НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
КАФЕДРА ІНФОРМАТИКИ ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

КУРСОВА РОБОТА

з дисципліни «Аналіз даних в інформаційних системах»

на тему: «Аналіз даних про захворюваність на COVID-19. Частина 2: Кластеризація та класифікація країн»

Студента 2 курсу ІТ-03 групи
Спеціальності: 121
«Інженерія програмного забезпечення»
Цуканової Марини Сергіївни

«ПРИЙНЯВ» з оцінкою

_____

доц. Ліхоузова Т.А. / доц. Олійник Ю.О.

_____
Підпис Дата

Київ - 2022 рік

Національний технічний університет України “КПІ ім. Ігоря Сікорського”

Кафедра інформатики та програмної інженерії

Дисципліна Аналіз даних в інформаційно-управляючих системах

Спеціальність 121 "Інженерія програмного забезпечення"

Курс 2 Група ІТ-03 Семестр 4

ЗАВДАННЯ

на курсову роботу студента

Цуканової Марини Сергіївни

1.Тема роботи Аналіз даних про захворюваність на COVID-19. Кластеризація та класифікація країн

2.Строк здачі студентом закінченої роботи 19.06.2022

3. Вхідні дані до роботи методичні вказівки до курсової робота, обрані дані з сайту
https://ourworldindata.org/coronavirus
https://docs.owid.io/projects/covid/en/latest/
https://github.com/CSSEGISandData/COVID-19

4.Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробці)

1.Постановка задачі
2.Аналіз предметної області
3.Розробка сховища даних
4.Інтелектуальний аналіз даних

5.Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

6.Дата видачі завдання 16.04.2022

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів курсової роботи	Термін виконання етапів роботи	Підписи керівника, студента
1.	Отримання теми курсової роботи	16.04.2022	
2.	Визначення зовнішніх джерел даних	16.05.2022	
3.	Пошук та вивчення літератури з питань курсової роботи	16.05.2022	
4.	Вибір формату зберігання даних	20.05.2022	
5.	Обґрунтування методів інтелектуального аналізу даних	21.05.2022	
6.	Застосування та порівняння ефективності методів інтелектуального аналізу даних	01.06.2022	
7.	Підготовка пояснювальної записки	10.06.2022	
8.	Здача курсової роботи на перевірку	19.06.2022	
9.	Захист курсової роботи	21.06.2022	

Студент _____ Цуканова М.С.

(підпис) (прізвище, ім'я, по батькові)

Керівник _____ доц. Ліхоузова Т.А

(підпис) (прізвище, ім'я, по батькові)

Керівник _____ доц. Олійник Ю.О.

(підпис) (прізвище, ім'я, по батькові)

"26" червня 2022

АНОТАЦІЯ

Пояснювальна записка до курсової роботи: 45 сторінок, 13 рисунків, 15 таблиць, 8 посилань.

Об’єкт дослідження: інтелектуальний аналіз даних.

Предмет дослідження: створення програмного забезпечення, що проводить аналіз даних та їх кластеризація з подальшим графічним відображенням результатів.

Мета роботи: проектування та реалізація сховища даних, а також реалізація програмного забезпечення для отримання даних зі сховища та їх подальшого аналізу.

Дана курсова робота включає в себе: опис проектування, створення та заповнення сховища даних за даною задачею за допомогою фізичної моделі бази даних, опис створення програмного забезпечення для інтелектуального аналізу даних та їх графічного відображення.

ЗМІСТ

ВСТУП

Останні 2 роки Україна як і весь світ переживає пандемію COVID-19, за цей час людство змогло стати свідком того, як хвороба швидкими темпами розповсюдюється світом та як швидко винаходиться вакцина для захисту від вірусу. Ведуться численні досліження щодо зв’язку між рівнем життя в країні до певної епідеї або пандемії та швидкістю розповсюдження хвороби, показників смертності та захворюваності підчас розпалу хвороби.

