

Прикарпатський національний університет імені Василя Стефаника

Факультет математики та інформатики

Кафедра комп'ютерних наук та інформаційних систем

ВИПУСКНА МАГІСТЕРСЬКА РОБОТА

Тема: «Прогнозування для розв'язання задач ефективного виробництва енергії на сонячних електростанціях»

Виконав: студент 2 курсу

гр. КНМ - 21

ОР магістр

Драбчук Павло Богданович

Спеціальність: 122 Комп'ютерні науки

Науковий керівник: професор, доктор
технічних наук Петришин Л. Б.

Рецензент:

Івано-Франківськ, 2022

АНОТАЦІЯ

Пояснювальна записка: 79 сторінок, 3 розділи, 10 таблиць, 27 рисунків, 6 додатків, 50 джерел.

Ключові слова: МЕТОД ПРОГНОЗУВАННЯ, МОДЕЛЬ ПРОГНОЗУВАННЯ, НЕБАЛАНСИ, ПРОГНОЗУВАННЯ, СОНЯЧНА ЕЛЕКТРОСТАНЦІЯ.

Об'єкт дослідження – методи прогнозування генерації енергії на сонячних електростанціях з можливістю визначення її кількості на певний період часу.

Мета роботи – створення авторського механізму прогнозування виробництва електроенергії на сонячних фермах на основі застосування методу Random Forest для розв'язання задач, пов'язаних з проблемою подання неточних прогнозів виробниками ВДЕ.

Методи дослідження – аналіз наукової літератури та законів, які регулюють ринок енергії, проведення порівняльного аналізу відомих методів, створення теоретичної моделі механізму прогнозування та його програмна реалізація.

Досліджено предметну область, встановлено критерії відшкодування за небаланси, проведено аналіз відомих методів прогнозування.

Розроблено теоретичну модель власного механізму прогнозування, а саме встановлено вимоги до вхідних та вихідних даних, обґрунтовано рішення та проаналізовано проблеми, що впливають на точність методу. Крім цього, створено діаграму станів алгоритму та визначено засоби для програмної реалізації.

На основі результатів, розроблено метод прогнозування за попередньо розробленою діаграмою з дотриманням встановлених вимог. Для унаочнення практичного застосування, створено систему, за допомогою якої, власник станції чи аналітик може створювати моделі прогнозування та на їхній основі формувати прогноз та оформляти звіт, який можна автоматично надіслати за призначенням.

Використання створеного механізму прогнозування дозволить власникам сонячних електростанцій мінімізувати розмір економічних втрат і тим самим заощадити кошти, які можна використати для модернізації чи ремонту обладнання.

ЗМІСТ

ПЕРЕЛІКИ СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ.....	6
ВСТУП.....	7
1 ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОСОБЛИВОСТЕЙ ПРЕДМЕТНОЇ ОБЛАСТІ ...	10
1.1 Аналіз предметної області.....	10
1.2 Критерії оцінювання методів.....	13
1.3 Аналіз та порівняння методів прогнозування.....	16
1.3.1 Метод Total Sky Imagers.....	17
1.3.2 Методи на основі Numerical Weather Prediction Models.....	18
1.3.3 Комбіновані методи.....	18
1.3.4 Висновки по методах прогнозування.....	19
1.4 Постановка задачі для вирішення проблеми.....	20
2 ТЕОРЕТИЧНА МОДЕЛЬ МЕТОДУ ПРОГНОЗУВАННЯ.....	23
2.1 Аналіз та обґрунтування методу прогнозування.....	23
2.1.1 Встановлення вимог до вхідних та вихідних даних.....	23
2.1.2 Обґрунтування рішень для методу прогнозування.....	25
2.1.3 Аналіз проблем, які впливають на точність методу.....	32
2.1.4 Процес прогнозування енергії та збереження моделі.....	33
2.2 Структура алгоритму.....	35
2.3 Засоби для розробки методу та системи.....	50
3 ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДУ ПРОГНОЗУВАННЯ.....	54
3.1 Реалізація алгоритму.....	54
3.2 Практичне застосування методу.....	62
3.3 Визначення економічної ефективності застосування.....	74
ВИСНОВКИ.....	78
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	80
ДОДАТОК А Програмний код для побудови моделей та обчислення критеріїв якості алгоритмів.....	86
ДОДАТОК Б Показники ефективності моделей, які сформовано для вибірки різних розмірів для кожної з груп.....	89

ДОДАТОК В Діаграми складених станів алгоритму прогнозування виробництва сонячної електроенергії.....	90
ДОДАТОК Г Алгоритмічні моделі ключових алгоритмів	92
ДОДАТОК Д Програмний код для завантаження даних про погоду за допомогою сервісу Open-Meteo та формування вихідної структури	95
ДОДАТОК Е Дані для обґрунтування економічної ефективності.....	96

ПЕРЕЛІКИ СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

ВДЕ	Відновлювана енергетика
СЕС	Сонячна електростанція
АЕС	Атомна електростанція
ТЕС	Теплова електростанція
NWP	Numerical Weather Prediction
RMSE	Root-mean-square error
MAE	Mean absolute error
R ²	R-squared, coefficient of determination
API	Application programming interface
БД	База даних
CSV	Comma-Separated values
SQL	Structured query language

ВСТУП

Об'єкт дослідження

Методи прогнозування продукування енергії на сонячних електростанціях з можливістю визначення її кількості на певний період часу.

Стан проблеми

Небаланси в енергетичній системі, спричинені через подання неточних прогнозів про виробництво електроенергії виробниками енергії на сонячних електростанціях, призводять до нарахування штрафних санкцій, які завдають фінансових збитків для виробників альтернативної енергії. З іншого боку, велика кількість неправильно поданих прогнозів негативно впливає на навантаження мережі та на виробництво енергії на електростанціях, таких як АЕС чи ТЕС.

Актуальність

Прогнозування кількості енергії на сонячних електростанціях дозволить оптимізувати роботу електроенергетичної системи, а також через запровадження відповідальності за небаланси для ВДЕ, зацікавленість до систем прогнозування виробництва електроенергії буде постійно зростати і спонукати виробників інвестувати в такі системи.

Мета

Метою роботи є створення методу прогнозування виробництва енергії на сонячних електростанціях на основі застосування методу Random Forest для розв'язання задач, пов'язаних з проблемою подання неточних прогнозів виробниками ВДЕ.

Методи дослідження

Аналіз предметної області, актуальних джерел та способів вирішення проблеми, дослідження та порівняння існуючих методів прогнозування, створення методу прогнозування, формування моделі та оцінка її ефективності.

Новизна

Результат даної розробки буде розміщено на відкритому ресурсі у мережі Інтернет, щоб власники сонячних електростанції могли безкоштовно випробувати

метод прогнозування на власних даних та надати зворотній зв'язок, щодо якості моделей.

Практична значимість

Методи прогнозування сонячної енергії можна застосовувати у галузях електроенергетики, а також беручи до уваги те, що основним завданням методу є прогнозування сонячного випромінювання, то отриману інформацію можна застосовувати у сільському господарстві для визначення сприятливих умов для посіву чи збору певних культур.

Завдання

Розробити метод прогнозування виробництва енергії на сонячних електростанціях та обґрунтувати його ефективність та доцільність застосування.

Економічна ефективність роботи

Мінімізація похибки прогнозування, шляхом застосування нових та вдосконалених методів прогнозування дозволить виробникам електроенергії на сонячних електростанціях уникнути штрафних санкцій, які негативно впливають на дохід компанії.

Інформаційна база

Під час виконання аналітичної та методичної частини роботи використовувались наукові статті, публікації, книги, які належать до досліджуваної тематики, а також закони та постанови, які регулюють ринок енергетики, звіти ДП «Гарантований покупець». На етапі моделювання діаграми станів використано актуальну версію офіційної документації UML. Для виконання практичної реалізації, застосовано матеріал, який наведено в документації обраних технологій розробки, до яких відносяться (Python, Pandas, NumPy, PyQt, PySide, MySQL тощо), а також використано API погодних сервісів OpenWeatherMap та Open-Meteo, які безкоштовно надають необхідні дані про погоду для подальшого передбачення генерації електроенергії. На етапі обґрунтування економічної ефективності взято до уваги постанову НКРЕКП від 30.09.2022, де наведено актуальні значення «зеленого» тарифу.

Апробація результатів

Робота апробована на наступних конференціях:

- міжнародна науково-технічна конференція студентів, аспірантів та молодих вчених «Комп'ютерні науки, інформаційні технології та системи управління»;
- 63 конференція студентських наукових кіл АГН. Конференція проходила 8 грудня 2022 року на базі НТУ «Гірничо-металургійна академія імені Станіслава Сташіца»;
- звітна наукова веб-конференція викладачів, докторантів, аспірантів та студентів університету за 2021 рік. Конференція проходила 07.04.2022 у Прикарпатському національному університеті імені Василя Стефаника;
- 62 конференція студентських наукових кіл АГН. Конференція проходила 9 грудня 2021 року на базі НТУ «Гірничо-металургійна академія імені Станіслава Сташіца».

Висновки

В першому розділі здійснено аналіз предметної області, а саме тенденції розвитку виробництва енергії на сонячних електростанціях, що спричиняє необхідність у прогнозуванні кількості виробленої енергії. Крім цього, досліджено критерії оцінювання моделей прогнозування, які застосовано для порівняння відомих методів прогнозування.

В другому розділі встановлено вимоги до вхідних та вихідних даних, сформовано теоретичну модель методу та механізму прогнозування, а також обґрунтування його ефективності. Для демонстрації поведінки алгоритму, створено діаграму станів та описано засоби для його програмної реалізації.

В третьому розділі наведено інформацію про реалізацію методу, зосереджено увагу на важливих частинах коду, а також практичне застосування. Крім цього, для підтвердження необхідності практичного застосування, розроблено систему, яка дозволяє здійснювати прогнозування з використанням власного алгоритму та інші функції. На завершення, визначено економічну ефективність застосування методів прогнозування кількості виробленої енергії на сонячних електростанціях.

1 ДОСЛІДЖЕННЯ ТА АНАЛІЗ ОСОБЛИВОСТЕЙ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної області

Практичне прогнозування – це одночасно наука та мистецтво. Наука полягає в правильному використанні складних статистичних інструментів, які незмінно підвищать точність прогнозу. Суть мистецтва в тому, що емпіричні дані зрідка дають однозначну відповідь, тому аналітик повинен робити свій вибір між альтернативними рішеннями, щоб обрати моделі, яка забезпечить найбільш точні прогнози [1, с. 3]. Збирання інформації з метою її подальшого використання відбувалось досить давно. Наприклад, перша спроба прокладання першого міжконтинентального кабелю для телеграфу була у серпні 1857 року через те, що за спостереженням мореплавців у цей період океан був найспокійніший [2].

Прогнозування використовується для багатьох цілей у різних сферах. Важливу роль воно відіграє саме в енергетичній галузі для забезпечення балансу в електроенергетичній системі, адже від неї залежить стабільність роботи промислових об'єктів та споживання електрики приватними домогосподарствами.

В Україні для забезпечення потреб соціально-економічного розвитку виробляється та споживається велика кількість електроенергії. Відповідно до інформації Міністерства енергетики виробництво та споживання електричної енергії зросло на 5-6% порівняно з 2020 роком [3]. Велику нішу у продукуванні енергії займають сонячні електростанції.

Географічне розташування України та сприятливий клімат слугує розвитку сонячної електроенергетики. Крім цього, активному розвитку сфери сприяло впровадження зеленого тарифу для підприємств на відновлюваних джерелах енергії (ВДЕ) у 2009 році. Також держава гарантувала 100% закупівлі зеленої енергії та деякі податкові пільги. Через п'ять років зелений тариф, який був найвищим серед європейських країн, поширився на побутові сонячні електростанції, у яких потужність становила до 30 кВт [4, с. 24]. Велику роль відіграє модернізація та покращення технологій фотоелектричних елементів, що значно збільшило їхню ефективність. На рисунку 1.1 наведено діаграму з

інформацією про встановлену потужність об'єктів відновлюваної електроенергетики станом на 01.07.2021 року Державного агентства з енергоефективності та енергозбереження України. Проаналізувавши отримані дані, видно, що кількість сонячних електростанцій постійно зростає та складає приблизно 79% від потужності усіх об'єктів відновлюваної електроенергетики.

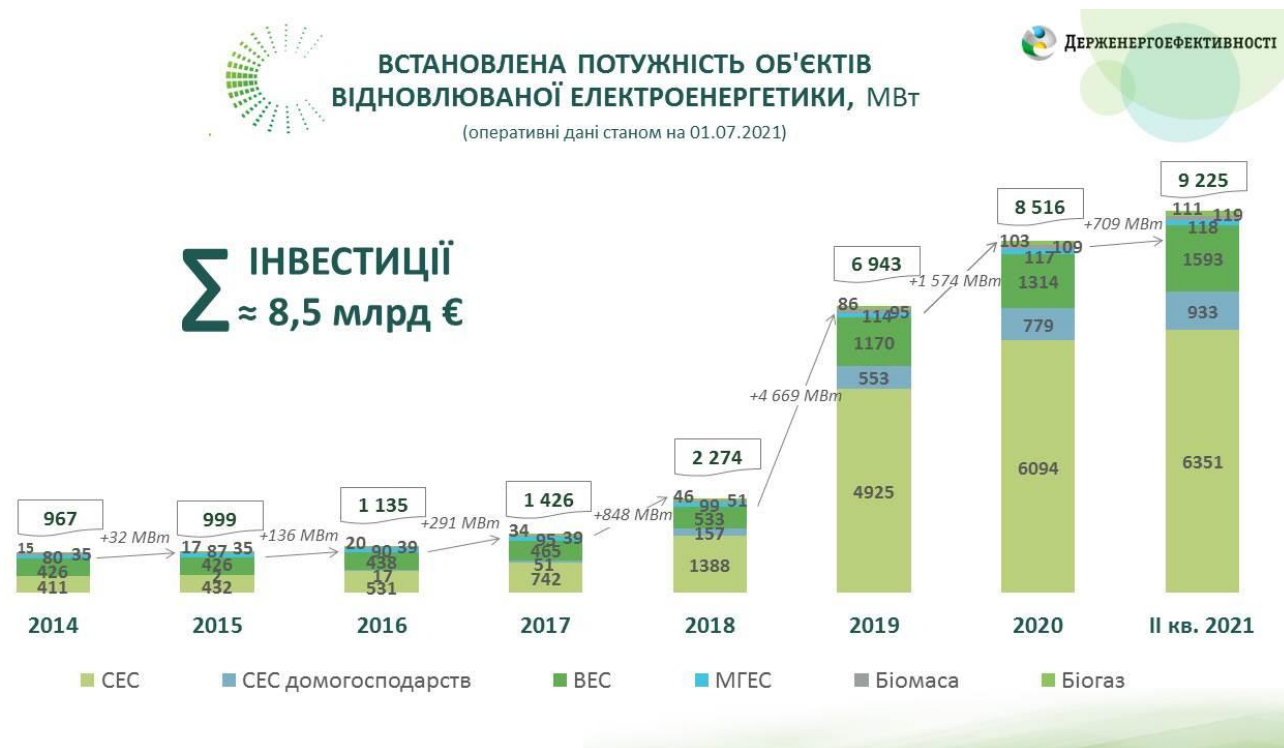


Рисунок 1.1 – Зростання відновлювальних джерел за 2014 - II квартал 2021 рр. [5]

Різне збільшення популярності встановлення сонячних електростанцій призвело до значної кількості небалансів у електроенергетичній системі. Через це електромережі працюють нестабільно та досягають нерівномірного навантаження. Щоб вирішити дану проблему в Україні, прийнято рішення, що виробники ВДЕ повинні подавати інформацію про прогнози виробництва електроенергії до ДП «Гарантований покупець», яке несло відповідальність за небаланси. Проте починаючи з січня 2021 р. виробники «зеленої енергії» повинні подавати власні прогнози генерування енергії, та у разі похибки більшої ніж 5% будуть нести відповідальність та отримувати рахунки (штрафи) за різницю між згенерованою електроенергією та прогнозованою [6].

рівняннях, які описують перетворення в середовищі на необхідний інтервал часу (переважно на декілька днів вперед) [9, с. 25-28].

Отже, створення власного методу прогнозування дасть змогу дослідити наявні та визначити їхні особливості, що дозволить їх порівнювати для досягнення максимальної ефективності алгоритму.

1.2 Критерії оцінювання методів

Прогнозування згенерованої кількості електроенергії на сонячних електростанціях є важливим етапом під час її виробництва. Складність спричинена великою кількістю варіантів погодних умов на території ферми та різноманітністю систем для продукування енергії. Неточність прогнозу може вплинути на надійність електроенергетичної системи та мережі, а також завдати фінансових збитків, які спричинені отриманням штрафів. Тому перед аналізом методів прогнозування варто визначити критерії для їхнього оцінювання [8, с. 38].

Під час створення моделей прогнозування дослідники застосовують різноманітні методи оцінки ефективності. Незважаючи на те, що регресійний аналіз займає значну частину сфери машинного навчання та обчислювальної статистики, консенсусу щодо єдиного критерію оцінки моделей ще не досягнуто, тому на цьому етапі я дослідив кілька з них, які зазвичай використовуються для оцінки результатів прогнозування в досліджуваній галузі.

Використання різних метрик дозволяє виявити особливості кожної моделі, шляхом аналізу отриманих показників. Переважно для вимірювання точності методу прогнозування використовують наступні показники: RMSE, MAE, R^2 .

Коефіцієнт середньоквадратичного відхилення (RMSE) забезпечує оцінку моделі протягом усього періоду прогнозування. Моделі прогнозування будуються з використанням доступної інформації: даних про попередні вимірювання, пояснень кожного параметру тощо. Очевидно, що з урахуванням різних недоліків створені прогнози будуть недосконалими, тому для оцінки відхилення результатів використовується цей критерій [10]. Його обчислюють за формулою (1.1), яку наведено нижче.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{p}_i - p_i)^2}, \quad (1.1)$$

де N – кількість точок, у періоді прогнозування;

\hat{p}_i – прогнозоване значення на i -му кроці часу;

p_i – фактичне значення на i -му кроці часу [8, с. 39].

Значення цього параметру є в межах діапазону від 0 до нескінченності, де нуль вказує на ідеальний випадок, тобто дані співпадають на 100%. Використання RMSE є дуже поширеним і вважається чудовим показником для обчислення помилки для числових прогнозів. Враховуючи його особливості, він підходить для порівняння точності різних моделей або конфігурацій однієї моделі з вдосконаленням конкретної змінної, але не з усіма, так як середньоквадратичне відхилення залежить від масштабу [11].

Показник середньої абсолютної похибки (MAE) вимірює середнє значення величини абсолютних помилок у наборі прогнозів, не враховуючи їхнього напрямку. Цей показник, як і попередній, використовується в багатьох напрямках: машинне навчання, аналіз часових рядів, біологічні науки, атмосферні науки тощо. Вони використовуються на етапі формування моделі (вибір оптимальних параметрів, порівняння та вибір моделі) [12]. Обчислюється за формулою (1.2), яку наведено нижче.

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{p}_i - p_i|, \quad (1.2)$$

де N – кількість точок, у періоді прогнозування;

\hat{p}_i – прогнозоване значення на i -му кроці часу;

p_i – фактичне значення на i -му кроці часу [8, с. 39].

Значення MAE може коливатись від 0 до нескінченності, крім того більше значення величини вказує на більшу дисперсію для деяких величин у вибірці. На відміну від RMSE, зміни в даному критерії є лінійними і тому інтуїтивно

зрозумілими. Крім того, аналізуючи формулу (1.2) помітно, що помилкам не надається більша вага за рахунок піднесення до квадрату, як у RMSE, а вони збільшуються лінійно [13]. Але тільки двох критеріїв RMSE та MAE для оцінки моделей недостатньо, тому варто використати ще один показник.

Статистичний коефіцієнт детермінації (R^2) показує частку дисперсії залежної змінної, значення якої передбачається на основі незалежної змінної. Розрахунок даного критерію складається з кількох кроків. Для цього потрібно взяти ряд спостережень, створити для них модель, використовуючи яку потрібно обчислити прогнозовані значення. Далі, застосовуючи формулу (1.3) потрібно розрахувати значення коефіцієнта детермінації [14].

$$R^2 = \frac{\sum (\hat{y}_i - \bar{y})^2}{\sum (y_i - \bar{y})^2}, \quad (1.3)$$

де y_i – фактичне значення на i -му кроці часу,

\hat{y}_i – прогнозоване значення на i -му кроці часу,

\bar{y} – середнє значення на i -му кроці часу,

N – кількість точок, у періоді прогнозування [15, с. 151].

Значення коефіцієнта детермінації варіюється в межах від 0 до 1, де одиниця вказує на ідеальну відповідність. Незважаючи на вказаний діапазон значень даного критерія, в окремих випадках (використовуючи інші формули для обчислення) він може дорівнювати від'ємним значенням, що буде вказувати на погано створену модель, а з іншого боку неможливо буде визначити наскільки ця модель є поганою. Тому, значення коефіцієнта в діапазоні $[0; 1]$ чітко інформує фахівця про продуктивність моделі [16, с. 17].

Обчислення значень цих критеріїв дозволяє оцінити точність та порівняти якість прогнозування, застосовуючи моделі з різними параметрами. Для уникнення недостовірних результатів, аналітик перед розрахунком цих коефіцієнтів повинен переконатись у репрезентативності вибірки. Також на результат прогнозування впливає точність прогнозу погоди та варіативність прогнозів хмарності.

1.3 Аналіз та порівняння методів прогнозування

Методи прогнозування вироблення енергії на сонячних електростанціях загалом розподіляють на три категорії:

- фізичні методи;
- статистичні методи;
- гібридні методи [8, с. 29].

Для здійснення якісного аналізу існуючих методів прогнозування вироблення енергії на сонячних електростанціях, розглянуто метод Total Sky Imagers та методи на основі Numerical Prediction Models. Обчислення ефективності алгоритмів, які використовуються для прогнозування буде проводитись з однаковим для всіх набором даних, параметри якого наведено у табл. 1.1. Множина містить інформацію про дату та час, погодні умови та кількість виробленої енергії.

Таблиця 1.1 Значення параметрів у навчальній та тестовій množинах

Вибірка	Параметр	Значення
Навчальна множина	Кількість елементів	3261
	Атмосферний тиск, мм. рт. ст.	730,5 – 770,8
	Сонячна радіація, Вт/м ²	0 – 997
	Температура, °С	-0,9 – 30,9
	Вологість, %	34 – 97
	Швидкість вітру, м/с	0 – 10
	Хмарність, %	0 – 100
Тестова множина	Кількість елементів	1087
	Атмосферний тиск, мм. рт. ст.	730,7 – 770,5
	Сонячна радіація, Вт/м ²	0 – 999
	Температура, °С	-1,2 – 28,1
	Вологість, %	37 – 97
	Швидкість вітру, м/с	0 – 10
	Хмарність, %	0 – 100

Перед застосуванням вибірки проведено попередню обробку даних. На цьому етапі підготовлено набір, який відповідає вимогам завдання. Його створено на основі відкритих даних, які зібрано в рамках проекту UK Power Networks [17], а саме обрано набір для сонячної електростанції, яка розташована в районі «УМСА». Його перевагою є наявність власної метеостанції, що дозволяє зберігати якомога точніші дані про погодні умови. Для отримання необхідної вибірки, з неї видалено повторювані екземпляри та порожні рядки, а також вибрано погодинні дані за період 01.03.2014 р. – 31.08.2014 р. Крім цього, видалено колонки з параметрами, які мають низький вплив на якість прогнозу та додано відсутній параметр «Хмарність». Значення щільності хмар для визначеного періоду та розташування отримано з використанням сервісу «Open-Meteo», який безкоштовно надає дані про погоду, зокрема історичні [18]. Також, отриманий набір має певну особливість – відсутність опадів за обраний період. Це дозволило не враховувати цей параметр при створенні моделей, використовуючи вибрані методи, щоб зосередити увагу саме на якості моделей для даних з мінімальною кількістю параметрів.

Далі наведено детальний опис та аналіз вищеназваних методів.

1.3.1 Метод Total Sky Imagers

Метод Total Sky Imagers застосовується для створення моделей на основі супутникових та хмарних зображень. Супутникові знімки хмарності з високою просторовою роздільною здатністю, використовуються для отримання необхідної інформації про рух хмар. Ці дані дають змогу визначити розташування хмар, яке застосовується для прогнозування сонячного випромінювання. До параметрів, які мають найбільший вплив на нього є хмарність та оптична глибина хмар. Обробка даних знімків дозволяє охарактеризувати хмари та виявити їхню мінливість. Розташування та структура хмар дозволяє передбачити сонячне випромінювання на певний період часу для необхідної ділянки, від якого залежить кількість виготовленої електроенергії [8, с. 32].

Похибка даного методу зростає з збільшенням площі, для якої він застосовується. Загальна середня похибка між прогнозованими та спостережуваними значеннями варіюється від 6% до 30%, при прогнозуванні на від 30 секунд до 5 хвилин відповідно [19, с. 4].

1.3.2 Методи на основі Numerical Weather Prediction Models

Методи, які використовують числові моделі прогнозування погоди (NWP), залежать від змін в атмосфері. Вони накопичують інформацію про поточні спостереження за погодою з метою обробки для подальшого прогнозування майбутніх станів погоди [8, с. 33].

Такі методи дозволяють здійснити прогнозування сонячної енергії на різних рівнях деталізації:

- від 1 до 3 годин;
- від 48 до 180 годин;
- довше 180 годин [19, с. 3].

Ефективність NWP як джерела даних для прогнозу сонячної радіації дуже мінлива і для деяких моделей може змінюватись на 200%.

Похибка методів прогнозування вироблення сонячної енергії, які використовують моделі NWP залежить від якості прогнозу погоди. Тому, для покращення результатів створюють регіональні моделі прогнозування погоди, щоб зменшити масштаб дослідження та збору даних [8, с. 32].

