НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

«На правах рукопису»
УДК 004.942.3

«До захисту допущено»
Завідувач кафедри ММСА
Тимощук О.Л.
«___»_____20__ р.

Магістерська дисертація
на здобуття ступеня магістра
зі спеціальності 124 Системний аналіз
на тему: «Кластеризація і прогнозування стану країн ООН за показниками сталого розвитку»

Виконав:
студент ІІ курсу, групи КА-02мп
Самсонюк Максим Вікторович _____

Керівник:
професор кафедри ММСА, д.т.н., проф.,
Бідюк П.І. _____

Рецензент:
декан ФІОТ КПІ ім. Ігоря Сікорського
д.т.н., проф.,
Теленик С.Ф. _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент (-ка) _____

Київ
2021

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти – другий (магістерський)

Спеціальність – 124 «Системний аналіз»

ЗАТВЕРДЖУЮ
Завідувач кафедри ММСА
_____ О.Л.Тимощук
(підпис)
«___»_____2021 р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Самсонюк Максим Вікторович

1. **Тема дисертації:** «Кластеризація і прогнозування стану країн ООН за показниками сталого розвитку», науковий керівник дисертації Бідюк Петро Іванович, доктор технічних наук, професор, затверджені наказом по університету від «02» листопада 2021 р. № 3651-с.
2. **Термін подання студентом дисертації** 12.12.2021
3. **Об'єкт дослідження**: показники сталого розвитку країн ООН.
4. **Предмет дослідження**: методи машинного навчання, статистичні критерії адекватності, критерії якості кластеризації, нестаціонарні часові ряди.
5. **Перелік завдань, які потрібно зробити:**
   1) Дослідити актуальність обраної теми та існуючих підходів до аналізу;
   2) Здійснити огляд технічної літератури за темою дослідження;
   3) Ознайомитись із різними існуючими методами кластеризації та прогнозування нестаціонарних часових рядів;
   4) Здійснити порівняльний аналіз існуючих методів, виявити їх переваги та недоліки;
   5) Сформувати базу даних для виконання кластеризації та список показників сталого розвитку, які будуть прогнозуватись;
   6) Обрати методи для кластеризації та побудови прогнозів показників сталого розвитку;

7) Розробити та протестувати систему, яка буде виконувати кластеризацію вхідних даних, будувати прогнози показників сталого розвитку;
8) Провести аналіз результатів;
9) Провести аналіз ринкових можливостей запуску стартап проекту;
10) Розробити концептуальні висновки;
11) Підготувати ілюстративний матеріал;
12) Оформити пояснювальну записку.

6. Орієнтовний перелік ілюстративного матеріалу:

1. Методи та підходи до вирішення задач кластеризації і прогнозування
2. Функціональна схема програмного продукту
3. Приклади виконання моделювання
4. Розрахункові таблиці з результатами
5. Таблиці по розробці стартап проекту

7. Орієнтовний перелік публікацій:

1) Самсонюк М.В, Бідюк П.І.. Задача кластеризації методами інтелектуального аналізу даних на прикладі статистики про сталий розвиток країн ООН, *VIII Міжнародна науково-технічна Internet-конференція* «Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами», 25-26 листопада 2021 року Київ.
2) Самсонюк М.В. Задача кластеризації методами інтелектуального аналізу даних на прикладі статистики про сталий розвиток країн ООН. Університетський науковий збірник «Системні науки та кібернетика» – К: NTUU «KPI», 2021. С. 136 - 155 (прийнято до друку).

8. Дата видачі завдання: 1 вересня 2021 року.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Отримання завдання на магістерську дисертацію	01.09.2021 – 05.09.2021	Виконано
2.	Огляд технічної літератури за темою	06.09.2021 – 12.09.2021	Виконано
3.	Вивчення матеріалів за темою дисертації, формулювання вступу, об’єкту, предмета, цілі, завдань, новизни, практичної значущості результатів.	13.09.2021 – 19.09.2021	Виконано
4.	Перший розділ. Дослідження актуальності аналізу показників сталого розвитку країн ООН. Огляд існуючих методів кластеризації та прогнозування	20.09.2021 – 26.09.2021	Виконано
5.	Другий розділ. Математична постановка задачі, обрання методів інтелектуального аналізу для подальшої роботи, обрання критеріїв порівняння результатів	27.09.2021 – 03.10.2021	Виконано
6.	Третій розділ. Розробка системи підтримки прийняття рішень	04.10.2021 – 10.10.2021	Виконано
7.	Розширення функціоналу системи	11.10.2021 – 17.10.2021	Виконано
8.	Аналіз результатів	18.10.2021 – 24.10.2021	Виконано
9.	Проведення аналізу ринкових можливостей стартап – проекту	25.10.2021 – 31.10.2021	Виконано
10.	Підготовка ілюстративного матеріалу	01.11.2021 – 14.11.2021	Виконано
11.	Оформлення пояснювальної записки	15.11.2021 – 26.11.2021	Виконано

Студент _____ М.В. Самсонюк

Науковий керівник дисертації _____ П.І. Бідюк

РЕФЕРАТ

Магістерська дисертація: 158 с., 31 рис., 54 табл., 1 додаток, 23 джерела.

МАШИННЕ НАВЧАННЯ, СТАТИСТИЧНИЙ АНАЛІЗ, ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ, КЛАСТЕРИЗАЦІЯ, СТАЛИЙ РОЗВИТОК, ПРОГНОЗУВАННЯ

Дана робота присвячена дослідженню методів інтелектуального аналізу даних у задачах кластеризації, а також методів прогнозування часових рядів у сфері сталого розвитку.

Об’єктом дослідження є показники сталого розвитку країн ООН.

Предметом дослідження являються методи машинного навчання, статистичні критерії адекватності, критерії якості кластеризації, часові ряди.

Метою дослідження є побудова системи, яка дозволятиме виконувати якісну кластеризацію даних та короткострокові прогнози показників сталого розвитку.

Актуальність роботи полягає в необхідності розробки такої системи, яка би дозволила виділити кластери тих країн, які відстають від плану сталого розвитку світу та виокремити ті цілі сталого розвитку, які найгірше виконуються кожною країною ООН.

Результатом роботи являється система підтримки прийняття рішень, яка виконує кластеризацію за вхідними даними – показниками сталого розвитку для країн ООН, а також виконує прогнозування показників на короткостроковий період.

Новизною роботи являється розробка нової оригінальної системи підтримки прийняття рішень, яка надає ряд переваг стосовно обробки даних, зокрема, можливість подубови ряду моделей інтелектуального аналізу даних, їх адаптації до вхідних даних, комбінування отриманих результатів кластеризації.

ABSTRACT

The topic: Clustering and state forecasting of the UN countries using sustainable development indexes

Master's thesis: 158 p., 31 figures, 54 tables, 1 supplement, 23 sources

MACHINE LEARNING, STATISTICAL ANALYSIS, INTELLECTUAL DATA ANALYSIS, CLUSTERIZATION, SUSTAINABLE DEVELOPMENT, FORECASTING

This work is devoted to the study of methods of data mining in clustering problems, as well as methods of forecasting time series in the field of sustainable development.

The object of the study is the indicators of sustainable development of the UN countries.

The subject of the study are the methods of machine learning, statistical criteria of adequacy, quality criteria of clustering, time series.

The aim of the study is to build a system that will perform high-quality data clustering and short-term forecasts of sustainable development indicators.

The relevance of the work lies in the need to develop a system that would identify clusters of countries that lag behind the plan of sustainable development of the world and identify those sustainable development goals that are worst met by each UN country.

The result of the work is a decision support system that performs clustering of input data - indicators of sustainable development for UN countries, as well as performs forecasting of indicators for the short term.

The novelty of the work is the development of a new original decision support system, which provides a number of advantages regarding data processing, in particular, the ability to build a number of models of data mining, their adaptation to input data, combining the results of clustering.

ЗМІСТ

ВСТУП

Задачі кластеризації та прогнозування на сьогодні є надзвичайно важливими у сфері інтелектуального аналізу даних. Математичні методи для моделювання процесів, кластеризації їх за різними ознаками застосовуються практично в усіх галузях. Наприклад, у медицині під час аналізу отриманих зображень (кластеризація стану пухлини за стадією, прогнозування швидкості росту тощо), у психіатрії (кластерний аналіз симптомів хворого, таких як шизофренія, паранойя і так далі для подальшого правильного лікування). Також часто застосовується кластерний аналіз у маркетинговій сфері (для кластеризації покупців на певні категорії, за купівельною спроможністю). Загалом, коли нам необхідно класифікувати великий масив даних до придатних для подальшої обробки груп, кластерний аналіз є корисним та ефективним. На сьогоднішній день існує велика кількість різних методів кластеризації, що дозволяє виконати обширний аналіз, застосувавши різні моделі, порівнявши їх між собою за допомогою спеціальних критеріїв і методів та обрати найкращу модель для подальшої роботи.

Для виконання прогнозування існує також багато підходів. Одним з таких підходів є регресійний. У ньому для аналізу того чи іншого процесу використовується теорія аналізу часових рядів. Загалом, часовим рядом називається набір даних, що зафіксовані в хронологічному порядку в конкретні моменти часу. Існує велика кількість моделей часового ряду, такі як авторегресія, авторегресія з ковзним середнім, авторегресія з інтегрованим ковзним середнім тощо. Вибір тієї чи іншої моделі залежить від вхідного часового ряду. Спочатку відбувається попередній його аналіз, який включає в себе тестування на нестаціонарність (змінне математичне сподівання), нелінійність, гетероскедастичність (змінна дисперсія) за допомогою спеціальних тестів. Потім, вивчивши природу та структуру вхідного датасету,

можна переходити до підготовки його до побудови моделі, що може включати в себе фільтрування, заповнення пропусків тощо. Після побудови достатньої кількості моделей, вивчається їх адекватність за допомогою спеціальних критеріїв та будуються прогнози значень на майбутнє. Прогнозування оцінюється також за допомогою критеріїв оцінки якості прогнозів. Дещо іншим підходом є інтелектуальний аналіз даних. В такому випадку математичні моделі є зовсім різними за своєю структурою, це можуть бути випадкові дерева, ліси, метод SVM (support vector machine), метод К-середніх, нейронні мережі і так далі.

У своїй магістерській дисертації в якості досліджуваного процесу я обрав сталий розвиток країн ООН. Це дуже актуальна тема на сьогодні, оскільки проблема забезпечення необхідними ресурсами людей нашої планети при збереженні стану навколишнього середовища є на часі, як ніколи. У якості вхідних даних виступатимуть показники сталого розвитку різних країн ООН. Основною задачею є побудова моделі для кластеризації країн за рівнем досягнення задач сталого розвитку, прогнозування показників на майбутнє для визначення перспектив тієї чи іншої країни у цій галузі, визначення у яких саме цілях сталого розвитку є прогалини та на що саме необхідно звернути увагу при керуванні та розподіленні фінансових активів.

У першому розділі магістерської дисертації розглянуті особливості розвитку сталого розвитку в Україні та світі, описані цілі сталого розвитку та актуальність проблеми. Також, описані основні підходи до кластеризації даних різними методами інтелектуального аналізу. Розглянуті різні методи для мтематичного моделювання процесів, їх основні недоліки та переваги.

У другому розділі дисертації розглянуті тести для дослідження вхідного процесу на стаціонарність, нелінійність та гетероскедастичність, описані обрані методи кластеризації і прогнозування. Крім того, описані критерії, за допомогою яких оцінюватиметься якість виконаної кластеризації, критерії адекватності моделей та критерії якості прогнозів.

РОЗДІЛ 1 АКТУАЛЬНІСТЬ ТЕМИ І ІСНУЮЧІ ПІДХОДИ ДО РОЗВ'ЯЗАННЯ ЗАДАЧ КЛАСТЕРИЗАЦІЇ І ПРОГНОЗУВАННЯ ПРОЦЕСІВ СТАЛОГО РОЗВИТКУ

1.1 Характеристика процесів сталого розвитку в Україні та світі

На сьогоднішній день у зв'язку з ростом населення нашої планети і зростання масштабів економічної системи постає складне питання споживання природних ресурсів. Зменшення їх запасів призводить до серйозних екологічних проблем, руйнувань, катастроф тощо. Поступово до світового співтовариства прийшло усвідомлення в необхідності реалізації кроків, напрямлених на зменшення негативних ефектів життєдіяльності людей. В результаті, в 1992 році на конференції ООН був прийнятий документ, який напрямлений на перехід людства до сталого розвитку, тобто, до економічного росту з одночасним вирішенням проблем негативного впливу даного процесу на навколишнє середовище [1].

Отже, сталим розвитком країн можна вважати такий процес змін у країні, який націлений на задоволення теперішніх потреб населення при збереженні навколишнього середовища і ресурсів, тобто без збитків для можливості наступних поколінь задовольняти свої власні потреби.

Основні цілі сталого розвитку були оголошені в 2015 році на Самміті ООН у Нью-Йорку та були задокументовані у «Повістці дня в галузі сталого розвитку на період до 2030 року» [2]. Усього програма включає в себе 17 цілей, серед яких:

1. Ліквідація бідності. Бідність – нехватка ресурсів для існування на стійкій основі, голод й недоїдання, обмежений доступ до соціальних послуг, дискримінація й ізоляція. Основна ціль – ліквідувати до 2030 року

крайню бідність для всіх людей по всьому світу. Зараз крайня бідність оцінюється як проживання на суму менш ніж 1,25$ на день.

2. Ліквідація голоду. Наразі голод є великою перешкодою на шляху до стійкого розвитку, так як він є причиною зниження працеспроможності, появи хвороб і т.д.

3. Здоров'я і благополуччя. Не дивлячись на значні досягнення у сфері медицини, нерівність у сфері доступа до медичних послуг й досі зберігається.

4. Якісна освіта. Освіта сприяє скороченню нерівності й досягненню гендерної рівності. На основі даних по 114 країнам з 1985 по 2005 роки було виявлено, що один додатковий рік навчання відповідає скороченню коефіцієнта Джині на 1.4%

5. Гендерна рівність. Основна ціль – надання жінкам і дівчатам рівного доступу до освіти, медико-санітарного обслуговування, роботі і участю в політичних та економічних процесах.

6. Чиста вода і санітарія. Раціональне використання водних ресурсів сприятиме покращенню управління виробництвом продуктів харчування і енергії й сприятиме в забезпеченні економічного росту.

7. Недороговартісна і чиста енергія. При спалюванні вуглеводневого палива відбувається викид в атмосферу парникових газів, які викликають зміну клімату й негативно впливають на навколишнє середовище. При відсутності стабільної подачі електроенергії країни не зможуть підживлювати свою економіку. Підприємства можуть підтримувати й зберігати екосистему шляхом розвитку гідроенергетики та біоенергетики.

8. Гідна робота та економічне зростання. Продуктивна зайнятість та гідна робота є надважливими елементами для досягнення справедливої глобалізації та скорочення бідності. Для вирішення цієї проблеми бізнесу необхідно створювати рівні можливості, завдяки яким люди зможуть

знайти роботу не зважаючи на стать, рівень доходу чи соціально-економічний статус.

9. Індустріалізація, інновації та інфраструктура. Зростання нових видів промисловості сприятиме підвищенню рівня життя.

10. Зменшення нерівності. Найбідніші 40% населення планети заробляють 25% від світового доходу. Необхідно підтримувати соціально вразливих та малозабезпечених

11. Сталі міста та населені пункти. Основна ціль – до 2030 року забезпечити загальний доступ до недорогого та доступного житла, зменшити забруднення навколишнього середовища.

12. Відповідальне споживання й виробництво. Ціль – зниження рівня відходів, перехід підприємців до нових підходів, моделей сталого споживання (наприклад, переробляти пластик, не викидати їжу тощо)

13. Боротьба зі зміною клімату. Ціль – впровадження у виробництві інновацій й здійснення довгострокових вкладень в забезпечення енергоефективносты та низько вуглеводного розвитку.

14. Збереження морських екосистем. Викиди сміття в океани негативно впливає на біорізноманітність. Необхідно зменшити використання пластмасових продуктів до мінімально можливого рівня й забезпечити прибирання берегових зон.

15. Збереження екосистем на суші. Ціль – раціональне лісо використання, боротьба з опустошенням і зупинка процесу втрати біорізноманіття.

16. Мир, правосуддя й ефективні інститути. Ціль – сприяння побудові миролюбного та відкритого суспільства в інтересах сталого розвитку.

17. Партнерство в інтересах сталого розвитку. Для досягнення цілей в області сталого розвитку, уряди держав, громадянське суспільство, науковці й бізнес мають діяти разом [2].

Насправді, на сьогоднішній день жодна країна не досягла усіх 17 цілей. Для оцінки досягнень країн був створений спеціальний індекс SDG (Sustainable Development Goals Index). Спеціалісти компаній SDSN і Bertelsmann Stiftung щорічно рейтинги країн на основі цього індексу. Наразі лідерами по досягненню цілей сталого розвитку є розвинені країни (Фінляндія, Швеція, Данія, Німеччина, …) [3]. Це зумовлено насамперед великими фінансовими можливостями по перерозподілу доходів та вирішенням низки соціальних проблем. По-друге, постіндустріальна структура економіки розвинених країн передбачає низький рівень матеріаломісткості виробництва і високу частку сфери послуг, особливо інформаційного характеру, що автоматично знижує навантаження на навколишнє середовище. Високий рівень життя призводить до зростання попиту з боку населення країни на екологічні та соціальні програми і проекти. Що стосується країн, що розвиваються, то відносно низький рівень вимог дає їх виробникам певну перевагу перед конкурентами з розвинених країн, дозволяючи виробляти продукцію з меншими витратами. Крім того, населення в силу більш низьких доходів менше зацікавлене в додаткових витратах суспільства на захист навколишнього середовища і соціальні гарантії працівникам. Попит населення і бізнесу подібних країн на високий рівень стандартів сталого розвитку також обмежується переважанням матеріаломісткого і трудомісткого виробництва в структурі економіки. Підвищення екологічних і соціальних вимог в даних галузях може викликати суттєве зниження їх рентабельності, а також зростання безробіття. Завдяки цьому, а також меншим фінансовим ресурсам уряди таких країн нездатні прийняти на себе зобов'язання слідувати нормам сталого розвитку в тій же мірі, що і розвинені країни [4].

Що стосується України, то у 2021 році вона в рейтингу піднялась на 36 місце серед 165 країн в щорічному рейтингу сталого розвитку [5]. Прогрес України був відмічений у досягненні 10 з 17 цілей, найбільший прогрес Україна показала в досягнення цілі з подолання бідності.

Проте загальна картина по всім країнам останнього року по всім країнам не досить втішна, оскільки пандемія COVID-19 сприяла зростанню безробіття та не стала запорукою сталого розвитку і економічного підйому. Тому глобальний показник індексу виявився нижчим за минулий рік.

Отже, дійсно, тема сталого розвитку на сьогодні є дуже актуальною як в Україні, так і в усьому світі. У випадку, якщо такі проблеми, як військові конфлікти, величезний розрив між бідними і багатими, руйнування навколишнього середовища не будуть вирішені, то усій природній та соціальній реальності загрожує повна загибель. Саме тому важливо кластеризувати країни по основним показникам сталого розвитку, аби зрозуміти, в яких державах є реальні проблеми та у яких саме галузях є прогалини. Важливо розуміти, які цілі поступово досягаються тією чи іншою державою, а які потребують значних допрацювань або повного розвороту у керуванні.

Також важливо прогнозувати показники сталого розвитку. Це дасть можливість оцінити перспективи кожної держави у досягненні цілей, показати у який кластер рухається держава, що очікувати у майбутньому при збереженні стратегії керування фінансовими ресурсами. Математичне моделювання може значно допомогти у вирішенні соціальних, економічних та екологічних проблемах сталого розвитку. Моделювання та прогнозування спроможне виявити ключові фактори розвитку, оцінити ефективність прийнятих управлінських рішень.

1.2 Методи інтелектуального аналізу даних для задач прогнозування

Математичні моделі на основі інтелектуального аналізу даних зараз завойовують все більшу популярність. Для задач прогнозування застосовують велику кількість методів, серед яких можна виділити такі як: випадкове дерево, випадковий ліс, лінійна та множинна регресії, метод опорних векторів (support vector machine), метод k-найближчих сусідів тощо. Чимало з цих методів можна застосувати не тільки до задачі прогнозування, а й до задач класифікації та кластеризації. Розглянемо найбільш популярні з них.

1.2.1 Метод k-nearest

K-алгоритм найближчих сусідів - метричний алгоритм для автоматичної класифікації об'єктів або регресії по найближчих для нього об'єктам, значення яких вже відомі. Алгоритм може бути застосований до вибору з великою кількістю атрибутів (багатовимірних).

Алгоритм KNN для задачі класифікації наступний:

Спочатку заданим числом k ми визначили те, скольки точок буде мати право голосу при визначенні класу. В результаті ми виявили ті точки, відстань (евклідова) від яких до нової являється мінімальною. Тепер можна приступити до простого невзваженого голосування. Відстань від кожної точки при голосуванні тут більше не грає ролі. Кожна точка, яка має право голосу, голосує за клас, до якого сама належить. Новій точці присвоюється значення середнього арифметичного тих точок, які мають право голосу.

У випадку однакової кількості голосів виконується взважене голосування. У такій ситуації враховується також і відстань до нової точки. Чем менша відстань, тим більш значний вклад вносить голос. Голоса для класу знаходиться за наступною формулою:

$$votes(class) = \sum_{t=1}^{n} \frac{1}{d^2(X,Y_i)}, \quad (1.1)$$

де $d^2(X, Y_i)$ - квадрат відстані від відомої точки Y_i до нової - X, n - кількість відомих точок класу, до якого розраховуються голоси.

Переваги: легкий для розуміння, швидкий та ефективний.

Недоліки: Необхідно обрати оптимальну кількість сусідніх точок k, що беруть участь у голосуванні [5].

1.2.2 Метод опорних векторів (support vector machine)

SVM (Support Vector Machine, машина опорних векторів) - це особливий клас алгоритмів, який характеризується використанням ядер, відсутністю локальних мінімумів, і використовується для вирішення задач класифікації і регресії. Для задачі регресії алгоритм Support Vector Machine наступний:

Нехай нам дано навчальну вибірку $X = (x_i, y_i)_{i=1}^{l}$. Вирішення задачі регресії шукаємо в лінійному випадку:

$$f(x) = (w, x) - w_0 \quad (1.2)$$

Для кожного такого вектору і вводимо функцію втрат:

$$a(x_i) = \left|\left(w, f(x_i)\right) - w_0 - y_i\right|_\varepsilon, \tag{1.3}$$

де $|z|_\varepsilon = \max(0, |z| - \varepsilon)$

Необхідно знайти таку функцію f_0, яка найкращим чином описує залежність $E(y|x) = f_0(x)$. Для побудови SVR вирішується пряме завдання мінімізації функціонала втрат, в припущенні, що рішення задається лінійною комбінацією деяких породжуючих функцій, з яких можемо скласти вектор-функцію: $f(x) = \left\|\begin{matrix} f_1(x) \\ f_2(x) \\ \dots \\ f_k(x) \end{matrix}\right\|$

Тоді функціонал приймає вигляд:

$$Q_\varepsilon(a, X) = \sum_{i=1}^{l}\left|\left(w, f(x_i)\right) - w_0 - y_i\right|_\varepsilon + r(w, w)^2 \to \min_{w, w_0} \quad , \tag{1.4}$$

де останній доданок запобігає коефіцієнтам w від нескінченного зростання.

В даному випадку задача мінімізації еквівалентна задачі квадратичного програмування з обмеженнями у вигляді нерівностей. Нехай $C = \frac{1}{2r}$

Введемо додаткові змінні ξ_i^+ та ξ_i^-:

$$\xi_i^+ = (a(x_i) - y_i - \varepsilon)_+, \ \xi_i^- = (-a(x_i) + y_i - \varepsilon)_-, i = 1, .., l \tag{1.5}$$

Геометричний сенс ξ_i^+ та ξ_i^- можемо бачити на рис. 1.1 [6]:

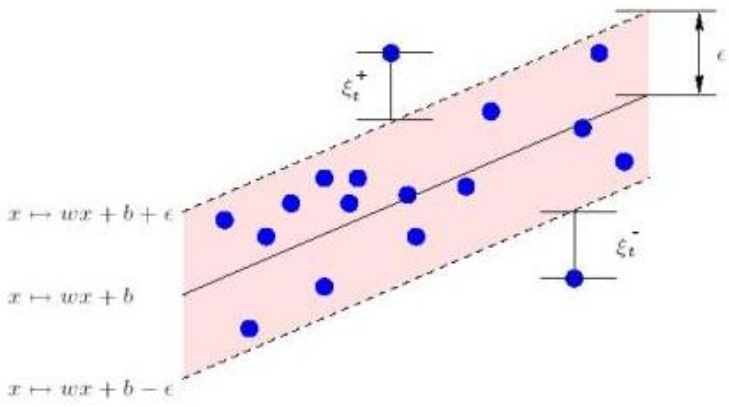

Рисунок 1.1 – Геометричний зміст (модель SVR)

Тепер можемо записати задачу мінімізації у вигляді задачі квадратичного програмування:

$$\begin{cases} \frac{1}{2}(w,w)^2 + C\sum_{i=1}^{l}(\xi_i^+ + \xi_i^-) \to min_{w,w_0,\xi_i^+,\xi_i^-} \\ (w,x_i) - w_0 \le y_i + \varepsilon + \xi_i^+ \\ (w,x_i) - w_0 \ge y_i - \varepsilon - \xi_i^- \\ \xi_i^- \ge 0, \quad i = 1,..,l \\ \xi_i^+ \ge 0, \quad i = 1,\dots,l \end{cases} \tag{1.6}$$

Переваги методу: висока ефективність, не чутлива до перенавчання.

Недоліки: Не найкращий вибір для роботи з великою кількістю факторів.

1.2.3 Метод випадкового лісу

RF (random forest) - це ансамблевий метод машинного навчання, що використовується для задач регресії та класифікації і представляє собою множину дерев рішень. У задачі регресії прогнози кожного дерева рішення

усереднюються, в завданні класифікації приймається рішення голосуванням за більшістю [7]. Алгоритм роботи Random Forest:

1. Вибирається підвибірка навчальної вибірки розміру samplesize - по ній будується дерево (для кожного дерева - своя підвибірка).
2. Для побудови кожного розщеплення в дереві переглядаємо max_features випадкових ознак (для кожного нового розщеплення - свої випадкові ознаки).
3. Вибираємо найкращі ознаки і розщеплення по ньому (по заздалегідь заданому критерію). Дерево будується, як правило, до вичерпання вибірки (поки в листі не залишаться представники тільки одного класу), але в сучасних реалізаціях є параметри, які обмежують висоту дерева, число об'єктів в листі і число об'єктів в підвибірки, при якому проводиться розщеплення [8].
4. На цьому етапі кожне побудоване дерево визначає своє значення прогнозуючої змінної
5. Обираємо результат прогнозу як середнє арифметичне всіх прогнозів побудованих дерев.

Візуалізація алгоритму зображена на рис.1.2.

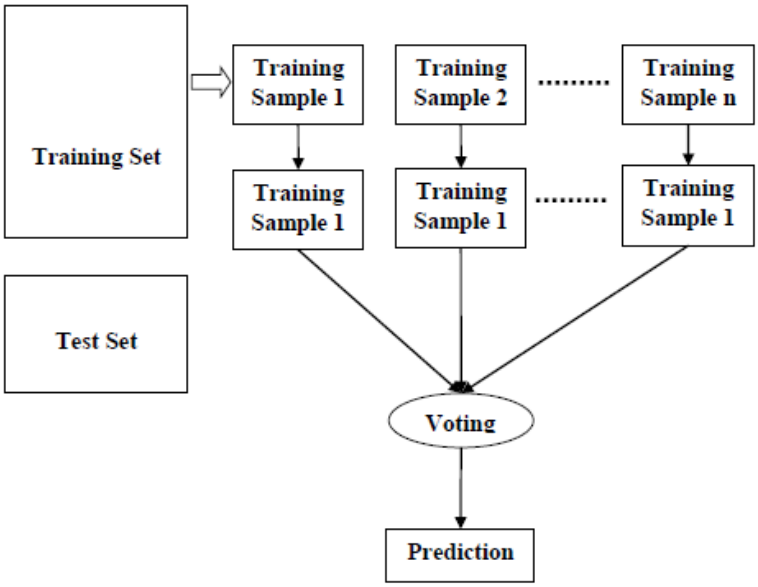

Рисунок 1.2 – Візуалізація алгоритму Random Forest

Переваги: Потужність, точність, добре працює для досить обширного кола проблем, включаючи і лінійні і нелінійні задачі.

Недоліки: Неінтерпретабельний, легко може виникнути перенавчання, необхідно підбирати раціональну кількість дерев рішень.

1.2.4 Нейронні мережі

Нейронні мережі (Neural network, NN) або Штучні нейронні мережі (Artificial neural networks, ANN) – один із видів машинного навчання.Сьогодні нейронні мережі використовують як альтернативу всім існуючим алгоритмам для машинного перекладу, розпізнавання мови та музики, обробки зображень, визначення об'єктів на фото та відео.

Загальна структура нейронної мережі виглядає наступним чином (див. рис. 1.3):

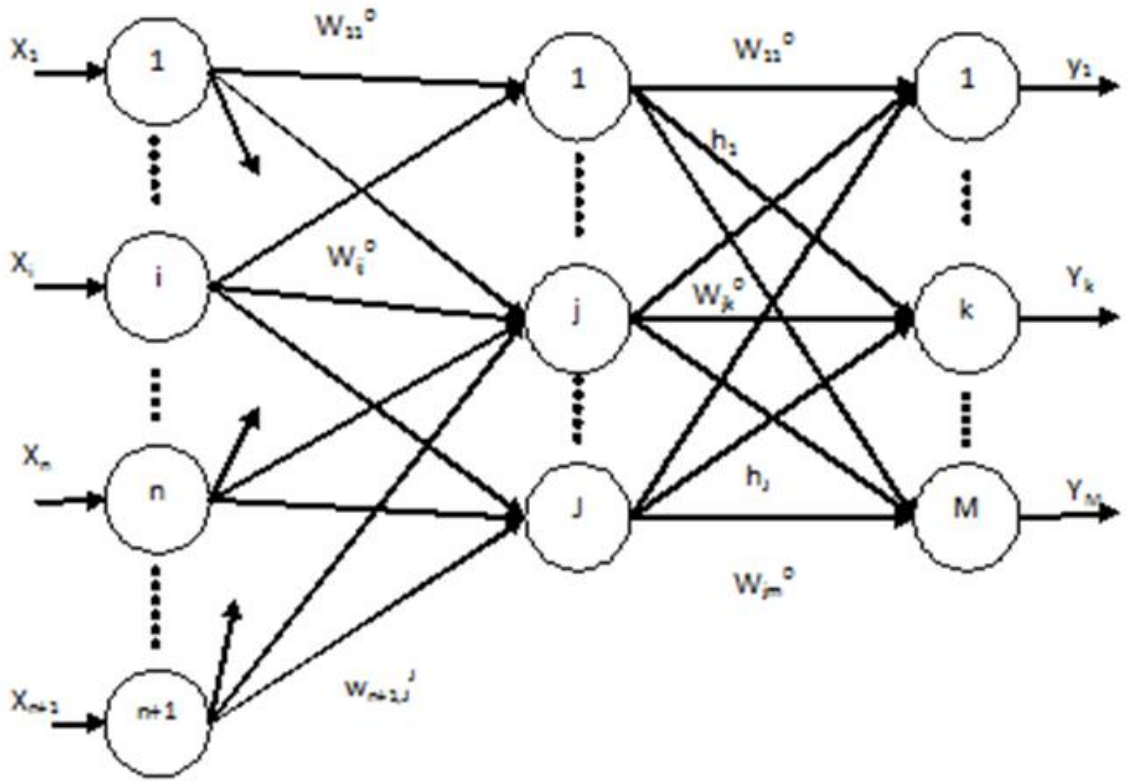

Рисунок 1.3 – Загальна структура нейронної мережі

$H = [h_j]$ – виходи прихованого шару,

$X = [x_i]$ – входи, X_{N+1}= 1,

$Y = [y_k]$ – виходи нейронної мережі

Вагова матриця:

а) вхідна $W_j = \|W_{ij}^I\|$;

б) вихідна $W_O = \|W_{ij}^O\|$.

Нехай

$$S_j = \sum_{i=1}^{N+1} x_i w_{ij}^I\,,\ h_j = f(S_j), \tag{1.7}$$

де f – функція активації.

Для навчання нейронної мережі використовується градієнтний алгоритм [9]:

1. Нехай $W(n)$ - початкове значення вагової матриці

$$W(n+1)=W(n)-\gamma_{n+1}\nabla_w e(W(n)) \ , \tag{1.8}$$

де γ_n - розмір кроку на n-й ітерації.

