03:00:00	0	3	80
1	sahara_event	2024-03-30 18:00:00	0	3	80
2	sahara_event	2024-04-04 21:00:00	0	3	80
3	sahara_event	2024-04-11 00:00:00	0	3	80
4	sahara_event	2024-04-23 06:00:00	0	3	80
5	sahara_event	2024-05-02 06:00:00	0	3	80

Рисунок 3.11 – Сформований датафрейм аномальних дат для моделі Prophet

Після попередньої обробки та виявлення аномалій увесь датасет поділено на три частини: тренувальний набір містить 324 спостереження, валідаційний - 8 рядків, тестовий - також 8 рядків. Такий розподіл дозволяє провести надійне налаштування гіперпараметрів моделі та об’єктивно оцінити її прогнозну здатність на незалежних даних (рис. 3.12).

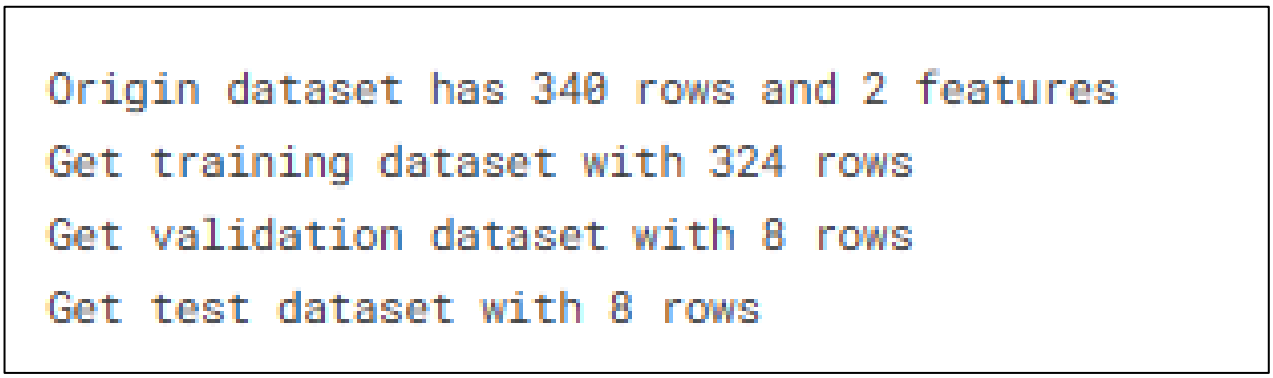

Рисунок 3.12 – Розподіл даних на тренувальний, валідаційний та тестовий набори

На рисунку 3.13 наведено конфігурацію моделі Prophet. Увімкнено щоденну та щотижневу сезонність, використано мультиплікативний режим сезонності. Параметр changepoint_prior_scale знижено до 0.05, що забезпечує стабільний і осмислений тренд без штучних зламів. Підключено датафрейм з 6 очищеними епізодами перенесення пилу Сахари (holidays_prior_scale=80), а також додано кастомну субдобову сезонність.

```
def prophet_modeling(result,
                     series_name,
                     train,
                     test,
                     holidays_df,
                     period_hours,
                     fourier_order_seasonality,
                     forecasting_period,
                     name_model,
                     type_data):
        model = Prophet(daily_seasonality=True,
                     weekly_seasonality=True,
                     yearly_seasonality=False,
                     changepoint_range=0.95,
                     changepoint_prior_scale=0.05,
                     seasonality_prior_scale=10,
                     holidays_prior_scale=80,
                     holidays=holidays_df,
                     seasonality_mode = 'multiplicative')
```

Рисунок 3.13 – Основні параметри моделі Prophet

На рисунку 3.14 зображено процес сітчастого пошуку оптимальних параметрів кастомної сезонності. Перевірялися періоди 6, 8, 12 та 18 годин (0.25–0.75 доби) та порядки ряду Фур'є 7, 12 та 21. Кожна комбінація навчалася на тренувальному наборі та оцінювалася на валідаційному.

```
%%time
# Models tuning
if is_Prophet:
    for period_hours in [6,8, 12, 18]:
        for fourier_order_seasonality in [7, 12, 21]:
            result, _ = prophet_modeling(result,
                series_name,
                train_ts,
                valid_ts,
                holidays_df,
                period_hours,
                fourier_order_seasonality,
                forecasting_points,
                f'{period_hours}_period_{fourier_order_seasonality}_order',
                'valid')
```

Рисунок 3.14 – Автоматичний перебір гіперпараметрів кастомної сезонності

На рисунку 3.15 зображений прогноз якості повітря та поширення пилу Сахари на тренувальних даних за допомогою Prophet. Модель успішно відтворює

всі основні піки, пов’язані з перенесенням сахарського пилу, та демонструє високу точність на валідаційному наборі.

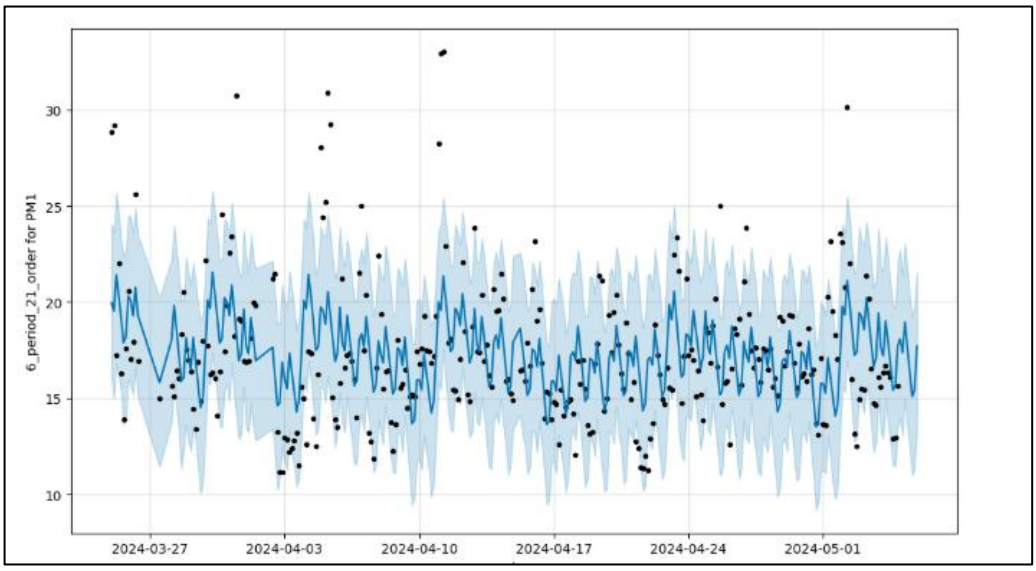

Рисунок 3.15 - Результат прогнозування якості повітря та поширення пилу Сахари за найкращою моделлю за валідаційними даними на основі Prophet

Prophet дозволяє візуалізувати окремі компоненти: лінію тренду, свята, у нашому випадку це аномальні дати та сезонність, це представлено на рисунку 3.16. Тренд - плавне зниження фонового рівня протягом весни 2024 року. Ефект аномальних дат (holidays) - чітко виражені піки під час епізодів перенесення сахарського пилу. Шотижнева сезонність - стабільні коливання залежно від дня тижня.

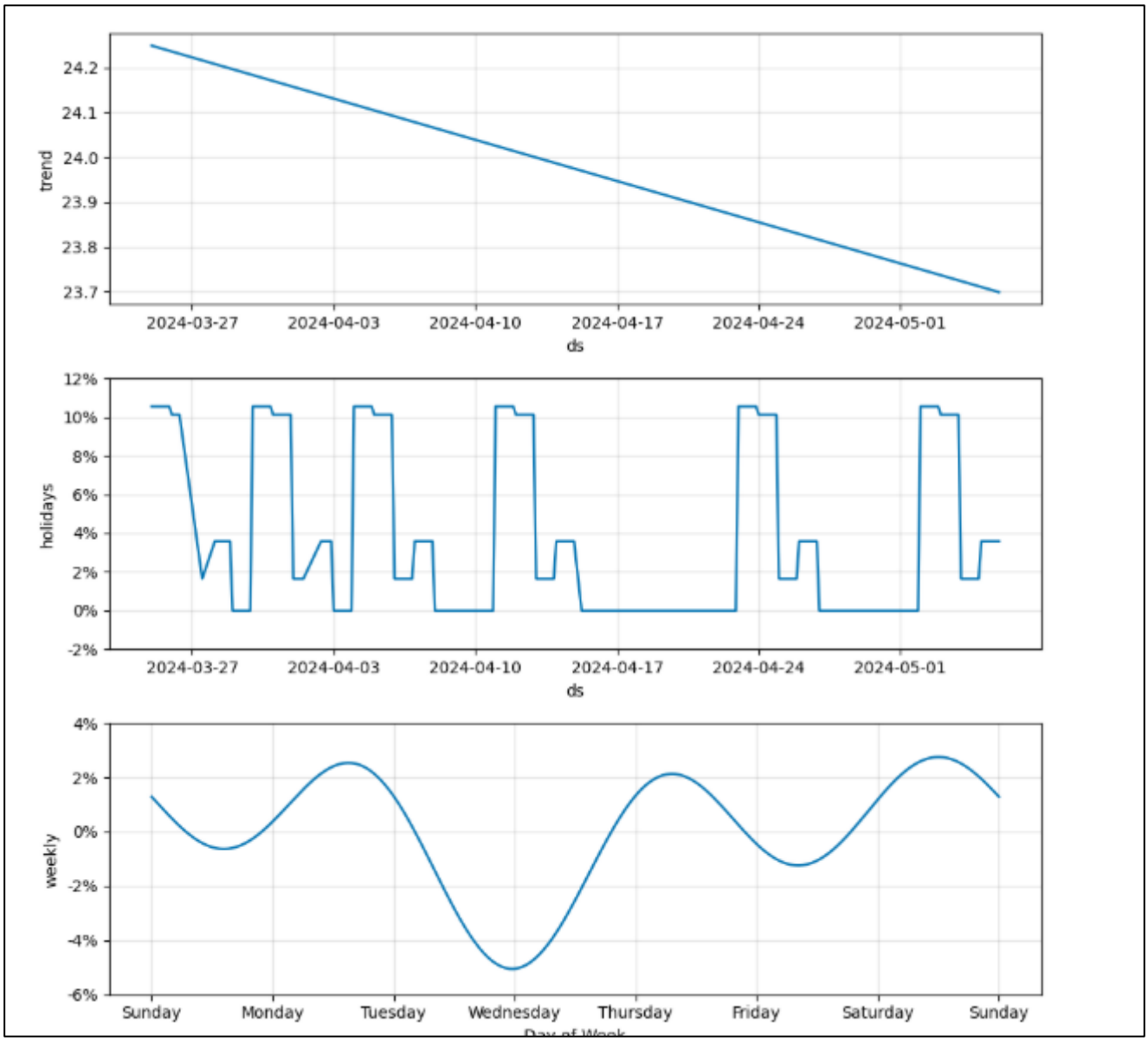

Рисунок 3.16 – Візуалізація графіків компонентів прогнозування за допомогою Prophet

Для прогнозування за допомого моделі ARIMA використаємо автоматизацію процесу побудови моделі ARIMA, а саме автоматичний перебір варіантів параметрів моделі (рис 3.17, 3.18).

```
Performing stepwise search to minimize aic
 ARIMA(0,0,0)(0,0,0)[56] intercept   : AIC=1768.308, Time=0.02 sec
 ARIMA(1,0,0)(1,0,0)[56] intercept   : AIC=1642.796, Time=10.26 sec
 ARIMA(0,0,1)(0,0,1)[56] intercept   : AIC=1661.472, Time=3.45 sec
 ARIMA(0,0,0)(0,0,0)[56]             : AIC=1768.308, Time=0.04 sec
 ARIMA(1,0,0)(0,0,0)[56] intercept   : AIC=1641.189, Time=0.13 sec
 ARIMA(1,0,0)(0,0,1)[56] intercept   : AIC=1642.825, Time=4.55 sec
 ARIMA(1,0,0)(1,0,1)[56] intercept   : AIC=1644.600, Time=28.61 sec
 ARIMA(2,0,0)(0,0,0)[56] intercept   : AIC=1642.491, Time=0.21 sec
 ARIMA(1,0,1)(0,0,0)[56] intercept   : AIC=1642.338, Time=0.21 sec
 ARIMA(0,0,1)(0,0,0)[56] intercept   : AIC=1659.665, Time=0.06 sec
 ARIMA(2,0,1)(0,0,0)[56] intercept   : AIC=1642.212, Time=0.30 sec
 ARIMA(1,0,0)(0,0,0)[56]             : AIC=1641.189, Time=0.11 sec

Best model:  ARIMA(1,0,0)(0,0,0)[56]
Total fit time: 47.979 seconds
                               SARIMAX Results
==========================================================================
Dep. Variable:                      y   No. Observations:                  324
Model:               SARIMAX(1, 0, 0)   Log Likelihood                -817.594
Date:                Tue, 02 Dec 2025   AIC                           1641.189
Time:                        14:15:07   BIC                           1652.531
Sample:                             0   HQIC                          1645.716
                                  - 324
Covariance Type:                  opg
==========================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
--------------------------------------------------------------------------
intercept      7.2360      0.722     10.026      0.000       5.822       8.651
ar.L1          0.5814      0.034     16.919      0.000       0.514       0.649
sigma2         9.0950      0.573     15.881      0.000       7.973      10.217
==========================================================================
Ljung-Box (L1) (Q):                   0.06   Jarque-Bera (JB):               141.71
Prob(Q):                              0.80   Prob(JB):                         0.00
Heteroskedasticity (H):               0.50   Skew:                             1.06
Prob(H) (two-sided):                  0.00   Kurtosis:                         5.45
==========================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
CPU times: user 1min 45s, sys: 1min 20s, total: 3min 6s
Wall time: 48 s
```