В цій курсовій роботі ми спробуємо самостійно знайти зв’язок між рівнем життя в країні та перетіком коронавірусної інфекції там, використовуючи методи інтелектуального аналізу. В якості методу інтелектуального аналізу даних було обрано кластеризацію, в результаті якої ми зможемо отримами визначено кількість груп, що об’єднує країни за деякими ознаками. За допомогою візуалізації та текстового виводу результатів ми зможемо проаналізувати яким чином країни були поділені на кластери кожним методом та чи впливає рівень життя в країні на поділ на кластери.

В ході роботи будуть сформовані та перевірені гіпотези про взаємозв’язки між факторами, також результати будуть агреговані та представлені графічно, для зручного перегляду та аналізу на вже більш високому рівні, для формулювання висновків.

Для реалізації поставленої задачі буде використано мову програмування Python3.8 та бібліотеки NumPy, Pandas, Matplotlib та Sklearn. Середа розробки – PyCharm.

1.ПОСТАНОВКА ЗАДАЧІ ДО ЧАСТИНИ 2

Під час виконання курсової роботи необхідно виконати наступні завдання:

- Обробити датасети та створити датафрейми згідно до обраних умов
- Дослідити зв’язки між полями створених датафреймів
- Обрати та виконати кластеризацію обраного датафрейму трьома способами
    - Перший метод кластеризації – метод K-середніх
    - Другий метод кластеризації – DBSCAN кластеризація
    - Третій метод кластеризації – ієрархічна кластеризація
- Проаналізувати отримані результати та зробити висновки щодо алгоритмів кластеризації, проаналізувати їх ефективність для обраного датасету
- Візуалізувати отримані результати за допомогою таблиць та графіків, створених за допомогою бібліотек мови Python
- Зробити висновки щодо зв’язку захворюваності на COVID-19 та загального рівня життя в країні

2.АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Виокремлюючи пандемію як найбільш гучне явище 2019-2022 років, людям потрібно розуміти її наслідки для світу та нашої країни в цілому. Ця тема є дуже широкою, адже до питання наслідків можна підходити з різних сторін. З одного боку, нас можуть цікавити загальні числа захворюваності та смертності, та як країни поділяються на групи за їх характеристиками, та як вони відреагували на вірус, що і є темою цієї частини та курсової роботи в цілому.

З іншого боку, нас можуть цікавити вплив вже існуючих умов та рівня життя в країнах на перетік захворюваності та вплив на кількість випадків хвороби та смертність від коронавірусної інфекції. Як бачимо, ця тема є дійсно широкою та потребує значного проміжку часу для детального аналізу та спостереження.

В рамках даної курсової роботи основними сутностями в аналізі даної проблеми будуть: захворюваність та смертність в країні, додаткові фактори, що характеризують перебіг коронавірусної інфекції та зовнішні фактори, що характеризують якість життя в країні.

У програмному забезпеченні буде реалізовано наступну функціональність, що включає в себе:

- Завантаження даних;
- Створення процесів обробки та очищення для завантажених даних;
- Формулювання гіпотез про взаємозв’язки між даними;
- Безпосередній аналіз даних обраними методами інтелектуального аналізу даних
- Графічне відображення отриманих результатів та їх аналіз;
- Перевірка сформульованих гіпотез;
- Підбиття підсумків.

3. ВИБІР ФОРМАТУ ЗБЕРІГАННЯ ДАНИХ

Для виконання курсової роботи було обрано 2 датасети відкритих даних. Перший датасет з сайту https://docs.owid.io/projects/covid/en/latest/ включає велику кількість різноманітної інформації щодо розповсюдження COVID-19 та загальної інформації про країни, в яких було зафіксовано вірус SARS-CoV-2. Розмір датасету (190426, 67). Інформація збережена в датасеті датується від 2020-02-24 до 2022-05-30.

Другий датасет з сайту https://github.com/CSSEGISandData/COVID-19 є датасетом представленим John Hopkins University. Цей датасет має інформацію щодо рівня щастя та його складових в різних країнах. Розмір датасету (156, 9).

