Вінницький національний технічний університет
Факультет комп'ютерних систем і автоматики
Кафедра системного аналізу та інформаційних технологій

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET

Пояснювальна записка до магістерської кваліфікаційної роботи

Виконав: студент 2 курсу, групи 2ІСТ-19м
спеціальності 126 – «Інформаційні системи та технології»
Шмундяк Д. О.

Керівник: д.т.н., проф. Мокін В. Б.

Рецензент: д.т.н., проф. Квєтний Р. Н.

Вінниця ВНТУ – 2020 року

Вінницький національний технічний університет
Факультет комп'ютерних систем і автоматики
Кафедра системного аналізу та інформаційних технологій

Освітньо-кваліфікаційний рівень магістр
Спеціальність 126 – «Інформаційні системи та технології»

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ

_____ д.т.н., проф. В. Б. Мокін

"__" _____ 2020 р.

ЗАВДАННЯ
на магістерську кваліфікаційну роботу студенту
Шмундяку Дмитру Олександровичу

1. Тема роботи: «Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet»,
керівник роботи: д.т.н., проф. Мокін В. Б.,
затверджені наказом закладу вищої освіти від "__"_____2020 року №__
2. Строк подання студентом роботи_____
3. Вихідні дані до роботи:
   - дані моніторингу вод Держводагентства за 2002-2019 роки.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):
   - обґрунтування доцільності створення інформаційної технології пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet
   - розроблення математичного та алгоритмічного апарату інформаційної технології пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet
   - розроблення програмного інструментарію інформаційної технології пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):
   - блок-схема алгоритму інформаційної технології;
   - графік передбачення моделі Prophet;
   - тенденція часового ряду моделі Prophet;
   - річна сезонність часового ряду моделі Prophet;
   - графік дельти передбачення, побудованого моделлю Prophet;
   - графік тренду передбачення, побудованого моделлю Prophet;
   - графік інтегралу значень передбачення, побудованого моделлю Prophet;
   - результат застосування інформаційної технології на реальних даних.

6. Консультанти розділів МКР

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4	Руда Л.П., к.е.н., доц. каф. ЕПВМ		

7. Дата видачі завдання_____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів МКР	Строк виконання етапів роботи	Примітка
1	Аналіз предметної області		
2	Огляд існуючих методів аналізу забруднення води		
3	Розроблення алгоритмічного і математичного апарату		
4	Розроблення програмного інструментарію інформаційної технології		
5	Економічна частина		
6	Оформлення матеріалів до захисту МКР		

Студент _____ Шмундяк Д. О.

Керівник роботи _____ Мокін В. Б.

Рецензент _____ Квєтний Р.Н.

РЕФЕРАТ

Магістерська кваліфікаційна робота: 94 сторінки, 5 таблиць, 62 рисунка, 25 джерел.

Мета роботи – створення інформаційної технології пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними спостережень якості води.

Об’єкт досліджень – процес автоматизації пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними спостережень якості води.

Предмет дослідження – інформаційна технологія пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними спостережень якості води.

У роботі розглянуто поточний стан водних ресурсів у світі й в Україні та сучасні методи аналізу їх стану та протидії розповсюдження забруднення. Було розроблено математичний та алгоритмічний апарат інформаційної технології пошуку можливих джерел забруднення. Проведено опис методів машинного навчання для вирішення завдання контролю забруднення та обрано набір програмно-технічних інструментів для реалізації інформаційної технології. Виконано тестування роботи інформаційної технології на реальних даних.

Прогнозні припущення про розвиток об’єкта дослідження – адаптування розробленої інформаційної технології пошуку можливих джерел підвищеного забруднення річки до реальних даних інших показників якості води та порівняння розрахованих даних із звітами системи підрозділів Держводагентства.

Галузь застосування – інформаційні технології, які направлені на пошук та локалізацію джерел забруднення річки.

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ, ЯКІСТЬ ВОДИ, ЧАСОВИЙ РЯД, МОДЕЛЬ PROPHET, ДЖЕРЕЛО ЗАБРУДНЕННЯ РІЧКИ, PYTHON.

ABSTRACT

Master`s qualification work: 94 pages, 5 tables, 62 pictures, 25 references.

The purpose of this work is to develop information technology for the detection of possible sources of increased anthropogenic load on the river based on the open monitoring data of water quality.

The research object is the automation of detection of possible sources of increased anthropogenic load on the river based on the open monitoring data of water quality.

The subject of the research is the information technology of detecting possible sources of increased anthropogenic load on the river based on the open monitoring data of water quality.

This work gives a review of the current state of water resources in the world and Ukraine particularly and modern methods to analyse them and prevent pollution spreading. The mathematical and algorithmic basis of the proposed information technology was formed. It also contains a description of machine learning methods for solving the problem of pollution control and selection of technology stack to implement information technology. Testing was done based on the real open data.

Estimated assumptions about the development of the research object - an adaptation of the developed information technology of detecting possible sources of increased river pollution to the other water quality parameters and comparing the calculated data with the reports from the subdivisions of the State Agency of Water Resources.

Scope of application – information technologies aimed at detecting sources of water pollution.

INFORMATION TECHNOLOGY, WATER QUALITY, TIME SERIES, PROPHET MODEL, SOURCE OF RIVER POLLUTION, PYTHON.

ЗМІСТ

ВСТУП 8
1 ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET 11
1.1 Аналіз предметної області 11
1.2 Огляд існуючих методів аналізу забруднення вод 17
1.3 Висновки 27
2 РОЗРОБЛЕННЯ МАТЕМАТИЧНОГО ТА АЛГОРИТМІЧНОГО АПАРАТУ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET 28
2.1 Огляд вхідних даних 28
2.2 Опис моделі розв’язання проблеми 34
2.3 Розроблення математичного та алгоритмічного апарату 39
2.4 Висновки 44
3 РОЗРОБЛЕННЯ ПРОГРАМНОГО ІНСТРУМЕНТАРІЮ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET 45
3.1 Вибір мови програмування 45
3.2 Реалізація допоміжних методів 47
3.3 Реалізація головного алгоритму роботи 56
3.4 Тестування 61
3.5 Висновки 62
4 ЕКОНОМІЧНА ЧАСТИНА 63
4.1 Оцінювання комерційного потенціалу розробки інформаційної технології пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet 63
4.2 Прогнозування витрат на виконання науково-дослідної роботи 65

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки 69
4.4 Розрахунок ефективності вкладених інвестицій та період їх окупності 71
4.5 Висновки 75
ВИСНОВКИ 76
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 78
Додаток А Технічне завдання 81
Додаток Б Лістинг програми 83
Додаток В Графічна частина 87

ВСТУП

Стан сучасного клімату та його постійні зміни призводять до зменшення кількості води. До цього всього, обсяги забруднення вод стрімко зростають і, через зменшення кількості води, концентрація забруднюючих речовин також зростає. Тому пріоритетом є заходи для підтримки екологічного стану водних ресурсів, які спрямовані на контроль та виявлення джерел можливого забруднення води. Згідно умов підписаної угоди про асоціацію з ЄС, Україна змушена виконувати правила, затверджені Водною директивою Європейського союзу [1]. Це зобов'язує країну до розробки планів по стабілізації екологічного стану води, який передбачає моніторинг води, проведення фізико-хімічного аналізу її стану тощо.

В багатьох країнах Європи, в тому числі й Україні, мережа моніторингу води не завжди функціонує достатньо добре. Існують великі прогалини у наявних даних, що ускладнює процес знаходження та локалізації джерел забруднення води і, як результат, реалізацію кроків проведення політики регулювання стану води.

Однією з проблемних ділянок на території України є ділянка річки Південний Буг, де відбувається забір води для підприємства «Вінницяводоканал», що є основним водопостачальником міста Вінниця. Через поганий стан очисних споруд, останні декілька років спостерігається регулярне погіршення стану води за основними характеристиками. У відкритому доступі є дані, розміщені Державним агентством водних ресурсів України, які включають інформацію з постів моніторингу про значення різних показників якості води за 1920-2020 роки [2, 3]. Це дає можливість аналізувати їх, але, через наявність прогалин на певні дати та різну періодичність проведених спостережень, не можна однозначно сказати, що є основними джерелами цього забруднення.

Отже, актуальним є розроблення та реалізація інформаційної технології пошуку можливих джерел підвищеного забруднення річки, яка дозволить на основі відкритих даних знайти та локалізувати джерела забруднення річки по роках та локаціях.

Об’єктом даного дослідження є процес автоматизації пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними спостережень якості води.

Предметом даного дослідження є інформаційна технологія пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними спостережень якості води.

Мета дослідження – створення інформаційної технології пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними державного ресурсу спостережень якості води. Досягнення поставленої мети призведе до створення інструментарію, який зробить даний процес прозорим.

Наукова новизна одержаних результатів. Наукова новизна, отримана в результаті проведеного дослідження, полягає у тому, що дістала подальшого розвитку інформаційна технологія пошуку можливих джерел підвищеного антропогенного навантаження на річку за відкритими даними за рахунок використання моделі Prophet з кусково-лінійною апроксимацією тренду для локалізації в часі та просторі забруднюючого впливу по похідній цього тренду. Особливістю запропонованої технології є її стійкість до аномальних завад з різкою зміною якості чи кількості води (аварій, межені, повені тощо), які спеціальним чином враховує блок параметрів «holidays» моделі Prophet, та зміни водності річки, за рахунок аналізу багаторічних даних та виявлення сталих трендів, що мають місце тривалий час.

Практичне значення одержаних результатів полягає у наступному:

- розроблено математичний і алгоритмічний апарат інформаційної технологію пошуку можливих джерел підвищеного забруднення річки;
- за допомогою пакету програм Prophet реалізовано програмний інструментарій, що базується на розробленому алгоритму;
- проведено успішне тестування на реальних даних 8 постів державної системи спостереження ділянки р. Південний Буг, від її витоку до м. Вінниця за 2002-2019 роки.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строго поставленою задачею, детально проаналізованою областю застосування технології та її основних методів, використання математичних методів для доведення теоретичних методів роботи інформаційної технології та побудови математично-алгоритмічного апарату системи, аналізом використаних програмних інструментів та аналізом отриманих даних, які отримані в результаті реалізації інформаційної технології.

Публікації. За результатами магістерської кваліфікаційної роботи опубліковано статтю в науковому фаховому періодичному виданні «Вісник Вінницького політехнічного інституту» категорії Б зі спеціальності 126 – «Інформаційні системи та технології», яке входить в Index Copernicus [4].

1 ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET

1.1 Аналіз предметної області

Можна з впевненістю сказати, що водні ресурси є найголовнішими з багатого переліку усіх природних ресурсів. Поняття водних ресурсів відноситься до усієї води планети, які перебувають у вільному та хімічно незв'язаному стані. В загальному, можна сказати що усі запаси включають в себе підземні води суходолу, величезні запаси вод Світового океану та поверхневі вод (до них відносяться річки та озера). Крім того, запаси води також зосередженні у льодовиках, ґрунтах та атмосферному шарі Землі [5]. Загальний розподіл об’ємів вод планети показано на рисунку 1.1 [6].

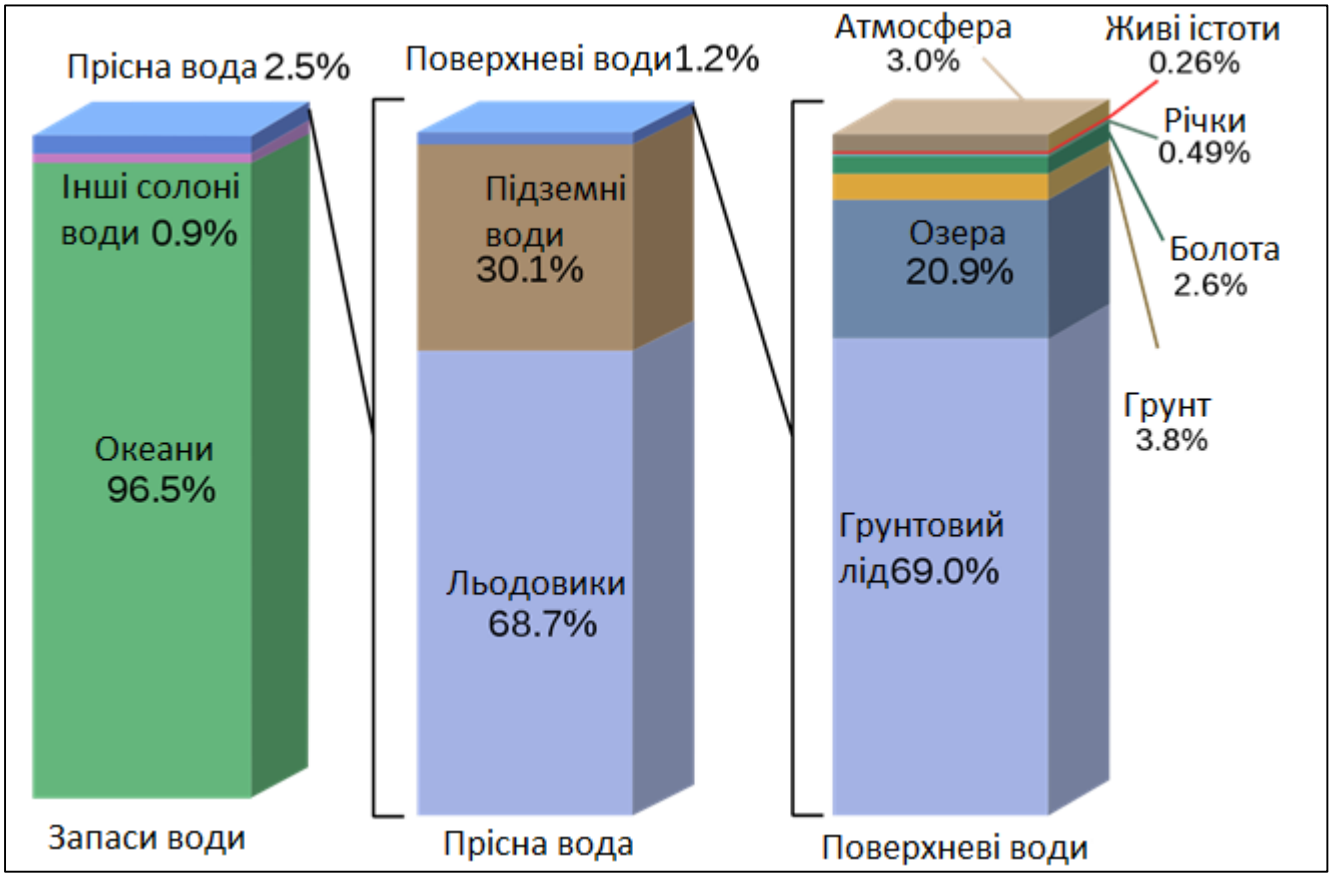


Рисунок 1.1 – Розподіл води на планеті

Із загальної площі планети, яка дорівнює приблизно 510 мільйонів квадратних кілометрів, трішки більше 70% припадає на територію, охоплену водою (у всіх зазначених вище видах). Загальні запаси води нашої планети нараховують приблизно 1.5 мільярди кубічних метрів, з яких, однак, лише 2-3% відсотки припадає на прісні води, які придатні для господарського використання.

Відсоток використання води для різних сфер варіюється для усіх материків та регіонів. Це зумовлене передусім тим, що деякі регіони більш розвинені і розподіл води в них має схожу тенденцію з наголосом на промисловість. В інших ж регіонах переважає сільське господарство. Також, не варто забувати про використання води у побуті, на яке припадає суттєва частина використання усіх водних ресурсів. На рисунку 1.2 наведено кругові діаграми основних споживачів води по регіонах нашої планети [6].

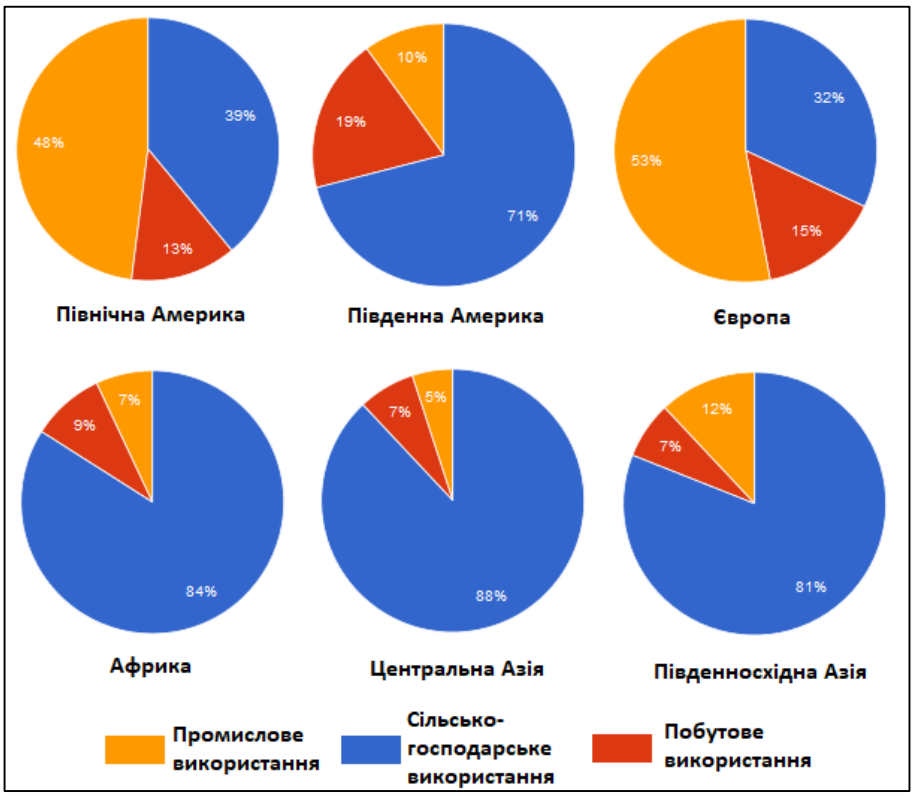


Рисунок 1.2 – Споживання води по регіонах планети

Знайшовши середні значення для кожної сфери-споживача води, можна зробити висновок, що до основних споживачів води відноситься [5, 6]:

- Сільське господарство – підрахунки кажуть, що близько 70% усієї використаної води у світі припадає на зрошення, а 20-25% з усього врожаю є нежиттєздатним, тобто ця вода витрачена дарма. Враховуючи показники щоденної потреби людини в їжі, можна сказати що на одну людину витрачається 2500 літрів води (для порівняння – щоденна потреба у воді для пиття складає всього 5 літрів);
- Промисловість – займає приблизно 20-25% світового споживання води. Головними споживачами є гідроелектростанції та ТЕС (для охолодження), різні нафтопереробні підприємства (хімічні процеси) та багато інших. Вода використовується у виробництві відновлюваної енергії на гідроелектростанціях, де сила води, завдяки оберту турбін, перетворюється на екологічно чисто добуту енергію. Також вода використовується в багатьох широкомасштабних промислових процесах (виробництво термоелектричної енергії, переробка нафти, виробництво добрив). Промислове споживання води, як правило, набагато нижче, ніж водовідведення, через закони, що вимагають очищення промислової сірої води та повернення її в навколишнє середовище. Теплоелектростанції, що використовують градирні, мають велику витрату, майже рівну їх відведенню, оскільки більша частина відведеної води випаровується в рамках процесу охолодження.
- Побутові потреби – 8-10% води використовується у світі для таких потреб побуту як приготування їжі, питна вода, особиста гігієна, прибирання та прання. У більшості розвинених країн вода, яка постачається для домашнього господарства є стандартною для питної води, проте дуже мала її частка споживається саме для пиття або приготування їжі.

Приймаючи до уваги те, що розподіл води по країні не завжди рівномірний (випадок України саме такий, адже на заході країни її запаси досить великі, а в південно-східних регіонах її мало) питання правильного її використання, та на додачу захист від забруднення, стає критично необхідним.

Забрудненням води називається забруднення водних ресурсів, зазвичай, за прямого впливу людини. Забруднення води виникає при занесенні у її середовища забруднювачів. Прикладом є спуск погано очищених стічних вод у природні

водойми, який може призвести до погіршення водних екосистем. Це ж, у свою чергу, може вести до проблем із здоров'ям у людей, які проживають вздовж течії та якимось чином споживають її: вживання у якості питної, купання, поливу городів тощо.

Класифікувати види забруднень води можна на забруднення поверхневих вод та забруднення підземних вод. Джерелами забруднення води можуть бути або точкові, або неточкові джерела. Точковими називають ті джерела, які мають одну ідентифіковану причину забруднення, наприклад, певна очисна споруда. Неточкові ж джерела є більш розкиданими (прикладом є сільськогосподарський стік). Забруднення відбувається з проміжком певного часу, маючи кумулятивний ефект. Впливу можуть зазнавати усі живі організми, які розташовані та мешкають у забруднених водах або піддаються непрямому впливу.

Територія України нараховується понад 70 тисяч річок, більшість з них невеликі (тільки 125 з них в довжині перевищують 100 км). Поверхневі води України зображено на рисунку 1.3 [7].

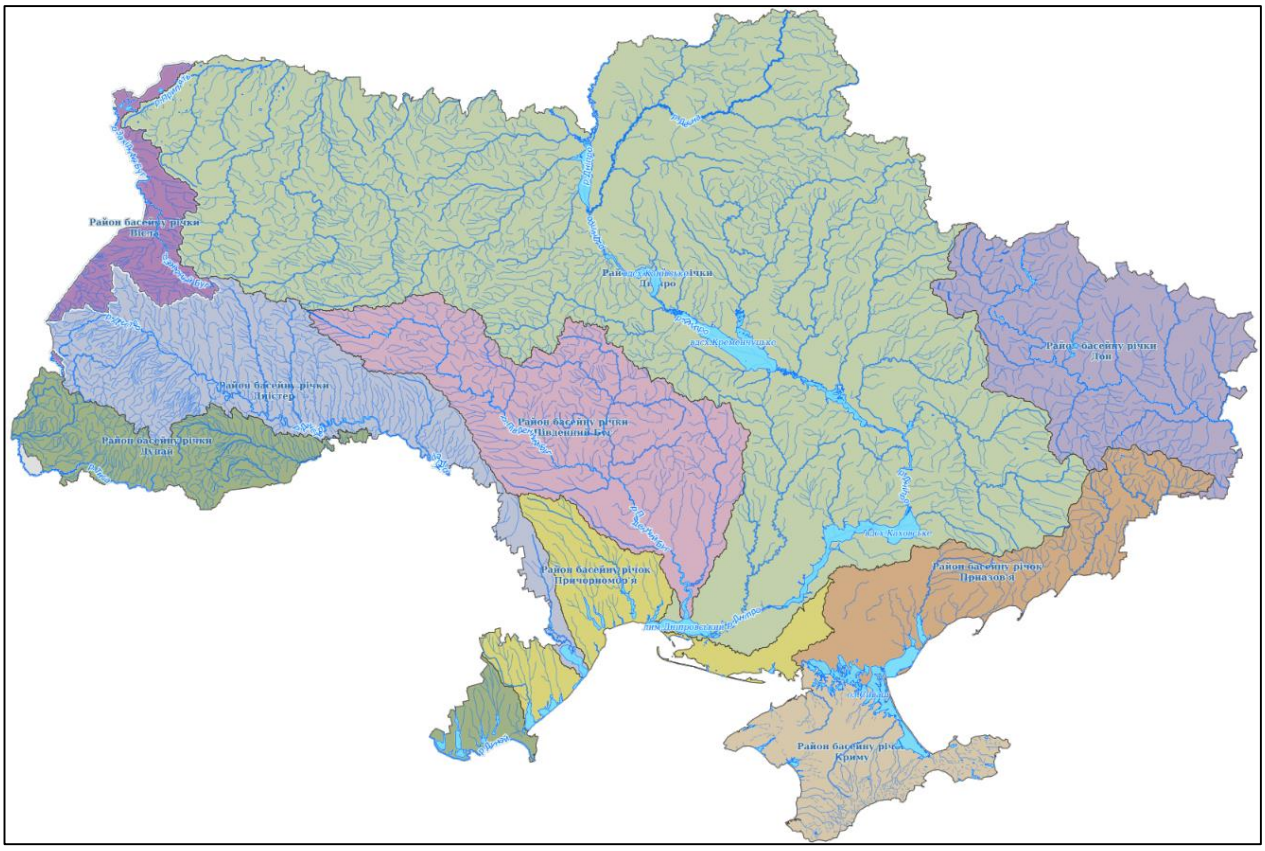


Рисунок 1.3 – Поверхневі ресурси України

Щоправда, існує велика різниця цього показнику в залежності від регіону – так от у Південних регіонах країни значення у 6-7 разів менше ніж у Південно-Західному, а він у свою чергу в 3 рази більший за відповідний показник у Донецькому регіоні. До того ж територіальний розподіл водних ресурсів не відповідає розміщенню промислових підприємств. Регіони Донбасу, Запоріжжя, Крим та південь України, де знаходяться найбільші споживачі води, досить бідні на ці самі ресурси.

Найбільшим і відповідно найголовнішим джерелом водних ресурсів України є річка Дніпро, яка простягається від Північного аж до Південного кордонів країни. ЇЇ ресурси нараховують 80 % усіх водних ресурсів України. Його басейн зображено на рисунку 1.4 [8].

Рисунок 1.4 – Басейн річки Дніпро

В середньому за рік об’єми стоку в гирлі річки досягають 53 м^3, в маловодні роки може зменшуватися до 30-45 км^3. Важливість Дніпра характеризується не тільки тим, що вона є джерелом води для усього населення вздовж течії, але і тим що забезпечує водою великі промислові центри, які знаходяться на півдні України.

Центральне водопостачання населення країни охоплює 70% населення. За рахунок підземних вод забезпечується 20% потреб. Все інше покривається річками: Дніпро, Десна, Південний Буг та інших. Проте більша кількість усіх річок і водоймищ, які забезпечують потреби населення, є екологічно небезпечними. Відхилення показників якості води у деяких містах та регіонах може сягати 75%.

Питання стану річок давно стало питанням національного масштабу [8, 9]. Майже всі водойми мають III-й або IV-й клас якості води, що визнає їх брудними та такими що не є безпечними для людей. Ще жахливішою ситуація є в басейнах Дніпра, Дністра, Південного Бугу, де якість води має V клас (дуже брудна).

Проблема погіршується ще й тим, що хлорування було і досі є основним методом очищення води для питних потреб, яка у свою чергу також веде до утворення небезпечних для людей хлорорганічних сполук. Та навіть за умови, що вода буде правильно очищена, знезаражена і позбавлена від неприємних смаків, її якість знову зіпсується через жахливий стан каналізаційних труб.

Суттєвою проблемою також є погана якість очищення стічних вод. У більшості регіонів погано розвинена інфраструктура очисних та санітарних споруд. Водопроводи можуть бути взагалі не обладнані знезаражувальними установками (Івано-Франківська, Тернопільська та Одеська області). Як результат, переважна більшість усіх виробничих відходів зливається в річку без очищення, що порушає санітарні норми та вимоги.

Якщо розглядати хімічний аналіз проб води, то найбільше відхилення зустрічаються у органолептичними показниками – 70-75%. Також є відхилення по рівню мінералізації води – до 25-30%, а також перевищення ГДК хімічних речовин – 15%.

Основними джерелами забруднення вод є [8]:

- промислові відходи – залізо, ртуть, радіоактивні частинки та синтетичні миючі засоби, які є стійкими до дії води, що дозволяє їм довго в ній знаходитися;
- стічні води – призводить до забруднення за рахунок різних домішок, які містять кислоти, мінеральні солі, нафтопродукти. Також у воду потрапляють

різні шкідливі мікроорганізми, які можуть викликати погіршення здоров’я людей та тварин;

- тверді відходи – забруднення скупченнями нерозчинних відходів, наприклад, пластик;
- фермерські відходи – органічні відходи та різні хімічні добрива, які потрапляють у підземні та поверхневі води;
- теплове забруднення – забруднення спричинене тим, що ТЕС та АЕС спускають теплу воду, що може змінити біологічні процеси мешканців цих водойм аж до їх можливої смерті.