Рисунок 3.17 – Автоматичний підбір найкращих параметрів та результат роботи найкращої моделі

На рисунку показано процес автоматичного перебору параметрів моделі ARIMA за допомогою бібліотеки pmdarima. Найкращою за критерієм AIC виявилася модель SARIMAX(1,0,0)×(1,0,0,56) з сезонністю 56 кроків (7 діб). Час підбору склав 47,97 секунди. Отримані коефіцієнти статистично значущі, однак подальша діагностика залишків показала суттєві проблеми.

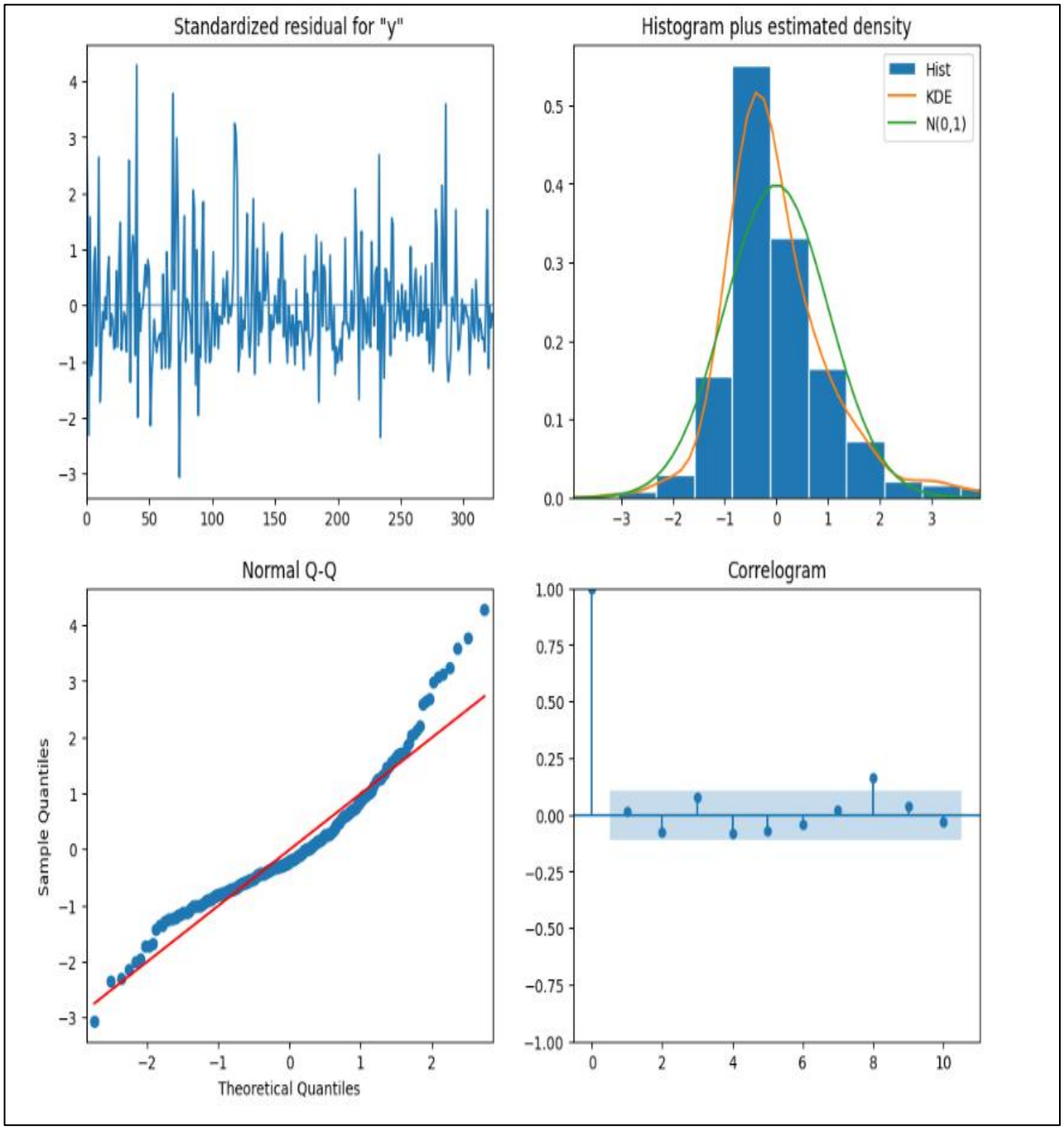

Рисунок 3.18 – Результат діагностики оптимальної моделі ARIMA

На рисунку 3.18 подано стандартний набір діагностичних графіків, що дозволяють оцінити якість апроксимації даних побудованою ARIMA-моделлю та перевірити виконання базових припущень авторегресійного підходу.

Перший графік (Standardized residual for “y”) відображає стандартизовані залишки моделі у часовій послідовності. У добре налаштованої ARIMA-моделі залишки мають поводитися як «білий шум»: не демонструвати тренду,

автокореляції чи структурних патернів. На графіку залишки розподілені нерівномірно, місцями утворюють групи підвищеної або зниженої дисперсії, що свідчить про нерівномірність мінливості та порушення припущення про сталу дисперсію.

Другий графік (Histogram plus estimated density) порівнює емпіричний розподіл залишків із ядровою оцінкою щільності та теоретичним нормальним розподілом. Видно, що форма гістограми не відповідає нормально розподіленим залишкам: розподіл сплощений, має виражені «хвости», а також асиметрію. Це свідчить про те, що модель не змогла повністю врахувати структуру варіацій у даних.

На графіку Normal Q–Q точки систематично відхиляються від теоретичної прямої, особливо в областях великих і малих значень. Це додатково підтверджує невідповідність залишків нормальному розподілу та вказує, що модель неадекватно опрацьовує аномальні або різко змінні значення ряду.

Останній графік (Correlogram) демонструє автокореляції залишків. Для коректної ARIMA-моделі всі значення повинні лежати в межах довірчого інтервалу, що свідчило б про відсутність автокореляції. У даному випадку кілька лагів перевищують допустимі межі, що означає наявність залишкової структури, яку модель не врахувала. Це вказує на недооцінку сезонності або інших системних залежностей у даних.

Узагальнено, представлена діагностика демонструє, що обрана ARIMA-модель не забезпечує бажаної якості прогнозування для цього часового ряду. Порушення нормальності, автокореляція залишків та неоднорідна дисперсія свідчать про недостатню здатність моделі адекватно описати динаміку PM1.

На рисунку 3.19 зображена таблиця результатів оцінки різних моделей прогнозування для валідаційних даних.

Таблиця з рисунку 3.19 містить наступні стовпці:

- name_model: Назва моделі та її параметрів;
- type_data: Тип даних (в даному випадку всі дані є валідаційними, тобто valid);

– rmse: Корінь середньоквадратичної помилки (Root Mean Squared Error), що вимірює середню відстань між передбаченими та фактичними значеннями;

– mape: Середня абсолютна відносна помилка (Mean Absolute Percentage Error), що вимірює середню відносну помилку в відсотках [30].

	name_model	type_data	rmse	mape	params
2	Prophet_6_period_21_order	valid	2.495077	9.179462	[6, 21]
6	Prophet_12_period_7_order	valid	2.503882	9.253697	[12, 7]
1	Prophet_6_period_12_order	valid	2.513019	9.275843	[6, 12]
4	Prophet_8_period_12_order	valid	2.513864	9.283632	[8, 12]
5	Prophet_8_period_21_order	valid	2.514524	9.281817	[8, 21]
0	Prophet_6_period_7_order	valid	2.520634	9.321122	[6, 7]
3	Prophet_8_period_7_order	valid	2.523805	9.297957	[8, 7]
10	Prophet_18_period_12_order	valid	2.529966	9.335925	[18, 12]
8	Prophet_12_period_21_order	valid	2.537303	9.338823	[12, 21]
7	Prophet_12_period_12_order	valid	2.543201	9.342272	[12, 12]
11	Prophet_18_period_21_order	valid	2.570037	9.583516	[18, 21]
9	Prophet_18_period_7_order	valid	2.575777	9.590737	[18, 7]
12	ARIMA_auto	valid	2.632098	11.360327	[1, 0, 0]

Number of models built - 13

Рисунок 3.19 – Таблиця результатів оцінки різних моделей прогнозування

Перевірено різні конфігурації періодів сезонності та порядків перетворення Фур'є.

Найкращий результат показала модель Prophet з кастомною сезонністю 6 годин та порядком Фур’є 21 (MAPE = 9,17 %, RMSE = 2,495).

Додаткове дослідження прогнозу SARIMAX на тестовому наборі показало, що модель фактично генерує майже горизонтальну пряму - константу, близьку до середнього значення ряду. Це свідчить про повну нездатність лінійної моделі SARIMAX враховувати різкі епізодичні сплески, зумовлені перенесенням сахарського пилу.

Для прогнозування даних інших станцій моделі autoARIMA не буде враховуватись.

На рисунку 3.20 зеленими крапками позначено фактичні значення тестового набору, синьою лінією - прогноз моделі Prophet_6_period_21_order. Модель точно відтворює загальну динаміку та більшість піків, включаючи зростання концентрації PM1 наприкінці тестового періоду. Отримане значення MAPE на тестовому наборі становить 7.38 %, що є гарним результатом для 24-годинного прогнозування концентрації часток у реальних умовах.