2. На кожній ітерації спочатку навчаємо вхідні ваги:

$$W_{ij}^{I}(n+1)=W_{ij}^{I}(n)-\gamma_{n+1}\frac{\partial e(W)}{\partial W_{ij}^{I}} \tag{1.9}$$

3. Знаходимо (навчаємо) вихідні ваги:

$$\frac{\partial e(W^0)}{\partial W_{jk}^0}=-2\left(d_k-y_k(W^0)\right)y_k(W^0)\left(1-y_k(W^0)\right)h_j \tag{1.10}$$

$$W_{ij}^0(n+1)=W_{ij}^0(n)-\gamma_{n+1}\frac{\partial e(W)}{\partial W_{ij}^0} \tag{1.11}$$

$$\frac{\partial e(W^I)}{\partial W_{ij}^I}=-\sum_{k=1}^{M}2\left(d_k-y_k(W^I)\right)\cdot y_k(W^0)\cdot\left(1-y_k(W^0)\right)w\cdot h_j(W)\cdot\left(1-h_j(W)\right)\cdot x_i \ , \tag{1.12}$$

де x_i, $i=\overline{1,N+1}$ - входи нейронної мережі, y_k, $k=\overline{1,M}$ - виходи НМ, h_j, $j=\overline{1,J}$ - виходи прихованого шару.

4. $n:=n+1$ - переходимо на наступну ітерацію.

Градієнтний метод являється першим запропонованим алгоритмом навчання, він простий в реалізації, але має наступні недоліки:

- повільно збігається;
- знаходить лише локальні екстремуми.

1.2.5 Метод групового урахування аргументів

Метод групового урахування аргументів – це метод обрання регресійних моделей оптимальної складності. Чим більше параметрів має модель, тим вона вважається складнішою. Для обрання моделі використовуються зовнішні критерії, спеціальні функціонали якості моделі, що обчислюються на тестовій вибірці. Алгоритм роботи МГУА полягає в наступному [10].

Нехай задана вибірка $D = \{(x_n, y_n)\}_{n=1}^{N}, x \in R^m$. Вибірка розбивається на навчальну і тестову. Далі визначається базова модель, наприклад, використовується поліном Колмогорова-Габора:

$$y = w_0 + \sum_{1=1}^{m} w_i x_i + \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} x_i x_j + \sum_{i=1}^{m} \sum_{j=1}^{m} \sum_{k=1}^{m} w_{ijk} x_i x_j x_k + \cdots, \quad (1.13)$$

де w – вектор параметрів (вагових коефіцієнтів).

Виходячи з поставлених задач обирається цільова функція – зовнішній критерій, що описує якість моделі.

На наступному кроці індуктивно породжуються моделі-претенденти. При цьому вводиться обмеження на довжину поліному базової моделі. Далі налаштовуються параметрі моделі. Для цього використовується внутрішній

критерій – критерій, що обчислюється з використанням навчальної вибірки. В якості нього виступає середньоквадратична похибка.

Для вибору моделі обчислюється якість побудованих моделей. При цьому використовується перевірочна вибірка і зовнішній критерій. Модель, яка досягає мінімуму по зовнішньому критерію вважається найоптимальнішою [10].

1.3 Існуючі підходи до кластеризації даних

Кластеризація – це процес розбиття заданої вибірки об’єктів (спостережень) на підмножини (як правило такі, що не перетинаються), що називаються кластерами таким чином, щоб кожен кластер складався зі схожих об’єктів, а об’єкти різних кластерів суттєво відрізнялись [13].

Загалом існує декілька підходів, що дозволяють кластеризувати вибірку даних. Основні з них це:

- методи розбиття;
- ієрархічні методи;
- методи на основі моделей;
- методи на основі щільності;
- методи на основі нейронних мереж.

У цьому пункті будуть описані найбільш популярні серед цих методів

1.3.1 Методи розбиття

Нехай задано n об’єктів, які ми маємо кластеризувати. Ця група методів утворює k множин з даних, кожна з яких представляє собою кластер. В загальному ця група методів працює наступним чином: при заданій кількості кластерів виконується деяке початкове розбиття. Далі об’єкти поступово переміщуються з одного кластеру в інший, намагаючись покращити задану цільову функцію [14].

Щоб знайти глобальний оптимум для цільової функції нам необхідно розглянути всі N(n,k) можливих розбиттів, де:

$$N(n,k) = \frac{1}{k!}\sum_{i=1}^{k}(-1)^{k-i}\binom{k}{i}i^{n} \tag{1.14}$$

Зі збільшенням значення n цей метод стає неефективним, оскільки в такому випадку він знаходить лише локальний екстремум. Типовими представниками цього класу методів є метод k-середніх та нечіткий метод k-середніх.

1.3.2 Ієрархічні методи

Основна ідея цієї групи методів полягає у тому, що вони намагаються у просторі за допомогою метрик відстані розділити об’єкти на групи. Вони бувають двох видів – агломеративні та методи розділення.

Агломеративні методи виконують об’єднання «знизу-догори». Тобто спочатку, кожна точка у просторі представляє собою окремий клас, після чого відбувається об'єднання найбільш схожих кластерів між собою. Така процедура повторюється до того моменту, поки не залишиться k заданих кластерів.

Методи розділення працюють навпаки «зверху-вниз». У таких методах спочатку всі точки представляються одним кластером, після чого цей кластер ділиться на кожному кроці до k необхідних.

Майже всі алгоритми ієрархічної кластеризації є агломеративними, оскільки у методах розділення на кожному етапі алгоритму розділення ми повинні розглянути кожен спосіб поділу даних, тобто вони потребують значних обчислювальних потужностей. Всього існує $2^{n-1} - 1$ способів розбиття при першому кроці алгоритму. Кількість способів розділення є величезною навіть для невеликого набору даних [14].

1.3.3 Методи на основі моделей

Методи на основі моделей описують будь-який метод кластеризації, де модель може бути пристосована до наших даних та математично формалізована. Процес правильного вибору моделі є досить складною задачею, оскільки для цього необхідно ретельно вивчити структуру, властивості та інші особливості вхідних даних.

Типи моделей, які зазвичай використовуються, як правило, базуються на функціях законів розподілу (методи на основі щільності) або на нейронних мережах (методи на основі нейронних мереж) [14].

1.4 Постановка задачі і висновки до розділу

У першому розділі дисертації була надана характеристика процесам сталого розвитку в Україні та світі, висвітлені основні цілі сталого розвитку до 2030 року. Загалом, ця тема дійсно є актуальною сьогодні, оскільки від цього залежить доля не лише нас, а й майбутніх поколінь. Математичне моделювання надає широкий спектр можливостей для аналізу, зокрема, такі задачі як кластеризація та прогнозування є одними з найбільш важливих задач аналізу даних. Вони дозволяють не лише виділити групу «відстаючих» країн по заданим показникам, а й надати важливу інформацію спеціалісту-аналітику, що допоможе стабілізувати та покращити досліджувану систему. Постановкою задачі дослідження є наступне:

1. Пошук даних для країн ООН по показникам сталого розвитку (історичні дані у хронологічному порядку), попередня обробка отриманих даних (фільтрація, нормування, заповнення пропусків тощо).
2. Дослідження та попередній аналіз даних (дослідження на сезонність, стаціонарність, нелінійність, гетероскедастичність тощо).
3. Побудова математичних моделей досліджуваного процесу.
4. Аналіз отриманих моделей-кандидатів за допомогою оцінки адекватності.
5. Побудова прогнозів та їх оцінка за допомогою критеріїв якості прогнозів.
6. Кластеризація країн по досліджуваним даним за допомогою різних методів інтелектуального аналізу.
7. Порівняння якості кластеризації різними методами за допомогою спеціальних критеріїв.

8. Обрання моделей, що виконують найкращу кластеризацію та прогнозування за досліджуваними даними.
9. Аналіз отриманих результатів.

РОЗДІЛ 2. ВИБІР І ОПИС СТРУКТУРИ МАТЕМАТИЧНИХ МОДЕЛЕЙ ДЛЯ ПРОГНОЗУВАННЯ І КЛАСТЕРИЗАЦІЇ

2.1 Регресійні моделі

Одним із найбільш популярних методів для моделювання заданого процесу є регресійний підхід. Він базується на теорії часових рядів. Загалом, часовий ряд – це хронологічно впорядкований набір даних, значення яких спостерігаються через однакові проміжки часу. Основною ціллю аналізу часових рядів є визначення природи часового ряду, його властивостей, а також прогнозування майбутніх значень по поточним та минулим даним. Це потребує певної формалізованості моделі, щоб вона була ідентифікована. В такому випадку за допомогою неї можна інтерпретувати вхідні дані для розуміння сезонності, стаціонарності тощо.

Більшість часових рядів містять елементи, які послідовно залежать один від одного . Таку залежність можна описати наступним рівнянням:

$$x(t) = a_0 + a_1 x(t-1) + a_2 x(t-2) + \cdots + a_n x(t-n) + \varepsilon, \quad (2.1)$$

де $a_i, i = 1, .., n$ – параметри модели, ε – випадкова складова (білий шум).

Тобто, кожне спостереження є сумою випадкової складової та лінійної комбінації попередніх значень. Така модель називається моделлю авторегресії n-го порядку (AP(n)).

Порядок моделі, тобто статистичні значущі попередні значення часового ряду, що потрібно включити до моделі авторегресії можна виявити за допомогою корелограм. Корелограми показують чисельно та графічно автокореляційну функцію (АКФ) для послідовності лагів із певного діапазону.

Лагом називається запізнення часового ряду, тобто, значення $x(t-k)$ є k-тим лагом. Проте, у випадках, коли перший член ряду тісно пов'язаний з другим, а другий – з третім, то періодична залежність може суттєво змінитись після виключення автокореляції першого порядку (тобто після взяття різниці з першим лагом). Тому частіше використовують часткову автокореляційну функцію (ЧАКФ). У ній відсутня залежність між проміжними спостереженнями. Вона ідентична до автокореляційної функції за виключенням того, що при її обчисленні видаляється вплив автокореляцій з меншими лагами [20].

Для виявлення тренду в часовому ряді використовують метод згладжування. Для цього дані локально усереднюють, вводять поняття ковзного середнього. У такому випадку кожен член ряду замінюється простим або взваженим середнім n сусідніх членів ряду (n – ширина вікна). Також замість середнього значення можна використати медіану (це допоможе боротись з викидами даних). Також можуть використовувати експоненційне ковзне середнє. Модель ковзного середнього виглядає наступним чином:

$$x(t) = b_0 + MA(t) + b_1 MA(t-1) + .. + b_m MA(t-m) + \varepsilon, \qquad (2.2)$$

де $b_i, i = 1, .., m$ – параметри моделі, MA – ковзне середнє

Якщо ми хочемо у модель авторегресії додати ефект сезонності, то до нього варто включити й ковзне середнє. В такому випадку отримуємо модель авторегресії з ковзним середнім або АРКС(p,q), де p – порядок авто регресії, q - порядок ковзного середнього. Загалом така модель виглядає наступним чином:

$$x(t) = a_0 + \sum_{i=1}^{p} a_i x(t-i) + \sum_{j=1}^{q} b_j MA(t-j) + \varepsilon \qquad (2.3)$$

Узагальненням моделі авторегресії з ковзним середнім є модель авторегресії з інтегрованим ковзним середнім АРІКС. Ця модель використовується у випадку, якщо вхідний часовий ряд є нестаціонарним (тобто математичне сподівання процесу змінюється з часом). Такі процеси називають інтегрованими (або процеси з трендом).

Процеси з трендом можуть бути описані за допомогою різних нелінійних функцій (поліноми, тригонометричні функції, експоненційна тощо). Приклад з поліноміальним трендом:

$$x(k) = a_0 + a_1 k + a_2 k^2 + \cdots + a_n k^n + \varepsilon(k) \qquad (2.4)$$

Модель АРІКС(p,d,q) (p –порядок авто регресії, d – порядок інтегрованості процесу, q – порядок ковзного середнього) загалом буде описуватись двома рівняннями, перше з яких – рівняння тренду (2.4), а друге – модель АРКС(p,q) з видаленим трендом. Друге рівняння описуватиме коливання, що накладаються на тренд [21].

Тренд видаляють за допомогою методу різниць (перші різниці видаляють тренд першого порядку, другі – другого порядку і так далі). Наприклад, якщо маємо процес з трендом першого порядку $x(k) = a_0 + a_1 k$, то процес видалення тренду виглядатиме наступним чином:

$$dx(k) = x(k) - x(k-1) = a_0 + a_1 k - a_0 - a_1(k-1) = a_1 \qquad (2.5)$$

2.2 Моделі кластеризації даних

Як вже було сказано вище, кластеризація є задачею розбиття набора даних на групи, що називаються кластерами. Ціль – розділити дані таким чином, щоб точки, які знаходяться в одному і тому ж кластері, були дуже схожі один з одним, а точки, що знаходяться в різних кластерах, відрізнялись одна від одної. Як і алгоритми класифікації, алгоритми кластеризації присвоюють (або прогнозують) кожній точці даних номер кластеру, якому вона належить.

Далі будуть описані основні алгоритми кластеризації, що будуть реалізовані у розділі 3 дисертації.

2.2.1 Кластеризація K-середніх

Кластеризація k-середніх – один із найпростіших і найпопулярніших алгоритмів кластеризації. Спочатку вибирається число кластерів k. Після вибору значення k алгоритм k-середніх відбирає точки, які будуть представляти центри кластерів (cluster centers) випадковим чином. Потім для кожної точки даних обчислюється його евклідова відстань до кожного центру кластера. У якості відстані береться середньоквадратична норма l_2, тобто цільовою функцією вважається :

$$S = \sum_{j=1}^{k} \sum \{ \left| x_i - \mu_j \right|^2 | x_i \epsilon c_j \}, \tag{2.6}$$

де x_i – і-тий об’єкт, $c_j -$ j-тий кластер з центром μ_j [13].

Кожна точка відповідає найближчому центру кластера. Алгоритм обчислює центроїди (centroids) - центри мас кластерів. Кожен центроїд - це вектор, елементи якого являють собою середні значення характеристик, обчислені по всіх точках кластера. Центр кластера зміщується в його центр ваги. Точки заново призначаються найближчого центру кластера. Етапи зміни центрів кластерів і перепризначення точок ітеративно повторюються до тих пір, поки кордони кластерів і розташування центроїдів не перестануть змінюватися, тобто, на кожній ітерації в кожен кластер будуть потрапляти одні і ті ж точки даних. Наступний приклад (рис. 2.1) ілюструє роботу алгоритму на синтетичному наборі даних [5].

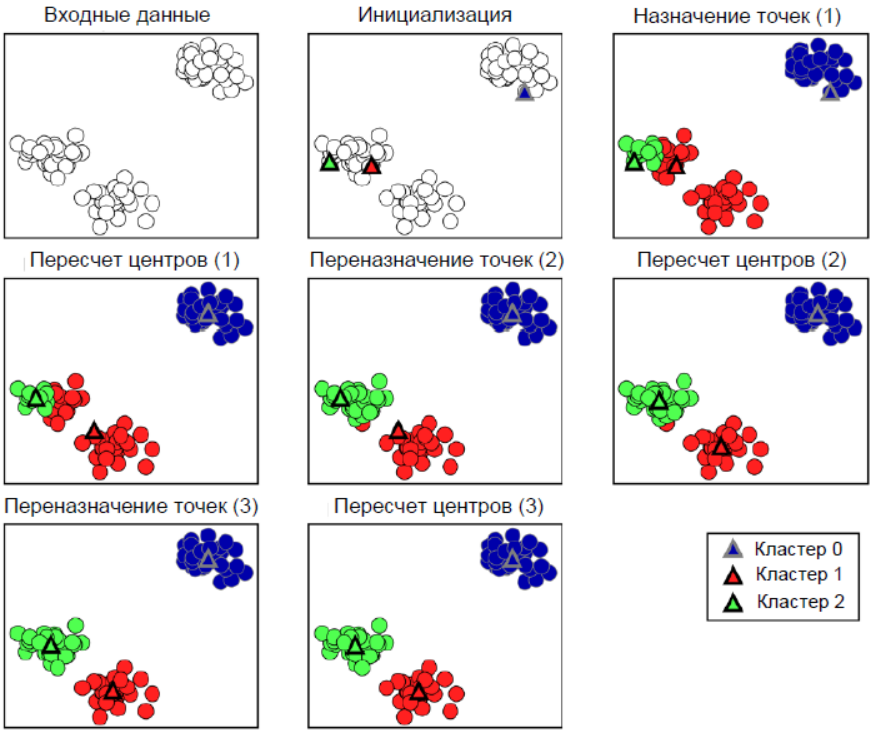

Рисунок 2.1 – приклад роботи алгоритму k-середніх

Центри кластерів представлені у вигляді трикутників, в той час як точки даних відображаються у вигляді кіл. Кольори вказують приналежність до кластеру. Попередньо було вказано, що шукаємо три кластери, тому алгоритм був ініціалізований за допомогою випадкового вибору трьох точок даних в

якості центрів кластерів (див. «Ініціалізація»). Потім запускається ітераційний алгоритм. По-перше, кожна точка даних призначається найближчого центру кластера (див. «Призначення точок (1)»). Потім центри кластерів переносяться в центри мас кластерів (див. «Перерахунок центрів (1)»). Потім процес повторюється ще два рази. Після третьої ітерації приналежність точок кластерним центрам не змінилася, тому алгоритм зупиняється.

Недоліки алгоритму:

Навіть якщо ви знаєте «правильну» кількість кластерів для конкретного набору даних, алгоритм k-середніх не завжди може виділити їх. Кожен кластер визначається виключно його центром, це означає, що кожен кластер має опуклу форму. В результаті цього алгоритм k-середніх може описати відносно прості форми. Крім того, алгоритм k-середніх передбачає, що всі кластери в певному сенсі мають однаковий «діаметр», він завжди проводить межу між кластерами так, щоб вона проходила точно посередині між центрами кластерів. Це іноді може призвести до несподіваних результатів. На рис. 2.2 можемо бачити, що алгоритм k-середніх не дозволяє виділити кластери більш складної форми.

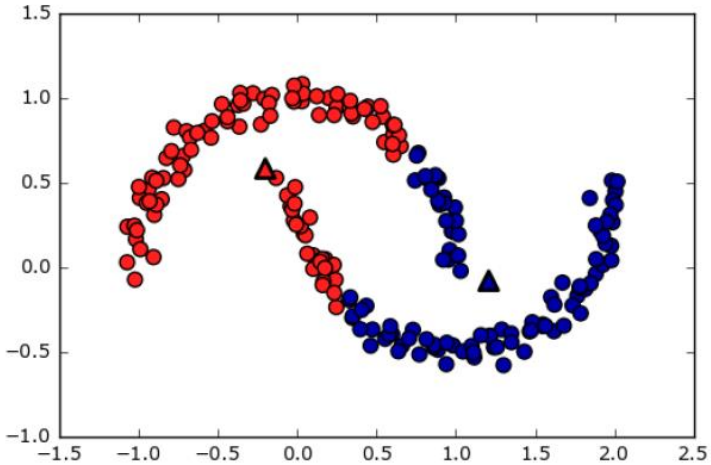

Рисунок 2.2 – кластеризація методом k-середніх у випадку складної форми вхідних даних

2.2.2 Агломеративна кластеризація

Алгомеративна кластеризація (agglomerative clustering) відноситься до сімейства алгоритмів кластеризації, в основі яких лежать однакові принципи: алгоритм починає свою роботу з того, що кожну точку даних заносить в свій власний кластер і по мірі виконання об'єднує два найбільш схожих між собою кластера до тих пір, поки не буде задоволено певний критерій зупинки. Критерій зупинки, реалізований в scikit-learn - це кількість кластерів, тому схожі між собою кластери об'єднуються до тих пір, поки не залишиться задане число кластерів. Є кілька критеріїв зв'язку (linkage), які задають точний спосіб вимірювання «найбільш схожого кластера». В основі цих критеріїв лежить відстань між двома існуючими кластерами. В якості відстані обираються різні функції:

1) мінімальна відстань:

$$d_{min}(c_i, c_j) = \min\{||x - y||, x \epsilon c_i, y \epsilon c_j, \tag{2.7}$$

При використанні даної функції алгоритм сприяє «витягуванню» кластерів. Основний недолік в цьому випадку – чутливість до шуму.

2) максимальна відстань

$$d_{\max}(c_i, c_j) = \max\{||x - y||, x \epsilon c_i, y \epsilon c_j \tag{2.8}$$

Алгоритм на основі максимальної відстані сприяє формуванню компактних кластерів, а основним недоліком є ефект порушення кластерів витягнутої форми.

3) середня відстань

$$\tilde{d}(c_i, c_j) = \frac{1}{n_i n_j} \sum_{x \epsilon c_i} \sum_{y \epsilon c_j} ||x - y|| \,, \tag{2.9}$$

де n_i, n_j – кількість об’єктів відповідно у i-тому та j-тому кластері

4) відстань між центрами кластерів

$$d_\mu(c_i, c_j) = ||\mu_i - \mu_j|| \tag{2.10}$$

На практиці найчастіше найкращі результати показують алгоритми на основі середньої відстані, проте, з точки зору ефективності з обчислювальної точки зору найбільш ефективні алгоритми на основі відстані між центрами кластерів [13].

Приклад роботи алгоритму (рис. 2.3) [5]:

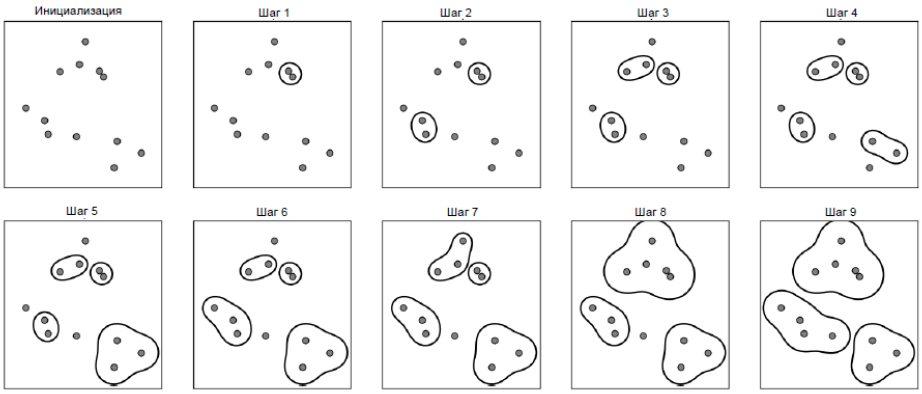

Рисунок 2.3 – приклад роботи агломеративної кластеризації

Спочатку кількість кластерів дорівнює кількості точок даних. Потім на кожному кроці об'єднуються два найближчих один до одного кластера. На

перших чотирьох кроках вибираються кластери, що складаються з окремих точок, і об'єднуються в кластери, що складаються з двох точок. На кроці 5 один з 2-точкових кластерів вбирає в себе третю точку і т.д. На кроці 9 у нас залишається три кластери. Оскільки ми встановили кількість кластерів рівним 3, алгоритм зупиняється.

2.2.3 DBSCAN

Ще один дуже корисний алгоритм кластеризації - DBSCAN (density-based spatial clustering of applications with noise - алгоритм кластеризації на основі щільності просторових даних з присутністю шуму). Основні переваги алгоритму DBSCAN полягають в тому, що користувачеві не потрібно заздалегідь задавати кількість кластерів, так як алгоритм може виділити кластери складної форми і здатний визначити точки, які не належать якомусь кластеру. DBSCAN працює трохи повільніше, ніж алгоритм агломеративної кластеризації і алгоритм k-середніх, але також може масштабуватись на відносно великі набори даних.

DBSCAN визначає точки, що розташовані в «густонаселених» областях простору характеристик, коли багато точок даних розташовані близько один до одного. Ці області називаються щільними (dense) областями простору характеристик. Ідея алгоритму DBSCAN полягає в тому, що кластери утворюють щільні області даних, які відокремлені один від одного відносно порожніми областями.

Точки, що знаходяться в щільній області, називаються ядровими точками (core points). Алгоритм DBSCAN має два параметри: min_samples і eps. Якщо принаймні min_samples точок знаходяться в радіусі околиці eps розглянутої

точки, то ця точка класифікується як ядро. Ядрові точки, відстані між якими не перевищують радіус околиці eps, поміщаються алгоритмом DBSCAN в один і той же кластер.

На старті алгоритм вибирає довільну точку. Потім він знаходить всі точки, віддалені від стартової точки на відстані, що не перевищує радіус околиці eps. Якщо безліч точок, що знаходяться в межах радіусу околиці eps, менше значення min_samples, стартова точка позначається як шум (noise), це означає, що вона не належить ніякому кластеру. Якщо ця множина точок більше значення min_samples, стартова точка позначається як ядерна і їй призначається мітка нового кластера. Потім перебираються всі сусіди цієї точки (що знаходяться в межах eps). Якщо вони ще не були присвоєні кластеру, їм присвоюється мітка щойно створеного кластеру. Якщо вони є ядровими точками, по черзі перебираються їх сусіди і т.д. Кластер зростає до тих пір, поки не залишиться жодної ядерної точки в межах радіусу околиці eps. Потім вибирається інша точка, яка ще не відвідали, і повторюється та ж сама процедура.

Приклад роботи алгоритму DBSCAN бачимо на рис.2.4 [5]:

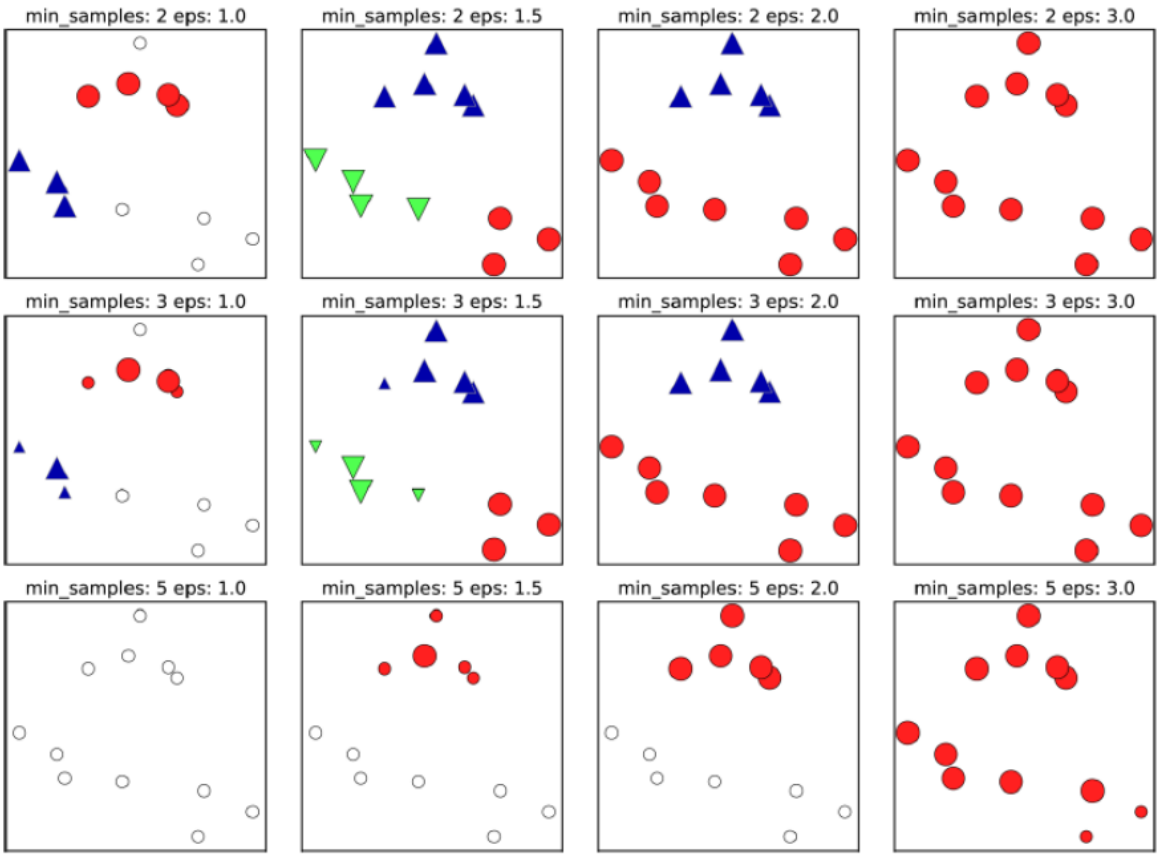

Рисунок 2.4 – приклад роботи алгоритму DBSCAN

На цьому графіку точки, які належать кластерам, пофарбовані суцільним кольором, а шумові точки - білим кольором. Ядрові точки показані в вигляді великих маркерів, тоді як граничні точки відображаються у вигляді невеликих маркерів. Збільшення значення eps (зліва направо на малюнку) означає включення більшої кількості точок в кластер. Дуже маленьке значення eps означатиме відсутність ядерних точок і може привести до того, що всі точки будуть позначені як шумові. Дуже велике значення eps призведе до того, що всі точки сформують один кластер. Значення min_samples головним чином визначає, чи будуть точки, розташовані в менш щільних областях, позначені як викиди або як кластери. Якщо збільшити значення min_samples, все, що могло б стати кластером з кількістю точок, що не перевищує min_samples, буде позначено як шум. Тому значення min_samples задає мінімальний розмір кластера.

2.2.4 Алгоритми нечіткої кластеризації

Вхідною інформацією для вирішення задач кластеризації є множина з N даних у вигляді n-мірних векторів ознак $X = \{x_1, x_2, x_3, \dots, x_n\}, x_k \in X, k = 1,2,..,N$. Результатом являється розподілення вхідних даних на m кластерів з деяким значенням рівня належності $w_{k,j} \in [0,1]$ k-го вектору ознак до j-го кластеру. При цьому виконується розрахунок N*m матриці $W = \{w_{k,j}\}$, яка називається матрицею нечіткого розбиття.

Коли елементи матриці W розглядаються як ймовірності гіпотез про належність вектора x_k до відповідних кластерів, кластеризація називається ймовірнісною. Найбільш суттєвим недоліком ймовірнісного підходу являється вимога, щоб сума функцій належності кожного вектору була рівна одиниці. Виконати це обмеження дозволяють так звані ймовірнісні методи фаззі-кластеризації.

Алгоритми фаззі-кластеризації, що засновані на цільових функціях, призначені для вирішення задач шляхом оптимізації деякого наперед заданого критерію якості кластеризації і являються найбільш строгими з математичної точки зору. Цільова функція, що мінімізується, має наступний вигляд:

$$E\left(w_{k,j}, c_j\right) = \sum_{k=1}^{N} \sum_{j=1}^{m} w_{k,j}^{\beta}\, d^2(x_k, c_j) \tag{2.11}$$

при обмеженнях:

$$\sum_{j=1}^{m} w_{k,j} = 1\,,\ \ k = 1, \dots, N \tag{2.12}$$

$$0 < \sum_{k=1}^{N} w_{k,j} < N,\ \ j = 1,..,m\,, \tag{2.13}$$

де c_j – прототип (центр) j-го кластера, β – невід'ємний параметр «фаззифікатор» (зазвичай , β = 2) , $d^2(x_k, c_j)$ – відстань між x_k і c_j.

Вирішення задачі нечіткої кластеризації можна звести до оптимізації функції Лагранжа:

$$L\left(w_{k,j}, c_j, \lambda_k\right) = \sum_{k=1}^{N} \sum_{j=1}^{m} w_{k,j}^{\beta} d^2\left(x_k, c_j\right) + \sum_{k=1}^{N} \lambda_k (\sum_{j=1}^{m} w_{k,j} - 1) =$$
$$= \sum_{k=1}^{N} (\sum_{j=1}^{m} w_{k,j}^{\beta} d^2\left(x_k, c_j\right) + \lambda (\sum_{j=1}^{m} w_{k,j} - 1), \qquad (2.14)$$

де λ_k - невизначений множник Лагранжа.