Як результат, це все є причиною величезного переліку захворювань, які виникають через споживання забрудненої води та появи багатьох негативних факторів, які впливають на навколишнє середовище. Саме тому забезпечення усіх галузей підприємства та всіх громадян чистою та безпечною водою є пріоритетним завданням соціально-економічної політики кожної країни, в тому числі й України.

1.2 Огляд існуючих методів аналізу забруднення вод

Головним виконавчим органом України, який реалізує політику управління та використання водних ресурсів є Державне агентство водних ресурсів. Держводагентство проводить заходи, спрямовані на покращення екологічного стану вод, виконує державний моніторинг поверхневих вод та впроваджує плани по управлінню річковими басейнами [10]. У межах кожного річкового басейну функціонує басейнова рада, основним завдання є проведення на локальному рівні заходів щодо узгодження співпраці місцевих органів влади, підприємств і організацій для забезпечення прийнятного екологічного стану водних ресурсів та контроль виконання планів по управлінню ресурсами річкового басейну. Головна сторінка веб-сайту Держводагентства наведено на рисунку 1.5 [10].

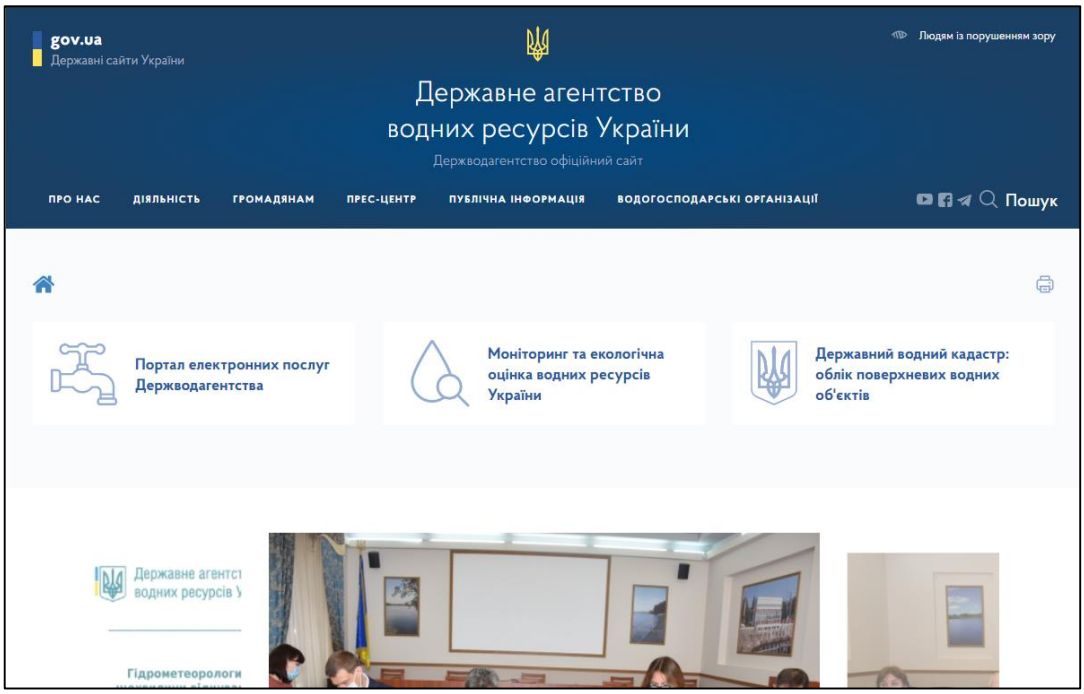


Рисунок 1.5 – Головна сторінка веб-сайту Держводагентства

Важливим механізмом екологічного контролю водних ресурсів є надання регулярної звітності від всіх водокористувачів. Водокористувачі зобов'язані подавати річний звіт за формою 2-ТП (водгосп) до відповідного локального органу управління Держводагентства [11]. Одна з таблиць даної форми наведена на рисунку 1.6.

тис. куб. м

№ рядка	Назва джерела водопостачання або водокористувача	КОДИ			Відстань від гирла, км	Забрано або одержано води за рік												
		типу джерела та водокористувача, що передає	поверхневого водного об'єкта (джерела постачання)	категорії якості води		усього	у тому числі за місяцями											
							I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII
А	Б	В	Г	Д	Е	1	2	3	4	5	6	7	8	9	10	11	12	13
1		/																
2		/																
3		/																
4		/																
5		/																

тис. куб. м

№ рядка	Ліміт викори-стання	Фактично використано води за рік							Передано іншим водокористувачам за рік				Втрати води за рік
		усього	у тому числі на потреби				на інші потреби		без використання		після використання		
			питні і санітарно-гігієнічні	виробничі (техноло-гічні)	зрошен-ня	питні і санітарно-гігієнічні користувачів, що не звітують	код виду викори-стання	об'єм	код категорії якості води	об'єм	код категорії якості води	об'єм	
А	14	15	16	17	18	19	20	21	22	23	24	25	26
1													
2													
3													
4													
5													

Рисунок 1.6 – Форма подачі звітності у питанні водокористування

В рамках звіту необхідно вказати перелік місць звідки проводиться водозабір та зазначити фактичне місце, звідки проводиться забір води, тип водокористування, річні і місячні об’єми отриманої води та спеціальні коди типів джерел водопостачання, які затверджені у Додатку №1 наказу Міністерства екології та природних ресурсів України про порядок ведення обліку водокористування. Наступною важливою частиною звіту є інформація про водовідведення, яка включає в себе назви приймачів стічних вод, річні об’єми відведених вод з поділом на кількість води, яка була очищена перед відводом та ту, яку було викинуто у забрудненому стані. Обов’язково необхідно вказати вміст забруднюючих речовин у зворотних водах. Також додаються різні додаткові показники, такі як: кількість робочих днів, річні ліміти забору, потужності очисних споруд тощо.

Процес звітування спрощується за рахунок спеціального порталу електронних послуг у рамках веб-сайту Держводагентства України [2], його головна сторінка зображена на рисунку 1.7.

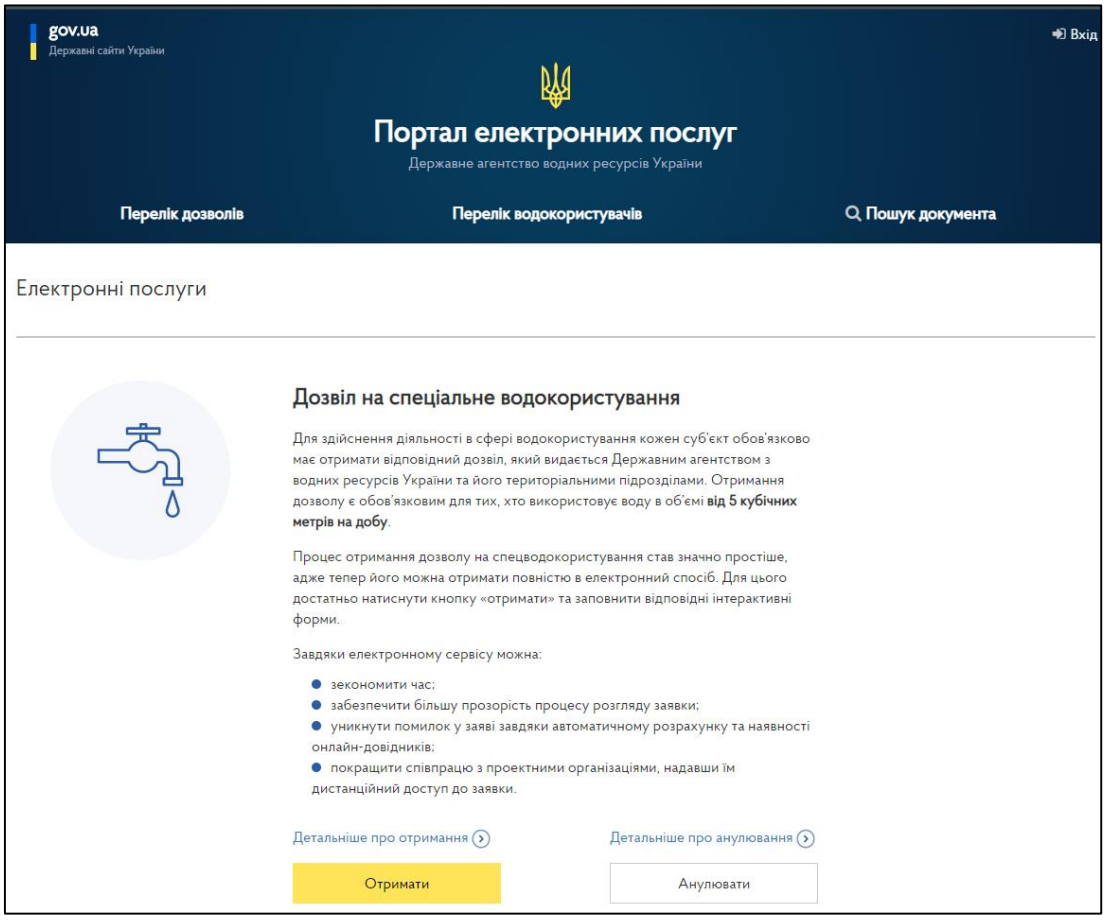


Рисунок 1.7 – Портал електронних послуг Держводагентства

Даний портал дозволяє вирішувати питання дозволів на використання води, можливість переглядати усі отримані дозволи та водокористувачів. Також можна формувати звіти у зручному веб-інтерфейсі та переглядати статистику звітності по водних регіонах, типах підприємств тощо. Для подачі звіту достатньо авторизуватися у системі за допомогою системи електронної ідентифікації, яка зображена на рисунку 1.8.

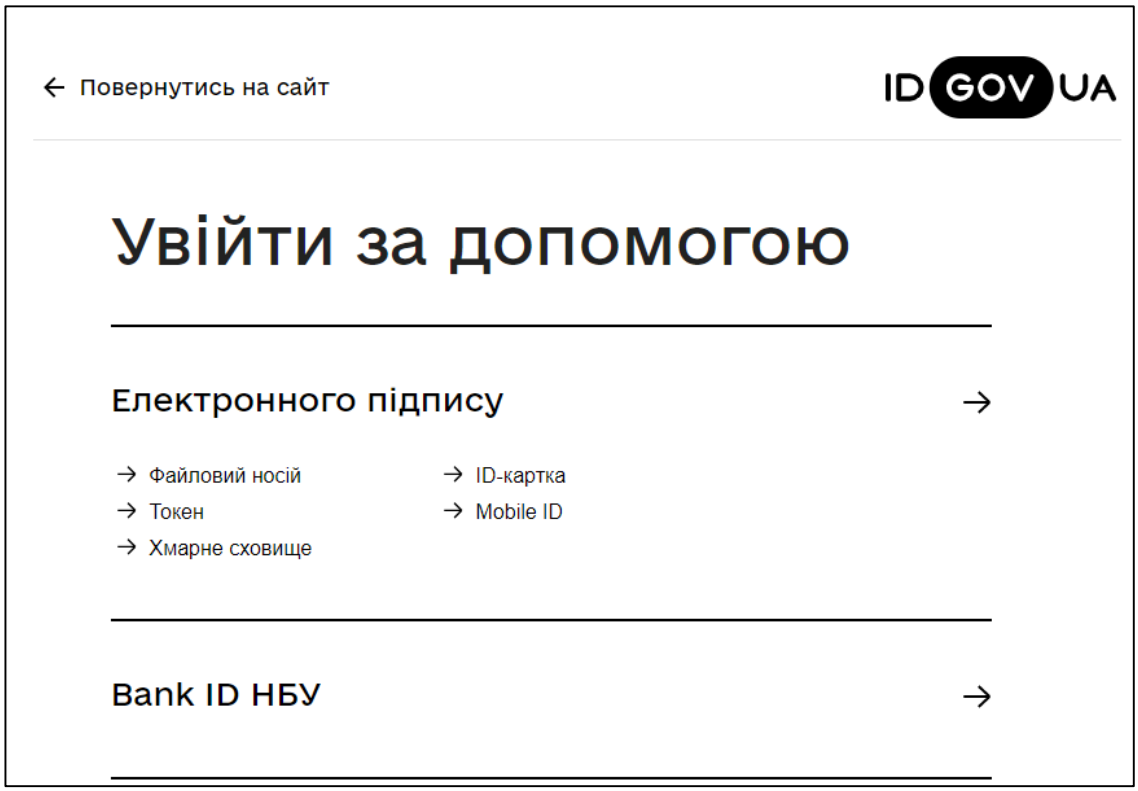


Рисунок 1.8 – Система електронної ідентифікації

Також, як було зазначено раніше, система дозволяє переглядати інформацію про водокористувачів: статус їх поточного дозволу, строки його дії та належність до відповідного органу агентства водних ресурсів. Інформація представляється у формі таблиці, як показано на рисунку 1.9.

306/ВН/49д-20	13.11.2020	36595337 КОМУНАЛЬНЕ ПІДПРИЄМСТВО "ГАЙСИНВОДОКАНАЛ" 23700, ВІННИЦЬКА ОБЛАСТЬ, ГАЙСИНСЬКИЙ РАЙОН, місто ГАЙСИН, вул. Гагаріна, буд. 31;	13.11.2023	Сектор у Вінницькій області Держводагентства	діючий	
305/ВН/49д-20	13.11.2020	20105221 ПРИВАТНО-ОРЕНДНЕ СІЛЬСЬКОГОСПОДАРСЬКЕ ПІДПРИЄМСТВО "НАПАДІВСЬКЕ" 22400, ВІННИЦЬКА ОБЛАСТЬ, КАЛИНІВСЬКИЙ РАЙОН, село НАПАДІВКА, вул. Першотравнева, буд. 3;	13.11.2025	Сектор у Вінницькій області Держводагентства	діючий	
304/ВН/49д-20	12.11.2020	25494693 ПРИВАТНЕ АКЦІОНЕРНЕ ТОВАРИСТВО "КАЛИНІВСЬКИЙ МАШИНОБУДІВНИЙ ЗАВОД" 22400, ВІННИЦЬКА ОБЛАСТЬ, КАЛИНІВСЬКИЙ РАЙОН, місто КАЛИНІВКА, вул. Незалежності, буд. 67;	12.11.2023	Сектор у Вінницькій області Держводагентства	діючий	

Рисунок 1.9 – Таблиця дозволів на водокористування

Для кожного суб’єкта є можливість перегляду детальної інформації, яка є нічим іншим як усією інформацією наведеною у формі 2-ТП (водгосп). На сторінці детальної інформації можна знайти інформацію про найменування водокористувача, мету використання та регіон його розміщення. Також є можливість перегляду списку місць водозабору. Приклад блоку з цією інформацією наведено на рисунку 1.10.

Перелік місць водокористування (водозабір)

Фактичне місце здійснення діяльності	Свердловина N° 1, глибина 80,0 м, дебіт 6,0 куб.м/год., розташована на території підприємства в м. Калинівка Калинівського району Вінницької області, басейн річки Жердь, басейн річки Десна, район басейну річки Південний Буг.
Код типу джерела водопостачання	(60) Підземний водоносний горизонт
Код та назва джерела водопостачання	ЧЕР/Ю.БУГ/0591/0018/Р.ЖЕРДЬ
Код та назва водогосподарської ділянки	М5.4.0.02 р. Південний Буг від гирла р. Іква до г/п Селище
Тип водокористування	Забір з підземних джерел
Розшифровка типу водокористування	Підземні джерела

Фактичне місце здійснення діяльності	Поверхневий водний об'єкт (став), площею 2,43 га, розташований в м. Калинівка Калинівського району Вінницької області, басейн річки Жердь, басейн річки Десна, район басейну річки Південний Буг.
Код типу джерела водопостачання	(40) Ставок руслових, водосховище руслове (крім каскаду дніпровських водосховищ)
Код та назва джерела водопостачання	ЧЕР/Ю.БУГ/0591/0018/Р.ЖЕРДЬ
Код та назва водогосподарської ділянки	М5.4.0.02 р. Південний Буг від гирла р. Іква до г/п Селище
Тип водокористування	Забір з поверхневих джерел
Розшифровка типу водокористування	Не зазначено

Рисунок 1.10 – Місця водозабору для відповідного водокористувача

Крім того, можна відобразити дану інформацію у форматі карти з точними координатами розміщення даного місця. На рисунку 1.11 зображено картографічне представлення місця водокористування.

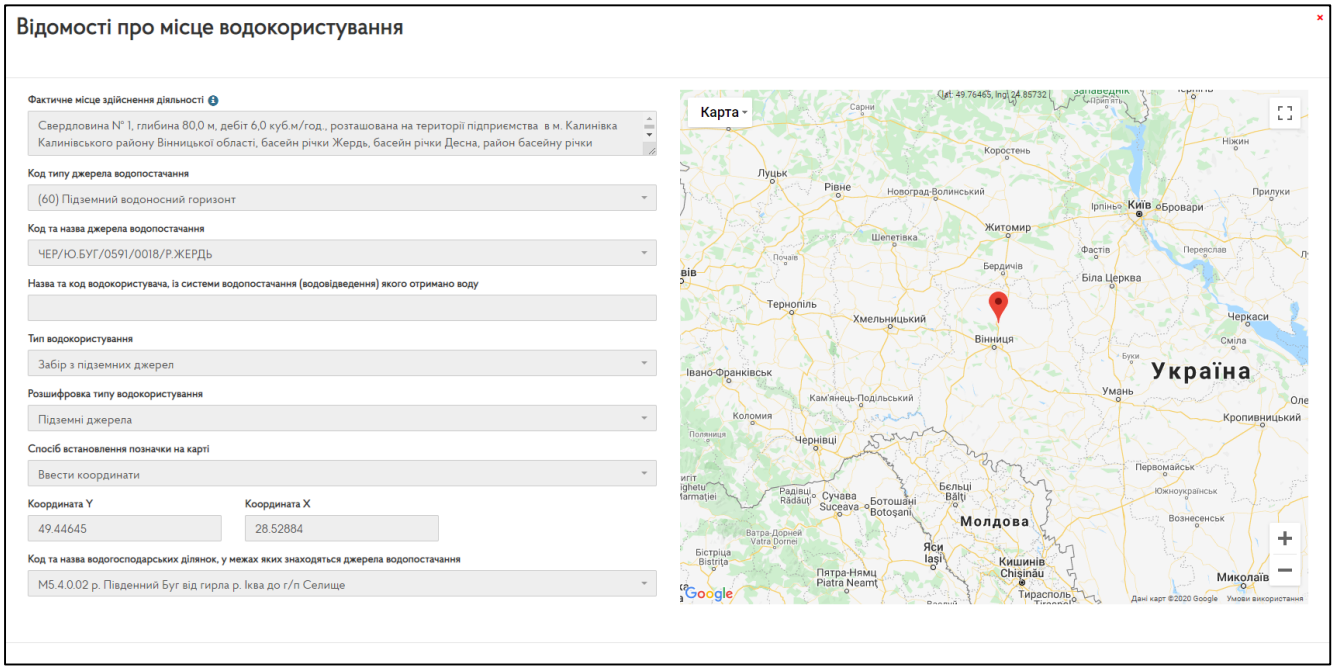


Рисунок 1.11 – Карта з координатами місця водокористування

Аналогічні можливості надаються і для відображення інформації про місця водовідведення, яка зображена на рисунку 1.12.

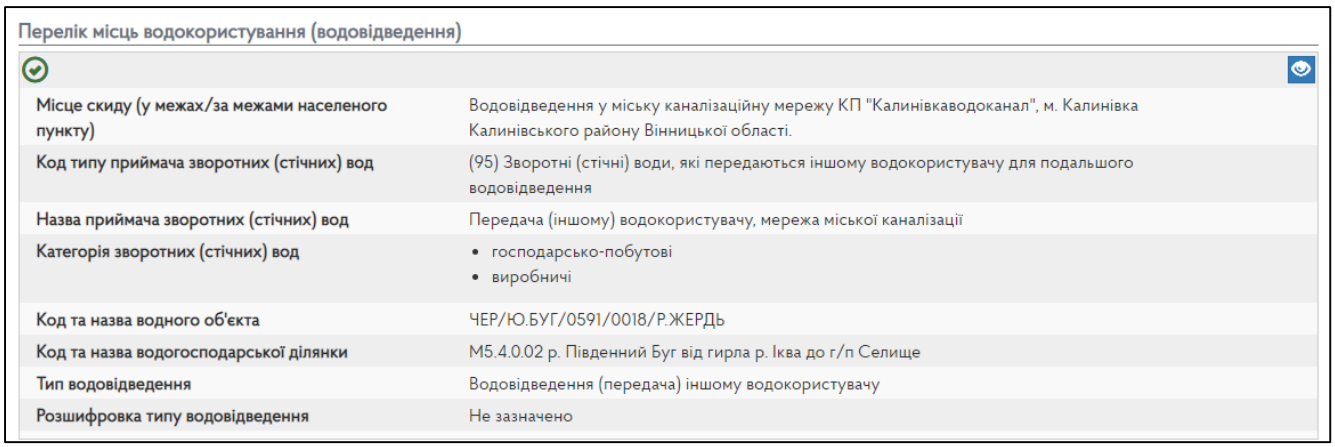


Рисунок 1.12 – Місця водовідведення для відповідного водокористувача

Важливою частиною звітних даних є таблиця з скинутими забруднюючими речовинами. Для кожної речовини вказано їх концентрацію у мг/дм^3, об’єми скиду в г/год та гранично допустимі концентрації. Приклад даної таблиці наведено на рисунку 1.13.

Допустимий обсяг скидання (м^3/год)	7.997
Допустимий обсяг скидання (тис. м^3/рік)	70.057
Фактичний обсяг скидання (м^3/год)	7.997

Забруднюючі речовини, скидання яких нормується	Фактична концентрація, мг/дм^3	Фактичний скид, г/год	Гранично-допустимі концентрації, мг/дм^3	ГДС, г/год	ГДС, перераховані у т/рік
Азот амонійний	0.25	2.00	0.25	2.00	0.018
БСК5	4.18	33.43	4.18	33.43	0.293
ХСК	30.0	239.91	30.0	239.91	2.102
Завислі речовини	6.9	55.18	6.9	55.18	0.483
Нафтопродукти	0.058	0.46	0.058	0.46	0.004
Нітрати	6.65	53.18	6.65	53.18	0.466
Нітрити	0.086	0.69	0.086	0.69	0.006
Сульфати	9.67	77.33	9.67	77.33	0.677
Фосфати	1.03	8.24	1.03	8.24	0.072
Хлориди	13.47	107.72	13.47	107.72	0.944
Сухий залишок	151.5	1211.55	151.5	1211.55	10.613
Залізо	0.386	3.09	0.386	3.09	0.027
Розчинений кисень	4.5	35.99	4.5	35.99	0.315

Інші показники та характеристики зворотних (стічних) вод	водневий показник (pH): в межах 6,5-8,5, температура: перевищення не більше 3°C температури водного обєкта найжаркішого місяця за 10 років

Рисунок 1.13 – Таблиця з переліком скинутих забруднюючих речовин

Усі вище зазначені дані разом з показаннями відповідних характеристик забруднення на постах перед і після подібних підприємств допомагають проаналізувати що стало джерелом забруднення на ділянці річки або ж, якщо різниця значень між постами спостереження не збігається з показниками викидів із звітів, виявити місця нелегальних викидів або можливих прихованих аварій, що сталися на цій ділянці річки.

Ще одним дуже зручним інструментом є генерування статистичний звітів по різній інформації та параметрах: використання води за потребами, скиди стічних вод та багато інших. Список усіх можливих варіантів зображено на рисунку 1.14.

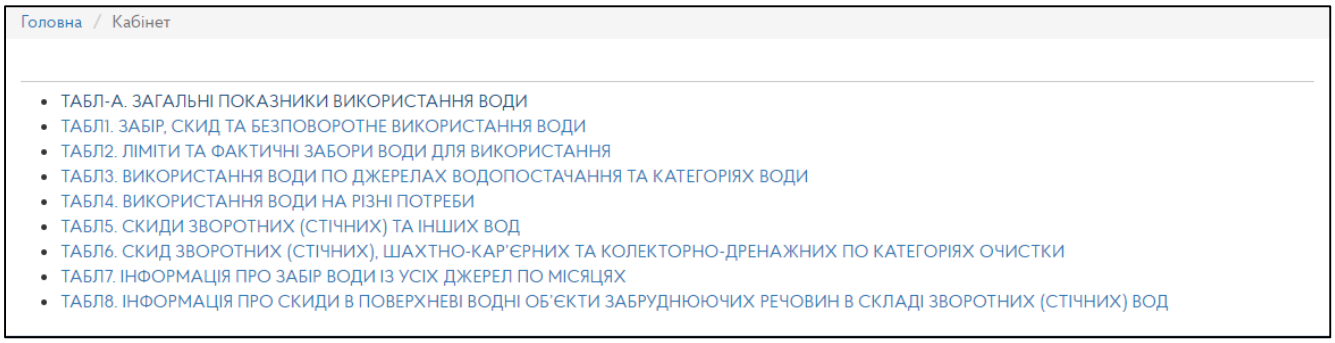


Рисунок 1.14 – Перелік статистичних звітів по водокористуванню

За допомогою меню фільтрації можна обрати часовий період, параметр відбору (регіон водокористування, водні ділянки або види діяльності суб'єктів водокористування) та обрати необхідні значення, як це показано на рисунку 1.15.

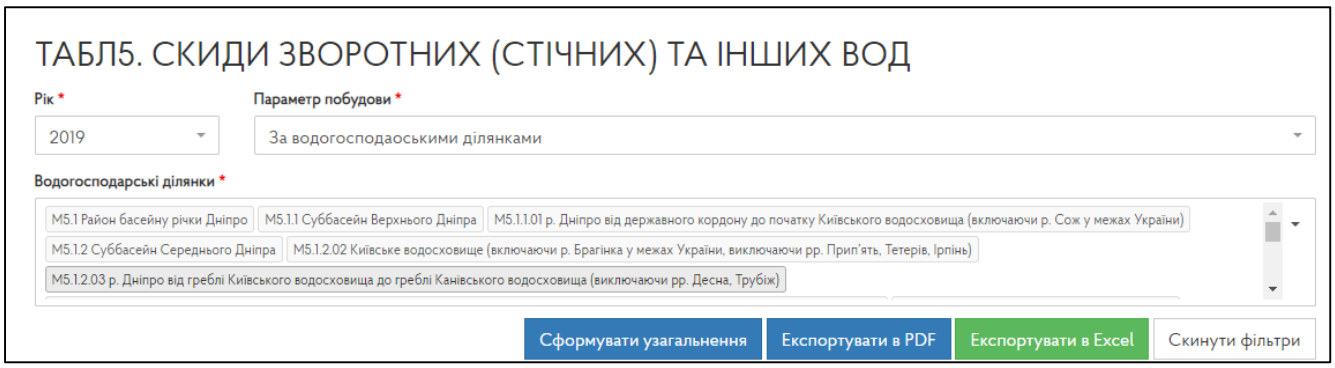


Рисунок 1.15 – Налаштування параметрів статистичного звіту

В результаті, буде сформовано таблицю з даними по вказаних параметрах із можливостю експортувати у зручний формат та завантажити на свій пристрій. Приклад звіту скидів стічних вод для басейну річки Дніпро за 2019 рік наведено на рисунку 1.16.