1.3.3 Комбіновані методи

Комбіновані (гібридні) методи – це комбінація двох або більше методів прогнозування для підвищення точності прогнозу. Ідея використання гібридних моделей полягає в тому, щоб подолати недоліки окремих моделей і використати переваги інших, об'єднати їх разом і створити нову гібридну модель для зменшення помилок прогнозу. Гібридні моделі можуть поєднувати лінійні моделі, нелінійні моделі або як лінійні, так і нелінійні моделі. Багато досліджень показали, що такі методи прогнозування перевершують індивідуальний прогноз [8, с. 38].

1.3.4 Висновки по методах прогнозування

Отже, проаналізувавши дані методи, кожен метод має свої особливості та кращий для вирішення певних задач. Комбіновані методи прогнозування дозволяють підвищити якість результату прогнозу, в порівнянні з іншими методами. В основу таких методів лягають різні алгоритми машинного навчання, найпоширенішими з яких є лінійна регресія [20, с. 7601-7603] та метод опорних векторів [21, с. 1-9]. Крім цього варто додати до порівняння один з ансамблевих методів – Випадковий ліс.

Для дослідження кожного алгоритму створено програмний код на мові Python, де реалізовано функції для опрацювання набору даних, побудови моделей з використанням бібліотеки scikit-learn [22], обчислення критеріїв точності та часу виконання, який наведено у додатку А. Для оцінки та порівняння вищеописаних методів усереднені значення отриманих критеріїв наведено у табл. 1.2.

Таблиця 1.2 Критерії якості порівнюваних алгоритмів

Критерії	Лінійна регресія	Метод опорних векторів	Випадковий ліс
RMSE, Вт	1,1557	1,7204	0,8613
MAE, Вт	0,7035	1,2108	0,4027
R^2	0,5597	0,0242	0,7554
Час побудови моделі, с	0,02	2,74	4,26

Отримані показники у вигляді таблиці дозволяють порівняти значення та визначити переваги та недоліки кожної моделі. Найкраще значення середньоквадратичної помилки досягнуто при використанні алгоритму випадкового лісу. Такого ж результату досягнуто для середньої абсолютної помилки. Коефіцієнт детермінації вказує на сильний зв'язок, якщо він має значення, яке дорівнює 1. Воно вказує на найвищу точність прогнозування серед наведених показників. Для наведеного набору даних та алгоритмів найменше значення спостерігається у методу опорних векторів, а найбільше у алгоритму

випадкових лісів. Але значення 0,75 не є високою оцінкою, тому є багато можливостей для вдосконалення даного методу. Час формування моделей також є важливим параметром. Проаналізувавши результати очевидно, що найшвидшим є алгоритм лінійної регресії, а найповільнішим алгоритм випадкових лісів.

Отже, аналізуючи отримані результати видно, що найточнішим є метод випадкових лісів. Час побудови моделі не є основним фактором для вирішення завдання, де однією з головних умов є мінімізувати похибку прогнозування, щоб запобігти фінансових втрат, які завдані отриманням штрафів. Крім того, тривалий час можна скоротити розпаралеленням процесу побудови моделі. Тому цей алгоритм, незважаючи на його швидкодію, варто застосувати основним при побудові власного комбінованого методу прогнозування.

1.4 Постановка задачі для вирішення проблеми

Головною метою роботи є вирішення проблеми небалансів в електромережі, пов'язаної з подачею виробниками власних прогнозів, шляхом створення власного комбінованого методу прогнозування. Оскільки це гібридний метод, то потрібно базовий метод вдосконалити, а також реалізувати новий механізм прогнозування, який дозволить значно покращити точність прогнозу.

Процес розробки авторського методу прогнозування повинен описуватись чітким планом, в якому описано основні кроки. Це дозволяє сформулювати загальне розуміння необхідної роботи. До того ж це дає змогу здійснювати коригування дій на кожному з етапів, у випадку виявлення проблем. Тому, для успішного вирішення проблеми необхідно виконати ряд кроків, а саме:

- провести аналіз обраного базового методу та визначити його недоліки;
- визначити та обґрунтувати можливості вдосконалення роботи опорного методу;
- розробити новий механізм прогнозування;
- обчислити показники ефективності прогнозування, використовуючи власний алгоритм та проаналізувати отримані значення;

- визначити проблеми, які можуть виникнути під час прогнозування та розробити методи для їх вирішення;
- встановити вимоги до вхідних та вихідних даних власного методу прогнозування;
- здійснити моделювання структури алгоритму з використанням діаграм UML;
- провести аналіз структури та особливостей механізму прогнозування та на його основі визначити ряд технічних засобів для програмної реалізації;
- здійснити програмну реалізацію та провести тестування на підготовлених наборах даних;
- дослідити зміни якості сформованих моделей для наборів з різним об'ємом даних;
- розробити інформаційну систему для формування прогнозу виробництва електроенергії на сонячних електростанціях з використанням власного методу прогнозування;
- здійснити тестування та виправлення помилок у роботі системи;
- обґрунтувати економічну ефективність застосування розробленого механізму прогнозування.

Кожен етап має чітку назву, яка свідчить про його завдання, але варто виокремити кілька пунктів, які можна описати детальніше. Це стосується кроків, які пов'язані з описом вхідних та вихідних даних, моделюванням алгоритму та розробкою інформаційної системи.

На етапі встановлення вимог до вхідних та вихідних даних потрібно визначити, які саме структури варто застосувати для значень. Основними вхідними для методу прогнозування є:

- вибірка з історичною інформацією про кількість згенерованої енергії та умови продукування;
- відомості про сонячну електростанцію;
- параметри моделі прогнозування.

В результаті мають бути сформованими наступні вихідні дані:

- модель прогнозування;
- характеристика сформованої моделі;
- інформація про прогнозоване значення виробленої енергії станом на наступну добу.

Етап моделювання передбачає створення діаграми станів для опису структури алгоритму. Цей тип діаграми дає можливість візуально відобразити переходи до конкретного стану, в якому перебуває об'єкт. Модель алгоритму дозволить визначити частини методу, де потрібно провести корегування перед програмною реалізацією.

Одним з важливих етапів дослідження є розробка інформаційної системи для прогнозування. Для неї можна виділити перелік особливостей, які повинні бути в ній реалізованими, а саме:

- взаємодія з інформацією про сонячну електростанцію;
- моніторинг виробництва та прогнозування електроенергії;
- створення моделей прогнозування з можливістю їхнього збереження;
- формування прогнозу виробництва на наступну добу;
- генерування звітів.

Отже, аналіз предметної області демонструє кілька основних факторів, які підтверджують необхідність даного дослідження. По-перше, постійне вдосконалення сонячних батарей робить їх більш привабливим для створення сонячних електростанцій, що буде породжувати їхнє збільшення. По-друге, збільшення штрафних нарахувань з кожним роком, буде спонукати власників електростанцій використовувати системи для прогнозування власного виробництва та інвестувати кошти в них.

Перед розробкою авторського методу прогнозування встановлено критерії оцінювання методів та обґрунтовано їхнє використання, а також проаналізовано та здійснено порівняння кількох методів прогнозування. В результаті, сформовано ряд кроків, виконання яких дозволить вирішити поставлене завдання.

2 ТЕОРЕТИЧНА МОДЕЛЬ МЕТОДУ ПРОГНОЗУВАННЯ

2.1 Аналіз та обґрунтування методу прогнозування

Створення ідеального прогнозу є складним процесом, бо кожен з них завжди може містити помилку, тому метою створення власного методу є її мінімізація. Для виконання цієї задачі потрібно використовувати різні методи прогнозування та у багатьох випадках вони будуть доповнювати один одного [1, с. 3].

Одним з основних завдань роботи є створення ефективного методу для прогнозування виробництва електроенергії на сонячних електростанціях. Для цього процес розробки розподілено на кілька етапів, на кожному з яких досліджено можливість покращення результатів. Далі наведено покроковий детальний опис та обґрунтування власних рішень.

2.1.1 Встановлення вимог до вхідних та вихідних даних

Перед розробкою методу прогнозування необхідно визначити та встановити вимоги до вхідних та вихідних даних, які будуть опрацьовуватись методом та повертатись як результат, відповідно. Основним джерелом є вибірка з історичною інформацією про кількість виробленої енергії та умови продукування. Оскільки, це досить великі об'єми даних, то для них варто встановити вимогу, щоб вони були поданими у вигляді файлу з розширенням csv. Крім того, необхідно, щоб структура файлу передбачала колонки (табл. 2.1), які містять дані відповідних типів, а назва файлу містила в собі найменування станції, наприклад «forest_road_data.csv».

Таблиця 2.1 Опис та вимоги до вибірки даних про генерацію енергії

Назва	Найменування	Тип даних	Приклад
Дата та час	date_time	Дата та час, які відповідає формату «дд.мм.рррр гг:хх»	01.02.2022 09:00
Дата	date	Дата яка відповідає формату «дд-мм-рррр»	2022-02-01

Час	time	Час, який відповідає формату «ГГ:ХХ:СС»	09:00:00
Рік	d_y	Ціле число	2022
Порядковий номер місяця	d_m		2
Номер дня	d_d		1
Година	t_h		9
Атмосферний тиск	pressure	Дійсне число	746,16126
Кількість опадів	precipitation		1,296
Значення сонячної радіації	solar_radiation		128,51709
Температура повітря	temperature		3,4
Швидкість вітру	wind_speed		9,4
Вологість повітря	humidity	Ціле число	91
Хмарність	clouds		100
Кількість виробленої енергії	energy	Дійсне число	702,34074

Наступними вхідними даними є відомості про сонячну електростанцію, які повинні містити інформацію про назву, географічне розташування, потужність, дату встановлення та сервіс для отримання прогнозу погоди. Також до початкових даних відносяться параметри моделі прогнозування.

В результаті роботи методу та обчислень мають бути сформованими вихідні дані, такі як: модель прогнозування, характеристика сформованої моделі та інформація про прогнозоване значення виробленої енергії станом на вказаний період. Вони повинні бути у вигляді структури, яку може прочитати аналітик та комп'ютер в подальшому їх використанні.

2.1.2 Обґрунтування рішень для методу прогнозування

Аналізуючи в попередньому розділі методи прогнозування, визначено, що базовим варто використати Random Forest, тому перед його застосуванням варто проаналізувати його алгоритм. Цей метод є модифікацією бутстрепової агрегації, що вирощує велику колекцію дерев прийняття рішень з подальшим усередненням [23, с. 587]. В цій колекції кожне дерево залежить від значень вибірки, яку сформовано випадковим чином та з однаковим розподілом для всіх дерев лісу [24]. Під час вирішення значної кількості задач, продуктивність Random Forest корелює з методом підсилювання (boosting) та їх простіше навчати. Ці характеристики, обґрунтовують популярність методу, через що його часто використовують у різних інтерпретаціях [23, с. 587].

Основною ідеєю є бутстрепової агрегації є зменшення дисперсії шляхом усереднення моделей. Середнє значення B випадкових змінних, в кожної з яких дисперсія – σ^2 , матиме дисперсію $\frac{1}{B}\sigma^2$. Якщо змінні однаково розподілені з позитивною попарною кореляцією ρ , то дисперсія середнього дорівнюватиме (2.1):

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2 \quad (2.1)$$

де ρ – кореляція;

σ – дисперсія;

B – кількість дерев [23, с. 588].

Коли значення кількості дерев B збільшується, то друга частина виразу (2.1) зменшується, а перші – не змінюється, а отже розмір кореляції пар дерев, створених з використанням бутстрепової агрегації обмежує переваги усереднення [23, с. 588].

Ідея методу Random Forest полягає, щоб покращити зменшення дисперсії бутстрепової агрегації шляхом зменшення кореляції між деревами і при цьому значно не збільшуючи дисперсію. Це досягається під час вирощування дерева на основі вхідної вибірки, яку сформовано випадковим чином. Також, в процесі створення дерева рішень, варто врахувати, що на завантаженому наборі даних

перед кожним розбиттям потрібно обирати претендентів для розбиття випадковим чином $m \leq p$ з вхідних змінних. Зазвичай значення m дорівнюють \sqrt{p} або навіть менше за 1 [23, с. 589].

Після вирощування B дерев $\{T(x; \Theta_b)\}_1^B$ прогностична змінна регресійної моделі Random Forest обчислюється за формулою (2.2);

$$\hat{f}_{rf}^B = \frac{1}{B} \sum_{b=1}^B T(x; \Theta_b), \quad (2.2)$$

де \hat{f}_{rf}^B – прогнозоване значення;

B – кількість дерев у лісі;

$T(x; \Theta_b)$ – результат обчислення значення конкретним деревом;

Θ_b – b -те випадкове дерево [23, с. 589].

Отже, оскільки Θ_b характеризує b -те випадкове дерево в визначених розділених змінних на кожному вузлі та значення кінцевого вузла, очевидно, що зменшення m зменшить значення кореляції між будь-якою парою дерев в ансамблі, а отже, відповідно до (2.1), зменшить дисперсію середнього [23, с. 589].

Алгоритм методу Random Forest для регресії наведено нижче:

1. Від $b = 1$ до B :

а) Створити початкову вибірку S^* розміру N з навчальних даних;

б) Виростити дерево рішень «випадкового» лісу T_b для завантажених даних, рекурсивно повторюючи наступні кроки для кожного кінцевого вузла дерева, доки не буде досягнуто мінімального розміру вузла n_{min} ;

i. Вибрати m змінних випадковим чином з p змінних;

ii. Вибрати найкращу змінну/точку поділу серед m ;

iii. Розділити вузол на два дочірні вузли;

2. Вивести ансамбль дерев $\{T_b\}_1^B$.

Прогнозоване значення у новій точці x обчислюється за формулою (2.3).

$$\hat{f}_{rf}^B = \frac{1}{B} \sum_{b=1}^B T_b(x), \quad (2.3)$$

де \hat{f}_{rf}^B – прогнозоване значення;

B – кількість дерев у лісі;

$T_b(x)$ – результат обчислення значення конкретним деревом [23, с. 588].

Для візуалізації вищеприписаного алгоритму, на рисунку 2.1 наведено схему роботи Random Forest.

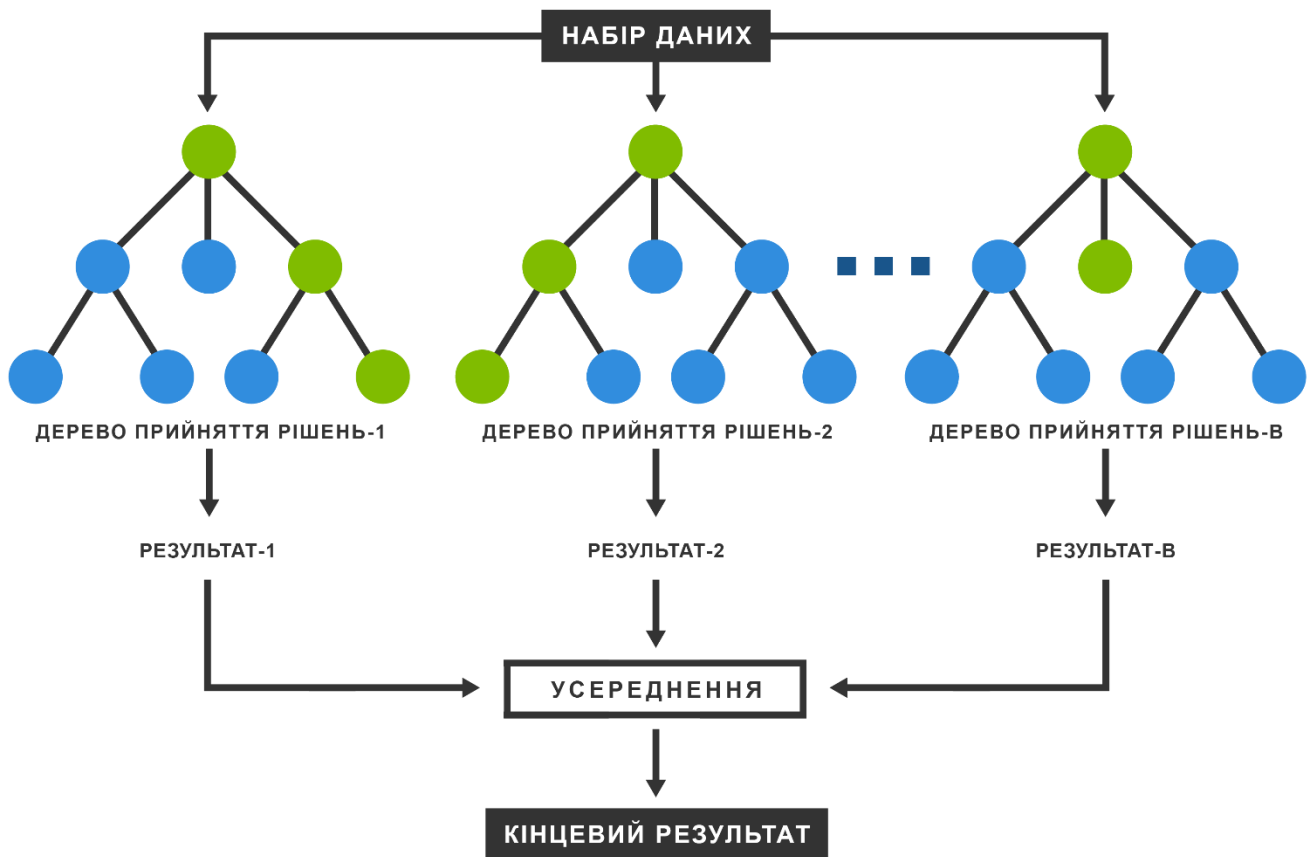


Рисунок 2.1 – Схема алгоритму Random Forest [25]

Особливістю методу Random Forest є обчислення значущості ознак набору даних. Коефіцієнт обчислюється при розбитті кожного дерева, а потім підраховується загальна кількість застосувань у кожному дереві для конкретної змінної [23, с. 593]. На рисунку 2.2 наведено гістограму, в результаті власного опрацювання, де наведено імена ознак та відсоток їхньої значущості при обчисленні прогнозованого значення. Розрахунки проведено для набору даних,

котрий містить інформацію про умови генерації електроенергії на сонячній електростанції [26].

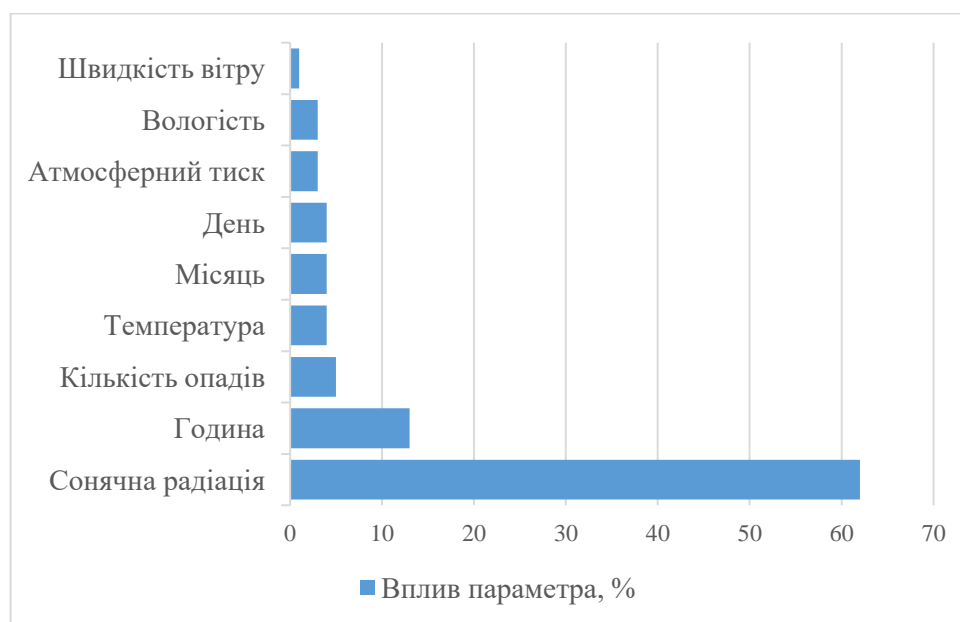


Рисунок 2.2 – Графік значень значущості ознак для лісу, вирощеного на даних про генерацію електроенергії

Проте, незважаючи на переваги в точності, моделі Random Forest можуть перенавчатись. Обмежувати цей процес можна шляхом контролю глибини окремих дерев. Ці дії дозволяють обмежити побудову правил для «зрілих» дерев для уникнення надмірної дисперсії [23, с. 596].

Авторський механізм прогнозування створено з двох складових: класифікація та регресія. Класифікація полягає у розбитті вхідних даних, які попередньо згруповані по порах року, на три групи: «Сонячна погода», «Хмарна погода» та «Погода з опадами», а регресія у побудові окремих моделей для кожної утвореної групи та загальної моделі для всього набору даних. Як було вказано раніше, для регресії обрано метод Random Forest через явну перевагу в роботі, в порівнянні з іншими методами. Розподілення даних дозволило покращити два фактори, а саме точність та середній час створення моделі. У табл. 2.2 наведено результати прогнозування для однієї і тієї ж самої вибірки даних. Підготовлений набір містить дані за одну період від 1 березня до 31 травня, тобто за одну пору року.

Таблиця 2.2 Показники ефективності прогнозування

Показник	Без групування	З групуванням		
		«Сонячна погода»	«Хмарна погода»	«Погода з опадами»
RMSE, Вт	166,348	16,3805	45,9832	112,9915
MAE, Вт	88,31	8,6924	18,7721	50,6523
R ²	0,864	0,94	0,964	0,967
Середній час створення, с	2,23	0,016	0,924	0,03

Отримані показники свідчать про позитивний вплив класифікації на результат прогнозування. Поділ спостережень на кілька класів відповідно до характеристик погоди, а потім побудова моделі прогнозування для кожної групи є важливим для покращення її якості [27, с. 4]. Але створення моделі для кожної групи не гарантує підвищення точності у всіх випадках. Утворені вибірки можуть містити невелику кількість даних, що послабить фінальний результат передбачення. На рисунку 2.3 наведено стовпчикову діаграму, де відображено зміну значення точності в залежності від розміру набору даних. Діаграму власноруч побудовано на основі даних, які наведено у табл. Б.1, що у додатку Б. Для обчислень підготовлено три набори даних, які містять значну кількість екземплярів, які віднесено до сонячної, хмарної та погоди з опадами [28], [29], [30].

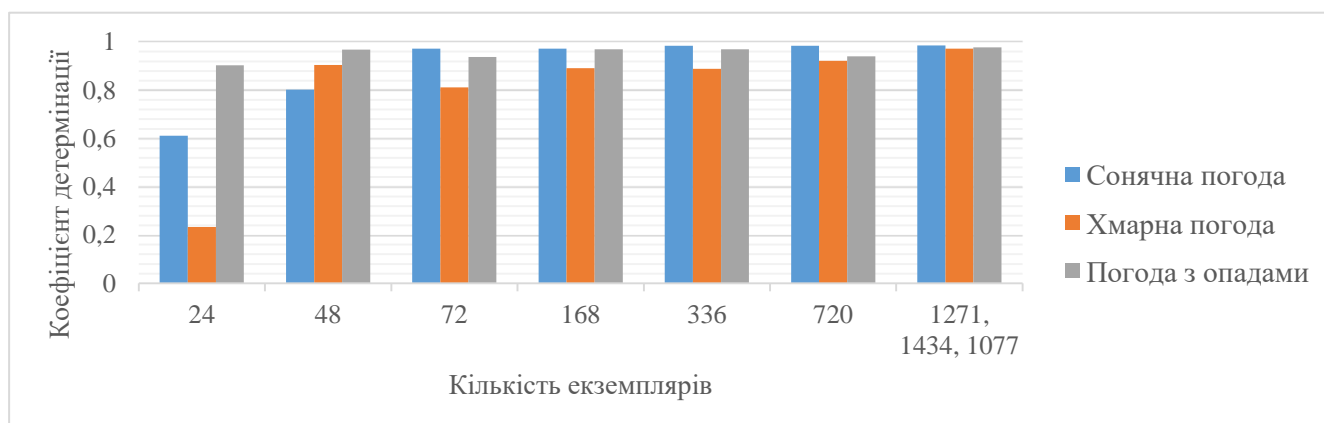


Рисунок 2.3 – Показники ефективності сформованих моделей для набору даних різного розміру

Візуалізувавши дані, можна встановити оптимальне значення та діапазон вибірки для отримання кращих прогнозів. Видно, що для обраного набору даних для кожної групи оптимальне значення відрізняється, наприклад для сонячної погоди це 168 записів, а для погоди з опадами – від 336 до 1008 записів. У табл. 2.3 наведено розміри наборів даних, які отримано в результаті аналізу отриманих даних.

Таблиця 2.3 Розміри вибірки даних для отримання найкращих результатів

Група	Мінімальна кількість	Оптимальна кількість
Сонячна погода	72	168
Хмарна погода	168	336-720
Погода з опадами	168	336-1008

Найменша кількість для групи «Сонячна погода» зумовлена тим, що для такої погоди немає великої кількості варіантів погодних умов, тому на етапі створення моделі будується порівняно мала кількість правил. Крім того, низька варіантність погоди позитивно впливає на час побудови моделі та формування прогнозу.

Отже, для різноманітних наборів даних оптимальний розмір для кожної погодної групи може відрізнятись. Це зумовлено великою кількістю варіацій умов вироблення енергії. Але постійний аналіз результатів прогнозування дозволить визначити тенденцію та скорегувати розміри вибірки для покращення моделі. До того ж проведені обчислення вказують, що для більш «складних» наборів даних потрібно збільшувати кількість екземплярів для побудови моделей.

Оскільки, інші значення погіршують точність прогнозування, прийнято рішення створювати загальну модель для кожного типу погоди без розподілення по сезонах для уникнення таких випадків. Її побудова відбувається у декілька кроків. Насамперед проводиться автоматичний аналіз результатів формування підмоделей, в результаті якого визначаються відсутні та формується масив, де описано їхні погодні групи та пори року. Далі для підготовлених вибірок,

класифікованих по погодних умовах без групування по порах року, відбувається побудова загальних підмоделей для кожної погодної групи. Цей алгоритм дозволяє охопити більше варіантів виробництва електроенергії для різних погодних умов, що розширює можливості формування прогнозу. У авторському методі загальна модель застосовується у наступних ситуаціях:

- відсутність однієї з погодних моделей для конкретної пори року;
- вхідні дані відсутні, або їхня кількість є недостатньою для формування підмоделі для конкретної погодної групи.