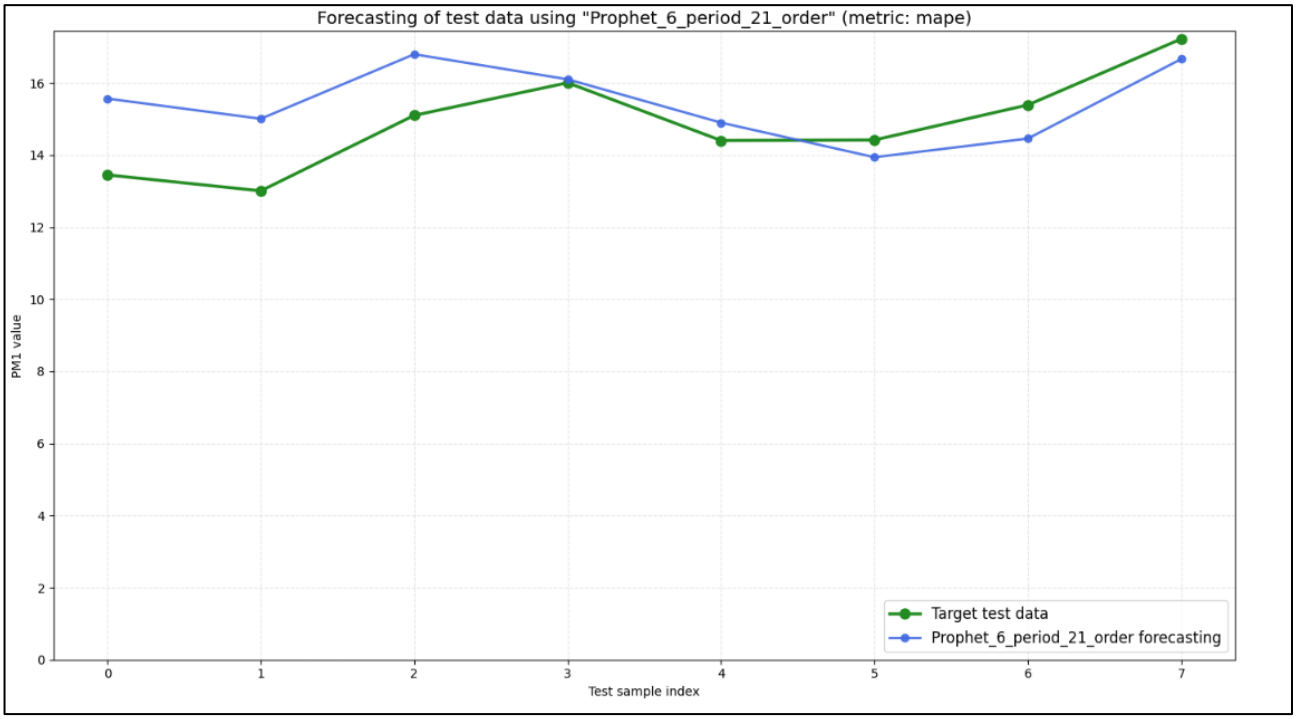

Рисунок 3.20 - Прогноз на тестовому наборі для станції ВНТУ

На рисунках 3.21 та 3.22 наведено результати прогнозування концентрації PM1 на тестовому наборі для двох додаткових моніторингових станцій мережі моніторингу якості повітря - станцій із кодами 1612 та 1872. Для станції 1612

найкращий результат показала конфігурація з періодом кастомної сезонності 8 годин та високим порядком Фур'є 21 (MAPE = 18.40 %, RMSE = 1.243). Модель непогано відтворює основний пік наприкінці тестового періоду, хоча й дещо раніше розпочинає зростання та спад прогнозу. Для станції 1872 найкращою виявилася модель із періодом 6 годин, але значно нижчим порядком Фур'є - лише 7 (MAPE = 25%, RMSE = 1.228). На графіку видно, що прогноз має виражену трикутну форму з різким підйомом та спадом.

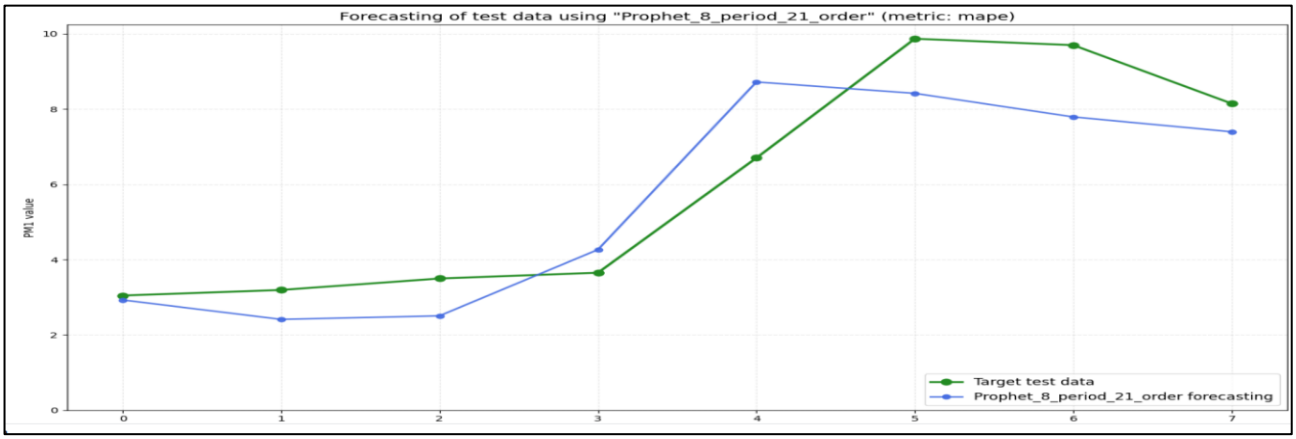

Рисунок 3.21 – Прогноз на тестовому наборі для станції 1612 (найкраща модель: Prophet_8_period_21_order)

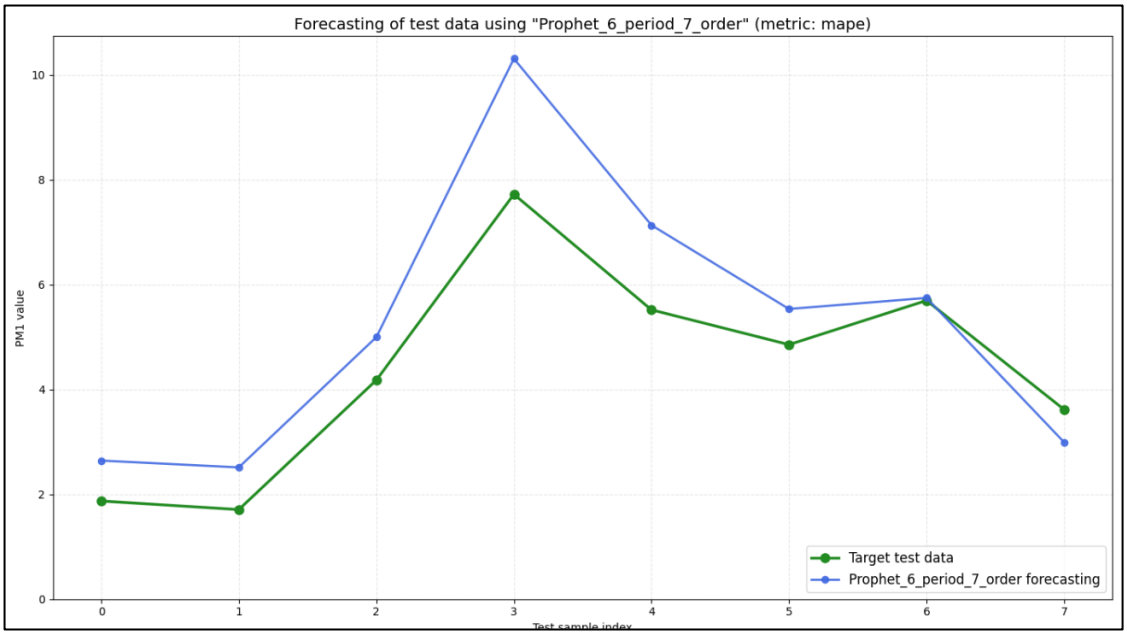

Рисунок 3.22 – Прогноз на тестовому наборі для станції 1872 (найкраща модель: Prophet_6_period_7_order)

Значно вища відносна помилка порівняно з станцією ВНТУ зумовлена високим рівнем шуму в даних навіть після агрегації та наявністю численних короткочасних піків, не пов’язаних із регіональними епізодами перенесення сахарського пилу (локальні викиди, пил від будівництва, дим від спалювання тощо). Ці піки не були включені до holidays_df, оскільки не відповідали критеріям регіонального характеру та пороговим значенням алгоритму виявлення аномалій

Отже, навіть на станціях із суттєво зашумленими даними та домінуванням локальних джерел забруднення запропонований підхід на основі Prophet із кастомною субдобової сезонністю та чітко виділеними епізодами сахарського пилу зберігає працездатність і забезпечує прийнятну точність прогнозування

3.4 Висновки

У третьому реалізовано інформаційну технологію короткострокового прогнозування концентрації PM1 з урахуванням епізодів перенесення сахарського пилу територією України.

Для структурного опису запропонованої технології використано нотацію UML. Сукупне використання UML-діаграм забезпечує цілісне уявлення про логіку роботи інформаційної технології, її місце в інфраструктурі та взаємодію компонентів, що полегшує подальшу реалізацію та супровід програмної системи.

Запропоновано та впроваджено ефективний алгоритм автоматичного виявлення аномальних епізодів, що дозволило сформувати очищений датафрейм holidays_df. Такий підхід забезпечує адекватне моделювання рідкісних, але екстремально сильних зовнішніх подій без їх дублювання чи надмірного переоцінення.

Розроблено та протестовано кілька моделей прогнозування, зокрема моделі часових рядів.

Найкращі результати на всіх етапах продемонстрували моделі Prophet. На станції ВНТУ найкраща конфігурація Prophet_6_period_21_order забезпечила на

валідаційному наборі MAPE = 9,17 % (RMSE = 2,495), а на незалежному тестовому наборі - MAPE = 7,38 % (RMSE = 1,276).

Проведено тестування розробленого підходу на даних додаткових станцій моніторингу. На станції 1612 найкращий результат дала модель Prophet_8_period_21_order (MAPE = 18,40 %, RMSE = 1,243), на станції 1872 - Prophet_6_period_7_order (MAPE = 25%, RMSE = 1,228). Погіршення точності порівняно з основною станцією пояснюється вищим рівнем шуму та домінуванням локальних джерел забруднення, не пов’язаних із сахарськими епізодами.

4. ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу так і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу» є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями [32].

Таблиця 4.1 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	4	5
2. Ринкові переваги (наявність аналогів)	3	4	3
3. Ринкові переваги (ціна продукту)	3	4	3
4. Ринкові переваги (технічні властивості)	5	3	4
5. Ринкові переваги (експлуатаційні витрати)	4	4	4
6. Ринкові перспективи (розмір ринку)	2	3	3
7. Ринкові перспективи (конкуренція)	3	2	3
8. Практична здійсненність (наявність фахівців)	4	4	4
9. Практична здійсненність (наявність фінансів)	2	2	2
10. Практична здійсненність (необхідність нових матеріалів)	3	3	4
11. Практична здійсненність (термін реалізації)	4	4	4

Продовження таблиці 4.1

12. Практична здійсненність (розробка документів)	3	3	3
Сума балів	40	40	42
Середньоарифметична сума балів $СБ_c$	40.6		

За результатами розрахунків, наведених в таблиці 4.1, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в [32].

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою « Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу » становить 40,6 бала, що, відповідно до [32], свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

4.2 Розрахунок узагальненого коефіцієнта якості розробки

Узагальнений коефіцієнт якості ($В_н$) для нового технічного рішення розрахуємо за формулою [33]:

$$В_н = \sum_{i=1}^{k} \alpha_i \cdot \beta_i , \qquad (4.1)$$

де k – кількість найбільш важливих технічних показників, які впливають на якість нового технічного рішення;

α_i – коефіцієнт, який враховує питому вагу *і-го* технічного показника в загальній якості розробки. Коефіцієнт α_i визначається експертним шляхом і при цьому має виконуватись умова $\sum_{i=1}^{k} \alpha_i = 1$;

β_i – відносне значення *і-го* технічного показника якості нової розробки.

Результати порівняння зведемо до таблиці 4.2.

Таблиця 4.2 – Порівняння основних параметрів розробки та аналога.

Показники (параметри)	Одиниця вимірю-вання	Аналог	Проектований продукт	Відношення параметрів нової розробки до аналога	Питома вага показника
1.Кількість використаних моделей машинного навчання	од.	1	3	3	0,15
2. Попередня обробка та очистка даних	од.	1	2	2	0,2
3. Точність прогнозу	%	83	93	1,12	0,3
4.Кількість станцій, дані яких прогнозувались	од.	3	4	1,33	0,2
5.Кількість графіків розвідувального аналізу	од.	2	8	4	0,15

Узагальнений коефіцієнт якості ($B_н$) для нового технічного рішення складе:

$$\mathrm{B_н} = \sum_{i=1}^{k} \alpha_i * \beta_i = 3 \cdot 0{,}15 + 2 \cdot 0{,}2 + 0{,}3 \cdot 1{,}12 + 0{,}2 \cdot 1{,}33 + 4 \cdot 0{,}15 = 2{,}05$$

Отже за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,05 рази.

4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов’язані з проведенням науково-дослідної роботи на тему «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

4.3.1 Витрати на оплату праці

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників ($З_о$) розраховуємо у відповідності до посадових окладів працівників, за формулою [32]:

$$З_о = \sum_{i=1}^{k} \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.2)$$

де k – кількість посад дослідників залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – число днів роботи конкретного дослідника, дн.;

T_p – середнє число робочих днів в місяці, T_p=20 дні.

$$З_о = 25200{,}00 \cdot 10 / 20 = 12600{,}00 \text{ (грн.)}$$

Проведені розрахунки зведемо до таблиці 4.3.