Назва водогосподарської ділянки	Скинуто зворотних (стічних), шахтно-кар'єрних та колекторно-дренажних вод												Скинуто транзитної води	Скинуто зворотних (стічних) та вод КДС в канали
	Всього	По приймачах								По категоріях води				
		В поверхневі водні об'єкти						В підземні горизонти	Не віднесених до водних об'єктів	Зворотних (стічних)	Шахтно-кар'єрних	Колекторно-дренажних		
		Всього	в тому числі											
			Забруднених		Нормативно чистих без очистки	Нормативно очищених	Некатегорованих							
			Без очистки	Недостатньо очищених										
Район басейну річки Дніпро	196.76	196.55	0.01	2.50	25.75	168.28	-	-	0.22	191.95	2.56	2.25	6.27	-
Суббасейн Верхнього Дніпра	2.29	2.29	-	2.00	-	0.29	-	-	-	0.25	2.00	0.04	-	-
Суббасейн Середнього Дніпра	35.77	35.69	0.01	0.04	9.48	26.16	-	-	0.08	33.61	0.13	2.03	5.43	-
р. Тетерів від витоку до г/п Житомир	21.47	21.47	0.01	0.01	1.81	19.63	-	-	-	21.25	-	0.22	1.25	-
р. Тетерів від г/п Житомир до гирла р. Ірша (включаючи р. Ірша)	3.02	3.01	-	-	0.05	2.97	-	-	0.00	2.92	0.05	0.05	0.06	-
р. Ірпінь	2.21	2.21	-	-	2.21	-	-	-	-	2.21	-	-	1.70	-
р. Рось від витоку до кордону Київської та Черкаської областей	4.62	4.62	-	-	4.62	-	-	-	-	4.62	-	-	2.42	-
Суббасейн Нижнього Дніпра	2.29	2.29	-	2.00	-	0.29	-	-	-	0.25	2.00	0.04	-	-
р. Самара від витоку до гирла р. Вовча	2.00	2.00	-	2.00	-	-	-	-	-	-	2.00	-	-	-
р. Вовча (виключаючи рр. Мокрі Яли, Гайчур)	0.29	0.29	-	-	-	0.29	-	-	-	0.25	-	0.04	-	-

Рисунок 1.16 – Звіт по скидам стічних вод в межах басейну Дніпра

Ще одним важливим ресурсом Держводагентства, є портал «Моніторинг та екологічна оцінка водних ресурсів України» - сучасна інформаційна система, розроблена з урахуванням норм Європейського Союзу. У системі передбачено засоби для отриманнях різних аналітичних звітів для усіх річкових басейнів, за адміністративно-територіальним принципом та за водогосподарськими організаціями. Використання даної системи підвищує якість інформаційної підтримки та управління водними ресурсами.

На рисунку 1.17 зображено пункти спостереження державної системи моніторингу якості води у вікні системи «Моніторинг та екологічна оцінка водних ресурсів України».

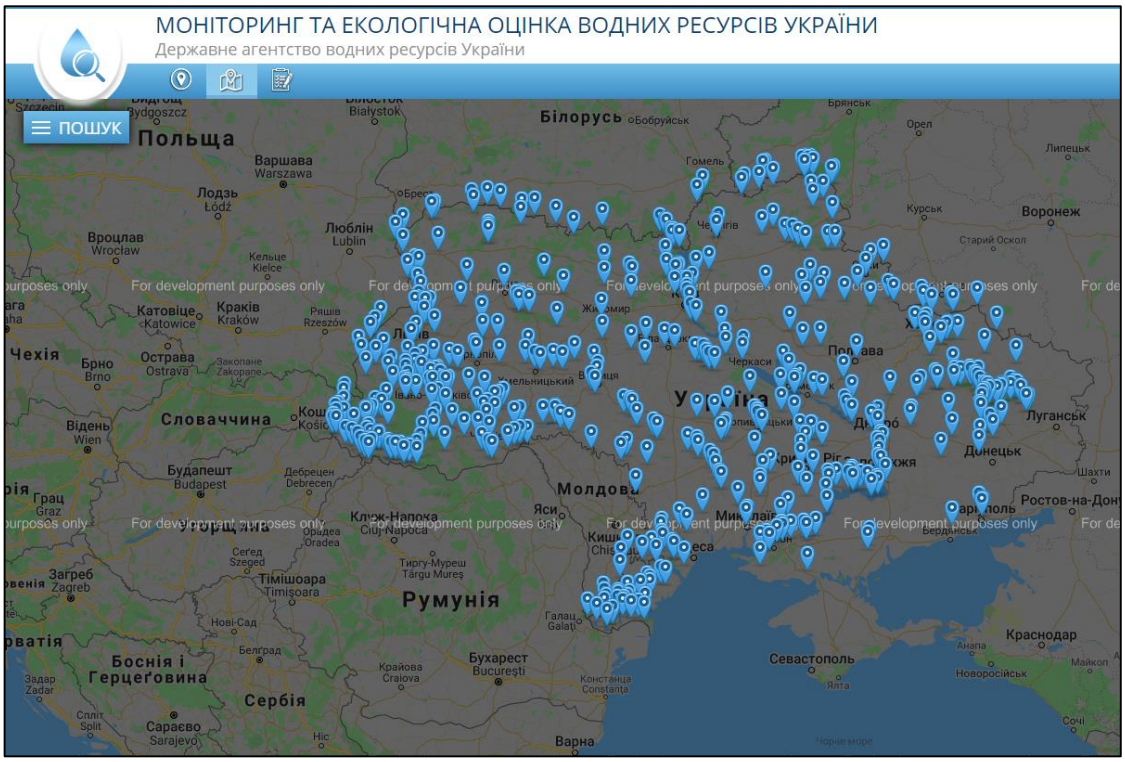


Рисунок 1.17 – Сторінка порталу моніторингу водних ресурсів України»

Для обробки цих даних та аналізу їх якості все частіше залучаються інформаційні технології, які дозволяють використати потужності комп'ютерних систем для більш детального опрацювання наявної інформації. Потужними засобами для подібного роду задач є машинне навчання та нейронні мережі.

Для використання подібних інструментів у більшості випадків вимагаються досить великі об'єми даних. Крім зазначених раніше статистичних даних водокористування та моніторингу також можуть залучатися картографічні та супутникові дані. Зображення обробляються нейронними мережами, що дозволяє сегментувати та класифікувати об'єкти на вказаній ділянці. Для отриманих даних визначаються найвагоміші атрибути та ознаки, які потім використовуються для створення моделі пошуку місць забруднення [12].

Головною проблемою використання подібних методів та підходів – це знайти джерела за історичними даними. Крім того, дані методи поки розвинені недостатньо.

1.3 Висновки

У розділі було розглянуто поточний стан та причини водного забруднення у світі та в Україні. Крім того, було проаналізовано основні методи та засоби контролю, аналізу та попередження погіршення стану якості води.

Враховуючи ту ситуацію, яка зараз склалася в України щодо забруднення води, дослідження, спрямовані на виявлення основних джерел забруднення з метою їх негайного регулювання, є особливо актуальними. Інфраструктура моніторингу забруднення води не надає достатньої кількості інформації про час та причини забруднень, що, у свою чергу, ускладнює контроль якості води та не дозволяє точно локалізувати проблему. Отже, необхідно розробити та реалізувати інформаційну технологію для пошуку джерел забруднень, яка дозволить усунити вищезазначені недоліки та проблеми.

2 РОЗРОБЛЕННЯ МАТЕМАТИЧНОГО ТА АЛГОРИТМІЧНОГО АПАРАТУ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET

2.1 Огляд вхідних даних

Вхідними даними для дослідження є відкриті дані Державного агентства водних ресурсів України [2, 3] на Єдиному державному веб-порталі відкритих даних, а також на веб-сайті «Моніторинг та екологічна оцінка водних ресурсів України». Інтерфейс ресурсу зображено на рисунку 2.1.

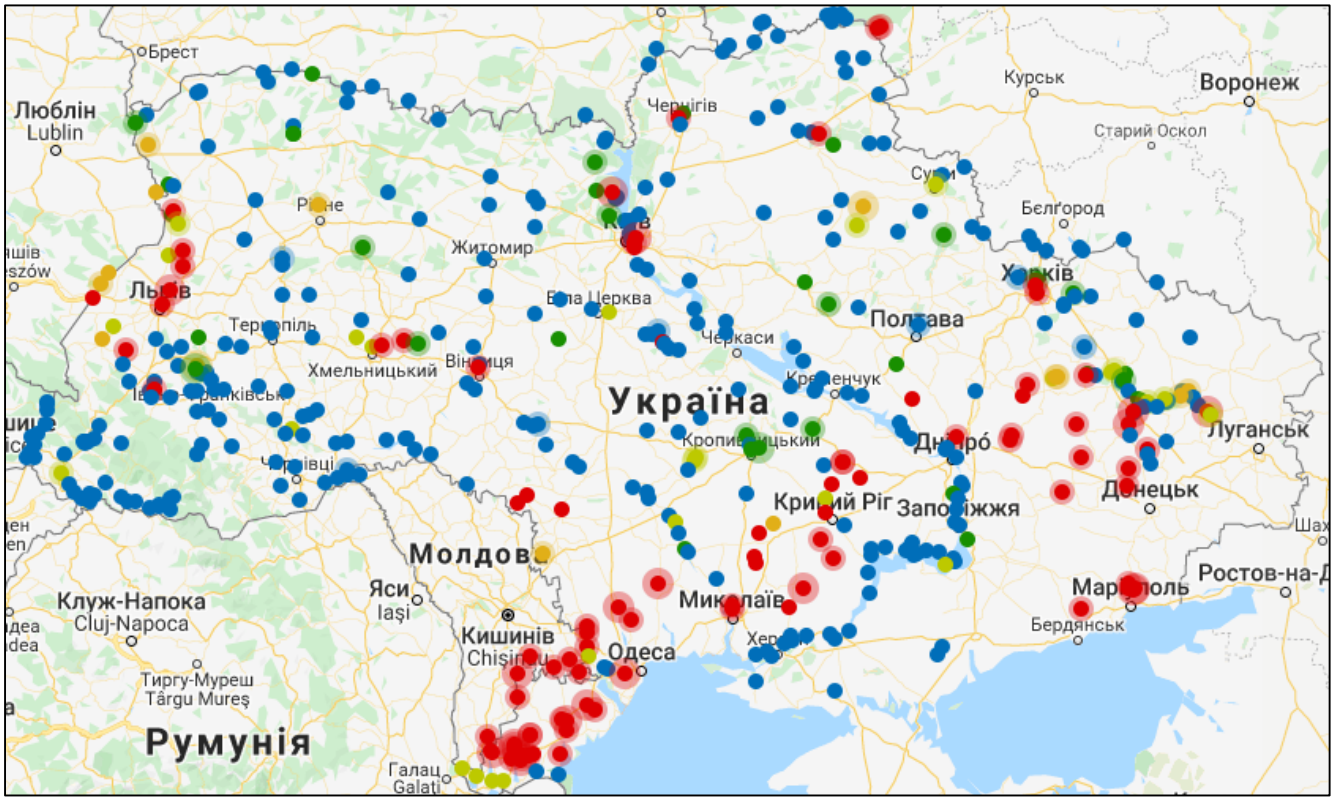


Рисунок 2.1 – Портал моніторингу водних ресурсів

Вони включають в себе інформації по постах спостереження за період з 1920 по 2020 роки. По кожному посту інформація представлена у значеннях наступних параметрів:

- азот загальний, мг/дм^3;
- біохімічне споживання кисню за 5 діб, мгО/дм^3;
- завислі (суспендовані) речовини, мг/дм^3;
- кисень розчинений, мгО2/дм^3;
- сульфат-іони, мг/дм^3;
- хлорид-іони, мг/дм^3;
- амоній-іони, мг/дм^3;
- нітрат-іони, мг/дм^3;
- нітрит-іони, мг/дм^3;
- фосфат-іони (поліфосфати), мг/дм^3.

Приклад вікна веб-системи зображено на рисунку 2.2.

Повна назва ПС: р. Південний Буг, 582 км, м. Вінниця, Сабарівське вдсх, питний в/з міста, вище міста
Район річкового басейну: Південний Буг
Назва головної лабораторії, що виконує ІЛВ: Лабораторія моніторингу вод та ґрунтів БУВР річки Південний Буг
Остання дата спостереження: 14.01.2020
Дані моніторингу

Показник	Фактичне значення	ГДК (ОБУВ)	Перевищення нормативу, раз
Азот загальний, мг/дм^3			
Біохімічне споживання кисню за 5 діб, мгО/дм^3	5,3	3	1,77
Завислі (суспендовані) речовини, мг/дм^3	8	15	Немає
Кисень розчинений, мгО2/дм^3	10,5	4	Немає
Сульфат-іони, мг/дм^3	34,9	100	Немає
Хлорид-іони, мг/дм^3	30,5	300	Немає
Амоній-іони, мг/дм^3	0,18	0,5	Немає
Нітрат-іони, мг/дм^3	2,3	40	Немає
Нітрит-іони, мг/дм^3	0,05	0,08	Немає
Фосфат-іони (поліфосфати), мг/дм^3	0,094		

Рисунок 2.2 – Значення моніторингу якості води на порталі «Моніторинг та екологічна оцінка водних ресурсів України»

Розглянемо дані виключно по річці Південний Буг. Від витоку річки до міста Вінниця розташовано 8 постів системи моніторингу: «Чорний Острів», «Хмельницький», «Копистин», «Меджибіж», «Щедрове», «Хмільник», «Гущинці (Калинівка)», «Вінниця». Дані пункти зображено на рисунку 2.3.

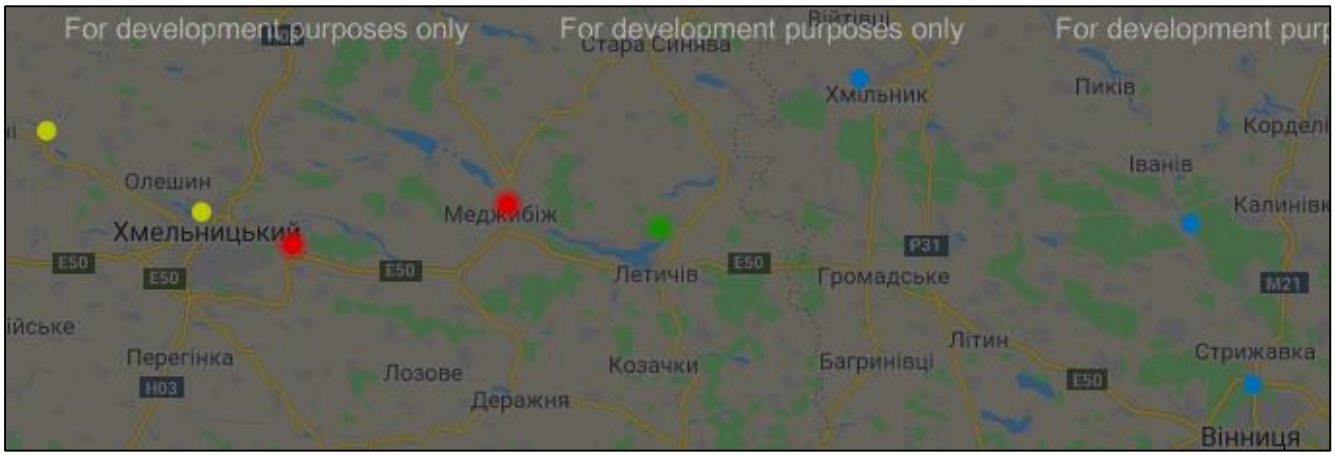


Рисунок 2.3 – Портал моніторингу водних ресурсів

Найбільші відмінності від нормованих значень спостерігаються по показниках концентрації азоту з перерахунку на нітрати NO_2, концентрація азоту амонійного NH_4 та біохімічне споживання кисню. По кожному з них сформуємо вибірку даних для кожної з зазначених вище станцій на проміжку 2002 – 2019 рр. Даний веб портал має функцію експорту, завдяки які можна завантажити дані для певного водного басейну за вказаний часовий проміжок. Вікно з параметрами експорту наведено на рисунку 2.4.

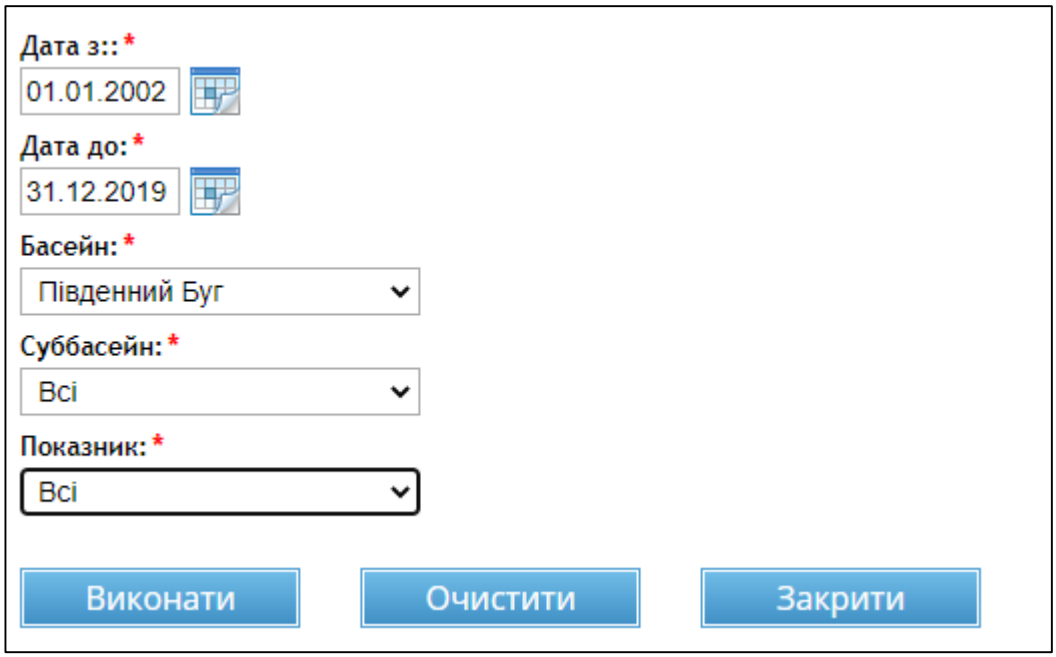
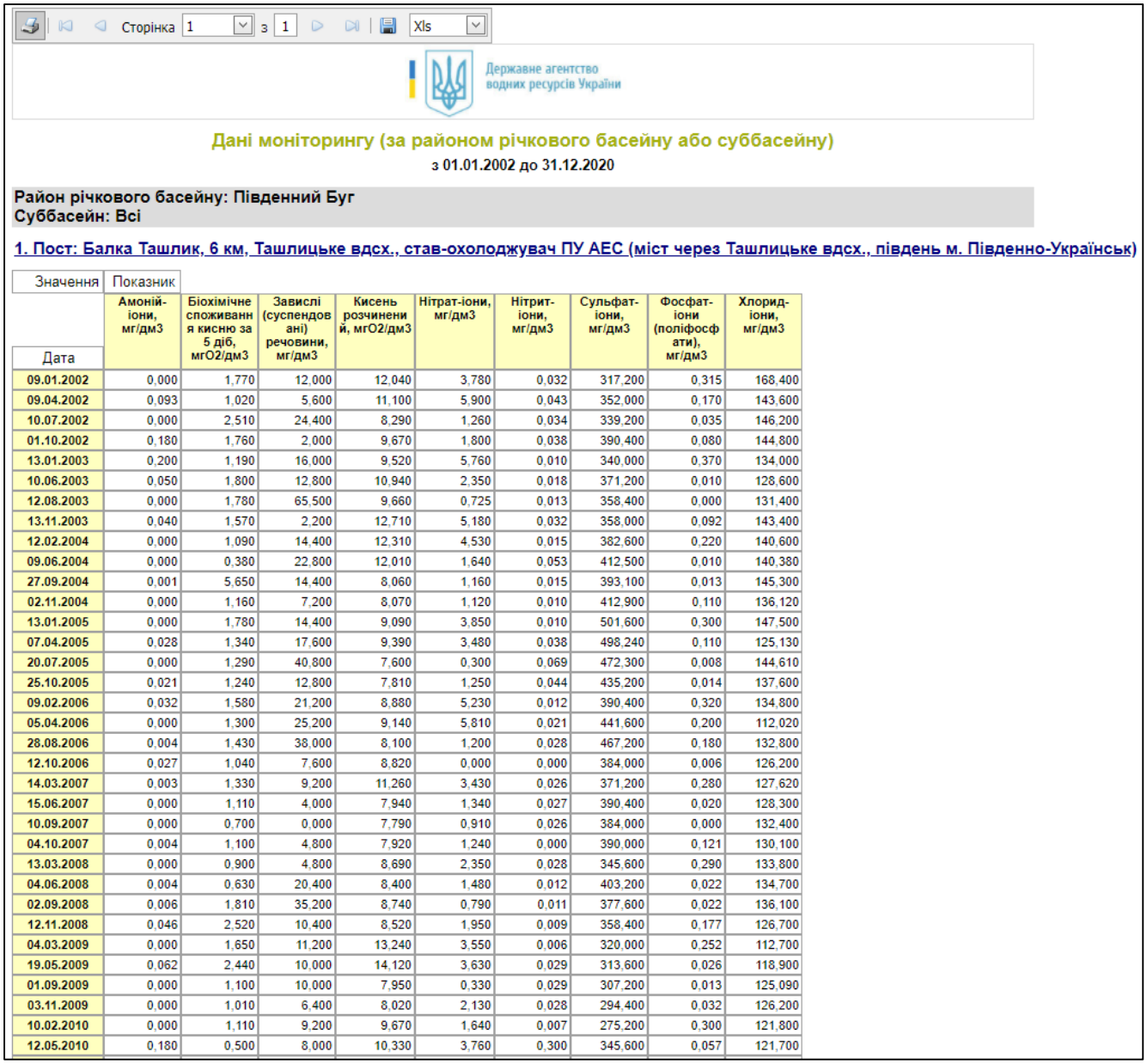


Рисунок 2.4 – Вікно експорту даних моніторингу

Результат експорту зображено на рисунку 2.5.



Сторінка 1 з 1 Xls

Державне агентство водних ресурсів України

Дані моніторингу (за районом річкового басейну або суббасейну)

з 01.01.2002 до 31.12.2020

Район річкового басейну: Південний Буг
Суббасейн: Всі

1. Пост: Балка Ташлик, 6 км, Ташлицьке вдсх., став-охолоджувач ПУ АЕС (міст через Ташлицьке вдсх., південь м. Південно-Український)

Значення / Показник / Дата	Амоній-іони, мг/дм3	Біохімічне споживання кисню за 5 діб, мгO2/дм3	Завислі (суспендовані) речовини, мг/дм3	Кисень розчинений, мгO2/дм3	Нітрат-іони, мг/дм3	Нітрит-іони, мг/дм3	Сульфат-іони, мг/дм3	Фосфат-іони (поліфосфати), мг/дм3	Хлорид-іони, мг/дм3
09.01.2002	0,000	1,770	12,000	12,040	3,780	0,032	317,200	0,315	168,400
09.04.2002	0,093	1,020	5,600	11,100	5,900	0,043	352,000	0,170	143,600
10.07.2002	0,000	2,510	24,400	8,290	1,260	0,034	339,200	0,035	146,200
01.10.2002	0,180	1,760	2,000	9,670	1,800	0,038	390,400	0,080	144,800
13.01.2003	0,200	1,190	16,000	9,520	5,760	0,010	340,000	0,370	134,000
10.06.2003	0,050	1,800	12,800	10,940	2,350	0,018	371,200	0,010	128,600
12.08.2003	0,000	1,780	65,500	9,660	0,725	0,013	358,400	0,000	131,400
13.11.2003	0,040	1,570	2,200	12,710	5,180	0,032	358,000	0,092	143,400
12.02.2004	0,000	1,090	14,400	12,310	4,530	0,015	382,600	0,220	140,600
09.06.2004	0,000	0,380	22,800	12,010	1,640	0,053	412,500	0,010	140,380
27.09.2004	0,001	5,650	14,400	8,060	1,160	0,015	393,100	0,013	145,300
02.11.2004	0,000	1,160	7,200	8,070	1,120	0,010	412,900	0,110	136,120
13.01.2005	0,000	1,780	14,400	9,090	3,850	0,010	501,600	0,300	147,500
07.04.2005	0,028	1,340	17,600	9,390	3,480	0,038	498,240	0,110	125,130
20.07.2005	0,000	1,290	40,800	7,600	0,300	0,069	472,300	0,008	144,610
25.10.2005	0,021	1,240	12,800	7,810	1,250	0,044	435,200	0,014	137,600
09.02.2006	0,032	1,580	21,200	8,880	5,230	0,012	390,400	0,320	134,800
05.04.2006	0,000	1,300	25,200	9,140	5,810	0,021	441,600	0,200	112,020
28.08.2006	0,004	1,430	38,000	8,100	1,200	0,028	467,200	0,180	132,800
12.10.2006	0,027	1,040	7,600	8,820	0,000	0,000	384,000	0,006	126,200
14.03.2007	0,003	1,330	9,200	11,260	3,430	0,026	371,200	0,280	127,620
15.06.2007	0,000	1,110	4,000	7,940	1,340	0,027	390,400	0,020	128,300
10.09.2007	0,000	0,700	0,000	7,790	0,910	0,026	384,000	0,000	132,400
04.10.2007	0,004	1,100	4,800	7,920	1,240	0,000	390,000	0,121	130,100
13.03.2008	0,000	0,900	4,800	8,690	2,350	0,028	345,600	0,290	133,800
04.06.2008	0,004	0,630	20,400	8,400	1,480	0,012	403,200	0,022	134,700
02.09.2008	0,006	1,810	35,200	8,740	0,790	0,011	377,600	0,022	136,100
12.11.2008	0,046	2,520	10,400	8,520	1,950	0,009	358,400	0,177	126,700
04.03.2009	0,000	1,650	11,200	13,240	3,550	0,006	320,000	0,252	112,700
19.05.2009	0,062	2,440	10,000	14,120	3,630	0,029	313,600	0,026	118,900
01.09.2009	0,000	1,100	10,000	7,950	0,330	0,029	307,200	0,013	125,090
03.11.2009	0,000	1,010	6,400	8,020	2,130	0,028	294,400	0,032	126,200
10.02.2010	0,000	1,110	9,200	9,670	1,640	0,007	275,200	0,300	121,800
12.05.2010	0,180	0,500	8,000	10,330	3,760	0,300	345,600	0,057	121,700

Рисунок 2.5 – Експортовані дані моніторингу для басейну річки Південний Буг

Як було сказано раніше, для дослідження будуть братися тільки 8 пунктів від витоку річки до міста Вінниця та для показників нітрату NO_2, концентрації азоту амонійного NH_4 та біохімічне споживання кисню. Крім того, для зручності подальшої роботи згрупуємо дані усіх необхідних станцій по обраних показниках якості води. Оскільки було вирішено працювати з трьома показниками, отримуємо три файли з даними відповідного показника якості води.

Приклад сформованої вибірки для показників NO_2 зображено на рисунку 2.6.

	A	B	C	D	E	F	G	H	I	J	K
1		NO2	NO2	NO2	NO2	NO2	NO2	NO2	NO2		
2	ID_Statior	27	28	29	30	31	32	33	34		
3	Date_Month										
4	2002-1	0.055	0.085	0.08							
5	2002-10						0.0415	0.0745	0.155		
6	2002-11	0.06	0.075	0.066667							
7	2002-12				0.137333	0.39					
8	2002-3	0.03	0.05				0.81	0.04	0.03		
9	2002-4							0.05	0.069		
10	2002-5	0.18	0.153333	0.193333	0.02	0.66					
11	2002-6	0.13	0.12	0.13							
12	2002-7	0.04	0.11	0.08							
13	2002-8	0.14	0.18	0.06							
14	2002-9	0.14	0.5	0.52	0.11	0.04	0.19				
15	2003-1	0.12	0.08								
16	2003-11	0.13		0.09	0.08	0.1					
17	2003-12	0.25	0.12	0.07	0.09	0.14					
18	2003-2	0.14	0.145	0.095	0.1		1				
19	2003-3	0.166667	0.12	0.246667		0.42		0.11	0.09		
20	2003-4	0.14	0.19	0.115							
21	2003-5	0.16	0.08	0.05			1	0.09	0.1		
22	2003-9	0.09	0.08	0.07	0.08	0.69	0.9	0.08			
23	2004-1	0.09	0.12	0.06							
24	2004-10	0.15	0.109333	0.22			0.56				
25	2004-12				0.04	0.57					
26	2004-2						0.07	0.07	0.064		
27	2004-4	0.04	0.05	0.14	0.08	0.22					
28	2004-5	0.06	0.08	0.08	0.13	0.48	0.01	0.36	0.01		
29	2004-7	0.07	0.0865	0.18				1.05			
30	2004-8	0.075	0.29	0.055				0.82	0.36		
31	2004-9	0.035	0.03	0.04	0.02	0.33	2.16	0	0.05		
32	2005-1	0.14	0.08	0.08	0.96	0.032					
33	2005-10	0.15	0.14	0.17			0.28	0.071			
34	2005-11	0.08	0.06	0.04							
35	2005-12	0.105	0.065	0.07							
36	2005-2	0.1	0.09	0.11	0.091	11.12	0.21	0.044	0.48		
37	2005-3	0.13	0.2	0.4							
38	2005-4	0.08	0.21	0.07				0.06	0.036		

Рисунок 2.6 –Значення показника NO_2 для станцій вздовж р. Південний Буг

Проаналізувавши кожну з вибірок, можна помітити, що усім даними притаманна випадкова періодичність. Для прикладу, порівняємо дані NO_2 для станцій «Вінниця» та «Хмельницький». Діаграма значень для кожної станції наведено на рисунку 2.7 та 2.8 відповідно.