Далі вирішується система рівнянь Куна-Такера:

$$\begin{cases} \frac{\partial L(w_{k,j}, c_j, \lambda_k)}{\partial w_{k,j}} = 0 \\ \nabla_{c_j} L\left(w_{k,j}, c_j, \lambda_k\right) = 0 \\ \frac{\partial L(w_{k,j}, c_j, \lambda_k)}{\partial \lambda_k} = 0 \end{cases} \qquad (2.15)$$

Отримуємо рішення у вигляді:

$$w_{k,j} = \frac{\left(d^2(x_k, c_j)\right)^{\frac{1}{1-\beta}}}{\sum_{l=1}^{m} (d^2(x_k, c_l)^{\frac{1}{1-\beta}}} \qquad (2.16)$$

$$c_j = \frac{\sum_{k=1}^{N} w_{k,j}^{\beta} x_k}{\sum_{k=1}^{N} w_{k.j}^{\beta}} \qquad (2.17)$$

$$\lambda_k = -(\sum_{l=1}^{m} (\beta d^2(x_k, c_l))^{\frac{1}{1-\beta}})^{1-\beta} \qquad (2.18)$$

Рівняння (1.6)-(1.8) породжують широкий клас процедур кластеризації. Обираючи $\beta = 2$ і евклідову метрику $d^2(x_k, c_j) = \|x_k - c_j\|^2$, отримуємо простий і ефективний алгоритм нечіткої кластеризації Бездека (fuzzy c-means):

$$w_{k,j} = \frac{\|x_k - c_j\|^{-2}}{\sum_{l=1}^{m} \|x_k - c_l\|^{-2}} \tag{2.19}$$

$$c_j = \frac{\sum_{k=1}^{N} w_{k,j}^2 \, x_k}{\sum_{k=1}^{N} w_{k,j}^2} \tag{2.20}$$

$$\lambda_k = -\sum_{l=1}^{m} \left(\frac{\|x_k - c_l\|^{-2}}{2} \right)^{-1} \tag{2.21}$$

Також до ймовірнісних алгоритмів кластеризації відносять алгоритми Густафсона-Кесселя, Гата-Геви та інші. Алгоритм нечітких С-середніх - найпопулярніший з алгоритмів нечіткої кластеризації, проте має деякі обмеження і сильні апріорні припущення про характер даних в вибірці, що звужують область його застосовності. Одне з обмежень алгоритму нечітких С-середніх є використання функції приналежності, яка визначається евклідовою метрикою. Нескладно помітити, що значення приналежності $w_{j,k}$ обчислене за алгоритмом буде однаковим для всіх точок, що знаходяться на однаковій відстані від центроїда c_j, а це значить, що кластери матимуть строго сферичну форму. У той же час в реальних задачах нерідко зустрічаються дані, в яких форма скупчень сильно відрізняється від (гіпер) сфери, або шкали значень окремих компонент векторів спостережень мають різний масштаб. Апріорне припущення про сферичну форму кластерів робить в таких випадках глобальний мінімум функціоналу, недосяжний для алгоритму нечітких С-середніх. Узагальненням алгоритму нечітких С-середніх є алгоритм Густафсона-Гесселя, здатний виділяти кластери еліпсоїдальної форми. Для цього в формулу розрахунку відстаней між векторами вводиться масштабуюча матриця А, яка змінює масштаб простору по кожній координаті [15].

2.3 Способи обрання кількості кластерів

Так як задача кластеризація є задачею без вчителя, ми завчасно не знаємо, яка кількість кластерів є найбільш оптимальною для вхідного набору даних. Існує досить велика кількість методів, що дозволяє знайти найоптимальнішу кількість кластерів. У даному пункті будуть описані деякі з них.

2.3.1 «Ліктьовий» метод

Ліктьовий метод заклечається в тому, що будується певна кількість моделей (для різної кількості кластерів) та обчислюється метрика WCSS (within-cluster sum of squares) – сума квадратей відстаней від точок до центрів кластерів:

$$WCSS = \sum_{P_i\ in\ Cluster\ 1} distance(P_i, C_1)^2 + \sum_{P_i\ in\ Cluster\ 2} distance(P_i, C_2)^2 + \\ + \cdots + \sum_{P_i\ in\ Cluster\ N} distance(P_i, C_N)^2, \quad (2.22)$$

де C_i – відповідний центроїд і-того кластеру

Очевидно, що ця метрика зі збільшенням кількості кластерів буде прямувати до 0, однак, починаючи з певного кластера це зменшення буде незначним. Відповідно, така кількість кластерів і буде оптимально.

Розглянемо приклад обрання оптимальної кількості кластерів ліктьовим

методом. Для цього був побудований графік залежності метрики wcss від кількості кластерів (див. рис.2.5):

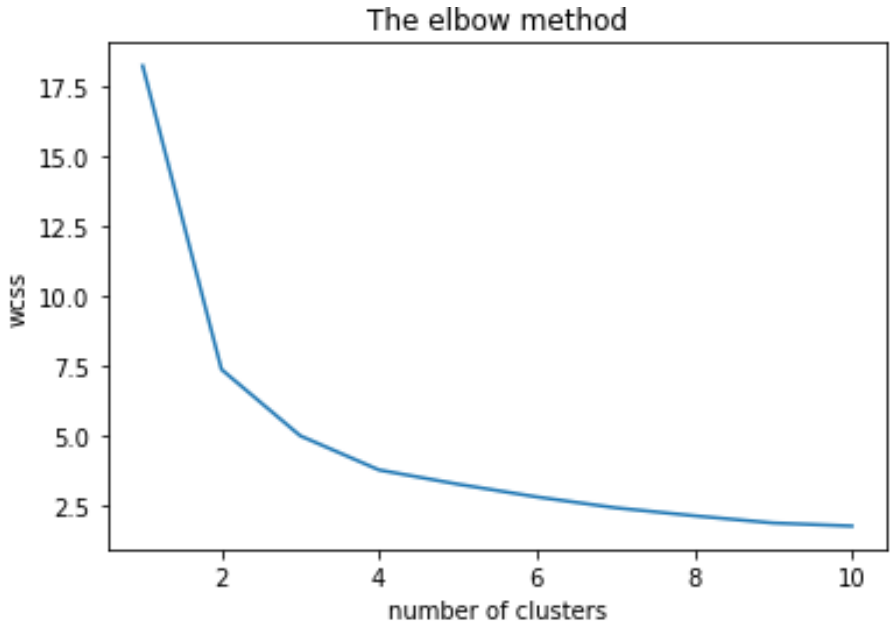

Рисунок 2.5 – приклад знаходження оптимальної кількості кластерів ліктьовим методом

Як можемо бачити, вже після 4 кластерів, зменшення метрики є незначним. Тому, оптимальною кількістю кластерів можемо вважати 3 або 4.

2.3.2 Метод дендрограми

Визначення оптимальної кількості кластерів за допомогою дендрограми використовується при ієрархічній кластеризації. Для цього будується спеціальне дерево,яке відображає процес поступового з’єднання кластерів. У даному методі для того, щоб визначити оптимальну кількість кластерів необхідно уважно подивитись на висоту, на якій будь-які два об’єкти

з’єднуться разом. Чим менше ця висота, тим більш схожими є з’єднувані об’єкти. Точну відповідь на те, скільки саме повинно бути кластерів, дендрограмма не надає , проте вона допомагає досліднику визначити вирішити, скільки саме кластерів варто пробувати моделювати.

Розглянемо даний метод на прикладі (див. рис. 2.6).

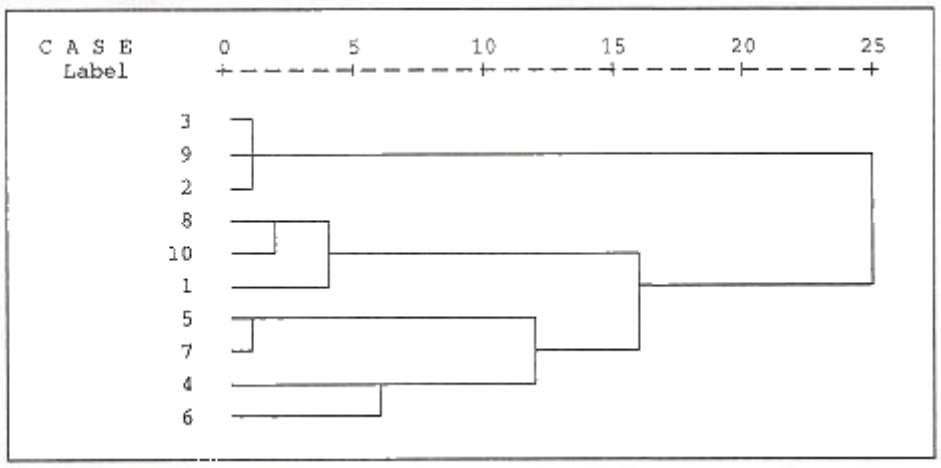

Рисунок 2.6 – приклад визначення оптимальної кількості кластерів за допомогою дендрограми

На дендрограмі (рис. 2.6) номера об'єктів слідують по вертикалі. По горизонталі відзначені відстані (в умовних одиницях), на яких відбувається об'єднання об'єктів в кластери. На перших кроках відбувається утворення кластерів: (3,9,2) і (5,7). Далі утворюється кластер (8, 10, 1) - відстані між цими об'єктами більше, ніж між тими, які були об'єднані на попередніх кроках. Наступний кластер - (4, 6). Далі в один кластер об'єднуються кластери (5, 7) і (4, 6), і т.д. Процес закінчується об'єднанням всіх об'єктів в один кластер. Кількість кластерів визначає по дендрограмі сам дослідник. Так, судячи з дендрограмі, в даному випадку можна виділити три або чотири кластери [19].

2.4 Критерії оцінки моделей та прогнозів

Для порівняння отриманих результатів моделювання та прогнозування різними методами аналізу даних використовувались наступні критерії адекватності моделей та критерії оцінки якості отриманих прогнозів.

2.4.1 Критерії адекватності моделей

1. Коефіцієнт детермінації (R^2):

$$R^2 = \frac{Var(\tilde{y})}{Var(y)}, \tag{2.23}$$

де $Var(\tilde{y})$ – дисперсія часового ряду, оціненого побудованою моделлю, $Var(y)$ – дисперсія точних значень часового ряду

За побудовою, коефіцієнт детермінації приймає значення від 0 до 1. Чим ближче значення до 1, тим більш адекватною є модель, тобто роль сторонніх факторів є малою, і модель добре апроксимує вихідні дані.

2. Критерій Дурбіна-Уотсона:

$$DW = \frac{\sum_{k=2}^{N}\left(\varepsilon(k)-\varepsilon(k-1)\right)^2}{\sum_{k=1}^{N}\varepsilon^2(k)} \tag{2.24}$$

Статистика Дарбіна-Уотсона приймає значення від 0 до 4. Він показує ступінь автокореляції залишків побудованої моделі. Для ідеальної моделі значення наближається до 2, що означає відсутність кореляції.

3. Сума квадратів похибок

$$SSE = \sum_{i=1}^{N}(y_i - \widetilde{y_i})^2 \tag{2.25}$$

Критерій *SSE* є сумою квадратів похибки побудованої моделі. Чим ближче значення *SSE* до 0, тим адекватнішою є модель [16].

2.4.2 Критерії якості прогнозів

1. Середньоквадратична похибка:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{T}(y_i - \tilde{y}_i)^2} \tag{2.26}$$

2. Середня абсолютна похибка у відсотках:

$$MAPE = \frac{1}{N}\frac{\sum_{i=1}^{N}|y_i - \widetilde{y_i}|}{y_i} \cdot 100\% \tag{2.27}$$

3. Коефіцієнт Тейла:

$$Theil = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{T}(y_i - \tilde{y}_i)^2}}{\sqrt{\frac{1}{N}\sum_{i=1}^{T}y_i^2} + \sqrt{\frac{1}{N}\sum_{i=1}^{T}\tilde{y}_i^2}} \tag{2.28}$$

Очевидно, що чим менше значення критеріїв середньоквадратичної похибки та середньої абсолютної похибки у відсотках, тим якіснішим є отриманий прогноз.

Коефіцієнт Тейла за своєю побудовою приймає значення від 0 до 1. Чим менше значення коефіцієнту Тейла, тим якіснішим є прогноз. У випадку прогнозовані значення точно співпадають з реальними і модель є ідеальною [16].

2.4.3 Критерії якості кластеризації

Так як при вирішенні задачі кластеризації, ми не знаємо завчасно правильне, достовірне розбиття, метрик для порівняння алгоритмів існує не дуже багато. Ідея цих метрик насправді полягає у тому, наскільки «схожими» виявляються точки у кожному кластері, наскільки «щільними» побудовані кластери є. Тому у своїй роботі я застосував наступні метрики:

1) компактність кластерів

Ідея даного методу в тому, що чим ближче один до одного знаходяться об'єкти всередині кластерів, тим краще поділ. Таким чином, необхідно мінімізувати суму квадратів відхилень:

$$WSS = \sum_{j=1}^{M} \sum_{i=1}^{|C_j|} \left(x_{ij} - \bar{c}_j\right)^2, \tag{2.29}$$

де M – кількість кластерів [17].

2) Silhouette criterion

Критерій силуету вказує на те, наскільки об’єкти схожі на свій кластер в порівнянні з іншими кластерами.

Нехай маємо кластеризацію на k кластерів, точка даних i потрапила у кластер C_i. Тоді:

$$a(i) = \frac{1}{|C_i|-1} \sum_{j \in C_i, i \neq j} d(i,j), \tag{2.30}$$

де $a(i)$ – середня відстань між точкою i та іншими точками у тому ж кластері. Дане число можемо інтерпретувати як те, наскільки добре присвоюється точці i відповідний кластер (чим менше це число, тим краще)

Також для кожної точки i позначимо

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i,j) \tag{2.31}$$

де b(i) – середня відстань i до всіх точок в будь-якому іншому кластері (не в тому, де знаходиться точка i). Кластер із цією найменшою середньою відстанню називається «сусіднім кластером» i він є наступним найкращим кластером для точки i.

Критерій силуету визначається наступним чином:

$$s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}} \tag{2.32}$$

Інакше можна записати:

$$s(i) = \begin{cases} 1 - \frac{a(i)}{b(i)}, \ \ if\ a(i) < b(i) \\ 0, \ if\ a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1, if\ a(i) > b(i) \end{cases} \tag{2.33}$$

Виходячи з означення, $-1 \leq s(i) \leq 1$

Сенс цього показника полягає в тому, наскільки щільно згруповані точки кожного кластеру. Чим ближчий до 1 показник s(i), тим кластеризація краща [18].

2.5 Висновки до розділу

Отже, у ході виконання другого розділу магістерської дисертації були обрані та описані математичні моделі для подальшого використання до поставленої задачі сталого розвитку. Зокрема, розглянуто регресійний підхід до моделювання процесів, а саме такі моделі як авторегресія, авторегресія з ковзним середнім, авторегресія з інтегрованим ковзним середнім, моделі для гетероскедастичних процесів. Описані різні методи інтелектуального аналізу даних для вирішення задачі кластеризації, розглянуто методи обрання оптимальної кількості кластерів.

Крім того, висвітлені критерії, за допомогою яких далі буде здійснене порівняння побудованих моделей, прогнозів, а також деякі специфічні критерії для аналізу моделей кластеризації. Так як ми не можемо поділити вибірку даних на тренувальну та перевірочну, такі критерії лише приблизно висвітлюють, наскільки «чітко» були розподілені об’єкти по кластерам (наскільки схожі об’єкти у одному кластері між собою, та наскільки різні об’єкти, що належать різним кластерам).

Задачею до наступного розділу є розробка програмного забезпечення для побудови математичних моделей, прогнозування та кластеризації вхідних даних, побудова порівняльних таблиць та обрання найкращих моделей серед моделей-кандидатів.

РОЗДІЛ 3 ПОБУДОВА МОДЕЛЕЙ КЛАСТЕРИЗАЦІЇ ТА ПРОГНОЗУВАННЯ ПОКАЗНИКІВ

3.1 Вибір функціональної платформи

Наразі існує велика кількість програмного інструментарію, що дозволяє виконувати дослідження по аналізу даних. У своїй магістерській дисертації мною була використана мова програмування Python, яка є найпоширенішою для аналізу даних. Вона містить чималий пакет бібліотек, функцій, що дозволяють швидко та зручно виконувати математичне моделювання та обчислення.

Програмний продукт був розроблений у середовищі програмування Spyder, що вбудований у дистрибутив Anaconda. Це інтуїтивно зрозуміле середовище, що дозволяє запускати код, має зручний інтерфейс та можливість у реальному часі змінювати значення змінних тощо. Також дана платформа добре взаємодіє з такими інструментами як Jupiter , Glueviz, RStudio, PyCharm, ...

3.2 Функціональна схема програмного продукту

Як було сказано вище, програмний продукт був розроблений мовою програмування Python у середовищі Spyder. У цьому пункті магістерської дисертації описано схему роботи програми, основні задачі та методи її роботи.

Перш за все, програма зчитує вхідні дані у форматі .xlsx, що містять інформацію про показники сталого розвитку для країн ООН. Вхідний датасет буде описаний у розділі 3.3, він міститиме 18 змінних. Після зчитування даних відбувається попередня їх обробка, а саме заповнення пропусків (середнім

значенням по стовпчику) та, у разі необхідності для деяких моделей, нормування. Після попередньої обробки даних необхідно проаналізувати датасет та визначити оптимальну кількість кластерів. Для кластеризації к-середнім методом це число визначається ліктьовим методом, у випадку агломеративної кластеризації – методом дендрограми. На наступному кроці після визначення оптимальної кількості кластерів будується ряд моделей для різного k (k обирається аналітиком на основі проведеного попереднього аналізу). Після побудови моделей здійснюється виведення результату у Excel-файл, а також визначаються центроїди побудованих кластерів для розуміння відмінностей між кластерами. На останньому кроці визначаються критерії якості кластеризації. У випадку моєї роботи, обраховуються критерії WSS та критерій Silhouette (описані у розділі 2 магістерської дисертації). На основі отриманих критеріїв визначається найкраща модель кластеризації.

Для побудови моделей процесів сталого розвитку та побудови прогнозів були використані часові ряди показників сталого розвитку. Програма працює наступним чином. Спочатку відбувається завантаження вхідного часового ряду, після чого він аналізується на стаціонарність, не лінійність, гетероскедастичність. На другому кроці здійснюється побудова графіку ряду, побудова часткової автокореляційної функції для визначення оптимального порядку моделі авторегресії. На третьому кроці будується модель авторегресії, знаходяться критерії адекватності моделі. У випадку задовільності критеріїв адекватності, виконується прогнозування даних на декілька кроків уперед. Останній етап – виведення результатів моделювання, рівняння моделі, побудова графіків прогнозів та розрахунок критеріїв якості прогнозів.

Зобразимо на рисунку функціональну схему програмного продукту (див. рис.3.1).

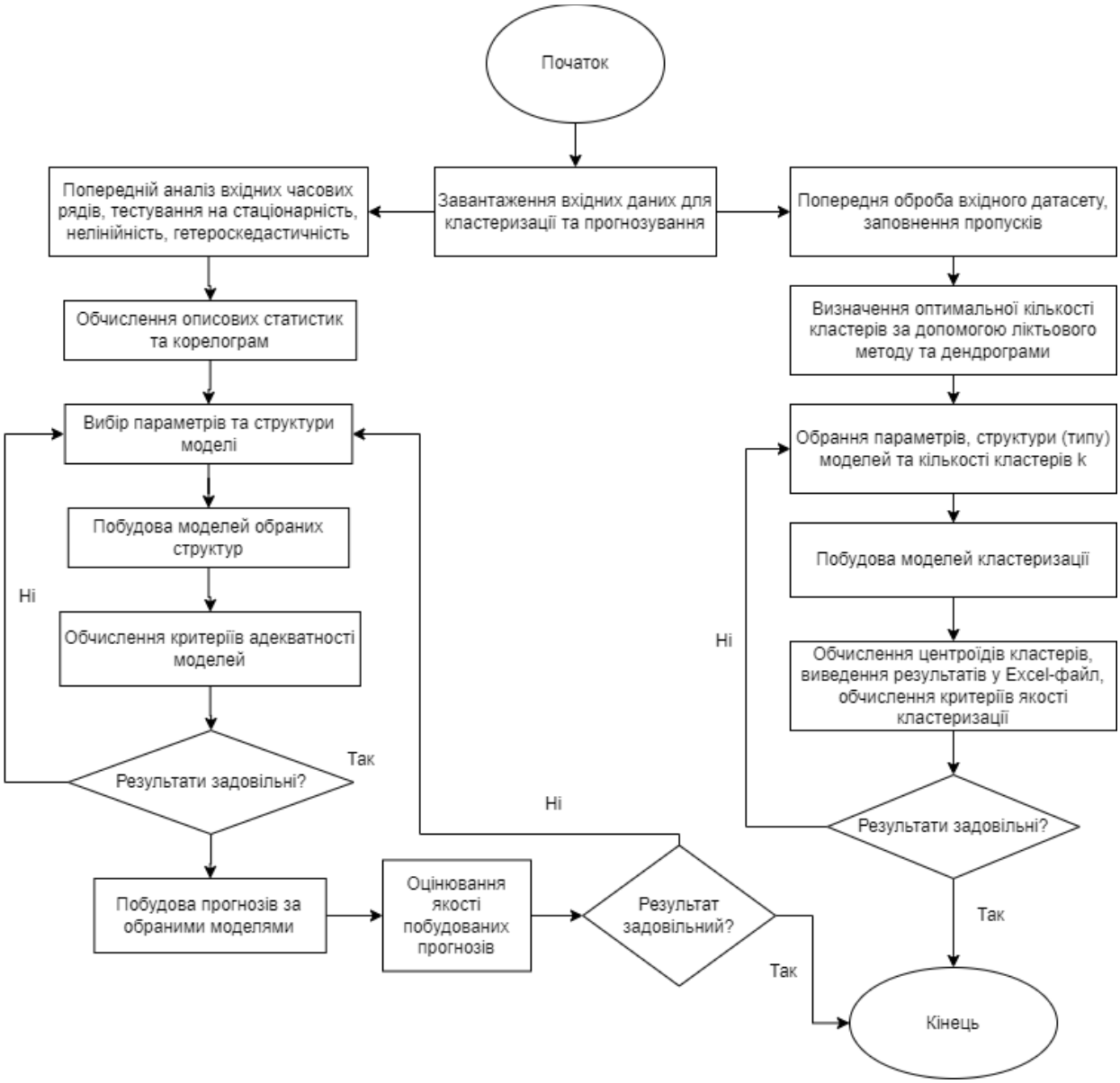

Рисунок 3.1 – Функціональна схема програмного продукту

Під час реалізації програмного продукту для ефективності обчислень та виконання необхідних задач були використані вбудовані бібліотеки Python, а саме:

1) numpy – для роботи з масивами даних та застосування математичних операцій лінійної алгебри;
2) pandas – для роботи з датафреймами та часовими рядами;
3) matplotlib – для роботи з графіками;

4) sklearn – для побудови моделей інтелектуального аналізу даних;

5) scipy – для побудови дендрограми;

6) statsmodels – для побудови корелограм та моделей регресії часових рядів.

3.3 Моделі кластеризації країн ООН за показниками сталого розвитку

Для виконання кластеризації були обрані дані за 2020 рік для 136 країн світу, що відображають досягнення кожної країни у кожній цілі сталого розвитку (загалом таких цілей 17, описані у першому розділі магістерської дисертації) [22]. Датасет вміщує в себе наступні показники:

1. Розрахунковий відсоток населення, що живе за межею бідності, що становить 3,20 доларів США на день. Оцінено з використанням історичних оцінок розподілу доходів, прогнозів зміни чисельності населення за віком та рівнем освіти, а також прогнозів ВВП. Показує рівень досягнення країною першої цілі сталого розвитку (ліквідації бідності).
2. Відсоток населення, споживання їжі якого є недостатнім для задоволення дієтичних енергетичних потреб протягом як мінімум одного року. Відповідає за другу ціль сталого розвитку – ліквідація голоду.
3. Середня кількість років, яку новонароджений може очікувати прожити, якщо він або вона проживе життя, схильне до впливу коефіцієнтів смертності залежно від статі та віку, що переважають на момент його або її народження, протягом певного року, протягом певного періоду часу. для цієї країни, території чи географічного регіону.

Відображає досягнення країни у третій цілі сталого розвитку (здоров'я і благополуччя)

4. Відсоток дітей шкільного віку, які відвідують початкову школу – четверта ціль сталого розвитку (якісна освіта).

5. Змодельована оцінка частки економічно активного жіночого населення віком 15 років і старше, поділена на таку ж пропорцію для чоловіків. Відображає п'яту ціль сталого розвитку – гендерна рівність.

6. Відсоток населення, що користується принаймні базовими послугами питної води, такими як питна вода з покращених джерел, за умови, що час збору не перевищує 30 хвилин для поїздки туди і назад, включаючи черги. Показник шостої цілі сталого розвитку – чиста вода.

7. Відсоток населення, яке користується хоча б базовими санітарними послугами, наприклад, покращеними санітарними спорудами, які не використовуються спільно з іншими домогосподарствами. Є показником також шостої цілі сталого розвитку – санітарія.

8. Відсоток населення, що має доступ до електрики. Відображає сьому ціль сталого розвитку – недороговартісна і чиста енергія.

9. Змодельована оцінка частки робочої сили, яка не має роботи, але доступна та активно шукає роботу. Індикатор відображає нездатність економіки створювати робочі місця для людей, які хочуть працювати, але не роблять цього (восьма ціль сталого розвитку – гідна робота та економічне зростання).

10. Відсоток населення, яке користувалося Інтернетом із будь-якого місця за останні три місяці. Доступ може бути через фіксовану або мобільну мережу (індустріалізація, інновації та інфраструктура).

11. Коефіцієнт Джині скоригований з урахуванням максимальних доходів, не врахованих в обстеженнях домашніх господарств. Цей індикатор являє собою середнє значення

нескоректованого та скоригованого Джіні. Є індикатором десятої цілі сталого розвитку (зменшення нерівності).

12. Забруднення повітря, що вимірюється як середньозважена за чисельністю населення середньорічна концентрація PM2,5 для населення країни. PM2,5 - це зважені частинки з аеродинамічним діаметром менше 2,5 мікрон, які здатні проникати глибоко в дихальні шляхи і можуть завдати серйозної шкоди здоров'ю. Відповідає 11 цілі сталого розвитку – зменшення забруднення навколишнього середовища.

13. Відсоток опитаного населення, яке відповіло «задоволено» на запитання «У місті чи районі, де ви живете, задоволені чи незадоволені системою громадського транспорту?».

14. Викиди від спалювання та окислення викопного палива та від виробництва цементу. Показник не включає викиди від палива, що використовується для міжнародних авіаційних та морських перевезень. Відповідає 13 цілі сталого розвитку – боротьба зі зміною клімату.

15. Індекс, що ґрунтується на реальних змінах у кількості видів у кожній категорії ризику зникнення у Червоній книзі видів, що знаходяться під загрозою зникнення. Є показником 14-15 цілей сталого розвитку (збереження екосистем суші та морських екосистем).

16. Кількість навмисних вбивств на 100 000 осіб. Є показником 16 цілі сталого розвитку (мир, правосуддя, ефективні інститути).

17. Рівні корупції в державному секторі за шкалою від 0 (найвищий рівень корупції, що сприймається) до 100 (найнижчий рівень корупції, що сприймається). Є показником 16 цілі сталого розвитку (мир, правосуддя, ефективні інститути).

18. Сума державних витрат на охорону здоров'я із внутрішніх джерел та загальних державних витрат на освіту (поточні, капітальні та трансферти), виражена у відсотках від ВВП.

Загалом датасет складається з 137 рядків (137 країн) та 18 стовпчиків (показники, за якими виконується кластеризація).

3.3.1 Кластеризація методом k-найближчих

Першочерговою задачею для виконання кластеризації є визначення оптимальної кількості кластерів для вхідної задачі. Для цього застосовувався ліктьовий метод, описаний у другому розділі магістерської дисертації.

Зобразимо графік залежності метрики WCSS від кількості кластерів (див. рис. 3.2)

.

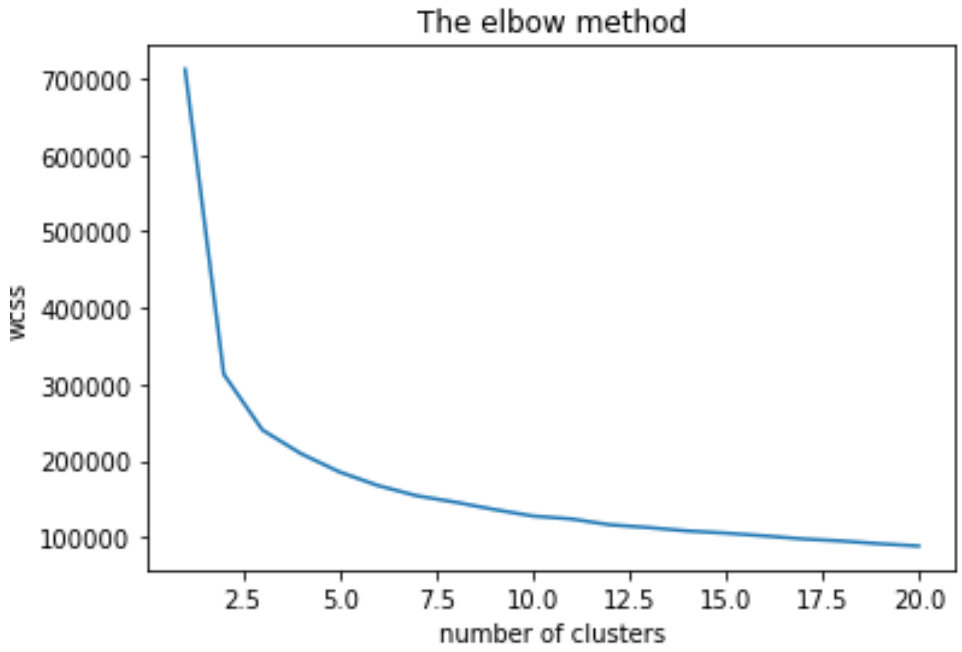

Рисунок 3.2 – Визначення оптимальної кількості кластерів для методу k-nearest

Для більшої наочності та точності визначення кількості кластерів побудуємо таблицю значень WCSS (див. табл. 3.1):

Таблиця 3.1 – Значення WCSS в залежності від кількості кластерів

Кількість кластерів (k)	WCSS	Різниця з попереднім значенням WCSS
0	711723.14	-
1	313378.298	-398344.842
2	240394.9559	-72983.3421
3	209427.2257	-30967.7302
4	185452.4821	-23974.7436
5	167605.7932	-17846.6889
6	154342.4752	-13263.318
7	145919.431	-8423.0442
8	136432.3855	-9487.0455
9	127896.1221	-8536.2634
10	124044.0414	-3852.0807
11	116412.8718	-7631.1696
12	112799.5308	-3613.341
13	108385.3244	-4414.2064
14	105510.3936	-2874.9308
15	101910.5862	-3599.8074
16	97871.63381	-4038.95239
17	95245.62407	-2626.00974
18	91544.91589	-3700.70818
19	88678.35011	-2866.56578

Можемо бачити, що метрика значно покращується приблизно до 5-9 кількості кластерів, далі покращення вже незначне. Тому доцільно брати k від 5 до 9 та порівняти отримані результати.

Наведемо спочатку результати для кількості кластерів k=5 (див. рис.3.3).

Country	0
Albania	4
Algeria	3
Angola	1
Argentina	4
Armenia	4
Australia	0
Austria	0
Azerbaijan	4
Bangladesh	2
Belarus	0
Belgium	0
Benin	1
Bolivia	4
Botswana	4
Brazil	4
Bulgaria	4

Рисунок 3.3 – Результати кластеризації методом к-середніх (к=5)

Для більш зручного представлення результатів, отримані значення завантажемо у Excel та згрупуємо країни по кластерам у вигляді таблиці (див. табл. 3.2).

Таблиця 3.2 – Кластеризація країн методом к-середніх при к=5

№ Кластеру	Об’єкти кластеру
0	Australia, Austria, Belarus, Belgium, Canada, Chile, Costa Rica, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Israel, Italy, Japan, Kazakhstan, Korea Rep., Latvia, Lithuania, Luxembourg, Malaysia, Malta, Netherlands, New Zealand, Norway, Poland, Portugal, Singapore, Slovak Republic, Slovenia, Spain, Sweden, Switzerland, United Arab Emirates, United Kingdom, United States, Uruguay
1	Angola, Benin, Burkina Faso, Central African Republic, Chad, Congo Rep., Ethiopia, Guinea, Haiti, Liberia, Madagascar, Malawi, Mali, Mozambique, Niger, Rwanda, Sierra Leone, South Sudan, Tanzania, Togo, Uganda, Zambia
2	Bangladesh, Cameroon, Cote d'Ivoire, Eswatini, Gambia, The,Ghana, Kenya, Lesotho, Mauritania, Myanmar, Namibia, Nepal,Nigeria,Pakistan,Senegal,Solomon Islands,Sudan
3	Algeria, Egypt, Arab Rep.,India, Iran,, Iraq, Jordan, Morocco, Saudi Arabia, Tunisia, Turkey
4	Albania, Argentina, Armenia, Azerbaijan, Bolivia, Botswana, Brazil, Bulgaria, Cambodia, China, Colombia, Dominican Republic, Ecuador, El Salvador, Georgia, Guatemala, Honduras, Indonesia, Jamaica, Kyrgyz Republic, Maldives, Mauritius, Mexico, Moldova, Mongolia, Montenegro, Nicaragua, North Macedonia, Panama, Paraguay, Peru, Philippines, Romania, Russian Federation, Serbia, South Africa, Sri Lanka, Tajikistan, Thailand, Ukraine, Uzbekistan, Venezuela, Vietnam

Визначимо координати центроїдів кожного кластеру (див. табл..3.3).