Для визначення якості загальних моделей по погодних умовах, обчислено коефіцієнти точності на підготовленому наборі даних [26] для основної, яку побудовано для всіх даних та для моделей, які побудовано для вибірки, котра містить дані за певний сезон, а саме весна. У табл. 2.4 наведено отримані значення критеріїв оцінювання для групи «Хмарна погода». Цю групу обрано для порівняння, через наявну найбільшу кількість екземплярів у наборі даних, які відносяться до неї, а також значний розмір вибірки дає можливість протестувати роботу методу з великим обсягом даних. Варто додати, що особливості обраного базового методу дозволяють обмежити розмір вхідних даних, відповідно до рекомендацій, які встановлено та описано вище.

Таблиця 2.4 Коефіцієнти ефективності порівнюванні з загальною моделлю

Модель	RMSE, Вт	MAE, Вт	R²
«Хмарна погода» (Весна)	0,74188	0,32421	0,935
Загальна модель «Хмарна погода»	5,47863	3,8542	0,894
Основна модель	8,70487	5,3013	0,77

Отримані результати вказують, що точність загальної моделі є більшою за основну, яка побудована для усієї вибірки даних, але меншою за побудовану з використанням групування по сезонах та класифікації по погодних умовах. Отже,

це свідчить, цю дану модель доцільно використовувати для формування прогнозу продукування електроенергії для виняткових ситуацій, які наведено вище.

2.1.3 Аналіз проблем, які впливають на точність методу

Під час розробки методу прогнозування визначено ряд причин, через які утворені моделі можуть повертати хибний результат. До них віднесено:

- неточність прогнозу погоди при продукування енергії, який зберігається у базі даних для подальшого прогнозування;
- недостатня кількість даних для проведення навчання та тестування;
- відсутність репрезентативної вибірки.

Щоб зменшити вплив таких факторів на результат, проаналізовано кожен причину та створено особливості методу, які їх вирішують.

Вирішення проблеми, яку наведено першою в списку, може мати декілька способів. Якщо взяти до уваги, що сонячна станція є великою і виробляє велику потужність, то точність погоди є важливим фактором при прогнозуванні, тому для отримання найкращих даних власнику необхідно встановити власну метеостанцію. Вона дозволить збирати інформацію про погоду станом на момент генерації. Іншим вирішенням даної проблеми, за відсутності метеостанції, є використання кількох сервісів для прогнозу погоди та аналізувати власноруч дані, які ними надаються. Такий аналіз протягом певного періоду дозволить аналітикам визначити один сервіс, який надає кращі дані для конкретного розташування сонячної станції. Щоб врахувати ситуацію отримання прогнозу погоди на різних сервісах, у методі додано можливість для вказування використовуваного сервісу у вхідних даних про сонячну станцію. Це дозволяє створювати прогнози, використовуючи одне джерело інформації.

Відсутність даних для певного періоду чи їхня недостатня кількість викликає застосування загальної моделі. Для того, щоб вона давала якомога точніші результати її варто тестувати на спеціально сформованих наборах даних, які допоможуть врахувати особливості виробництва енергії індивідуально для

кожної станції. Також перед прогнозуванням рекомендовано, щоб аналітик проводив попередній аналіз даних та враховував особливості розташування станції.

2.1.4 Процес прогнозування енергії та збереження моделі

Основним завданням методу є прогнозування кількості енергії на необхідний проміжок часу. Оскільки, прогнозу виробництва можна робити на декілька днів, то аналітик може робити декілька прогнозів для однієї доби та аналізувати отримані значення. На цьому етапі виконуються три основні кроки: отримання прогнозу погоди на вказаний період та інших вхідних даних, групування отриманої інформації та створення прогнозу з подальшим його збереженням.

Прогноз погоди завантажується з сервісу, який вказано в інформації про сонячну станцію для розташування сонячної електростанції. Отримані дані обробляються та об'єднуються у асоціативний масив, де ключами є назви атрибутів, які використовуються при прогнозуванні, а значеннями є отримана погодинна інформація. До складових прогнозу погоди та інших атрибутів відносяться:

- рік, місяць, день, година;
- сонячна радіація;
- температура повітря;
- кількість опадів;
- щільність хмар;
- тиск;
- вологість;
- швидкість вітру.

Групування вхідного масиву даних перед розрахунками електроенергії відбувається аналогічно до того, як на етапі формування моделі. Отримані групи використовуються як вхідні дані при прогнозуванні для відповідної моделі.

Результатом методу є сформована модель, використання якої дозволяє отримати прогнозовані значення, у вигляді асоціативного масиву, де ключем є дата, на яку здійснюється прогнозування, а значенням є масив словників з елементами,

які вказують годину, кількість енергії та використану модель. Застосована модель позначається символьним кодом, по якому її можна конкретно встановити. Якщо символна послідовність складається з двох символів, то перший вказує на пору року (W – зима, S – весна, U – літо, A – осінь), а другий на погодну групу. Односимвольний код, який позначається однією з чотирьох літер S (сонячна), C (хмарна), P (з опадами), G (базова) вказує одну з загальних моделей. Такий вигляд результуючих даних дозволяє зручно інтегрувати цей метод у систему для прогнозування енергії. Нижче наведено опис частини цієї структури даних.

```
{
  "12.07.2022":
    [
      {
        "hour": 0,
        "quantity_energy": 0,
        "used_model": 'UC'
      },
      {
        "hour": 1,
        "quantity_energy": 0,
        "used_model": 'P'
      },
      ...
      {
        "hour": 12,
        "quantity_energy": 48.5142,
        "used_model": 'US'
      },
      ...
      {
        "hour": 23,
        "quantity_energy": 0,
        "used_model": 'G'
      }
    ]
}
```

Для того, щоб утворену модель для прогнозування можна було б використовувати в подальшому, створено можливість її збереження. Модель зберігається у вигляді архіву, у якому збережено інформацію про неї у вигляді файлу JSON та окремі підмоделі для відповідної групи. Чітка структура створеного файлу дає змогу відтворити його вміст в зворотному напрямі. Крім того, оскільки

всі підмоделі містяться у архіві, їх можна окремо аналізувати та використовувати для вдосконалення.

Отже, наведені обчислення вказують на підвищення точності прогнозування, шляхом поділу вибірки за порою року та класифікації її відповідно до характеристик погоди. Крім того, послідовне встановлення обмежень для розміру множини для створення моделі дало змогу визначити мінімальну та оптимальну кількості екземплярів для отримання кращих результатів, а аналіз певних проблем, які впливають на якість моделі дозволив прийняти рішення для їхнього опрацювання та уникнення. Також опис процесу прогнозування дає можливість наочно зрозуміти вигляд результату застосування створеної моделі.

2.2 Структура алгоритму

Для моделювання дискретної поведінки алгоритму, створено діаграму станів UML, де показано послідовність подій, через які об'єкт проходить протягом свого існування [31, с. 305]. Поведінка цієї діаграми складається з однієї або кількох областей, кожна з яких містить граф, що містить набір вершин, з'єднаних між собою дугами, що представляють переходи. Виконання діаграми станів запускається відповідними подіями [31, с. 306-307]. Варто виокремити основні елементи діаграми станів, до яких відносяться:

- Області (Regions) – ортогональна частина складеного стану або кінцевого автомата, яка може містити стани та переходи;
- Вершини (Vertices) – абстрактний клас, який описує характеристики різних вузлів діаграми станів;
- Стани (States) – моделюють ситуацію під час виконання поведінки діаграми, під час якої виконується інваріантна умова;
- Кінцевий стан (FinalState) – особливий тип стану, що показує завершеність наявного регіону;
- Псевдостан (Pseudostate) – абстракція, яка охоплює різні типи транзитивних (перехідних) вершин у діаграмі станів, які як правило

використовуються для об'єднання кількох переходів у складені. Нижче наведено основні їх види:

- Точка входу (Entry point) та точка виходу (Exit point) – вказують на початок та кінець діаграми станів чи складеного стану, а також забезпечують інкапсуляцію;
- Вибір (Choice) – реалізує динамічний вибір, на основі умов вихідних переходів;
- Розгалуження (Fork) – служить для поділу вхідного переходу на два або більше переходів, що закінчуються на вершинах в ортогональних регіонах складеного стану;
- Об'єднання (Join) – служить спільною цільовою вершиною для двох або більше переходів, що походять від вершин у різних ортогональних областях;
- З'єднання (Junction) – використовується для з'єднання кількох переходів у складені шляхи між станами;
- Переходи (Transitions) – однонапрямлена дуга, яка проходить від однієї вихідної вершини до цільової, яка визначає дійсний фрагмент поведінки діаграми станів [31, с. 307-315].

Для створення діаграми станів використано програмний засіб StarUML. Це програмне забезпечення призначене для моделювання систем, використовуючи UML та має багато особливостей, з яких можна виокремити: кросплатформність, сумісність з стандартом UML 2.x, експорт діаграм тощо. Варто додати, що програма має детальну онлайн-документацію, де описано особливості роботи кожного модуля та покрокові інструкції для кожної з доступних діаграм [32].

Щоб змоделювати структуру алгоритму, власноруч створено діаграму станів (рис. 2.4), де визначено три основні стани, які пов'язані переходами: «Завантаження даних», «Створення підмоделей» та «Збереження моделі». Вони зображені як підмашинні стани, бо для покращення сприйняття їхню графічну складову винесено окремо, а також вони містять стани, опис можливостей яких наведено нижче.

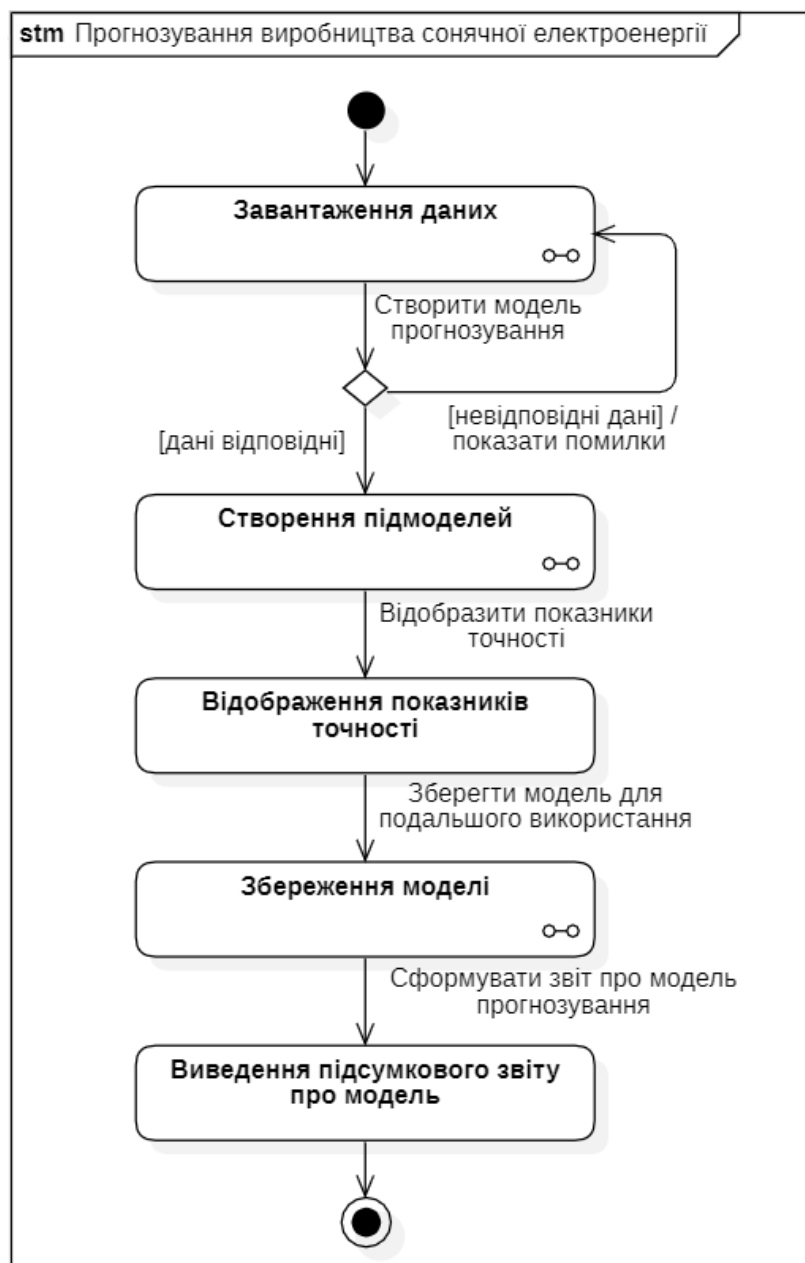


Рисунок 2.4 – Діаграма станів «Прогнозування виробництва сонячної електроенергії»

Стан «Завантаження даних» моделює ситуацію під час якої виконуються підстави для обробки даних. Для цього стану визначено поведінки, які мають виконатись при вході та виході, а саме отримання даних та повернення згрупованих опрацьованих даних відповідно. Відповідно до входньої поведінки, перейшовши до однойменного стану відбувається наступний перехід з вказаною причиною «Перевірка структури даних» до вибору стану, який здійснюється в залежності від вказаної умови. Якщо входні дані є відповідними, то відбувається перехід до стану

«Підготовка даних», якщо ні – «Отримання даних». В успішному випадку об'єкт набуває наступного стану «Групування даних за погодними умовами», а після нього до композитного стану «Завантаження даних». Їх зображено на діаграмі у вигляді складеного стану з відповідним позначенням у правому куті, а їхній зміст описано на окремих діаграмах. Дане рішення реалізовано для покращення графічного сприйняття схеми. Вищеописаний фрагмент діаграми станів наведено на рисунку 2.5.

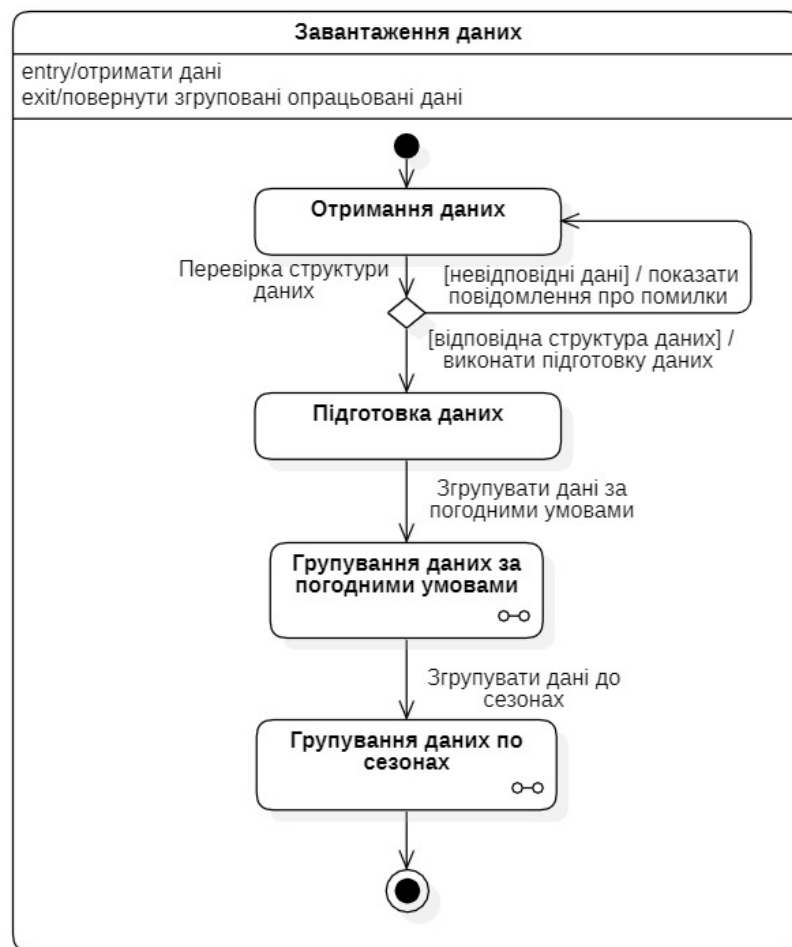


Рисунок 2.5 – Діаграма складеного стану «Завантаження даних»

Декомпозиція стану «Групування даних за погодними умовами» (рис. 2.6) містить два послідовні стани «Формування загальної вибірки» та «Класифікація за погодними умовами», перехід від першого до другого відбувається з встановленою поведінкою «Згрупувати дані для формування нових наборів». Далі наступний стан визначається умовою, яка вказана у квадратних дужках. Для даного вибору об'єктом порівняння є назва класу, яку повертає класифікатор, а результатом –

перехід до одного з станів, в якому відбувається збереження екземпляру набору даних, до відповідної вибірки. Після збереження виконується поведінка, яка описана у частині внутрішньої діяльності стану.

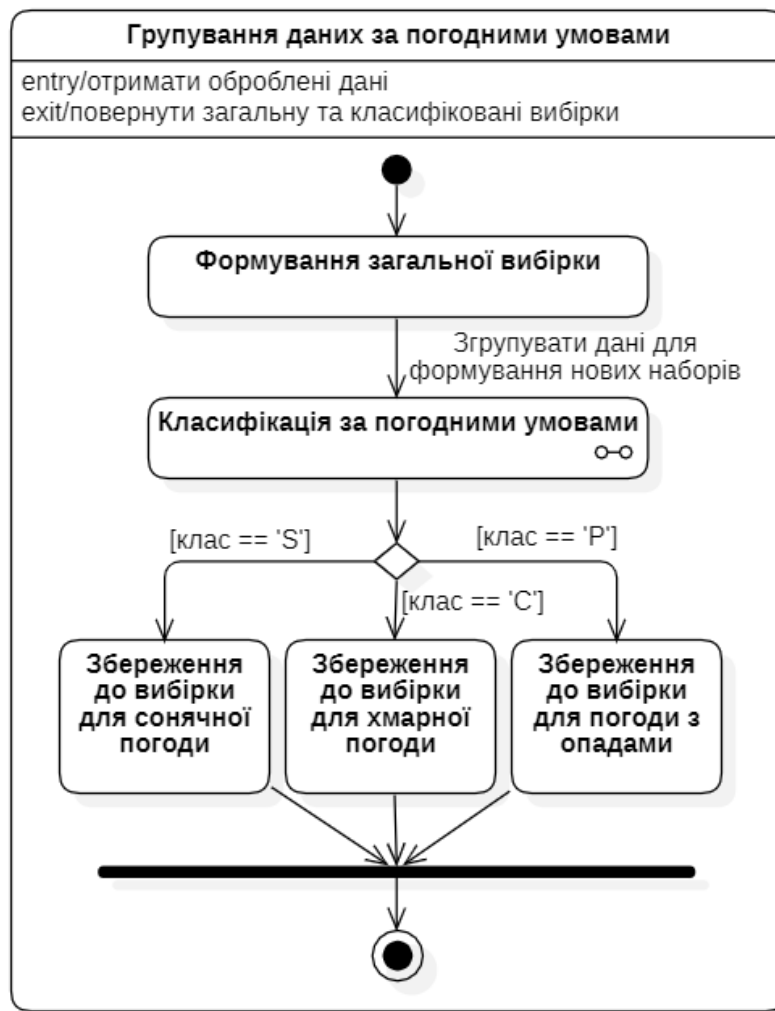


Рисунок 2.6 – Діаграма складеного стану «Групування даних за погодними умовами»

Композитний стан «Класифікація за погодними умовами» має певні особливості. При вході у цей стан відбувається отримання значень хмарності та опадів для обробки. Далі здійснюється перехід до псевдостану вибору для оцінки кількості опадів. Якщо вона є більшою за нуль, то набувається стан «Клас 'P'», в іншому випадку відбувається порівняння значення хмарності з використанням вибору. Умовою набуття стану «Клас 'C'» є хмарність, яка є більшою за 10%, в іншому випадку відбувається перехід до стану «Клас 'S'». Після встановлення одного з класів виконується поведінка, встановлена для виходу зі складеного стану,

а саме повернення класу. На рисунку 2.7 наведено діаграму для стану «Класифікація за погодними умовами».

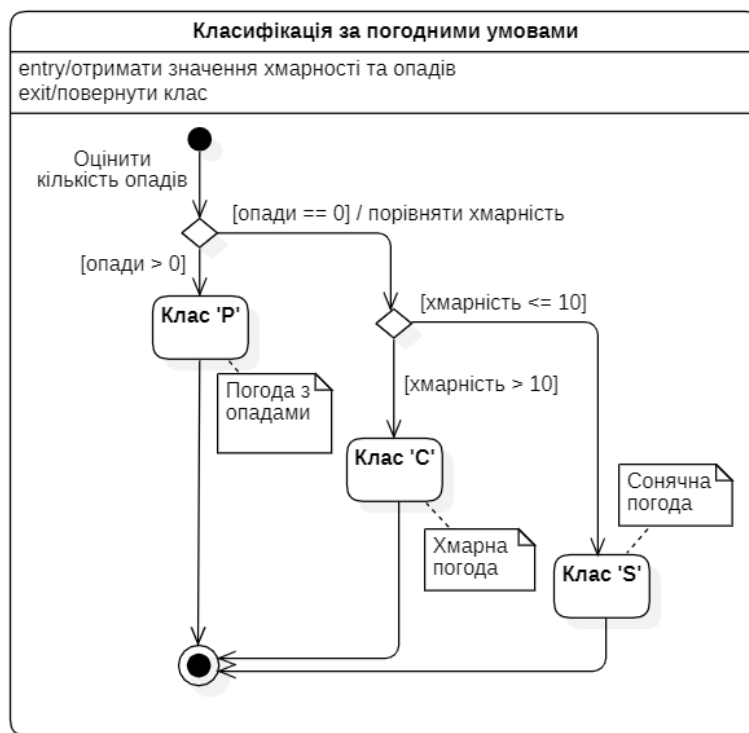


Рисунок 2.7 – Діаграма складеного стану «Класифікація за погодними умовами»

Процес розбиття даних, які класифіковано за погодними характеристиками, відповідно до пори року, описано у вигляді композитного стану «Групування даних по сезонах» (рис. 2.8). Його декомпозиція містить три послідовні стани, перший з яких це «Отримання згрупованих даних за погодними умовами». В момент перебування в цьому стані, об'єкт має містити три вибірки даних, які належать до кожної погодної групи, якщо вони наявні. Перехід до наступного стану «Отримання даних для конкретної погодної групи» відбувається з встановленою поведінкою «Надати конкретну погоду вибірку». Далі здійснюється групування даних по порах року та перехід до стану «Збереження групи даних у масив для пори року». Наступним є перехід до псевдостану розгалуження, де перевіряється наявність нерозподілених даних, якщо такі є, то об'єкт повертається до одного з попередніх станів, а якщо немає, то відбувається вихід з складеного стану, який описаний поведінкою, яка повертає класифіковані вибірки за погодними умовами та сезонами.

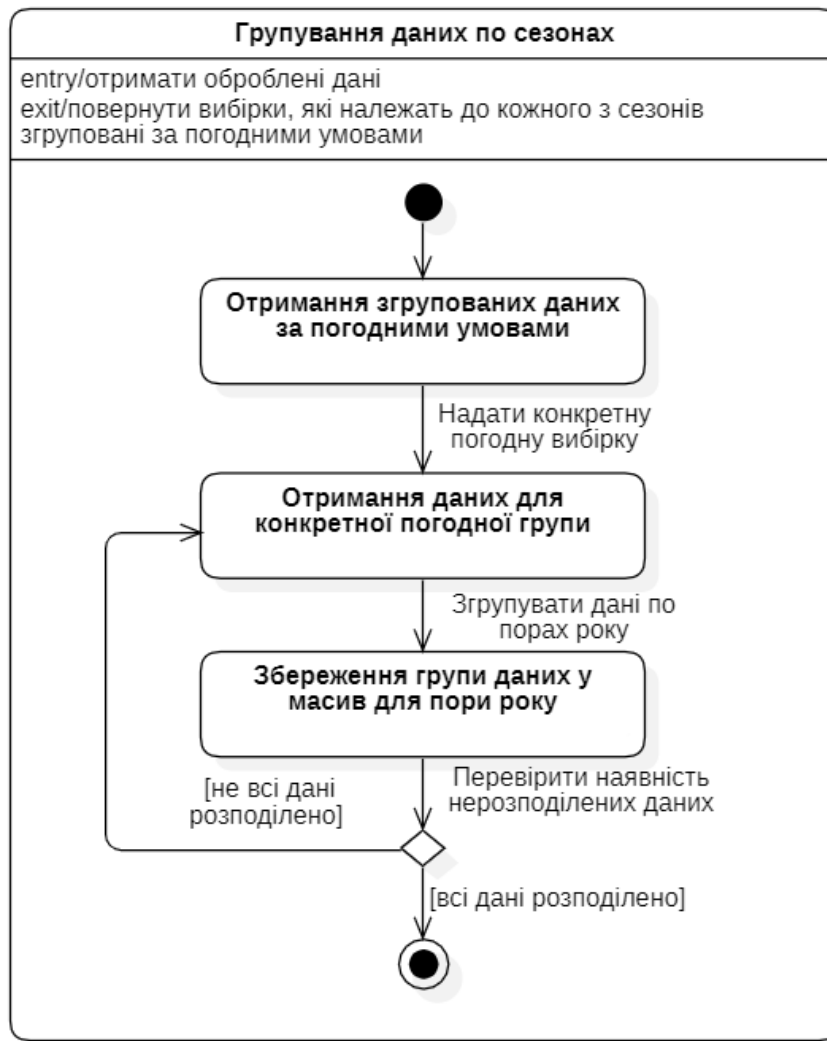


Рисунок 2.8 – Діаграма складеного стану «Групування даних по сезонах»

Після завантаження даних відбувається перехід до вибору, після якого об'єкт може перейти до складеного стану «Створення підмоделей» або повернутись до початку наявного. Перехід від псевдостану вибору здійснюється залежно від вхідної умови переходу. Композитний стан «Створення підмоделей» містить наступні стани:

- отримання загальної та для кожної погоди вибірок;
- розподілення даних на навчальну та тестову множини;
- отримання вхідних параметрів для кожної підмоделі;
- паралельне генерування кожної підмоделі;
- прогнозування виробленої електроенергії;
- обчислення показників точності.