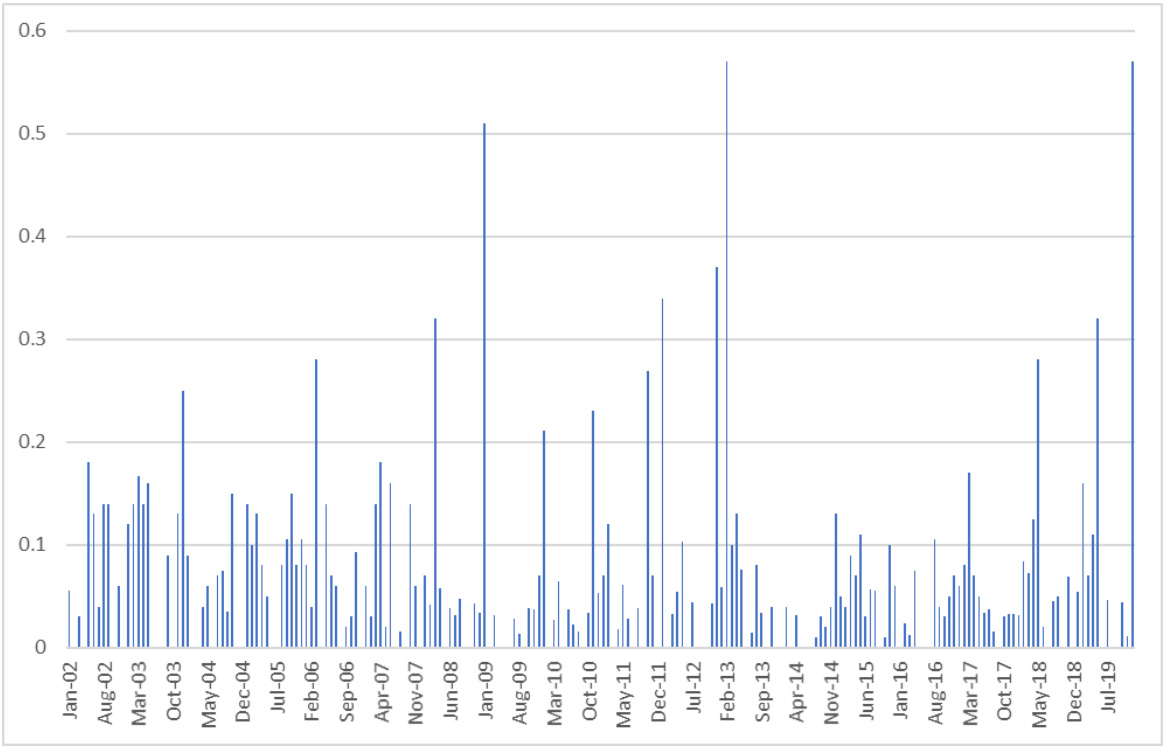


Рисунок 2.7 –Значення показника NO_2 для станцій «Вінниця»

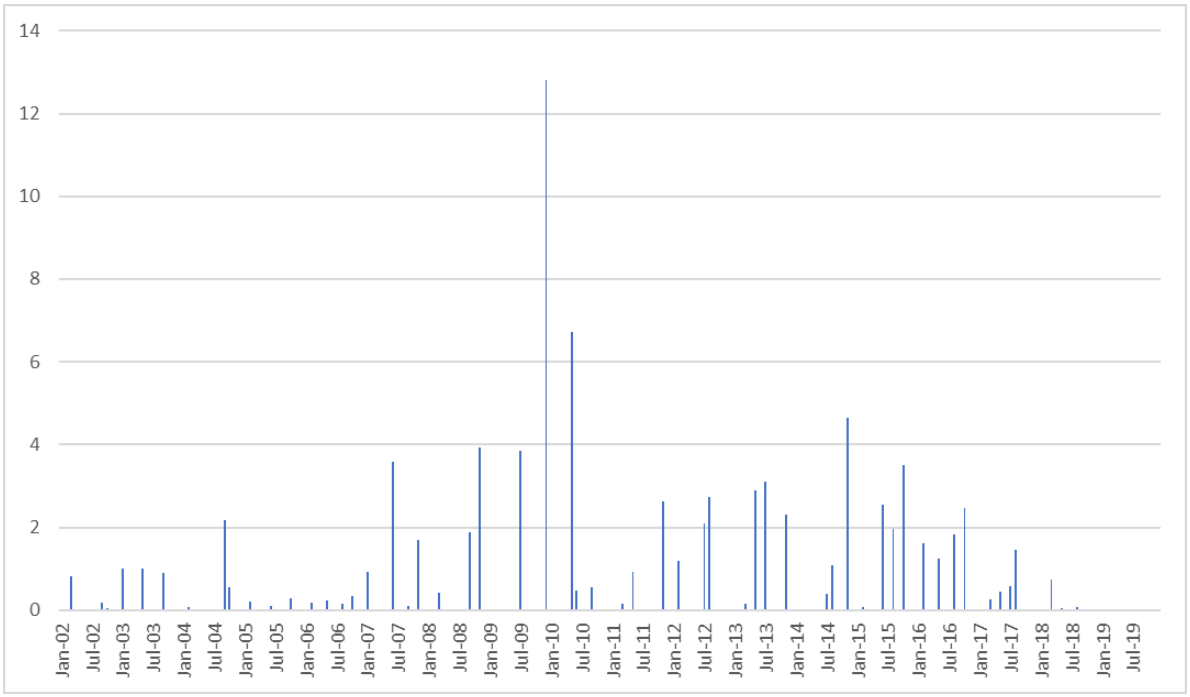


Рисунок 2.8 –Значення показника NO_2 для станцій «Хмельницький»

Для багатьох дат дані зовсім відсутні, а наявність певної періодичності в місяць чи квартал говорить про те, що виміри могли робитися у випадковий день чи місяць відповідно протягом цього періоду.

2.2 Опис моделі розв’язання проблеми

Проведений аналіз даних в підрозділі 2.1 показує, що наявним даним моніторингу якості води притаманна змінна періодичність та часткова відсутність даних, що значно ускладнює процес визначення реальних джерел забруднення води. Пропонується застосувати можливості машинного навчання для аналізу та локалізації причин та джерел підвищеного забруднення річки.

Машинне навчання – це підрозділ штучного інтелекту, що дає інформаційній системі можливості для навчання та автоматичного самонавчання на основі досвіду, тобто з кожним наступним набором даних модель стає розумнішою і більш точною для виконання поставленої задачі [13, 14]. При цьому, людина не приймає прямої участі у процесі навчання. Усі алгоритми машинного навчання можна поділити на дві категорії: контрольовані (або навчання з учителем) або неконтрольовані (навчання без учителя).

Види машинного навчання зображені на рисунку 2.9.

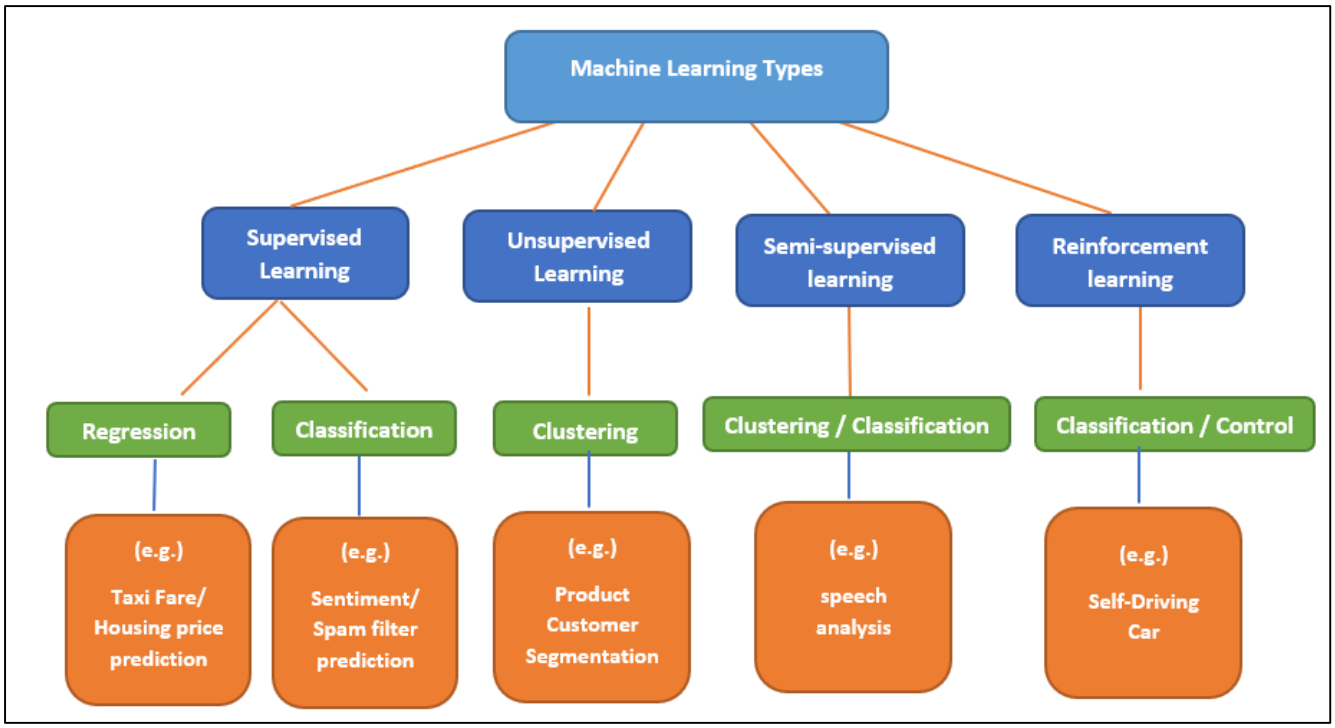


Рисунок 2.9 – Види машинного навчання

Для проведення детальнішого та більш ефективного пошуку можливих джерел забруднень доцільно застосовувати часові ряди [15, 16]. Прогнозування часових рядів вважається одним із найбільш застосовуваних наукових методів для роботи з даними, який знаходить застосування у багатьох галузях.

Часовий ряд відноситься до послідовності спостережень, зібраних через певні інтервали часу: кожен день, місяць, квартал або рік. Аналіз часових рядів передбачає розробку моделей, які використовуються для опису часового ряду та розуміння ознак, які лежать в основі даних. Приклад часового ряду зображено на рисунку 2.10.

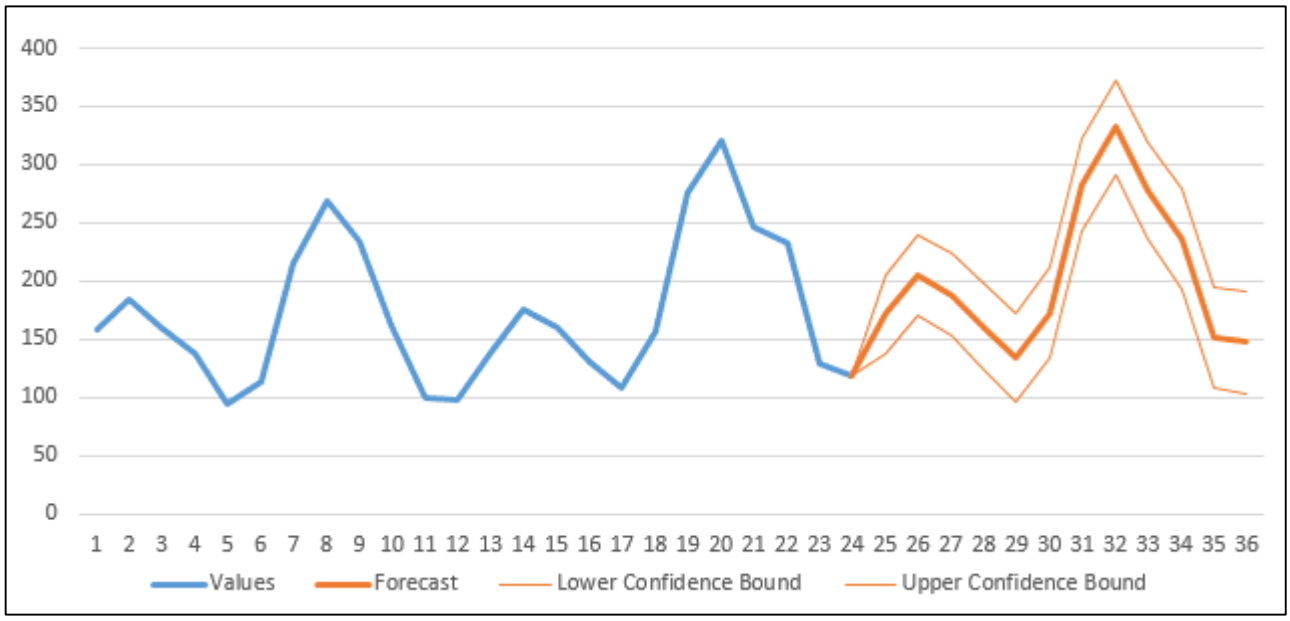


Рисунок 2.10 – Часовий ряд

Основними компонентами часових рядів є [17]:

- тенденція – описує поведінку збільшення/зменшення часових рядів і часто представлене лінійною моделлю;
- сезонність – певний повторюваний шаблон або цикл поведінки;
- нерегулярність / шум – несистематичний аспект часового ряду, що відхиляється від загальноприйнятих значень моделі;
- циклічність – повторювані зміни у часових рядах та розміщення цих змін в циклі.

Серед багатьох методів, спрямованих на досягнення точності та мінімізації помилок при роботі з часовими рядами, існує кілька досить класичних та сучасних методів машинного навчання, які довели свою точність та обчислювальну актуальність. Найпопулярнішими моделями [17] для прогнозування часових рядів є:

1) Наївна модель – передбачає, що наявність будь якої ознаки в класі не пов'язано з наявністю будь якої іншої ознаки. Навіть за умови залежності цих ознак одна від одної, вони мають незалежний внесок у значення ймовірності. Теорема Байеса дозволяє розрахувати апостеріорну ймовірність $P(c|x)$ на основі $P(c)$, $P(x)$ та $P(x|c)$.

$$P(c|x) = \frac{P(x|c)*P(c)}{P(x)}, \tag{2.1}$$

де $P(c|x)$ – апостеріорна ймовірність даного класу с при даному значені х;
$P(c)$ – апріорна ймовірність даного класу;
$P(x|c)$ – ймовірність даного значення ознаки при даному класі;
$P(x)$ – апріорна ймовірність даного значення ознаки.

2) Експоненціальна модель згладжування – метод згладжування даних часових рядів за допомогою функції експоненціального вікна. Експоненціальні функції використовуються для присвоєння експоненціального зменшуваних ваг з часом. Послідовність даних представлена $\{x_t\}$, що починається в $t = 0$, а результат експоненціального алгоритму згладжування записується як $\{s_t\}$, що розглядається як найкраща оцінка того, яким буде наступне значення. Коли послідовність спостережень починається в момент часу $t = 0$, найпростіша форма експоненціального згладжування надається формулами $s_0 = x_0$ та $s_t = ax_t + (1 - a)s_{t-1}$, де a це фактор згладжування і $0 < a < 1$.

3) ARIMA/SARIMA. ARIMA це комбінація авторегресивного підходу (AR) та підходу ковзного середнього (MA). Модель ARIMA містить параметри, які враховують сезон і тенденцію. Крім того, вона дозволяють застосовувати авторегресивні та ковзні середні терміни для обробки автокореляції, яка є в даних.

SARIMA це сезонна ARIMA, яка вона розширює застосування звичайної моделі, додаючи лінійну комбінацію сезонних минулих значень та / або помилок прогнозу.

З аналізу даних, проведеного в розділі 2.1, видно, що пости спостереження за якістю води в більшості випадків розташовані досить віддалено від метеопостів та, що робить досить складним їх коректне співставлення та використання моделі множинної регресії та тих, які основані на ній. До того ж, через хаотичну періодичність розташування постів, використання класичних моделей часових рядів типу авторегресії та проінтегрованого ковзного середнього (АРПКС) унеможливлюється.

Можливим засобом вирішення цієї задачі є використання моделі з пакету програм Prophet, розробленого компанією Facebook. Prophet – це технологія прогнозування даних часових рядів, реалізована для мов програмування Python та R, заснована на адитивній моделі, де нелінійні тенденції відповідають річній, тижневій та щоденній сезонності, плюс святкові ефекти [18]. Найкраще це працює з часовими рядами, що мають сильний сезонний ефект та кількома сезонами історичних даних. Prophet стійкий до відсутніх даних та змін у тренді, і, як правило, добре обробляє викиди. Головними перевагами даної моделі є [4, 18]:

- підтримка роботи з періодичними і неперіодичними рядами;
- можливість коректної роботи навіть за умови великої кількості відсутніх (або пропущених) даних;
- можливість виконувати прогнозування для вказаного інтервалу;
- можливість побудови зон невизначеності;
- використання як лінійної, так і логістичної моделі тренду, які можна обмежити точками зміни тренду;
- можливість додавати складові сезонності (як правило, рядами Фур’є заданого порядку);
- застосовування набору даних типу «holidays», що дає змогу вказувати значення і дати відомих подій, які тим чи іншим чином могли впливати на значення у відповідній точці.

На рисунку 2.11 зображено приклад прогнозу, який дозволяє будувати модель Prophet.

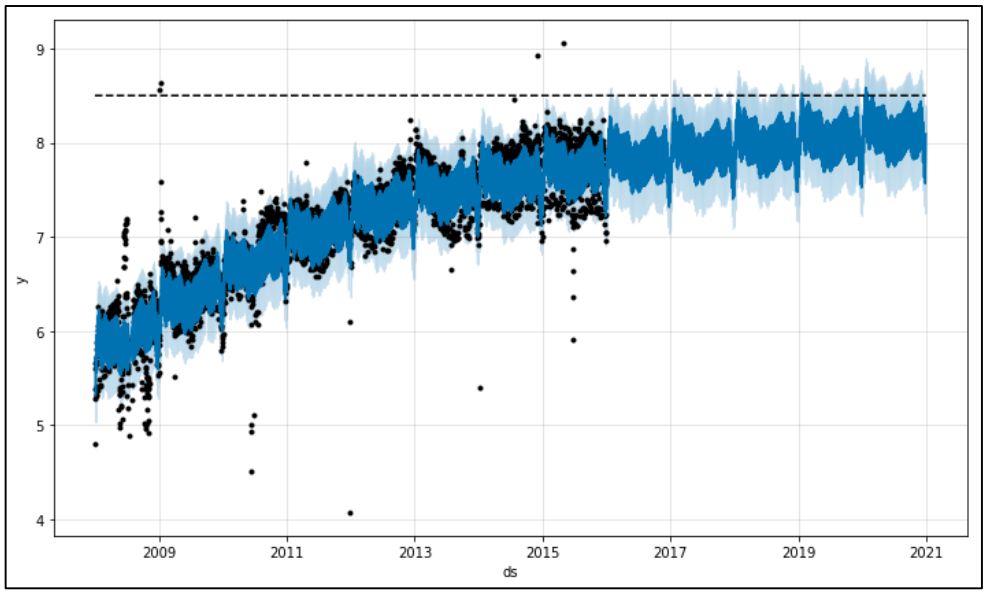


Рисунок 2.11 – Графік передбачення проведеного за допомогою Prophet

Крім того, як було сказано раніше, Prophet дозволяє переглянути передбачення, розбите на окремі сезонні компоненти: тренд, річну та тижневу сезонність. Приклад даного графіку зображено на рисунку 2.12.

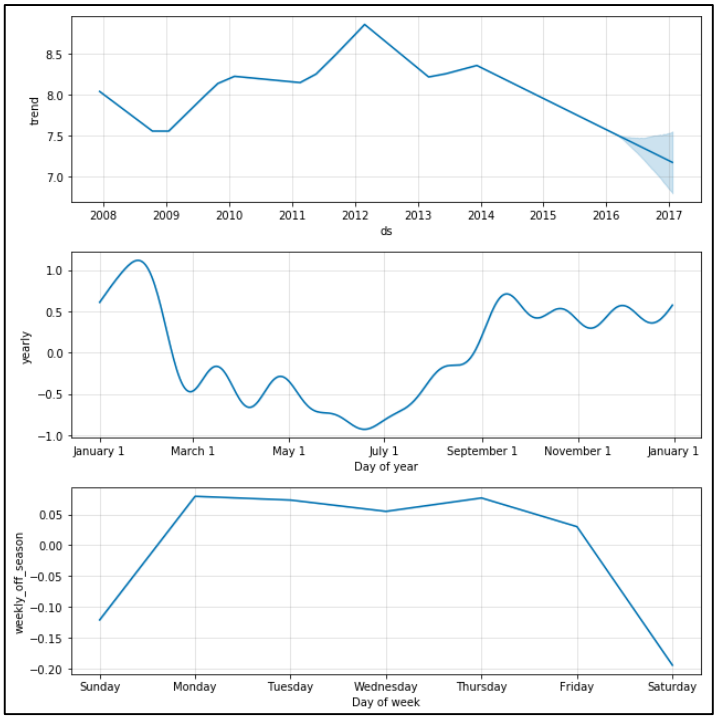


Рисунок 2.12 – Графік окремих сезонних компонентів передбачення

2.3 Розроблення математичного та алгоритмічного апарату

Для розв’язання задачі необхідно побудувати модель Prophet, використовуючи лінійну регресію, для усіх обраних показників забруднення, точки зміни тренду буде встановлено на перший день року. Для кожного випадку початку забруднення необхідно знайти нахили тренду та обрати прирости з найвагомішими значеннями, але для лише для таких випадків, де виявлений приріст не зменшувався, а продовжував збільшуватися. Тобто, знайти часові точки з яких почалося забруднення та проводжується досі, інші прирости при цьому відкидаються (прирівнюються до нуля). Далі необхідно розглянути місяці, які потрапляють у діапазон між точками зміни. Виявлені імпульси масштабують у діапазон [0, 1] та додаються по кожному посту спостереження та показниках, отриманих на них. Даний процес необхідно повторювати, і при цьому змінювати налаштування моделі, поки не буде досягнуто задовільного значення похибки для контрольних даних та виявлених значення забруднюючих речовин на певних постах та за певний часовий проміжок. Крім того, як було зазначено раніше, модель Prophet містить параметр типу «holidays», який дозволяє вказати час відомих аномалій. Необхідно, наприклад, вказати дати відомих аварій або певних скидів води, що дозволить більш точно проаналізувати наявні значення моніторингу і уникнути хибних результатів.

Модель Prophet повертає початковий тренд k_0 ряду до першої точки зміни d_0 і m приростів $\Delta k_i, i = \overline{1, m}$ тренду між наступними m точками аж до кінця тренувального інтервалу даних. За необхідності, потім по цій моделі можуть передбачатись дані на наступні періоди. При цьому, дати $d_i, i = \overline{1, m}$ розташування цих точок змін вже не зазнають. На рисунку 2.13 показано прирости Δk_i тренду між точками зміни d_i.

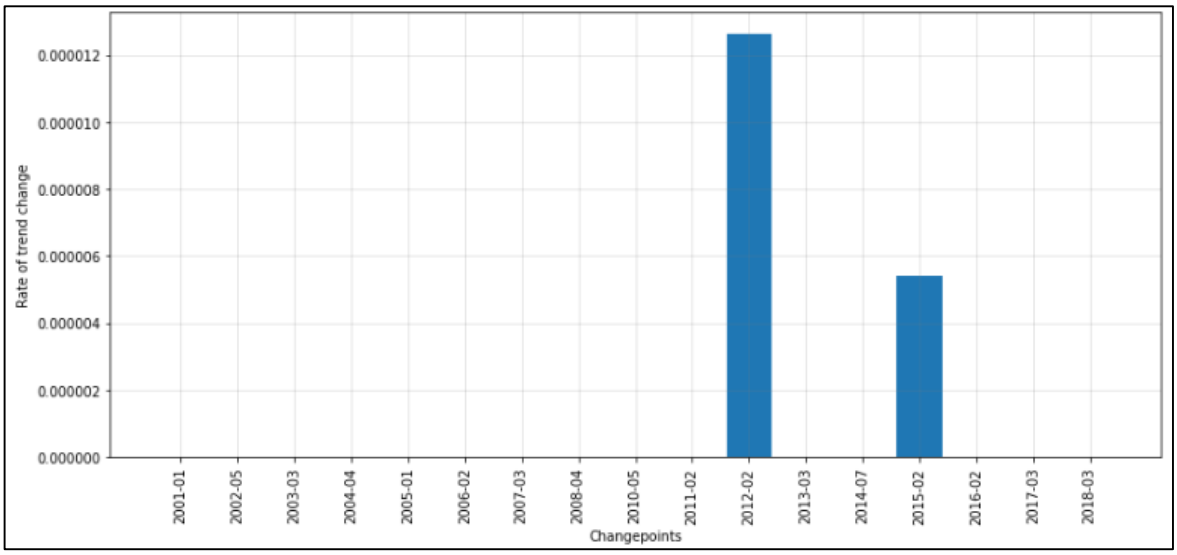


Рисунок 2.13 – Значення приростів Δk_i

Головним є знаходження моментів приростів, тому початковий тренд значення не має і слід аналізувати тільки прирости тренду відносно нього. Результатом роботи моделі Prophet є:

$$\widehat{y_i} = (k + \Delta k_i)d_i, i = \overline{1, m}\,, \tag{2.2}$$

де $\widehat{y_i}$ – стандартизоване значення тренду в i-й точці зміни d_i, тобто масштабоване між мінімальним і максимальним значеннями та поділене на середньоквадратичне відхилення. Графік результату зображено на рисунку 2.14.

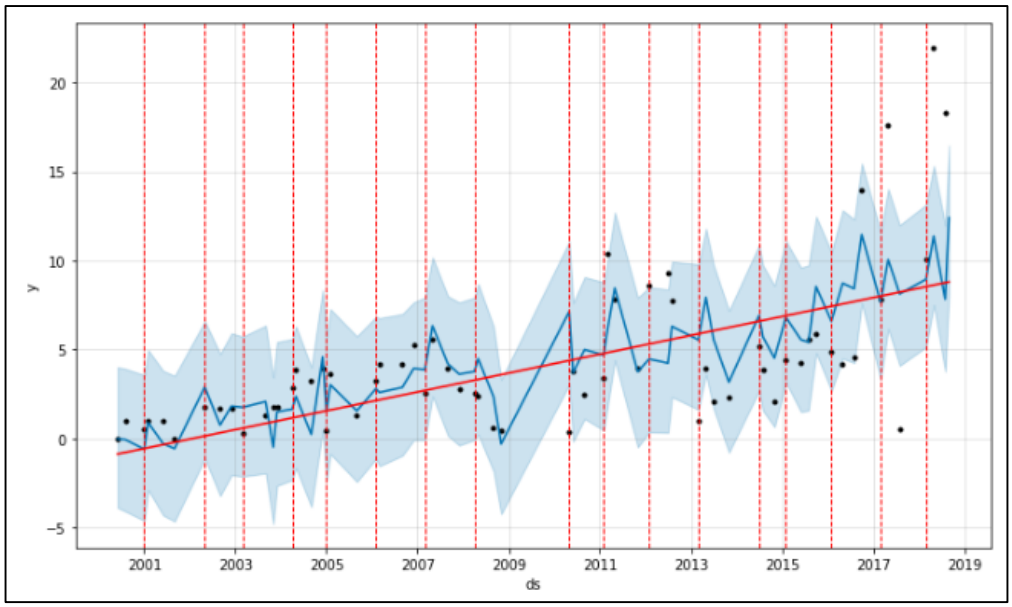


Рисунок 2.14 – Результат роботи моделі Prophet

Для виявлення ділянок, де прирости трендів є суттєвими (більшими заданого значення), необхідно застосувати інтегрування. Тому, знайдемо S суми приростів тренду з накопиченням:

$$S = \sum_{i=1}^{m} \Delta k_i. \tag{2.3}$$

Значення сум приростів тренду зображено на рисунку 2.15.