Таблиця 3.3 - Координати цетроїдів кластерів методом k-means (k=5)

Кластер	Координати центроїда
0	(0.523636 2.71295 80.5 98.9223 80.3221 99.4053 98.5451 100 6.90614 86.9527 36.2701 12.8616 63.2727 7.90452 0.889705 1.9308 66.8864 10.8239)
1	(75.4745 21.8091 63.3235 82.7861 90.0711 57.6071 23.8505 35.1807 5.26909 14.2203 45.25 39.7076 42.2273 0.275273 0.874409 6.85882 29.7273 4.98918)
2	(40.6111 13.7 65.2954 88.0284 72.2244 76.2741 42.8508 66.5257 8.67056 30.0013 46.0238 43.623 54.7222 0.661722 0.857 9.35767 34.1667 6.29211)
3	(12.563 7.25 75.0339 96.2296 28.3669 95.7962 89.7928 99.4836 12.257 64.8246 42.0263 56.4923 58.9 5.2442 0.8727 2.9258 38.1 7.4496)
4	(11.79 8.44302 74.1671 94.7852 70.6099 93.5953 86.1889 97.6689 8.51302 59.3224 46.2539 20.1354 62.5349 3.5204 0.821953 10.4925 36.0465 7.55444)

В даному випадку отримали наступні значення критеріїв якості кластеризації:

WSS = 185452.48214228416

Silhouette criterion = 0.28649303016826544

Виконаємо кластеризацію при k=6 методом k-середніх та зведемо результати у таблицю (див. табл. 3.4).

Таблиця 3.4 – Кластеризація країн методом к-середніх при к=6

№ Кластеру	Об’єкти кластеру
0	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria Colombia Costa Rica Croatia Dominican Republic Ecuador Georgia Greece Hungary Italy Kazakhstan Lithuania Malaysia Maldives Mauritius Mexico Moldova Montenegro North Macedonia Panama Paraguay Peru Poland Romania Russian Federation Serbia Slovak Republic Thailand Ukraine Vietnam
1	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Haiti Liberia Madagascar Malawi Mali Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
2	Bangladesh Cameroon Cote d'Ivoire Eswatini Gambia, The Ghana Kenya Lesotho Mauritania Namibia Nepal Nigeria Pakistan Senegal Solomon Islands Sudan Vanuatu
3	Algeria Egypt, Arab Rep. India Iran, Islamic Rep. Iraq Jordan Morocco Saudi Arabia Tunisia Turkey
4	Bolivia Botswana Cambodia China El Salvador Guatemala Honduras Indonesia Jamaica Kyrgyz Republic Mongolia Myanmar Nicaragua Philippines South Africa Sri Lanka Tajikistan Uzbekistan Venezuela
5	Australia Austria Belgium Canada Chile Cyprus Czech Republic Denmark Estonia Finland France Germany Iceland Ireland Israel Japan Korea, Rep. Latvia Luxembourg Malta Netherlands New Zealand Norway Portugal Singapore Slovenia Spain Sweden Switzerland United Arab Emirates United Kingdom United States Uruguay

Визначимо координати центроїдів відповідних кластерів (див. табл.3.5).

Таблиця 3.5 - Координати цетроїдів кластерів методом k-means (k=6)

Кластер	Координати центроїда
0	(3.87056 4.61667 76.4061 96.2234 73.6571 96.8206 92.693 99.7945 8.77111 72.9022 43.7847 17.257 57.8611 4.50725 0.847806 5.58767 42 7.97569)
1	(75.4745 21.8091 63.3235 82.7861 90.0711 57.6071 23.8505 35.1807 5.26909 14.2203 45.25 39.7076 42.2273 0.275273 0.874409 6.85882 29.7273 4.98918)
2	(40.6111 13.7 65.2954 88.0284 72.2244 76.2741 42.8508 66.5257 8.67056 30.0013 46.0238 43.623 54.7222 0.661722 0.857 9.35767 34.1667 6.29211)
3	(12.563 7.25 75.0339 96.2296 28.3669 95.7962 89.7928 99.4836 12.257 64.8246 42.0263 56.4923 58.9 5.2442 0.8727 2.9258 38.1 7.4496)
4	(20.5168 12.8026 71.7721 93.7353 66.9667 89.7593 79.4178 93.3382 7.44895 44.5057 47.1939 23.7442 67.7895 3.34353 0.815316 14.9543 32.5263 7.15174)
5	(0.320606 2.63848 81.3948 99.436 82.0339 99.6739 98.8047 100 6.49788 89.0113 34.8685 11.9243 66.0303 8.29918 0.887182 1.60921 72.4545 11.5372)

Отримали наступні критерії якості кластеризації:

WSS = 167605.7931943795

Silhouette criterion = 0.25926288843337275

Виконаємо кластеризацію при k=7 методом k-середніх та зведемо результати у таблицю (див. табл. 3.6).

Таблиця 3.6 – Кластеризація країн методом к-середніх при к=7

№ Кластеру	Об’єкти кластеру
0	Australia Austria Belgium Canada Chile Cyprus Czech Republic Denmark Estonia Finland France Germany Iceland Ireland Israel Japan Korea, Rep. Latvia Lithuania Luxembourg Malta Netherlands New Zealand Norway Portugal Singapore Slovenia Spain Sweden Switzerland United Arab Emirates United Kingdom United States Uruguay
1	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Liberia Madagascar Malawi Mali Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
2	Bangladesh Cameroon India Mauritania Nepal Nigeria Pakistan Senegal Sudan
3	Bolivia Cambodia El Salvador Guatemala Honduras Indonesia Jamaica Kyrgyz Republic Mongolia Myanmar Nicaragua Philippines South Africa Sri Lanka Tajikistan Uzbekistan Venezuela
4	Botswana Cote d'Ivoire Eswatini Gambia, The Ghana Haiti Kenya Lesotho Namibia Solomon Islands Vanuatu

Продовження таблиці 3.6

5	Algeria Egypt, Arab Rep. Iran, Islamic Rep. Iraq Jordan Morocco Saudi Arabia Tunisia Turkey
6	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria China Colombia Costa Rica Croatia Dominican Republic Ecuador Georgia Greece Hungary Italy Kazakhstan Malaysia Maldives Mauritius Mexico Moldova Montenegro North Macedonia Panama Paraguay Peru Poland Romania Russian Federation Serbia Slovak Republic Thailand Ukraine Vietnam

Визначимо центроїди кожного з кластерів (див. табл. 3.7).

Таблиця 3.7 - Координати цетроїдів кластерів методом k-means (k=7)

Кластер	Координати центроїда
0	(0.342941 2.63441 81.236 99.4473 82.1091 99.6112 98.6444 100 6.55471 88.7928 35.1418 11.9027 65.3529 8.19879 0.890176 1.69626 72.0882 11.4372)
1	(76.46 20.5524 63.2888 82.2807 90.1965 57.2328 23.3337 34.6999 4.82952 13.3511 45.4268 40.8713 42.8095 0.274524 0.881905 6.86733 30.2857 5.05048)
2	(42.8844 10.8889 68.0684 84.258 56.6584 79.2634 49.3812 70.6487 7.73333 29.0328 42.5392 68.684 56.3333 0.801111 0.849 6.98478 30 4.56167)

Продовження таблиці 3.7

3	(20.8647 12.7441 71.9994 94.0362 65.104 89.5436 79.2299 94.622 6.99 43.7829 46.7153 22.3464 67.2353 3.15818 0.810353 15.7856 30.3529 6.87129)
4	(41.9127 20.0727 62.787 91.6572 84.5207 75.1153 39.4606 63.7023 11.2718 32.9067 50.2579 23.7645 51.5455 0.833 0.849909 11.6658 39.5455 8.38418)
5	(9.82444 6.5 75.5057 96.062 28.4761 96.143 93.1539 99.9556 12.8289 68.1996 41.8933 52.7004 57.2222 5.61422 0.895111 2.90878 37.8889 7.744)
6	(3.87889 4.61667 76.4461 96.045 73.5314 96.6902 92.4543 99.7945 8.67583 72.1443 43.7033 18.297 58.9722 4.56864 0.840917 5.47539 41.5 7.88592)

Отримали наступні оцінки якості кластеризації:

WSS = 154342.47520611744

Silhouette criterion = 0.2621331800921847

3.3.2 Кластеризація агломеративним методом

Агломеративний метод відноситься до групи методів ієрархічної кластеризації, тому визначення оптимальної кількості кластерів відбувається за

допомогою дендрограми. Побудуємо дендрограму для початкового набору даних (18 змінних) (див. рис.3.4).

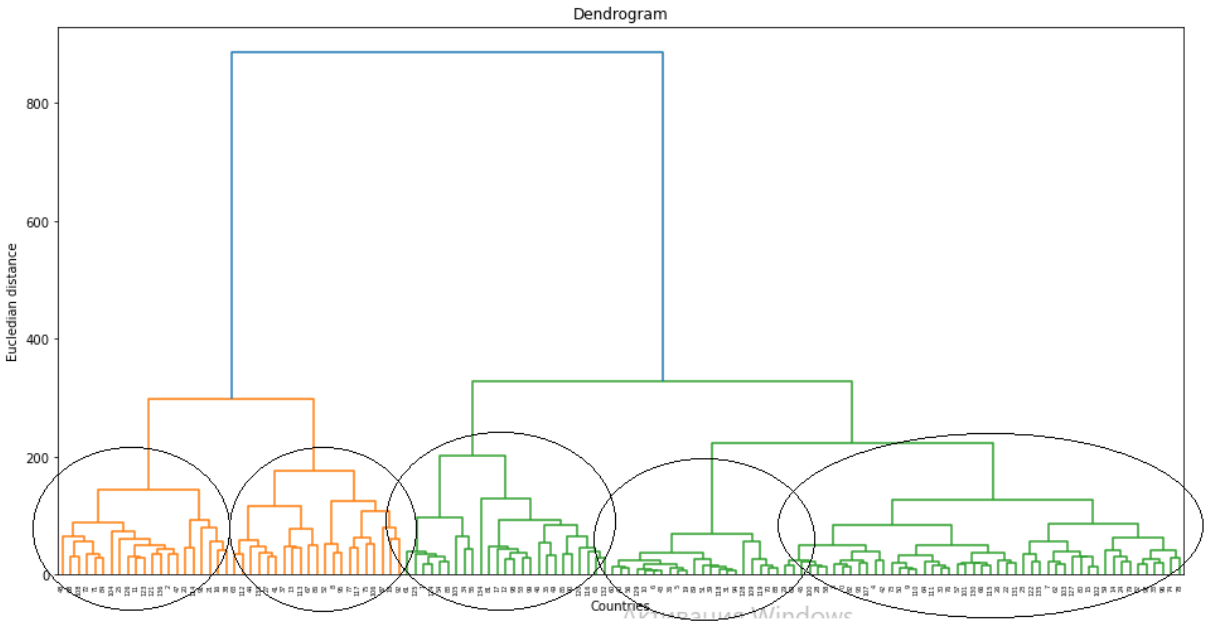

Рисунок 3.4 – Дендрограма вхідних даних початкового датасету

По аналізу дендрограми можна виділити від 5 до 7 кластерів (на рисунку 3.4 обведені у кола). Тому будуватимемо моделі саме для такої кількості кластерів.

Виконаємо кластеризацію при k=5 та одразу згрупуємо об’єкти по кластерам у вигляді таблиці (див. табл. 3.8).

Таблиця 3.8 – Кластеризація країн агломеративним методом при к=5

№ Кластеру	Об’єкти кластеру
0	Algeria Bolivia Cambodia Egypt, Arab Rep. El Salvador Guatemala Honduras Indonesia Iran, Islamic Rep. Iraq Jordan Kyrgyz Republic Mongolia Morocco Myanmar Nicaragua Peru Philippines Saudi Arabia Sri Lanka Tajikistan Tunisia Turkey Uzbekistan Venezuela
1	Bangladesh Botswana Cameroon Cote d'Ivoire Eswatini Gambia, The Ghana India Kenya Lesotho Mali Mauritania Namibia Nepal Nigeria Pakistan Senegal Solomon Islands South Africa Sudan Vanuatu
2	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria Chile China Colombia Costa Rica Croatia Cyprus Czech Republic Dominican Republic Ecuador Georgia Greece Hungary Israel Italy Jamaica Kazakhstan Korea, Rep. Latvia Lithuania Malaysia Maldives Malta Mauritius Mexico Moldova Montenegro North Macedonia Panama Paraguay Poland Portugal Romania Russian Federation Serbia Slovak Republic Slovenia Spain Thailand Ukraine United States Uruguay Vietnam
3	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Haiti Liberia Madagascar Malawi Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
4	Australia Austria Belgium Canada Denmark Estonia Finland France Germany Iceland Ireland Japan Luxembourg Netherlands New Zealand Norway Singapore Sweden Switzerland United Arab Emirates United Kingdom

Визначимо центроїди кожного кластеру (див. табл. 3.9).

Таблиця 3.9 – Координати центроїдів для агломеративного алгоритму (K=5)

Номер кластеру	Координати центроїда
0	(15.994 10.698 73.6882 95.3755 51.3865 91.8132 83.863 96.5284 8.1356 52.2727 44.1243 33.6186 63.32 3.80012 0.84484 8.88772 32.28 6.94256)
1	(42.9933 13.4 65.1082 86.7217 71.5429 78.3387 45.6608 68.2579 10.2538 31.4268 47.5449 44.1926 54.9048 1.16324 0.858476 11.0384 36.2381 6.92519)
2	(3.07551 4.20408 77.3885 96.6871 75.2636 97.4222 94.291 99.9252 8.46041 75.3348 42.3459 16.8676 59.7143 5.15049 0.847796 5.5279 46.7347 8.54288)
3	(75.611 22.6048 63.3486 83.9182 90.9628 56.6236 23.1131 34.4321 5.16286 14.2784 45.6451 39.7206 42.0476 0.280143 0.869333 6.66638 29.7143 4.99419)
4	(0.278095 2.46524 81.8156 99.5692 83.356 99.6886 98.8084 100 5.84714 91.4221 33.045 10.7756 68.1429 9.13795 0.901619 0.945857 79.0952 12.1969)

Отримані критерії якості кластеризації:

WSS = 196142.18824472974

Silhouette criterion = 0.2260561641660123

Виконаємо кластеризацію при k=6. Результати зведемо у таблицю (див. табл. 3.10).

Таблиця 3.10 – Кластеризація країн агломеративним методом при к=6

№ Кластеру	Об’єкти кластеру
0	Bangladesh Botswana Cameroon Cote d'Ivoire Eswatini Gambia, The Ghana India Kenya Lesotho Mali Mauritania Namibia Nepal Nigeria Pakistan Senegal Solomon Islands South Africa Sudan Vanuatu
1	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Haiti Liberia Madagascar Malawi Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
2	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria Chile China Colombia Costa Rica Croatia Cyprus Czech Republic Dominican Republic Ecuador Georgia Greece Hungary Israel Italy Jamaica Kazakhstan Korea, Rep. Latvia Lithuania Malaysia Maldives Malta Mauritius Mexico Moldova Montenegro North Macedonia Panama Paraguay Poland Portugal Romania Russian Federation Serbia Slovak Republic Slovenia Spain Thailand Ukraine United States Uruguay Vietnam
3	Bolivia Cambodia El Salvador Guatemala Honduras Indonesia Kyrgyz Republic Mongolia Myanmar Nicaragua Peru Philippines Sri Lanka Tajikistan Uzbekistan Venezuela
4	Australia Austria Belgium Canada Denmark Estonia Finland France Germany Iceland Ireland Japan Luxembourg Netherlands New Zealand Norway Singapore Sweden Switzerland United Arab Emirates United Kingdom
5	Algeria Egypt, Arab Rep. Iran, Islamic Rep. Iraq Jordan Morocco Saudi Arabia Tunisia Turkey

Визначимо центроїди кожного з отриманих кластерів (див. табл. 3.11).

Таблиця 3.11 – Координати центроїдів для агломеративного алгоритму (K=6)

Номер кластеру	Координати центроїда
0	(42.9933 13.4 65.1082 86.7217 71.5429 78.3387 45.6608 68.2579 10.2538 31.4268 47.5449 44.1926 54.9048 1.16324 0.858476 11.0384 36.2381 6.92519)
1	(75.611 22.6048 63.3486 83.9182 90.9628 56.6236 23.1131 34.4321 5.16286 14.2784 45.6451 39.7206 42.0476 0.280143 0.869333 6.66638 29.7143 4.99419)
2	(3.07551 4.20408 77.3885 96.6871 75.2636 97.4222 94.291 99.9252 8.46041 75.3348 42.3459 16.8676 59.7143 5.15049 0.847796 5.5279 46.7347 8.54288)
3	(19.4644 13.0594 72.6659 94.9894 64.2736 89.3777 78.6369 94.6007 5.49562 43.3138 45.3793 22.8851 66.75 2.77969 0.816562 12.2509 29.125 6.49175)
4	(0.278095 2.46524 81.8156 99.5692 83.356 99.6886 98.8084 100 5.84714 91.4221 33.045 10.7756 68.1429 9.13795 0.901619 0.945857 79.0952 12.1969)
5	(9.82444 6.5 75.5057 96.062 28.4761 96.143 93.1539 99.9556 12.8289 68.1996 41.8933 52.7004 57.2222 5.61422 0.895111 2.90878 37.8889 7.744)

Отримані значення критеріїв якості кластеризації:

WSS = 175691.5083246773; Silhouette criterion = 0.24751779190998077

Виконаємо кластеризацію при k=7. Результати зведемо у таблицю (див. табл.. 3.12)

Таблиця 3.12 – Кластеризація країн агломеративним методом при к=7

№ Кластеру	Об’єкти кластеру
0	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Haiti Liberia Madagascar Malawi Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
1	Bolivia Cambodia El Salvador Guatemala Honduras Indonesia Kyrgyz Republic Mongolia Myanmar Nicaragua Peru Philippines Sri Lanka Tajikistan Uzbekistan Venezuela
2	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria Chile China Colombia Costa Rica Croatia Cyprus Czech Republic Dominican Republic Ecuador Georgia Greece Hungary Israel Italy Jamaica Kazakhstan Korea, Rep. Latvia Lithuania Malaysia Maldives Malta Mauritius Mexico Moldova Montenegro North Macedonia Panama Paraguay Poland Portugal Romania Russian Federation Serbia Slovak Republic Slovenia Spain Thailand Ukraine United States Uruguay Vietnam
3	Bangladesh Cameroon India Mali Mauritania Nepal Nigeria Pakistan Senegal Sudan
4	Australia Austria Belgium Canada Denmark Estonia Finland France Germany Iceland Ireland Japan Luxembourg Netherlands New Zealand Norway Singapore Sweden Switzerland United Arab Emirates United Kingdom
5	Algeria Egypt, Arab Rep. Iran, Islamic Rep. Iraq Jordan Morocco Saudi Arabia Tunisia Turkey
6	Botswana Cote d'Ivoire Eswatini Gambia, The Ghana Kenya Lesotho Namibia Solomon Islands South Africa Vanuatu

Визначимо координати кожного з отриманих кластерів (див. табл. 3.13).

Таблиця 3.13 – Координати центроїдів для агломеративного алгоритму (K=7)

Номер кластеру	Координати центроїда
0	(75.611 22.6048 63.3486 83.9182 90.9628 56.6236 23.1131 34.4321 5.16286 14.2784 45.6451 39.7206 42.0476 0.280143 0.869333 6.66638 29.7143 4.99419)
1	(19.4644 13.0594 72.6659 94.9894 64.2736 89.3777 78.6369 94.6007 5.49562 43.3138 45.3793 22.8851 66.75 2.77969 0.816562 12.2509 29.125 6.49175)
2	(3.07551 4.20408 77.3885 96.6871 75.2636 97.4222 94.291 99.9252 8.46041 75.3348 42.3459 16.8676 59.7143 5.15049 0.847796 5.5279 46.7347 8.54288)
3	(45.857 10.31 67.5414 81.7334 58.1272 79.1632 48.3766 68.6738 7.71 27.4295 41.9806 65.7591 55.3 0.7383 0.8622 7.3763 30 4.5939)
4	(0.278095 2.46524 81.8156 99.5692 83.356 99.6886 98.8084 100 5.84714 91.4221 33.045 10.7756 68.1429 9.13795 0.901619 0.945857 79.0952 12.1969)
5	(9.82444 6.5 75.5057 96.062 28.4761 96.143 93.1539 99.9556 12.8289 68.1996 41.8933 52.7004 57.2222 5.61422 0.895111 2.90878 37.8889 7.744)
6	(40.39 16.2091 62.8962 91.2565 83.739 77.5891 43.1918 67.8797 12.5664 35.0606 52.6033 24.5866 54.5455 1.54955 0.855091 14.3675 41.9091 9.04455)

Отримані критерії якості кластеризації для агломеративної кластеризації при к=7:

WSS = 160163.28092867509

Silhouette criterion = 0.25134310224463385

3.3.3 Кластеризація методом DBSCAN

Перш за все, при реалізації методу DBSCAN необхідно визначити значення параметрів min_sample та eps.

Параметр min_sample рекомендовано брати як подвійне число розмірності вхідних даних. Так як всього на вході подається 18 змінnb[, то min_sample = 36.

Параметр eps обирається наступним чином. Спочатку обчислюється середня відстань між кожною точкою та її k сусідами (k = обраному значенню min_samples). Потім на графіку k-відстані наносять середні k-відстані у порядку зростання. Оптимальне значення для ε обирається у точці максимальної кривизни (тобто там, де графік має найбільший нахил). Зобразимо даний графік (див. рис. 3.5).

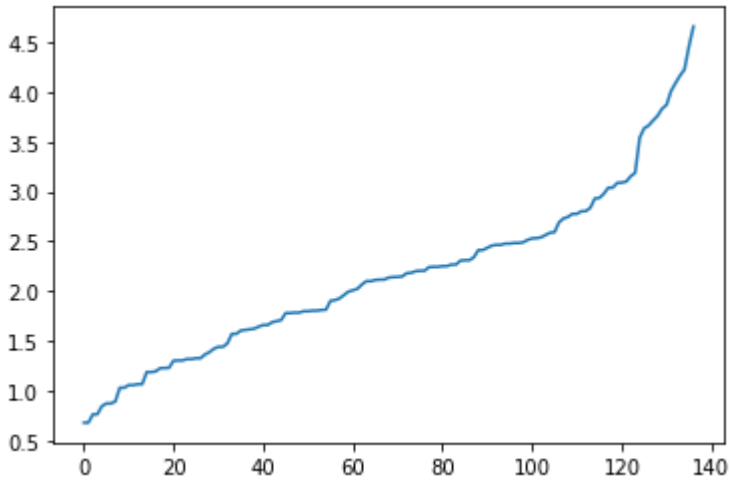

Рисунок 3.5 – Визначення оптимального значення eps

Виходячи з рисунку 3.5 та табличних значень побудованих точок визначено, що найбільшу кривизну графік має при ε = 3.19

Отримані результати кластеризації відобразимо у таблиці (табл. 3.14).

Таблиця 3.14 – Результати кластеризації методом DBSCAN

Номер кластеру	Країни
0	Albania Argentina Armenia Australia Austria Azerbaijan Belarus Belgium Bolivia Brazil Bulgaria Canada Chile China Colombia Costa Rica Croatia Cyprus Czech Republic Denmark Dominican Republic Ecuador Estonia Finland France Georgia Germany Greece Hungary Iceland Indonesia Ireland Israel Italy Japan Kazakhstan Korea, Rep. Kyrgyz Republic Latvia Lithuania Luxembourg Malaysia Maldives Malta Moldova Montenegro Morocco Netherlands New Zealand North Macedonia Norway Panama Paraguay Peru Poland Portugal Romania Russian Federation Serbia Slovak Republic Slovenia Spain Sweden Switzerland Thailand Tunisia Turkey Ukraine United Kingdom United States Uruguay Uzbekistan Vietnam
-1	Algeria Angola Bangladesh Benin Botswana Burkina Faso Cambodia Cameroon Central African Republic Chad Congo, Rep. Cote d'Ivoire Egypt, Arab Rep. El Salvador Eswatini Ethiopia Gambia, The Ghana Guatemala Guinea Haiti Honduras India Iran, Islamic Rep. Iraq Jamaica Jordan Kenya Lesotho Liberia Madagascar Malawi Mali Mauritania Mauritius Mexico Mongolia Mozambique Myanmar Namibia Nepal Nicaragua Niger Nigeria Pakistan Philippines Rwanda Saudi Arabia Senegal Sierra Leone Singapore Solomon Islands South Africa South Sudan Sri Lanka Sudan Tajikistan Tanzania Togo Uganda United Arab Emirates Vanuatu Venezuela, RB Zambia

Під значенням -1 алгоритм DBSCAN позначає об’єкти білого шуму. Отже, по суті ми отримали лише один кластер, що є абсолютно неінформативно для поставленої задачі.

Спробуємо застосувати алгоритм DBSCAN для спрощеного датасету. Спрощений датасет складається з чотирьох змінних, за якими здійснювалась кластеризація, а саме:

1) індикатор GINI – статистичний показник для оцінки економічної рівності. Його придумав економіст Джині Коррадо. Він показує рівномірність розподілу доходу або багатства між членами суспільства.. Значення індексу 0 відповідає абсолютній рівності, 1 – абсолютній нерівності;

2) індекс стану здоров'я населення;

3) показник рівня життя;

4) індекс сталого розвитку.

Відобразимо графік, за допомогою якого визначимо оптимальне значення параметру eps (див. рис. 3.6).

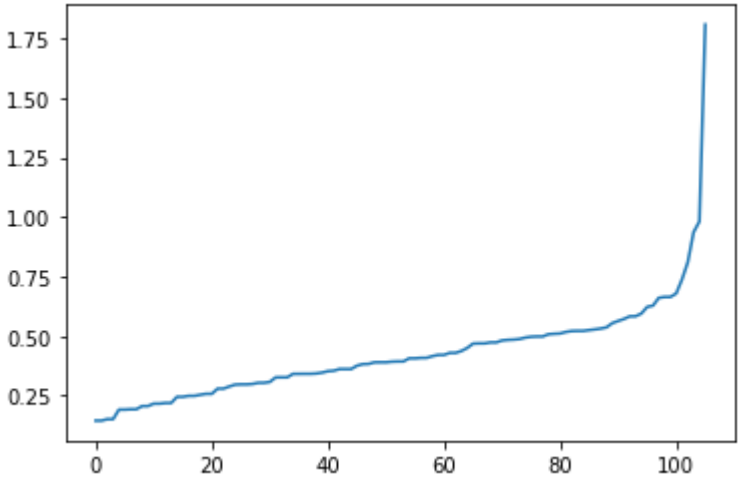

Рисунок 3.6 – Обчислення оптимального eps

Як можемо бачити, оптимальне значення параметру eps приблизно дорівнює 0.8

Результати кластеризації представимо у вигляді таблиці (див. табл. 3.15).

Країни, що потрапили до кластеру -1, алгоритм визначає як шумові точки.

Таблиця 3.15 – Результати кластеризації методом DBSCAN

Номер кластеру	Країни
0	Algeria, Armenia, Azerbaijan, Bangladesh, Benin, Bolivia, Bosnia and Herzegovina, Botswana, China, Dominican Republic, Egypt, El Salvador, Georgia, Guatemala, Honduras, India, Indonesia, Jamaica, Yordaniya, Kazakhstan, Kyrgyzstan, Malawi, Malaysia, Moldova..Republic, Mongolia, Morocco, Pakistan, Paraguay, Peru, Filippini, Russian Federation, Senegal, South Africa, Sri Lanka, Tajikistan, Tanzania., Thailand, Trinidad, Tunisia, Turkey, Uzbekistan, Venezuela, Vjetnam
1	Albania, Australia, Austria, Belgium, Canada, Croatia, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Ireland, Israel, Italy, Japan, Korea Republic, Latvia, Lithuania, Niderlandi, New Zeland, Norway, Poland, Portugal, Rumuniya, Slovachchina, Slovenia, Spain, Sweden, Switzerland, Great Britain, USA, Uruguay
2	Cambodia, Cameroon, Gambia, Kenya, Madagascar, Mozambique, Nepal, Niger, Uganda, Zambia
-1	Argentina, Brazil, Bulgaria, Chile, Colombia, Costa Rica, Cyprus, Ecuador, Ethiopia, Iceland, Luxembourg, Mexico, Namibia, Nicaragua, Panama, Ukraine, United by Arabskiye Emirati, Zimbabwe

Як можемо бачити, все одно чимало країн були позначені як шум, а нас цікавило розподілити всі країни у кластери. Тому при заданій постановці задачі на даному наборі даний метод dbscan не показує хороших результатів.

3.3.4 Кластеризація нечітким методом Бездека (нечіткий метод k-середніх)

Задаємо параметр нечіткості = 2, кількість ітерацій = 10000. Для адекватності порівняння будуватимемо кластеризацію для кількості кластерів 5-7, як і в попередніх методах.

На відміну від звичайних методів кластеризації, нечіткий метод на виході дає ступені належності кожного об'єкту до всіх кластерів. У таблиці нижче $w_0, w_1, w_2, \ldots$ - міри належності до відповідного кластеру. Результат (а саме до якого кластеру слід віднести об'єкт) визначається максимальним значенням w_i.