Таблиця 4.3 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту (проєктний менеджер)	25200,00	1260,00	10	12600,00
Консультант з екологічних питань	22000,00	1100,00	5	5500,00
Інженер-програміст	28000,00	1400,00	30	42000,00
Фахівець з аналітично-математичних досліджень	24000,00	1200,00	10	12000,00
Консультант-аналітик цифрових обчислюваних систем	27500,00	1375,00	5	6875,00
Всього				78975,00

Основна заробітна плата робітників

Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт НДР на тему «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу» розраховуємо за формулою:

$$З_р = \sum_{i=1}^{n} C_i \cdot t_i \,, \tag{4.3}$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$\mathrm{С_і} = \frac{\mathrm{М_м} \cdot \mathrm{К_і} \cdot \mathrm{К_с}}{T_р \cdot T_{\mathrm{зм}}}, \tag{4.4}$$

де $М_М$ – розмір мінімальної місячної заробітної плати, приймемо $Мм$=8000,00 грн;

$К_і$ – коефіцієнт міжкваліфікаційного співвідношення (табл. Б.2, додаток Б) [32];

$К_с$ – мінімальний коефіцієнт співвідношень місячних тарифних ставок;

$Т_р$ – середнє число робочих днів в місяці, приблизно $Т_р$ = 20 дн;

$t_{зм}$ – тривалість зміни, год.

$С_1$ = 8000,00 · 1,10 · 1,15 / (20 · 8) = 63,25 (грн.)

$З_{р1}$ = 63,25 · 8,00 = 506,00 (грн.)

Таблиця 4.4– Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Встановлення допоміжного офісного обладнання	8,00	2	1,10	63,25	506,00
Монтаж робочого місця розробника системи прогнозування	10,00	2	1,10	63,25	632,50
Інсталяція програмного забезпечення	3,00	5	1,70	97,75	293,25
Встановлення цифрових обчислювальних систем	5,00	4	1,50	86,25	431,25
Тренування цифрової експериментальної моделі	5,00	4	1,50	86,25	431,25
Формування бази даних прогнозного аналізу	16,00	3	1,35	77,63	1242,00
Інші допоміжні роботи	12,00	3	1,35	77,63	931,56
Всього					4467,81

Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 … 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = (З_о + З_р) \cdot \frac{Н_{дод}}{100\%}, \quad (4.5)$$

де $Н_{дод}$ – норма нарахування додаткової заробітної плати. Приймемо 10%.

$З_{дод} = (78975{,}00 + 4467{,}81) \cdot 10\% / 100\% = 8344{,}28,$ (грн.)

4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_н = \left(З_о + З_р + З_{дод}\right) \cdot \frac{Н_{зп}}{100\%}, \quad (4.6)$$

де $Н_{зп}$ – норма нарахування на заробітну плату. Приймаємо 22%.

Зн = (78975,00 + 4467,81+ 8344,28) · 22% / 100% = 20193,16 (грн.)

4.3.3 Сировина та матеріали

Витрати на матеріали (*М*), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^{n} Н_j \cdot Ц_j \cdot К_j - \sum_{j=1}^{n} В_j \cdot Ц_{в\,j}, \quad (4.7)$$

де $Н_j$ – норма витрат матеріалу *j*-го найменування, кг;

n – кількість видів матеріалів;

$Ц_j$ – вартість матеріалу *j*-го найменування, грн/кг;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = *1,1 … 1,15*);

$В_j$ – маса відходів *j*-го найменування, кг;

$Ц_{вj}$ – вартість відходів *j*-го найменування, грн/кг.

$М_1$ *= 3,000 · 186,00 · 1,05 - 0 · 0 = 585,90* (грн.)

Проведені розрахунки зведемо до таблиці.

Таблиця 4.5 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 од, грн	Норма витрат, од.	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Папір канцелярський офісний ECONOMIC (A4-500)	171,00	3,000	0	0	513,00
Папір для заміток ECONOMIC (A5)-60	77,00	4,000	0	0	308,00
Начиння канцелярське 4Office	374,00	2,000	0	0	748,00
Картридж для принтера HP LJ 33A	1329,00	2,000	0	0	2658,00
Флеш пам'ять USB Kingston DataTraveler Exodia 64GB	229,00	2,000	0	0	458,00
Всього					4865,00

4.3.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі *($К_в$)*, які використовують при проведенні НДР на тему «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу», розраховуємо, згідно з їхньою номенклатурою, за формулою:

$$К_в = \sum_{j=1}^{n} Н_j \cdot Ц_j \cdot К_j \,, \tag{4.8}$$

де H_j – кількість комплектуючих j-го виду, шт.;

$Ц_j$ – покупна ціна комплектуючих j-го виду, грн;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = *1,1 … 1,15*).

$К_в = 1 \cdot 3399.00 \cdot 1,1 = 3789.9$ *грн.*

Проведені розрахунки зведемо до таблиці 4.6.

Таблиця 4.6 – Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
SSD-накопичувач внутрішній Kingston NV3 1TB M.2 2280 NVMe PCIe 4.0 x4 3D NAND (SNV3S/1000G)	1	3399,00	3738,90
USB-хаб Defender 83505 Septima Slim 7-port USB2.0	1	961,00	1057,10
Всього			4796,00

4.3.5 Спецустаткування для наукових (експериментальних) робіт

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{спец} = \sum_{i=1}^{k} Ц_i \cdot C_{пр.i} \cdot K_i, \quad (4.9)$$

де $Ц_i$ – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{пр.i}$ –кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, (K_i = *1,10…1,12);*

k – кількість найменувань устаткування.

$B_{спец} = 37199,00 \cdot 1 \cdot 1,1 = 40918,95$ (грн.)

Отримані результати зведемо до таблиці:

Таблиця 4.7 – Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Серверне обладнання на основі ПК INTEL CORE I3 12100F / GTX 1650	1	37199,00	40918,90
Маршрутизатор інтернет WiFi6 Xiaomi Router AX3000T (DVB4423GL)	1	1777,00	1954,70
Всього			42873,60

4.3.6 Програмне забезпечення для наукових (експериментальних) робіт

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{прг} = \sum_{i=1}^{k} Ц_{iпрг} \cdot C_{прг.i} \cdot K_i, \quad (4.10)$$

де $Ц_{iпрг}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{прг.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10…1,12$);

k – кількість найменувань програмних засобів.

$B_{прг} = 410,00 \cdot 2 \cdot 1,1 = 861,00$ (грн.)

Отримані результати зведемо до таблиці:

Таблиця 4.8 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Високошвидкісний доступ до мережі (міс)	2	410,00	861,00
ОС Windows PRO 11	1	6999,00	7 698,90
Всього			8 559,90

4.3.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_в} \cdot \frac{t_{вик}}{12}, \quad (4.11)$$

де $Ц_б$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$A_{обл} = (43199{,}00 \cdot 2) / (3 \cdot 12) = 2399{,}39$ (грн.)

Проведені розрахунки зведемо до таблиці 4.9.

Таблиця 4.9 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Програмно-аналітичний комплекс Компютер INTEL CORE I5 13400F / RTX 3060 12GB / B760 / DDR5 2x16GB 6000Mhz / SSD 480GB / 600W	43199,00	3	2	2399,94
Персональний комп'ютер Ноутбук Acer Extensa 15 EX215-57-57UN	25999,00	3	2	1444,38

Продовження таблиці 4.9

Спеціалізоване робоче місце розробника інформаційної технології	10000,00	5	2	333,33
Пристрій виводу текстової інформації Принтер HP Laser	10599,00	4	2	441,63
Приміщення лабораторії розробки та дослідження	450000,00	35	2	2142,86
Прикладний пакет Microsoft Office 2024	9875,00	3	2	548,61
Мережеве обладнання передачі цифрових даних	6700,00	4	2	279,17
Всього				7589,92

4.3.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію (B_e) розраховуємо за формулою:

$$B_e = \sum_{i=1}^{n} \frac{W_{yi} \cdot t_i \cdot Ц_e \cdot K_{вni}}{\eta_i}, \quad (4.12)$$

де W_{yi} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

$Ц_e$ – вартість 1 кВт-години електроенергії, грн; приймемо Цe = 12,56 грн;

$K_{вni}$ – коефіцієнт, що враховує використання потужності, $K_{вni} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$B_e = 0,6 \cdot 240,0 \cdot 12,56 \cdot 0,95 / 0,97 = 753,60$ (грн.)

Проведені розрахунки зведемо до таблиці.

Таблиця 4.10 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Програмно-аналітичний комплекс Компютер INTEL CORE I5 13400F / RTX 3060 12GB / B760 / DDR5 2x16GB 6000Mhz / SSD 480GB / 600W	0,6	240,0	1771,34
Персональний комп'ютер Ноутбук Acer Extensa 15 EX215-57-57UN	0,08	240,0	236,18
Спеціалізоване робоче місце розробника інформаційної технології	0,07	240,0	206,66
Пристрій виводу текстової інформації Принтер HP Laser	0,12	2,4	3,54
Мережеве обладнання передачі цифрових даних	0,10	240,0	295,22
Серверне обладнання на основі ПК INTEL CORE I3 12100F / GTX 1650 4GB	0,4	240,0	1180,90
Всього			3693,84

4.3.9 Службові відрядження

Витрати за статтею «Службові відрядження» розраховуємо як 20…25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{св} = \left(З_о + З_р\right) \cdot \frac{H_{св}}{100\%}, \quad (4.13)$$

де $H_{св}$ – норма нарахування за статтею «Службові відрядження», приймемо $Н_{св} = 20\%$.

$B_{св}$ = (78975,00 + 4467,81) · 20 / 100% = 16688,56 (грн.)

4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати розраховуємо як 30…45% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{сп} = \left(З_о + З_р\right) \cdot \frac{Н_{сп}}{100\%}, \quad (4.14)$$

де $Н_{сп}$ – норма нарахування за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації», приймемо $Н_{сп}$= 30%.

$B_{сп}$ = (78975,00 + 4467,81) · 30 / 100% = 25032,84 (грн.)

4.3.11 Інші витрати

Витрати за статтею «Інші витрати» розраховуємо як 50…100% від суми основної заробітної плати дослідників та робітників за формулою:

$$І_в = \left(З_о + З_р\right) \cdot \frac{Н_{ів}}{100\%}, \quad (4.15)$$

де $Н_{ів}$ – норма нарахування за статтею «Інші витрати», приймемо $Н_{ів}$ = 50%.

$І_в$ = (78975,00 + 4467,81) · 50 / 100% = 41721,40 (грн.)

4.3.12 Накладні (загальновиробничі) витрати

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100…150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = \left(З_о + З_р\right) \cdot \frac{Н_{нзв}}{100\%}, \quad (4.16)$$

де $Н_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», приймемо $Н_{нзв}$ = 100%.

$B_{нзв}$ = (78975,00 + 4467,81) · 100 / 100% = 83442,81 (грн.)

Витрати на проведення науково-дослідної роботи на тему «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = З_о + З_р + З_{дод} + З_н + М + К_в + В_{спец} + В_{прг} + А_{обл} + В_е + В_{св} + В_{сп} + І_в + В_{нзв}. \quad (4.17)$$

$B_{заг}$= 78975,00+4467,81+8344,28+20193,16 +4865,00+4796,00+42873,60+ 8559,90+ 7589,92+3693,84+16688,56 +25032,84 +41721,40 +83442,81 = 351244,12 (грн.)

Загальні витрати *ЗВ* на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (4.18)$$

де η - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, приймемо η =0,9.

ЗВ = 372372,18 / 0,9 = 390 271,24 (грн.)

4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

Результати дослідження проведені за темою «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу» передбачають комерціалізацію протягом 4-х років реалізації на ринку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

ΔN – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються;

Таблиця 4.11 – Ріст споживачів

Показник	1-й рік	2-й рік	3-й рік	4-й рік
Збільшення кількості споживачів, осіб	500	900	1100	1150

N– кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, приймемо 6000 осіб;

$Ц_б$ – вартість послуги у році до впровадження інформаційної системи, приймемо 7200,00 грн;

$\pm \Delta Ц_o$ – зміна вартості послуги від впровадження результатів, приймемо 828,00 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta П_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [32]:

$$\Delta П_i = \left(\pm \Delta Ц_o \cdot N + Ц_o \cdot \Delta N\right)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.19)$$

деλ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2025 році ставка податку на додану вартість складає 20%, а коефіцієнт λ *=0,8333*;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту). Приймемо ρ =38*%*;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2025 році ϑ *=18%*;

Збільшення чистого прибутку 1-го року:

$\Delta П_1$ = (828,00·6000,00+7200,00·500)·0,83·0,38·(1-18/100)=2215924,70 (грн.)