На основі проведеного аналізу предметної області було розроблено сховище даних, що складається з:

- Двох початкових датасетів
- Чотирьох датафреймів
  - corona_dataset_clear – датафрейм, що має в основі датасет owid-covid-data, але тільки зі значеннями, які були обрані для аналізу.
  - corona_dataset_aggregated – датафрейм, створений з датафрейму corona_dataset_aggregated, але кожній країні відноситься лише один рядок.
  - happiness_aggregated – датафрейм, що має в основі датасет world-happiness-report.csv, але тільки зі значеннями, що були обрані для аналізу.
  - df_plus_happiness – датафрейм, над яким виконуються всі операції з аналізу даних, створений шляхом конкатенації датасетів corona_dataset_aggregated та happiness_aggregated.

В таблиці 3.1.1 наведено частину опису полів першого датасету (owid-covid-data). Оскільки цей датасет має велику кількість полів, то опис всіх полів знаходиться в Додатку Б. Таблиця 1. В таблиці 3.1.2 наведено опис полів другого датасету (world-happiness-report).

Таблиця 3.1.1 – Таблиця датасету owid-covid-data (частина)

Номер поля	Назва поля	Опис поля	Допустимі значення
1	iso_code	Код країни	any string
2	continent	Континент розташування країни	any string
3	location	Назва країни	any string
4	date	Дата спостереження	date year-month-day
5	total_cases	К-ть підтверджених випадків захворювання	any float
8	total_deaths	К-ть смертей за весь період	any float
11	total_cases_per_million	К-ть захворювань на мільйон осіб	any float
14	total_deaths_per_million	К-ть смертей з мільйона осіб	any float
17	reproduction_rate	Швидкість розповсюдження хвороби	any float
48	stringency_index	Індекс жорсткості*	any float
49	population	К-ть населення країни	any float
50	population_density	Щільність населення	any float
51	median_age	Середній вік населення	any float
62	life_expectancy	Ймовірна тривалість життя	any float
63	human_development_index	Індекс людського розвитку**	any float

*Індекс жорсткості розраховується за 9ма параметрами, такими як закриття шкіл, закриття робочих місць, скасування культурних подій, заборона суспільних зборів, закриття громадського транспорту, необхідність залишатися вдома, кампанії з освіти громадян, заборона переміщень всередині країни та контроль закордонних переміщень.

**Індекс людського розвитку розраховується за 3ма параметрами: ймовірна тривалість життя, якість освіти та ВВП на людину.

Таблиця 3.1.2 – Таблиця датасету world-happiness-report.csv

Номер поля	Назва поля	Опис поля	Допустимі значення
1	Overall rank	Рейтинг рівня щастя	any integer
2	Country or region	Назва країни або регіону	any string
3	Score	Сумарна оцінка	any float
4	GBP per capita	Оцінка ВВП на одну особу	any float
5	Social support	Оцінка соціальної підтримки населення	any float
6	Healthy life expectancy	Оцінка ймовірності здорового життя	any float
7	Freedom to make life choices	Оцінка свободи приймати рішення	any float
8	Generosity	Оцінка щедрості населення	any float
9	Perceptions of corruption	Оцінка кількості випадків корупції	any float

В таблицях 3.1.3-3.1.7 наведено опис полів кожного датафрейму сховища даних про захворюваність на COVID-19.

Таблиця 3.1.3 – Таблиця датафрейму corona_dataset_clear

Номер поля	Назва поля	Опис поля	Допустимі значення
1	location (index)	Назва країни	any string
2	total_cases_per_million	К-ть випадків на мільйон осіб	any float
3	total_deaths_per_million	К-ть смертей на мільйон осіб	any float
4	reproduction_rate	Швидкість розповсюдження хвороби	any float
5	stringency_index	Індекс жорсткості	any float
6	population_density	Щільність населення	any float
7	median_age	Середній вік населення	any float
8	life_expectancy	Ймовірна тривалість життя	any float
9	human_development_index	Індекс людського розвитку	any float