Кожен стан пов'язаний переходами з власною поведінкою. Варто виокремити ортогональний стан «Паралельне генерування кожної підмоделі». До нього відбувається чотири переходи з власною поведінкою одночасно, які реалізовано за допомогою псевдостану розгалуження. Вони прямують до елемента вибору, де у випадку наявності даних відбувається перехід до стану створення відповідної підмоделі, а у протилежному – обхід цього стану та виведення на екран повідомлення про відсутність вибірки. Діаграму композитного стану «Створення загальних підмоделей» виокремлено для покращення сприйняття та позначено на діаграмі спеціальним символом. Перехід до нього відбувається завжди, незалежно від вхідного множини. Далі відбувається перехід до псевдостану об'єднання, щоб розрахувати прогнозовані значення для тестової множини. Цей процес відбувається у складеному стані «Прогнозування виробленої електроенергії», який наведено винесено на окрему діаграму. Вищеописану діаграму наведено на рисунку В.1, що у додатку В.

Варто виокремити діаграму для побудови загальних моделей для різних погодних груп, якщо відсутні дані для конкретної пори року. Діаграма стану «Створення загальної підмоделі» (рис. 2.9) має визначені поведінки, що мають виконатись на вході та виході, які вказані за допомогою зарезервованих міток, а саме «Перевірити вхідну множину даних» та «Повернути загальну підмодель» відповідно. Для цієї схеми визначено стани «Завантаження набору для конкретної погодної групи», «Створення загальної підмоделі для конкретної погодної групи», перехід до яких відбувається поступово.

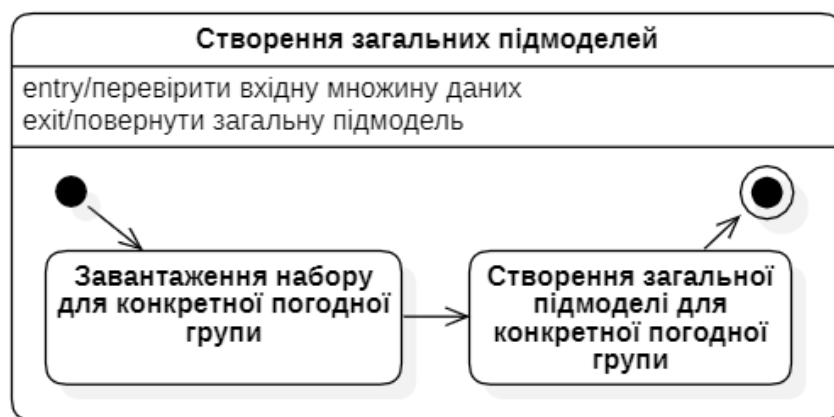


Рисунок 2.9 – Діаграма композитного стану «Створення загальної підмоделі»

Крім цього, важливою є діаграма для стану «Прогнозування виробленої електроенергії», яку винесено окремо, а на основній її позначено як композитний стан. Вона містить звичайні та композитні стани, псевдостани вибору та об'єднання, а також вказані дії при вході у стан та виході з нього. Спершу здійснюється перехід до стану «Визначення періоду» для отримання прогнозу погоди та інших вхідних даних, після чого поведінка набуває складеного стану «Формування вхідних даних для прогнозування». Він містить три послідовні стани: «Завантаження прогнозу погоди», «Завантаження даних про сонячну радіацію» та «Формування масиву даних за визначеною структурою». Після дії, яка має виконатись при виході, відбувається перехід до композитного стану «Групування даних за погодними умовами», опис якого наведено вище. Наступним є – «Завантаження моделі», після якого здійснюється перехід до вибору з умовою наявності підмоделі для обчислення прогнозованих значень. Далі, за допомогою псевдостану об'єднання, результати формуються у відповідний масив та виконується дія, яка вказана для виходу з складеного стану. Діаграму вищеописаного стану зображено на рисунку В.2 у додатку В.

Вихід з складеного стану «Створення підмоделей» супроводжується черговим переходом до «Відображення показників точності», а далі до – «Збереження моделі» (рис. 2.10). До останнього входять наступні стани:

- отримання підмоделей;
- створення архіву;
- створення файлу з підмоделлю;
- збереження підмоделі в архіві;
- формування файлу з інформацією про прогнозування;
- збереження файлу в архіві;
- збереження архіву.

Варто зазначити, що до архіву додаються наявні створені підмоделі за допомогою псевдостану вибору. При виході з складеного стану виконується виведення інформації про збережений файл.

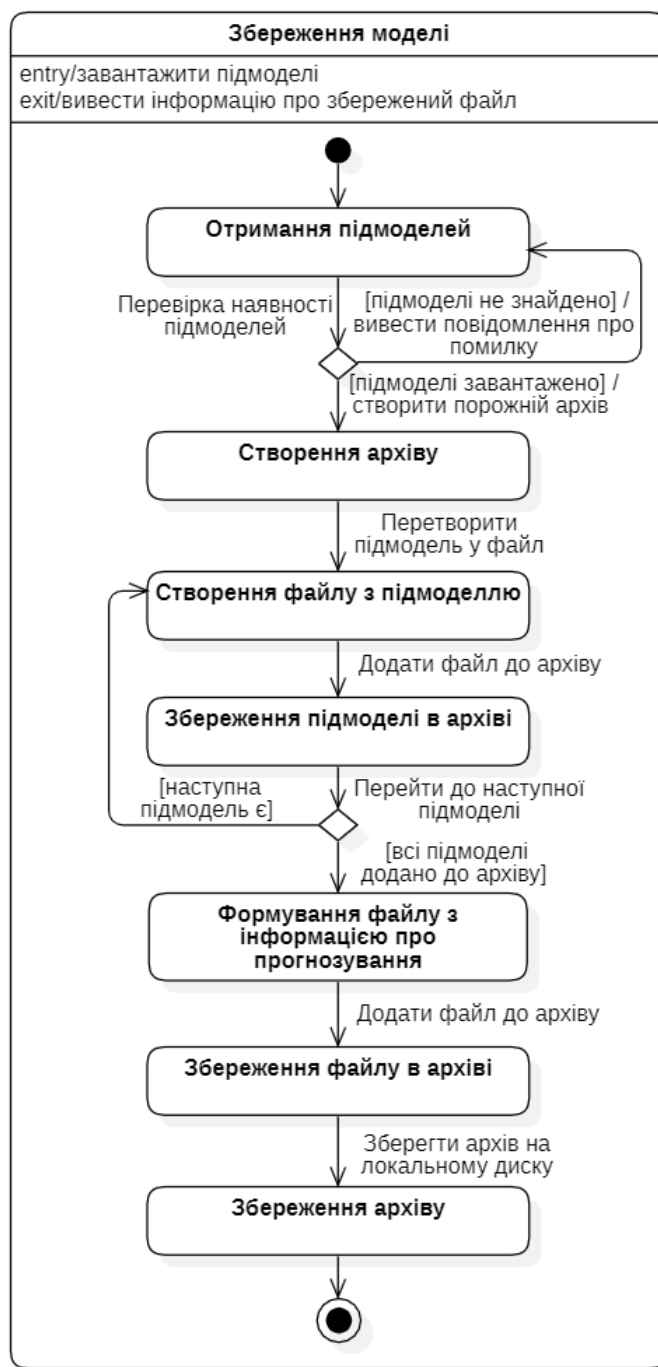


Рисунок 2.10 – Діаграма складеного стану «Збереження моделі»

Останній станом, до якого відбувається перехід є «Виведення підсумкового звіту про модель», після чого здійснюється перехід до кінцевого стану, що вказує на завершення роботи визначеної діаграми.

Ключові алгоритми, які використовуються у процесі прогнозування, а саме завантаження даних, групування даних відповідно до погодної групи, групування даних по порах року та формування прогнозу на певний період часу, описано за

допомогою алгоритмічних моделей у вигляді блок-схем. Їх власноруч реалізовано у програмному середовищі для моделювання StarUML.

Процес завантаження вхідних даних (вибірки, інформації про сонячну станцію та параметрів моделі) описано послідовно, де кожен крок представлено у вигляді окремого блоку. Спершу відбувається завантаження набору даних, котрий містить інформацію про генерування електроенергії, використовуючи функцію для читання файлів csv. Послідовну перевірку вхідних даних за допомогою власних функцій зображено у вигляді блоків «Функція». Далі, у блоці «Умова» перевіряються результати перевірки і залежності від результатів опрацювання відбувається перехід до ряду блоків. Результатом цієї функції є повернення булевого значення, який зображено у вигляді овального прямокутника, що символізує завершення процесу. Результат моделювання наведено на рисунку 2.11.

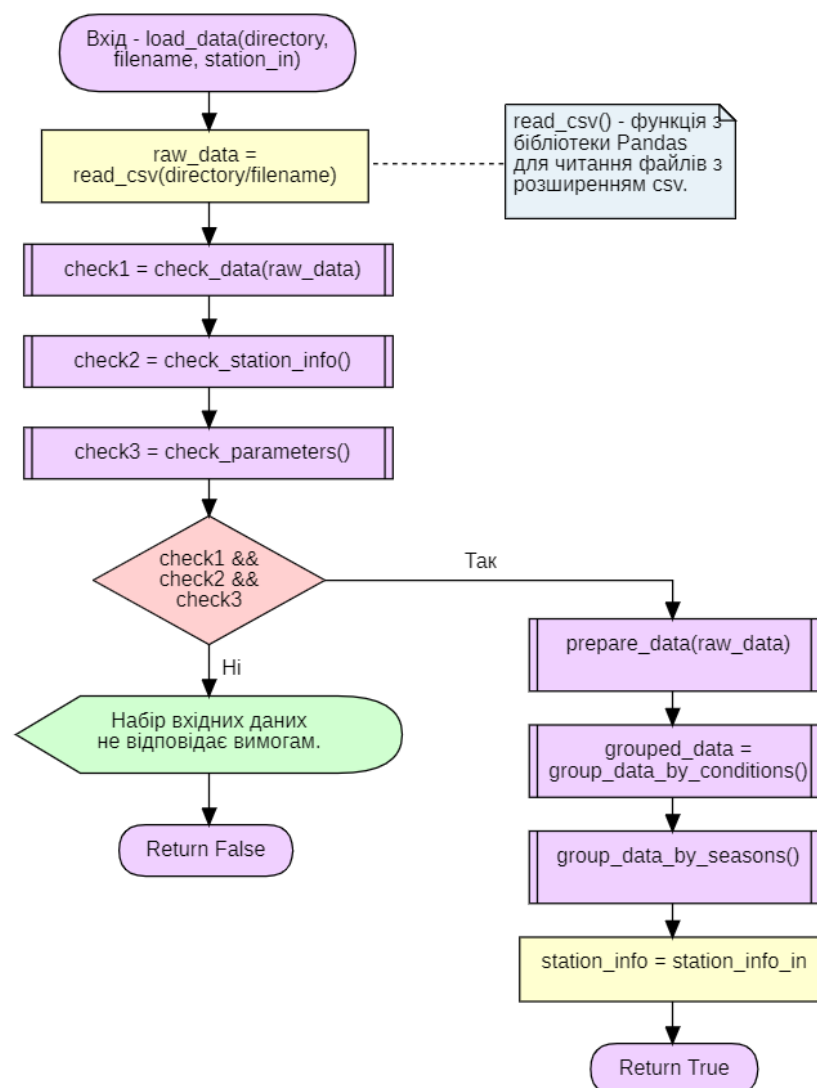


Рисунок 2.11 – Блок-схема функції для завантаження даних

Блок-схема алгоритму для групування даних за погодними умовами передбачає на початку ініціалізацію даних, для який буде групування. Це зроблено тому, що ця функція використовуватиметься під час розбиття вхідного набору даних, який збережено в межах класу та даних про погоду, які є параметром описаної функції. Далі за допомогою блоку «Цикл з параметром» зображено послідовне опрацювання вхідної множини даних та розподілення по відповідних масивах (рис. 2.12), яке містить отримання значень хмарності та кількості опадів, визначення класу, яке наведено у вигляді блоку «Функція», та подальше розподілення до відповідного масиву з використанням блоку «Умова». Крім цього, на діаграмі наведено блок для виведення повідомлення про невідповідність класу. Завершення функції зображено у вигляді блоку «Кінець», де описано список структур, які вона поверне. Повне зображення алгоритму для групування даних по погодних умовах наведено на рисунку Г.1, що у додатку Г.

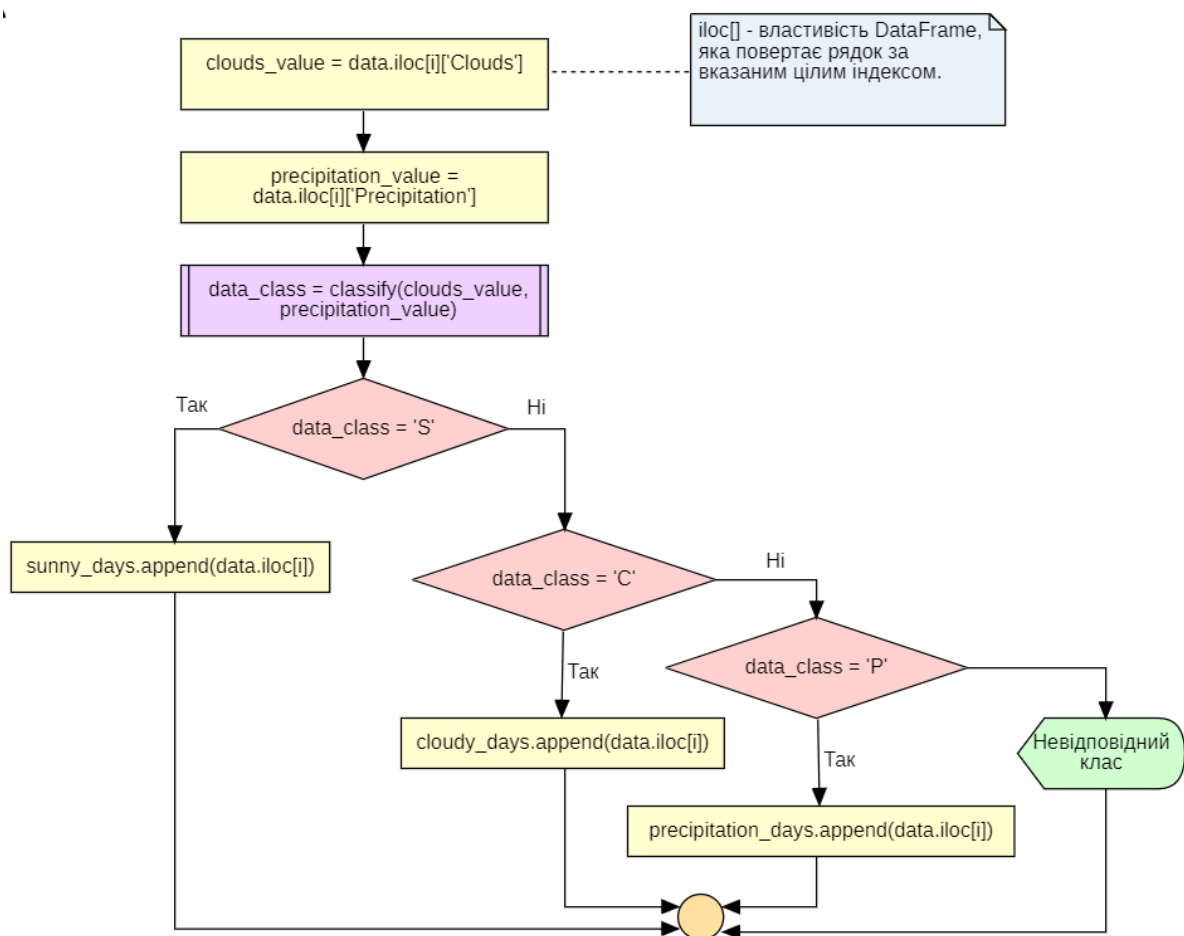


Рисунок 2.12 – Фрагмент блок-схеми, на якому зображено послідовне розбиття даних до відповідних масивів

Блок-схема алгоритму групування даних по порах року дозволить чітко зобразити важливі етапи цього процесу. На початку за допомогою блоку «Процес» наведено оголошення асоціативного масиву, де ключами є пора року, а значеннями масив, який містить початкове та кінцеве значення порядкових номерів місяця. Далі блок «Цикл з параметром» відображає процес послідовного розбиття даних кожної погодної групи по порах року. Також на блок-схемі відображено розгалуження, яке виконується в залежності від наявності даних і яке зображено у вигляді блоку «Умова». Етап збереження результату групування зображено за допомогою блоку «Функція». Додаткові відомості про використані функції наведено у вигляді коментарів. Фрагмент блок-схеми, де описано процес формування масиву даних для конкретної пори року зображено на рисунку 2.13, а повне зображення наведено на рисунку Г.2, що у додатку Г.

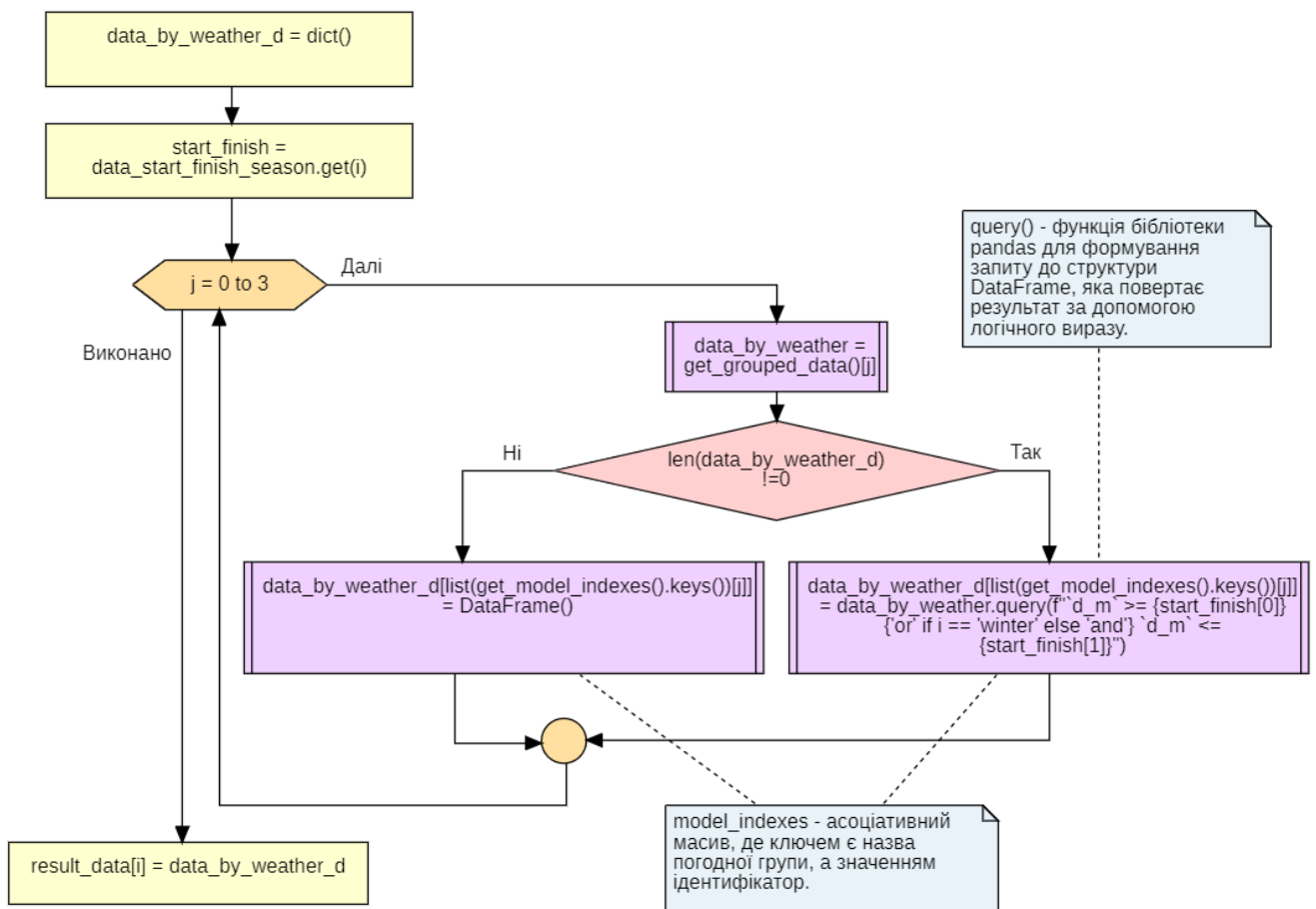


Рисунок 2.13 – Фрагмент алгоритмічної моделі формування масиву даних для конкретної погоди

Алгоритм прогнозування виробництва електроенергії є основним у розробленому методі, тому що він містить кроки, які забезпечують завантаження даних прогнозу погоди та їхнє групування по погодних умовах, а також визначення конкретної моделі для передбачення для визначеної групи.

У першій частині цієї алгоритмічної моделі описано підготовку для прогнозування (рис. 2.14). Блок «Функція» позначає процес, де формується дата рядкового типу, на яку здійснюється прогноз. Присвоєння початкових значень для змінних, які зберігатимуть інформацію про прогнозовані значення та кількість електроенергії зображено за допомогою елементу «Процес». Підготовчий етап завершується формуванням множин, шляхом використання власної функції. Опис додаткових методів, які використовуються на цьому етапі, наведено за допомогою елементу «Коментар».

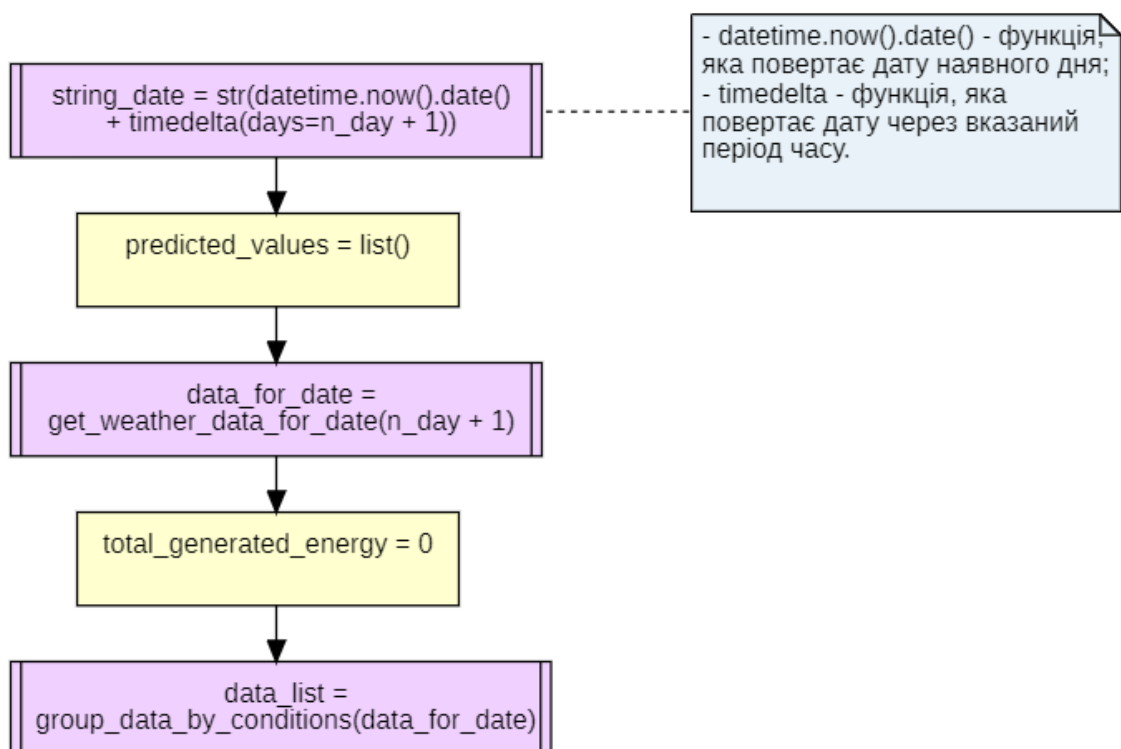


Рисунок 2.14 – Фрагмент блок-схеми, який містить блоки, які зображають завантаження прогнозу погоди та їхнє групування

Важливе місце займає частина прогнозування (рис. 2.15), яка послідовно виконується для сформованих груп, за допомогою циклу з параметрами, який зображено на діаграму однойменним елементом. Далі у блоці «Умова» наведено

перевірку даних на наявність елементів і в залежності від результату два переходи: якщо «Так», то продовження процесу прогнозування, якщо «Ні», то перехід до наступної ітерації циклу. Процеси вибору моделі та прогнозування зображено у вигляді елементу «Функція», а обчислення суми прогнозованої енергії – у вигляді елементу «Процес». Сформований прогноз зберігається у вигляді визначеної на попередніх етапах структури та в подальшому упорядковується погодинно та зберігається. Ці процеси наведено на схему у вигляді блоків «Процес». Вивід результату зображено за допомогою блоку «Виведення». Послідовне повторення вищеописаних дій для необхідної кількості днів відображено за допомогою елементу «Цикл з параметром». Повне графічне відображення блок-схеми для алгоритму прогнозування генерації електроенергії на сонячних електростанціях наведено на рисунку Г.3, що у додатку Г.