Збільшення чистого прибутку 2-го року:

$\Delta П_2$ = (828,00·6000,00+7200,00·1400)·0,83·0,38·(1-18/100)=3891834,14 (грн.)

Збільшення чистого прибутку 3-го року:

$\Delta П_3$ = (828,00·6000,00+7200,00·2500)·0,83·0,38·(1-18/100)=5940167,90 (грн.)

Збільшення чистого прибутку 4-го року:

$\Delta П_4$ = (828,00·6000,00+7200,00·3650)·0,83·0,38·(1-18/100)=8081607,74 (грн.)

Приведена вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \quad (4.20)$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, τ =0,1;

t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$ПП$ =2215924,70/(1+0,1)^1+3891834,14/(1+0,1)^2+5940167,90/(1+0,1)^3+

+8081607,74/(1+0,1)^4=2015386,09+3214734,83+4464536,62+5522229,75=

=15216887,29 (грн.)

Величина початкових інвестицій PV, які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{інв} \cdot ЗВ, \quad (4.22)$$

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{інв}$=2;

ЗВ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 390271,24 (грн.)

$PV = k_{інв} \cdot ЗВ = 2 \cdot 390271{,}24 = 780542{,}48$ (грн.)

Абсолютний економічний ефект $E_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV. \tag{4.22}$$

де *ПП* – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 15216887,29 (грн.);

PV – теперішня вартість початкових інвестицій, 780542,48 (грн.)

$E_{абс} = ПП - PV = 15216887{,}29 - 780542{,}48 = 14436344{,}81$ (грн.)

Внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_в = \sqrt[Тж]{1 + \frac{E_{абс}}{PV}} - 1, \tag{4.23}$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, 14436344,81 грн;

PV – теперішня вартість початкових інвестицій, 780542,48 (грн.);

$T_ж$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

$E_в = \sqrt[Тж]{1 + \frac{E_{абс}}{PV}} - 1 = (1+14436344{,}81 / 780542{,}48)^{1/4}-1=1{,}1$

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{мін}$:

$$\tau_{мін} = d + f, \tag{4.24}$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2025 році в Україні d =0,1;

f – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,25.

$\tau_{\text{мін}}$= 0,1+0,25 = 0,35 < 1,1 свідчить про те, що внутрішня економічна дохідність інвестицій $E_в$, вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_в}, \tag{4.25}$$

де $E_в$ – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок}= 1 / 1,1 = 0,90 \text{ р.}$$

$T_{ок} <$ 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

4.5 Висновки

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу» становить 40,6 бала, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

При оцінюванні за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,05 рази.

Також термін окупності становить 0,90 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу».

ВИСНОВКИ

У процесі виконання магістерської кваліфікаційної роботи було здійснено ґрунтовний аналіз предметної області, розкрито сутність поставленої проблеми та обґрунтовано актуальність задачі прогнозування поширення пилу Сахари в повітрі над територією України.

У першому розділі проаналізовано проблематику забруднення атмосферного повітря сахарським пилом як складову ширшої екологічної проблеми якості повітря. Детально розглянуто основні можливості, переваги та недоліки наявних систем моніторингу якості повітря. На основі цього аналізу обрано оптимальні засоби й інформаційні технології для розв'язання поставленої задачі, наведено їх опис і здійснено аргументоване обґрунтування вибору.

Другий розділ присвячено підготовці даних і проведенню розвідувального аналізу для задачі прогнозування поширення пилу Сахари в Україні. Було сформовано та опрацьовано набір даних, а також розроблено ноутбук, який дає змогу візуалізувати показники якості повітря за окремими годинами та датами. Це дозволило наочно дослідити просторовий розподіл показника PM_1 на території Вінницької області та краще зрозуміти особливості структури даних.

У третьому розділі розроблено інформаційну технологію прогнозування поширення пилу Сахари в повітрі України на основі алгоритмів обробки даних і моделей прогнозування. Проведено експериментальні дослідження на реальних даних з різних станцій моніторингу та продемонстровано роботу розробленої інформаційної технології в цілому, з акцентом на забезпеченні точності й своєчасності прогнозів рівня забруднення повітря сахарським пилом. Виконано відбір оптимальної моделі, якою виявився Prophet: для даних зі станції ВНТУ ця модель показала найкращі результати з показниками RMSE = 1.276 та MAPE = 7.38%.

У четвертому розділі магістерської кваліфікаційної роботи було проведено дослідження, яке показало, що комерційний потенціал розробки інформаційної технології аналізу та прогнозування поширення пилу Сахари у повітрі України за

даними громадського моніторингу оцінюється як високий, досягаючи 40,6 бала. Технічно ця розробка перевершує існуючі аналоги, з узагальненим коефіцієнтом якості, що виявляється в 2,05 рази вищим. Термін окупності становить 0,90 року, що відповідає вимогам комерційної привабливості та може зацікавити потенційних інвесторів.

Таким чином, створена інформаційна технологія, яка дозволяє більш точно прогнозувати поширення пилу Сахари в атмосферному повітрі України за даними громадського моніторингу показника PM_1, який найкраще характеризує вміст цього пилу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Скринник Т.В., Мокін В.Б., Розроблення інформаційної технології аналізу та прогнозування поширення пилу сахари у повітрі україни за даними громадського моніторингу на основі Prophet. LV Всеукраїнська науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет (2026) URL: https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2026/paper/view/26669/21948 . (Дата звернення: 09.12.2025).

2. Air Quality Index Scale and Color Legend URL: https://aqicn.org/scale/ (дата звернення: 23.10.2025).

3. Advances in air quality research – current and emerging challenges. URL: https://acp.copernicus.org/articles/22/4615/2022/ (дата звернення: 23.10.2025).

4. Review of the UK Air Quality Index : a report by the Committee on the Medical Effects of Air Pollutants / Chairman J. G. Ayres, Chairman of the Standards Advisory Subgroup S. Holgate. – London : Health Protection Agency, 2011. – Crown Copyright. – Produced for the Committee on the Medical Effects of Air Pollutants. – URL: https://web.archive.org/web/20111115230340/http://www.comeap.org.uk/images/stories/Documents/Reports/comeap%20review%20of%20the%20uk%20air%20quality%20index.pdf (дата звернення: 23.10.2025).

5. AQI (Air Quality Index) in Ukraine – real-time air pollution, IQAir, 2025. URL: https://www.iqair.com/ukraine (дата звернення: 23.10.2025).

6. World Health Organization. Ambient air pollution: Health impacts. WHO, 2023. URL: https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health (дата звернення: 23.10.2025).

7. Respiratory system URL: completehttps://commons.wikimedia.org/wiki/File:Respiratory_system_complete_en.sv (дата звернення: 23.10.2025).

8. SaveEcoBot. URL: https://www.saveecobot.com/ (дата звернення: 24.10.2025).

9. Карта моніторингу якості повітря EcoCity. URL: https://reborn.eco-city.org.ua/ (дата звернення: 24.10.2025).

10. 6 Major Causes of Air Pollution and Most Common Pollutants. URL: https://www.machengineering.com/major-causes-of-air-pollution/ (дата звернення: 24.10.2025).

11. Sharp increase in Saharan dust intrusions over the western Euro-Mediterranean in February–March 2020–2022 and associated atmospheric circulation. URL: https://acp.copernicus.org/articles/24/4083/2024 / (дата звернення: 24.10.2025).

12. Супутникові знімки Землі WorldView. URL: https://worldview.earthdata.nasa.gov / (дата звернення: 24.10.2025).

13. Van Rossum, G., & Drake, F. The Python Language Reference. Python Software Foundation, 2024. URL: https://docs.python.org/3/reference/ (дата звернення: 27.10.2025).

14. McKinney, W. Python for Data Analysis. 3rd ed. O'Reilly Media, 2022.

15. Harris, C. R. et al. Array programming with NumPy. Nature, 2020. DOI: 10.1038/s41586-020-2649-2

16. The Pandas Development Team. pandas-dev/pandas: Pandas 2.2.2 documentation., 2024. URL: https://pandas.pydata.org/ (дата звернення: 27.10.2025).

17. Hunter, J. D. Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 2007.

18. Virtanen, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 2020.

19. Бібілотека scikit-learn. URL: https://scikit-learn.org/stable/

20. Plotly Technologies Inc. Plotly Open Source Graphing Library for Python., 2024. URL: https://plotly.com/python/ (дата звернення: 28.10.2025).

21. Prophet. URL: https://pypi.org/project/fbprophet(дата звернення: 29.10.2025).

22. Taylor S. J., Letham B. Forecasting at scale. PeerJ Preprints. 2017. 5:e3190v2. DOI: https://doi.org/10.7287/peerj.preprints.3190v21 (дата звернення: 29.10.2025).

23. Hyndman R. J., Athanasopoulos G. Forecasting: principles and practice. 2nd ed. Oxford : Oxford University Press, 2014.

24. Air Quality Monitoring from EcoCity. URL: https://www.kaggle.com/datasets/vbmokin/air-quality-monitoring-from-ecocity (дата звернення: 29.10.2025).

25. Anomaly detection Sahara Vinnytsia. URL: https://www.kaggle.com/code/tarasskrynnyk/anomaly-detection-sahara-vinnytsia (дата звернення: 03.11.2025).

26. Sahara Influence forecasting and EDAv2. URL: https://www.kaggle.com/code/tarasskrynnyk/sahara-influence-forecasting-and-edav2 (дата звернення: 21.11.2025).

27. Tylkowski, J., Hojan, M. “Threshold Values of Extreme Hydrometeorological Events on the Polish Baltic Coast.” Water, 2018, 10(10), 1337

28. Sahara's Dust in the Region - 2D Analysis. URL: https://www.kaggle.com/code/vbmokin/sahara-s-dust-in-the-region-2d-analysis (дата звернення: 03.11.2025).

29. Object Management Group. Unified Modeling Language (UML) Specification, Version 2.5.1. OMG, 2017.

30. Мокін В. Б., Дратований М. В. Наука про дані: машинне навчання та інтелектуальний аналіз даних : електронний навчальний посібник комбінованого (локального та мережевого) використання. Вінниця : ВНТУ, 2024, 258 с. URL: https://docs.vntu.edu.ua/card.php?id=8163 (дата звернення: 10.11.2025).

31. Шмундяк Д. О., Копняк В. Є. Метод ідентифікації локальних аномалій значень показників стану довкілля з використанням декомпозиції на півхвилі, Вісник Вінницького політехнічного інституту, 2024. № 1, с. 88–100. DOI: https://doi.org/10.31649/1997-9266-2024-172-1-88-100 (дата звернення: 11.11.2025).

32. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

33. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа – Вінниця : ВНТУ, 2016. – 113 с.

Додаток А

Технічне завдання

Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ

____ д.т.н., проф. Віталій МОКІН

«__» _____2025 року

ТЕХНІЧНЕ ЗАВДАННЯ
на магістерську кваліфікаційну роботу
ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ПОШИРЕННЯ ПИЛУ САХАРИ У ПОВІТРІ УКРАЇНИ ЗА ДАНИМИ ГРОМАДСЬКОГО МОНІТОРИНГУ
08-34.МКР.012.00.000 ТЗ

Керівник: д.т.н., проф.

_____ Олександр МОКІН

«__» _____ 2025 р.

Розробив студент гр. 2ІСТ-24м

_____Тарас СКРИННИК

«__» _____ 2025 р.