Виконаємо спочатку кластеризацію при k=5. Результатом є матриця належності (див. табл. 3.16)

Таблиця 3.16 – Матриця належності до кластерів при К=5

Країна	Обраний кластер	w_0	w_1	w_2	w_3	w_4
Albania	0	0.696609	0.013336	0.026308	0.152267	0.111481
Algeria	3	0.322061	0.047756	0.095178	0.387717	0.147288
Angola	1	0.036334	0.656323	0.23221	0.048558	0.026574
Argentina	0	0.789298	0.007376	0.014342	0.090508	0.098476
Armenia	0	0.583831	0.016237	0.033878	0.23937	0.126684
Australia	4	0.061215	0.005297	0.008885	0.026021	0.898582
Austria	4	0.032726	0.002689	0.004555	0.01408	0.94595
Azerbaijan	0	0.535861	0.022819	0.042978	0.189136	0.209206
Bangladesh	2	0.150554	0.138564	0.319617	0.30108	0.090185
Belarus	0	0.527651	0.011795	0.021843	0.099769	0.338942
Belgium	4	0.036997	0.003093	0.005222	0.015513	0.939175
Benin	1	0.025349	0.724551	0.196506	0.034067	0.019527
Bolivia	3	0.249226	0.046588	0.12488	0.4869	0.092407
Botswana	3	0.215474	0.107424	0.24181	0.289738	0.145554
Brazil	0	0.474636	0.031399	0.062021	0.270035	0.161909
Bulgaria	0	0.705444	0.011166	0.022933	0.137531	0.122926
Burkina Faso	1	0.038442	0.677015	0.203174	0.050664	0.030705
Cambodia	3	0.21097	0.084741	0.223413	0.379692	0.101184
Cameroon	2	0.081058	0.282719	0.464387	0.117066	0.05477

Продовження таблиці 3.16

Canada	4	0.063831	0.006056	0.010017	0.028092	0.892005
Central African Republic	1	0.054145	0.592696	0.240913	0.070723	0.041523
Chad	1	0.059093	0.573744	0.244233	0.076443	0.046487
Chile	4	0.314993	0.01305	0.023709	0.099584	0.548664
China	3	0.336111	0.035973	0.075654	0.391231	0.161032
Colombia	0	0.471816	0.021072	0.043188	0.336747	0.127178
Congo, Rep.	1	0.053065	0.532605	0.301037	0.074373	0.03892
Costa Rica	0	0.41244	0.01712	0.031505	0.158767	0.380168
Cote d'Ivoire	2	0.041111	0.117733	0.747766	0.067574	0.025815
Croatia	0	0.612965	0.011417	0.021394	0.100122	0.254103
Cyprus	4	0.329094	0.015823	0.027998	0.096081	0.531004
Czech Republic	4	0.304214	0.011143	0.020145	0.090032	0.574466
Denmark	4	0.109774	0.014221	0.0228	0.055867	0.797339
Dominican Republic	0	0.579718	0.01671	0.033711	0.259235	0.110626
Ecuador	0	0.489814	0.012256	0.026406	0.393858	0.077667
Egypt, Arab Rep.	3	0.2508	0.087656	0.159423	0.349082	0.153039
El Salvador	3	0.258482	0.059711	0.119759	0.42526	0.136788
Estonia	4	0.032103	0.002511	0.004249	0.013053	0.948084
Eswatini	2	0.144912	0.162632	0.384579	0.219664	0.088213
Ethiopia	1	0.038855	0.610581	0.268077	0.052481	0.030006
Finland	4	0.093743	0.010854	0.017685	0.045193	0.832527
France	4	0.051095	0.003109	0.005405	0.017976	0.922415
Gambia, The	2	0.046627	0.124754	0.723643	0.075817	0.02916
Georgia	0	0.599352	0.012989	0.026649	0.178127	0.182883
Germany	4	0.040152	0.003593	0.006048	0.01771	0.932497
Ghana	2	0.118918	0.170387	0.46284	0.167886	0.07997
Greece	0	0.529324	0.020433	0.037704	0.151251	0.261288
Guatemala	3	0.234799	0.044333	0.102393	0.524239	0.094237
Guinea	1	0.032012	0.673596	0.228544	0.041838	0.02401
Haiti	1	0.07288	0.419482	0.359781	0.096152	0.051705
Honduras	3	0.20478	0.071363	0.159921	0.466872	0.097064
Hungary	0	0.644534	0.009491	0.017811	0.099035	0.22913
Iceland	4	0.090465	0.009449	0.015344	0.040212	0.84453
India	3	0.186739	0.129271	0.251197	0.311949	0.120844
Indonesia	3	0.214234	0.027923	0.066695	0.612609	0.078538
Iran, Islamic Rep.	3	0.325471	0.042025	0.082193	0.410577	0.139735
Iraq	3	0.248585	0.07589	0.145623	0.400406	0.129495
Ireland	4	0.077533	0.005245	0.009176	0.029135	0.878911
Israel	4	0.199266	0.009237	0.016272	0.058609	0.716617
Italy	0	0.583684	0.012086	0.022757	0.113677	0.267795
Jamaica	3	0.334594	0.046243	0.092804	0.351371	0.174988
Japan	4	0.067666	0.00447	0.007737	0.026068	0.894058

Продовження таблиці 3.16

Jordan	3	0.337081	0.041713	0.079799	0.359512	0.181894
Kazakhstan	0	0.561837	0.016602	0.030612	0.140244	0.250706
Kenya	2	0.065229	0.245859	0.546302	0.098366	0.044245
Korea, Rep.	4	0.20328	0.01228	0.021381	0.077356	0.685703
Kyrgyz Republic	3	0.259858	0.02984	0.063083	0.552246	0.094974
Latvia	4	0.269146	0.008866	0.016193	0.066607	0.639189
Lesotho	2	0.09153	0.297659	0.414944	0.130544	0.065324
Liberia	1	0.051056	0.618231	0.226442	0.064444	0.039827
Lithuania	4	0.372556	0.01732	0.031336	0.108025	0.470763
Luxembourg	4	0.114402	0.012952	0.02115	0.057575	0.793921
Madagascar	1	0.043385	0.682347	0.183784	0.055824	0.03466
Malawi	1	0.037532	0.708621	0.175774	0.048448	0.029626
Malaysia	0	0.539728	0.011237	0.020774	0.108018	0.320244
Maldives	0	0.470202	0.024216	0.045907	0.287923	0.171753
Mali	2	0.05528	0.398308	0.426737	0.080817	0.038858
Malta	4	0.329882	0.011282	0.020246	0.088582	0.550009
Mauritania	2	0.100823	0.199515	0.473808	0.162433	0.063421
Mauritius	0	0.509231	0.016271	0.031854	0.237487	0.205158
Mexico	0	0.44153	0.023884	0.048034	0.36523	0.121322
Moldova	0	0.51034	0.025203	0.051002	0.226216	0.187238
Mongolia	3	0.272658	0.062353	0.15509	0.386621	0.123276
Montenegro	0	0.703928	0.0112	0.021698	0.116004	0.147171
Morocco	3	0.370883	0.034447	0.069011	0.373795	0.151863
Mozambique	1	0.038125	0.693862	0.187845	0.050596	0.029572
Myanmar	3	0.176236	0.088807	0.242675	0.402223	0.090059
Namibia	2	0.120938	0.201987	0.43538	0.155603	0.086093
Nepal	2	0.18526	0.151178	0.278959	0.257251	0.127352
Netherlands	4	0.100485	0.011488	0.01886	0.051207	0.81796
New Zealand	4	0.141373	0.017054	0.02762	0.068311	0.745641
Nicaragua	3	0.190297	0.060981	0.143298	0.518913	0.086512
Niger	1	0.092005	0.413576	0.300764	0.119676	0.073978
Nigeria	2	0.094235	0.314991	0.391181	0.132361	0.067231
North Macedonia	0	0.661449	0.014566	0.028496	0.172181	0.123308
Norway	4	0.104921	0.013362	0.021379	0.052467	0.807871
Pakistan	2	0.154035	0.134549	0.318297	0.304338	0.088782
Panama	0	0.478182	0.018667	0.03885	0.349221	0.115079
Paraguay	0	0.620567	0.015924	0.032225	0.233948	0.097336
Peru	0	0.443032	0.023887	0.05589	0.372035	0.105156
Philippines	3	0.206432	0.033748	0.08759	0.604239	0.067991
Poland	0	0.441921	0.012341	0.022633	0.097132	0.425973
Portugal	4	0.277337	0.011692	0.020864	0.079325	0.610783
Romania	0	0.730985	0.009566	0.019478	0.129417	0.110555
Russian Federation	0	0.612709	0.016285	0.030851	0.16714	0.173014

Продовження таблиці 3.16

Rwanda	1	0.079216	0.435317	0.319723	0.102462	0.063282
Saudi Arabia	3	0.282053	0.077299	0.129255	0.285993	0.225401
Senegal	2	0.092511	0.14301	0.554488	0.152608	0.057383
Serbia	0	0.636276	0.01571	0.030247	0.150482	0.167285
Sierra Leone	1	0.03571	0.719523	0.171239	0.045109	0.028419
Singapore	4	0.201576	0.026883	0.044174	0.122915	0.604451
Slovak Republic	0	0.507601	0.010218	0.018939	0.090733	0.372509
Slovenia	4	0.151851	0.007667	0.013535	0.049818	0.777129
Solomon Islands	2	0.071504	0.339574	0.434716	0.10143	0.052776
South Africa	3	0.256599	0.078966	0.169752	0.352788	0.141895
South Sudan	1	0.074933	0.522759	0.25077	0.092081	0.059458
Spain	4	0.128782	0.006891	0.011921	0.041537	0.810868
Sri Lanka	3	0.283619	0.038697	0.077365	0.479048	0.12127
Sudan	2	0.087216	0.275381	0.449582	0.131079	0.056743
Sweden	4	0.106162	0.013447	0.021588	0.054028	0.804774
Switzerland	4	0.147011	0.019306	0.031125	0.081755	0.720803
Tajikistan	3	0.228788	0.060373	0.125064	0.475276	0.110499
Tanzania	1	0.028779	0.706395	0.203277	0.03922	0.02233
Thailand	0	0.551188	0.016936	0.032792	0.231338	0.167745
Togo	1	0.02476	0.72904	0.194145	0.033776	0.018278
Tunisia	0	0.382531	0.035437	0.07135	0.354613	0.156068
Turkey	0	0.44729	0.026747	0.053016	0.318725	0.154222
Uganda	1	0.02913	0.686326	0.223359	0.039498	0.021687
Ukraine	0	0.580628	0.017218	0.034871	0.24587	0.121413
United Arab Emirates	4	0.284509	0.034099	0.058431	0.178671	0.44429
United Kingdom	4	0.045437	0.004064	0.006787	0.019808	0.923904
United States	4	0.125798	0.007893	0.013568	0.045937	0.806804
Uruguay	4	0.215571	0.011681	0.021027	0.082059	0.669661
Uzbekistan	3	0.306958	0.030335	0.060959	0.488057	0.113692
Vanuatu	2	0.099047	0.185296	0.497156	0.151317	0.067184
Venezuela, RB	3	0.238725	0.132263	0.200702	0.273669	0.15464
Vietnam	0	0.498285	0.022786	0.04652	0.258578	0.173831
Zambia	1	0.025002	0.747572	0.174754	0.033925	0.018746

Згрупуємо отримані результати по кластерам (див. табл. 3.17).

Таблиця 3.17 – Результати кластеризації методом Бездека (К=5)

Номер кластеру	Країни
0	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria Colombia Costa Rica Croatia Dominican Republic Ecuador Georgia Greece Hungary Italy Kazakhstan Malaysia Maldives Mauritius Mexico Moldova Montenegro North Macedonia Panama Paraguay Peru Poland Romania Russian Federation Serbia Slovak Republic Thailand Tunisia Turkey Ukraine Vietnam
1	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Haiti Liberia Madagascar Malawi Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
2	Bangladesh Cameroon Cote d'Ivoire Eswatini Gambia, The Ghana Kenya Lesotho Mali Mauritania Namibia Nepal Nigeria Pakistan Senegal Solomon Islands Sudan Vanuatu
3	Algeria Bolivia Botswana Cambodia China Egypt, Arab Rep. El Salvador Guatemala Honduras India Indonesia Iran, Islamic Rep. Iraq Jamaica Jordan Kyrgyz Republic Mongolia Morocco Myanmar Nicaragua Philippines Saudi Arabia South Africa Sri Lanka Tajikistan Uzbekistan Venezuela
4	Australia Austria Belgium Canada Chile Cyprus Czech Republic Denmark Estonia Finland France Germany Iceland Ireland Israel Japan Korea, Rep. Latvia Lithuania Luxembourg Malta Netherlands New Zealand Norway Portugal Singapore Slovenia Spain Sweden Switzerland United Arab Emirates United Kingdom United States Uruguay

Визначимо координати центроїді отриманих кластерів (див. табл. 3.18).

Таблиця 3.18 – Координати центроїдів, метод Бездека (К=5)

Номер кластеру	Координати центроїда
0	(5.7822 5.1122 76.0029 95.8378 70.6146 96.0597 91.5118 98.883 8.99305 70.7594 42.9778 20.7114 58.3227 4.67677 0.857653 5.30998 42.125 7.95583)
1	(72.7742 20.5772 63.4285 85.6809 89.4696 59.248 24.8291 38.0443 5.46695 16.0645 46.1791 37.6134 43.3912 0.382774 0.866177 7.11166 30.5017 5.44305)
2	(47.4253 15.9653 65.234 85.9963 75.5775 72.7254 40.2491 61.3624 7.82558 28.6389 45.3294 39.6224 50.1172 0.861428 0.872696 9.19992 34.0614 5.89723)
3	(16.3543 9.82971 73.3259 94.7339 60.4578 91.4756 81.0274 94.3558 8.0766 51.4318 45.4248 28.9198 64.8871 3.22079 0.819699 9.34753 34.8143 7.0786)
4	(1.11424 2.88381 80.9651 99.077 81.8118 99.2646 97.9556 99.7061 6.70278 87.4231 35.3687 11.9636 64.8611 8.1339 0.890407 1.95405 70.9872 11.6128)

Визначимо критерії якості кластеризації:

WSS = 197141.49530813273

Silhouette criterion = 0.23179375687414597

Виконаємо кластеризацію нечітким методом при К=6. Результатом є матриця належності (див. табл. 3.19).

Таблиця 3.19 – Матриця належності до кластерів при К=6

Country	Кластер	w_0	w_1	w_2	w_3	w_4	w_5
Albania	0	0.591428	0.012074	0.013555	0.072453	0.083074	0.227415
Algeria	5	0.239954	0.041831	0.047134	0.236519	0.112706	0.321855
Angola	1	0.024047	0.466724	0.417439	0.044275	0.018019	0.029497
Argentina	0	0.743639	0.006092	0.006805	0.036836	0.065328	0.1413
Armenia	0	0.422934	0.014509	0.016439	0.106852	0.093648	0.345618
Australia	4	0.058753	0.004346	0.004745	0.014608	0.889853	0.027695
Austria	4	0.023706	0.001672	0.001829	0.005905	0.955517	0.011371
Azerbaijan	0	0.467963	0.020111	0.02245	0.100676	0.154818	0.233982
Bangladesh	3	0.111057	0.123797	0.14493	0.370957	0.070661	0.178597
Belarus	0	0.62406	0.008876	0.00986	0.043161	0.196715	0.117327
Belgium	4	0.029314	0.002072	0.002266	0.007083	0.945756	0.013508
Benin	1	0.0169	0.497739	0.41942	0.031803	0.013355	0.020783
Bolivia	3	0.154641	0.037807	0.044311	0.414337	0.063542	0.285362
Botswana	3	0.166635	0.098199	0.112881	0.28532	0.116171	0.220793
Brazil	0	0.356748	0.027679	0.031066	0.144678	0.121656	0.318173
Bulgaria	0	0.613425	0.010076	0.011379	0.065908	0.089905	0.209307
Burkina Faso	1	0.029175	0.477268	0.382818	0.051605	0.02385	0.035284
Cambodia	3	0.142329	0.071425	0.084384	0.404791	0.073002	0.224069
Cameroon	2	0.069514	0.28805	0.354324	0.149573	0.04842	0.090119
Canada	4	0.054878	0.004496	0.004898	0.014475	0.894482	0.026771
Central African Republic	1	0.040447	0.42891	0.380501	0.069438	0.03176	0.048944
Chad	1	0.045163	0.42052	0.367453	0.076488	0.036323	0.054053
Chile	4	0.359316	0.012535	0.013866	0.056786	0.415862	0.141635
China	5	0.244944	0.031476	0.035857	0.234627	0.121289	0.331807
Colombia	5	0.299357	0.017903	0.020183	0.135533	0.090946	0.436078
Congo, Rep.	1	0.037705	0.408326	0.403441	0.074066	0.028545	0.047916
Costa Rica	0	0.383397	0.015605	0.017285	0.08008	0.28607	0.217562
Cote d'Ivoire	2	0.067561	0.253396	0.344638	0.193495	0.044408	0.096501
Croatia	0	0.698041	0.007819	0.008697	0.038871	0.136281	0.110291
Cyprus	0	0.396027	0.014461	0.015947	0.056333	0.393293	0.123939
Czech Republic	4	0.373379	0.01082	0.01196	0.051166	0.425341	0.127334
Denmark	4	0.09803	0.011337	0.012296	0.032521	0.791213	0.054604
Dominican Republic	5	0.38278	0.01506	0.016926	0.109652	0.083375	0.392206
Ecuador	5	0.238703	0.009358	0.010611	0.101409	0.049148	0.590771
Egypt, Arab Rep.	3	0.189549	0.076065	0.085267	0.273389	0.11901	0.256721
El Salvador	5	0.185583	0.052163	0.058955	0.294635	0.105304	0.303359
Estonia	4	0.026579	0.001756	0.00192	0.006125	0.951588	0.012032
Eswatini	3	0.120959	0.16601	0.196355	0.268872	0.077347	0.170457
Ethiopia	1	0.026544	0.444196	0.425587	0.049936	0.021045	0.032694
Finland	4	0.082837	0.00845	0.009186	0.025295	0.830487	0.043746
France	4	0.08095	0.003961	0.004351	0.014944	0.864561	0.031233
Gambia, The	2	0.070845	0.245888	0.335982	0.201518	0.046242	0.099525

Продовження таблиці 3.19

Georgia	0	0.492282	0.011603	0.013075	0.080786	0.131064	0.271191
Germany	4	0.027485	0.00212	0.002317	0.007176	0.947461	0.01344
Ghana	3	0.112018	0.192268	0.233649	0.239058	0.077774	0.145233
Greece	0	0.515559	0.017232	0.019122	0.077037	0.183356	0.187693
Guatemala	3	0.156725	0.038464	0.04421	0.353636	0.070386	0.336579
Guinea	1	0.020893	0.472194	0.427967	0.037684	0.01604	0.025222
Haiti	2	0.056136	0.364093	0.368911	0.101327	0.040807	0.068726
Honduras	3	0.136866	0.060279	0.069327	0.399048	0.07123	0.26325
Hungary	0	0.703824	0.00673	0.007484	0.036855	0.126192	0.118915
Iceland	4	0.085619	0.007687	0.008353	0.023188	0.833561	0.041591
India	3	0.141003	0.113981	0.130124	0.313357	0.095299	0.206236
Indonesia	3	0.146531	0.02555	0.029424	0.371541	0.061279	0.365675
Iran, Islamic Rep.	5	0.237756	0.036888	0.041398	0.22839	0.107022	0.348547
Iraq	3	0.1819	0.065881	0.074179	0.301481	0.099578	0.276981
Ireland	4	0.102616	0.005734	0.006302	0.021199	0.821445	0.042703
Israel	4	0.278213	0.009877	0.010883	0.039563	0.571788	0.089676
Italy	0	0.629695	0.009266	0.010309	0.047936	0.161934	0.14086
Jamaica	5	0.245099	0.040557	0.045781	0.215182	0.133618	0.319763
Japan	4	0.089792	0.004929	0.005408	0.01866	0.842501	0.03871
Jordan	5	0.25576	0.036594	0.040921	0.200682	0.140182	0.325862
Kazakhstan	0	0.548356	0.014019	0.015572	0.068143	0.173596	0.180314
Kenya	2	0.062139	0.29261	0.371447	0.14668	0.043864	0.08326
Korea, Rep.	4	0.24226	0.012351	0.013588	0.049462	0.576407	0.105932
Kyrgyz Republic	5	0.189818	0.02802	0.031774	0.286624	0.076755	0.38701
Latvia	4	0.374763	0.009065	0.010036	0.040841	0.462547	0.102748
Lesotho	2	0.076305	0.290551	0.322675	0.154002	0.056463	0.100003
Liberia	1	0.039496	0.452221	0.365449	0.064921	0.031415	0.046497
Lithuania	0	0.422133	0.01554	0.017185	0.062049	0.344411	0.138682
Luxembourg	4	0.107143	0.010933	0.011888	0.034391	0.775257	0.060388
Madagascar	1	0.035645	0.481626	0.351364	0.059658	0.029136	0.04257
Malawi	1	0.030761	0.500048	0.3553	0.052254	0.024817	0.03682
Malaysia	0	0.594519	0.008982	0.009958	0.046392	0.198455	0.141694
Maldives	0	0.349434	0.021184	0.023528	0.128491	0.128313	0.34905
Mali	2	0.041199	0.358376	0.428213	0.088366	0.029962	0.053884
Malta	0	0.412307	0.010586	0.011686	0.048447	0.393298	0.123677
Mauritania	2	0.09162	0.220081	0.267606	0.232209	0.060021	0.128464
Mauritius	0	0.390111	0.014366	0.016044	0.099669	0.149947	0.329863
Mexico	5	0.286432	0.020363	0.022901	0.149389	0.087848	0.433067
Moldova	0	0.417117	0.022453	0.025326	0.12607	0.140189	0.268845
Mongolia	3	0.190151	0.05325	0.062188	0.334688	0.090225	0.269498
Montenegro	0	0.688935	0.008714	0.009746	0.048591	0.092357	0.151657
Morocco	5	0.272658	0.030078	0.033859	0.192339	0.114449	0.356616
Mozambique	1	0.029973	0.493613	0.362595	0.053392	0.023854	0.036574

Продовження таблиці 3.19

Myanmar	3	0.110445	0.069697	0.082251	0.481968	0.060687	0.194952
Namibia	2	0.110814	0.218499	0.254801	0.197856	0.080907	0.137123
Nepal	3	0.14675	0.137083	0.157488	0.267989	0.103424	0.187265
Netherlands	4	0.090033	0.009263	0.010083	0.029351	0.810348	0.050922
New Zealand	4	0.140059	0.014987	0.016272	0.043428	0.710386	0.074867
Nicaragua	3	0.119747	0.048946	0.056368	0.470834	0.060096	0.24401
Niger	1	0.071815	0.332361	0.329941	0.120586	0.059043	0.086254
Nigeria	2	0.07571	0.290219	0.333	0.147758	0.055745	0.097568
North Macedonia	0	0.54457	0.013089	0.014686	0.079712	0.091556	0.256387
Norway	4	0.09288	0.010506	0.011388	0.030088	0.804383	0.050754
Pakistan	3	0.114487	0.12138	0.142241	0.368024	0.069923	0.183946
Panama	5	0.293299	0.01594	0.018002	0.133661	0.082536	0.456563
Paraguay	0	0.420824	0.014868	0.016727	0.104619	0.075874	0.367088
Peru	5	0.281654	0.020635	0.023756	0.193088	0.075649	0.405218
Philippines	3	0.130103	0.028684	0.033408	0.444667	0.048881	0.314257
Poland	0	0.534881	0.010254	0.011368	0.04727	0.271609	0.124619
Portugal	4	0.351363	0.011472	0.012645	0.04804	0.464033	0.112446
Romania	0	0.613412	0.008956	0.010084	0.059989	0.083749	0.22381
Russian Federation	0	0.528683	0.014409	0.016056	0.080116	0.126691	0.234045
Rwanda	1	0.060716	0.358792	0.354495	0.103978	0.04943	0.072589
Saudi Arabia	5	0.228045	0.067469	0.074357	0.205623	0.179632	0.244875
Senegal	3	0.096998	0.188717	0.24276	0.268521	0.062846	0.140158
Serbia	0	0.595562	0.013059	0.014623	0.070443	0.114216	0.192097
Sierra Leone	1	0.029468	0.50522	0.358048	0.048756	0.023902	0.034606
Singapore	4	0.190184	0.02433	0.026497	0.07965	0.54742	0.131919
Slovak Republic	0	0.625453	0.00771	0.00856	0.038458	0.21036	0.109459
Slovenia	4	0.215981	0.0086	0.009476	0.03532	0.653168	0.077455
Solomon Islands	2	0.059118	0.330366	0.368383	0.121509	0.04506	0.075565
South Africa	3	0.186127	0.06942	0.079539	0.284013	0.108872	0.272029
South Sudan	1	0.058043	0.38984	0.34867	0.089177	0.046926	0.067344
Spain	4	0.183316	0.00781	0.008572	0.029904	0.7041	0.066298
Sri Lanka	5	0.205319	0.034882	0.039092	0.26027	0.0953	0.365137
Sudan	2	0.074182	0.278824	0.333807	0.163402	0.05006	0.099726
Sweden	4	0.093433	0.010612	0.011507	0.030957	0.801121	0.052369
Switzerland	4	0.137698	0.01675	0.018175	0.051178	0.690565	0.085635
Tajikistan	3	0.160269	0.051835	0.05901	0.367263	0.083123	0.278501
Tanzania	1	0.019449	0.499627	0.404216	0.037057	0.015507	0.024143
Thailand	0	0.43048	0.015147	0.016939	0.102968	0.124656	0.30981
Togo	1	0.016263	0.50003	0.419866	0.031268	0.012337	0.020235
Tunisia	5	0.284361	0.030899	0.034837	0.189475	0.117327	0.343101
Turkey	5	0.328374	0.02324	0.026157	0.151187	0.114137	0.356906
Uganda	1	0.01855	0.465937	0.442989	0.035387	0.01417	0.022967
Ukraine	0	0.436582	0.015672	0.017673	0.112408	0.092185	0.325481

Продовження таблиці 3.19

United Arab Emirates	4	0.263654	0.030769	0.033821	0.112792	0.369223	0.18974
United Kingdom	4	0.034489	0.002663	0.002906	0.008902	0.934256	0.016784
United States	4	0.159747	0.008401	0.009206	0.031516	0.723955	0.067175
Uruguay	4	0.248032	0.01188	0.01311	0.051398	0.558746	0.116835
Uzbekistan	5	0.217778	0.027406	0.030865	0.235057	0.088499	0.400395
Vanuatu	2	0.095755	0.217682	0.259399	0.229872	0.067487	0.129805
Venezuela, RB	3	0.188442	0.11639	0.126368	0.222701	0.123923	0.222175
Vietnam	0	0.387976	0.020289	0.02292	0.135884	0.129445	0.303486
Zambia	1	0.017551	0.519392	0.394618	0.033037	0.013542	0.021861

Згрупуємо отримані результати по кластерам (див. табл. 3.20).

Таблиця 3.20 – Результати кластеризації методом Бездека (К=6)

№ кластеру	Країни
0	Albania Argentina Armenia Azerbaijan Belarus Brazil Bulgaria Costa Rica Croatia Cyprus Georgia Greece Hungary Italy Kazakhstan Lithuania Malaysia Maldives Malta Mauritius Moldova Montenegro North Macedonia Paraguay Poland Romania Russian Federation Serbia Slovak Republic Thailand Ukraine Vietnam
1	Angola Benin Burkina Faso Central African Republic Chad Congo, Rep. Ethiopia Guinea Liberia Madagascar Malawi Mozambique Niger Rwanda Sierra Leone South Sudan Tanzania Togo Uganda Zambia
2	Cameroon Cote d'Ivoire Gambia, The Haiti Kenya Lesotho Mali Mauritania Namibia Nigeria Solomon Islands Sudan Vanuatu
3	Bangladesh Bolivia Botswana Cambodia Egypt, Arab Rep. Eswatini Ghana Guatemala Honduras India Indonesia Iraq Mongolia Myanmar Nepal Nicaragua Pakistan Philippines Senegal South Africa Tajikistan
4	Australia Austria Belgium Canada Chile Czech Republic Denmark Estonia Finland France Germany Iceland Ireland Israel Japan Korea, Rep. Latvia Luxembourg Netherlands New Zealand Norway Portugal Singapore Slovenia Spain Sweden Switzerland United Arab Emirates United Kingdom United States Uruguay

Продовження таблиці 3.20

5	Algeria China Colombia Dominican Republic Ecuador El Salvador Iran, Islamic Rep. Jamaica Jordan Kyrgyz Republic Mexico Morocco Panama Peru Saudi Arabia Sri Lanka Tunisia Turkey Uzbekistan

Визначимо координати центроїді отриманих кластерів (див. табл. 3.21).

Таблиця 3.21 – Координати центроїдів, метод Бездека (К=6)

Номер кластеру	Координати центроїда
0	(4.42152 4.43602 76.495 96.3696 72.4914 96.6615 93.2815 99.1443 8.64336 73.9605 41.5262 19.5396 57.3881 5.18445 0.872375 4.17321 44.7261 8.26555)
1	(69.7222 20.2743 63.5782 85.2643 87.7788 60.8121 26.8376 40.9448 5.8912 17.561 46.0716 38.0978 44.1358 0.424114 0.867528 7.57343 30.8568 5.55881)
2	(64.9199 18.9336 63.8989 84.2975 85.2525 62.8081 29.653 45.9817 6.39575 19.8748 45.6916 40.0078 45.2575 0.494395 0.870593 8.39244 31.2495 5.55385)
3	(22.7579 12.2387 71.0469 93.231 61.8416 87.9481 70.9617 87.3189 7.62871 43.0326 45.1288 34.0064 64.108 2.67679 0.827355 9.44976 33.2476 6.57352)
4	(0.900816 2.75982 81.2725 99.2052 82.7358 99.4163 98.2221 99.7595 6.54821 88.5126 34.7143 11.0974 65.4589 8.41769 0.894747 1.72558 73.4193 11.989)
5	(11.9466 7.71243 74.8638 95.1827 63.4017 93.7379 85.614 96.6933 8.87057 59.868 45.6815 25.4397 62.8 3.59493 0.816769 8.73419 37.7916 7.5137)

Визначимо критерії якості кластеризації:

WSS = 203914.71807186952; Silhouette criterion = 0.1892074353648808

Виконаємо кластеризацію нечітким методом Бездека при К=7. Результатом є матриця належності (див. табл. 3.22).

Таблиця 3.22 – Матриця належності до кластерів при К=7

Country	w_0	w_1	w_2	w_3	w_4	w_5	w_6
Albania	0.487845	0.011122	0.024771	0.115367	0.073454	0.276318	0.011122
Algeria	0.199987	0.0364	0.08168	0.279722	0.096759	0.269051	0.0364
Angola	0.019966	0.403925	0.103599	0.02919	0.015285	0.02411	0.403925
Argentina	0.654585	0.005811	0.012828	0.064601	0.059071	0.197294	0.005811
Armenia	0.33173	0.012777	0.030492	0.171043	0.079465	0.361715	0.012777
Australia	0.062213	0.003878	0.007128	0.017886	0.876146	0.028872	0.003878
Austria	0.022166	0.001315	0.002447	0.006467	0.955847	0.010442	0.001315
Azerbaijan	0.410194	0.017538	0.03677	0.134004	0.130323	0.253633	0.017538
Bangladesh	0.092643	0.107291	0.267866	0.222888	0.061267	0.140754	0.107291
Belarus	0.67616	0.006515	0.013479	0.050747	0.133778	0.112806	0.006515
Belgium	0.028422	0.001674	0.003098	0.007881	0.944511	0.01274	0.001674
Benin	0.013983	0.425347	0.086701	0.020471	0.011279	0.016872	0.425347
Bolivia	0.136795	0.036267	0.121385	0.352906	0.060619	0.25576	0.036267
Botswana	0.137958	0.08291	0.206155	0.208655	0.098862	0.182549	0.08291
Brazil	0.290253	0.02399	0.05352	0.188453	0.103033	0.316762	0.02399
Bulgaria	0.512328	0.009307	0.02191	0.103847	0.079243	0.264057	0.009307
Burkina Faso	0.022737	0.39868	0.101703	0.032345	0.018925	0.026929	0.39868
Cambodia	0.12448	0.065718	0.204763	0.276945	0.067223	0.195153	0.065718
Cameroon	0.053518	0.229967	0.293875	0.086627	0.038248	0.067799	0.229967
Canada	0.052956	0.003688	0.006663	0.016089	0.891387	0.02553	0.003688
Central African Republic	0.031979	0.367826	0.123455	0.045277	0.025598	0.038038	0.367826
Chad	0.035771	0.356148	0.130677	0.049895	0.029278	0.042082	0.356148
Chile	0.379189	0.011118	0.022502	0.078502	0.342866	0.154706	0.011118
China	0.203843	0.027528	0.065588	0.281308	0.104074	0.29013	0.027528
Colombia	0.223096	0.014976	0.035075	0.215236	0.073961	0.422679	0.014976
Congo, Rep.	0.030968	0.352879	0.152687	0.047986	0.024012	0.03859	0.352879
Costa Rica	0.357699	0.013601	0.027986	0.116392	0.237558	0.233163	0.013601
Cote d'Ivoire	0.02685	0.099906	0.666795	0.051067	0.018278	0.037198	0.099906
Croatia	0.736556	0.005583	0.011738	0.044952	0.090903	0.104684	0.005583
Cyprus	0.433944	0.012378	0.024201	0.06981	0.317199	0.130091	0.012378
Czech Republic	0.412351	0.009444	0.01903	0.070815	0.340324	0.138592	0.009444
Denmark	0.093398	0.009436	0.016393	0.035064	0.784852	0.051419	0.009436
Dominican Republic	0.276447	0.012561	0.029023	0.176063	0.067343	0.426001	0.012561
Ecuador	0.155706	0.007395	0.018616	0.209392	0.037601	0.563895	0.007395
Egypt, Arab Rep.	0.1621	0.067156	0.129892	0.255775	0.104051	0.213871	0.067156
El Salvador	0.154778	0.045695	0.102048	0.308283	0.091637	0.251864	0.045695
Estonia	0.027953	0.001538	0.00286	0.007495	0.946205	0.012412	0.001538
Eswatini	0.090795	0.127803	0.306297	0.159935	0.060344	0.127024	0.127803
Ethiopia	0.022303	0.385912	0.128059	0.032831	0.01805	0.026932	0.385912
Finland	0.078987	0.006967	0.012356	0.027377	0.826127	0.04122	0.006967
France	0.1106	0.004337	0.008324	0.023403	0.807862	0.041137	0.004337

Продовження таблиці 3.22

Gambia, The	0.031155	0.10822	0.630366	0.058528	0.02104	0.042472	0.10822
Georgia	0.410827	0.010212	0.024054	0.126802	0.109367	0.308526	0.010212
Germany	0.022778	0.001487	0.002743	0.006883	0.95369	0.010932	0.001487
Ghana	0.075927	0.131764	0.388682	0.120454	0.054195	0.097215	0.131764
Greece	0.487909	0.014813	0.030501	0.101154	0.151299	0.19951	0.014813
Guatemala	0.12922	0.034294	0.092738	0.379972	0.062206	0.267276	0.034294
Guinea	0.01781	0.409439	0.102783	0.025327	0.013958	0.021244	0.409439
Haiti	0.045575	0.294334	0.21069	0.065953	0.033873	0.055241	0.294334
Honduras	0.118761	0.055618	0.140268	0.345611	0.065501	0.218623	0.055618
Hungary	0.724424	0.004973	0.010485	0.047568	0.086796	0.120782	0.004973
Iceland	0.086059	0.006562	0.011587	0.026316	0.82182	0.041095	0.006562
India	0.119095	0.099616	0.20279	0.230002	0.082841	0.16604	0.099616
Indonesia	0.11457	0.021924	0.062895	0.45462	0.051986	0.272082	0.021924
Iran, Islamic Rep.	0.194213	0.031775	0.069744	0.293777	0.090771	0.287945	0.031775
Iraq	0.153854	0.058274	0.121365	0.294678	0.087252	0.226304	0.058274
Ireland	0.126812	0.005816	0.01128	0.030173	0.768515	0.051587	0.005816
Israel	0.331814	0.009121	0.01774	0.053751	0.475915	0.102538	0.009121
Italy	0.639756	0.007358	0.015582	0.063545	0.120472	0.145929	0.007358
Jamaica	0.202736	0.035113	0.078797	0.246646	0.114345	0.287249	0.035113
Japan	0.111949	0.005039	0.009634	0.027474	0.794013	0.046852	0.005039
Jordan	0.214386	0.031617	0.067439	0.257246	0.119506	0.27819	0.031617
Kazakhstan	0.516958	0.011982	0.024578	0.093146	0.141394	0.19996	0.011982
Kenya	0.044401	0.209547	0.369827	0.075953	0.032313	0.058412	0.209547
Korea, Rep.	0.268478	0.011379	0.021804	0.066937	0.504259	0.115764	0.011379
Kyrgyz Republic	0.143451	0.02298	0.055718	0.407726	0.062005	0.28514	0.02298
Latvia	0.437988	0.007943	0.016215	0.055114	0.359487	0.115311	0.007943
Lesotho	0.059002	0.226388	0.273523	0.093907	0.044843	0.07595	0.226388
Liberia	0.031413	0.370473	0.123115	0.042585	0.02542	0.036522	0.370473
Lithuania	0.441553	0.013359	0.026912	0.077117	0.280125	0.147574	0.013359
Luxembourg	0.106207	0.009503	0.016896	0.039878	0.758195	0.059816	0.009503
Madagascar	0.027326	0.389019	0.101916	0.037833	0.022735	0.032151	0.389019
Malawi	0.023559	0.400434	0.095709	0.032761	0.019349	0.027755	0.400434
Malaysia	0.615458	0.007085	0.014689	0.062647	0.145381	0.147655	0.007085
Maldives	0.282412	0.018012	0.038647	0.197962	0.106331	0.338623	0.018012
Mali	0.034298	0.307061	0.226183	0.056051	0.025612	0.043735	0.307061
Malta	0.462106	0.008956	0.01789	0.065197	0.305436	0.131461	0.008956
Mauritania	0.065345	0.160909	0.3592	0.121095	0.044172	0.08837	0.160909
Mauritius	0.319136	0.012179	0.027218	0.163077	0.122026	0.344186	0.012179
Mexico	0.217449	0.017137	0.039186	0.237944	0.072101	0.399046	0.017137
Moldova	0.352716	0.019616	0.045191	0.16155	0.118915	0.282394	0.019616
Mongolia	0.163597	0.048297	0.142779	0.279693	0.081317	0.23602	0.048297
Montenegro	0.637869	0.007713	0.016856	0.072872	0.077589	0.179386	0.007713
Morocco	0.223995	0.025984	0.058981	0.263812	0.097018	0.304226	0.025984