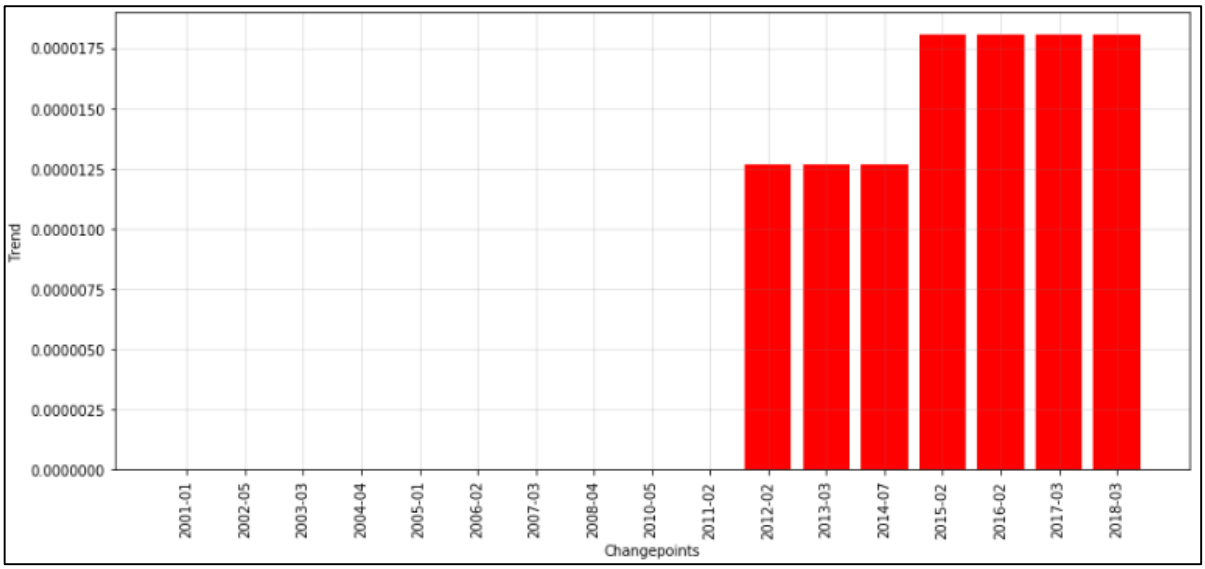


Рисунок 2.15 – Значення сум приростів тренду з накопиченням

Але також потрібно враховувати, що ряд приростів Δk_i тренду може бути знакозмінним. У той же час, потрібно шукати тільки ті імпульси, які привносили не епізодичний вплив, а продовжували накопичення концентрації. Для рішення цієї задачі повторно проведемо інтегрування:

$$J = \sum_{j=i}^{m} S_i = \sum_{q=j}^{m} \sum_{j=i}^{m} \Delta k_{qj}. \tag{2.4}$$

Після цього необхідно відшукати ті імпульси, для яких функція J продовжувала монотонно зростати.

Отже, пропонуємо розв’язувати задачу за таким алгоритмом:

1. Обираємо показники забруднення, які мають систематичне відхилення від гранично допустимих показників на постах спостереження;

2. Для усунення можливих похибок у розрахунках, видаляємо усі некоректні або зайві дані;

3. Для покращення точності, додаємо набір даних з датами відомих аномалій на відповідних локаціях;

4. Вхідні дані розділяємо на тренувальні та тестові. Тренувальний набір даних дозволить проаналізувати та визначити параметри моделі, а тренувальні – для вибору оптимальної моделі серед знайдених раніше;

5. Обираємо перше січня як значення початкового наближення зміни тренду та встановлюємо його для кожного обраного раніше показника забруднення води;

6. Виконуємо налаштування Prophet та створюємо різні моделі для кожного з показників забруднення води. Для також враховуємо, сформовані на кроці 3, дати відомих аномалій («holidays») та значення сезонності;

7. Використовуючи дані, які були сформовані та відібрані на кроці 4, необхідно знайти оптимальні параметри налаштування моделі, яка була обрана на кроці 6, з метою отримання якомога меншого значення похибки;

8. Нехай показники якості води задані циклом $i = 1 \dots n$. На кожній його ітерації для усіх ділянок річки, заданих циклом $j = 1 \dots m$ та часових точок changepoints, заданих циклом $k = 1 \dots l$, рахуємо бальну y_{ijk} оцінку приросту тренду x цього показника, яка більшим значенням дає значно більшу вагу, наприклад, шляхом взяття квадрату від їх значень:

$$y_{ijk} = \left(\frac{x_{ijk} - \min\limits_{i} x_{ijk}}{\max\limits_{i} x_{ijk} - \min\limits_{i} x_{ijk}} \right)^2. \quad (2.5)$$

9. Для кожної k-ї дати кожної ділянки між j-ми постами спостережень слід додати усі ці бальні оцінки та ранжувати їх за зменшенням для кожної ділянки річки:

$$J_{jk} = \sum_i y_{ijk}. \quad (2.6)$$

10. Серед отриманих результатів необхідно знайти ті частини річки та дати в яких значення J_{jk} було найбільше. При цьому варто також враховувати наявні звіти усіх водокористувачів на відповідних ділянках про їх використання води, скиди стічних води, результатами еко-інспекційних перевірок, даними супутникових спостережень, даними громадського екологічного моніторингу та іншою інформацією [19-21], здійснити пошук, коли у визначені дати на визначених ділянках могли з'явитись джерела скидання вод, та рекомендувати органам Держекоінспекції та Держводагентства провести більш ретельну перевірку виявлених фактів

Блок-схема алгоритму запропонованої інформаційної технології наведена на рисунку 2.16.

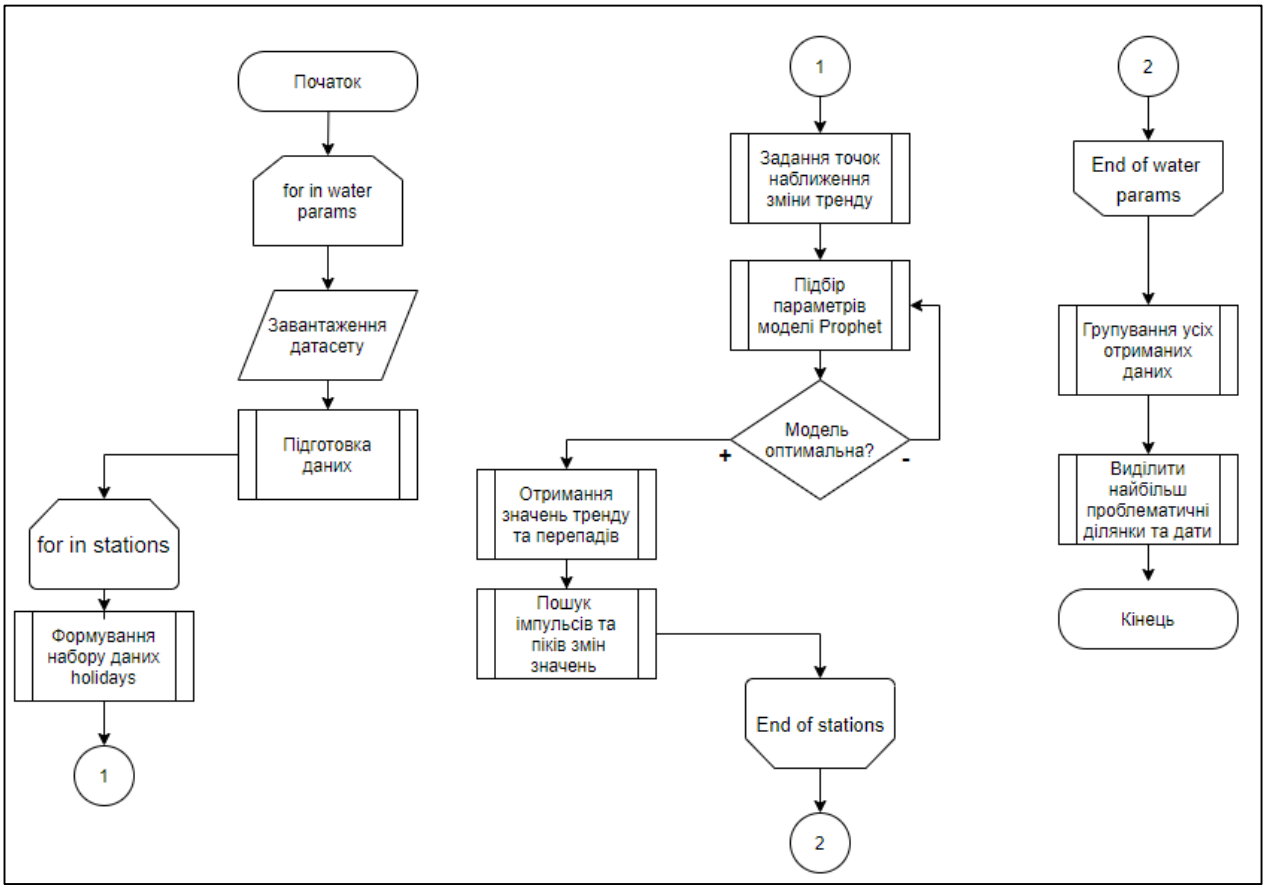


Рисунок 2.16 – Блок-схема алгоритму інформаційної технології

2.4 Висновки

В даному розділі проведено аналітичний огляд даних спостережень якості води за 1920-2020 роки та обрано найвагоміші показники для подальшої роботи.

Розроблено математичних та алгоритмічний апарат інформаційної технології пошуку джерел забруднення річки та описано модель для прогнозування часових рядів – Prophet, яка буде використовуватися для реалізації запропонованої інформаційної технології.

3 РОЗРОБЛЕННЯ ПРОГРАМНОГО ІНСТРУМЕНТАРІЮ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET

3.1 Вибір мови програмування

Позиції основних мов для Data Science (та Machine Learning) давно та впевнено посідають Python та R [22]. Обидві мови чудово підходять для роботи з статистикою, що суттєво ускладнює вибір між ними. Проведемо аналіз та порівняння даних для визначення тієї, яка більше підходить для розробки інформаційної технології пошуку можливих джерел забруднення.

Python – інтерпретована мова програмування високого рівня та загального призначення [22, 23]. Її мовні конструкції та об’єктно-орієнтований підхід мають на меті допомогти програмістам писати чіткий логічний код для малих та великих проектів. Синтаксис ядра Python досить простий та мінімалістичний. Але, незважаючи на це, стандартна бібліотека нараховує великий перелік корисних функцій. Основними його перевагами є:

- простий, але досить виразний синтаксис. Маючи, якісь базові знання з Java або C#, буде досить просто та приємно від того, як легко можна створити свою першу програму;
- великий перелік бібліотек і мова йде не тільки про бібліотеки для машинного навчання;
- висока культура документації. Офіційна документація досить зрозуміла та детальна і переважна більшість бібліотек також притримуються цього.

Серед основних його недоліків можна зазначити те, що Python не вважається дуже швидкою мовою програмування, проте за умови оптимальної реалізації алгоритмів, його швидкодія досить задовільна.

Основними бібліотеками для Machine Learning для мови Python є:

- pandas - бібліотека для обробки та аналізу даних. Містить у собі структури даних та методи для роботи з таблицями і рядами. Бібліотека підтримує основні та найпопулярніші способи зберігання даних: CSV, Excel, SQL, HTML і багато іншого;
- numpy – бібліотека для роботи з багатовимірними масивами та матрицями. Крім того, вона надає багато математичних функцій та обчислювальних алгоритмів для роботи з цими структурами, які до того ж досить швидкі;
- Matplotlib - бібліотека візуалізації даних двовимірної та тривимірною графікою. Підтримуються широкий перелік графіків та діаграм: графіки (line plot), діаграми розкиду (scatter plot), стовпчасті діаграми (bar chart) і гістограми (histogram), кругові діаграми (pie chart) та багато інших;
- scikit-learn – бібліотека, яка надає широкий вибір алгоритмів навчання з учителем і без. Вона спеціалізується на алгоритмах класифікації (прогноз ознаки, безліч допустимих значень якого обмежена), регресії (прогноз ознаки з речовими значеннями), а також кластеризації (розбиття даних по класах).

R – мова програмування та середовище вільного програмного забезпечення для статистичних обчислень та графіки [22, 24]. Мова R широко використовується серед статистиків для розробки статистичного програмного забезпечення та аналізу даних. R пропонує широкий спектр статистичних (лінійне та нелінійне моделювання, аналіз часових рядів, класифікація, кластеризація) та графічних методів, і є дуже розширюваним. Перевагами даної мови є:

- потужна екосистема машинного навчання та велика кількість бібліотек статистичних методів;
- незвичайний синтаксис, який добре заточений під потреби статистики;
- вбудована підтримка векторних обчислень.

Як було сказано вище, синтаксис мови є досить незвичним, що одночасно робить його напевно головним недоліком. Програміст, який має досвід роботи з іншими мовами програмування, буде некомфортно себе почувати, використовуючи її.

Основними бібліотеками для машинного навчання є:

- Dplyr – бібліотека для роботи з даними;
- Ggplot2 і Esquisse – потужні бібліотеки для побудови графіків;
- Shiny – бібліотека для створення веб-додатків з візуалізацією досліджень;
- Caret, randomForest, Mlr та багато інших – десятки бібліотек з методами машинного навчання.

Як видно з аналізу, обидві мови справедливо являються найпопулярнішими мовами для машинного навчання. Порівнюючи їх, варто зазначити, що R в основному використовується для аналізу даних та статистики, тоді як Python забезпечує більш загальний підхід. R важко вивчити на початку, тоді як Python є лінійним і плавним для вивчення. І R, і Python можуть обробляти величезний розмір бази даних.

Враховуючи все це, було вирішено, що для розробки інформаційної технології пошуку можливих джерел підвищеного забруднення річки буде використовуватися мова програмування Python.

3.2 Реалізація допоміжних методів

Як було зазначено в підрозділі 2.2, для вирішення поставленої задачі буде використовуватися модель з пакету Prophet. Для можливості роботи з цією бібліотекою необхідно додати її до нашого файлу. Код додавання Prophet зображено на рисунку 3.1.

```
# models
from fbprophet import Prophet
from fbprophet.plot import plot_plotly
from fbprophet.plot import add_changepoints_to_plot
```

Рисунок 3.1 – Підключення бібліотеки Prophet

Для роботи моделі Prophet необхідно задати їй точки зміни тренду (в термінології бібліотеки – changepoint). Згідно алгоритму, сформованого в підрозділі 2.4, ці точки необхідно встановити на перші дати кожного року. Спочатку додаємо до нашого файлу ще дві бібліотеки, які необхідні для маніпуляції з даними: pandas і numpy. Програмний код для додавання цих бібліотек зображено на рисунку 3.2.

```
import numpy as np
import pandas as pd
```

Рисунок 3.2 – Підключення бібліотеки pandas і numpy

Тепер реалізуємо метод для генерування точок зміни тренду. Метод повинен отримувати як параметр набір даних та повернути список точок, встановлених на перший місяць кожного наявного в наборі даних року. Реалізація даного метода зображена на рисунку 3.3.

```
def set_changepoints(df):
    # Calc changepoints for "ds" in dataframe df with sorted by data with interval = year (xxxx-x,...)

    # Determine range of years
    df_dates = df.copy()
    df_dates['ds_dt'] = pd.to_datetime(df_dates['ds'], format='%Y-%m', errors='ignore')
    df_dates['month'] = pd.DatetimeIndex(df_dates['ds_dt']).month
    df_dates['year'] = pd.DatetimeIndex(df_dates['ds_dt']).year
    start_month = df_dates.month[0]
    start_year = df_dates.year[0]
    num_last = len(df_dates.ds)-1
    end_month = df_dates.month[num_last]
    end_year = df_dates.year[num_last]
    diff = (df_dates.ds_dt[num_last]-df_dates.ds_dt[0])
    print('different =', diff.days, 'days = ', diff.days // 30, 'months')
    start_range = start_year + 1 if start_month < 10 else start_year + 2
    changepoints = []

    # Set changepoints at intervals of a year
    end_range = end_year + 1 if end_month > 2 else end_year
    print('Year, number month in each year, sum of data in each year:')
    for year in range(start_range, end_range):
        df_i = df_dates[df_dates['year'] == year].sort_values(by=['month']).reset_index(drop=True)
        if len(df_i) > 0:
            print(year, len(df_i), round(df_i['y'].sum(),1))
            year_i = df_i.loc[0,'year']
            month_i = df_i.loc[0,'month']
            changepoints.append(f'{year_i}-{month_i}')

    return changepoints
```

Рисунок 3.3 – Метод для генерування списку точок зміни тренду

Приклад результату виконання методу зображено на рисунку 3.4

```
[
    '1995-1', '1996-1', '1997-1', '1998-1', '1999-1',
    '2000-1', '2001-2', '2002-1', '2003-1', '2004-1',
    '2005-1', '2006-1', '2007-1', '2008-1', '2009-1',
    '2010-1', '2011-1', '2012-2', '2013-1', '2014-2',
    '2015-1', '2016-1', '2017-1', '2018-1', '2019-1',
]
```

Рисунок 3.4 – Приклад результату роботи методу set_changepoints

Наступним необхідно реалізувати метод, який буде створювати модель Prophet для набору даних та для заданих точок зміни тренду. Метод повинен приймати зазначені дані як параметри та повертати новий об’єкт Prophet, яка підганяється під заданий набір даних. Даний метод зображено на рисунку 3.5.

```
def m_fbprophet(df, changepoints, n_changepoints):
    # Return object of fbprophet with given parameters and calculated trend

    num = len(df) // 6
    if (changepoints == []) and (num < n_changepoints):
        n_changepoints = num

    m = Prophet(
                daily_seasonality = False,
                yearly_seasonality = 4,
                weekly_seasonality = False,
                changepoints = changepoints,
                n_changepoints = n_changepoints
                )

    # calculation linear trend by FB Prophet
    m.fit(df, algorithm='Newton')

    return m
```

Рисунок 3.5 – Метод для ініціалізації моделі Prophet

Наступним кроком є реалізація метода, який буде безпосередньо виконувати передбачення за допомогою створеної моделі Prophet. Також, для деталізації даних необхідно, щоб метод відмалював графіки зробленого прогнозу та окремих

показників тренду та сезонності. Для цього можна використати спеціальні методи об’єкту Prophet: plot та plot_components. Програмна реалізація методі зображена на рисунку 3.6.

```
def fbprophet_plot(df, title, changepoints, n_changepoints, time_prediction, output):
    # Calculation fbprophet prediction on 1 year for df with changepoints as the initial approximatio
    # and if output = True then output data and draw plots


    # Initialization FB Prophet object and calculation trend
    model = m_fbprophet(df, changepoints, n_changepoints)

    future = model.make_future_dataframe(periods=time_prediction, freq = "M")
    forecast = model.predict(future)

    if output:
        print('fbprophet: ' + title)
        print('model.delta', model.params['delta'])

        fig_forecast = model.plot(forecast)
        for changepoint in model.changepoints:
            plt.axvline(changepoint, ls='--', lw=1, color='red')
        a = add_changepoints_to_plot(fig_forecast.gca(), model, forecast)
        fig_components = model.plot_components(forecast)

    return forecast[['ds', 'trend', 'trend_lower', 'trend_upper']], model.params['delta'][0], model.changepoints
```

Рисунок 3.6 – Метод для передбачення за допомогою об’єкту Prophet

Приклад результату виклику методу зображено на рисунку 3.7.

```
model.delta [[-3.09729528e-10 -3.35826771e-10  1.22718070e-10  1.24794262e-11
  -2.60453199e-10  2.61515776e-10  1.48921777e-10  1.67742882e-05
   7.64990283e-11  1.02133055e-10 -1.19439392e-10  3.42438568e-10
   4.49595984e-10  5.80899524e-06 -1.00671677e-10 -2.99734859e-10
   2.44806102e-06  2.28980788e-10  2.57829864e-10 -1.48519541e-10
   3.42687066e-10  1.25811042e-10  1.67621691e-10 -1.62183544e-10
  -3.33104990e-10]]
```

Рисунок 3.7 – Результат виклику методу для передбачення

Графік передбачення, згенерований реалізованим методом, зображено на рисунку 3.8, а на рисунку 3.9 наведено приклад графіку сезонності.

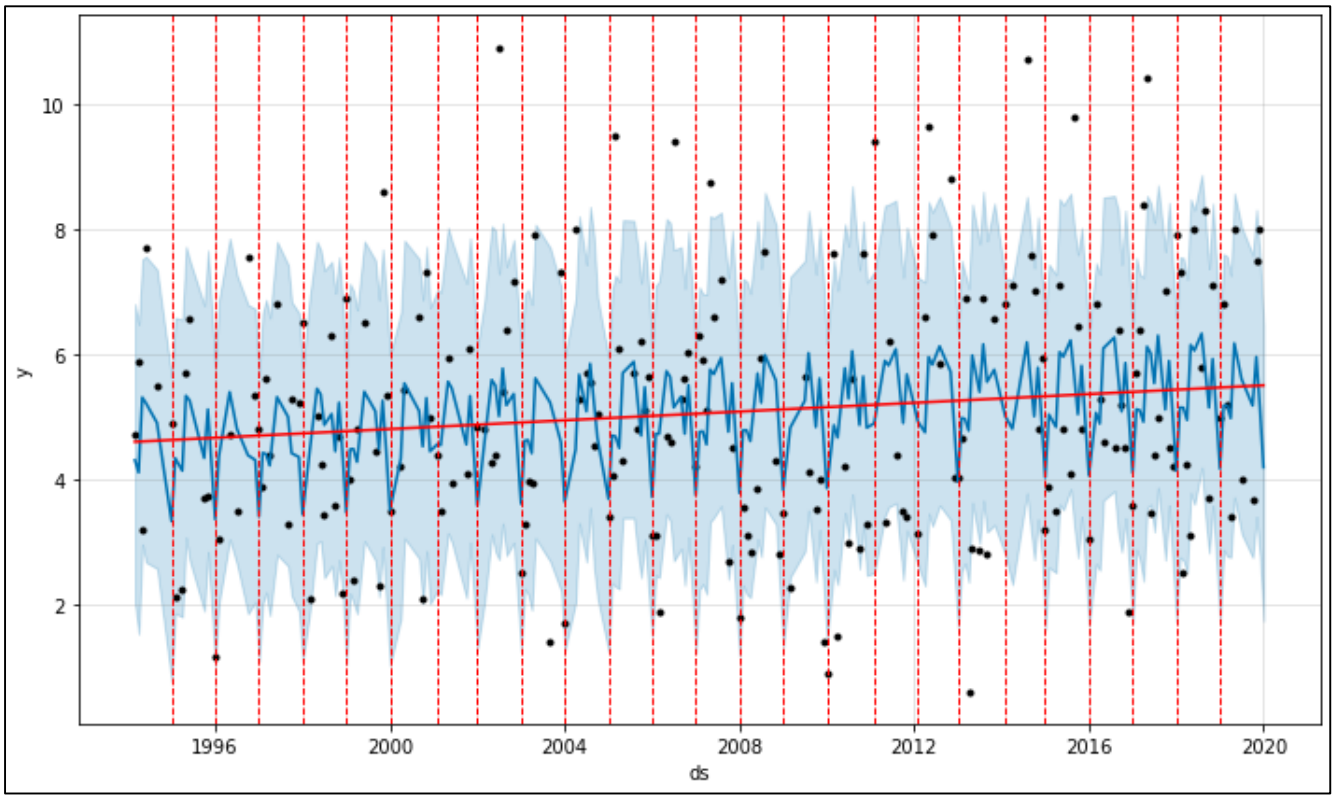


Рисунок 3.8 – Графік передбачення, виконаний об’єктом Prophet

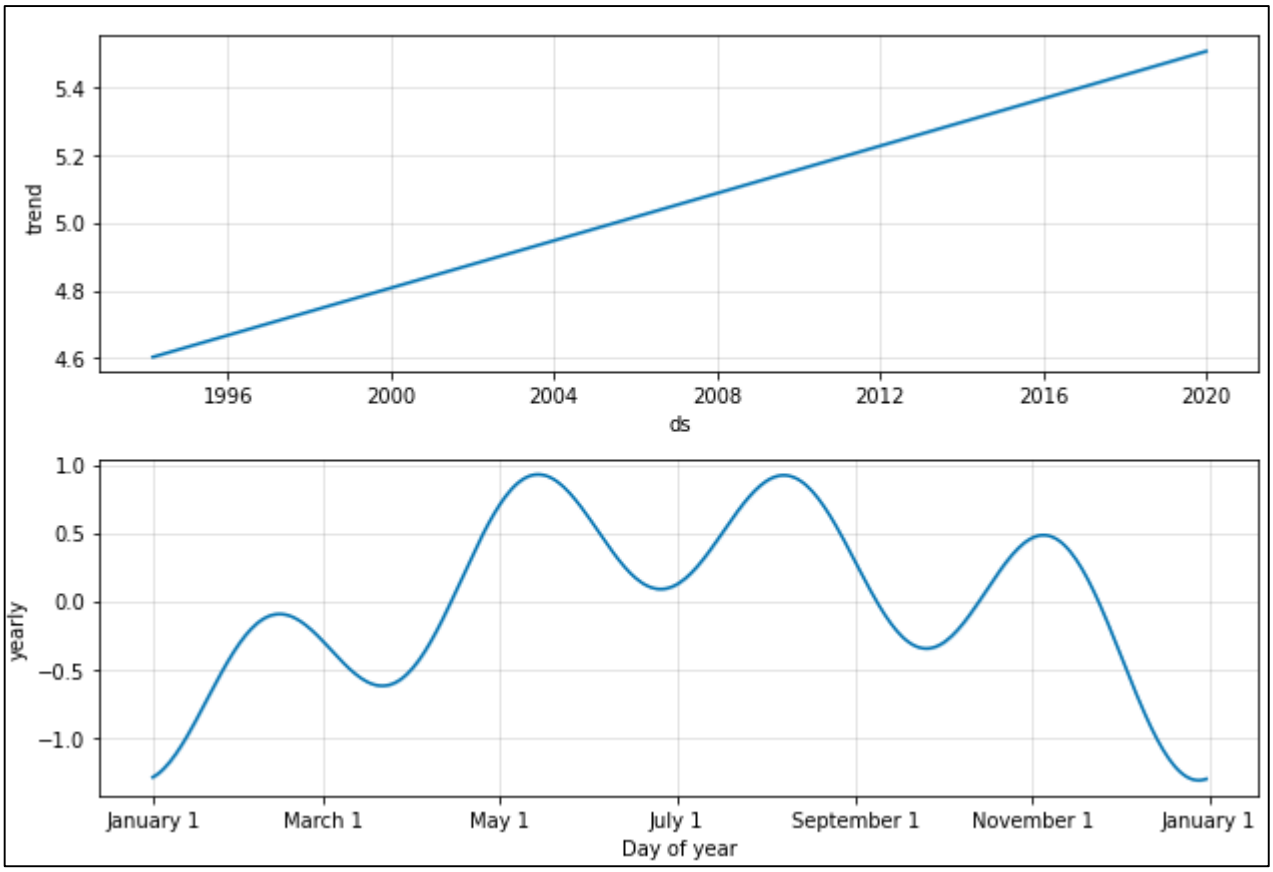


Рисунок 3.9 – Показники сезонності

Також необхідно реалізувати метод, який буде малювати графіки дельти зробленого передбачення. Крім того, необхідно побудувати графік тренду, знайшовши похідні від них та графік інтегралу цих значень. Код методу зображено на рисунку 3.10.

```
def deltas_plot(data, deltas, changepoints, title):
    # Draw plot with trend deltas for changepoints

    # Delta
    fig, ax = plt.subplots(figsize=(12,6))
    ax.set_title(title, fontsize=20)
    ax.bar(range(len(changepoints)), deltas)
    ax.grid(True, which='both', c='gray', ls='-', lw=1, alpha=0.2)
    ax.set_ylabel('Rate of trend change')
    ax.set_xlabel('Changepoints')
    ax.set_xticks(range(len(changepoints)))
    ax.set_xticklabels(changepoints, rotation='vertical')
    fig.tight_layout()
    plt.show()

    # Trend
    fig, ax = plt.subplots(figsize=(12,6))
    ax.set_title(title, fontsize=20)
    ax.bar(range(len(changepoints)), np.cumsum(deltas), color='r')
    ax.grid(True, which='both', c='gray', ls='-', lw=1, alpha=0.2)
    ax.set_ylabel('Trend')
    ax.set_xlabel('Changepoints')
    ax.set_xticks(range(len(changepoints)))
    ax.set_xticklabels(changepoints, rotation='vertical')
    fig.tight_layout()
    plt.show()

    # Integral
    fig, ax = plt.subplots(figsize=(12,6))
    ax.set_title(title, fontsize=20)
    ax.bar(range(len(changepoints)), np.cumsum(np.cumsum(deltas)), color='g')
    ax.grid(True, which='both', c='gray', ls='-', lw=1, alpha=0.2)
    ax.set_ylabel('Integral of parameter')
    ax.set_xlabel('Changepoints')
    ax.set_xticks(range(len(changepoints)))
    ax.set_xticklabels(changepoints, rotation='vertical')
    fig.tight_layout()
    plt.show()
```

Рисунок 3.10 – Метод для побудови графіків змін передбачення

Графіки, побудовані реалізованим методом, зображено на рисунках 3.11 – 3.13.