Вінниця 2025

1. Підстава для проведення робіт.
Підставою для виконання роботи є наказ №__ по ВНТУ від «__» _____2025р., та індивідуальне завдання на МКР, затверджене протоколом №__ засідання кафедри САІТ від «__» _____ 2025р.
2. Джерела розробки:
1) Наука про дані: машинне навчання та інтелектуальний аналіз даних: електронний навчальний посібник комбінованого (локального та мережевого) використання / В. Б. Мокін, М. В. Дратований – Вінниця: ВНТУ, 2024. – 258 с. – URL: https://docs.vntu.edu.ua/card.php?id=8163
2) Шмундяк Д. О., Копняк В. Є. Метод ідентифікації локальних аномалій значень показників стану довкілля з використанням декомпозиції на півхвилі, Вісник Вінницького політехнічного інституту, 2024. № 1, с. 88–100. DOI: https://doi.org/10.31649/1997-9266-2024-172-1-88-100
3. Мета і призначення роботи.
Метою дослідження є підвищення точності прогнозування поширення пилу Сахари в атмосферному повітрі України за даними громадського моніторингу шляхом створення інформаційної техноголії
4. Вихідні дані для проведення робіт.
Air Quality Monitoring from EcoCity. URL: https://www.kaggle.com/datasets/vbmokin/air-quality-monitoring-from-ecocity
5.Методи дослідження.
В даній роботі використовуються методи часових рядів та машинного навчання для прогнозування поширення пилу Сахари.
6. Етапи роботи і терміни їх виконання:
а) Загальна характеристика об’єкту дослідження _____ – _____
б) Підготовка та розвідувальний аналіз даних _____ – _____
в) Створення інформаційної технології прогнозування _____ – _____
г) Прогнозування поширення пилу _____ – _____
д) Економічна частина _____ – _____
е) Оформлення матеріалів до захисту МКР _____ – _____
7.Очікувані результати та порядок реалізації.
Отримання інформаційної технології аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу.
8. Вимоги до розробленої документації.
Текстова та ілюстративна частини роботи оформлені у відповідності до вимог «Методичних вказівок до виконання магістерських кваліфікаційних робіт для студентів спеціальності 126 «Інформаційні системи та технології» (освітня програма «Інформаційні технології аналізу даних та зображень»).
9. Порядок приймання роботи.
Публічний захист «___» _____ 2025 р.
Початок розробки «___» _____ 2025 р.
Граничні терміни виконання МКР «___» _____ 2025 р.
Розробив студент групи 2ІСТ-24м _____ Тарас СКРИННИК

Додаток Б

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: « Інформаційна технологія аналізу та прогнозування поширення пилу Сахари у повітрі України за даними громадського моніторингу »

Тип роботи: магістерська кваліфікаційна робота

Підрозділ: кафедра САІТ, ФІІТА, гр. 2ІСТ-24м

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism 18,88%

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне):

- ■ Запозичення, виявлені у роботі, є законними і не містять ознак плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту
- □ У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- □ У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як спроб укриття плагіату, фабрикації, фальсифікації, що суперечить вимогам законодавства та нормам академічної доброчесності. Робота до захисту не приймається.

Експертна комісія:

Віталій МОКІН, зав. каф. САІТ _____
(підпис)

Сергій ЖУКОВ, доц. каф. САІТ _____
(підпис)

Особа, відповідальна за перевірку _____ Сергій ЖУКОВ
(підпис)

З висновком експертної комісії ознайомлений(-на)

Керівник _____ Олександр МОКІН д.т.н., проф.
(підпис)

Здобувач _____ Тарас СКРИННИК

Додаток В

Фрагмент лістингу програми

```
# Import libraries
import random
import os
import sys
import numpy as np
import pandas as pd
import requests
# Date
import datetime as dt
from datetime import date, timedelta, datetime
# EDA
import matplotlib.pyplot as plt
from matplotlib.pylab import rcParams
import plotly.express as px
import plotly.graph_objects as go
from plotly.offline import init_notebook_mode
init_notebook_mode(connected=True)
import seaborn as sns
# Time Series - EDA and Modelling
import statsmodels.api as sm
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.stattools import adfuller
from statsmodels.tsa.seasonal import seasonal_decompose
from statsmodels.tsa.arima_model import ARIMA
from scipy.stats import shapiro, probplot
# Metrics
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_error, mean_absolute_percentage_error
# Modeling and preprocessing
from sklearn.preprocessing import StandardScaler, MinMaxScaler
from sklearn.model_selection import train_test_split, GridSearchCV
from prophet import Prophet
from statsmodels.stats.stattools import jarque_bera
import warnings
from contextlib import contextmanager
warnings.filterwarnings("ignore")
warnings.filterwarnings("ignore", category=UserWarning)
warnings.filterwarnings("ignore", category=RuntimeWarning)
```

```
warnings.filterwarnings("ignore", message="Iteration limit reached")
warnings.filterwarnings("ignore", message="The optimizer returned code 9")
@contextmanager
def suppress_output():
    with open(os.devnull, 'w') as devnull:
        old_stdout, old_stderr = sys.stdout, sys.stderr
        try:
            sys.stdout, sys.stderr = devnull, devnull
            yield
        finally:
            sys.stdout, sys.stderr = old_stdout, old_stderr
def get_data(df, id_station):
    if id_station in df['id_station'].unique():
        station_data = df[df['id_station'] == id_station].copy()
        station_data['datetime'] = pd.to_datetime(station_data['date']) + pd.to_timedelta((station_data['time_point'] - 1) * 20, unit='m')
        new_dataframe = pd.DataFrame({'ds': station_data['datetime'], 'y': station_data['value']})
        return new_dataframe
    else:
        print(f"Station ID {id_station} does not exist in the dataset.")
column_names = ['id_station', 'location', 'date', 'time_point', 'Column6','indicator','dimension','value']
df = pd.read_csv('/kaggle/input/air-quality-monitoring-from-ecocity/ECOCITY_Archive_651_561_2024-03-25_2024-05-10.csv', sep=',',
names=column_names)
df = get_data(df, 1315)
df
# Descriptive statistics and basic visualization
print(f"Date range: {df['ds'].min()} — {df['ds'].max()}")
print(f"Number of observations: {len(df)}")
# Statistical summary
stats = df['y'].describe()
display(stats)
# Additional metrics
print("Coefficient of variation (CV):", round((stats['std'] / stats['mean']) * 100, 2), "%")
print("Skewness:", round(df['y'].skew(), 3))
print("Kurtosis:", round(df['y'].kurtosis(), 3))
# Visualization of PM1 values
fig, axes = plt.subplots(1, 3, figsize=(15, 4))
# Time series
axes[0].plot(df['ds'], df['y'], color='steelblue', linewidth=2)
axes[0].set_title("PM1 Concentration Over Time")
axes[0].set_xlabel("Date")
axes[0].set_ylabel("PM1 (µg/m³)")
```

```
axes[0].grid(alpha=0.5)
# Histogram
axes[1].hist(df['y'], bins=20, color='orange', alpha=0.7, edgecolor='black')
axes[1].set_title("PM1 Value Distribution")
# Boxplot
axes[2].boxplot(df['y'], vert=False, patch_artist=True,
            boxprops=dict(facecolor='lightblue'))
axes[2].set_title("PM1 Boxplot")
plt.tight_layout()
plt.show()
# Correlation analysis and lag relationships
# ACF and PACF plots
fig, ax = plt.subplots(1, 2, figsize=(12, 4))
plot_acf(df['y'], lags=30, ax=ax[0])
plot_pacf(df['y'], lags=30, ax=ax[1], method="ywm")
ax[0].set_title("ACF (Autocorrelation)")
ax[1].set_title("PACF (Partial Autocorrelation)")
plt.tight_layout()
plt.show()



# Lag correlation heatmap
max_lag = 10
lags = pd.concat([df['y'].shift(i) for i in range(max_lag + 1)], axis=1)
lags.columns = [f"Lag_{i}" for i in range(max_lag + 1)]
corr = lags.corr()
plt.figure(figsize=(8, 6))
sns.heatmap(corr, annot=True, fmt=".2f", cmap="coolwarm", square=True)
plt.title("Lag Correlation Heatmap (0–10)")
plt.tight_layout()
plt.show()
Resample to daily with max, Mean
df_mean=df.copy()
df.set_index('ds', inplace=True)
df_mean.set_index('ds', inplace=True)
df_mean = df_mean.resample('D').mean()
df_mean.index = df_mean.index.strftime('%Y-%m-%d')
df_mean = df_mean.dropna()
df_mean.reset_index(inplace=True)
df = df.resample('3H').max()
df = df.dropna()
```

```
df.reset_index(inplace=True)
df=df.iloc[:-16].copy()
display(df)
if is_anomalies:
    anomalous_by_level = df[df['y'] > 29]['ds'].copy()
    print(f"   Found {len(anomalous_by_level)} timestamps with PM1 > 29 µg/m³")
    display(anomalous_by_level.dt.strftime('%Y-%m-%d %H:%M').unique())
# Visual diagnostic plot
plt.figure(figsize=(13, 5))
plt.plot(df2['ds'], df2['y_diff'], color='gray', alpha=0.7, label='ΔPM1 (3-hour change)')
plt.scatter(df2[df2['y_diff'].abs() >= 7]['ds'],
            df2[df2['y_diff'].abs() >= 7]['y_diff'],
            color='red', s=80, edgecolor='darkred', label='|ΔPM1| ≥ 7 µg/m³', zorder=5)
plt.axhline(7, color='red', linestyle='--', alpha=0.6)
plt.axhline(-7, color='red', linestyle='--', alpha=0.6)
plt.title('Sharp Changes in PM1 Concentration — Indicators of Saharan Dust Onset')
plt.xlabel('Date')
plt.ylabel('ΔPM1 (µg/m³ per 3 hours)')
plt.legend()
plt.grid(alpha=0.3)
plt.tight_layout()
plt.show()
anomalous_by_jump = df2[df2['y_diff'].abs() >= 7]['ds'].copy()
print(f"   Detected {len(anomalous_by_jump)} sharp jumps (|Δy| ≥ 7 µg/m³)")
# Combine both methods, remove duplicates, and ensure proper datetime type
all_candidates = pd.concat([anomalous_by_level, anomalous_by_jump]).drop_duplicates()
candidates = pd.DatetimeIndex(all_candidates).sort_values()
# Filter: keep only the first date of each cluster (Saharan events last 2–4 days)
sahara_start_dates = [candidates[0]]  # first event is always kept
min_gap_between_episodes = pd.Timedelta(days=4)
for current_date in candidates[1:]:
    if current_date - sahara_start_dates[-1] >= min_gap_between_episodes:
        sahara_start_dates.append(current_date)
    # else: skip — belongs to the same ongoing episode
sahara_start_dates = pd.DatetimeIndex(sahara_start_dates)
print(f"   Final number of clean Saharan dust episodes: {len(sahara_start_dates)}")
print("   Episode start dates:")
for d in sahara_start_dates.strftime('   • %Y-%m-%d'):
    print(d)
if 'is_anomalies' in globals() and is_anomalies and len(sahara_start_dates) > 0:
    holidays_df = pd.DataFrame({
```

```
        'holiday': 'sahara_event',
        'ds': sahara_start_dates,
        'lower_window': 0,
        'upper_window': 3,
        'prior_scale': 80
    })
    print(f"\nProphet holidays dataframe created with {len(holidays_df)} Saharan dust events")
    display(holidays_df)
else:
    holidays_df = pd.DataFrame(columns=['holiday', 'ds', 'lower_window', 'upper_window', 'prior_scale'])
    print("\nAnomaly mode disabled → empty holidays dataframe")
def prophet_modeling(result,
                     series_name,
                     train,
                     test,
                     holidays_df,
                     period_hours,
                     fourier_order_seasonality,
                     forecasting_period,
                     name_model,
                     type_data):
        model = Prophet(daily_seasonality=True,
                     weekly_seasonality=True,
                     yearly_seasonality=False,
                     changepoint_range=0.95,
                     changepoint_prior_scale=0.05, # ← ключове
                     seasonality_prior_scale=10,
                     holidays_prior_scale=80,
                     holidays=holidays_df,
                     seasonality_mode = 'multiplicative')
                     #seasonality_mode = 'additive'


        #model.add_seasonality(name='seasonality', period=period_days,
        # fourier_order=fourier_order_seasonality,
        # mode = 'multiplicative', prior_scale = 0.3)


    # ============================================================
    # CUSTOM SUB-DAILY SEASONALITIES (6h, 12h, 24h) — PROPERLY IN DAYS
    # ============================================================
    # Prophet expects ALL seasonality periods in DAYS, not hours.
    # Our data has 3-hour resolution - we want to capture:
```

```
#   • 6-hour cycle  → period = 6/24  = 0.25 days
#   • 8-hour cycle  → period = 9/24  = 0.3 days
#   • 12-hour cycle → period = 12/24 = 0.50 days
#   • 24-hour cycle → period = 18/24 = 0.75 day
    model.add_seasonality(
    name='custom',
    period=period_hours/24,
    fourier_order=fourier_order_seasonality,
    mode='multiplicative',
    prior_scale=8
    )
    # Training model for df
    model.fit(train)
    # Make a forecast
    future = model.make_future_dataframe(periods = forecasting_period,freq='3H') # freq='3H'
    forecast = model.predict(future)
    # 1. Manual amplification of Saharan dust peaks
    # ---------------------------------------------------------------------
    # Prophet's built-in holiday effect (even with high prior_scale) often underestimates
    # the real magnitude of Saharan dust events (PM₁ can jump 3–6 times).
    # We therefore apply an additional physically-based multiplier only on
    # the identified Saharan episodes (first day + following 2 days).
    sahara_dates = holidays_df[holidays_df['holiday'] == 'anomalous_dates']['ds']
    for date in sahara_dates:
        mask = (forecast['ds'] >= date) & (forecast['ds'] <= date + pd.Timedelta(days=2))
        forecast.loc[mask, 'yhat'] *= 1.7


    # 2. Global bias correction
    # ---------------------------------------------------------------------
    # Prophet sometimes shows a small systematic bias on the training set.
    # We remove it so that the forecast is centred on the observed values.
    train_pred = model.predict(train[['ds']])['yhat']
    bias = (train_pred - train['y']).mean()
    forecast['yhat'] -= bias
    ypred = forecast['yhat'].iloc[-len(test):].values
    # Draw plot of the values with forecasting data
    figure = model.plot(forecast, xlabel = 'ds', ylabel = f"{name_model} for {series_name}")
    # Draw plot with the components (trend and seasonalities) of the forecasts
    figure_component = model.plot_components(forecast)
    # Save results
    n = len(result)
```

```
        result.loc[n,'name_model'] = f"Prophet_{name_model}"
        result.loc[n,'type_data'] = type_data
        result.at[n,'params'] = [period_hours]+[fourier_order_seasonality]
        result.at[n,'ypred'] = ypred

        return result, ypred
%%time
# Models tuning
if is_Prophet:
    for period_hours in [6,8, 12, 18]:
        for fourier_order_seasonality in [7, 12, 21]:
            result, _ = prophet_modeling(result,
                series_name,
                train_ts,
                valid_ts,
                holidays_df,
                period_hours,
                fourier_order_seasonality,
                forecasting_points,
                f'{period_hours}_period_{fourier_order_seasonality}_order',
                'valid')
ef arima_forecasting(result, model, params, name_model, df, type_data):
    # Data df (validation or test) forecasting on the num days by the model
    # with params and save metrics to result

    ypred = model.forecast(steps=len(df))

    n = len(result)
    result.loc[n,'name_model'] = name_model
    result.loc[n,'type_data'] = type_data
    result.at[n,'params'] = params
    result.at[n,'ypred'] = ypred

    return result
%%time
if is_ARIMA:
    # Automatic tuning of the ARIMA model
    exog_train = train_ts['is_sahara_event'] if 'is_sahara_event' in train_ts.columns else None
    exog_valid = valid_ts['is_sahara_event'] if 'is_sahara_event' in valid_ts.columns else None
    model_auto = pm.auto_arima(train_ts['y'],
```

```
        exogenous=exog_train,

        start_p=0, max_p=5,
        start_q=0, max_q=5,
        d=None,
        seasonal=True,
        m=56,
        start_P=0, max_P=3,
        start_Q=0, max_Q=3,
        D=None,

        trace=True,
        error_action='ignore',
        suppress_warnings=True,
        stepwise=True,
        stationary=True,
        trend='c',                          # дозволяємо слабкий тренд
        information_criterion='aic',
        maxiter=100,
        n_fits=80
    )
    print(model_auto.summary())
def model_training_forecasting(result, train_valid, y, test, ytest,
                               name_model, type_model, params, type_test='1'):
    # Model training for df and y
    # Forecasting ypred
    # type_model = 'Prophet' or "ARIMA" or 'Other ML'
    # return params and metrics in the dataframe result

    if type_model=='Prophet':
        season_hours_optimal = params[0]
        fourier_order_seasonality_optimal = params[1]
        model_opt = None
        _, ypred = prophet_modeling(result,
                                    series_name,
                                    train_valid,
                                    test,
                                    holidays_df,
                                    season_hours_optimal,
                                    fourier_order_seasonality_optimal,
                                    forecasting_points,
```

```
                    f'{type_model}_optimal',
                    'test')
elif type_model == 'ARIMA':
# params = [p, d, q] is what you get from auto_arima
    p, d, q = params[0], params[1], params[2]
    # Get EXOGENEOUS VARIABLE
    exog_train_full = train_valid[['is_sahara_event']] if 'is_sahara_event' in train_valid.columns else None
    exog_test = test[['is_sahara_event']] if 'is_sahara_event' in test.columns else None
    print(f"Training SARIMAX({p},{d},{q}) with exogenous Sahara events...")

    from statsmodels.tsa.statespace.sarimax import SARIMAX

    # model training
    model_opt = SARIMAX(
        endog=train_valid['y'],
        exog=exog_train_full,
        order=(p, d, q),
        seasonal_order=(0, 0, 0, 0),  # we do not use seasonality (because m is unstable)
        trend='c',
        enforce_stationarity=False,
        enforce_invertibility=False
    ).fit(disp=False, maxiter=1000, method='lbfgs')
    # Diagnostics
    try:
        fig = model_opt.plot_diagnostics(figsize=(12, 10))
        plt.suptitle(f"SARIMAX({p},{d},{q}) Diagnostics", fontsize=14)
        plt.tight_layout()
        plt.show()
    except:
        print("  → Diagnostics skipped (optional)")
    # FORECAST WITH EXOGENEOUS VARIABLE
    if exog_test is not None and len(exog_test) == len(test):
        forecast_obj = model_opt.get_forecast(steps=len(test), exog=exog_test)
    else:
        forecast_obj = model_opt.get_forecast(steps=len(test))
    ypred = forecast_obj.predicted_mean.values
    print(f"  → SARIMAX forecast completed. Predicted {len(ypred)} points.")
# Scoring and saving results into the dataframe result
n = len(result)-1
result.loc[n,'name_model'] = f"{type_model}_optimal"
result.loc[n,'type_data'] = "test"
```

```
    result.loc[n,'type_model'] = type_model
    result.at[n,'params'] = params
    result.at[n,'ypred'] = ypred
    #result = result_add_metrics(result, n, ytest, ypred)

    return result, model_opt, ypred
def get_optimal_model_and_forecasting(result, main_metrics, start_points):
    # Choosion the optimal model from dataframe result by main_metrics
    # Tuning optimal model for big dataset train+valid
    # Test forecasting and drawing it
    # Returns the optimal model and it's name
    if len(result) > 0:
        # Get parameters of the optimal model from dataframe result by main_metrics
        opt_name_model, opt_type_model, opt_params_model = get_params_optimal_model(result,
                                                           main_metrics)
        # Set datasets for the final tuning and testing by optimal model
        if (opt_type_model=='Prophet') or (opt_type_model=='ARIMA'):
            train_valid = train_valid_ts.copy()
            y_train_valid = train_valid_ts['y'].copy()
            test = test_ts.copy()
            ytest = test_ts['y'].copy()

        # Optimal model training for train+valid and test forecasting
        result, model_opt, ypred = model_training_forecasting(result, train_valid, y_train_valid,
                                              test, ytest,
                                              opt_name_model, opt_type_model,
                                              opt_params_model, '1')

        # Calculation metrics for recovering prediction ypred for test dataset by the optimal model
        result = result_recover_and_metrics(result, test_ts, 'test', start_points)

        # Drawing plot for prediction for the test data
        ytest_plot = ytest.copy()
        ypred_plot = ypred.copy()

        # Drawing
        plt.figure(figsize=(14,8))
        x = np.arange(len(ytest_plot))
        plt.plot(
            x,
            ytest_plot,
```

```
    color='forestgreen',
    linewidth=2.5,
    marker='o',
    markersize=8,
    label="Target test data"
)
plt.plot(
    x,
    ypred_plot,
    color='royalblue',
    linewidth=2,
    marker='o',
    markersize=6,
    alpha=0.9,
    label=f"{opt_name_model} forecasting"
)
plt.title(
    f'Forecasting of test data using "{opt_name_model}" (metric: {main_metrics})',
    fontsize=14
)
plt.xlabel("Test sample index")
plt.ylabel("PM1 value")
plt.grid(True, linestyle='--', alpha=0.3)
plt.ylim(0)
plt.legend(loc='lower right', fontsize=12)
plt.tight_layout()
plt.show()
return opt_name_model
```