Продовження таблиці 3.22

Mozambique	0.023318	0.398229	0.099661	0.033623	0.018931	0.028009	0.398229
Myanmar	0.103998	0.068702	0.227224	0.298155	0.059992	0.173227	0.068702
Namibia	0.079341	0.156404	0.338414	0.112309	0.059328	0.0978	0.156404
Nepal	0.121798	0.117039	0.215666	0.188352	0.087664	0.152441	0.117039
Netherlands	0.086907	0.007822	0.013995	0.032919	0.801691	0.048844	0.007822
New Zealand	0.142258	0.013276	0.023376	0.050176	0.681919	0.07572	0.013276
Nicaragua	0.109701	0.04777	0.131395	0.394526	0.058533	0.210303	0.04777
Niger	0.058153	0.282993	0.177403	0.08144	0.048614	0.068403	0.282993
Nigeria	0.060426	0.242094	0.239439	0.094208	0.04558	0.07616	0.242094
North Macedonia	0.450143	0.011801	0.025937	0.127189	0.079255	0.293872	0.011801
Norway	0.088023	0.008668	0.015034	0.032191	0.799844	0.047572	0.008668
Pakistan	0.094574	0.104209	0.269152	0.224772	0.060059	0.143025	0.104209
Panama	0.209947	0.013042	0.031292	0.21997	0.065647	0.447059	0.013042
Paraguay	0.302781	0.012658	0.029252	0.16765	0.062576	0.412424	0.012658
Peru	0.214296	0.017956	0.049906	0.25064	0.064211	0.385035	0.017956
Philippines	0.111018	0.026972	0.086009	0.45148	0.045594	0.251954	0.026972
Poland	0.586403	0.008065	0.016502	0.058538	0.197144	0.125284	0.008065
Portugal	0.393579	0.010153	0.020156	0.063896	0.377576	0.124487	0.010153
Romania	0.495857	0.008268	0.01934	0.100967	0.073695	0.293604	0.008268
Russian Federation	0.454425	0.01258	0.026786	0.118221	0.106103	0.269306	0.01258
Rwanda	0.049784	0.299881	0.181624	0.069206	0.041196	0.058428	0.299881
Saudi Arabia	0.199591	0.059054	0.10463	0.208254	0.156353	0.213064	0.059054
Senegal	0.059308	0.117785	0.469517	0.113377	0.039789	0.082439	0.117785
Serbia	0.541195	0.011474	0.024754	0.100784	0.096019	0.2143	0.011474
Sierra Leone	0.022678	0.403951	0.093761	0.030689	0.018703	0.026266	0.403951
Singapore	0.185319	0.021709	0.038808	0.093965	0.508135	0.130355	0.021709
Slovak Republic	0.695767	0.005431	0.011259	0.044368	0.135494	0.10225	0.005431
Slovenia	0.263176	0.008252	0.016121	0.049956	0.563756	0.090487	0.008252
Solomon Islands	0.046286	0.255733	0.275223	0.072737	0.036166	0.058122	0.255733
South Africa	0.155591	0.060787	0.144251	0.25221	0.094986	0.231388	0.060787
South Sudan	0.046361	0.330955	0.139494	0.060891	0.038099	0.053245	0.330955
Spain	0.225142	0.007652	0.014587	0.042972	0.6225	0.079495	0.007652
Sri Lanka	0.161695	0.029303	0.066915	0.345579	0.079155	0.288051	0.029303
Sudan	0.056655	0.220764	0.291897	0.096537	0.039364	0.07402	0.220764
Sweden	0.088339	0.008789	0.015306	0.033396	0.796141	0.049242	0.008789
Switzerland	0.135147	0.014707	0.025747	0.05898	0.666251	0.084461	0.014707
Tajikistan	0.137111	0.046899	0.108016	0.358845	0.074521	0.227707	0.046899
Tanzania	0.016067	0.417679	0.092062	0.023888	0.013079	0.019545	0.417679
Thailand	0.350569	0.012908	0.028189	0.162156	0.102156	0.331115	0.012908
Togo	0.013484	0.427292	0.084819	0.020182	0.010458	0.016472	0.427292
Tunisia	0.235847	0.026862	0.061217	0.251077	0.100046	0.298089	0.026862
Turkey	0.268719	0.020145	0.044893	0.222369	0.096301	0.327428	0.020145
Uganda	0.01551	0.418119	0.094295	0.023017	0.012102	0.018837	0.41812

Продовження таблиці 3.22

Ukraine	0.341735	0.013617	0.031374	0.177961	0.07748	0.344216	0.013617
United Arab Emirates	0.251315	0.027197	0.050849	0.134542	0.326764	0.182136	0.027197
United Kingdom	0.031136	0.002037	0.003722	0.009327	0.936777	0.014965	0.002037
United States	0.186576	0.008151	0.015441	0.044273	0.658791	0.078616	0.008151
Uruguay	0.266257	0.01102	0.022129	0.071906	0.486789	0.130879	0.01102
Uzbekistan	0.167366	0.022811	0.051646	0.346597	0.07228	0.316489	0.022811
Vanuatu	0.062559	0.141762	0.417221	0.108357	0.045419	0.08292	0.141762
Venezuela, RB	0.1576	0.099519	0.153788	0.19638	0.105537	0.187658	0.099519
Vietnam	0.320295	0.017606	0.040947	0.183606	0.108697	0.311244	0.017606
Zambia	0.01405	0.428409	0.079926	0.020909	0.011089	0.017208	0.428409

Отримали матрицю належності (див. табл. 3.22). Згрупуємо отримані результати по кластерам (див. табл. 3.23).

Таблиця 3.23 – Результати кластеризації методом Бездека (К=7)

№ кластеру	Країни
0	Albania Argentina Azerbaijan Belarus Bulgaria Chile Costa Rica Croatia Cyprus Czech Republic Georgia Greece Hungary Italy Kazakhstan Latvia Lithuania Malaysia Malta Moldova Montenegro North Macedonia Poland Portugal Romania Russian Federation Serbia Slovak Republic Thailand Vietnam
1	Angola Burkina Faso Guinea Haiti Liberia Madagascar Malawi Mozambique Rwanda Sierra Leone South Sudan Tanzania Togo Zambia
2	Bangladesh Cameroon Cote d'Ivoire Eswatini Gambia, The Ghana Kenya Lesotho Mauritania Namibia Nepal Pakistan Senegal Solomon Islands Sudan Vanuatu
3	Algeria Bolivia Botswana Cambodia Egypt, Arab Rep. El Salvador Guatemala Honduras India Indonesia Iran, Islamic Rep. Iraq Kyrgyz Republic Mongolia Myanmar Nicaragua Philippines South Africa Sri Lanka Tajikistan Uzbekistan Venezuela
4	Australia Austria Belgium Canada Denmark Estonia Finland France Germany Iceland Ireland Israel Japan Korea, Rep. Luxembourg Netherlands New Zealand Norway Singapore Slovenia Spain Sweden Switzerland United Arab Emirates United Kingdom United States Uruguay
5	Armenia Brazil China Colombia Dominican Republic Ecuador Jamaica Jordan Maldives Mauritius Mexico Morocco Panama Paraguay Peru Saudi Arabia Tunisia Turkey Ukraine
6	Benin Central African Republic Chad Congo, Rep. Ethiopia Mali Niger Nigeria Uganda

Визначимо координати центроїді отриманих кластерів (див. табл. 3.24).

Таблиця 3.24 – Координати центроїдів, метод Бездека (К=7)

Номер кластеру	Координати центроїда
0	(3.5147 4.10053 76.9419 96.8133 73.489 97.1371 94.4858 99.3964 8.28814 75.9567 40.5833 18.7948 57.2293 5.50395 0.87917 3.61839 46.5999 8.49198)
1	(70.6905 20.1415 63.581 84.9096 88.1664 60.1596 26.5051 40.3858 5.78673 17.1489 46.0297 38.8986 43.9859 0.428836 0.867602 7.61629 30.5665 5.4665)
2	(41.8254 15.0326 65.7449 87.668 72.821 76.1634 43.0766 65.6925 7.84605 31.3797 45.057 37.8677 51.5812 1.00446 0.8732 8.95978 35.3388 5.83992)
3	(16.5782 9.9799 73.1691 94.7204 60.1247 91.3145 80.7844 94.2411 8.01835 50.7138 45.2829 29.3258 65.1639 3.1971 0.822082 9.23405 34.6215 7.02493)
4	(0.752288 2.69559 81.4184 99.2702 83.2018 99.5124 98.4148 99.8284 6.45047 89.1242 34.3355 10.6408 65.7669 8.56819 0.898185 1.60445 74.6902 12.1969)
5	(9.89601 7.08382 75.3375 95.3235 65.4733 94.6221 87.2764 97.6827 8.94426 62.8566 45.5639 23.6254 62.1872 3.74702 0.818282 8.28275 38.6452 7.59816)
6	(3.87889 4.61667 76.4461 96.045 73.5314 96.6902 92.4543 99.7945 8.67583 72.1443 43.7033 18.297 58.9722 4.56864 0.840917 5.47539 41.5 7.88592)

Визначимо критерії якості кластеризації:

WSS = 192280.19294332594

Silhouette criterion = 0.15973058927852285

3.4 Прогнозування показників сталого розвитку регресійними моделями

Для виконання прогнозування стану країн на майбутнє були обрані наступні показники сталого розвитку:

1. Індекс сталого розвитку (SDI) – це показник ефективності, призначений для оцінки екологічної ефективності націй у забезпеченні людського розвитку. Він розраховується як частка двох цифр: індекс розвитку на основі індексу людського розвитку, що розраховується як середнє геометричне індексу тривалості життя, індексу освіти та модифікованого індексу доходу; і індекс екологічного впливу.
2. Очікувана тривалість життя (роки).
3. Валовий національний дохід на душу населення (дол.. США).
4. Викиди CO2 на душу населення (тонн) [23].

Загалом датасет містить інформацію про 139 країн з цими показниками, що зафіксовані щорічно з 1991 до 2019 року. У якості навчальної вибірки вирішено взяти дані з 1991 по 2015 рік, для перевірочної вибірки – з 2016 по 2019 рік.

Для прикладу наведемо результати для однієї країни – Великобританії.

Перш за все, побудуємо графіки вхідних процесів (див. рис. 3.7, рис. 3.8, рис. 3.9, рис. 3.10).

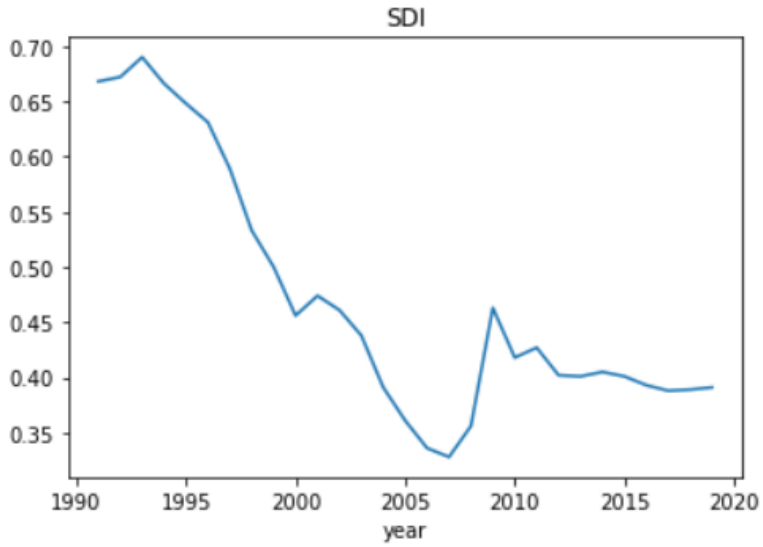

Рисунок 3.7 – Графік вхідного процесу індексу сталого розвитку для Великобританії

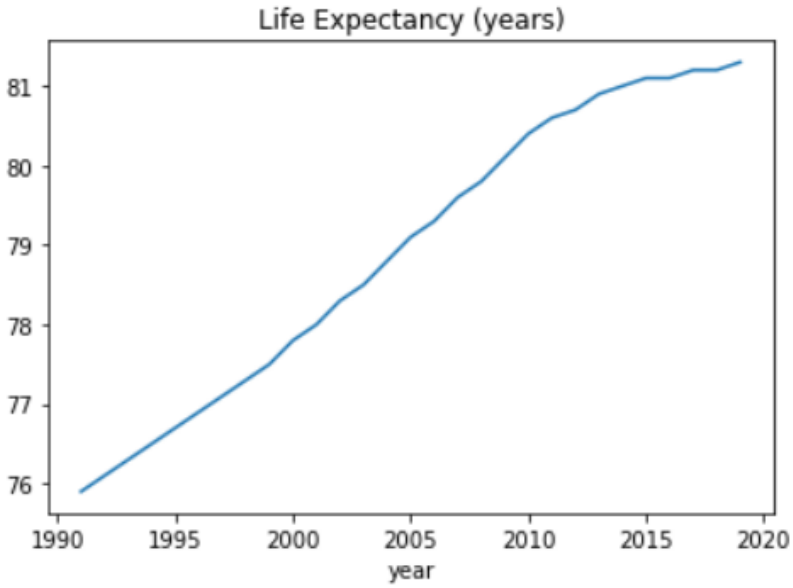

Рисунок 3.8 – Графік вхідного процесу очікуваної тривалості життя для Великобританії

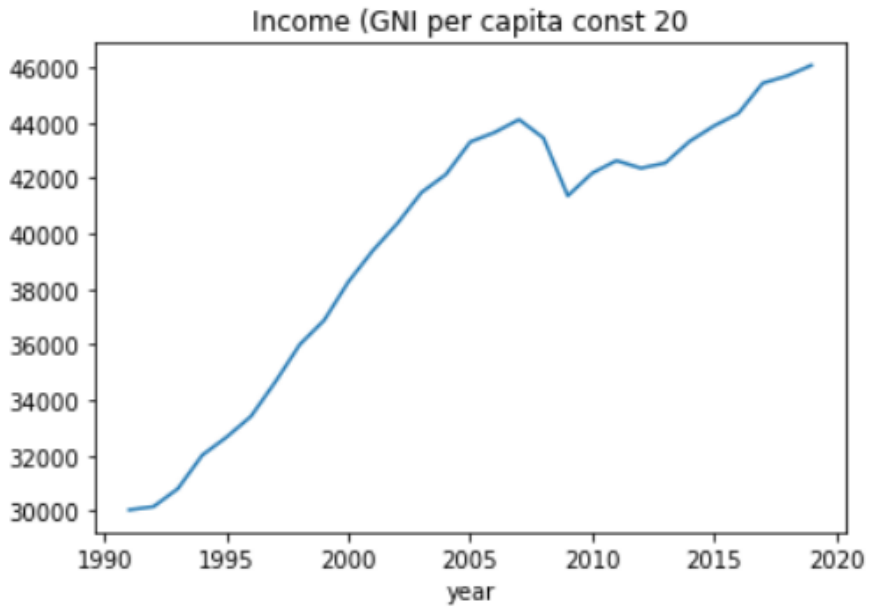

Рисунок 3.9 – Графік вхідного процесу валового національного доходу на душу населення для Великобританії

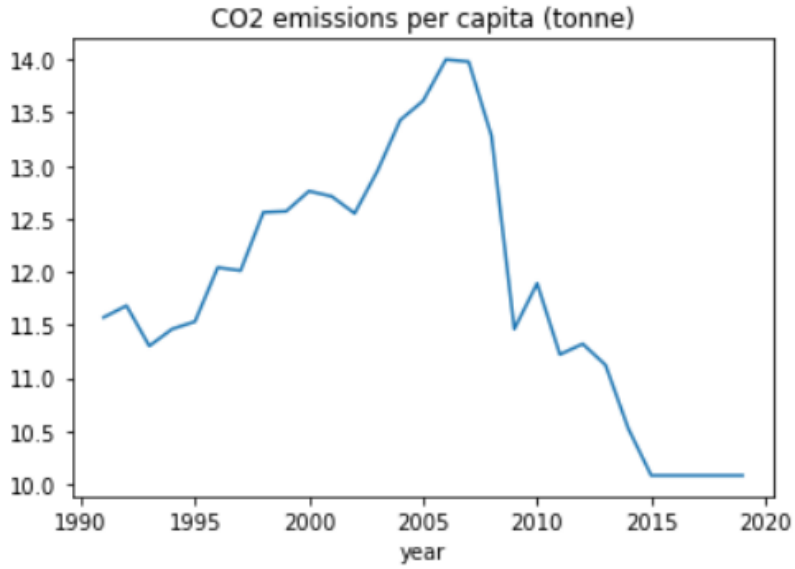

Рисунок 3.10 – Графік вхідного процесу викидів co2 на душу населення

Для моделювання вхідних процесів була обрана модель авто регресії. Спочатку необхідно визначити доцільний порядок авторегресії. Для цього побудуємо часткові автокореляційні функції та визначимо статистично значущі лаги (див. рис. 3.11, рис. 3.12, рис.3.13, рис. 3.14).

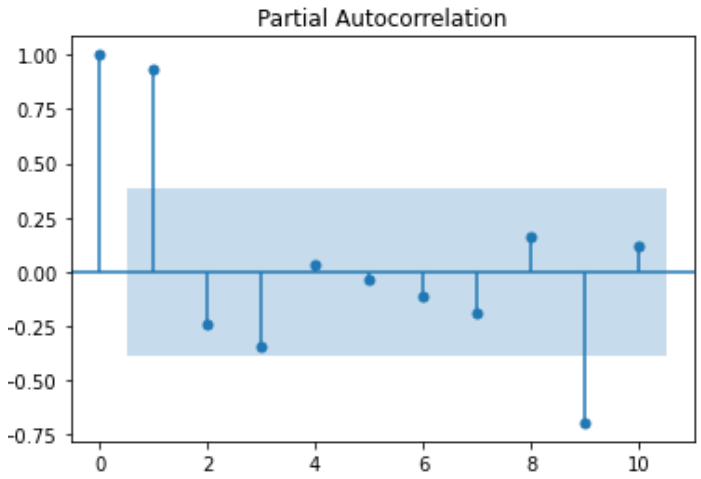

Рисунок 3.11 – ЧАКФ першого процесу

Можемо бачити, що статистично значущими є лаги 1 та 9, тому доцільно будувати авто регресію 9-го порядку АР(9).

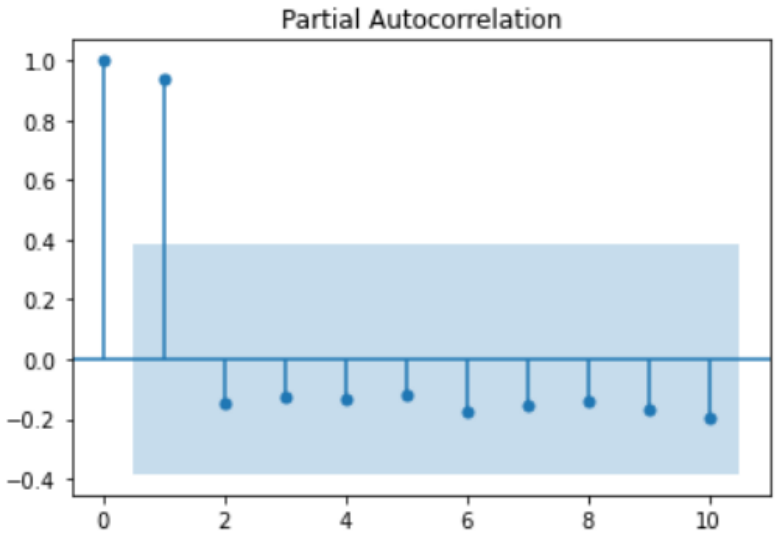

Рисунок 3.12 - ЧАКФ другого процесу

Статистично значущим є лише перший лаг, тому будуватимемо авто регресію першого порядку АР(1).

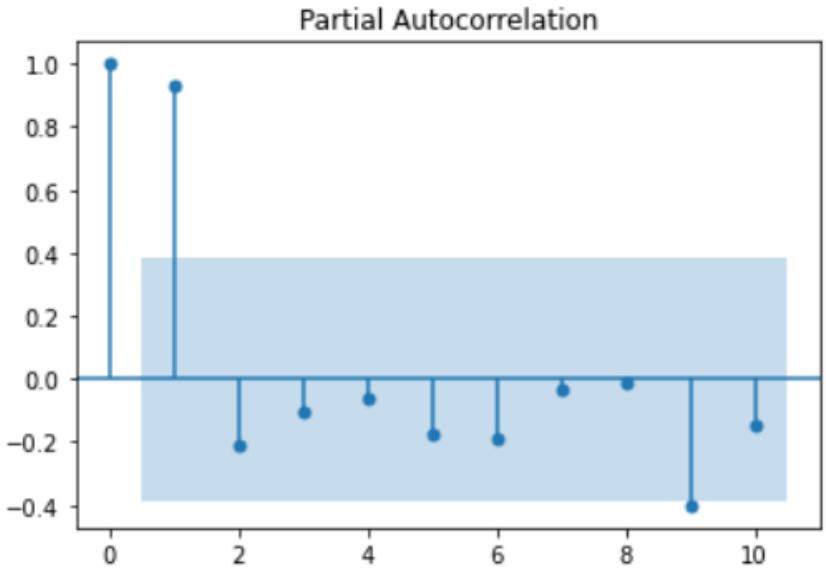

Рисунок 3.13 - ЧАКФ третього процесу

Статистично значущим є перший лаг, тому будуватимемо авторегресію першого порядку АР(1)

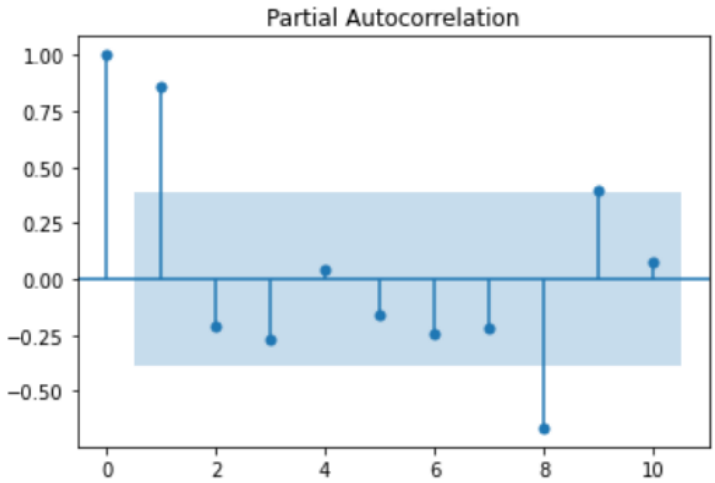

Рисунок 3.14 – ЧАКФ четвертого процесу

Статистично значущими є лаги 1 та 8, тому будуватимемо авторегресію 8-го порядку АР(8).

Опишемо результати моделювання та прогнозування першого процесу (індекс сталого розвитку). На рис. 3.15 зобразимо графіки реальних та прогнозованих значень вхідного процесу.

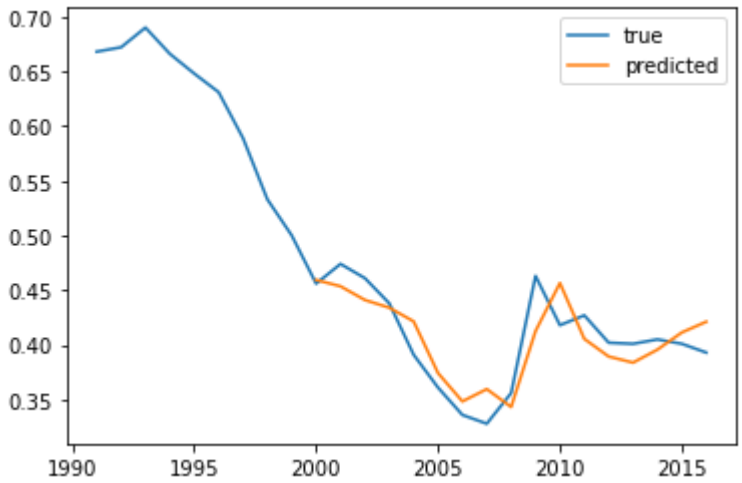

Рисунок 3.15 – Графіки реальних та прогнозованих значень SDI

Отримане рівняння моделі:

$$y(k) = 0.209 + 0.666y(k-1) - 0.129y(k-2) - \\ -0.0579y(k-3) - 0.0379y(k-4) + 0.0424y(k-5) + \\ +0.0568y(k-6) - 0.1496y(k-7) + 0.864y(k-8) - \\ -0.747y(k-9) \quad (3.1)$$

За одержаним рівнянням побудуємо прогнозовані значення на 3 роки вперед (рис. 3.16):

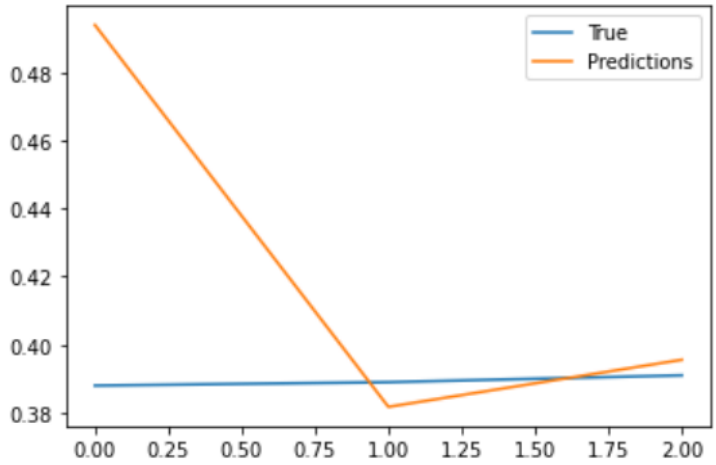

Рисунок 3.16 – Графік прогнозованих та реальних значень тестової вибірки SDI

Отримані точні значення прогнозів та реальні значення відобразимо у таблиці (див. табл. 3.25). У табл. 3.26 занесемо значення критеріїв якості.

Таблиця 3.25 – Прогнозовані та реальні значення SDI

Рік	Реальні значення	Прогнозовані
2017	0.388000	0.493858
2018	0.389000	0.381743
2019	0.391000	0.395573

Таблиця 3.26 - Отримані критерії адекватності моделі та якості прогнозів моделі АР(9)

Критерії адекватності моделі	Sum squared resid	0.009187484781499854
	Durbin-Watson	2.005936262745162
	R squared	0.7033638214371474
Критерії якості прогнозів	RMSE	0.06131746771457523
	MAPE	10.106006855203702
	Theil	0.07514491791772163

Дослідимо другий вхідний процес оцікуваної тривалості життя у Великобританії. На рис. 3.17 зобразимо графіки реальних та прогнозованих значень вхідного процесу.

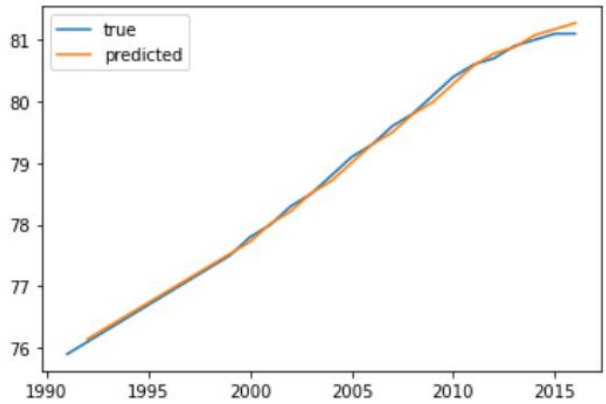

Рисунок 3.17 – Графік прогнозованих та реальних значень процесу life

Отримане рівняння моделі:

$$y(k) = 1.3068 + 0.986\, y(k-1) \tag{3.2}$$

За одержаним рівнянням побудуємо прогнозовані значення на 3 роки вперед (див. рис. 3.18):

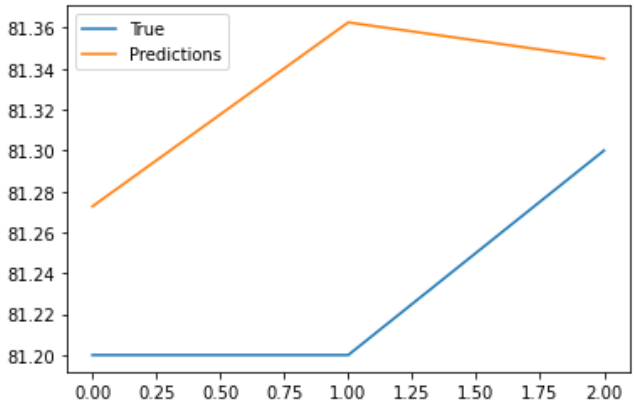

Рисунок 3.18 – Графік прогнозованих та реальних значень тестової вибірки Life

Отримані точні значення прогнозів та реальні значення відобразимо у таблиці (див. табл. 3.27). У табл. 3.28 занесемо значення критеріїв якості.

Таблиця 3.27– Прогнозовані та реальні значення Life

Рік	Реальні значення	Прогнозовані
2017	81.2	81.272644
2018	81.2	81.362615
2019	81.3	81.344914

Таблиця 3.28 - Отримані критерії адекватності моделі та якості прогнозів моделі АР(1) процесу Life

Критерії адекватності моделі	Sum squared resid	0.12463386720871744
	Durbin-Watson	1.1050198935728306
	R squared	0.9981821189591946
Критерії якості прогнозів	RMSE	0.10604698916013346
	MAPE	0.11499059132187317
	Theil	0.0006523556636952678

Дослідимо третій вхідний процес - Валовий національний дохід на душу населення для Великобританії. На рис. 3.19 зобразимо графіки реальних та прогнозованих значень вхідного процесу.

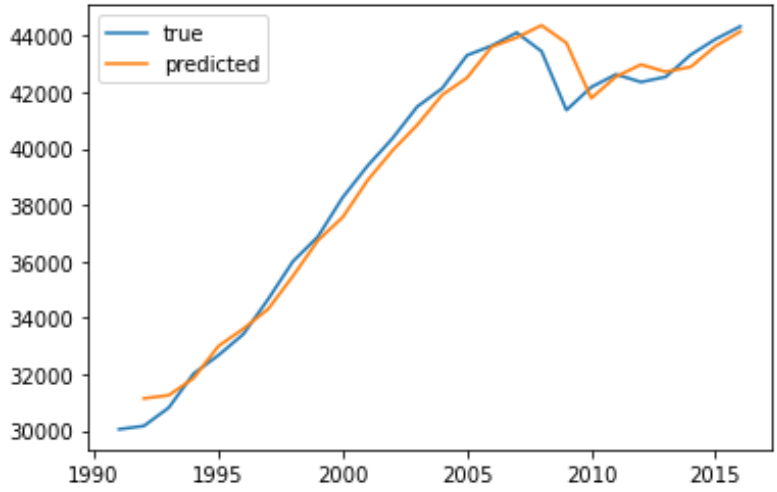

Рисунок 3.19 – Графік прогнозованих та реальних значень тестової вибірки ВНД

Отримане рівняння моделі:

$$y(k) = 2879.7404 + 0.9406\, y(k-1) \tag{3.3}$$

За одержаним рівнянням побудуємо прогнозовані значення на 3 роки вперед (див. рис.3.20):

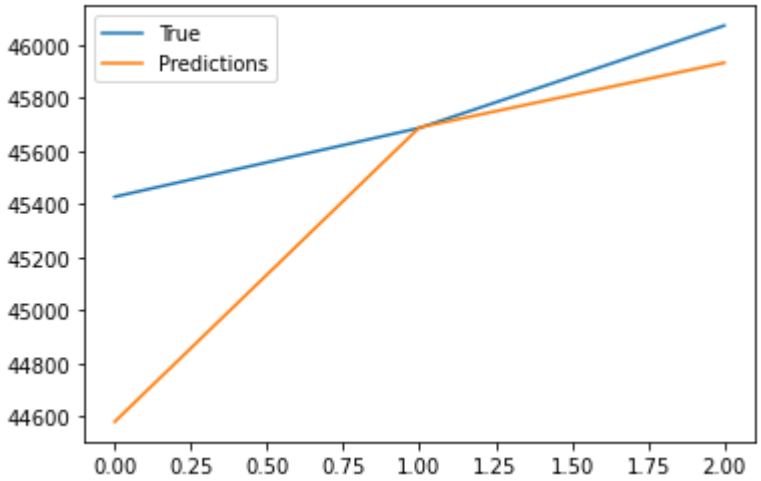

Рисунок 3.20 – Графік прогнозованих та реальних значень тестової вибірки ВНД

Отримані точні значення прогнозів та реальні значення відобразимо у таблиці (див. табл. 3.29). У табл. 3.30 занесемо значення критеріїв якості.