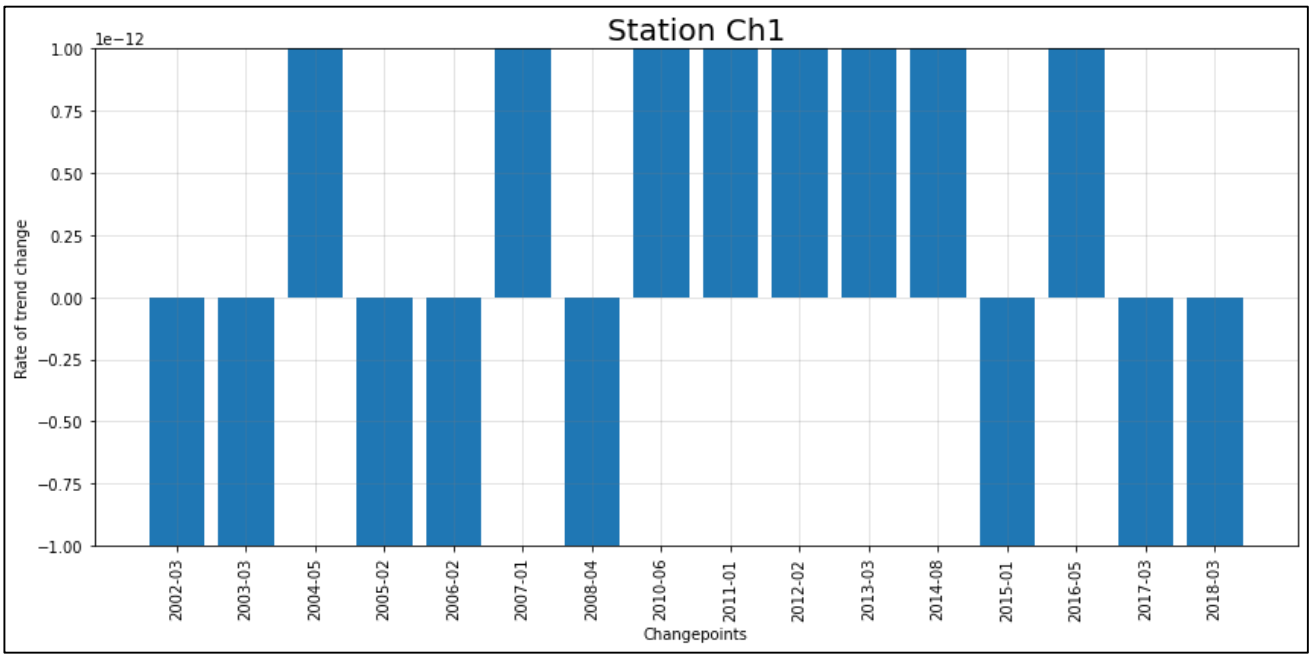


Рисунок 3.11 – Графік дельти зробленого передбачення

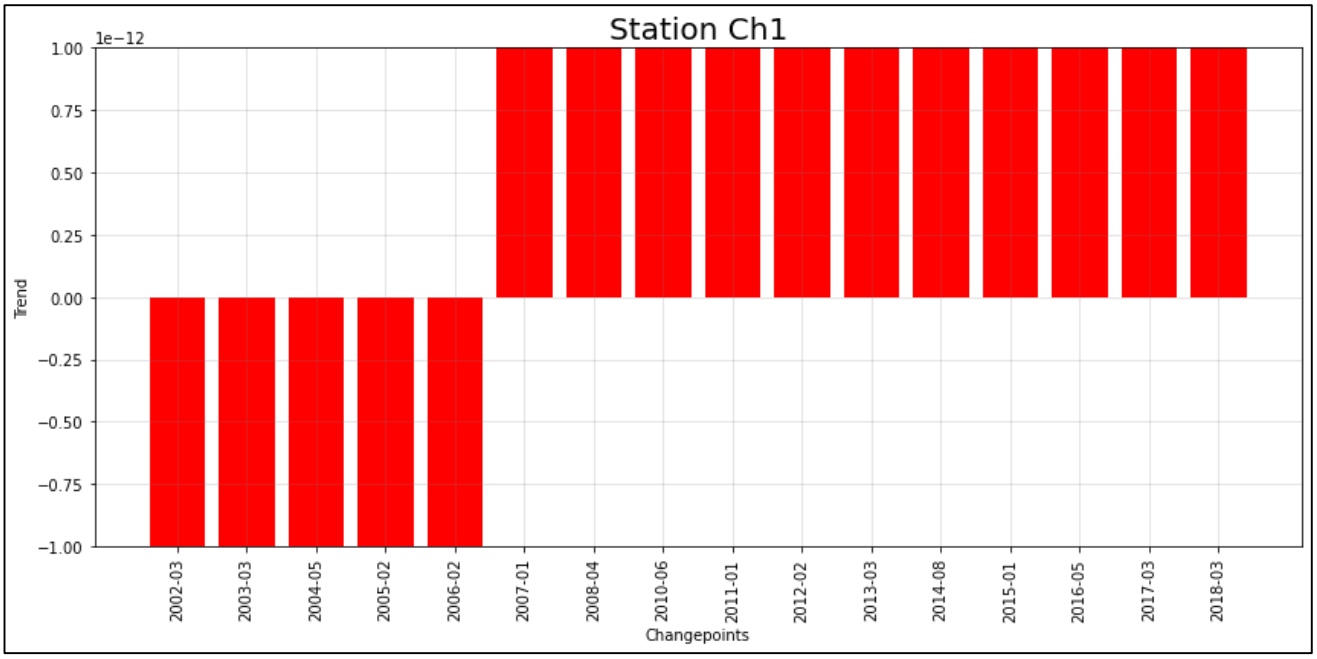


Рисунок 3.12 – Графік тренду зробленого передбачення

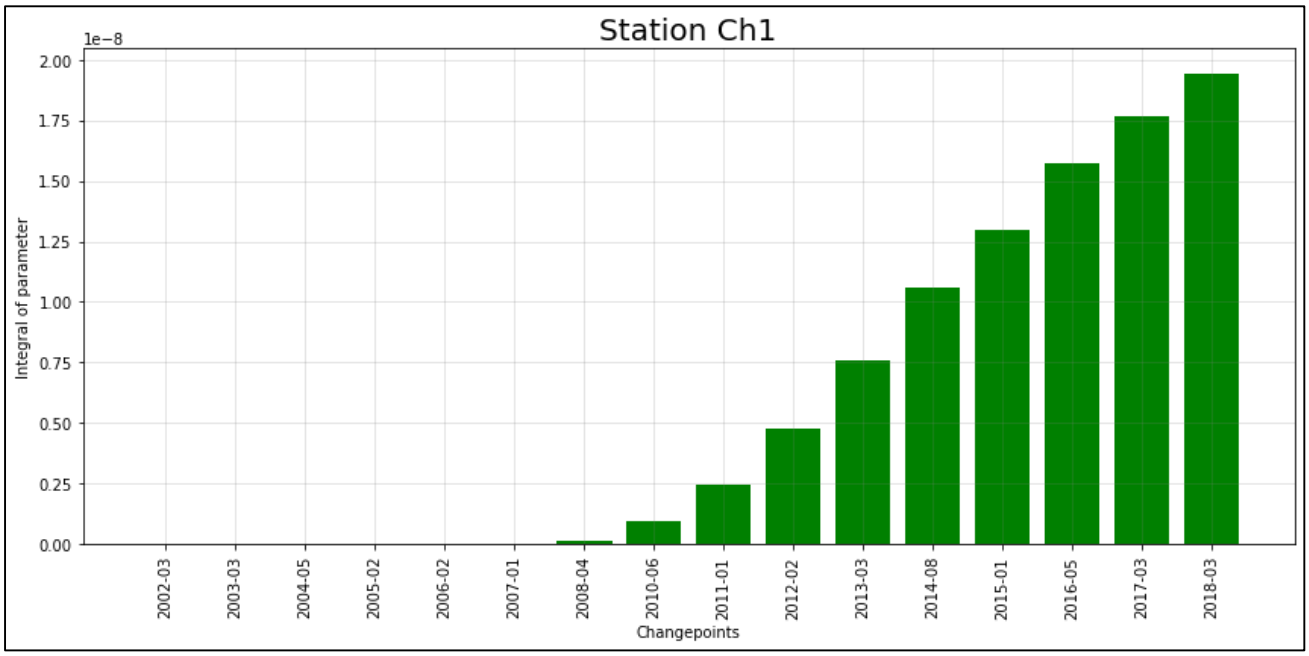


Рисунок 3.13 – Графік інтегралу значень зробленого передбачення

Одним з найголовніших методів є метод для знаходження моментів приростів трендів. Для цього необхідно інтегрування, тобто знайти суми приростів тренду з накопиченням. Також необхідно знайти саме ті прирости, вплив яких не зник, тобто ті імпульси після яких концентрація проводжувала зріст. Спочатку створимо невелику допоміжну функцію для знаходження монотонного зростання. Її код зображено на рисунку 3.14.

```
def monotonically_increasing(l):
    return all(x < y for x, y in zip(l, l[1:]))

def get_increasing_index(data):
    result = -1
    for index in range(len(data)):
        is_increasing = monotonically_increasing(data[index:])
        if (is_increasing):
            result = index
            print('nonzero start from -',result, 'len -',len(data))
            #print(data)
            break

    return result
```

Рисунок 3.14 – Метод знаходження монотонного приросту

Реалізуємо метод знаходження імпульсів, його код зображено на рисунку 3.15.

```
def impact_add(df, changepoints, deltas, col):
    # Add deltas to df
    # col - name of a new column in df

    # Calculation integral of trend
    rank = np.cumsum(np.cumsum(deltas))

    # Set zeros before increasing
    index = get_increasing_index(rank)
    rank_increasing = np.zeros(len(rank))
    if(index > -1):
        rank_increasing = np.concatenate((np.zeros(index), rank[index:]), axis=None)
    #rank = (rank - min(rank))/(max(rank)-min(rank))
    rank_increasing = rank_increasing/max(rank_increasing)

    df[col] = 0
    changepoints_list = changepoints.tolist()

    # Find peaks in deltas
    impact_biggest = list(locate(deltas, lambda x: x > 0.05*max(deltas)))

    # Find impulses
    impulse = np.zeros(len(rank))
    for i in range(len(changepoints)):
        if (rank_increasing[i] > 0) and i in impact_biggest:
            df.loc[(df['ds'] >= changepoints_list[i]), col] = rank_increasing[i]
            impulse[i] = rank_increasing[i]
    print('rank',rank)
    print('impulse',impulse)
    return df
```

Рисунок 3.15 – Метод знаходження імпульсів приросту

Метод додає новий набір імпульсів до загального об’єкту та відображає результат для поточної ітерації. Приклад результату його виконання зображено на рисунку 3.16.

```
rank [ 3.93878463e-10  3.08327326e-10  3.40575524e-10  6.38196207e-10
  2.88007280e-10 -1.67579255e-10 -8.17705588e-10 -1.59025833e-09
 -1.86584319e-09 -1.34283534e-09 -1.27560140e-09 -1.31635290e-09
 -1.73478640e-09 -1.14586788e-09  1.76146130e-08  3.07272199e-06
  5.82212159e-05  1.13369661e-04  1.68517891e-04  2.23666705e-04
  2.78815254e-04  3.33964239e-04  3.89113857e-04  4.44263798e-04
  4.99414267e-04]
impulse [0.         0.         0.         0.         0.         0.
 0.         0.         0.         0.         0.         0.
 0.         0.         0.         0.00615265 0.116579   0.
 0.         0.         0.         0.         0.         0.
 0.        ]
```

Рисунок 3.16 – Результат роботи методу для знаходження імпульсів приросту

3.3 Реалізація головного алгоритму роботи

В попередньому підрозділі було реалізовані усі основні складові частини для реалізації алгоритму, сформованого в підрозділі 2.3. Спочатку додаємо налаштування з назвами файлів з даними, назви параметрів якості води та назви станцій. Даний код зображено на рисунку 3.17.

```
path = ['bod-in-river-water.csv', 'NO2_in_river_water.csv', 'NH4_in_river_water.csv']
parameters = ['BOD', 'NO2', 'NH4']
time_prediction = 1 # months
stations = ['V', 'K', 'ChV', 'Ch4', 'Ch3', 'Ch2', 'Ch1', 'Ch0']
stations_ua = ['Вінниця', 'Гущинці (Калинівка)', 'Хмільник', 'Щедрове', 'Меджибіж', 'Копистин', 'Хмельницький', 'Чорний Острів']
```

Рисунок 3.17 – Початкові конфігурації

Необхідно у циклі опрацювати кожен набір даних. Для цього створюємо цикл for, в якому зчитуємо дані за допомогою бібліотеки pandas та видаляємо з них непотрібні дані. Код зображено на рисунку 3.18.

```
for i in range(len(path)):
    # Download data
    train = pd.read_csv(path[i], sep = ',')
    print(path[i])
    train = train.drop([0, 1], axis = 0).reset_index(drop=True)
    train.columns = ['Date', 'V', 'K', 'ChV', 'Ch4', 'Ch3', 'Ch2', 'Ch1', 'Ch0']
```

Рисунок 3.18 – Циклічне опрацювання файлів з даними

На кожній ітерації циклу, зображеного на рисунку 3.16, необхідно для кожної з станцій, використовуючи розроблені в підрозділі 3.1 методи, знайти точки зміни тренду, створити об’єкт передбачення Prophet, з нього порахувати дельти та вивести з них необхідні імпульси зміни тренду. Усі ці дані будуть додаватись до результуючого набору. Реалізація зображена на рисунку 3.19, а приклад її результату – на рисунку 3.20.

```
for st in stations:
    # Preparing data
    df = train[['Date', st]]
    df.columns = ['ds','y']
    df = df.dropna().reset_index(drop=True)
    print('number of data -', len(df))
    #print(df)

    # Calculation the changepoints for fbprophet
    changepoints = []
    changepoints = set_changepoints(df)

    # Calculation
    df_pred, deltas, changepoints_new = fbprophet_plot(df, 'Water quality prediction', changepoints,
                                                       25, time_prediction, True)

    # Calc and draw deltas
    deltas_plot(df, deltas, [x.strftime('%Y-%m') for x in changepoints_new], "Station " + st)

    # Add new deltas to df
    impact = impact_add(impact, changepoints_new, deltas, parameters[i]+'-'+st)
```

Рисунок 3.19 – Опрацювання даних по окремих станціях

	ds	BOD-V	BOD-K	BOD-ChV	BOD-Ch4	BOD-Ch3	BOD-Ch2	BOD-Ch1	BOD-Ch0	NO2-V	NO2-K
0	1994-01-31	0.000000	0.000000	0	0.000000	0.000000	0	0.000000	0.000000	0	0.000000
1	1994-02-28	0.000000	0.000000	0	0.000000	0.000000	0	0.000000	0.000000	0	0.000000
2	1994-03-31	0.000000	0.000000	0	0.000000	0.000000	0	0.000000	0.000000	0	0.000000
3	1994-04-30	0.000000	0.000000	0	0.000000	0.000000	0	0.000000	0.000000	0	0.000000
4	1994-05-31	0.000000	0.000000	0	0.000000	0.000000	0	0.000000	0.000000	0	0.000000
...	...	...	...	...	...	...	...	...	...	...	...
302	2019-03-31	0.491322	0.116579	0	0.142881	0.887915	0	0.808082	0.271737	0	0.333068
303	2019-04-30	0.491322	0.116579	0	0.142881	0.887915	0	0.808082	0.271737	0	0.333068
304	2019-05-31	0.491322	0.116579	0	0.142881	0.887915	0	0.808082	0.271737	0	0.333068
305	2019-06-30	0.491322	0.116579	0	0.142881	0.887915	0	0.808082	0.271737	0	0.333068
306	2019-07-31	0.491322	0.116579	0	0.142881	0.887915	0	0.808082	0.271737	0	0.333068

307 rows × 25 columns

Рисунок 3.20 – Результуючий набір даних, отриманих після опрацювання усіх станцій по окремих показниках

Підрахуємо загальне значення по кожній з станцій. Для цього додаємо код, зображений на рисунку 3.21.

```
col_total = ['ds']
for st in stations:
    par = [0]*len(impact)
    for par_name in parameters:
        par += impact[par_name+"-"+st]
    impact[st+'_total'] = par/len(parameters)
    col_total.append(st+'_total')
```

Рисунок 3.21 – Знаходження результуючого значення для кожної станції

Результат виконання даного коду зображено на рисунку 3.22.

	ds	V_total	K_total	ChV_total	Ch4_total	Ch3_total	Ch2_total	Ch1_total	Ch0_total
0	1994-01-31	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	1994-02-28	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2	1994-03-31	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	1994-04-30	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
4	1994-05-31	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
...	...	...	...	...	...	...	...	...	...
302	2019-03-31	0.327736	0.182022	0.025933	0.089293	0.386242	0.155707	0.562512	0.090579
303	2019-04-30	0.327736	0.182022	0.025933	0.089293	0.386242	0.155707	0.562512	0.090579
304	2019-05-31	0.327736	0.182022	0.025933	0.089293	0.386242	0.155707	0.562512	0.090579
305	2019-06-30	0.327736	0.182022	0.025933	0.089293	0.386242	0.155707	0.562512	0.090579
306	2019-07-31	0.327736	0.182022	0.025933	0.089293	0.386242	0.155707	0.562512	0.090579

307 rows × 9 columns

Рисунок 3.22 – Результуючі значення для усіх станцій

Даний набір необхідно відфільтрувати для отримання десяти дат, коли значення приймали найбільших значень між кожною парою з двох сусідніх станцій. Реалізація цього коду зображена на рисунку 3.23, а приклад результату для станцій Вінниця та Гущинці (Калинівка) – на рисунку 3.24.

```
n=10 # number of worst dates from each parameter
worst_dates = []
for i in range(len(stations)):
    st = stations_ua[i]
    st2 = stations_ua[i+1]+' station' if i < len(stations)-2 else 'springhead'
    print('The most problem dates between',st,'station and',st2)
    df = impact_total[['ds',stations[i]+'_total']].nlargest(10, stations[i]+'_total', keep='first')
    display(df)
    worst_dates += df[:n]['ds'].tolist()

worst_dates_all = list(set(worst_dates))
```

Рисунок 3.23 – Фільтрування даних для визначення дат з найбільшими значеннями

	ds	V_total
228	2013-01-31	0.327736
229	2013-02-28	0.327736
230	2013-03-31	0.327736
231	2013-04-30	0.327736
232	2013-05-31	0.327736
233	2013-06-30	0.327736
234	2013-07-31	0.327736
235	2013-08-31	0.327736
236	2013-09-30	0.327736
237	2013-10-31	0.327736

Рисунок 3.24 – Результат фільтрації даних

Отримані дані групуємо по роках та відкидаємо ті, для яких розрахунки не дали ніяких значень. Код та його результат зображені на рисунку 3.25 та 3.26 відповідно.

```
#All worst plots and dates
impact_total[impact_total['ds'].isin(worst_dates_all)]
impact_year = impact_total.groupby(impact.ds.dt.to_period("Y")).sum()
impact_year_interesting = impact_year.loc[~(impact_year==0).all(axis=1)]
```

Рисунок 3.25 – Групування даних по роках

ds	V_total	K_total	ChV_total	Ch4_total	Ch3_total	Ch2_total	Ch1_total	Ch0_total
2002	0.233558	0.000000	0.000000	0.000000	0.042192	0.000000	0.000000	0.000000
2003	0.314363	0.000000	0.000000	0.000000	0.086598	0.000000	0.000000	0.000000
2004	0.402775	0.000000	0.000000	0.000000	0.167494	0.000000	0.000000	0.000000
2005	0.402775	0.000000	0.000000	0.000000	0.359179	0.000000	0.067918	0.000000
2006	0.402775	0.000000	0.000000	0.000000	0.583785	0.000000	0.074092	0.000000
2007	0.700619	0.000000	0.311196	0.000000	0.604204	0.000000	0.115861	0.000000
2008	1.782258	0.000000	0.311196	0.000000	0.604204	0.000000	0.208265	0.000000
2009	2.042791	0.385680	0.311196	0.000000	0.604204	0.035482	0.239066	0.000000
2010	2.295605	0.410290	0.311196	0.000000	0.848374	0.085157	0.492680	0.000000
2011	3.296659	0.938343	0.311196	0.458333	1.269685	0.514657	1.473536	0.000000
2012	3.570756	1.193577	0.311196	1.023895	1.543780	0.844403	1.916383	0.000000
2013	3.932827	1.650828	0.311196	1.071522	2.544332	1.258741	2.437039	0.000000
2014	3.932827	2.139814	0.311196	1.071522	2.919797	1.602051	2.790137	0.000000
2015	3.932827	2.184267	0.311196	1.071522	3.594352	1.868489	4.754254	1.086947
2016	3.932827	2.184267	0.311196	1.071522	3.960554	1.868489	5.852971	1.086947
2017	3.932827	2.184267	0.311196	1.071522	4.549823	1.868489	6.665042	1.086947
2018	3.932827	2.184267	0.311196	1.071522	4.634899	1.868489	6.750139	1.086947
2019	2.294149	1.274156	0.181531	0.625054	2.703691	1.089952	3.937581	0.634053

Рисунок 3.26 – Набір згрупованих по роках показників

Для візуалізації отриманих даних, додаємо код, який зображений на рисунку 3.27. Він повинен відмалювати діаграму для кожної станції з розрахованими значеннями по роках.

```
# Draw result plot
fig = plt.figure(figsize=(20,10))
ax = fig.add_subplot(111, projection='3d')
colors = ['r', 'b', 'g', 'c', 'm', 'lime', 'orange', 'blue']
xs = impact_year_interesting.index.astype(str).tolist()
for i in range(len(stations_ua)):
    ys = impact_year_interesting[stations[i]+"_total"].values
    #print(xs, ys, stations_ua[i])
    ax.bar(xs, ys, zs=i, zdir='y', color=colors[i], alpha=0.8)

ax.set_xlabel('Date', labelpad=15)
ax.set_ylabel('Station', labelpad=15)
ax.set_zlabel('Impact')
ax.set_yticks(range(len(stations_ua)))
ax.set_yticklabels(stations_ua)
plt.show()
```

Рисунок 3.27 – Візуалізація результатів роботи програми

3.4 Тестування

Протестуємо роботу розробленої технології. Для перевірки роботи візьмемо набір даних показників моніторингу води на пунктах від витоку р. Південний Буг до м. Вінниці (Чорний Острів», «Хмельницький», «Копистин», «Меджибіж», «Щедрове», «Хмільник», «Гущинці (Калинівка)», «Вінниця»). Набір містить отримані значення по показниках $БСК_5$ (біохімічне споживання кисню за 5 днів, гранично допустиме значення 3,0 мг/л), концентрація азоту з перерахунку на нітрати NO_2 (гранично допустима концентрація 45,0 мг/л) та концентрація азоту амонійного NH_4 (2,0 мг/л) за період 2002-2019 рр. За перерахованими показниками на згаданих постах, протягом останніх декількох років, спостерігаються відхилення від допустимих значень.

На рисунку 3.28 зображено діаграму, яка у просторі та часі відображає значення приростів забруднюючих речовин на відповідних постах моніторингу якості води ділянки р. Південний Буг від витоку до м. Вінниці

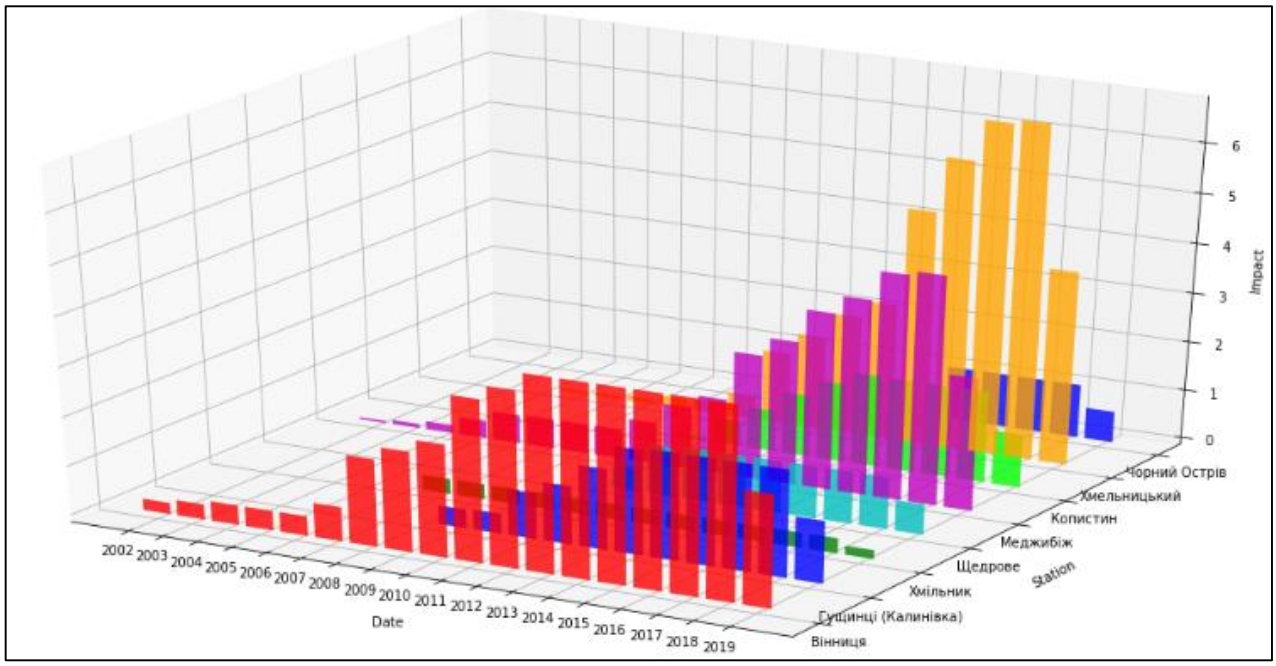


Рисунок 3.28 – Результат застосування розробленої технології на реальних даних

Проаналізувавши рисунок, видно що на пунктах «Хмельницький», «Меджибіж» та «Вінниця», починаючи з 2012-2013 років, спостерігається збільшення забруднення річки. Також варто зазначити, що в 2019 році показники забруднення різко пішли на спад, що також може бути досить дивною поведінкою.