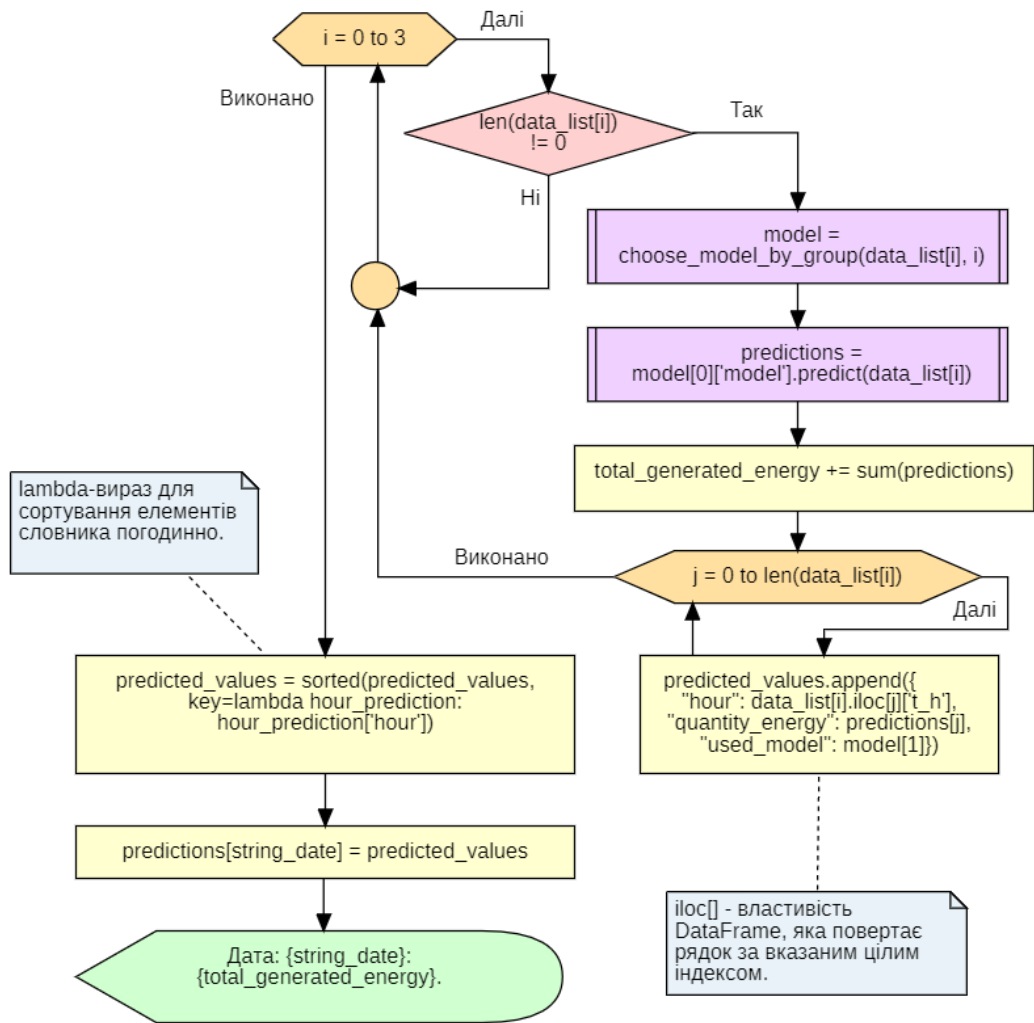


Рисунок 2.15 – Фрагмент алгоритмічної моделі погодинного прогнозування виробництва електроенергії

Таким чином, розроблена діаграма станів демонструє поведінку алгоритму і дає можливість проаналізувати та визначити роботу об'єкта, який перебуває в одному з станів, а алгоритмічні моделі дозволяють детально описати ключові моменти в механізмі прогнозування.

2.3 Засоби для розробки методу та системи

Вибір засобів розробки є важливим етапом розробки після проектування системи. Він здійснювався на основі інформації про розробку сучасних аналогів та вивчення особливостей різних технологій. Розроблені діаграми дають можливість виявити частини програмного забезпечення, які потребують більших витрат ресурсів та часу роботи, для яких варто продумати декілька альтернативних шляхів для досягнення максимального рівня продуктивності.

Для розробки методу прогнозування обрано мову програмування Python та бібліотеки, які призначені для машинного навчання. Обрана мова програмування є лідером у рейтингах мов програмування для обробки даних та машинного навчання. Його особливості дозволяють поєднувати потужність багатьох мов програмування з простотою використання пакетів скриптових мов, до яких належить MATLAB. Python містить в собі великий набір бібліотек та засобів для роботи з великими наборами даних. Вони дозволяють проводити завантаження та візуалізацію даних, виконувати статистичні обчислення, здійснювати побудову моделей прогнозування тощо [33, с. 5].

Для роботи з наборами даних обрано бібліотеку Pandas, яка забезпечує роботу з гнучкими структурами даних [33, с. 10]. Цей пакет дозволить реалізувати певні деталі, які необхідні для реалізації методу завдяки своїм особливостям, основні з яких наведено у наступному списку:

- швидкий та ефективний об'єкт DataFrame для обробки даних з вбудованим індексуванням. Він дозволяє проводити модифікацію та аналіз даних, подібно до таблиць Excel та SQL;
- засоби для читання та запису даних різних форматів. До них відносяться текстові файли, CSV, JSON та інші;

- гнучка зміна форми наборів даних. За допомогою бібліотеки можна додавати та видаляти стовпці, виконувати високопродуктивне об'єднання даних [34].

Щоб виконувати обчислення з великими наборами даних для методу, взято пакет NumPy на Python. Він містить функціональні можливості для роботи багатовимірними масивами та засобами чисельних обчислень [35].

Для отримання базового методу, на основі якого створено власний, використано бібліотеку scikit-learn. Даний пакет містить великий набір різних засобів для машинного навчання. Інструменти бібліотеки дозволяють виконувати попередню обробку вхідного набору даних, розв'язувати задачі класифікації, регресії та кластеризації. Крім цього, пакет містить детальну документацію, де можна отримати додаткові відомості про реалізовані алгоритми та їхні параметри [36].

Оскільки на етапі аналізу проблем визначено, що потрібно додати можливість завантаження прогнозу погоди з різних сервісів, то обрано два сервіси, а саме OpenWeatherMap та Open-Meteo. Перший з них надає безкоштовний доступ для погодних даних, використовуючи API. Крім цього, служба надає платні послуги, контроль за якими здійснюється з використанням персональних ключів API. Проте, значення кількості сонячної радіації можна отримати тільки користуючись однією з платних підписок, що є суттєвим недоліком для вирішення поставленої задачі [37]. Другий сервіс – Open-Meteo, надає усі дані про погоду безкоштовно для некомерційного використання. Також особливістю цього сервісу є відсутність ключів API. Крім того, перевагою даної погодної служби є наявність спеціалізованих даних, наприклад, кількість сонячної радіації, вологість ґрунту, якість повітря тощо [18]. Обидва сервіси мають детальну документацію API, де описано структуру вихідних даних та вимоги до запитів.

Щоб реалізувати збереження створеної моделі, для проекту використано бібліотеку Joblib та вбудований модуль мови Python – zipfile. Joblib створений для забезпечення конвеєрної обробки великих даних [38]. У власному методі, функції цієї бібліотеки варто використати для перетворення підмоделей у файли. Створені

файли, відповідно до діаграми, створеної у попередньому підрозділі, будуть об'єднані у архів з використанням функцій модуля `zipfile`. Використання функцій цього модуля дозволяє виконувати операції створення, читання, запису тощо [39].

Крім того, для демонстрації практичного застосування методу, розроблено комплексну інформаційну систему з можливостями керування зібраними даними, формування прогнозу виробництва та надсилання звіту про отримані значення.

Основним засобом, який використано для розробки системи є Python, Ця мова програмування дозволяє створювати складні графічні інтерфейси користувача (GUI) і веб-сервіси, а також інтегрувати їх в існуючі системи.

Для створення інтерфейсу прийнято рішення використовувати PyQt з прив'язкою PySide. Ці засоби поєднують в собі переваги Qt і Python та дозволяють створити мультиплатформний додаток із зручним інтерфейсом [40].

Оскільки система повинна виконувати операції збереження та опрацювання даних, надсилання електронних листів, генерування звітів тощо, то для реалізації обрано наступний перелік додаткових програмних засобів (бібліотеки, драйвери, модулі) Python для забезпечення функціональних можливостей системи:

- `configparser` – модуль для реалізації базової мови конфігурації, використання якого дає можливість зберігати налаштування в одному файлі, вміст якого користувачі можуть змінювати [41];
- `bcrypt` – модуль для хешування паролю з використанням алгоритму шифрування BCrypt [42];
- `mysql-connector` – драйвер MySQL, який реалізовано Python для зв'язку з базами даних MySQL за допомогою API [43];
- `smtplib` – модуль, який визначає об'єкт сеансу клієнта SMTP, який можна використовувати для надсилання електронних листів [44];
- `email` – бібліотека для надсилання листів електронної пошти [45];
- `shutil` – модуль для виконання високорівневих операцій з файлами або сукупністю файлів, наприклад, у ньому реалізовано методи для створення копій та їхнього видалення [46].

Враховуючи, що в системі передбачено відображення даних у вигляді графіків, доцільно використати віджет QChart. Він дозволяє створити графічне представлення послідовностей даних та інших об'єктів графіків, таких як осі та легенда, а набір наявних функцій дає можливість будувати діаграми різного типу (стовпчикові, лінійні, секторні тощо) [47].

Отже, аналіз власного методу підтверджує, що групування вхідного набору даних за певними характеристиками дозволяє отримати кращі результати, але за умови дотримання вимог до розмірів утворених вибірок. Також, аналіз факторів, які можуть погіршити точність дає змогу створити програмні рішення для їхньої мінімізації, а моделювання алгоритму – визначити основні кроки прогнозування, щоб на цьому етапі внести певні корективи перед обранням засобів для розробки та програмною реалізацією.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДУ ПРОГНОЗУВАННЯ

3.1 Реалізація алгоритму

Після моделювання алгоритму та вибору засобів реалізації, встановивши найкращі рішення, реалізовано алгоритм прогнозування генерації енергії. Це здійснено у вигляді класу, який має ряд параметрів, які визначають дані та змінні, що використовуються для реалізації основних функцій.

Першим і одним з основних кроків є завантаження даних. Для цього створено функцію `load_data()`, яка приймає три параметри: назву каталогу, файл з даними та дані про сонячну станцію у вигляді словника. Визначена функція повертає булеве значення в залежності від стану завантаження та перевірки даних. Для перевірки вхідних даних спершу відбувається їхнє читання за допомогою функції `read_csv()` з бібліотеки `pandas`, а потім аналіз за допомогою власних функцій, які виконують перевірку вмісту файлу, даних про станцію та параметрів прогнозування. Якщо всі дані є відповідними, то виконується автоматична попередня обробка даних, групування даних по погодних групах та порах року та збереження інформації про станцію, а також аналізування даних та в результаті функція повертає істинне значення. У випадку невідповідності якоїсь групи даних вимогам – повертається значення `False` та на екран виводиться відповідне повідомлення. Нижче наведено програмну реалізацію вищеописаної функції.

```
def load_data(self, directory, filename, station_info) -> bool:
    raw_data = pd.read_csv(f"{directory}/{filename}")

    if self.check_data(raw_data) and self.check_station_info() \
        and self.check_parameters():
        self.prepare_data(raw_data)
        self.grouped_data = self.group_data_by_conditions()
        self.group_data_by_seasons()
        self.station_info = station_info
        print("Дані завантажено успішно.\nІнформація про набір даних:")
        self.analyse_data()

        return True
    else:
        print("Набір вхідних даних не відповідає вимогам.")
        return False
```

Виконання кожної функції супроводжується текстовими підказками у консольному вікні (рис. 3.1). У випадку помилок чи невідповідності структури якихось вхідних даних, виводиться відповідне повідомлення. Також передбачено виведення на екран списку відсутніх колонок у наборі даних, які визначаються у функції `check_data()`. Це зроблено для перевірки виконання вимог до вхідного файлу. Параметри перевіряються на правильність введених значень, щоб не було неточностей під час побудови моделі. Після успішного завантаження, додатково з'являється на екрані опис отриманої вибірки, у який входить кількість усіх записів, сонячних днів, хмарних днів, днів з опадами, а також період генерації.

```
Завантаження даних розпочато...
Перевірка вхідного набору даних:
  >> Набір даних відповідає вимогам.
Перевірка структури даних про сонячну електростанцію.
  >> Структура даних про сонячну електростанцію відповідає вимогам.
Перевірка параметрів моделей.
  >> Параметри відповідають вимогам.
Дані завантажено успішно.

Інформація про набір даних:
Кількість усіх даних: 7184.
Кількість сонячних днів: 510.
Кількість хмарних днів: 6674.
Кількість днів з опадами: 0.
Період: 8.1.2014 - 8.11.2014.
```

Рисунок 3.1 – Результат виконання функції `load_data()`

Серед усіх використаних функцій, які виконуються після успішної перевірки даних варто виокремити дві: `group_data_by_condition()` для групування даних по погодних характеристиках та `group_data_by_seasons()` для розбиття даних по порах року.

На етапі моделювання визначено, що процес групування даних по погодних умовах буде використовуватись на двох етапах роботи методу: створення моделі та

прогнозування. Тому функцію реалізовано таким чином, що параметр `input_data` для прийняття вхідних даних, є необов'язковим. Наступним кроком є послідовне застосування функції класифікації для кожного екземпляру вхідного набору даних. Для цього застосовується власна функція `classify()`, яка приймає вхідні параметри: значення хмарності та кількість опадів, і в залежності від них приймає рішення та повертає «клас» у вигляді одного з символів «S», «C» або «P». Отримані значення використовуються для визначення структури, до якої варто віднести екземпляр. Результатом функції є список, який містить чотири структури `DataFrame`, які містять дані про сонячну, хмарну, погоду з опадами та увесь набір даних. Нижче наведено лістинг функції `group_data_by_conditions()` для групування даних по погодних умовах.

```
def group_data_by_conditions(self, input_data=None):
    data = self.data if input_data is None else input_data
    sunny_days, cloudy_days, precipitation_days = pd.DataFrame(), \
        pd.DataFrame(), pd.DataFrame()

    for i in range(len(data)):
        clouds_value = data.iloc[i]['Clouds']
        precipitation_value = data.iloc[i]['Precipitation']

        data_class = self.classify(clouds_value, precipitation_value)

        if data_class == 'S':
            sunny_days = sunny_days.append(data.iloc[i],
                ignore_index=True)
        if data_class == 'C':
            cloudy_days = cloudy_days.append(data.iloc[i],
                ignore_index=True)
        if data_class == 'P':
            precipitation_days = precipitation_days.append(
                data.iloc[i],
                ignore_index=True)

    return [sunny_days, cloudy_days, precipitation_days, data]
```

Процес розбиття даних по порах року реалізовано у вигляді функції `group_data_by_seasons()`, результатом виконання якої є структура у вигляді словника, де ключами є назви сезонів, а значеннями – ще один словник, де ключем є погодна група, а значенням – дані у вигляді `DataFrame`, які відповідають умовам.

Щоб визначити множини даних, які відносяться до певної пори року, використано функцію `query()`, яка дозволяє для групи даних у вигляді `DataFrame` створити запит, де вказано умови вибору. У даному випадку, значення фільтруються за номером місяця, значення яких попередньо збережено у вигляді списку. Запит застосовується до кожної погодної групи та результат зберігається у вигляді змінної класу `grouped_data_by_seasons()`.

```
def group_data_by_seasons(self):
    data_start_finish_season = {
        "winter": [12, 2],
        "spring": [3, 5],
        "summer": [6, 8],
        "autumn": [9, 11],
    }

    result_data = {}

    for i in self.season_indexes.keys():
        data_by_weather_d = {}
        start_finish = data_start_finish_season.get(i)

        for j in range(len(self.model_indexes) - 1):
            data_by_weather = self.grouped_data[j]
            if not data_by_weather.empty:
                data_by_weather_d[
                    list(self.model_indexes.keys())[j]] \
                    = data_by_weather.query(
                        f"`d_m` >= {start_finish[0]} "
                        f"({'or' if i == 'winter' else 'and'}) "
                        f"`d_m` <= {start_finish[1]}")
            else:
                data_by_weather_d[list(
                    self.model_indexes.keys())[j]] = \
                    pd.DataFrame()

        result_data[i] = data_by_weather_d

    self.grouped_data_by_seasons = result_data
```

Після завантаження та опрацювання вхідних даних відбувається перехід до побудови моделі. Алгоритм цього процесу детально описано на діаграмі станів та реалізовано у вигляді функції класу, результатом якої є структура, яка містить набір підмоделей, які відповідають порам року та погодним умовам. Функція складається з декількох основних частин: побудова підмоделей по порах року, створення

загальних моделей для кожного типу погоди, створення загальної моделі на основі усіх опрацьованих даних. Перш за все проводиться аналіз згрупованих даних, щоб визначити наявність вибірки для сезонної погодної групи. Результатом є список булевих значень, порядковий номер яких позначає відповідну погоднію групу. Цей аналіз застосовується для встановлення необхідності у побудові загальної моделі для погодної групи. Перед побудовою моделей здійснюється перевірка наявності необхідної кількості даних, розмір яких встановлено на етапі аналізу. Якщо умови є істинними, то для створення підмоделі викликається функція `create_model()`.

Побудова кожної підмоделі є важливим етапом реалізації методу прогнозування. Для цього, у функції `create_model()` описано два параметри: вхідний набір даних та порядковий номер моделі, який використовується для визначення конкретних параметрів для погодної групи. Як було описано в попередньому розділі, базовим методом є Random Forest, який має значну кількість параметрів для корегування точності моделі. Для того, щоб користувач міг встановлювати ці значення для кожної підмоделі окремо, на етапі завантаження моделі можна завантажити масив з ними. Крім цього, для визначення затраченого часу на обчислення, за допомогою функції `time()` однойменного модуля визначається час у секундах до та після побудови дерева рішень, різниця між якими покаже необхідне значення. Також для розрахунку критеріїв оцінки моделі викликається власна функція `calculate_statistic_indicators()`, яка обчислює значення показників RMSE, MAE та R^2 , та повертає їх як результуючі значення. Результатом функції `create_model()` є модель та її коефіцієнти точності, які в подальшому буде збережено для прогнозування. Для їхнього зберігання використовується змінна класу, яка у вигляді словника вміщує створені підмоделі. Результат формування зображується на екрані у вигляді таблиці (рис. 3.2).

Таблиця містить колонки: пора року, погодні група та коефіцієнти точності. Отримана таблиця зображує наявні підмоделі, а також дозволяє аналітику перевірити правильність роботи методу. Для її створення використано функцію `tabulate()` однойменного модуля.

Season	Group	Code	RMSE	MAE	R2	Time
Winter	Cloudy	WC	5.1948	6.1558	0.9167	1.2613
Spring	Sunny	SS	1.8522	1.7228	0.946	0.1256
Spring	Cloudy	SC	6.7428	7.3239	0.898	1.3689
Summer	Sunny	US	1.1377	1.7841	0.944	0.031
Summer	Cloudy	UC	5.831	3.8551	0.901	1.9762
Autumn	Sunny	AS	1.534	1.2906	0.937	0.0256
Autumn	Cloudy	AC	5.6463	4.2531	0.891	1.1717
-	Sunny	S	9.0973	8.1042	0.813	0.8312
-	General	G	12.3715	7.4397	0.783	2.1094

Рисунок 3.2 – Результат створення моделі з критеріями оцінювання

Процес прогнозування реалізовано у вигляді функції `predict_energy()`, яка приймає один параметр, а саме кількість днів, для яких потрібно здійснити прогнозування. По замовчуванню це значення дорівнює 1. Першим кроком для обчислення прогнозу генерації енергії, є отримання прогнозу погоди для розташування, яке вказано при ініціалізації даних та за допомогою погодного сервісу, який також вказано в структурі, що містить інформацію про сонячну станцію. Для даного методу реалізовано процес отримання даних про погоду за допомогою двох сервісів: `OpenWeatherMap` та `Open-Meteo` у вигляді відповідних функцій. У них реалізовано отримання даних з використанням API, а результатом цих функцій є набір даних у вигляді структури `DataFrame`. Лістинг функції для завантаження даних про погоду з сервісу `Open-Meteo` та формування вихідної структури даних наведено у додатку Д. Після завантаження прогнозу погоди, він розбивається по погодних умовах за допомогою одного з вищеперерахованих

методів. Далі з використанням циклу `for`, послідовно для кожної групи даних, які утворено після класифікації по погодних характеристиках, визначається яку саме підмодель варто застосувати для наявної групи за допомогою функції `choose_model_by_group()`. Її завданням є визначити наявну модель та повернути її та кодову послідовність, яка їй відповідає. Це відбувається в кілька етапів: визначення пори року та перевірка наявності погодної підмоделі для сезону. Далі, якщо таку модель знайдено, то формується символічний код та повертається разом з моделлю, у протилежному випадку – повертається загальна модель для відповідної групи погоди (сонячна, хмарна чи з опадами). Також передбачено варіант відсутності загальної погодної моделі і у такій ситуації результатом є основна модель, яку сформовано на множині усіх даних. Визначивши модель, здійснюється прогнозування, обчислення загальної суми виробленої енергії та формування структури, яка містить погодинну інформацію про прогноз, а саме: годину, кількість енергії та застосовану модель. Результат прогнозування зберігається у змінній `predictions`, яка належить класу, а також виводиться на екран. Нижче наведено програмний код вищеописаної функції для прогнозування продукування енергії.

```
def predict_energy(self, quantity_days=1):
    if quantity_days < 1:
        print("Невірно введено кількість днів.")
        return False

    for n_day in range(quantity_days):
        string_date = str(datetime.now().date()
                        + timedelta(days=n_day + 1))

        predicted_values = []

        data_for_date = \
            self.get_weather_data_for_date(n_day + 1)

        total_generated_energy = 0
        data_list = self.group_data_by_conditions(data_for_date)

        for i in range(3):
            if not data_list[i].empty:
                model = self.choose_model_by_group(data_list[i], i)
```

```

        predictions = \
            model[0]['model'].predict(data_list[i])

        total_generated_energy += sum(predictions)

        for j in range(len(data_list[i])):
            predicted_values.append(
                {
                    "hour": data_list[i].iloc[j]['t_h'],
                    "quantity_energy": predictions[j],
                    "used_model": model[1]
                }
            )

        predicted_values = sorted(predicted_values,
                                   key=lambda hour_prediction:
                                   hour_prediction['hour'])

        self.predictions[string_date] = predicted_values
        print(f"Data: {string_date}: {total_generated_energy}.")

```

Для можливості зберігання моделі для використання на різних пристроях, реалізовано функцію `save_model()`, яка приймає один необов'язковий параметр – назву моделі прогнозування. Оскільки, авторська модель складається з декількох під моделей, то здійснено реалізацію так, щоб усі файли зберігались у одному архіві за визначеною структурою. Процес збереження здійснюється так, як визначено на етапі моделювання.

Для реалізації дій, які б виконували зворотні операції до попереднього методу, тобто завантаження моделі, створено функцію `load_model()` з обов'язковим параметром, який вказує на ім'я архіву, у якому збережено модель. Алгоритм завантаження реалізовано таким чином, щоб об'єкт класу зберіг усі значення, та можна було б продовжити роботу з пошуку найкращих моделей прогнозування. Варто додати, що функції збереження та завантаження моделі є корисними, тому що можуть бути використаними для резервного копіювання результатів.

Отже, під час розробки, дотримано вимог, які встановлено на попередньому етапі та описано у попередньому розділі, а отримані результати свідчать про коректну його роботу. Крім того, розробка функцій для імпортування та експортування моделі дозволить в подальшому їх зберігати та у разі потреби використовувати для прогнозування на інших сонячних електростанціях.

3.2 Практичне застосування методу

Основним завданням методу прогнозування є розрахунок значення виробленої енергії на сонячній електростанції. Для демонстрації практичного застосування, розроблено систему, де власники сонячних ферм мають змогу контролювати виробництво та прогнозування кількості енергії, а також в яку інтегровано розроблений метод. Крім цього, вона дозволяє виконувати наступні операції:

- реєстрація, перегляд інформації, оновлення даних та видалення сонячних електростанцій;
- здійснення моніторингу за виробництвом та прогнозуванням енергії;
- перегляд доходу електростанції;
- створення моделей прогнозування з різними параметрами з можливістю збереження їх у базі даних чи експорту на зовнішній носій;
- формування прогнозу на одну або кілька діб;
- генерування звіту про прогнозування у форматі XML;
- перегляд, завантаження та видалення моделей прогнозування сонячної електростанції.

Вищенаведені функції забезпечують виконання основних завдань, які постають перед власником сонячної електростанції. Наприклад, формування прогнозу та надсилання звіту можна виконати в декілька кроків, встановивши відповідні налаштування.

Для розробки програми спроектовано базу даних, в якій описано таблиці для збереження інформації для роботи системи. З них можна виокремити таблиці для збереження інформації про користувачів, сонячні електростанції, моделі прогнозування та критерії оцінювання.

Щоб зберігати облікові дані користувачів створено таблицю «users» та її опис наведено у табл. 3.1. Збережена інформація використовуються для процедури авторизації, а також для впорядкування даних у інших таблицях, які відносяться до певного користувача.

Таблиця 3.1 Опис таблиці «users» у базі даних

Поле	Назва поля у таблиці	Тип даних	Призначення
Ідентифікатор сутності	id	Ціле число (int)	Поєднання даних з іншими таблицями
Ім'я користувача	name	Рядок (varchar (45))	Використання при формуванні звітів та інших документів
Прізвище користувача	surname		
Електронна адреса	email		
Пароль облікового запису	password	Рядок байтів (binary (60))	Авторизація у системі
Роль користувача в системі	role	Ціле число (int)	Розподілення доступу до функцій системи

Завданням таблиці «solar_power_plants» є зберігання інформації про сонячні електростанції користувачів, яка буде використовуватись при формуванні моделей прогнозування. Також варто додати, що поле «Час та дата реєстрації» додано для визначення днів та часу найбільшої активності реєстрації користувачів, щоб збільшити потужність системи саме на цей період. Поле «Погодний сервіс» може набувати цілочисельне значення, яке відповідає ідентифікатору відповідного сервісу у системі. Оскільки, в системі реалізовано підтримку двох таких сервісів, то для них присвоєно наступні ідентифікатори:

- погодні дані власної метеостанції – 0;
- OpenWeatherMap – 1;
- Open-Meteo – 2.

Ідентифікатор «0» зарезервовано для підтримки можливості надання власником сонячної електростанції даних про погоду з власної метеостанції. У табл. 3.2 наведено детальний опис та призначення основних полів таблиці «solar_power_plants».