Таблиця 3.1.4 – Таблиця датафрейму corona_dataset_aggregated

Номер поля	Назва поля	Опис поля	Допустимі значення
1	location	Назва країни	any string
2	total_cases_per_million	К-ть випадків на мільйон осіб	any float
3	total_deaths_per_million	К-ть смертей на мільйон осіб	any float
4	reproduction_rate	Швидкість розповсюдження хвороби	any float
5	stringency_index	Індекс жорсткості	any float
6	population_density	Щільність населення	any float
7	median_age	Середній вік населення	any float
8	life_expectancy	Ймовірна тривалість життя	any float
9	human_development_index	Індекс людського розвитку	any float

Таблиця 3.1.5 – Таблиця датафрейму happiness_aggregated

Номер поля	Назва поля	Опис поля	Допустимі значення
1	Country or region	Назва країни	any string
2	GPD per capita	ВВП на одну особу	any float
3	Social support	Оцінка соціальної підтримки населення	any float
4	Freedom to make life choices	Оцінка свободи приймати рішення	any float

При з’єднанні датафреймів за допомогою Inner Join в якості індексів використовувались поля location в датафреймі corona_dataset_aggregated та Country or region в датафреймі happiness_aggregated. Оскільки деякі назви полів в масивах є задовгими, частину з них було перейменовано. Зміни в назвах наведено в таблиці 3.1.6.

Таблиця 3.1.6 – Таблиця зі зміною назв датафрейму df_plus_happiness

Номер поля	Назва поля	Опис поля
1	total_cases_per_million	cases_per_mil
2	total_deaths_per_million	deaths_per_mil
3	reproduction_rate	reprod_rate
4	stringency_index	string_index
5	population_density	pop_density
6	median_age	median_age
7	life_expectancy	life_expect
8	human_development_index	hum_dev_index
9	GPD per capita	gdp_per_capita
10	Social support	social_support
11	Freedom to make life choices	life_choices

Таблиця 3.1.7 – Таблиця датафрейму df_plus_happiness

Номер поля	Назва поля	Опис поля	Допустимі значення
1	cases_per_mil	К-ть випадків на мільйон осіб	any float
2	deaths_per_mil	К-ть смертей на мільйон осіб	any float
3	reprod_rate	Швидкість розповсюдження хвороби	any float
4	string_index	Індекс жорсткості	any float
5	pop_density	Щільність населення	any float
6	median_age	Середній вік населення	any float
7	life_expect	Ймовірна тривалість життя	any float
8	hum_dev_index	Індекс людського розвитку	any float
9	gdp_per_capita	ВВП на одну особу	any float
10	social_support	Оцінка соціальної підтримки населення	any float
11	life_choices	Оцінка свободи приймати рішення	any float

4.ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ

4.1 Обґрунтування вибору методів інтелектуального аналізу даних

Для обробки інформації та виявлення в моделей та тенденцій необхідно використовувати інтелектуальних аналіз даних. В даній курсовій роботі для аналізу вже існуючих даних про захворюваність на COVID-19 та розповсюджуваність вірусу Sars-Cov-2, а також для відстеження залежностей між захворюваністю та загальним рівнем життя в країнах було обрано метод кластеризації. Завдяки кластеризації та поділенню країн на певні групи можна відстежити тенденції захворюваності та проаналізувати чи впливає загальний рівень життя в країні на швидкість та інтенсивність захворювань. Таким чином, можна зробити висновки щодо доцільності покращення життя в країні на загальному рівні для захищення її жителів під час наступних пандемій.

Для кластеризації даних було обрано два методи: K-means кластеризація та DBSCAN кластеризація. Ці два види кластеризації є досить популярними, тому по завершенню курсової роботи будуть наведені аргументи щодо більшої ефективності обох методів. Оскільки ми маємо діло з багатовимірними датафреймами, а кластеризація зазвичай відбувається на двохмірних масивах, для зменшення кількості вимірів було обрано та порівняно два методи: PCA(Principal component analysis, укр. Метод головних компонент) та T-SNE(t-distributed Stochastic Neighbor Embedding, укр. Т-розподілене вкладення стохастичної близькості)

Подальший аналіз даних та порівняння ефективності методів кластеризації буде описано в наступних підрозділах[4].