Таблиця 3.29 – Прогнозовані та реальні значення ВНД

Рік	Реальні значення	Прогнозовані
2017	45427	44580.843708
2018	45686	45688.095483
2019	46071	45931.230667

Таблиця 3.30 - Отримані критерії адекватності моделі та якості прогнозів моделі АР(1) процесу ВНД

Критерії адекватності моделі	Sum squared resid	11209136.958979623
	Durbin-Watson	1.3512493729892097
	R squared	0.9785167411584582
Критерії якості прогнозів	RMSE	495.1499195324164
	MAPE	0.7235458161056371
	Theil	0.005433289279442756

Дослідимо четвертий вхідний процес - викиди CO2 на душу населення для Великобританії. На рис. 3.21 зобразимо графіки реальних та прогнозованих значень вхідного процесу.

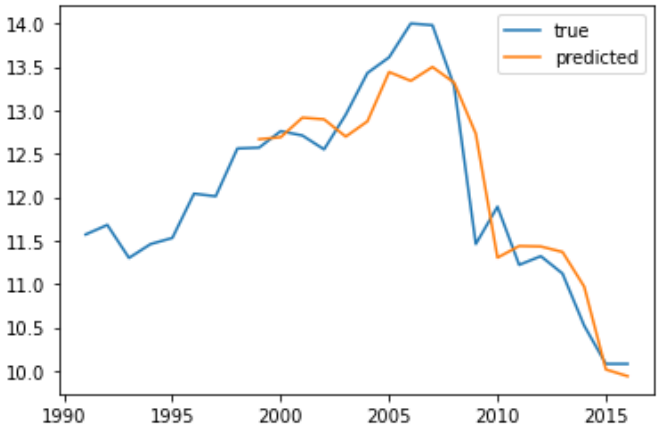

Рисунок 3.21 – Графік прогнозованих та реальних значень тестової вибірки CO2

Отримане рівняння моделі:

$$y(k) = 7.125 + 0.823\, y(k-1) + 0.169\, y(k-2) - \\ -0.2955\, y(k-3) - 0.1004 y(k-4) + 0.0213 y(k-5) + \\ +0.244\, y(k-6) + 0.0563 y(k-7) - 0.499 y(k-8) \quad (3.4)$$

За одержаним рівнянням побудуємо прогнозовані значення на 3 роки вперед (див.рис.3.22):

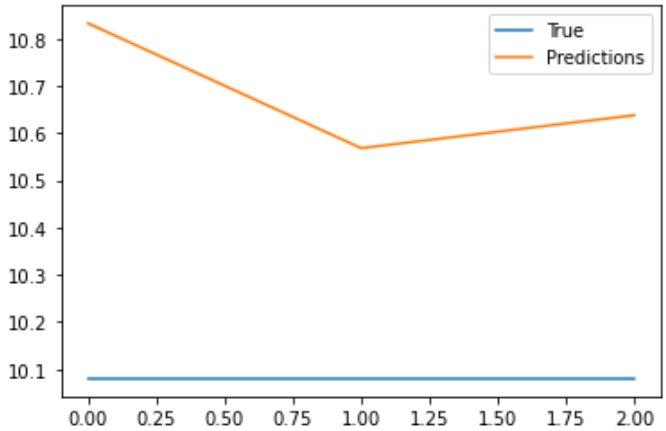

Рисунок 3.22 – Графік прогнозованих та реальних значень тестової вибірки CO2

Отримані точні значення прогнозів та реальні значення відобразимо у таблиці (див. табл. 3.31). У табл. 3.32 занесемо значення критеріїв якості.

Таблиця 3.31 – Прогнозовані та реальні значення CO2

Рік	Реальні значення	Прогнозовані
2017	10.08	10.832901
2018	10.08	10.568353
2019	10.08	10.638153

Таблиця 3.32 - Отримані критерії адекватності моделі та якості прогнозів моделі АР(8) процесу CO2

Критерії адекватності моделі	Sum squared resid	3.5535506045756313
	Durbin-Watson	2.0316700319669128
	R squared	0.8722994946151916
Критерії якості прогнозів	RMSE	0.6101594018959922
	MAPE	5.950421100024767
	Theil	0.02939055709757921

3.5 Порівняння отриманих результатів

У ході виконання попереднього пункту магістерської дисертації було виконано кластеризацію наступними методами інтелектуального аналізу даних: метод к-середніх (при к від 5 до 7), методом агломеративної кластеризації (при к від 5 до 7), методом DBSCAN (не показав хороших результатів) та нечітким методом Бездека (при к від 5 до 7).

Для порівняння отриманих моделей були застосовані метрики порівняння алгоритмів кластеризації (WSS та критерій Silhouette). Побудуємо таблицю порівняння для всіх моделей (див. табл. 3.33).

Таблиця 3.33 – Порівнянння алгоритмів кластеризації за критеріями якості

Метод кластеризації	WSS	Silhouette критерій
k-середніх (k=5)	185452.4821	0.28649303
k-середніх (k=6)	167605.7932	0.259262888
k-середніх (k=7)	154342.4752	0.26213318
Агломеративний метод (k=5)	196142.1882	0.226056164
Агломеративний метод (k=6)	175691.5083	0.247517792
Агломеративний метод (k=7)	160163.2809	0.251343102
Нечіткий метод (k=5)	197141.4953	0.231793757
Нечіткий метод (k=6)	203914.7181	0.189207435
Нечіткий метод (k=7)	192280.1929	0.159730589

Отже, можемо бачити, що за критеріями якості і за WSS, і за критерієм силуету найкращою моделлю виявилась модель k-середніх при кількості кластерів К=7. Серед інших алгоритмів гарно себе показали також агломеративний метод при К=7 та метод нечіткої кластеризації Бездека при К=5.

Побудуємо також окремо зведену таблицю для критеріїв адекватності моделей вхідних процесів та якості прогнозів (див. табл. 3.34).

Таблиця 3.34 – Порівняння результатів моделювання та прогнозування

Назва вхідного процесу	Назва моделі	Назва критерію		Значення критерію
CO2	AP(9)	Критерії адекватності моделі	Sum squared resid	0.009187484781499854
			Durbin-Watson	2.005936262745162
		Критерії якості прогнозів	R squared	0.7033638214371474
			RMSE	0.06131746771457523
			MAPE	10.106006855203702
			Theil	0.07514491791772163
Life	AP(1)	Критерії адекватності моделі	Sum squared resid	0.12463386720871744
			Durbin-Watson	1.1050198935728306
			R squared	0.9981821189591946
		Критерії якості прогнозів	RMSE	0.10604698916013346
			MAPE	0.11499059132187317
			Theil	0.0006523556636952678
ВНД	AP(1)	Критерії адекватності моделі	Sum squared resid	11209136.958979623
			Durbin-Watson	1.3512493729892097
			R squared	0.9785167411584582
		Критерії якості прогнозів	RMSE	495.1499195324164
			MAPE	0.7235458161056371
			Theil	0.005433289279442756

Продовження таблиці 3.34

CO2	AP(9)	Критерії адекватності моделі	Sum squared resid	0.009187484781499854
			Durbin-Watson	2.005936262745162
			R squared	0.7033638214371474
		Критерії якості прогнозів	RMSE	0.06131746771457523
			MAPE	10.106006855203702
			Theil	0.07514491791772163

3.6 Висновки до розділу

Отже, у ході виконання третього розділу магістерської дисертації був розроблений програмний продукт мовою програмування Python у середовищі програмування Spyder, який є складовою дистрибутиву Anaconda. Він дозволяє виконувати кластеризацію вхідних даних різними методами інтелектуального аналізу даних, а саме: к-середніх, агломеративним, DBSCAN та нечітким методом к-середніх (Бездека). Була виконана кластеризація країн ООН за показниками сталого розвитку (всього на вхід було подано 18 показників, які описують відповідні цілі сталого розвитку).

Найкращою моделлю серед одержаних виявилась модель k-середніх при кількості кластерів k = 7. До 0-го кластеру увійшли країни, що мають практично всі найкращі показники сталого розвитку. Це найбільш розвинені країни, такі як Австралія, США, Австрія, Бельгія, Канада, Чилі і так далі. В основному, в цю групу належать країни Європи. Серед показників сталого розвитку цієї групи, найгіршим є показник викиду co2 на душу населення (він є

найгіршим серед усіх кластерів). Тому цим країнам необхідно більше уваги приділяти збереженню довкіллю та зменшенню викидів вуглекислого газу, переходу на безпечні та екологічні джерела енергії. Також не найкращими показниками є рівень безробіття та гендерної рівності.

До 1-го кластеру увійшли країни, які мають практично всі найгірші показники. Серед найгірших показників виділяються наступні: поріг бідності 3,2$/день (76% населення), відсоток недоїдання (20,5%), % дітей, що навчаються у школі (82%), базові послуги води (57%), базові санітарні послуги (23%), доступ до електрики (34%), доступ до інтернету (13%), задоволеність транспортом (30%), ВВП на науку та здоров'я (5.05). Проте, у цієї групи є показники найкращі серед інших груп, а саме рівень безробіття (4,8%), викиди co2 на душу населення (0,27), гендерна рівність (90%). До цього кластеру входять країни Центральної, Східної та Західної Африки, такі як Ангола, Бенін, Чад, Конго, Мозамбік тощо. Основна проблема цих країн полягає у забезпеченні основними базовими послугами людей (вода, санітарні умови, доступ до навчання, до Інтернету), а також бідність і голод.

До 2-го кластеру увійшли країни, які мають одні з найгірших показників сталого розвитку. Серед найгірших показників можна виділити % дітей, що навчаються у школі (84%), забрудненість повітря (68.68 $мг/м^3$), рівень корупції (30%), ВВП на науку і здоров'я (4.56% від ВВП), інші показники також одні з найгірших (гірші лише у країн з 1-го кластеру). Серед хороших показників можна виділити лише викиди co2 на душу населення (0.8). До цієї групи входять країни Південної Азії та деякі країни Африки, такі як Бангладеш, Камерун, Непал, Індія, Пакистан тощо.

До 3-го кластеру увійшли країни, які мають середні значення практично всіх показників сталого розвитку. Серед найгірших можна виділити Індекс виживання видів з Червоної книги (0.81), кількість умисних вбивств на 100 тис. населення (15.78), рівень корупції (30.35). Найкращим показником цієї групи є рівень задоволеності громадським транспортом (67%). До цієї групи увійшли

країни, в основному, Центральної Америки та країни Азії, такі як Болівія, Таджикістан, Узбекистан, Венесуела, Монголія, Індонезія, Ель Сальвадор тощо. Основна проблема цих країн – правосуддя та мир.

До 4-го кластеру увійшли країни, які мають погані показники % недоїдання (20.07%), найгірший показник тривалості життя (62.7 років), доступ до базових потреб у воді (75%), доступ до базових санітарних послуг (39%), доступ до електрики (63%), рівень безробіття (11.27%), найгірший коефіцієнт Джині (50.25), високий рівень умисних вбивств (11.66). Серед хороших показників у цих країнах виділяється гендерна рівність (84.5%), викиди co2 на душу населення (0.8). До цієї групи належать країни Африки та деякі країни Океанії, а саме: Ботсвана, Гамбія, Гаїті, Кенія, Лесото, Соломонські острови, Вануату тошо.

До 5-го кластеру увійшли країни, які мають, в основному, хороші показники сталого розвитку, проте мають найгірший показник гендерної рівності (28.47%), найвищий рівень безробіття (12.82%), високий показник забрудненості повітря (52.7 мг/м^3) та високий рівень викиду co2 на душу населення (5.61). Серед хороших показників можна виділити низький рівень бідності та недоїдання (9.82% та 6.5% відповідно), високі відсотки доступу до базових умов води, санітарії та електрики (96%, 93% та 99,9%), найвищий відсоток виживання видів з Червоної книги, низький рівень вбивств (2.9), досить висока тривалість життя (75.5 років). До цієї групи належать країни Середнього Сходу та Північної Африки, а саме: Алжир, Єгипет, Туніс, Іран, Ірак, Саудівська Аравія, Туреччина, Марокко. Основна проблема цих країн – гендерна рівність, безробіття та екологія.

До 6-го кластеру належать країни, які мають показники сталого розвитку вище середнього по бідності (3.8%), недоїданню (4.6%), тривалості життя (76.4 роки), % дітей що навчаються у школі (96%), базові умови по воді (96%), санітарії (92%) та електриці (99.8%), доступу до інтернету (72%), невелике забруднення повітря (18.3 мг/м^3), невелику кількість умисних вбивств (5.47 на

100 тис. населення). Проте, є показники нище середнього, серед яких рівень безробіття (8.67%), рівень викиду co2 на душу населення (4.56), індекс виживання видів з Червоної книги (0.84). Всі інші показники мають середнє значення серед всіх кластерів. До цього кластеру увійшли країни Закавказзя (Грузія, Вірменія, Азербайджан), Східної Європи (Болгарія, Україна, Білорусь, Росія, Словаччина, Польща, Румунія, Угорщина), країни Балканського півострова (Албанія, Хорватія, Північна Македонія, Сербія, Чорногорія, Греція), країни Південної Америки (Аргентина, Бризилія, Перу, Колумбія, Еквадор тощо), Італія та деякі інші країни. Загалом, можна сказати, що у цей кластер увійшли країни, що розвиваються. Найбільші проблеми ці країни мають у питаннях безробіття та екології.

Також за допомогою регресійних моделей була реалізована можливість моделювання процесів сталого розвитку та прогнозування значень на майбутнє. Зокрема, як приклад, наведено дослідження процесів сталого розвитку Великобританії, змодельовані та спрогнозовані такі показники як індекс сталого розвитку, середня тривалість життя, валовий національний дохід на душу населення, викиди co2 на душу населення. Програмний продукт дозволяє виконати таке моделювання та прогнозування для 139 країн.

РОЗДІЛ 4 РОЗРОБКА ВЛАСНОГО СТАРТАП-ПРОЕКТУ

У сучасному світі щодня створюються чимало стартап-проектів, які є досить цікавими та перспективними, проте лише одиницям вдається вийти на ринок та отримати реальний прибуток від реалізації. Насамперед, створюваний продукт повинен вирішувати актуальні задачі, що постають перед людьми, він має бути оригінальним або кращим за відповідні аналоги. Разом з тим, авторами стартап-проекту необхідно прорахувати всі очікувані та неочікувані витрати на реалізацію, виконати фінансово-економічний та маркетинговий аналіз продукту та організувати вихід продукту на ринок.

Якщо говорити про стартапи, що дозволяють виконувати кластерний аналіз та прогнозувати значення вхідного процесу, то їх на сьогоднішній день існує не дуже велика кількість. Саме тому основною задачею є створення оригінального та кращого за аналоги програмного продукту, який дозволяє швидко та зручно виконати кластерний аналіз і, в той же час, спрогнозувати значення вхідного процесу на короткостроковий період. Продукт має бути зручним у використанні, зрозумілим у своїй реалізації, виконувати точно і чітко поставлені перед ним завдання.

4.1 Карта стартап-проекту

Стартап-проект заключається у створенні системи підтримки прийняття рішень, яка дозволяє кластеризувати дані за вхідними параметрами, а саме за показниками сталого розвитку, а також виконувати короткострокові прогнози регресійними моделями. На основі даної системи пропонується надавати

рекомендації правлінням держав стосовно розподілу державного бюджету у різні сфери, спираючись на те, які саме показники сталого розвитку та цілі сталого розвитку найбільше відстають від ідеальних значень.

У таблиці 4.1 представлена основна інформація про проект, розкрита ідея та рішення поставленої задачі.

Таблиця 4.1 – Інформаційна карта проекту

Назва проекту	Система кластеризації та прогнозування показників сталого розвитку країн ООН методами інтелектуального аналізу даних
Автори проекту	Самсонюк Максим Вікторович
Коротка анотація	Проект спрямований на обробку датасету, що містить показники сталого розвитку для країн ООН та їх історичні дані та на допомогу користувачеві прийняти рішення стосовно розподілу державного бюджету на покращення відстаючих сфер.
Термін реалізації проекту	6 місяців
Необхідні ресурси	Персональний комп’ютер з встановленим програмним забезпеченням, доступ до електрики, фінансові кошти на оплату комунальних послуг, фінансові кошти на оплату заробітної платні виконавцям на термін 6 місяців, приміщення з доступом до необхідних комунікацій.

Продовження таблиці 4.1

Опис проблеми, які вирішує проект	Система дає змогу комплексно обробити вхідні дані про сталий розвиток, розбити країни на кластери, кожен з яких відповідає певним недолікам, а також дозволяє виконувати короткострокові прогнозування показників. Це дозволяє комплексно зрозуміти ситуацію у країні, у який бік рухається та усунути виявлені проблеми у державному управлінні.
Головні цілі та завдання проекту	Мета проекту – створення системи, яка якісно виконуватиме кластеризацію вхідних даних, будуватиме якісні короткострокові прогнози та надаватиме рекомендації аналітику стосовно недоліків та проблем у державному управлінні.
Очікувані результати	Автономна система підтримки прийняття рішень, яка здатна працювати з великим об’ємом даних та на необмеженій кількості користувачів в автономному режимі.

4.2 Технологічний аудит ідеї проекту

Опишемо основну ідею стартап-проекту у вигляді таблиці (див. табл. 4.2).

Таблиця 4.2 – Опис ідеї стартапу

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Основна ідея проекту заключається у створенні системи, яка буде комплексно аналізувати ситуацію із досягненням цілей сталого розвитку країнами ООН, точно будувати короткострокові прогнози. За допомогою неї аналітики зможуть надавати рекомендації стосовно розподілу державного бюджету та направляти ресурси на досягнення цілей сталого розвитку	Опрацювання історичних даних по зміні показників сталого розвитку країн ООН, побудова точних кластерів, які дозволяють виявити групи країн, відстаючих у тій чи іншій цілі сталого розвитку	Користувач за допомогою розробленої системи може контролювати ситуацію з досягненням цілей сталого розвитку, зможе отримати рекомендації стосовно подальшої ситуації у країні, на основі чого користувач зможе правильно та доцільно надати рекомендації стосовно розподілення державного бюджету

Наступним кроком являється аналіз існуючих конкурентних проектів, визначення їх недоліків та переваг, також далі наведений опис переваг розробленого стартап-проекту і визначення його доцільності. Результати аналізу відобразимо у вигляді таблиці (табл. 4.3).

Таблиця 4.3 – Визначення слабких, сильних та нейтральних характеристик ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W	N	S
		Власний проект	SPSS	The Unscrambler	XL Stat			
1	Точність кластеризації	Надає кращий результат із списку реалізованих алгоритмів	Свій власний алгоритм	Свій власний алгоритм	Свій власний алгоритм			+
2	Можливість побудови прогнозів	Є можливість побудови прогнозів регресійними моделями	Є	Є	Є		+	
3	Ризики невірної кластеризації та прогнозу	Залежить від обсягу вхідних даних	Мають власний інтерфейс	Мають власний інтерфейс	Мають власний непростий інтерфейс		+	
4	Користувацький інтерфейс	Програма реалізована у середовищі програмування	Має власний інтерфейс	Має власний інтерфейс	Вбудовано у Excel	+		

Проаналізуємо реальність технічно здійснити ідею проекту (табл. 4.4).

Таблиця 4.4 – Технологічна здійсненність продукту

№ п/п	Ідея проекту	Технології і реалізації	Наявність технологій	Доступність технологій
1	Створення системи, яка буде комплексно аналізувати ситуацію із досягненням цілей сталого розвитку країнами ООН, точно будувати короткострокові прогнози.	Використання мови програмування Python	Наявні	Доступні
2		Використання мови програмування JavaScript	необхідні допрацювання	Доступні
3		Використання мови програмування R	необхідні допрацювання	Доступні
Обрана технологія реалізації ідеї проекту: Python				

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Проведемо попередній аналіз ринку для запуску стартап-проекту та наведемо у таблиці 4.5.

Таблиця 4.5 – Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники ринку (найменування)	Характеристика
1	Кількість головних гравців, од	2
2	Загальний обсяг продажів	1млн $
3	Динаміка ринку (якісна оцінка)	Позитивна, зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Відсутні
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	11%

Після аналізу ринку можна зробити висновок, що він є сприятливим для створення програмного продукту, оскільки динаміка ринку позитивна, а конкуренти відсутні. Наступним кроком необхідно охарактеризувати основні групи потенційних користувачів продукту і скласти опис вимог кожної такої групи (див. табл. 4.6)

Таблиця 4.6 – Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Кластеризація країн за показниками сталого розвитку	Дослідники, науковці, державні діячі, що зацікавлені у розумінні стану різних країни в питаннях сталого розвитку	Цікавлять проблемні групи країн та особливості сталого розвитку кожної групи	Комплексний опис особливостей кожного кластеру за статистикою про сталий розвиток
2	Прогнозування та контроль показників сталого розвитку	Державні діячі та науковці, що контролюють витрати державного бюджету на досягнення цілей сталого розвитку	Цікавлять прогнози на короткі терміни для конкретної країни	Простота у використанні, точність прогнозування, комплексний опис кожного етапу прогнозування

Тепер необхідно проаналізувати фактори загроз, які можуть завадити успішному запуску стартап-проекту на ринок (див. табл. 4.7).

Таблиця 4.7 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Конкуренція	Можливий вихід на ринок конкуруючих проектів	Знайти та описати найбільші переваги власного продукту, що більше буде цікавий користувачеві
2	Ресурси для моделювання	При подачі на вхід великого обсягу даних можлива нехватка ресурсів комп'ютера для її обробки та отримання хороших результатів	Завчасно правильно розрахувати необхідні технічні ресурси для обробки вхідного масиву даних та, за необхідності, орендувати додаткові ресурси.
3	Ціна збуту	Конкуренти можуть коштувати менше через нижчу якість	Акцентувати увагу на якості та продумати комунікаційну стратегію

Проаналізуємо фактори можливостей розробленого стартап-проекту (див. табл. 4.8).

Таблиця 4.8 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Проста масштабованість	Моделі можуть легко адаптуватися під різні об’єми вхідних датасетів та різні вхідні історичні дані	Швидке розширення можливостей продукту, поліпшення якості моделювання
2	Якість результатів	Надавати найбільш якісні послуги по прогнозуванню та кластеризації	Виконувати порівняльний аналіз побудованих моделей та пропонувати найкращі результати
3	Створення позитивного іміджу	Комунікація із користувачами, опитування з приводу недоліків продукту і їх усунення, забезпечення задоволеності клієнтів	Визначення цільової аудиторії, її потреби, створення якісної рекламної кампанії

Далі розглянемо питання конкуренції, а саме визначимо її тип та рівень. Результати аналізу наведемо у таблиці 4.9.

Таблиця 4.9 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	У чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції: досконала конкуренція	Багато продуктів та експертів	Розробка продукту, що відрізнятиметься від конкурентів, буде кращий по якості і функціоналу
2. За рівнем конкурентної боротьби: міжнародний	Представлені проекти, розроблені у різних країнах	Розширити цільову аудиторію, зробити мультимовність
3. За галузевою ознакою: внутрішньогалузева	Можуть працювати з різними галузями	Покращити масштабованість та персоналізації
4. Конкуренція за видами товарів: товарно-родова	Конкуренція з прогнозами інших систем та експертів	Покращення якості прогнозів та кластеризації, розширити кількість побудованих моделей
5. За характером конкурентних переваг: нецінова	Різні компанії пропонують різну точність	Покращення реалізованих алгоритмів
6. За інтенсивністю: марочна	Існують компанії з сильним брендом	Створення маркетингової стратегії для вибудови нового бренду

Далі необхідно виконаємо аналіз конкуренції за моделлю 5 сил конкуренції Майкла Портера, результати якого зведено у таблиці 4.10

Таблиця 4.10 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти у галузі	Потенційні конкуренти	Постачальники	Клієнти	Товарозамінники
	Інші існуючі системи	Якість прогнозів та кластеризації, ціна, капіталовкладення	Фактори сили постачальників	Контроль якості, порівняння цін, система інформації	Сила бренду, якість, ціна, машстаби
Висновки	В майбутньому можлива інтенсивна конкуренція	Можливості входження на ринок, нові потенційні конкуренти	Постачальники відсутні	Клієнти не впливають на умови роботи на ринку	Товарозамінники відсутні

Маючи результати аналізу конкуренції (таблиця 4.10), характеристики ідеї стартап-проекту (таблиця 4.5), характеристики потенційних клієнтів і їх вимоги до продукту (таблиця 4.6) та фактори ринкового середовища (таблиці 4.7 і 4.8) було сформульовано та обґрунтовано перелік факторів конкурентоспроможності (таблиця 4.11).

Таблиця 4.11 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Якість прогнозів	Система виконує моделювання рядом алгоритмів, виконує підбір оптимальних параметрів та обирає кращі варіанти моделей і прогнозів
2	Масштабованість та гнучкість	Система може пристосовуватись до вхідного набору даних, будувати прогнози по історичним даним
3	Обслуговування	Система технічної підтримки надає рекомендації, відповіді користувачам на незрозумілі моменти у роботі програми

Тепер можна провести аналіз сильних та слабких сторін продукту (табл. 4.12).

Таблиця 4.12 – Порівняльний аналіз сильних та слабких сторін системи

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів						
			-3	-2	-1	0	1	2	3
1	Якість прогнозів	20				+			
2	Масштабованість та гнучкість	17				+			
3	Обслуговування	10		+					

Далі проведемо SWOT-аналіз продукту (табл. 4.13).

Таблиця 4.13 – SWOT-аналіз стартап-проекту

Сильні сторони Якість прогнозів Масштабованість та гнучкість Обслуговування	Слабкі сторони Відсутність бренду Не сформована база клієнтів Система вбудована в дистрибутив Anaconda
Можливості Проста масштабованість Якісні результати Створення позитивного іміджу	Загрози Конкуренція Ресурси Ціна збуту

Завдяки проведенню SWOT-аналізу, ми змогли визначити сильні та слабкі сторони, можливості та загрози, пов'язані з конкуренцією та плануванням стартап-проекту. Далі спроектуємо альтернативну ринкову поведінку для інтеграції стартап-проекту на ринок та приблизний час реалізації системного комплексу, з урахуванням потенційних проектів, що можуть бути виведені на ринок (табл. 4.14).

Таблиця 4.14 – Альтернативи ринкового впровадження стартап проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Вихід на ринок «сирого» продукту з можливістю побудови невеликої кількості моделей, отримання неточних результатів	20%	2 місяці
2	Вихід на ринок готового продукту з обранням найкращих результатів серед множини побудованих моделей, мультимовністю, можливістю побудови прогнозів	60%	6 місяців

Отже, в результаті детального аналізу ринкового та конкурентного середовища, факторів загроз та можливостей, сильних та слабких сторін продукту можна зробити висновок, що на ринку склалися сприятливі умови для впровадження товару і, що даний товар відповідає вимогам користувачів.

4.4 Розроблення ринкової стратегії стартап-проекту

Для розробки ринкової стратегії продукту, у першу чергу, необхідно проаналізувати цільову аудиторію проекту (табл. 4.15).

Таблиця 4.15 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит у межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Дослідники сталого розвитку	Висока	30%	Висока	Висока
2	Державні діячі	Середня	20%	Низька	Середня
3	Науковці у питаннях сталого розвитку	Низька	15%	Середня	Висока
Які цільові групи обрано: 1, 2					

Визначимо базову стратегію розвитку продукту (табл. 4.16).

Таблиця 4.16 – Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	1 та 2	Стратегія недиференційованого маркетингу	Масштабування та якість моделювання	Оптимальних витрат

Для роботи в обраних сегментах ринку сформовано базову стратегію розвитку (таблиці 4.17, 4.18).

Таблиця 4.17 – Визначення базової стратегії конкурентної поведінки

Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
Так	Шукати нових	Ні	Стратегія виклику лідера

Таблиця 4.18 – Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
Висока точність прогнозу Висока якість кластеризації Простота у використанні Масштабованість	Використання передових технологій інтелектуального аналізу даних для кластеризації та прогнозування показників сталого розвитку	Якість прогнозів Масштабованість Обслуговування Універсальність	Найбільш якісні прогнози показників сталого розвитку Легкість використання Якісна кластеризація даних

4.5 Розроблення маркетингової програми стартап-проекту

Сформуємо ключові переваги концепції потенційного товару (таблиця 4.19) та побудувати концепцію маркетингових комунікацій (таблиця 4.20).

Таблиця 4.19 – Ключові переваги концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Якісні короткострокові прогнози	Побудова прогнозів на основі історичних даних про сталий розвиток	Адаптування моделі до вхідних даних, аналіз та підбір оптимальних параметрів моделі, автоматичне покращення та перенавчання моделі
2	Якісна кластериазація даних	Побудова моделей на основі вхідного датасету показників сталого розвитку	Автоматичний підбір параметрів моделі, побудова моделей різної структури та обрання найкращого варіанту кластеризації
3	Проста масштабованість	Побудова прогнозів та кластерів для довільного вхідного датасету	Можливість легко додати у список прогнозів необхідни параметр сталого розвитку, за яким хочемо здійснити кластеризацію або прогнозування

Таблиця 4.20 – Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуютьс я цільові клієнти	Ключові позиції, обрані для позиціону вання	Завдання рекламног о повідомле ння	Концепція рекламного звернення
1	Пошук спеціалізован их систем для прогнозуванн я та кластеризації	Професійні канали комунікації	Точність Якість Зовнішні та внутрішні чинники	Визначити сильні сторони та переваги продукту	Реклама у професійних учасників ринку
2	Пошук доступного та дешевого продукту, що виконує базові задачі кластеризації та прогнозуванн я	Інтернет, форуми, реклами від інфлюенсерів	Простота Дешева ціна Відносна якість	Вселити довіру у бренд та продукт	Реклама у лідерів думок Вивіски в публічних місцях

4.6 Висновки до розділу

У ході виконання четвертого розділу магістерської дисертації була розроблена стратегія виходу на ринок продукту, розроблена маркетингова стратегія. Для цього на початку була сформована інформаційна карта стартап-проекту, описана його основна ідея та визначена технологічна здійсненність.

Також був проведений аналіз конкурентів, визначити сильні, слабкі та нейтральні сторони власного продукту відносно конкурентів.

Другим кроком було визначено ринкові можливості запуску стартап-проекту. Для цього був проведений комплексний аналіз ринку, визначена потенційна цільова аудиторія, фактори можливостей та загроз, проведений SWOT-аналіз продукту.

На наступному етапі було сформовано стратегію стартап-проекту, визначеня ринкова та маркетингова стратегії, визначені базові стратегії виходу на ринок для залучення аудиторії. Після цього було визначено ключові переваги концепції потенційного товару та сформовано концепт маркетингових комунікацій.

Загалом, можемо сказати, що продукт може бути розглянутий як стартап , що спрямований на стратегії голубих океанів (не зайнятого ринку), так як основні конкуренти відсутні. Тому вважаю даний продукт потенційно інвестиційно привабливим, він має можливість вистрілити на ринку та зайняти свою нішу серед потенційної аудиторії.

ВИСНОВКИ ПО РОБОТІ

Дана магістерська робота була присвячена аналізу процесів сталого розвитку на основі статистики показників по країнам ООН. Зокрема, була створена система, яка дозволяє виконувати кластеризацію даних (країн) на відповідні кластери різними методами інтелектуального аналізу даних. На основі побудованих моделей були обчислені критерії якості кластеризації, обрана найкраща модель та проаналізовані отримані результати, описаний кожен кластер, його особливості та відмінності від інших кластерів. Разом з тим, розроблений програмний продукт дозволяє виконувати короткострокові прогнози за допомогою регресійних моделей. Програма написана мовою програмування Python у середовищі Spyder.