Таблиця 3.2 Опис таблиці «solar_power_plants» у базі даних

Поле	Назва поля у таблиці	Тип даних	Призначення
Ідентифікатор	id	Ціле число (int)	Поєднання даних з іншими таблицями
Ідентифікатор власника системи	user_id		Визначення усіх станцій для одного користувача
Назва сонячної станції	name	Рядок (varchar (45))	Використання при формуванні звітів та використанні системи
Довгота	lon	Число з плаваючою крапкою (double)	Використання при формуванні прогнозу
Широта	lat		
Погодний сервіс	weather_service	Ціле число (int)	
Час та дата реєстрації	registration_date _time	Час і дата (datetime (6))	Для контролю завантаженості системи
Потужність	power	Ціле число (int)	Для формування моделей прогнозування
Дата встановлення	installation_date	Дата (date)	

Особливості структури моделі передбачають збереження її у вигляді кількох файлів, тому для цього спроектовано дві таблиці «models» та «submodels», між якими встановлено зв'язок «багато до одного». В даному випадку, основною таблицею є «models», яка зберігає інформацію про ідентифікатор власний і сонячної електростанції та назву моделі, а таблиця «submodels» має наступні поля:

- ідентифікатор підмоделі;
- ідентифікатор моделі;

- назва файлу;
- символний код підмоделі.

Така структура дасть можливість упорядковувати різну кількість підмоделей без постійної зміни конструкції таблиці, а також оптимізувати роботу бази даних, шляхом зменшення навантаження на запити до БД.

Щоб зберігати коефіцієнти точності для кожної підмоделі, створено таблицю «`statistic_indicators`». Вона поєднана зв'язком «один до одного» з таблицею «`submodels`», а для зв'язку з «`model`» застосовано тип «багато до одного». У табл. 3.3 наведено детальний опис та призначення кожної колонки цієї таблиці.

Таблиця 3.3 Опис таблиці «`statistic_indicators`» у базі даних

Поле	Назва поля у таблиці	Тип даних	Призначення
Ідентифікатор	id	Ціле число (int))	Поєднання даних з іншими таблицями
Ідентифікатор моделі	model_id		Для визначення усіх коефіцієнтів для моделі
Ідентифікатор підмоделі	submodel_id		Для встановлення підмоделі
Коефіцієнт RMSE	RMSE	Десяткове число (decimal (10, 5))	Зберігання критеріїв оцінювання
Коефіцієнт MAE	MAE		
Коефіцієнт детермінації	r2		
Затрачений час	duration	Десяткове число (decimal (10, 3))	

Після створення бази даних реалізовано бізнес-логіку системи та розроблено інтерфейс. Для реалізації цього використано засоби, які визначено на етапі вибору

засобів та описано у попередньому розділі, основними з яких є Python, як основна мова програмування та PyQt з прив'язкою PySide для інтерфейсу.

Розроблена програма має два режими функціонування: користувацький та адміністраторський. Користувач чи адміністратор під час використання системи працює з однією з зареєстрованих сонячних електростанцій. Її можна обрати, використовуючи комбінований список, який розміщено на кожній сторінці. Як особливість, у розділі «Сонячні станції» можна встановити одну станцію за замовчуванням для всіх сторінок, що дозволяє зручніше працювати з обраною електростанцією. Після створення моделі прогнозування чи формування прогнозу у користувача з'являється можливість формування звіту. Нижче детально описано роботу системи.

Після запуску програми на екрані з'являється вікно для авторизації (рис. 3.3), яку користувачу потрібно виконати для отримання доступу до функцій системи. Для цього йому потрібно ввести власну електронну адресу та пароль.

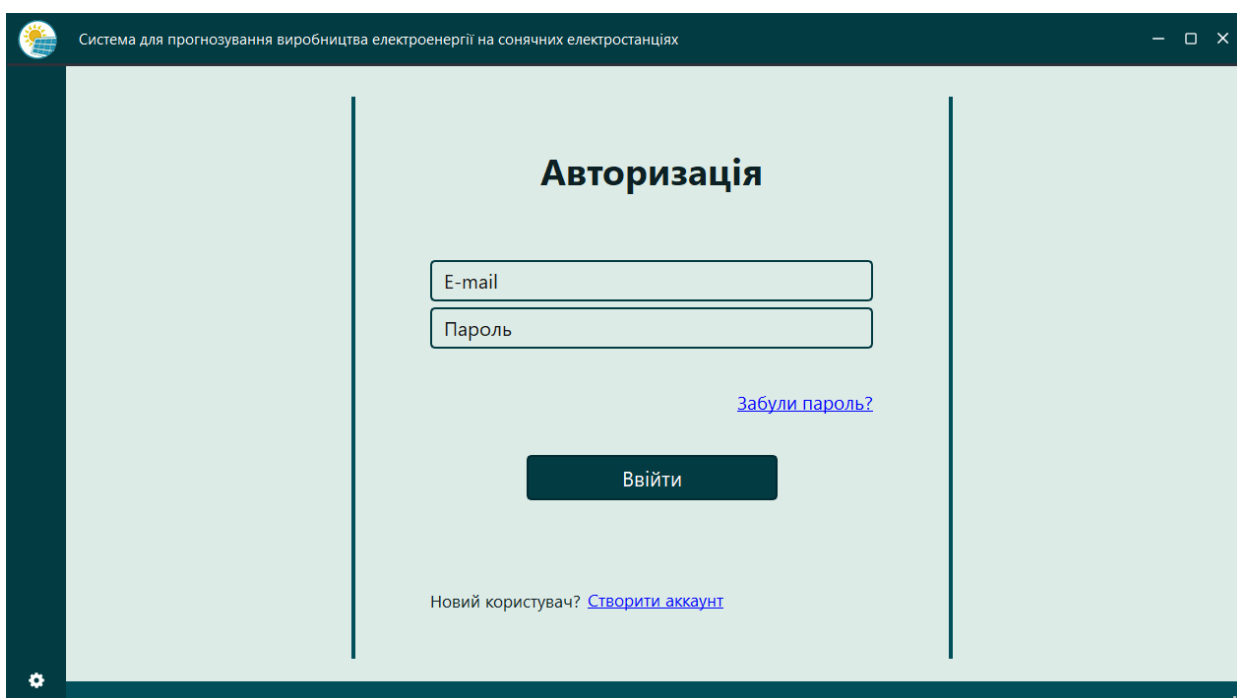


Рисунок 3.3 – Вікно для авторизації користувача

Під час авторизації програма здійснює запит до бази даних для отримання електронної адреси та паролю. Останній зберігається у вигляді хешу, який формується за допомогою функції bcrypt. У випадку виявлення недостовірних

даних на екран виводиться повідомлення про помилку під час входу. На цьому етапі власник сонячної електростанції може змінити пароль чи створити новий аккаунт, перейшовши за зовнішнім посиланням.

Успішно авторизувавшись, у системі завантажуються з бази даних всі необхідні дані для відображення, відповідно до ролі користувача. Спершу власник електростанції потрапляє на домашню сторінку, де може переглянути статистичну інформацію про власні сонячні ферми, вироблену та прогнозовану електроенергію, процес звітності та робочий стан. Для навігації у програмі створено ліве бокове меню з рядом пунктів, назви яких можна переглянути, розгорнувши його. Відображення домашньої сторінки з розгорнутим лівим меню наведено на рисунку 3.4. У користувача з роллю «User» навігаційна панель має наступні пункти: «Головна», «Сонячні станції», «Створення моделей», «Формування прогнозу», «Моделі прогнозування», «Прогноз погоди» та «Вихід», а у адміністратора системи (роль «Admin») додатково відображається пункт «Адміністратор».



	Найменування	Кількість виробленої енергії за попередню добу	Кількість виробленої енергії	XML-звіт	Стан
1	Перша станція	33.38	1074.266	Не надіслано	Працює
2	Продуктивна станція	36.175	2400.87	Немає	Не працює
3	Мала станція	10.093	622.868	Немає	На обслуговуванні
4	Сонячна станція	24.792	1669.444	Надіслано	Працює
5	Нова станція	0.556	37.008	Не надіслано	Працює
6	Зелена станція	1.996	122.293	Немає	Працює
7	Новітня станція	1.759	125.248	Немає	На обслуговуванні

Загалом:
- за добу: 108.946 кВт-год;
- за весь час: 6.064 МВт-год;
- працюючих станцій: 4.

Рисунок 3.4 – Домашня сторінка з лівим меню

Керування сонячними електростанціями здійснюється у розділі «Сонячні станції» (рис. 3.5). У ньому користувач може додавати, переглядати інформацію, редагувати дані та видаляти станцію. Для кожної операції виконується відповідний

запит до бази даних. Також у цьому розділі розміщено комбінований список з назвами сонячних ферм користувача та є можливість встановити обрану електростанцію за замовчуванням. Після виконання даної операції на екрані відображається назва обраної станції та кнопка для скидання. Цю функціональність реалізовано для покращення користування системою з великою кількістю власних зареєстрованих сонячних ферм.

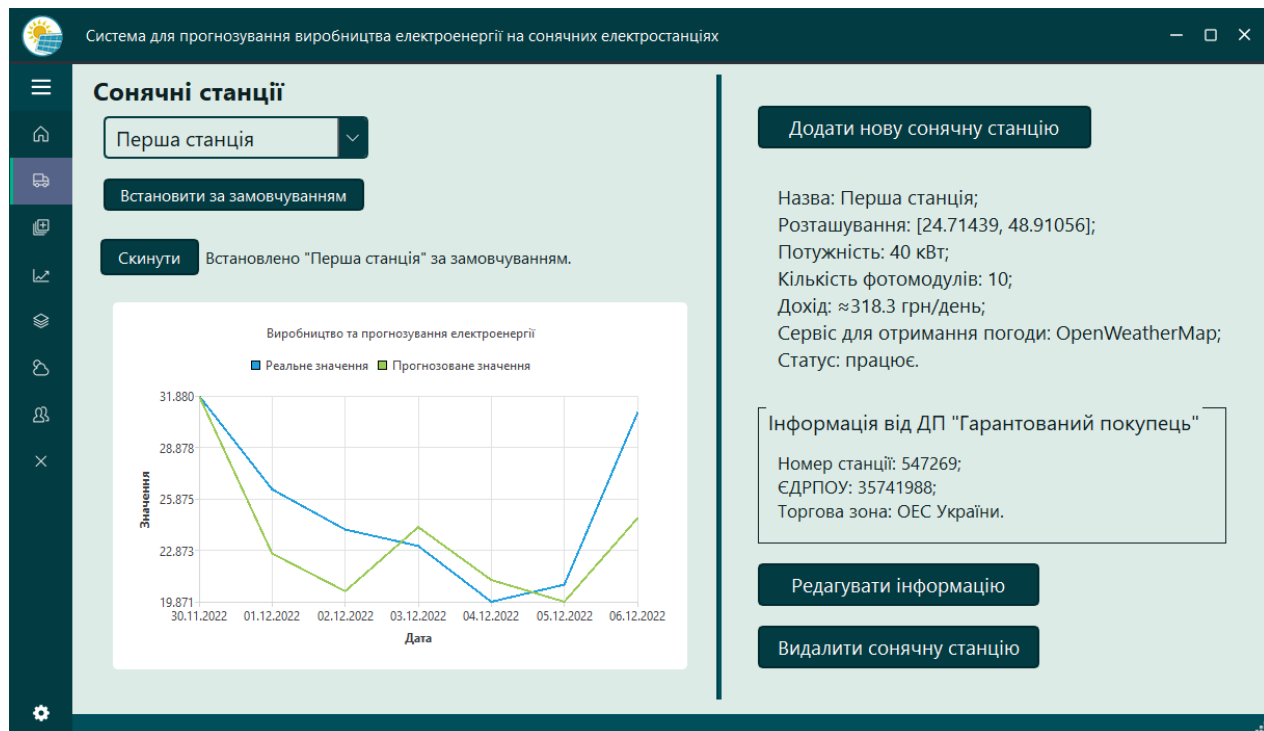


Рисунок 3.5 – Відображення сторінки «Сонячні станції»

Графік відображає дані про реальне та прогнозоване значення за сім крайніх днів, не враховуючи день перегляду. Таке зображення дає можливість візуально аналізувати процес прогнозування. Для додавання нової сонячної електростанції чи оновлення існуючої необхідно натиснути на кнопки «Додати нову сонячну станцію» або «Редагувати інформацію» відповідно. В результаті відкриваються додаткові вікна, де можна ввести інформацію чи змінити наявну. При оновленні даних, у поля, які призначені для опису електростанції вносяться значення, що збережені у базі даних (рис. 3.6). Для значень довготи та широти реалізовано поля вводу для введення дробових чисел з п'ятьма символами після коми, а для потужності та кількості фотомодулів – цілочисельне поле. Список погодних

сервісів завантажується до випадного списку автоматично. У цьому вікні є можливість змінити стан станції. При обиранні «На обслуговуванні» додатково з'являються поля для введення значень годин. Окремим блоком виділено інформацію, яка надається ДП «Гарантований покупець». Поля «Номер станції» та «ЄДРПОУ» реалізовано так, щоб у них можна було ввести тільки числові значення. Крім цього, додано функцію для визначення розташування (географічних координат) ввівши адресу.

Оновлення інформації про сонячну електростанцію

Оновлення інформації про сонячну електростанцію

Назва

Довгота

Широта

Потужність

Кількість фотомодулів

Дата встановлення

Погодний сервіс

Стан

Визначити розташування

Інформація від ДП "Гарантований покупець"

Номер станції

ЄДРПОУ

Торгова зона

Зберегти Скасувати

Рисунок 3.6 – Вікно для оновлення інформації про сонячну електростанцію

Натиснувши на кнопку «Видалити сонячну станцію» на екрані з'явиться діалогове вікно, де потрібно підтвердити видалення обраної електростанції. Після чого станція видаляється з бази даних та на екрані з'являється відповідне повідомлення та оновлюється вміст вікна.

Розділ «Створення моделей» є одним з ключових у системі. Тут користувач може створювати різні моделі для прогнозування виробництва електроенергії шляхом коригування набору даних чи параметрів методу, аналізувати та порівнювати отримані результати. Крім цього, найкращі моделі користувач може зберегти у базі даних, експортувати на зовнішній носій та зберегти результати у вигляді звіту. Для цього, йому необхідно у списку моделей натиснути правою кнопкою миші на потрібну та обрати необхідний пункт з контекстного меню. Інтерфейс вищеописаного розділу наведено на рисунку 3.7.

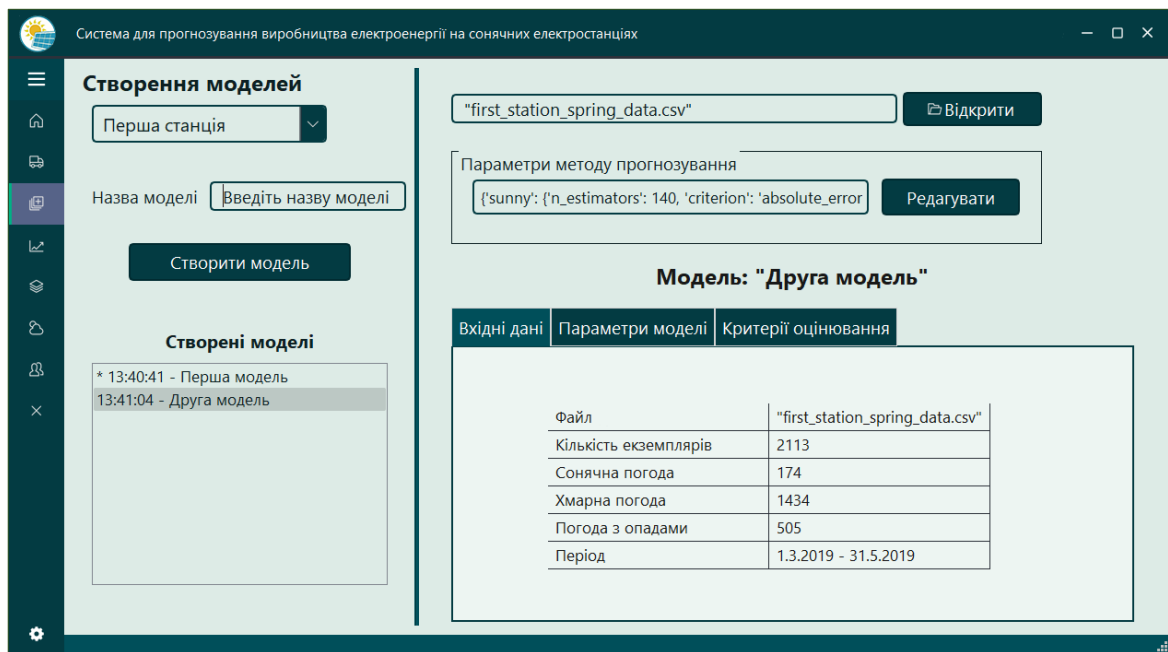


Рисунок 3.7 – Інтерфейс розділу «Створення моделей»

Першим кроком для формування моделі прогнозування є завантаження наборів даних. Для цього потрібно натиснути на кнопку «Відкрити» та обрати попередньо підготовлені файли. На цьому етапі користувач може обрати один або декілька файлів з одним з розширенням CSV. Після вибору файлів вони проходять перевірку на відповідність структури та об'єднуються для наступного етапу.

Далі користувачу потрібно встановити параметри методу прогнозування. Для цього потрібно натиснути на кнопку «Редагувати». В результаті відкриється додаткове вікно з можливістю коригування значень для отримання найкращих результатів прогнозу. Також у вікні розміщено кнопку «Встановити по замовчуванню», результатом натискання на яку є скидання параметрів за

замовчуванням для обраної погодної групи. Збережені параметри будуть наведені на сторінці «Створення моделей».

Наступним кроком потрібно ввести назву моделі для ідентифікації та натиснути кнопку «Створити модель». Для формування моделей використовується розроблений метод прогнозування. Після цього до списку створених моделей додається новий об'єкт з назвою у форматі «* гг.хх.сс – Назва», де позначення зірочкою (*) вказує на те, що модель не збережено у базі даних. Для збереження, експорту чи видалення моделі зі списку потрібно натиснути на праву кнопку миші по назві моделі у списку і вибрати у контекстному меню один з трьох пунктів «Зберегти», «Експортувати» чи «Видалити» відповідно. Кожного разу після створення нової моделі користувач має змогу проаналізувати результат, а саме переглянути інформацію про вхідні дані, параметри методу та коефіцієнти точності. Ця інформація відображається у блоку з панеллю наступних вкладок: «Вхідні дані», «Параметри методу» та «Критерії оцінювання».

Для отримання прогнозу виробництва електроенергії створено розділ «Формування прогнозу» (рис. 3.8). У ньому користувач може розрахувати значення кількості згенерованої енергії використовуючи наявні моделі та на основі результатів сформувати XML-звіт.

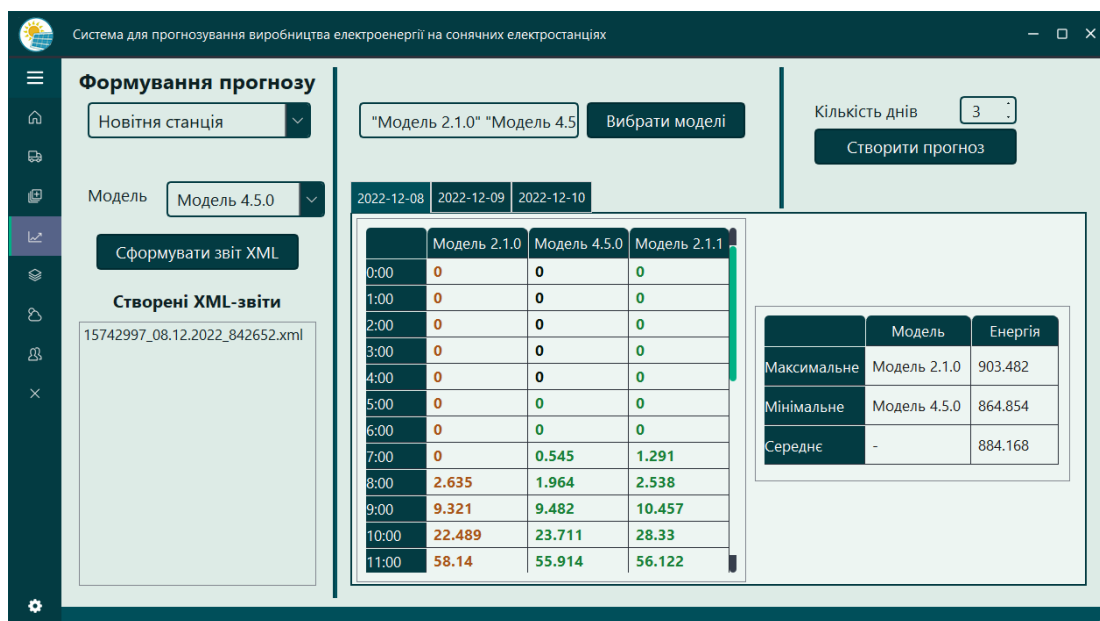


Рисунок 3.8 – Відображення сторінки «Формування прогнозу» (з результатами прогнозування)

Для отримання прогнозованого значення, в залежності від обраної сонячної електростанції, у системі завантажується список наявних моделей. За замовчуванням для прогнозування обрано усі моделі. Вибрати конкретні можна у списку, який відкривається у додатковому вікні (рис. 3.9), що з'являється після натискання на кнопку «Вибрати», яка розташована біля їхнього переліку. Список містить назви моделей, де біля кожної є елемент прапорця. Також у даному вікні є кнопка, натискаючи на яку можна вибирати усі моделі або виконувати протилежну дію. Для збереження свого вибору потрібно натиснути на кнопку «Зберегти», а для анулювання свого рішення – «Скасувати».

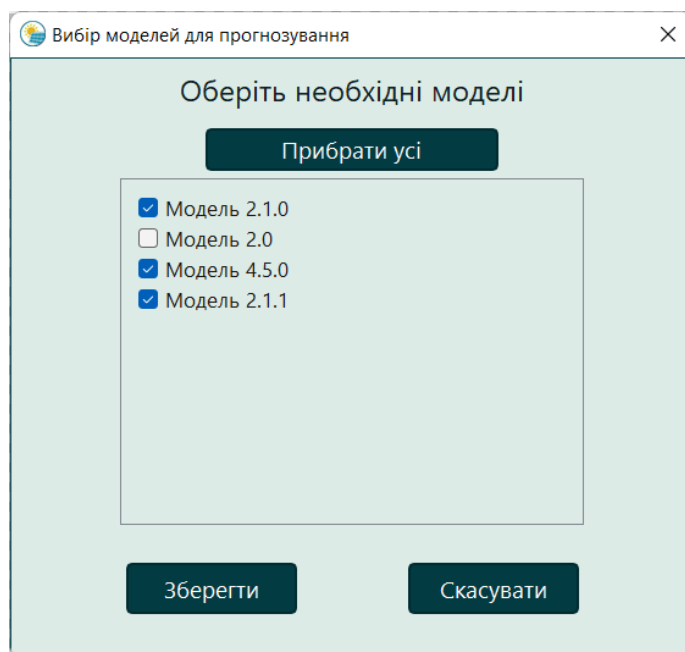


Рисунок 3.9 – Діалогове вікно для вибору моделей прогнозування

Важливою функцією цієї частини програми є формування прогнозу щодо кількості електроенергії на наступну добу. Особливістю методу є те, що дані розбиваються погодинно і створюється прогноз для кожної години окремо та як результат обчислюється сума отриманих значень. Обравши моделі, користувачу потрібно натиснути на кнопку «Отримати прогноз». Процес розрахунку кількості енергії супроводжується індикатором виконання та текстовими підказками. У системі передбачено прогнозування на кілька днів, кількість яких можна попередньо обрати. Після закінчення процесу прогнозування на екрані демонструються таблиці з результатами та з'являється можливість сформувати

XML-звіт, який можна переглянути, надіслати або видалити. У таблиці значення енергії виводиться різними кольорами, які вказують на застосовану модель. Зеленим позначено енергію, яку розраховано за моделлю для пори року та погоди, коричневим – за загальною погодною моделлю, а чорним позначається використання моделі, яку побудовано на усіх даних. Значення кольору встановлюється залежно від коду моделі. Дане рішення реалізовано для візуального сприйняття результату прогнозування. Таблиця, де наведено максимальне та мінімальне значення необхідна, щоб аналітик при формуванні звіту зміг обрати той прогноз, який найбільше корелює зі значенням середнього.

Звіт формується відповідно до інструкції, яку надано ДП «Гарантований покупець». У ній наведено інформацію про періодичність надання даних, дії для аналізу помилок тощо, а також наведено детальний опис структури файлу звіту з прикладом у вигляді XML-файлу [48]. Варто відзначити простоту та розгорнутість опису файлу, а саме його важливих структурних елементів. Чітка документація, яка подана у файлі дозволить зменшити кількість неправильно поданих даних та проблем при його створенні. Також вибір формату XML дозволить аналітику та комп'ютеру читати дані без проблем. Крім цього для користувача формується таблиця, де наведено мінімальне, максимальне та середнє значення.

Розділ «Список моделей» призначений для контролю та керування наявними моделями прогнозування для кожної електростанції. Дані представлено у вигляді таблиці, де можна переглянути назву та дату створення. Крім цього, натиснувши на кнопку «Детальніше» можна переглянути докладну інформацію про створену модель, в яку входить деталі про набір даних та параметри методу прогнозування. Також модель можна видалити, для цього потрібно клацнути на кнопку «Видалити». Після цього з'явиться діалогове вікно, де потрібно буде підтвердити операцію.

Користуючись категорією «Прогноз погоди» у користувача є можливість завантажити прогнози для розташування сонячної станції, використовуючи різні погодні сервіси та порівнювати отриману інформацію. Окрім ключових сторінок у

системі є додаткові елементи, які дозволяють нею керувати. Наприклад зліва є додатковий блок «Налаштування», де можна здійснити наступні операції:

- переглянути довідку;
- написати листа до технічної підтримки;
- перейти до налаштувань;
- вийти з свого аккаунту в системі.