3.5 Висновки

В розділі виконано порівняльний аналіз та обрано мову програмування Python для реалізації розробленого раніше алгоритму інформаційної технології для пошуку джерел забруднення річки.

Було реалізовано інформаційну технологію пошуку джерел забруднення річки та перевірено її роботу на реальних даних моніторингу води Держводагентства на пунктах спостереження від витоку р. Південний Буг до м. Вінниці.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Оцінювання комерційного потенціалу розробки інформаційної технології пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки. Для проведення технологічного аудиту було залучено 3-х незалежних експертів. Проведемо оцінку комерційного потенціалу інформаційної технології за 12-ма критеріями, наведеними в таблиці 4.1 за 5-ти бальною шкалою.

Таблиця 4.1 – Критерії оцінювання комерційного потенціалу розробки та їх бальна оцінка

Бали (за 5-ти бальною шкалою)					
Кри-те-Рій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші,ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші,ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі,ніж в аналогів	Технічні та споживчі властивості продукту значно кращі,ніж в аналогів
5	Експлуатаційні витрати значно вищі,ніж в аналогів	Експлуатаційні витрати дещо вищі,ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі,ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів

Продовження таблиці 4.1

Бали (за 5-ти бальною шкалою)					
Крите.	0	1	2	3	4
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної,так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної,так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси,які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали,що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10років	Термін реалізації ідеї більший за 5років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х Років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту,що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Отримані результати оцінювання комерційного потенціалу інформаційної технології за 5-ти бальною шкалою наведено в таблиці 4.2.

Таблиця 4.2 – Результати оцінювання комерційного потенціалу розробки

Критерії			
	Експерт 1	Експерт 2	Експерт 3
	Бали, виставлені експертами:		
1	4	3	3
2	2	2	2
3	4	4	4
4	2	3	3
5	4	4	4
6	3	3	4
7	1	1	2
8	3	3	3
9	4	4	4
10	4	3	3
11	3	3	3
12	4	4	4
Сума балів	$СБ_1 = 38$	$СБ_2 = 37$	$СБ_3 = 39$
Середньоарифметична сума балів	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{3} = 38$		

Оскільки з даних, зазначених в таблиці 4.2, було отримано значення середньоарифметичної суми балів дорівнює 38, то рівень комерційного потенціалу відповідає показнику рівню «вище середнього».

4.2 Прогнозування витрат на виконання науково-дослідної роботи

Для розробки нового програмного продукту необхідні такі витрати. Основна заробітна плата для розробників визначається за формулою 4.1:

$$З_о = \frac{М}{Т_р} \cdot t \ \text{[грн]}, \qquad (4.1)$$

де М – місячний посадовий оклад конкретного розробника;

$T_р$ – кількість робочих днів у місяці, $T_р$ = 21 дні;

t – число днів роботи розробника, t = 42 днів.

В таблиці 4.3 наведені розрахунки заробітних плат для керівника і програміста.

Таблиця 4.3 – Розрахунки основної заробітної плати

Працівник	Посадовий оклад , грн.	Оплата за робочий день, грн.	Число днів роботи, t	Витрати на оплату праці, грн.
Науковий керівник	8 447	402.25	3	1 206
Інженер-програміст	10 000	476.2	42	20 000
Всього:				21 206

Також необхідно розрахувати додаткову заробітну плату $З_д$ всіх розробників. Дане значення розраховується як 10-12% від суми основної заробітної плати всіх розробників:

$$З_д = (0{,}1 \dots 0{,}12) \cdot З_о = 0{,}121\,206 = 2\,120{,}6 \text{ (грн)}. \quad (4.2)$$

Нарахування на заробітну плату $Н_{зп}$ для працівників бюджетної сфери у 2020 році становить 22% від суми основної та додаткової заробітної плати:

$$Н_{зп} = (З_о + З_д) \cdot 22\% = (21\,206 + 2\,120{,}6) \cdot 0{,}22 = 5\,131{,}85 \text{ (грн)}. \quad (4.3)$$

Розрахунок амортизаційних витрат для обладнання виконується за такою формулою:

$$А = \frac{Ц \cdot Н_а}{100} \cdot \frac{Т}{12}, \quad (4.4)$$

де Ц – балансова вартість обладнання, грн;

$Н_а$ – річна норма амортизаційних відрахувань %

Т – Термін використання (Т=2 міс.).

Результат розрахунків наведено на рисунку 4.4.

Таблиця 4.4 – Розрахунок амортизаційних відрахувань

Найменування програмного забезпечення	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн
Персональний комп'ютер	12000	25	2	500
ПЗ для роботи з Python	8 000	15	2	200
Всього:				700

Розрахуємо витрати на комплектуючі. Витрати на комплектуючі розрахуємо за формулою:

$$К = \sum_1^n Н_і \cdot Ц_і \cdot К_і, \quad (4.5)$$

де n – кількість комплектуючих;

$Н_і$ – кількість комплектуючих і-го виду;

$Ц_і$ – покупна ціна комплектуючих і-го виду, грн;

$К_і$ – коефіцієнт транспортних витрат (приймемо $К_і$ = 1,1).

Таблиця 4.5 - Витрати на комплектуючі, що були використані для наукової розробки.

Найменування матеріалу	Одиниці виміру	Ціна, грн.	Витрачено	Вартість витрачених матеріалів, грн.
Диск	шт.	23	1	23
Офісний папір	уп.	120	1	120

Продовження таблиці 4.5

Ручка	шт.	8	1	7
Всього з урахуванням транспортних витрат				150

Витрати на силову електроенергію розраховуються за формулою:

$$В_е = В \cdot П \cdot Ф \cdot К_п \,, \tag{4.6}$$

де В – вартість 1кВт-години електроенергії (В=2,3 грн/кВт);

П – установлена потужність комп'ютера (П=0,5кВт);

Ф – фактична кількість годин роботи комп'ютера (Ф=252 год.);

$К_п$ – коефіцієнт використання потужності ($К_п < 1$, $К_п = 0{,}6$).

$$В_е = 2{,}3 \cdot 0{,}5 \cdot 252 \cdot 0{,}6 = 68{,}04 \text{ (грн)}.$$

Інші витрати $І_в$ можна прийняти як (100…300)% від суми основної заробітної плати розробників та робітників, які були виконували дану роботу, тобто:

$$І_в = (1..3) \cdot (З_о + З_р). \tag{4.7}$$

Отже, розрахуємо інші витрати:

$$І_в = 1 \cdot (21\,206 + 2\,120{,}6) = 23\,326{,}6 \text{ (грн)}.$$

Сума всіх попередніх статей витрат дає витрати на виконання даної частини роботи:

$$В = З_о + З_д + Н_{зп} + А + К + В_е + І_в, \tag{4.8}$$

В = 21 206+2 120,6+5 131,85 +700+150+68,04+23 326,6 = 52 632,08 (грн).

Розрахуємо загальну вартість наукової роботи $B_{заг}$ за формулою:

$$В_{заг} = \frac{В_{ін}}{\alpha}, \quad (4.9)$$

де α – частка витрат, які безпосередньо здійснює виконавець даного етапу роботи, у відносних одиницях = 0.9.

$$В_{заг} = \frac{52\ 632,09}{0.9} = 58480.1 \text{ (грн)}.$$

Прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи здійснюється за формулою:

$$ЗВ = \frac{В_{заг}}{\beta}, \quad (4.10)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи.

Отже, розрахуємо загальні витрати:

$$ЗВ = \frac{58480,1}{0,9} = 64977.88 \text{ (грн)}.$$

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

Спрогнозуємо кількісно, яку вигоду можна отримати у майбутньому від впровадження результатів виконаної наукової роботи.

Виконання наукової роботи та впровадження її результатів буде здійснюватися протягом одного року. Основні позитивні результати від впровадження розробки очікуються протягом 3-х років після її впровадження. Одним із основних позитивних результатів є зростання величини прибутку.

При реалізації результатів наукової розробки покращується якість програмного продукту, що дозволяє підвищити ціну його реалізації на 200 грн. Кількість одиниць реалізації програмного засобу також збільшиться: протягом першого року — на 800 шт., протягом другого року — ще на 500 шт., протягом третього року — ще на 300 шт.

Реалізація продукції до впровадження результатів наукової розробки складала 40 шт., а ціна — 350 грн.

Спрогнозуємо збільшення чистого прибутку підприємства від впровадження результатів наукової розробки у кожному році відносно базового за такою формулою 4.11:

$$\Delta П_i = \sum_1^n (\Delta Ц_o \cdot N + Ц_o \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\upsilon}{100}\right), \qquad (4.11)$$

де $\Delta П_o$ – покращення основного оціночного показника від впровадження результатів розробки у даному році. Зазвичай таким показником може бути ціна одиниці нової розробки;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$Ц_o$ – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки;

λ – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт 0,8333;

ρ – коефіцієнт, який враховує рентабельність продукту. Рекомендується приймати = 0,2...0,3;

υ – ставка податку на прибуток, у 2020 році, υ = 18%.

Тоді, збільшення чистого прибутку підприємства протягом першого року складе:

$$\Delta П_1 = [200 \cdot 40 + (200 + 350) \cdot 800] \cdot 0{,}8333 \cdot 0{,}2 \cdot \left(1 - \frac{18}{100}\right) = 61\,224{,}22 \text{ (грн).}$$

Збільшення чистого прибутку підприємства протягом другого року (відносно базового року, тобто року до впровадження результатів наукової розробки) складе:

$$\Delta П_2 = [200 \cdot 40 + (200 + 350) \cdot (800 + 500)] \cdot 0{,}8333 \cdot 0{,}2 \cdot \left(1 - \frac{18}{100}\right) = 98\,806{,}05 \text{ (грн).}$$

Збільшення чистого прибутку підприємства протягом третього року (відносно базового року, тобто року до впровадження результатів наукової розробки) складе:

$$\Delta П_3 = [200 \cdot 40 + (200 + 350) \cdot (800 + 500 + 300)] \cdot 0{,}8333 \cdot 0{,}2 \cdot \left(1 - \frac{18}{100}\right) = 121\,355{,}15 \text{ (грн).}$$

Отже, протягом трьох років підприємство може розраховувати на збільшення чистого прибутку від реалізації наукової розробки.

4.4 Розрахунок ефективності вкладених інвестицій та період їх окупності

Визначимо абсолютну і відносну ефективність вкладених інвестором інвестицій та розрахуємо термін окупності.

Абсолютна ефективність $E_{абс}$ вкладених інвестицій розраховується за формулою:

$$E_{абс} = (ПП - PV), \quad (4.12)$$

де ПП – приведена вартість всіх чистих прибутків, які отримає підприємство (організація) від реалізації результатів наукової розробки, грн.;

PV – теперішня вартість інвестицій PV = ЗВ = 58 457,87 грн.

На рисунку 4.1 зображено як буде виглядати рух платежів (інвестицій та додаткових прибутків).

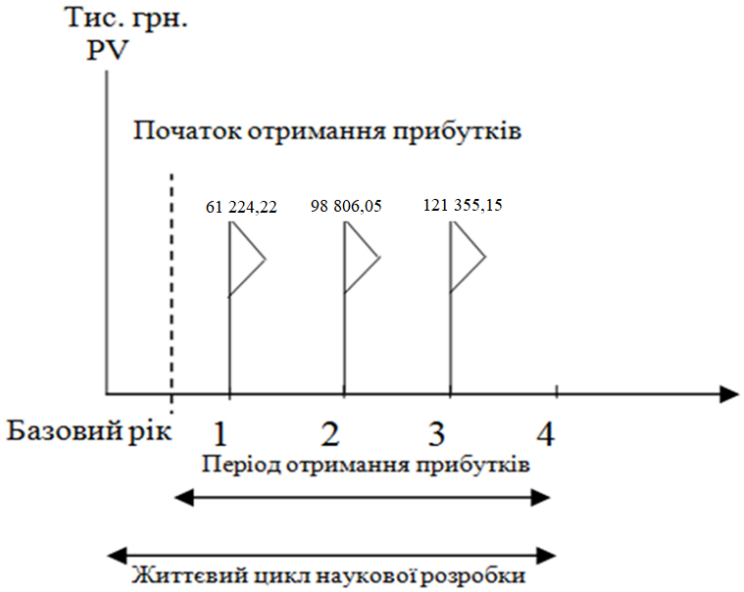


Рисунок 4.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів наукової роботи

У свою чергу, приведена вартість всіх чистих прибутків ПП розраховується за формулою:

$$ПП = \sum_1^T \frac{\Delta П_i}{(1+\tau)^t}, \quad (4.13)$$

де $\Delta П_i$ –збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

т – період часу, протягом якого виявляються результати впровадженої НДДКР, 3 роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

t – період часу (в роках) від моменту отримання чистого прибутку до точки "0".

Отже, розрахуємо вартість чистого прибутку:

$$ПП = \frac{58\,457,87}{(1+0,1)^0} + \frac{61\,224,22}{(1+0,1)^2} + \frac{98\,806,05}{(1+0,1)^3} + \frac{121\,355,15}{(1+0,1)^4} = 266\,178,05 \text{ (грн).}$$

Тоді розрахуємо $E_{абс}$:

$$Е_{абс} = 266\,178,05 - 64977.88 = 207\,720,18 \text{ (грн).}$$

Оскільки $E_{абс} > 0$, то вкладання коштів на виконання та впровадження результатів НДДКР буде доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій $Е_в$ за формулою:

$$Е_в = \sqrt[Т]{1 + \frac{Е_{абс}}{PV}} - 1, \qquad (4.14)$$

де $Е_{абс}$ – абсолютна ефективність вкладених інвестицій, грн;

PV – теперішня вартість інвестицій PV = ЗВ, грн;

Т – життєвий цикл наукової розробки, роки.

Тоді будемо мати:

$$Е_в = \sqrt[3]{1 + \frac{207\,720,18}{64977.88}} - 1 = 0,67 \text{ або } 67\%$$

Далі, розраховану величина $Е_в$ порівнюємо з мінімальною (бар'єрною) ставкою дисконтування $\tau_{мін}$, яка визначає ту мінімальну дохідність, нижче за яку

інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування $\tau_{мін}$ визначається за формулою:

$$\tau = d + f, \tag{4.15}$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2020 році в Україні d = 0,2;

f – показник, що характеризує ризикованість вкладень, величина f = 0,1.

$$\tau = 0{,}2 + 0{,}1 = 0{,}3.$$

Оскільки $E_в$ = 66% > $\tau_{мін}$ = 0,3 = 30%, то інвестор буде зацікавлений вкладати гроші в дану наукову розробку.

Термін окупності вкладених у реалізацію наукового проекту інвестицій $T_{ок}$ розраховується за формулою:

$$T_{ок} = \frac{1}{E_в}, \tag{4.16}$$

Якщо Ток< 3…5-ти років, то фінансування наукової розробки є доцільним.

$$T_{ок} = \frac{1}{0{,}67} = 1{,}5 \text{ (роки)}.$$

Обрахувавши термін окупності даної наукової розробки, можна зробити висновок, що фінансування даної наукової розробки буде доцільним, оскільки Ток<5 років.

4.5 Висновки

Результати проведених розрахунків дають можливість зробити висновок про доцільність розробки та впровадження нашої наукової роботи. Це підтверджують такі показники як:

– абсолютна ефективність вкладених інвестицій, яка дорівнює 207 720,18 грн., що є більшим 0 і вказує на те, що інвестор може бути зацікавленим у нашій розробці;

– відносна ефективність наукової розробки становить 67%, що є вищим за мінімальну ставку дисконтування (30%), тому вкласти кошти у нашу розробку є вигідніше, ніж покласти кошти на депозит;

– термін окупності вкладених у реалізацію наукового проекту інвестицій складе 1,5 роки, що є менше 5-ти і вказує швидку окупність вкладених інвестицій.

Крім того, розраховано, що наукова розробка приноситиме установі додатковий прибуток протягом 3-х років за рахунок покращення її якості порівняно з існуючими аналогами.

ВИСНОВКИ

Під час виконання магістерської кваліфікаційної роботи розглянуто проблему пошуку основних антропогенних джерел забруднення водойм за регулярними багаторічними даними моніторингу вод України.

Проаналізовано наявні відкриті дані державної системи моніторингу України за 1920-2020 рр. і було виявлено наявність чималих пропусків та аномальних даних. Для вирішення поставленої задачі було обрано модель Prophet, передусім, через її можливості роботи із зашумленими даними з певною періодичністю.

Розроблено й охарактеризовано математичний і алгоритмічний апарат запропонованої технології та створена за рахунок використання моделі Prophet з кусково-лінійною апроксимацією тренду для локалізації в часі та просторі забруднюючого впливу по похідній цього тренду. Доведено, що особливістю запропонованої технології є її стійкість до аномальних завад з різкою зміною якості чи кількості води (аварій, межені, повені тощо), які спеціальним чином враховує блок параметрів «holidays» моделі Prophet, та зміни водності річки, за рахунок аналізу багаторічних даних та виявлення сталих трендів, що мають місце тривалий час.

Створена програма для реалізації цієї технології на Python. Проведено її успішне випробування на ділянці р. Південний Буг на реальних даних 8 постів державної системи моніторингу вод від витоку р. Південний Буг до м. Вінниці за даними 2002—2019 рр. по показниках БСК5 (біохімічне споживання кисню за 5 днів), концентрація нітратів NO2 та концентрація азоту амонійного NH4, за якими останніми роками часто фіксуються наднормативні значення.

Під час виконання економічної частини кваліфікаційної роботи на основі розрахунків було доведено, що продукт є економічно доцільним, оскільки витрати на розробку вказаного засобу з використанням відповідної технології становлять 58 457 грн. Показник ефективності вкладених інвестицій $E_{абс}$ = 207 720,18 (грн), відносної – 67%, а термін окупності інвестицій становить 1,5 роки.

За результатами магістерської кваліфікаційної роботи опубліковано статтю в науковому фаховому періодичному виданні «Вісник Вінницького політехнічного інституту» категорії Б зі спеціальності 126 – «Інформаційні системи та технології», яке входить в Index Copernicus.

Отже, поставлені задачі магістерської кваліфікаційної роботи розв’язані в повному обсязі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Водний Кодекс України. Введений в дію Постановою ВР № 214/95-ВР від 06.06.95. Відомості Верховної Ради (ВВР), № 24, ст. 189, 1995.

2. Єдиний державний веб-портал відкритих даних [Електронний ресурс] – Режим доступу до ресурсу: https://data.gov.ua/

3. Моніторинг та екологічна оцінка водних ресурсів України [Електронний ресурс] – Режим доступу до ресурсу: http://monitoring.davr.gov.ua/

4. Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet / В. Б. Мокін, О. В. Слободянюк, О. М. Давидюк, Д. О. Шмундяк // Вісник Вінницького політехнічного інституту. — 2020. — № 4 — С. 15-24.

5. Water pollution [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Water_pollution

6. Water distribution on Earth [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Water_distribution_on_Earth

7. Державний водний кадастр: [Електронний ресурс] – Режим доступу до ресурсу: http://geoportal.davr.gov.ua

8. Забруднення річок України: причини та наслідки [Електронний ресурс] – Режим доступу до ресурсу: https://ns-plus.com.ua/2019/07/10/zabrudnennya-richok-ukrayiny-prychyny-ta-naslidky/

9. Типи забруднення води і їх наслідки [Електронний ресурс] – Режим доступу до ресурсу: https://www.akvantis.com.ua/ua/stati-i-obzory/tipy-zagryazneniya-vody-i-ih-posledstviya

10. Моніторинг та екологічна оцінка водних ресурсів України [Електронний ресурс] – Режим доступу до ресурсу: https://www.davr.gov.ua/

11. Постанова про затвердження порядку здійснення державного моніторингу вод [Електронний ресурс] – Режим доступу до ресурсу: https://zakon.rada.gov.ua/laws/show/758-2018-п

12. Інформаційна технологія пошуку незареєстрованих місць надходження забруднених вод у річку за супутниковими і векторними даними геопорталів / В. Б. Мокін, Л. М. Скорина, Є. М. Крижановський // Наукові праці ДонНТУ, «Інформатика, кібернетика та обчислювальна техніка», № 2, с. 30-35, 2018

13. Machine learning [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Machine_learning

14. What is Machine Learning? A definition [Електронний ресурс] – Режим доступу до ресурсу: https://expertsystem.com/machine-learning-definition/

15. Математичні методи ідентифікації динамічних систем : навчальний посібник / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін. – Вінниця : ВНТУ, 2010. – С. 124-128

16. Time series [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Time_series

17. Machine learning for time series forecasting [Електронний ресурс] – Режим доступу до ресурсу: https://codeit.us/blog/machine-learning-time-series-forecasting

18. Prophet [Електронний ресурс] – Режим доступу до ресурсу: https://facebook.github.io/prophet/

19. Удосконалення технології аналізу даних дозвільної документації зі спеціального водокористування в системі Держводагентства / В. Б. Мокін, Л. М. Скорина, А. Р. Ящолт // Вісник Вінницького політехнічного інституту, № 4, с. 22-31, 2017

20. Інструкція про порядок розробки та затвердження гранично допустимих (ГДС) речовин у водні об'єкти із зворотними водами. Харків, Україна: УкрНЦОВ, 1994.

21. Комп’ютеризовані регіональні системи державного моніторингу поверхневих вод: моделі, алгоритми, програми, монографія / В. Б. Мокін, і А. Р. Ящолт. – Вінниця, Україна: ВНТУ, 2005, 78-85 с.

22. R Vs Python: What’s the Difference? [Електронний ресурс] – Режим доступу до ресурсу: https://www.guru99.com/r-vs-python.html

23. Изучаем Python, 4-е издание / Лутц М – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с.

24. Hadley Wickham (Author), Garrett Grolemund. R for Data Science: Import, Tidy, Transform, Visualize, and Model Dataб 2017 – 495 с

25. Методичні вказівки до виконання та оформлення магістерських кваліфікаційних робіт для студентів спеціальності 126 –«Інформаційні системи та технології» денної форми навчання / Уклад. В. Б. Мокін, С. О. Жуков, А. Р. Ящолт, О. М. Козачко, Л. М. Скорина. – Електронне видання. – Вінниця : ВНТУ, 2018. – 48 с.

Додаток А

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет комп'ютерних систем і автоматики

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ
_____д. т. н., проф. В.Б. Мокін
(підпис)
"__" _____ 2020 р

ТЕХНІЧНЕ ЗАВДАННЯ

на магістерську кваліфікаційну роботу

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПОШУКУ МОЖЛИВИХ ДЖЕРЕЛ ПІДВИЩЕНОГО ЗАБРУДНЕННЯ РІЧКИ З ВИКОРИСТАННЯМ МОДЕЛІ PROPHET

08–53.МКР.007.02.00.ТЗ

Керівник магістерської
кваліфікаційної роботи
д.т.н., проф.

_____В.Б. Мокін
(підпис)
"__" _____ 2020 р.
Розробив студент гр. 2ІСТ-19м

_____Д. О. Шмундяк
(підпис)
"__" _____ 2020 р.

Вінниця 2020

1. Підстава для проведення робіт

Підставою для виконання роботи є наказ № __ по ВНТУ від «__» _____ 2020 р., та індивідуальне завдання на МКР, затверджене протоколом № __ засідання кафедри САІТ від «__» _____ 2020 р.

2. Джерела розробки

– Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet / В. Б. Мокін, О. В. Слободянюк, О. М. Давидюк, Д. О. Шмундяк // Вісник Вінницького політехнічного інституту. — 2020. — № 4 — С. 15-24.

– Моніторинг та екологічна оцінка водних ресурсів України [Електронний ресурс] – Режим доступу до ресурсу: http://monitoring.davr.gov.ua/

– Prophet [Електронний ресурс] – Режим доступу до ресурсу: https://facebook.github.io/prophet/

3. Мета і призначення роботи

Розробка інформаційної технології для пошуку можливих джерел підвищеного забруднення річки.

4. Вихідні дані для проведення робіт

– дані моніторингу вод Держводагентства за 2002-2019 роки.

5. Методи дослідження

– моделі машинного навчання;
– прогнозування та аналіз часових рядів.

6. Етапи роботи і терміни їх виконання

a) Аналіз предметної області ..... __. – __
b) Огляд моделі розв’язання задачі ..... __ – __
c) Розроблення математичного та алгоритмічного апарату . __ – __
d) Розроблення інформаційної технології. ..... __ – __
e) Тестування розробленої інформаційної технології на реальних даних..... __ – __

7. Очікувані результати та порядок реалізації

Створений математичний і алгоритмічний апарат та програмне забезпечення, яке реалізує його.

8. Вимоги до розробленої документації

Пояснювальна записка оформлена у відповідності до вимог «Методичних вказівок до виконання та оформлення магістерських кваліфікаційних робіт для студентів спеціальності 126 – «Інформаційні системи та технології» денної форми навчання».

9. Порядок приймання роботи

Публічний захист «___» _____2020 р.
Початок розробки «___» _____2020 р.
Граничні терміни виконання МКР «___» _____2020 р.

Розробив студент групи 2ІСТ-19м _____ Шмундяк Д. О.