Для виконання кластеризації був обраний датасет, який складався з 18 факторів (показників сталого розвитку), кожен з яких відображав досягнення тієї чи іншої країни у відповідній цілі сталого розвитку. Датасет включає в себе інформацію про 137 країн. Для прогнозування були обрані чотири процеси, що вміщують в себе історичні дані про окремі показники сталого розвитку (дані з 1991 по 2019 рік).

У першому розділі була описана актуальність поставленої задачі, описані існуючі методи та підходи до кластеризації та прогнозування даних, описані цілі сталого розвитку

У другому розділі описані математичні моделі, що були обрані для прогнозування та кластеризації, визначені способи обрання кількості кластерів та описані використані критерії, за допомогою яких здійснювалось порівняння реалізованих алгоритмів.

У третьому розділі була обрана функціональна платформа для написання програмного продукту, наведена його функціональна схема. Також наведений опис вхідних даних для кластеризації та прогнозування, після чого були

описані результати кластеризації різними методами інтелектуального аналізу, серед яких метод к-середніх, агломеративна кластеризація, DBSCAN та нечіткий метод Бездека. Далі наведені побудовані регресійні моделі вхідних процесів, побудовані прогнозовані значення на майбутнє. На основі отриманих критеріїв якості кластеризації була обрана найкраща модель та описаний кожен кластер, отриманий цією моделлю.

У четвертому розділі розглянуто роботу над стартап-проектом, який представлений у вигляді системи підтримки прийняття рішень у сфері сталого розвитку. Ця стратегія є стратегією голубого океану, тобто зайняття ніші, що не задіяна в Україні.

У перспективі для подальшого дослідження можуть бути використані інші підходи до кластеризації та прогнозування даних, а саме нейронні мережі, метод групового урахування аргументів, методи кластеризації на основі моделей тощо. Також, у подальшому варто розробити робочий програмний інтерфейс для більш зручної роботи програми, введення даних та отримання результатів. Крім того, слід автоматизувати підбір найкращої моделі для вхідного датасету та автоматизувати підбір деяких параметрів моделей. Інший напрямок проведення майбутніх досліджень – пошук зовнішніх факторів впливу на процеси формування показників сталого розвитку, такі як глобальні пандемії, природні катаклізми, революції тощо. Така інформація надала би можливість будувати більш точні кластери та прогнози.

ПЕРЕЛІК ПОСИЛАНЬ

1. Цілі сталого розвитку та Україна. *Урядовий портал*. URL: https://www.kmu.gov.ua/diyalnist/cili-stalogo-rozvitku-ta-ukrayina (дата звернення: 15.09.2021).
2. Повістка дня в галузі сталого розвитку. *Організація Об'єднаних Націй*. URL: https://www.un.org/ru/documents/decl_conv/conventions/agenda21.shtml (дата звернення: 15.09.2021).
3. Фарниа Л. Статистика по Целям устойчивого развития в Европе. Венеция: Фонд Эни Энрико Маттеи, 2019, 17 с.
4. Глазырина И.П., Михеев И.Е., Егидарев Е.Г., Симонов Е.А. Экологический демпинг в планах развития Сибири и Дальнего Востока. *Эко*. 2012. №10. С. 35-51.
5. Sustainable Development Report 2021. URL: https://dashboards.sdgindex.org/rankings (дата обращения: 20.09.2021).
6. SVM Регресія (приклад). URL: http://www.machinelearning.ru/wiki/index.php?title=SVM_%D1%80%D0%B5%D0%B3%D1%80%D0%B5%D1%81%D1%81%D0%B8%D1%8F_%28%D0%BF%D1%80%D0%B8%D0%BC%D0%B5%D1%80%29 (дата звернення: 22.09.2021).
7. William W.H. Machine learning methods in the environmental sciences. New York: Cambridge University Press, 2009. 345 p.
8. Amit, Y., Geman, D. Shape quantization and recognition with randomized trees. *Neural Computation*. 1997. Vol. 9. P. 1545–1588.
9. Barron A.R. Approximation and estimation bounds for artificieal nearal networks. *Machine learning*. 1994. Vol. 14. P. 115-133.

10. Стрижов В. В. Методы индуктивного порождения регрессионных моделей. Москва: ВЦ РАН, 2008. 61 с.
11. Згуровський М. З., Бідюк П. І., Терентьєв О. М., Просянкіна-Жарова Т. І. Байєсівські мережі в системах підтримки прийняття рішень. Київ: ТОВ «Видавниче Підприємство «Едельвейс», 2015. 300 с.
12. Звягин Л.С. Метод байесовских сетей и ключевые аспекты байесовского моделирования. *Сборник докладов*: материалы XXII Международной конференции по мягким вычислениям и измерениям (SCM-2019), г. Санкт-Петербург, 23-25 мая 2019 р., Санкт-Петербург: СПбГЭТУ «ЛЭТИ», 2019. С. 30-34.
13. Шумейко А.А., Сотник С.Л. Інтелектуальний аналіз даних. Введення в data mining. Дніпро: Біла Е.А., 2012. 212 с.
14. Mercer D. P. Clustering large datasets. Oxford: Linacre college, 2003. 48 p.
15. Зайченко Ю.П., Гончар М.А. Нечеткие методы кластерного анализа в задачах автоматической классификации в экономике. *Вісник НТУУ «КПІ»*, 2007. № 47. С.197-204.
16. Бідюк П. І, Романенко В. Д., Тимощук. О. Л. Аналіз часових рядів. Київ: ВПК "Політехніка", 2013. 599 с.
17. Алгоритмы кластеризации на службе Data Mining. URL: https://loginom.ru/blog/data-mining-clustering (дата звернення: 01.10.2021).
18. Peter J. Rousseeuw. Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis. Fribourg: *Journal of Computational and Applied Mathematics*. 1986. Vol. 20. P. 53–65.
19. Ермолаев О.Ю. Математическая статистика для психологов. Москва: Московський психолого-соціальний інститут, 2003. 336 с.
20. Юрченко М.Є. Прогнозування та аналіз часових рядів. Чернігів: ЧНТУ, 2018. 90 с.
21. Бідюк П. І. Часові ряди: моделювання і прогнозування: монографія. Київ: ЕКМО, 2003. 144 с.

22. Sustainable development report. URL: https://dashboards.sdgindex.org/explorer (дата звернення 15.10.2021).
23. Sustainable development index: Time series (1990-2019). URL: https://www.sustainabledevelopmentindex.org/time-series (дата звернення 20.10.2021)

ДОДАТОК А КОД ПРОГРАМИ

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt


def cm_to_inch(value):
    return value/2.54


def select_cluster(U):
    Clusters = []
    t = 0
    for i in range(137):
        t = 0
        s = U[0][i]
        for j in range(len(U)):
            if (U[j][i] > s):
                t = j
                s = U[j][i]
        Clusters.append(t)
    return Clusters

def criteria(U):
    s = 0
    for i in range (len(U)):
        for j in range(137):
            s = s + U[i][j]*U[i][j]
    return s

def max_elem (y):
    t=y[0]
    for i in range(len(y)):
        if (y[i]>t): t=y[i]
    return t

def wss_cr (X,centroids,y):
```

```
    s = 0
    for i in range(len(X)):
        t = y[i]
        s = s + (LA.norm(X[i]-centroids[t]))*(LA.norm(X[i]-centroids[t]))
    return s



df = pd.read_excel('d://data_diplom.xlsx')
country = df.iloc[:,0]

X=df.iloc[:,1:].values

#kmeans---------------------------------

from sklearn.cluster import KMeans

#elbow method----------------------------------------

wcss = []
for i in range (1,21):
    kmeans=KMeans(n_clusters=i,init='k-means++',max_iter=300,n_init=10,random_state=0)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)


plt.plot(range(1,21),wcss)
plt.title('The elbow method')
plt.xlabel('number of clusters')
plt.ylabel('wcss')
plt.show()

kmeans_5 = KMeans(n_clusters=5,init='k-means++',max_iter=300,n_init=10,random_state=0)
y_kmeans_5 = kmeans_5.fit_predict(X)
centroids_5 = kmeans_5.cluster_centers_
result_kmeans5=pd.DataFrame(data=y_kmeans_5,index=country)
result_kmeans5.to_excel('D://result1_disser_kmeans5.xlsx')

kmeans_6 = KMeans(n_clusters=6,init='k-means++',max_iter=300,n_init=10,random_state=0)
y_kmeans_6 = kmeans_6.fit_predict(X)
```

```
centroids_6 = kmeans_6.cluster_centers_
result_kmeans6=pd.DataFrame(data=y_kmeans_6,index=country)
result_kmeans6.to_excel('D://result1_disser_kmeans6.xlsx')

kmeans_7 = KMeans(n_clusters=7,init='k-means++',max_iter=300,n_init=10,random_state=0)
y_kmeans_7 = kmeans_7.fit_predict(X)
centroids_7 = kmeans_7.cluster_centers_
result_kmeans7=pd.DataFrame(data=y_kmeans_7,index=country)
result_kmeans7.to_excel('D://result1_disser_kmeans7.xlsx')

#kmeans end-------------------------

#Agglomerative clustering

import scipy.cluster.hierarchy as sch
from sklearn.cluster import AgglomerativeClustering

fig = plt.figure(figsize=(16,8))
dendrogram = sch.dendrogram(sch.linkage(X,method='ward'))

plt.title('Dendrogram')
plt.xlabel('Countries')
plt.ylabel('Eucledian distance')
plt.show()

hc5 = AgglomerativeClustering(n_clusters = 5, affinity='euclidean',linkage='ward')
y_hc5 = hc5.fit_predict(X)
result_hc5 = pd.DataFrame(data = y_hc5, index = country)
result_hc5.to_excel('D://result_aggl5.xlsx')

hc6 = AgglomerativeClustering(n_clusters = 6, affinity='euclidean',linkage='ward')
y_hc6 = hc6.fit_predict(X)
result_hc6 = pd.DataFrame(data = y_hc6, index = country)
result_hc6.to_excel('D://result_aggl6.xlsx')

hc7 = AgglomerativeClustering(n_clusters = 7, affinity='euclidean',linkage='ward')
y_hc7 = hc7.fit_predict(X)
result_hc7 = pd.DataFrame(data = y_hc7, index = country)
result_hc7.to_excel('D://result_aggl7.xlsx')
```

```
from sklearn.neighbors.nearest_centroid import NearestCentroid
clf = NearestCentroid()

clf.fit(X, y_hc5)
centr_hc5=clf.centroids_

clf.fit(X,y_hc6)
centr_hc6 = clf.centroids_

clf.fit(X,y_hc7)
centr_hc7 = clf.centroids_


#DBSCAN

from sklearn.cluster import DBSCAN

dbscan = DBSCAN()
clusters = dbscan.fit_predict(X)

from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaler.fit(X)
X_scaled = scaler.transform(X)


from sklearn.neighbors import NearestNeighbors

neighbors = NearestNeighbors(n_neighbors=36)
neighbors_fit = neighbors.fit(X_scaled)
distances, indices = neighbors_fit.kneighbors(X_scaled)

distances = np.sort(distances, axis=0)
distances = distances[:,1]
plt.plot(distances)

dbscan = DBSCAN(min_samples=36, eps = 3.19)
clusters = dbscan.fit_predict(X_scaled)
```

```
result_dbscan = pd.DataFrame(data = clusters, index = country)
result_dbscan.to_excel('D://result_dbscan.xlsx')


#fuzzy-clustering ----------------------------------


# K++ initialization Algorithm:

import random
import scipy
import numpy.linalg as LA

def initialize(X, K):
    C = [X[0]]
    for k in range(1, K):
        D2 = scipy.array([min([scipy.inner(c-x,c-x) for c in C]) for x in X])
        probs = D2/D2.sum()
        cumprobs = probs.cumsum()
        np.random.seed(20)          # fixxing seeds
        #random.seed(0)             # fixxing seeds
        r = scipy.rand()
        for j,p in enumerate(cumprobs):
            if r < p:
                i = j
                break
        C.append(X[i])
    return C

a = initialize(X,6)

# Now the Fuzzy c means algorithm:
m = 2    # Fuzzy parameter (it can be tuned)
r = (2/(m-1))

# Initial centroids:
c1,c2,c3,c4,c5,c6 = a[0],a[1],a[2],a[3],a[4],a[5]

# prepare empty lists to add the final centroids:
cc1,cc2,cc3,cc4,cc5,cc6 = [],[],[],[],[],[]
```

```
n_iterations = 10000

for j in range(n_iterations):
    u1,u2,u3,u4,u5,u6 = [],[],[],[],[],[]

    for i in range(len(X)):
        # Distances (of every point to each centroid):
        a = LA.norm(X[i]-c1)
        b = LA.norm(X[i]-c2)
        c = LA.norm(X[i]-c3)
        d = LA.norm(X[i]-c4)
        e = LA.norm(X[i]-c5)
        f = LA.norm(X[i]-c6)

        # Pertenence matrix vectors:
        U1 = 1/(1 + (a/b)**r + (a/c)**r + (a/d)**r + (a/e)**r + (a/f)**r)
        U2 = 1/((b/a)**r + 1 + (b/c)**r + (b/d)**r + (b/e)**r + (b/f)**r)
        U3 = 1/((c/a)**r + (c/b)**r + 1 + (c/d)**r + (c/e)**r + (c/f)**r)
        U4 = 1/((d/a)**r + (d/b)**r + (d/c)**r + 1 + (d/e)**r + (d/f)**r)
        U5 = 1/((e/a)**r + (e/b)**r + (e/c)**r + (e/d)**r + 1 + (e/f)**r)
        U6 = 1/((f/a)**r + (f/b)**r + (f/c)**r + (f/d)**r + (f/e)**r + 1)

        # We will get an array of n row points x K centroids, with their degree of pertenence
        u1.append(U1)
        u2.append(U2)
        u3.append(U3)
        u4.append(U4)
        u5.append(U5)
        u6.append(U6)

    # now we calculate new centers:
    c1 = (np.array(u1)**2).dot(X) / np.sum(np.array(u1)**2)
    c2 = (np.array(u2)**2).dot(X) / np.sum(np.array(u2)**2)
    c3 = (np.array(u3)**2).dot(X) / np.sum(np.array(u3)**2)
    c4 = (np.array(u4)**2).dot(X) / np.sum(np.array(u4)**2)
    c5 = (np.array(u5)**2).dot(X) / np.sum(np.array(u5)**2)
    c6 = (np.array(u6)**2).dot(X) / np.sum(np.array(u6)**2)
```

```
    cc1.append(c1)
    cc2.append(c2)
    cc3.append(c3)
    cc4.append(c4)
    cc5.append(c5)
    cc6.append(c6)

    if (j>5):
        change_rate1 = np.sum(3*cc1[j] - cc1[j-1] - cc1[j-2] - cc1[j-3])/3
        change_rate2 = np.sum(3*cc2[j] - cc2[j-1] - cc2[j-2] - cc2[j-3])/3
        change_rate3 = np.sum(3*cc3[j] - cc3[j-1] - cc3[j-2] - cc3[j-3])/3
        change_rate4 = np.sum(3*cc4[j] - cc4[j-1] - cc4[j-2] - cc4[j-3])/3
        change_rate5 = np.sum(3*cc5[j] - cc5[j-1] - cc5[j-2] - cc5[j-3])/3
        change_rate6 = np.sum(3*cc6[j] - cc6[j-1] - cc6[j-2] - cc6[j-3])/3
        change_rate =
np.array([change_rate1,change_rate2,change_rate3,change_rate4,change_rate5,change_rate6])
        changed = np.sum(change_rate>0.0000001)
        if changed == 0:
            break

print(c1)  # to check a centroid coordinates   c1 - c5 ... they are the last centroids calculated, so supposedly they
converged.
print(U1)  # this is the degree of pertenence to each centroid (so n row points x K centroids columns).


U=[u1,u2,u3,u4,u5,u6]
criteria_for_6 = criteria(U)
result = pd.DataFrame(data = select_cluster(U), index = country)
result[1] = u1
result[2] = u2
result[3] = u3
result[4] = u4
result[5] = u5
result[6] = u6
result.to_excel('D://result_fuzzy_6.xlsx')


a = initialize(X,5)

# Now the Fuzzy c means algorithm:
```

```
m = 2    # Fuzzy parameter (it can be tuned)
r = (2/(m-1))

# Initial centroids:
c1,c2,c3,c4,c5 = a[0],a[1],a[2],a[3],a[4]

# prepare empty lists to add the final centroids:
cc1,cc2,cc3,cc4,cc5 = [],[],[],[],[]


n_iterations = 10000

for j in range(n_iterations):
    u1,u2,u3,u4,u5 = [],[],[],[],[]

    for i in range(len(X)):
        # Distances (of every point to each centroid):
        a = LA.norm(X[i]-c1)
        b = LA.norm(X[i]-c2)
        c = LA.norm(X[i]-c3)
        d = LA.norm(X[i]-c4)
        e = LA.norm(X[i]-c5)


        # Pertenence matrix vectors:
        U1 = 1/(1 + (a/b)**r + (a/c)**r + (a/d)**r + (a/e)**r  )
        U2 = 1/((b/a)**r + 1 + (b/c)**r + (b/d)**r + (b/e)**r  )
        U3 = 1/((c/a)**r + (c/b)**r + 1 + (c/d)**r + (c/e)**r  )
        U4 = 1/((d/a)**r + (d/b)**r + (d/c)**r + 1 + (d/e)**r  )
        U5 = 1/((e/a)**r + (e/b)**r + (e/c)**r + (e/d)**r + 1  )


        # We will get an array of n row points x K centroids, with their degree of pertenence
        u1.append(U1)
        u2.append(U2)
        u3.append(U3)
        u4.append(U4)
        u5.append(U5)
```

```
    # now we calculate new centers:
    c1 = (np.array(u1)**2).dot(X) / np.sum(np.array(u1)**2)
    c2 = (np.array(u2)**2).dot(X) / np.sum(np.array(u2)**2)
    c3 = (np.array(u3)**2).dot(X) / np.sum(np.array(u3)**2)
    c4 = (np.array(u4)**2).dot(X) / np.sum(np.array(u4)**2)
    c5 = (np.array(u5)**2).dot(X) / np.sum(np.array(u5)**2)


    cc1.append(c1)
    cc2.append(c2)
    cc3.append(c3)
    cc4.append(c4)
    cc5.append(c5)

    if (j>5):
        change_rate1 = np.sum(3*cc1[j] - cc1[j-1] - cc1[j-2] - cc1[j-3])/3
        change_rate2 = np.sum(3*cc2[j] - cc2[j-1] - cc2[j-2] - cc2[j-3])/3
        change_rate3 = np.sum(3*cc3[j] - cc3[j-1] - cc3[j-2] - cc3[j-3])/3
        change_rate4 = np.sum(3*cc4[j] - cc4[j-1] - cc4[j-2] - cc4[j-3])/3
        change_rate5 = np.sum(3*cc5[j] - cc5[j-1] - cc5[j-2] - cc5[j-3])/3
        change_rate = np.array([change_rate1,change_rate2,change_rate3,change_rate4,change_rate5])
        changed = np.sum(change_rate>0.0000001)
        if changed == 0:
            break

print(c1)  # to check a centroid coordinates   c1 - c5 ... they are the last centroids calculated, so supposedly they converged.
print(U1)  # this is the degree of pertenence to each centroid (so n row points x K centroids columns).


U=[u1,u2,u3,u4,u5]
criteria_for_5 = criteria(U)
result = pd.DataFrame(data = select_cluster(U), index = country)
result[1] = u1
result[2] = u2
result[3] = u3
result[4] = u4
result[5] = u5
result.to_excel('D://result_fuzzy_5.xlsx')
```

```
a = initialize(X,7)

# Now the Fuzzy c means algorithm:
m = 2    # Fuzzy parameter (it can be tuned)
r = (2/(m-1))

# Initial centroids:
c1,c2,c3,c4,c5,c6,c7 = a[0],a[1],a[2],a[3],a[4],a[5],a[6]

# prepare empty lists to add the final centroids:
cc1,cc2,cc3,cc4,cc5,cc6,cc7 = [],[],[],[],[],[],[]


n_iterations = 10000

for j in range(n_iterations):
    u1,u2,u3,u4,u5,u6,u7 = [],[],[],[],[],[],[]

    for i in range(len(X)):
        # Distances (of every point to each centroid):
        a = LA.norm(X[i]-c1)
        b = LA.norm(X[i]-c2)
        c = LA.norm(X[i]-c3)
        d = LA.norm(X[i]-c4)
        e = LA.norm(X[i]-c5)
        f = LA.norm(X[i]-c6)
        g = LA.norm(X[i]-c7)

        # Pertenence matrix vectors:
        U1 = 1/(1 + (a/b)**r + (a/c)**r + (a/d)**r + (a/e)**r + (a/f)**r + (a/g)**r)
        U2 = 1/((b/a)**r + 1 + (b/c)**r + (b/d)**r + (b/e)**r + (b/f)**r + (b/g)**r)
        U3 = 1/((c/a)**r + (c/b)**r + 1 + (c/d)**r + (c/e)**r + (c/f)**r + (c/g)**r)
        U4 = 1/((d/a)**r + (d/b)**r + (d/c)**r + 1 + (d/e)**r + (d/f)**r + (d/g)**r)
        U5 = 1/((e/a)**r + (e/b)**r + (e/c)**r + (e/d)**r + 1 + (e/f)**r + (e/g)**r)
        U6 = 1/((f/a)**r + (f/b)**r + (f/c)**r + (f/d)**r + (f/e)**r + 1 + (f/g)**r)
        U7 = 1/((g/a)**r + (g/b)**r + (g/c)**r + (g/d)**r + (g/e)**r + (g/f)**r + 1)

        # We will get an array of n row points x K centroids, with their degree of pertenence
        u1.append(U1)
        u2.append(U2)
```

```
        u3.append(U3)
        u4.append(U4)
        u5.append(U5)
        u6.append(U6)
        u7.append(U7)

    # now we calculate new centers:
    c1 = (np.array(u1)**2).dot(X) / np.sum(np.array(u1)**2)
    c2 = (np.array(u2)**2).dot(X) / np.sum(np.array(u2)**2)
    c3 = (np.array(u3)**2).dot(X) / np.sum(np.array(u3)**2)
    c4 = (np.array(u4)**2).dot(X) / np.sum(np.array(u4)**2)
    c5 = (np.array(u5)**2).dot(X) / np.sum(np.array(u5)**2)
    c6 = (np.array(u6)**2).dot(X) / np.sum(np.array(u6)**2)
    c7 = (np.array(u7)**2).dot(X) / np.sum(np.array(u7)**2)

    cc1.append(c1)
    cc2.append(c2)
    cc3.append(c3)
    cc4.append(c4)
    cc5.append(c5)
    cc6.append(c6)
    cc7.append(c7)

    if (j>5):
        change_rate1 = np.sum(3*cc1[j] - cc1[j-1] - cc1[j-2] - cc1[j-3])/3
        change_rate2 = np.sum(3*cc2[j] - cc2[j-1] - cc2[j-2] - cc2[j-3])/3
        change_rate3 = np.sum(3*cc3[j] - cc3[j-1] - cc3[j-2] - cc3[j-3])/3
        change_rate4 = np.sum(3*cc4[j] - cc4[j-1] - cc4[j-2] - cc4[j-3])/3
        change_rate5 = np.sum(3*cc5[j] - cc5[j-1] - cc5[j-2] - cc5[j-3])/3
        change_rate6 = np.sum(3*cc6[j] - cc6[j-1] - cc6[j-2] - cc6[j-3])/3
        change_rate7 = np.sum(3*cc7[j] - cc7[j-1] - cc7[j-2] - cc7[j-3])/3
        change_rate =
np.array([change_rate1,change_rate2,change_rate3,change_rate4,change_rate5,change_rate6,change_rate7])
        changed = np.sum(change_rate>0.0000001)
        if changed == 0:
            break

print(c1)  # to check a centroid coordinates   c1 - c5 ... they are the last centroids calculated, so supposedly they
converged.
print(U1)  # this is the degree of pertenence to each centroid (so n row points x K centroids columns).
```

```
U=[u1,u2,u3,u4,u5,u6,u7]
criteria_for_7 = criteria(U)
result = pd.DataFrame(data = select_cluster(U), index = country)
result[1] = u1
result[2] = u2
result[3] = u3
result[4] = u4
result[5] = u5
result[6] = u6
result[7] = u7
result.to_excel('D://result_fuzzy_7.xlsx')

#fuzzy-clustering end --------------------------------


from sklearn.metrics import silhouette_score

#results
print(wss_cr (X,centroids_5,y_kmeans_5))
print(silhouette_score(X, y_kmeans_5))

print(wss_cr (X,centroids_6,y_kmeans_6))
print(silhouette_score(X, y_kmeans_6))

print(wss_cr (X,centroids_7,y_kmeans_7))
print(silhouette_score(X, y_kmeans_7))

print(wss_cr (X,centr_hc5,y_hc5))
print(silhouette_score(X, y_hc5))

print(wss_cr (X,centr_hc6,y_hc6))
print(silhouette_score(X, y_hc6))

print(wss_cr (X,centr_hc7,y_hc7))
print(silhouette_score(X, y_hc7))

centers = [c1,c2,c3,c4,c5]
print(wss_cr (X,centers,select_cluster(U)))
```

```
print(silhouette_score(X, select_cluster(U)))

centers_fuzzy_6 = [c1,c2,c3,c4,c5,c6]
print(wss_cr (X,centers_fuzzy_6,select_cluster(U)))
print(silhouette_score(X, select_cluster(U)))

centers_fuzzy_7 = [c1,c2,c3,c4,c5,c6,c7]
print(wss_cr (X,centers_fuzzy_7,select_cluster(U)))
print(silhouette_score(X, select_cluster(U)))

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.vector_ar.var_model import VAR
import statsmodels.api as sm
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.graphics.tsaplots import plot_pacf
from statsmodels.stats.stattools import durbin_watson
from sklearn.metrics import r2_score
import math

class PACF(object):
    def __init__(self, time_s):
        self.time_s = np.array(time_s)
        self.pacf_results = {}
        self.time_s_mean = self.time_s.mean()
        self.time_s_var = self.count_var()

    def count_var(self):
        return  ((self.time_s - self.time_s_mean)**2).sum() / (self.time_s.shape[0] - 1)

    def count_r(self, s):
        to_div = sum( (self.time_s[i] - self.time_s_mean) * (self.time_s[i-s] - self.time_s_mean) for i in range(s,
len(self.time_s)))
        divider = (self.time_s_var)*(len(self.time_s) - 1)
        return to_div / divider

    def F(self, k,j):

        if (k,j) in self.pacf_results.keys():
```

```
            return self.pacf_results[(k,j)]


        if k == j == 1 :
            f = self.count_r(1)
        else:
            to_div = self.count_r(k) - sum(self.F(k - 1, i) * self.count_r(k - i) for i in range(1, k))
            divider = 1 - sum(self.F(k - 1, i) * self.count_r(i) for i in range(1, k))

            f = to_div/divider

        if k == j :
            self.pacf_results[(k, j)] = f
            return self.pacf_results[(k, j)]
        else:
            self.pacf_results[(k, j)] = self.F(k - 1, j) - f * self.F(k - 1, k - j)
            return self.pacf_results[(k, j)]

    def simmetric_F(self, k):
        return self.F(k,k)

def autoregressive_order(series):
    pacf_object = PACF(series)
    pacf_results = [pacf_object.simmetric_F(i) for i in range(10)]
    print ('PACF: \n')
    for i in range (10):
        print (pacf_results[i])
    sm.graphics.tsa.plot_pacf(series, lags = 10)
    p = -1
    for i in range (10):
        if (abs(pacf_results[i]) > 0.2):
            p = i
    print ('Доцільний порядок авторегресії = ', p)
    return p

def ssr(resid):
    squared_resid = (resid)**2
    return  sum(squared_resid)
```

```
def results_of_modeling_AR(params,y_true,y_pred,resid,p):
    print('Рівняння моделі авторегресії порядку',p,':\n')
    print('y(k)=',params[0]*(1-(sum(params)-params[0])),'+')
    for i in range(len(params)-1):
        if (i <= p): print(params[i + 1],'y( k -',i + 1,')')
        else:  print('+',params[i + 1],'ma( k-',i + 1,')')
    print('Параметри адекватності моделі:')
    print('SSR = ', ssr(resid))
    print('DW = ', durbin_watson(resid) )
    if (len(params) == p+1) : print('R Squared = ', r2_score(y_true[p:],y_pred))
    else: print('R Squared = ', r2_score(y_true[1:],y_pred))



def Theil(y_true, y_pred):
    y_true_2 = []
    y_pred_2 = []
    for i in range(len(y_true)):
        y_true_2.append(y_true[i]**2)
        y_pred_2.append(y_pred[i]**2)
    return rmse(y_true,
y_pred)/(math.sqrt((1/len(y_true))*sum(y_true_2))+math.sqrt((1/len(y_true))*sum(y_pred_2)))

def MAPE(y_true, y_pred):
    resid = y_true - y_pred
    return ((1/len(y_true))*(sum(abs(resid)/abs(y_true)))*100)


def MAPE1(y_true, y_pred):
    resid = []
    for i in range(len(resid)):
        resid.append(abs(y_true[i]-y_pred[i]))
    y_true_abs=[]
    for i in range(len(y_true)):
        y_true_abs.append(abs(y_true[i]))
    t = []
    for i in range(len(resid)):
        t.append(resid[i]/y_true_abs[i])
    return ((1/len(y_true))*(sum(t))*100)
```

```
def rmse(y_true, y_pred):
    squared_resid = []
    for i in range(len(y_true)):
        squared_resid.append((y_true[i]-y_pred[i])* (y_true[i]-y_pred[i]))
    return math.sqrt((1/len(y_true))*sum(squared_resid))



def predictions_AR(train, test, model_order):
    temp = [x for x in train]
    pred = []
    predict = []
    for t in range(len(test)):
        model = sm.tsa.ARMA(temp,order=(model_order,0))
        model_fitted = model.fit(method='css')
        output = model_fitted.forecast()
        yhat = output[0]
        pred.append(yhat)
        ik = test.values
        obs = ik[t]
        temp.append(obs)
        print('predicted=%f, true=%f' % (yhat, obs))
    plt.plot(test.values, label = 'True')
    plt.plot(pred, label = 'Predictions')
    plt.legend()
    plt.show()
    for i in range(len(pred)):
        predict.append(pred[i][0])
    print('Якість прогнозу:')
    print ('RMSE = ',rmse(test.values,predict))
    print ('MAPE = ',MAPE(test.values,predict))
    print ('Theil = ',Theil(test.values,predict))
    return predict


dataframe_sdi = pd.read_excel('d://SDI_TR.xlsx')
dataframe_life = pd.read_excel('d://life.xlsx')
dataframe_gni = pd.read_excel('d://gni.xlsx')
```

```
dataframe_co2 = pd.read_excel('d://co2.xlsx')
dataframe_footprint = pd.read_excel('d://footprint.xlsx')
dataframe2 = pd.read_excel('d://SDI.xlsx')

years = dataframe_sdi.iloc[:,0]
countries = dataframe2.iloc[:,1]

dataframe_test = pd.DataFrame(index = years)

country_ = "United Kingdom"


dataframe_test[0] = dataframe_sdi[country_].values
dataframe_test[0].plot(title = 'SDI')
dataframe_test[1] = dataframe_life[country_].values
dataframe_test[1].plot(title = 'Life Expectancy (years)')
dataframe_test[2] = dataframe_gni[country_].values
dataframe_test[2].plot(title = 'Income (GNI per capita const 20')
dataframe_test[3] = dataframe_co2[country_].values
dataframe_test[3].plot(title = 'CO2 emissions per capita (tonne)')
#dataframe_test[4] = dataframe_footprint[country_].values
#dataframe_test[4].plot(title = 'Material Footprint per capita')

split = len(dataframe_test) - 3
X_train = dataframe_test[0:split]
X_test = dataframe_test[split:len(dataframe_test)]

for i in range (4):
    series = dataframe_test[i]
    plot_pacf(X_train[i], lags = 10)



p = []
for i in range(4):
    p.append(autoregressive_order(X_train[i]))


X_train = X_train.astype('float32')
#AR(1)
```

```
for i in range(5):

    model = sm.tsa.ARMA(X_train[i],order=(p[i],0))
    model_fitted = model.fit(method='css')
    plt.plot(X_train[i],label = 'true')
    plt.plot(model_fitted.predict(), label='predicted')
    plt.legend()
    plt.show()
    results_of_modeling_AR(model_fitted.params,X_train[i],model_fitted.predict(),model_fitted.resid,p[i])

    #Результати виконання прогнозування AP(1)
    pred1AR1 = predictions_AR(X_train[i], X_test[i], p[i])




    model = sm.tsa.ARMA(series,order=(p[0],0))
    model_fitted = model.fit(method='css')
    plt.plot(X_train[i],label = 'true')
    plt.plot(model_fitted.predict(), label='predicted')
    plt.legend()
    plt.show()
    results_of_modeling_AR(model_fitted.params,X_train[i],model_fitted.predict(),model_fitted.resid,p[i])

    #Результати виконання прогнозування AP(1)
    pred1AR1 = predictions_AR(series, X_test[i], p[i])

test = X_test[0]
```