3.3 Визначення економічної ефективності застосування

Для обґрунтування економічної ефективності змодельовано задачу, де обчислюється дохід власника сонячної електростанції потужністю 40 кВт за один місяць з врахуванням штрафів, які спричинені подачею неточних прогнозів. Для порівняння якості прогнозування використано розроблений метод та метод Лінійної регресії. На рисунку 3.10 наведено графік, який побудовано на основі даних, які наведено у табл. Е.1 у додатку Е, який демонструє розмір похибки прогнозування двох вищевказаних методів.

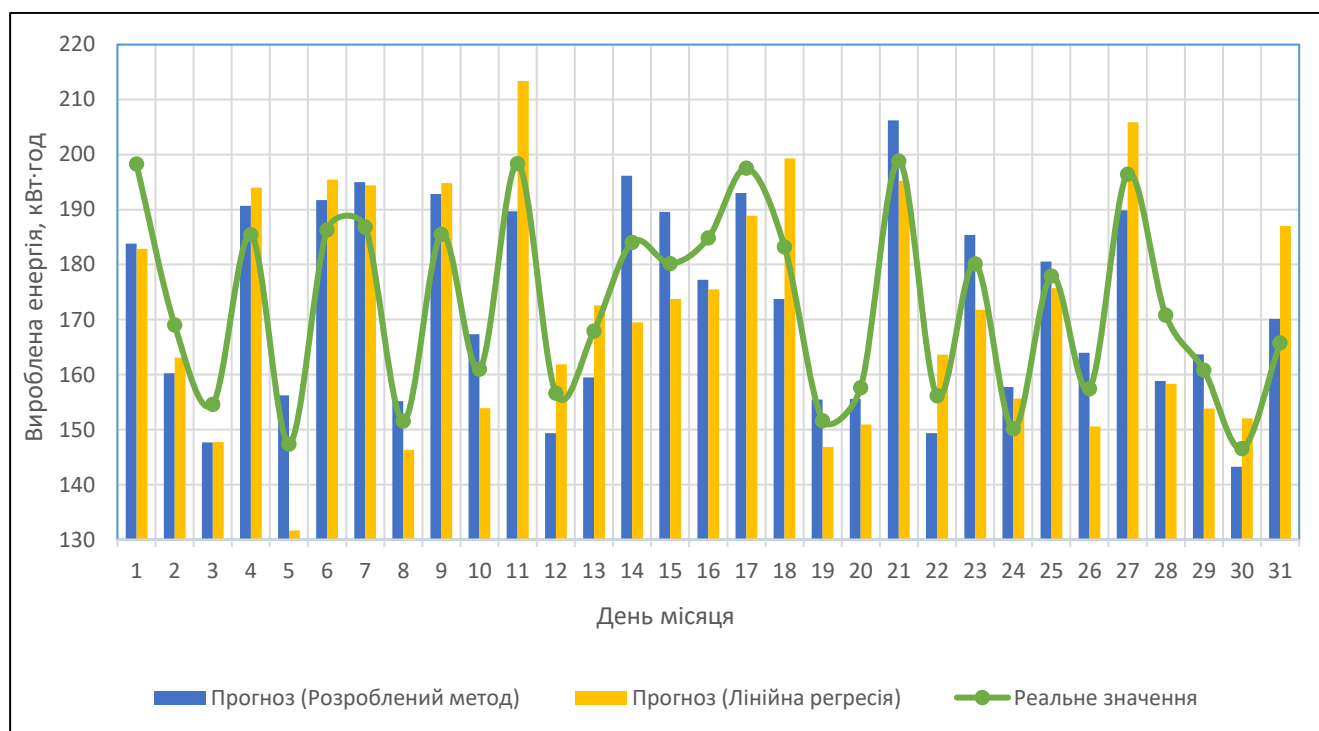


Рисунок 3.10 – Графік виробленої енергії в порівнянні з прогнозованими значеннями

На графіку видно, що різниця між реальним значенням та прогнозованим є меншою використовуючи власний метод. Для розрахунку цієї неточності обчислено різниці між реальним та прогнозованими значеннями та їхні відносні похибки. Результати обчислення наведено у табл. Е.2 у додатку Е. Підрахувавши кількість випадків, коли похибка є більшою за 5%, отримано значення 4 для розробленого методу та 7 для методу Лінійна регресія. Далі обчислено штраф для кожного такого дня та загальну суму. Значення «зеленого» тарифу для обраної станції дорівнює – 657,63 коп./кВт·год (без ПДВ) [49]. Під час обчислень враховано вплив позитивних та негативних небалансів на дохід. При позитивному – надлишок енергії купують за мінімальною ціною, а при негативному – виробник повинен придбати енергію за власний кошт та за завищеною ціною [50]. Щоб врахувати це при обчисленнях, значення суми, яку отримає власник станції при позитивному небалансі за 1 кіловат енергії дорівнює 20% від вартості «зеленого» тарифу, що становить 131,53 коп./кВт·год (без ПДВ), а при негативному втричі більший –1972,89 коп./кВт·год (без ПДВ). Фінансові втрати при позитивному небалансі обчислено за формулою (3.1), тобто скільки недоотримав власник станції через неточний прогноз.

$$\text{Втрати}_{\text{поз}} = (\text{Енергія}_p - \text{Енергія}_n) \cdot (\text{ЗТ} - \text{Вартість}_n), \quad (3.1)$$

де $\text{Втрати}_{\text{поз}}$ – втрати при позитивному небалансі,

Енергія_p – реальне значення виробленої електроенергії,

Енергія_n – прогнозоване значення виробленої електроенергії,

ЗТ – значення зеленого тарифу,

Вартість_n – вартість одного кіловата при позитивному небалансі (131,53 коп.).

Фінансові втрати при негативному небалансі обчислено за формулою (3.2).

$$\text{Втрати}_{\text{нег}} = |\text{Енергія}_p - \text{Енергія}_n| \cdot (\text{Вартість}_n), \quad (3.2)$$

де $\text{Втрати}_{\text{нег}}$ – втрати при негативному небалансі,

Енергія_р – реальне значення виробленої електроенергії,

Енергія_п – прогнозоване значення виробленої електроенергії,

Вартість_н – вартість одного кіловата при негативному небалансі (1972,89 коп.).

Підсумки розрахунків наведено у табл. 3.4. Загальну суму втрат наведено для стовідсоткового відшкодування, але враховуючи правила, на даний час втрати будуть становити тільки 20%.

Таблиця 3.4 Сума втрат за небаланси протягом одного місяця

Метод	Дата	Реальне значення, кВт·год	Прогнозоване значення, кВт·год	Похибка, %	Втрати, (грн)	Загалом (грн)
Розроблений метод	01.05.2020	198,28	183,7909	6,81	76,228	553,28
	05.05.2020	147,39	156,2357	5,66	174,516	
	14.05.2020	184,03	196,1762	6,19	239,631	
	28.05.2020	170,81	158,8533	6,54	62,905	
Лінійна регресія	01.05.2020	198,28	182,8507	7,22	81,174	1339,173
	05.05.2020	147,39	131,6653	9,64	82,728	
	11.05.2020	198,35	213,3531	7,03	295,995	
	14.05.2020	184,03	169,5001	7,32	76,442	
	18.05.2020	183,22	199,2963	8,07	317,168	
	28.05.2020	170,81	158,3124	6,82	65,750	
	31.05.2020	165,75	187,0343	11,38	419,916	

Отже, підбиваючи підсумки видно, що витрати компанії на виплату штрафів, отримуючи прогноз за допомогою методу Лінійна регресія є у 2,42 рази більшими ніж з використанням розробленого методу. Збережені кошти власник сонячної електростанції може використати на модернізацію чи обслуговування обладнання. До того ж очевидно, що дохід компанії залежить від кількості виробленої енергії, тому аналітики на основі накопичених даних про генерацію електроенергії протягом року можуть скласти прогнозований план продукування по місяцях. Але

варто врахувати, що у період часу, коли сонячна активність є найбільшою, різко погіршилась погода, то генерація, а отже і дохід компанії пропорційно зменшується. Тому в таких ситуаціях варто з максимальною обережністю відноситись до прогнозування кількості виробленої енергії сонячними станціями, щоб не втратити велику частину доходу через сукупність цих чинників.

Також результат розробки буде розміщено на відкритому ресурсі у мережі Інтернет. З одного боку, це дасть змогу власникам сонячних електростанцій випробовувати створений метод та порівнювати результати прогнозування з тими, які вони отримують, використовуючи власні моделі. Звіти використання моделей на електростанціях, які мають різне географічне розташування дозволять отримати більше інформації про їхню якість. З іншого боку, те, що методи прогнозування будуть у відкритому доступі, дозволить розробникам та дослідникам їх вдосконалювати для досягнення кращих результатів. Також створені моделі можна застосовувати в навчальних цілях у профільних освітніх закладах.

ВИСНОВКИ

В результаті проведеної роботи, створено теоретичну модель та програмну реалізацію методу прогнозування виробництва електроенергії на сонячних станціях на основі застосування методу Random Forest для розв'язання задач, пов'язаних з проблемою подання неточних прогнозів виробниками ВДЕ.

Для реалізації поставленої мети здійснено аналіз низки літературних джерел, до яких входять книги та наукові статті обраної тематики, постанови та закони, що регулюють ринок енергетики, в результаті чого отримано інформацію про зростання використання відновлювальних джерел енергії, а також критерії та правила нарахування відшкодування за небаланси.

Перед створенням власної реалізації механізму прогнозування, проаналізовано та встановлено критерії оцінювання якості моделі, в ході чого встановлено, що найбільш поширеними є показники: RMSE, MAE та коефіцієнт детермінації (R^2). Також проведено дослідження методів прогнозування Total Sky Imagers, методів на основі Numerical Weather Prediction Models та комбінованих. Для кожного методу виявлено його особливості обчислення, дані, які впливають на якість сформованих моделей прогнозування та їхні похибки, а створені моделі порівняно за встановленими критеріями. В результаті визначено, що для зменшення похибки варто створювати комбінований метод, основою якого є Random Forest. Проаналізувавши відомі способи вирішення проблеми, сформовано ряд кроків для успішного власного вирішення.

Наступним етапом було створення теоретичної моделі методу прогнозування, в результаті якого встановлено вимоги до вхідних та вихідних даних, проаналізовано роботу методу та обґрунтовано ефективність власних рішень. Також здійснено аналіз проблем, які впливають на точність методу та способи їхнього вирішення.

Для наведення поведінки алгоритму, створено діаграму станів з детальним описом кожної схеми, після чого, встановлено набір інструментів розробки. Основною мовою програмної реалізації методу прогнозування та системи обрано

Python з рядом допоміжних бібліотек та модулів, а для розробки користувацького інтерфейсу PyQt з PySide.

Реалізація механізму прогнозування здійснювалась відповідно до створеної діаграми станів, в результаті чого створено засіб, де користувачу потрібно завантажити набір даних про генерацію енергії та ввести інформацію про сонячну станцію та параметри для кожної моделі прогнозування. Крім цього, для наведення одного з прикладів практичного застосування, створено програмну реалізацію інформаційної системи для власників сонячних електростанцій, у яку інтегровано метод прогнозування, а також розроблено ряд функцій для створення та упорядкування різної кількості моделей, формування та надсилання звіту тощо.

На завершення, визначено економічну ефективність, а саме обґрунтовано з наведенням обчислень дієвість використання розробленого методу прогнозування сонячної енергії, що дасть можливість власникам сонячних ферм мінімізувати фінансові втрати, які негативно впливають на дохід компанії.

Отже, незважаючи на значну кількість методів прогнозування виробленої сонячної енергії, створення нового методу дозволить краще дослідити їхнє застосування та розробку. Також на основі нових методів можна створювати кращі, вдосконалюючи їх, що дозволяє постійно вдосконалювати якість моделі. Крім цього, послідовне збільшення розміру відшкодування за неточні прогнози буде збільшувати попит на якісні методи прогнозування, а власників сонячних електростанцій спонукати інвестувати кошти в їх розробку.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] M. K. Evans, *Practical Business Forecasting*, Oxford: Blackwell, 2003.
- [2] A. Marsh, «The First Transatlantic Telegraph Cable Was a Bold, Beautiful Failure», 31 жовтня 2019. [Електронний ресурс]. Режим доступу: <https://spectrum.ieee.org/the-first-transatlantic-telegraph-cable-was-a-bold-beautiful-failure>. [Дата звернення: 12 жовтня 2022].
- [3] Міністерство енергетики, «Інформація про виробничу діяльність електроенергетичних підприємств ПЕК за 12 місяців 2021 року (оперативні дані)», січень 2022. [Електронний ресурс]. Режим доступу: http://mpe.kmu.gov.ua/minugol/control/uk/publish/article?art_id=245619746&cat_id=245183225. [Дата звернення: 11 січня 2022].
- [4] І. Сотник, Є. Коваленко, А. Мартим'янов та М. Нікуліна, «Економічна конкурентноспроможність проєктів «Зеленого» енергобізнесу в Україні», *Науковий вісник ІФНТУНГ*, с. 21-36, 2022.
- [5] Управління комунікації та зв'язків з громадськістю, «Статистика розвитку відновлюваної електроенергетики на кінець I півріччя 2021 року», 7 липня 2021. [Електронний ресурс]. Режим доступу: <https://saee.gov.ua/uk/news/3930>. [Дата звернення: 4 жовтня 2021].
- [6] Укрінформ, «Гарантований покупець оштрафував виробників «зеленої» енергетики за небаланси», Укрінформ, 23 лютого 2021. [Електронний ресурс]. Режим доступу: <https://www.ukrinform.ua/rubric-economy/3195791-garantovanij-pokupec-ostrafuvav-virobnikiv-zelenoi-energetiki-za-nebalansi.html>. [Дата звернення: 10 жовтня 2021].
- [7] «Закон України «Про внесення змін до деяких законів України щодо удосконалення умов підтримки виробництва електричної енергії з альтернативних джерел енергії», 21 липня 2020. [Онлайновий]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/810-20>. [Дата звернення: 20 жовтня 2021].

- [8] D. Chaturvedi and Isha, «Solar Power Forecasting: A Review», *International Journal of Computer Applications*, с. 29-39, 6 липня 2016.
- [9] М. П. Кузнєцов та О. В. Лисенко, «Можливості короткотермінового прогнозування сонячної енергії», *Відновлювальна енергетика*, с. 25-28, 2017.
- [10] D. Kaplan, «Stats for Data Science», 2019.
- [11] D. Christie and S. Neill, «Measuring and Observing the Ocean Renewable Energy Resource», in *Comprehensive Renewable Energy*, Elsevier, 2022, с. 149-175.
- [12] D. S. KumariKarunasingha, «Root mean square error or mean absolute error? Use their ratio as well», *Information Sciences*, с. 609-629, березень 2022.
- [13] P. Schneider and F. Xhafa, «Anomaly detection: Concepts and methods», in *Anomaly Detection and Complex Event Processing over IoT Data Streams*, Mara Conner, 2022, с. 49-66.
- [14] J. Fernando, «R-Squared Formula, Regression, and Interpretations», Investopedia, 12 вересня 2021. [Електронний ресурс]. Режим доступу: <https://www.investopedia.com/terms/r/r-squared.asp>. [Дата звернення: 6 жовтня 2022].
- [15] L. Fahrmeir, R. Künstler, I. Pigeot and G. Tutz, *Statistik. Der Weg zur Datenanalyse*, Berlin/Heidelberg: Springer Spektrum, 2016.
- [16] D. Chicco, M. J. Warrens and G. Jurman, «The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation», *PeerJ Computer Science*, 5 липня 2021.
- [17] UK Power Networks, «Photovoltaic (PV) Solar Panel Energy Generation data - London Datastore Logo», London Datastore Logo, 2016. [Електронний ресурс]. Режим доступу: <https://data.london.gov.uk/dataset/photovoltaic--pv--solar-panel-energy-generation-data>. [Дата звернення: 28 березня 2022].
- [18] Open-Meteo, «Free Open-Source Weather API», [Електронний ресурс]. Режим доступу: <https://open-meteo.com/en>. [Дата звернення: 12 жовтня 2022].

- [19] V. Kostylev and A. Pavlovski, «Solar Power Forecasting Performance – Towards Industry Standards», 2011.
- [20] M. AlShafeey and C. Csáki, «Evaluating neural network and linear regression photovoltaic power forecasting models based on different input methods», *Elsevier*, с. 7601-7603, 2021.
- [21] U. Kumar Das, K. Soon Tey, M. Seyedmahmoudian, M. Yamani Idna Idris, S. Mekhilef, B. Horan and A. Stojcevski, «SVR-Based Model to Forecast PV Power Generation under Different Weather Conditions», *ResearchGate*, с. 1-9, 2017.
- [22] Scikit-learn. developers, «scikit-learn: machine learning in Python», [Електронний ресурс]. Режим доступу: <https://scikit-learn.org/stable/>. [Дата звернення: 12 квітня 2022].
- [23] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*, Springer, 2017.
- [24] L. Breiman, «Random Forests», *Machine Learning*, с. 5-32, 2001.
- [25] TIBCO Software, «What is a Random Forest?», [Електронний ресурс]. Режим доступу: <https://www.tibco.com/reference-center/what-is-a-random-forest>. [Дата звернення: 4 листопада 2022].
- [26] «Набір даних для визначення якості загальних моделей». Режим доступу: https://raw.githubusercontent.com/PavloDrabchuk/data-sets/main/data_ymca.csv.
- [27] M. Meng and C. Song, «Daily Photovoltaic Power Generation Forecasting Model Based on Random Forest Algorithm for North China in Winter», *Sustainability*, 13 березня 2020.
- [28] «Набір даних про виробництво електроенергії на сонячній електростанції Мадриду». Режим доступу: https://raw.githubusercontent.com/PavloDrabchuk/data-sets/main/madrid_data.csv.
- [29] «Набір даних станції Forest Road». Режим доступу: https://raw.githubusercontent.com/PavloDrabchuk/data-sets/main/forest_road_data.csv.

- [30] «Набір даних про виробництво електроенергії на сонячній електростанції в Рейк'явіку». Режим доступу: https://raw.githubusercontent.com/PavloDrabchuk/data-sets/main/reykjavik_data.csv.
- [31] Object Management Group, Unified Modeling Language. Version 2.5.1, Object Management Group, 2017.
- [32] MKLabs Co.,Ltd, «Introduction - StarUML documentation», MKLabs Co.,Ltd, [Електронний ресурс]. Режим доступу: <https://docs.staruml.io/>. [Дата звернення: 8 листопада 2022].
- [33] S. Guido and A. Müller, Introduction to Machine Learning with Python, United States of America: O'Reilly Media, 2017.
- [34] Pandas, «pandas - Python Data Analysis Library», [Електронний ресурс]. Режим доступу: <https://pandas.pydata.org/about/>. [Дата звернення: 4 вересня 2022].
- [35] NumPy, «NumPy», [Електронний ресурс]. Режим доступу: <https://numpy.org/>. [Дата звернення: 4 вересня 2022].
- [36] Scikit-learn, «scikit-learn: machine learning in Python», [Електронний ресурс]. Режим доступу: <https://scikit-learn.org/stable/index.html>. [Дата звернення: 4 вересня 2022].
- [37] OpenWeather, «Current weather and forecast - OpenWeatherMap», [Електронний ресурс]. Режим доступу: <https://openweathermap.org/>. [Дата звернення: 8 листопада 2022].
- [38] Joblib, «Joblib: running Python functions as pipeline jobs», [Електронний ресурс]. Режим доступу: <https://joblib.readthedocs.io/en/latest/>. [Дата звернення: 4 вересня 2022].
- [39] Python Software Foundation, «zipfile – Work with ZIP archives – Python 3.10.6 documentation», [Електронний ресурс]. Режим доступу: <https://docs.python.org/3/library/zipfile.html>. [Дата звернення: 4 вересня 2022].

- [40] Riverbank Computing, «What is PyQt?», Riverbank Computing, [Електронний ресурс]. Режим доступу: <https://riverbankcomputing.com/software/pyqt/>. [Дата звернення: 12 жовтня 2022].
- [41] Python Software Foundation, «configparser – Configuration file parser», [Електронний ресурс]. Режим доступу: <https://docs.python.org/3.10/library/configparser.html>. [Дата звернення: 12 жовтня 2022].
- [42] Python Software Foundation, «bcrypt - PyPi», [Електронний ресурс]. Режим доступу: <https://pypi.org/project/bcrypt>. [Дата звернення: 12 жовтня 2022].
- [43] Python Software Foundation, «mysql-connector-python 8.0.31 - PyPi», [Електронний ресурс]. Режим доступу: <https://pypi.org/project/mysql-connector-python>. [Дата звернення: 12 жовтня 2022].
- [44] Python Software Foundation, «smtplib — SMTP protocol client», [Електронний ресурс]. Режим доступу: <https://docs.python.org/3/library/smtplib.html>. [Дата звернення: 12 жовтня 2022].
- [45] Python Software Foundation, «email — An email and MIME handling package», [Електронний ресурс]. Режим доступу: <https://docs.python.org/3/library/email.html>. [Дата звернення: 12 жовтня 2022].
- [46] Python Software Foundation, «shutil — High-level file operations», [Електронний ресурс]. Режим доступу: <https://docs.python.org/3/library/shutil.html>. [Дата звернення: 12 жовтня 2022].
- [47] The Qt Company Ltd., «QChart - Qt for Python», The Qt Company Ltd., [Електронний ресурс]. Режим доступу: <https://doc.qt.io/qtforpython/PySide6/QtCharts/QChart.html>. [Дата звернення: 8 листопада 2022].
- [48] ДП «Гарантований покупець», «Інструкція з надання прогнозних даних», 2021. [Електронний ресурс]. Режим доступу: https://gpee.com.ua/get_document/82. [Дата звернення: 7 липня 2022].
- [49] Національна комісія, що здійснює державне регулювання у сферах енергетики та комунальних послуг, «Постанова №1237 Про встановлення

«зелених» тарифів на електричну енергію, вироблену генеруючими установками приватних домогосподарств», 30 вересня 2022.

[50] М. Гріцишина, «Правила врегулювання небалансів на ринку електричної енергії: що змінилося», Юридична Газета, 23 лютого 2021. [Електронний ресурс]. Режим доступу: <https://jur-gazeta.com/publications/practice/energetichne-pravo/pravila-vregulyuvannya-nebalansiv-na-rinku-elektrichnoyi-energiyi-shcho-zminilosya.html>. [Дата звернення: 4 жовтня 2021].