Додаток Б

Лістинг програми

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# preprocessing
from sklearn.preprocessing import LabelEncoder, StandardScaler, MinMaxScaler
from datetime import timedelta, datetime
from more_itertools import locate

# models
from fbprophet import Prophet
from fbprophet.plot import plot_plotly
from fbprophet.plot import add_changepoints_to_plot

import warnings
warnings.filterwarnings("ignore")

path = ['bod-in-river-water.csv', 'NO2_in_river_water.csv', 'NH4_in_river_water.csv']
parameters = ['BOD', 'NO2', 'NH4']
time_prediction = 1 # months
stations = ['V', 'K', 'ChV', 'Ch4', 'Ch3', 'Ch2', 'Ch1', 'Ch0']
stations_ua = ['Вінниця', 'Гущинці (Калинівка)', 'Хмільник', 'Щедрове', 'Меджибіж',
'Копистин', 'Хмельницький', 'Чорний Острів']


def m_fbprophet(df, changepoints, n_changepoints):
    # Return object of fbprophet with given parameters and calculated trend

    num = len(df) // 6
    if (changepoints == []) and (num < n_changepoints):
        n_changepoints = num

    m = Prophet(
                daily_seasonality = False,
                yearly_seasonality = 4,
                weekly_seasonality = False,
                changepoints = changepoints,
                n_changepoints = n_changepoints
                )

    # calculation linear trend by FB Prophet
    m.fit(df, algorithm='Newton')

    return m

def fbprophet_plot(df, title, changepoints, n_changepoints, time_prediction, output):
    # Calculation fbprophet prediction on 1 year for df with changepoints as the
initial approximatio
    # and if output = True then output data and draw plots


    # Initialization FB Prophet object and calculation trend
    model = m_fbprophet(df, changepoints, n_changepoints)
```

```
    future = model.make_future_dataframe(periods=time_prediction, freq = "M")
    forecast = model.predict(future)

    if output:
        print('fbprophet: ' + title)
        print('model.delta', model.params['delta'])

        fig_forecast = model.plot(forecast)
        for changepoint in model.changepoints:
            plt.axvline(changepoint, ls='--', lw=1, color='red')
        a = add_changepoints_to_plot(fig_forecast.gca(), model, forecast)
        fig_components = model.plot_components(forecast)

    return forecast[['ds', 'trend', 'trend_lower', 'trend_upper']],
model.params['delta'][0], model.changepoints


def monotonically_increasing(l):
    return all(x < y for x, y in zip(l, l[1:]))

def get_increasing_index(data):
    result = -1
    for index in range(len(data)):
        is_increasing = monotonically_increasing(data[index:])
        if (is_increasing):
            result = index
            print('nonzero start from -',result, 'len -',len(data))
            break

    return result

def impact_add(df, changepoints, deltas, col):
    # Add deltas to df
    # col - name of a new column in df

    # Calculation integral of trend
    rank = np.cumsum(np.cumsum(deltas))

    # Set zeros before increasing
    index = get_increasing_index(rank)
    rank_increasing = np.zeros(len(rank))
    if(index > -1):
        rank_increasing = np.concatenate((np.zeros(index), rank[index:]), axis=None)
    #rank = (rank - min(rank))/(max(rank)-min(rank))
    rank_increasing = rank_increasing/max(rank_increasing)

    df[col] = 0
    changepoints_list = changepoints.tolist()

    # Find peaks in deltas
    impact_biggest = list(locate(deltas, lambda x: x > 0.05*max(deltas)))

    # Find impulses
    impulse = np.zeros(len(rank))
    for i in range(len(changepoints)):
        if (rank_increasing[i] > 0) and i in impact_biggest:
            df.loc[(df['ds'] >= changepoints_list[i]), col] = rank_increasing[i]
            impulse[i] = rank_increasing[i]
```

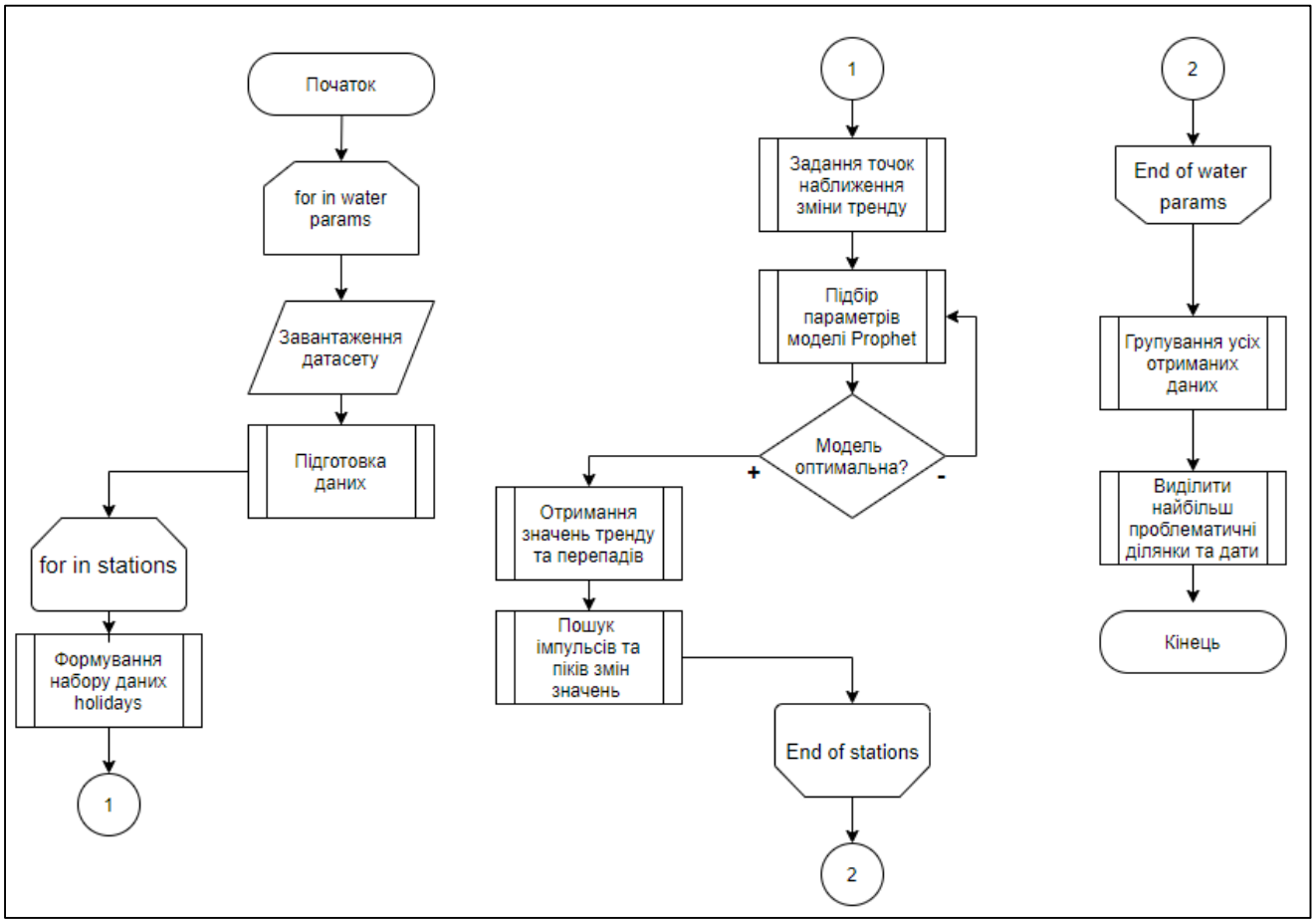
```
    print('rank',rank)
    print('impulse',impulse)
    return df

def calc(path):
    # All calculation for parameter from files with path

    # Result
    impact = pd.DataFrame()
    impact['ds'] = pd.date_range(start='1994/1/1', end='2019/8/1', freq='M')

    for i in range(len(path)):
        # Download data
        train = pd.read_csv(path[i], sep = ',')
        print(path[i])
        train = train.drop([0, 1], axis = 0).reset_index(drop=True)
        train.columns = ['Date', 'V', 'K', 'ChV', 'Ch4', 'Ch3', 'Ch2', 'Ch1', 'Ch0']

        for st in stations:
            # Preparing data
            df = train[['Date', st]]
            df.columns = ['ds','y']
            df = df.dropna().reset_index(drop=True)
            print('number of data -', len(df))
            #print(df)

            # Calculation the changepoints for fbprophet
            changepoints = []
            changepoints = set_changepoints(df)

            # Calculation
            df_pred, deltas, changepoints_new = fbprophet_plot(df, 'Water quality
prediction', changepoints,
                                                               25, time_prediction,
True)

            # Calc and draw deltas
            deltas_plot(df, deltas, [x.strftime('%Y-%m') for x in changepoints_new],
"Station " + st)

            # Add new deltas to df
            impact = impact_add(impact, changepoints_new, deltas, parameters[i]+'-
'+st)

    impact = impact.sort_values(by='ds').reset_index(drop=True)
    return impact

impact

col_total = ['ds']
for st in stations:
    par = [0]*len(impact)
    for par_name in parameters:
        par += impact[par_name+"-"+st]
    impact[st+'_total'] = par/len(parameters)
    col_total.append(st+'_total')
col_total

impact_total = impact[col_total]
```

```
impact_total

impact.to_csv('impact.csv', index = False)
impact_total.to_csv('impact_total.csv', index = False)

impact_total.describe()

n=10 # number of worst dates from each parameter
worst_dates = []
for i in range(len(stations)):
    st = stations_ua[i]
    st2 = stations_ua[i+1]+' station' if i < len(stations)-2 else 'springhead'
    print('The most problem dates between',st,'station and',st2)
    df = impact_total[['ds',stations[i]+'_total']].nlargest(10, stations[i]+'_total',
keep='first')
    display(df)
    worst_dates += df[:n]['ds'].tolist()

worst_dates_all = list(set(worst_dates))
print(len(worst_dates_all),len(worst_dates))

#pd.options.display.float_format = '{:,.6f}'.format

#All worst plots and dates
impact_total[impact_total['ds'].isin(worst_dates_all)]

impact_year = impact_total.groupby(impact.ds.dt.to_period("Y")).sum()
impact_year
impact_year_interesting = impact_year.loc[~(impact_year==0).all(axis=1)]
impact_year_interesting

# Draw result plot
fig = plt.figure(figsize=(20,10))
ax = fig.add_subplot(111, projection='3d')
colors = ['r', 'b', 'g', 'c', 'm', 'lime', 'orange', 'blue']
xs = impact_year_interesting.index.astype(str).tolist()
for i in range(len(stations_ua)):
    ys = impact_year_interesting[stations[i]+"_total"].values
    #print(xs, ys, stations_ua[i])
    ax.bar(xs, ys, zs=i, zdir='y', color=colors[i], alpha=0.8)
ax.set_xlabel('Date', labelpad=15)
ax.set_ylabel('Station', labelpad=15)
ax.set_zlabel('Impact')
ax.set_yticks(range(len(stations_ua)))
ax.set_yticklabels(stations_ua)
plt.show()
```

Додаток В

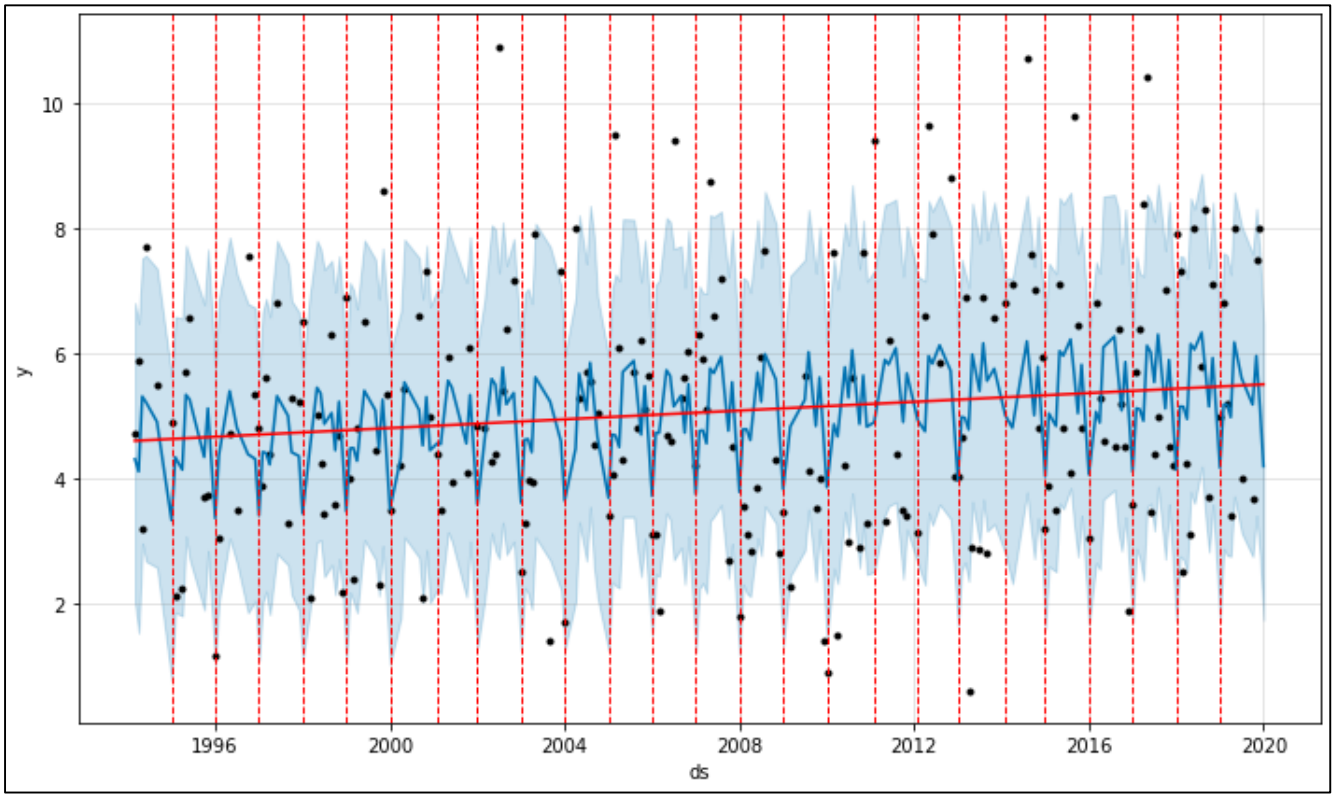
Графічна частина

Блок-схема алгоритму інформаційної технології



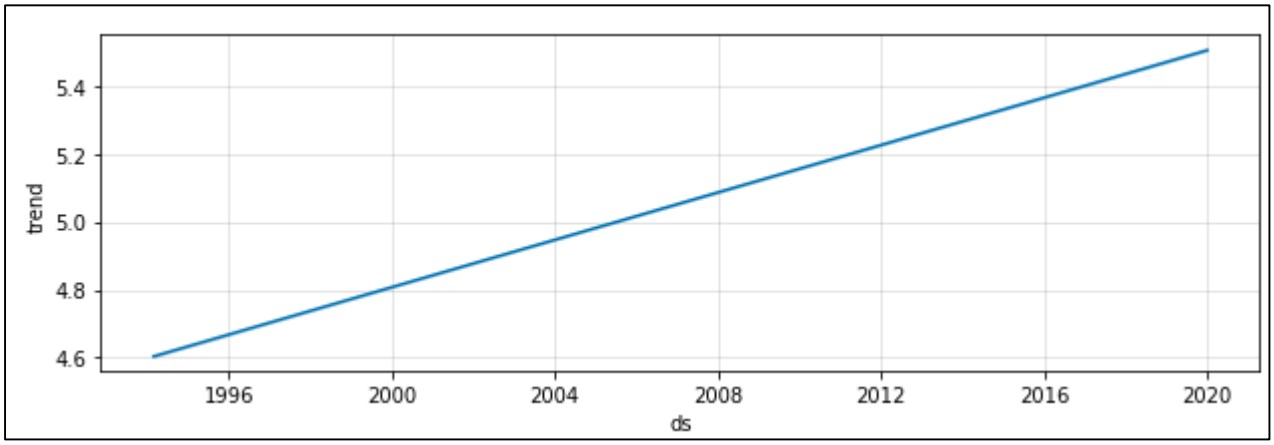
					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 1	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.					Блок-схема алгоритму інформаційної технології	2ІСТ-19м		
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Графік передбачення моделі Prophet



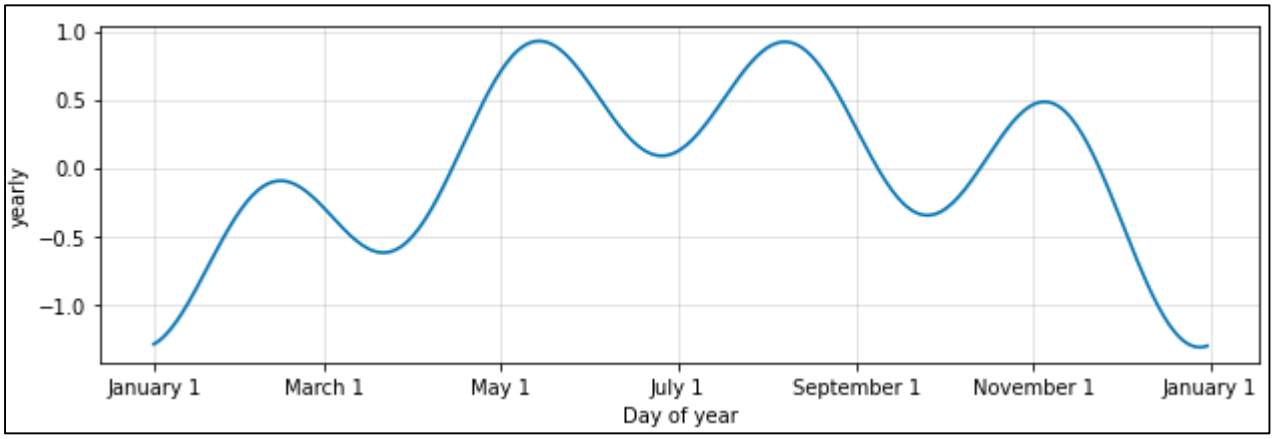
					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 2	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.					Графік передбачення моделі Prophet	2ІСТ-19м		
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Тенденція часового ряду моделі Prophet



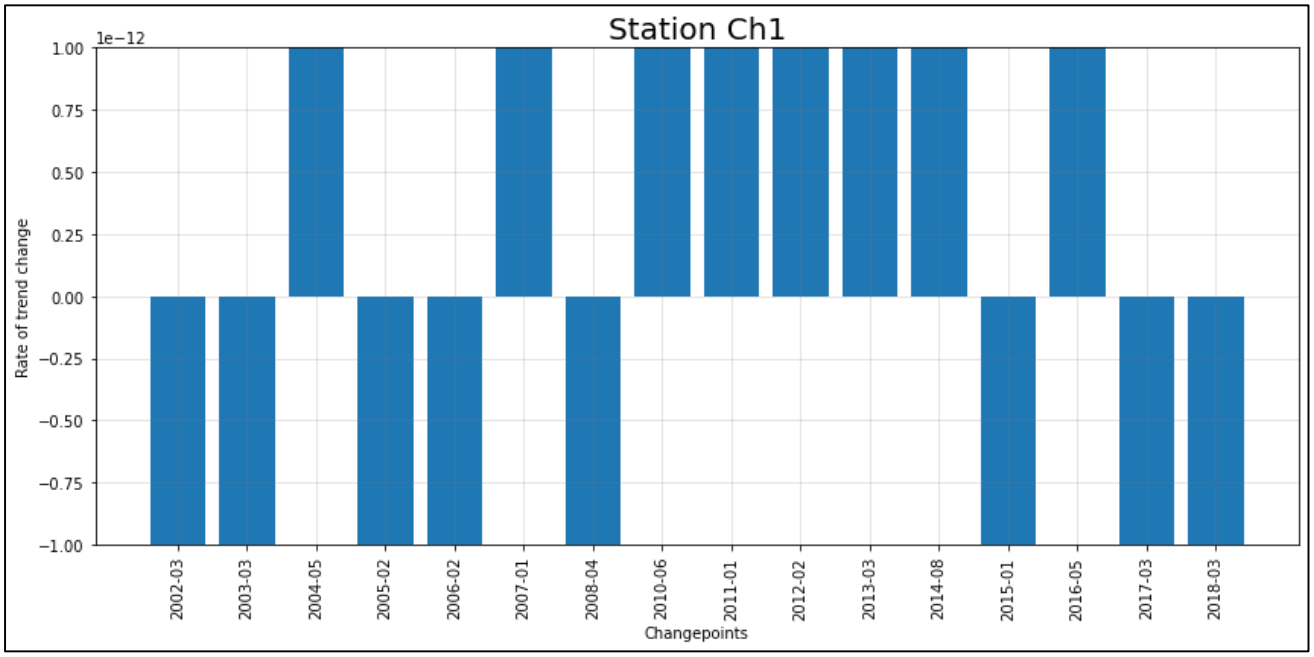
					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 3	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.					Тенденція часового ряду моделі Prophet	2ІСТ-19м		
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Річна сезонність часового ряду моделі Prophet



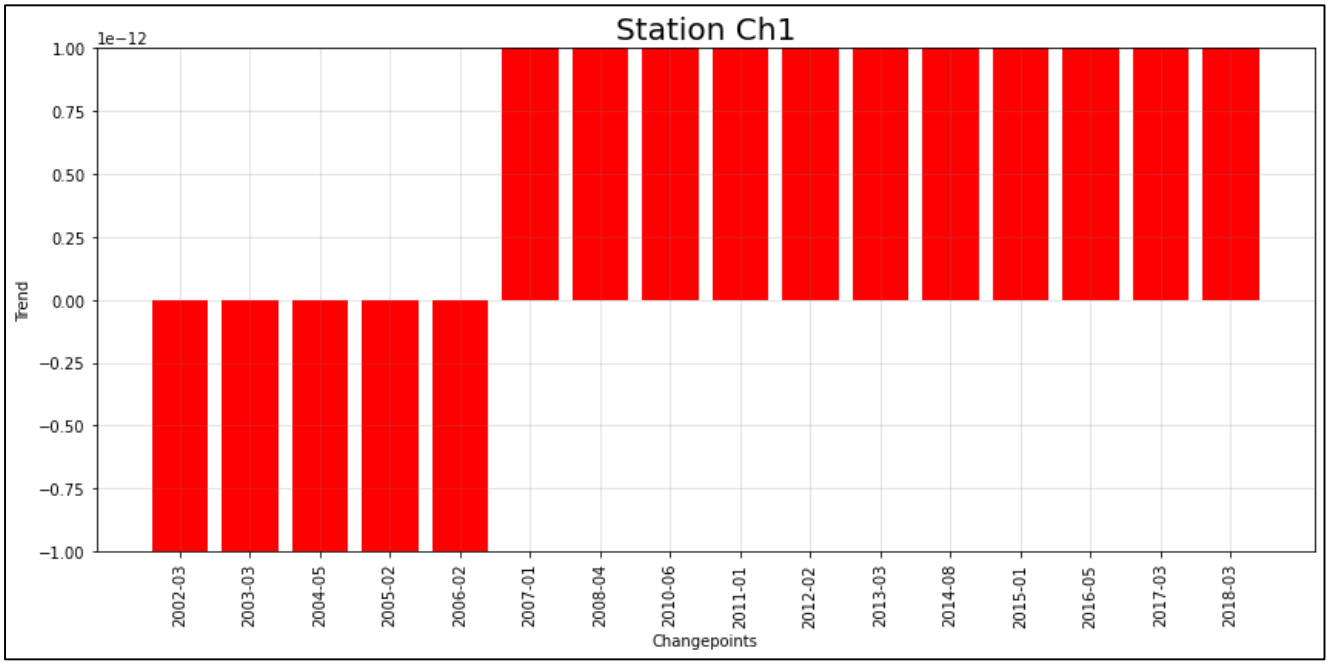
					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 4	Аркушів 8	
Перевір.		Мокін В.Б.			Річна сезонність часового ряду моделі Prophet	2ІСТ-19м		
Т. Контр.								
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Графік дельти передбачення, побудованого моделлю Prophet



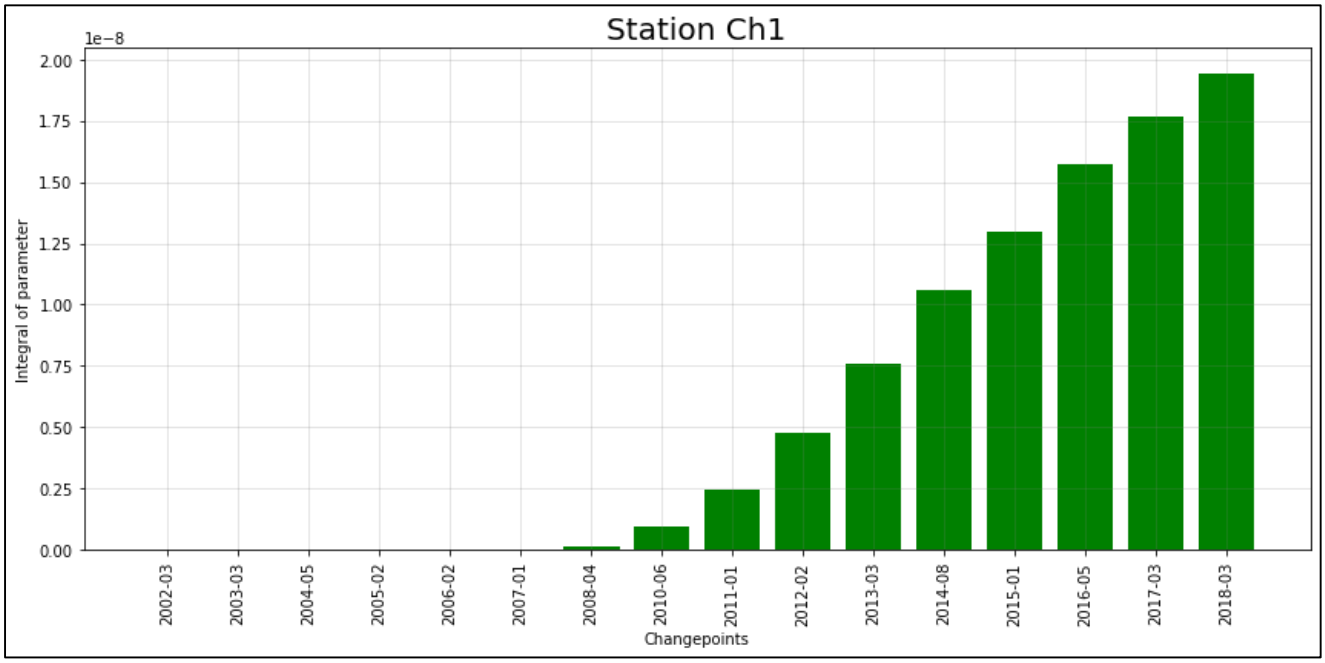
					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 5	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.					Графік дельти передбачення, побудованого моделлю Prophet	2ІСТ-19м		
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Графік тренду передбачення, побудованого моделлю Prophet



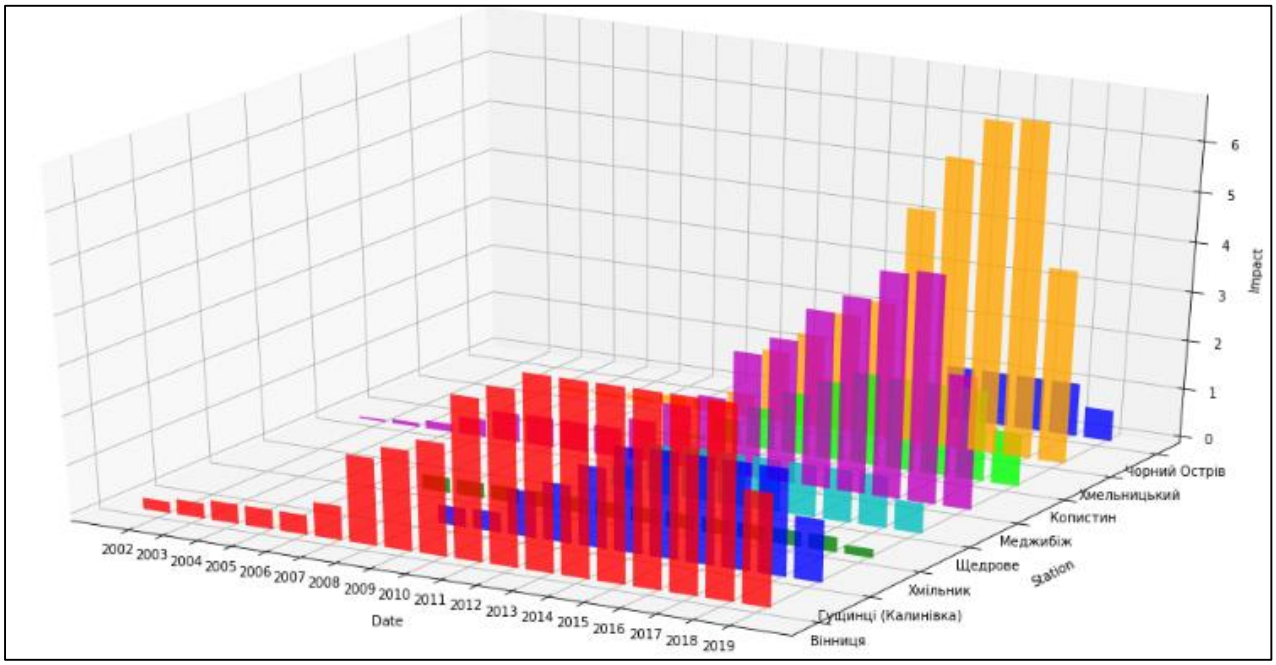
					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 6	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.					Графік тренду передбачення, побудованого моделлю Prophet	2ІСТ-19м		
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Графік інтегралу значень передбачення, побудованого моделлю Prophet



					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 7	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.								
Реценз.		Квєтний Р.Н.			Графік інтегралу значень передбачення, побудованого моделлю Prophet	2ІСТ-19м		
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						

Результат застосування інформаційної технології на реальних даних



					08–53.МКР.007.02.00.ПЛ		
					Інформаційна технологія пошуку можливих джерел підвищеного забруднення річки з використанням моделі Prophet	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата				1 : 1
Розроб.		Шмундяк Д.О.				Арк. 8	Аркушів 8	
Перевір.		Мокін В.Б.						
Т. Контр.					Результат застосування інформаційної технології на реальних даних	2ІСТ-19м		
Реценз.		Квєтний Р.Н.						
Н. Контр.		Жуков С.О.						
Затверд.		Мокін В.Б.						