Додаток Г

ІЛЮСТРАТИВНА ЧАСТИНА

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ПОШИРЕННЯ ПИЛУ САХАРИ У ПОВІТРІ УКРАЇНИ ЗА ДАНИМИ ГРОМАДСЬКОГО МОНІТОРИНГУ

Нормоконтроль: к.т.н., доцент

_____ Сергій ЖУКОВ

«___» _____ 2025 р.

[27]:

	id_station	location	date	time_point	Column6	indicator	dimension	value	address
0	1315	49.23327 28.409161	2024-03-25	1	18	PM1.0	ug/m3	26.8422	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
1	1315	49.23327 28.409161	2024-03-25	2	20	PM1.0	ug/m3	25.4215	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
2	1315	49.23327 28.409161	2024-03-25	3	18	PM1.0	ug/m3	24.4250	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
3	1315	49.23327 28.409161	2024-03-25	4	20	PM1.0	ug/m3	22.8410	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
4	1315	49.23327 28.409161	2024-03-25	5	20	PM1.0	ug/m3	22.6225	Vinnytsia, str. Khmelnytsky shose, 95, VNTU, b...
...	...	...	...	...	...	...	...	...	...
19642	1876	49.2048561 28.5288355	2024-05-10	68	20	PM1.0	ug/m3	6.3255	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19643	1876	49.2048561 28.5288355	2024-05-10	69	20	PM1.0	ug/m3	5.8295	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19644	1876	49.2048561 28.5288355	2024-05-10	70	20	PM1.0	ug/m3	3.7260	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19645	1876	49.2048561 28.5288355	2024-05-10	71	20	PM1.0	ug/m3	3.9925	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru
19646	1876	49.2048561 28.5288355	2024-05-10	72	20	PM1.0	ug/m3	4.0595	Prov. Bohdana Khmel'nyts'koho, Vinnyts'ki Khutoru