ДОДАТОК А

Програмний код для побудови моделей та обчислення критеріїв якості алгоритмів

```
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn import metrics

from sklearn import linear_model
from sklearn.svm import SVR
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import r2_score

import math
import time

def evaluate_statistic_indicators(test_value, predict_value):
    errors = abs(predict_value - test_value)

    MAE = np.mean(errors)
    RMSE = math.sqrt(metrics.mean_squared_error(
        test_value,
        predict_value))
    r2 = round(r2_score(test_value, predict_value), 6)

    return MAE, RMSE, r2

def print_statistic_indicators(test_value, predict_value):
    MAE, RMSE, r2 = evaluate_statistic_indicators(
        test_value,
        predict_value)

    print(f"Mean Absolute Error: {round(MAE, 4)} Вт."
          f" (0 - Найкраще значення).")
    print(f"RMSE: {round(RMSE, 4)} Вт. (0 - Найкраще значення).")
    print(f"R2: {round(r2, 4)}. (1 - Найкраще значення).")

def create_train_and_test_samples(folder, filename):
    raw_input_data = pd.read_csv(f"{folder}/{filename}.csv")
    raw_input_data.drop(
        ['Site', 'DateTime', 't_date', 't_time'],
        inplace=True,
        axis=1)

    labels = np.array(raw_input_data['Energy'])
    raw_input_data = raw_input_data.drop(labels='Energy', axis=1)
```

```

train_data, test_data, train_labels, test_labels =
    train_test_split(
        raw_input_data,
        labels,
        test_size=0.25,
        random_state=42)
return train_data, test_data, train_labels, test_labels

def create_model(
    model_info,
    train_data,
    test_data,
    train_labels,
    test_labels):
    start_p = time.time()

    model = list(model_info.values())[0]
    model.fit(train_data, train_labels)
    predictions = model.predict(test_data)

    finish_p = time.time()

    execution_time = finish_p - start_p

    print(f"Назва методу: {list(model_info.keys())[0]}.")
    print_statistic_indicators(test_labels, predictions)
    print(f"Час виконання ({list(model_info.keys())[0]}):"
          f"{round(execution_time, 2)} секунд.\n")

    result_coefficients = list(evaluate_statistic_indicators(
        test_labels, predictions))
    result_coefficients.append(round(execution_time, 2))

    return result_coefficients

data_c = {}
models_info = [
    {
        «Лінійна регресія»: linear_model.LinearRegression()
    },
    {
        «Метод опорних векторів»: SVR(
            kernel="rbf",
            C=95,
            gamma=0.15,
            epsilon=0.12)
    },
    {
        «Випадковий ліс»: RandomForestRegressor(
            n_estimators=240,

```

```

random_state=40,
bootstrap=True)
    }
]

filename = «data_ymca»
train_data, test_data, train_labels, test_labels =
create_train_and_test_samples("data", filename)

train_data_describe = train_data.describe()
test_data_describe = test_data.describe()

# print(train_data_describe)
# print(test_data_describe)

for model in models_info:
    data_c[list(model.keys())[0]] = create_model(
                                                model,
                                                train_data,
                                                test_data,
                                                train_labels,
                                                test_labels)

result_table = pd.DataFrame(data_c,
                             index=['MAE', 'RMSE', 'R2', 'Час виконання'])

print("Зведена таблиця результатів")
print(result_table)

```

ДОДАТОК Б

Показники ефективності моделей, які сформовано для вибірки різних розмірів для кожної з груп

Таблиця Б.1 Показники ефективності для набору даних різного розміру

Група	Розмір вибірки, кількість екземплярів	RMSE, Вт	MAE, Вт	R ²
Сонячна погода	24 (1 доба)	1,4481	1,4481	0,612
	48 (2 доби)	488,33	356,808	0,803
	72 (3 доби)	193,546	141,502	0,972
	168 (1 тиждень)	182,911	134,554	0,973
	336 (2 тижні)	143,98	105,728	0,984
	720 (30 діб)	151,321	105,361	0,984
	1271 (≈ 53 доби)	127,083	94,097	0,986
Хмарна погода	24 (1 доба)	206,518	126,586	0,233
	48 (2 доби)	113,703	100,459	0,905
	72 (3 доби)	235,028	186,295	0,812
	168 (1 тиждень)	232,236	158,142	0,891
	336 (2 тижні)	220,209	108,487	0,889
	720 (30 діб)	126,53	45,7173	0,922
	1434 (≈ 60 діб)	40,22	19,603	0,973
Погода з опадами	24 (1 доба)	179,394	135,288	0,904
	48 (2 доби)	194,463	158,677	0,968
	72 (3 доби)	230,957	163,522	0,938
	168 (1 тиждень)	159,973	102	0,97
	336 (2 тижні)	141,656	92,9335	0,97
	720 (30 діб)	188,932	105,703	0,941
	1077 (≈ 45 діб)	102,228	60,7498	0,978

ДОДАТОК В

Діаграми складених станів алгоритму прогнозування виробництва сонячної електроенергії

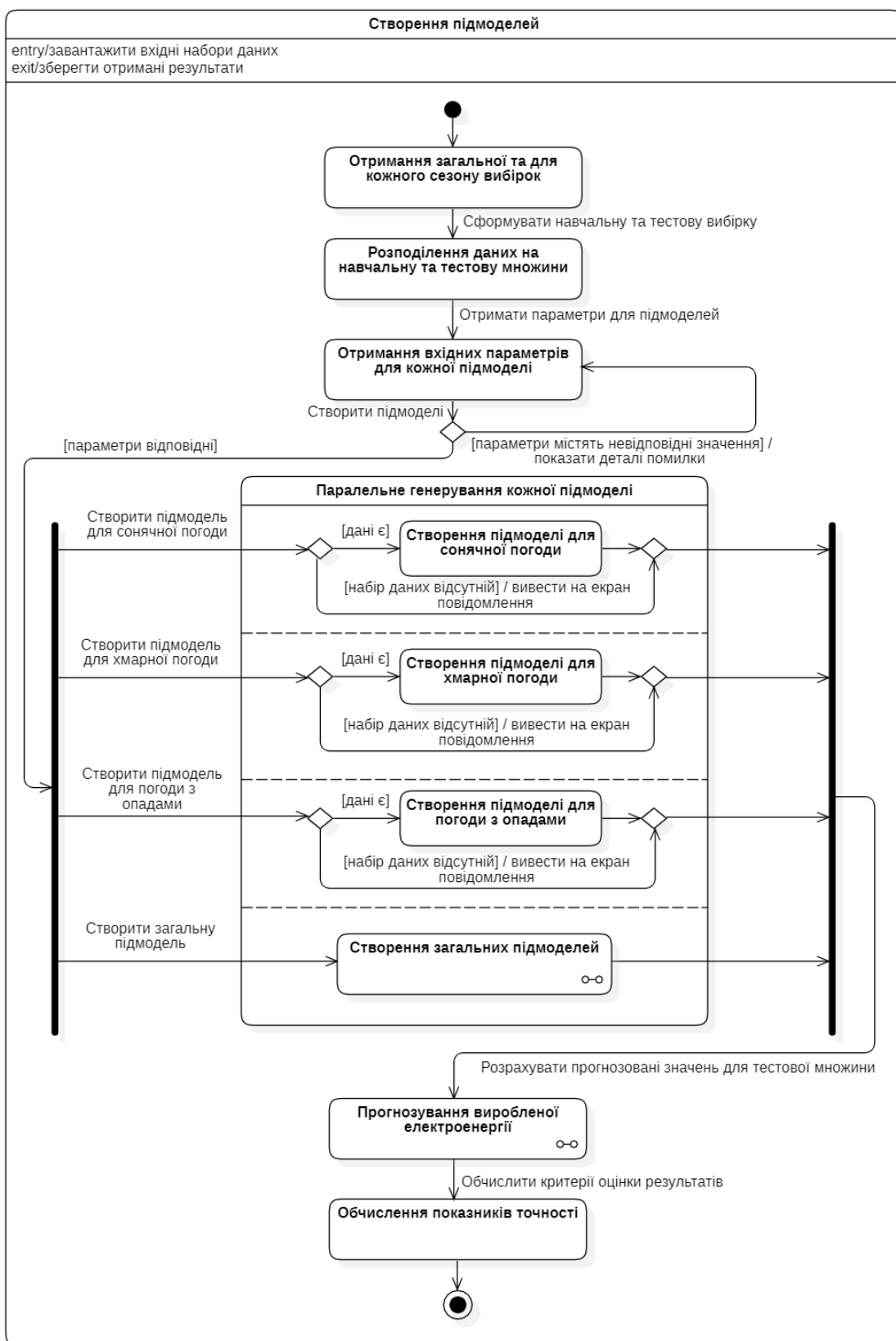


Рисунок В.1 – Діаграма складеного стану «Створення підмоделей»

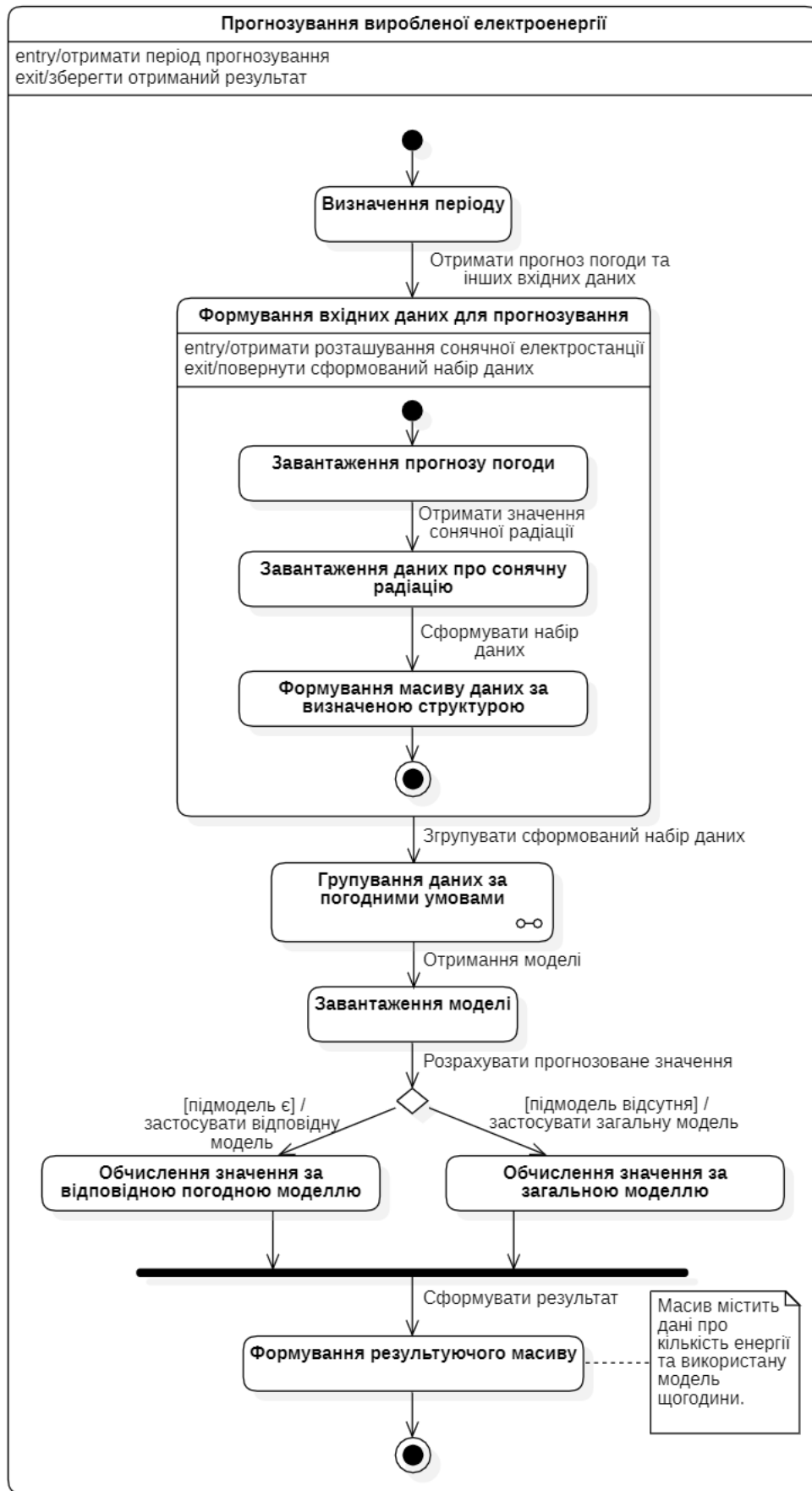


Рисунок В.2 – Діаграма композитного стану «Прогнозування виробленої електроенергії»

ДОДАТОК Г

Алгоритмічні моделі ключових алгоритмів

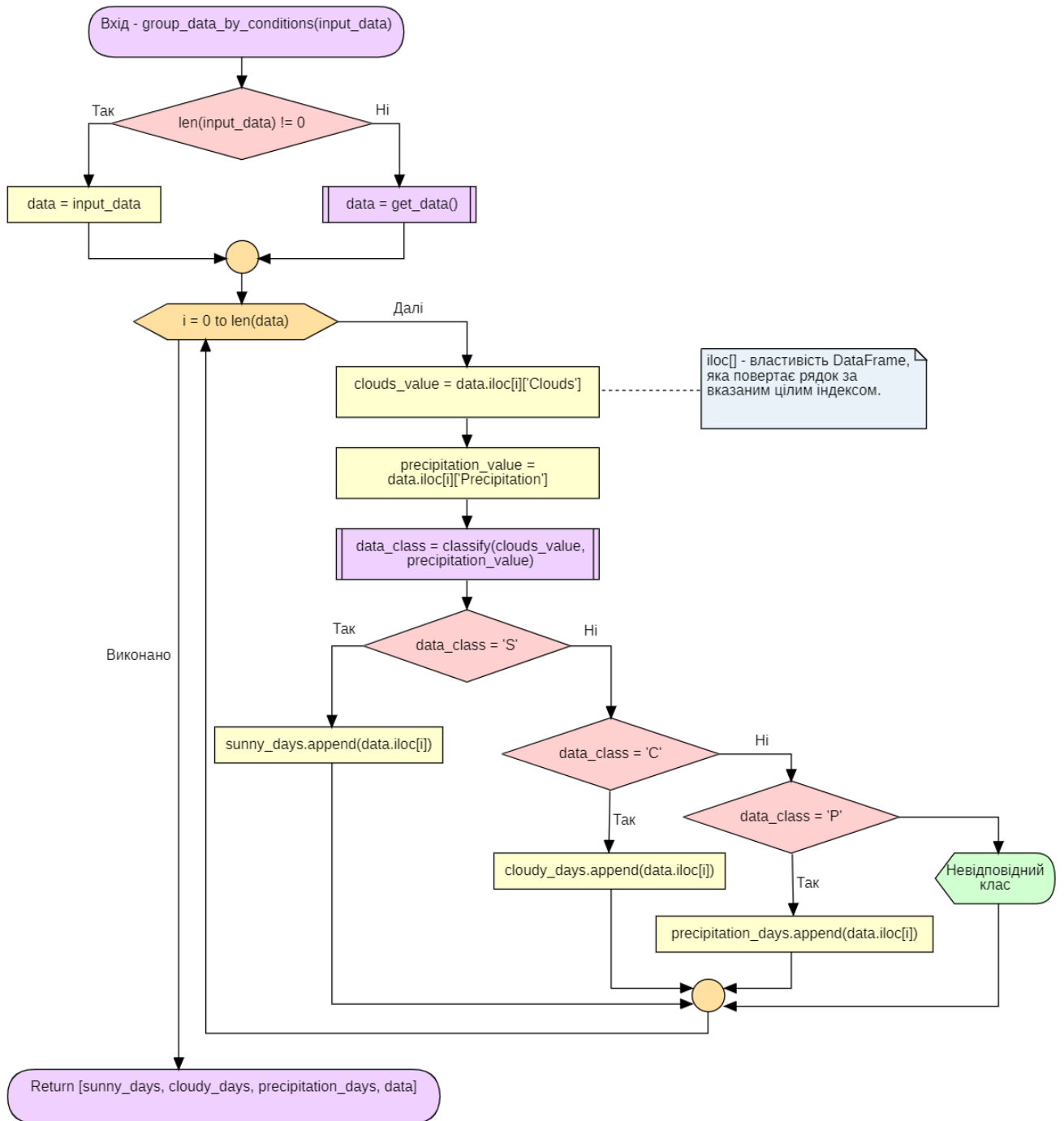


Рисунок Г.1 – Блок-схема алгоритму групування даних по погодних умовах

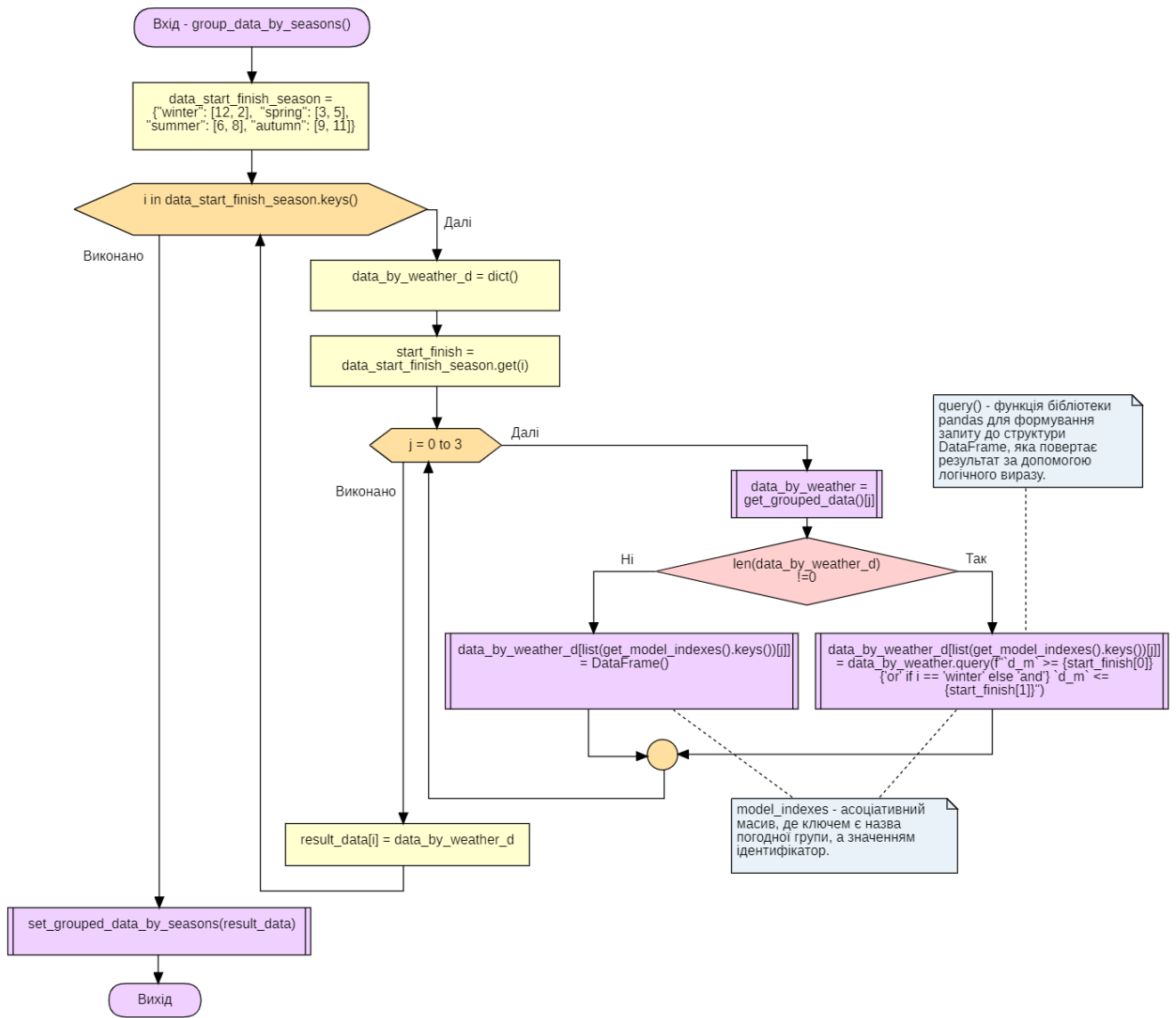


Рисунок Г.2 – Блок-схема алгоритму групування даних по порах року

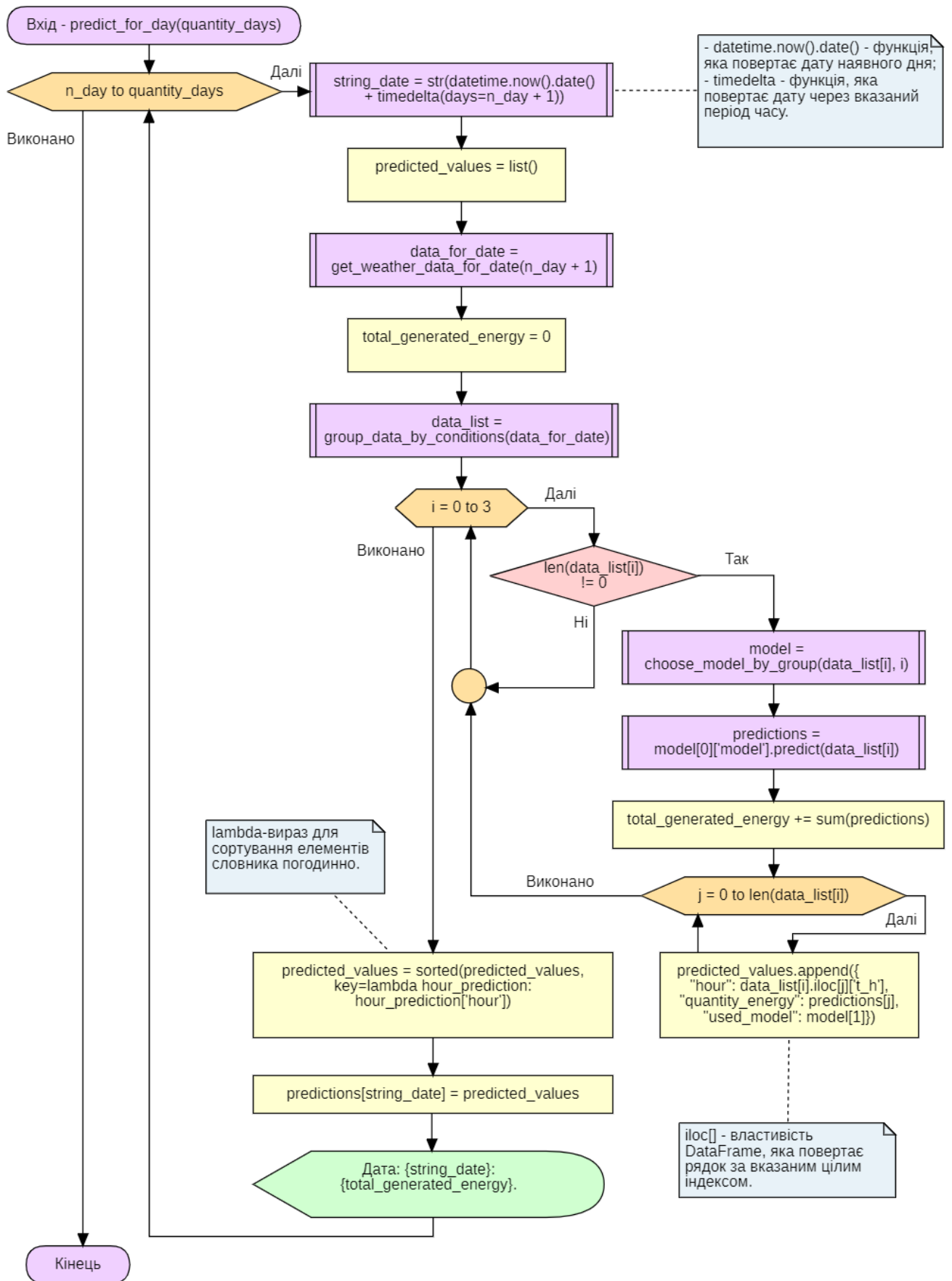


Рисунок Г.3 – Блок-схема алгоритму прогнозування виробництва електроенергії

ДОДАТОК Д

Програмний код для завантаження даних про погоду за допомогою сервісу Open-Meteo та формування вихідної структури

```
def get_weather_data_for_date_with_open_meteo(self, count_days=1):
    timezone = pytz.timezone('GMT')

    lat, lon = self.station_info["lat"], self.station_info["lon"]
    start_date = str(datetime.now().date() + timedelta(days=1))
    end_date = str(datetime.now().date() +
timedelta(days=count_days))

    response = requests.get(
f"https://api.open-
meteo.com/v1/forecast?latitude={lat}&longitude={lon}"
f"&hourly=temperature_2m,relativehumidity_2m,precipitation,"
f"surface_pressure,cloudcover,windspeed_10m,shortwave_radiation&"
f"windspeed_unit=ms&timeformat=unixtime&start_date="
f"{start_date}&end_date={end_date}").json()

    response_data = pd.json_normalize(response)

    year, month, day, hour = [], [], [], []

    for date_timestamp in response_data['hourly.time'][0]:
        date_f = datetime.fromtimestamp(date_timestamp,
tz=timezone)
        year.append(date_f.year)
        month.append(date_f.month)
        day.append(date_f.day)
        hour.append(date_f.hour)

    df = pd.DataFrame(
        {
            'd_y': year,
            'd_m': month,
            'd_d': day,
            't_h': hour,
            'Bar': [i * 0.750061683 for i in
response_data['hourly.surface_pressure'][0]],
            'Rain': response_data['hourly.precipitation'][0],
            'SolarRad':
response_data['hourly.shortwave_radiation'][0],
            'TempOut': response_data['hourly.temperature_2m'][0],
            'OutHum':
response_data['hourly.relativehumidity_2m'][0],
            'WindSpeed': response_data['hourly.windspeed_10m'][0],
            'Clouds': response_data['hourly.cloudcover'][0],
        })

    return df
```

ДОДАТОК Е

Дані для обґрунтування економічної ефективності

Таблиця Е.1 Кількість виробленої електроенергії та прогнозованої її кількості

Дата	Реальне значення, кВт·год	Прогнозоване значення (Розроблений метод), кВт·год	Прогнозоване значення (Лінійна регресія), кВт·год
01.05.2020	198,28	183,7909	182,8507
02.05.2020	169,04	160,2194	163,0807
03.05.2020	154,61	147,6811	147,7133
04.05.2020	185,46	190,648	193,9739
05.05.2020	147,39	156,2357	131,6653
06.05.2020	186,31	191,7163	195,4602
07.05.2020	186,85	194,9716	194,4258
08.05.2020	151,54	155,1938	146,3227
09.05.2020	185,48	192,8351	194,8265
10.05.2020	160,97	167,3538	153,869
11.05.2020	198,35	189,6641	213,3531
12.05.2020	156,59	149,3678	161,8345
13.05.2020	167,90	159,4845	172,5675
14.05.2020	184,03	196,1762	169,5001
15.05.2020	180,18	189,5822	173,7356
16.05.2020	184,84	177,2142	175,4831
17.05.2020	197,53	193,0239	188,8981
18.05.2020	183,22	173,7309	199,2963
19.05.2020	151,64	155,4603	146,8244
20.05.2020	157,62	155,549	150,9381
21.05.2020	198,81	206,2293	195,2206
22.05.2020	156,15	149,3523	163,5919
23.05.2020	180,15	185,3633	171,7442
24.05.2020	150,15	157,7212	155,6539
25.05.2020	177,90	180,5682	175,7315
26.05.2020	157,45	163,9613	150,5759
27.05.2020	196,44	189,8507	205,8555
28.05.2020	170,81	158,8533	158,3124
29.05.2020	160,83	163,6547	153,8399
30.05.2020	146,53	143,2146	152,0266
31.05.2020	165,75	170,1027	187,0343
Загалом	5348,79	5348,7705	5326,2057

Таблиця Е.2 Результати обчислення різниці та відносної похибки між реальним та прогнозованим значенням

Дата	Розроблений метод		Лінійна регресія	
	Різниця, кВт·год	Похибка, %	Різниця, кВт·год	Похибка, %
01.05.2020	14,49	6,81	15,43	7,22
02.05.2020	8,82	4,96	5,96	3,41
03.05.2020	6,93	4,29	6,90	4,27
04.05.2020	5,19	2,72	8,52	4,39
05.05.2020	8,85	5,66	15,72	9,64
06.05.2020	5,41	2,82	9,15	4,68
07.05.2020	8,12	4,17	7,58	3,90
08.05.2020	3,66	2,36	5,22	3,33
09.05.2020	7,35	3,81	9,34	4,80
10.05.2020	6,39	3,82	7,10	4,22
11.05.2020	8,68	4,19	15,01	7,03
12.05.2020	7,23	4,41	5,24	3,24
13.05.2020	8,41	4,77	4,67	2,71
14.05.2020	12,14	6,19	14,53	7,32
15.05.2020	9,40	4,96	6,45	3,45
16.05.2020	7,62	3,96	9,35	4,82
17.05.2020	4,50	2,23	8,63	4,19
18.05.2020	9,48	4,92	16,08	8,07
19.05.2020	3,82	2,46	4,82	3,08
20.05.2020	2,07	1,30	6,68	4,07
21.05.2020	7,42	3,60	3,59	1,77
22.05.2020	6,80	4,17	7,44	4,55
23.05.2020	5,22	2,81	8,40	4,46
24.05.2020	7,57	4,80	5,50	3,53
25.05.2020	2,67	1,48	2,17	1,20
26.05.2020	6,52	3,97	6,87	4,18
27.05.2020	6,58	3,24	9,42	4,58
28.05.2020	11,95	6,54	12,49	6,82
29.05.2020	2,82	1,72	6,99	4,17
30.05.2020	3,31	2,21	5,50	3,62
31.05.2020	4,35	2,56	21,28	11,38