19647 rows × 9 columns

Рисунок Г.1 – Створений датасет Air Quality Monitoring from EcoCity

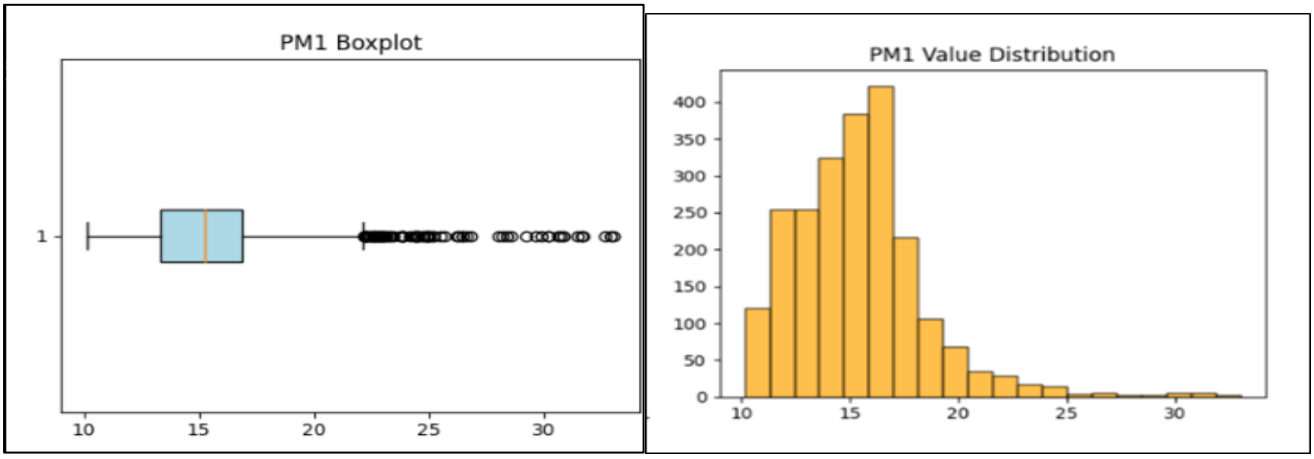

Рисунок Г.2 – Діграми розподілу даних

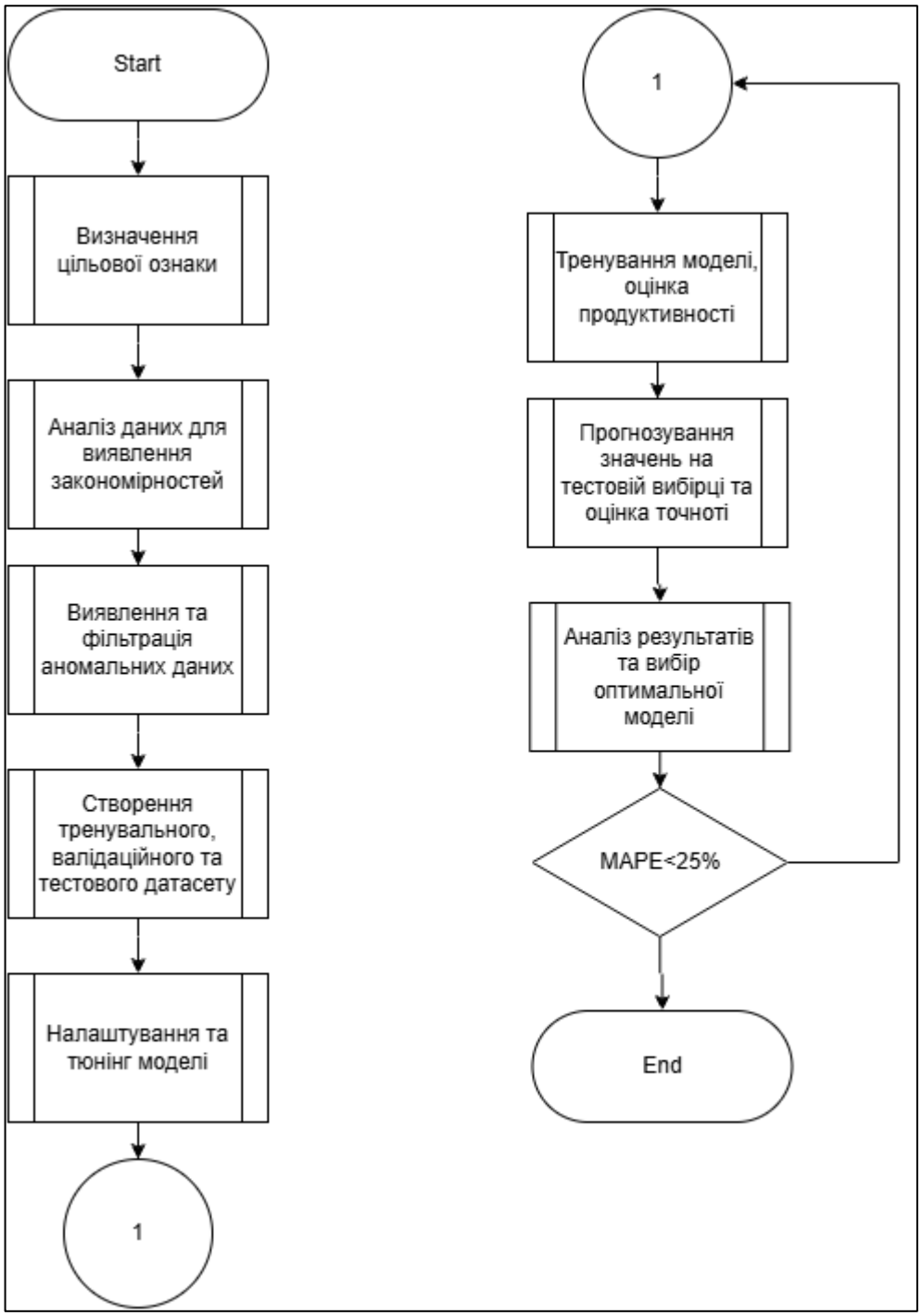

Рисунок Г.3– Блок-схема алгоритму інформаційної технології аналізу та прогнозування поширення пилу Сахари Україною

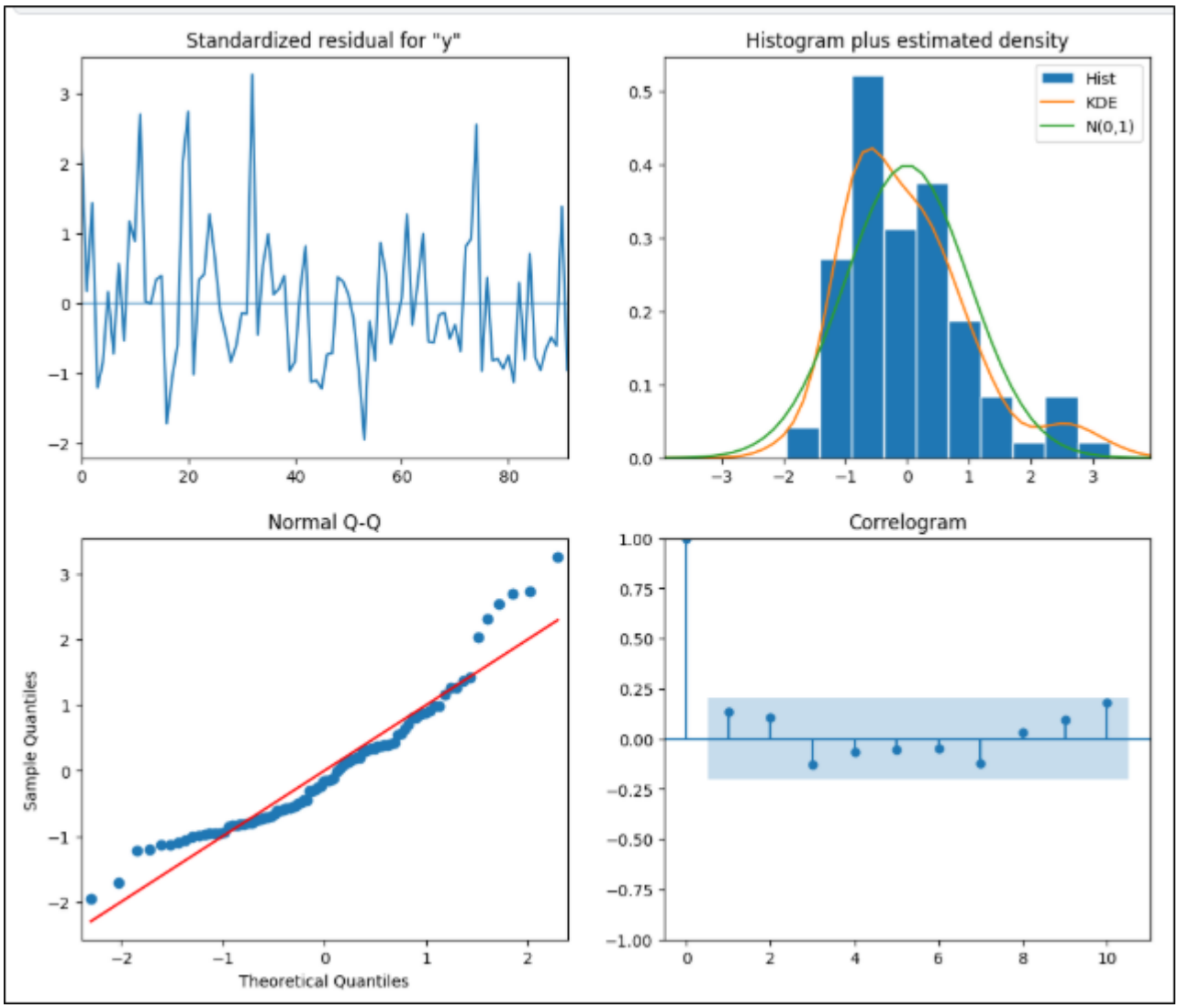

Рисунок Г.4 - Результат діагностики оптимальної моделі ARIMA

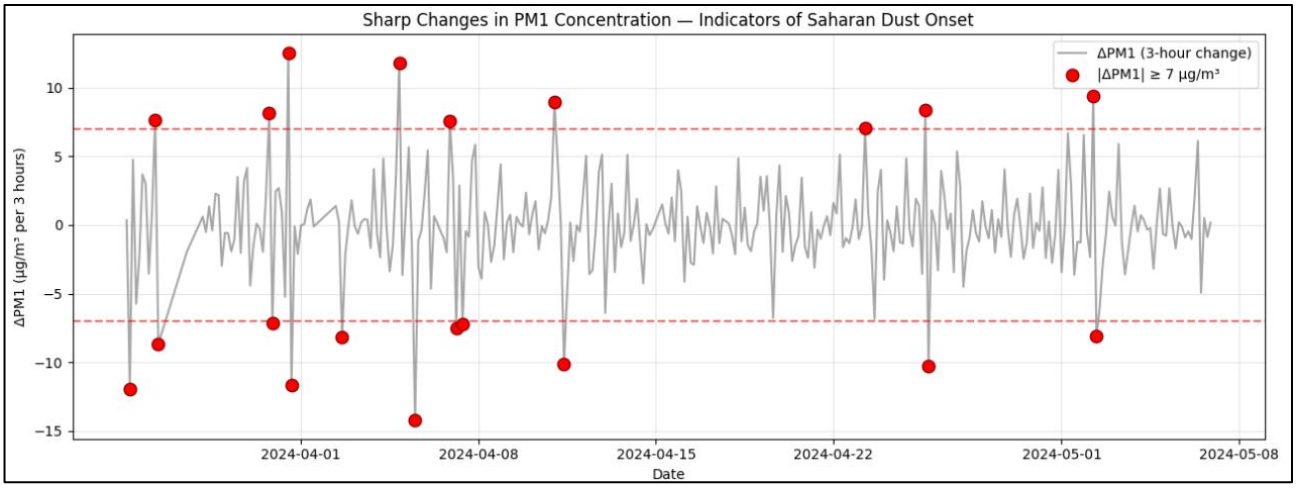

Рисунок Г.5 – Виявлення різких стрибків концентрації |ΔPM1

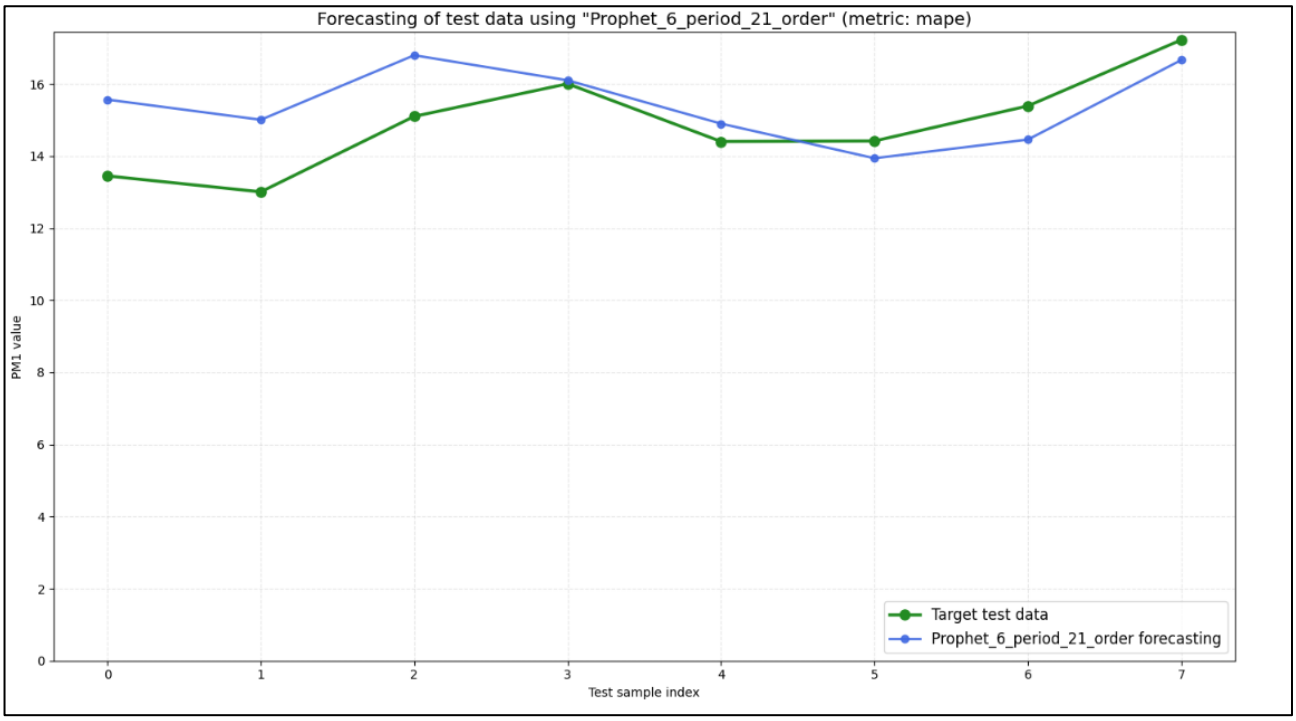

Рисунок Г.6 - Результати прогнозування якості повітря та поширення пилу Сахари за допомогою найкращої моделі для станції ВНТУ 1315