4.2 Теоретична складова

Кластерний аналіз – це задача розбиття заданої вибірки об’єктів на підмножини(кластери), так, щоб кожен кластер складався зі схожих об’єктів, а об’єкти різних кластерів істотно відрізнялись. До основних завдань кластерного аналізу відносять розробку класифікації, породження та перевірка гіпотез на основі дослідження даних. Існує 7 типових кластерних моделей, в цій курсовій роботи ми використаємо та порівняємо три: центроїдну модель, а саме метод K-

means, модель засновану на щільності – DBSCAN та модель зв'язності – ієрархічну кластеризацію.

K-means кластеризація:

Мета методу – розділити n спостережень на k кластерів, так щоб кожне спостереження належало до кластера з найближчим до нього середнім значенням. Метод базується на мінімізації суми квадратів відстаней між кожним спостереженням та центром його кластера, тобто функції

$$\sum_{i=1}^{N} d(x_i, m_j(x_i))^2,$$

де d- метрика, $x_i -$ і-ий об’єкт даних, а $m_j(x_i)$ – центр кластера

Алгоритм k-means кластеризації

1. Визначення кількості кластерів, в цій курсовій роботі було використано метод ліктя.
2. Випадковим чином обирається k спостережень, які на цьому кроці вважаються центрами кластерів
3. Кожне спостереження «приписується» до одного з n кластерів — того, відстань до якого найкоротша
4. Розраховується новий центр кожного кластера як елемент, ознаки якого розраховуються як середнє арифметичне ознак об'єктів, що входять у цей кластер
5. Повторення ітерацій, поки кластерні центри не стануть стійкими, дисперсія всередині кластера буде мінімізована, а між кластерами — максимізована

Метод ліктя (Elbow method)

Використовується в алгоритмах без вчителя для визначення оптимальної кількості кластерів. Кількість кластерів можна порахувати за допомогою спотворення (англ. Distortion) або інерції, суми квадратів відстаней об’єктів до їх найближчого центру кластера. В даній курсовій роботі кількість кластерів розраховується за допомогою методу ліктя з інерцією.

DBSCAN кластеризація

Алгоритм кластеризації заснований на щільності: для заданої множини точок у деякому просторі цей алгоритм відносить в одну групу точки, що найбільш щільно розташовані(точки з багатьма сусідами) та розмічає точки, які лежать в областях з невеликою щільністю як викиди (англ. Outliers)

Алгоритм роботи DBSCAN можна описати наступними кроками:

1. Знайти точки у кожному $\varepsilon -$ околі кожної точки та визначити ядрові точки, у яких більше ніж $minPts$ сусідів.
2. Знайти компоненти зв'язності для ядрових точок на графі сусідів, виключивши з розгляду точки, які не є ядровими.
3. Приєднати кожну неядрову точку до найближчого кластера, за умови, що кластер знаходиться в eps околі, інакше помітити її як шумову.

Для знаходження ε використовується метод коліна (knee method). Коліно відповідає порогу, де відбувається різка зміна вздовж кривої k-відстані, де k-кількість найближчих сусідів.

Зменшення кількості вимірів за допомогою PCA

Метод головних компонент дає можливість за m числом початкових ознак виділити n головних компонент, або узагальнених ознак. Математична модель PCA базується на логічному припущенні, що значення множини взаємозалежних ознак породжуються деякий загальний результат.

Ієрархічна кластеризація

Ієрархічна кластеризація (англ. hierarchical cluster analysis, HCA) – метод кластерного аналізу, який намагається побудувати ієрархію кластерів. Існує два типи стратегій побудови HCA: агломератові (підхід 'знизу-вгору') та розділювальні (підхід 'згори-вниз'). В агломератовій стратегії спочатку кожна точка має власний кластер, а далі пари кластерів об'єднуються при підйомі по ієрархії. В розділювальній стратегії спочатку всі точки знаходяться у єдиному кластері, потім відбувається рекурсивне розбиття при русі вниз по ієрархії.

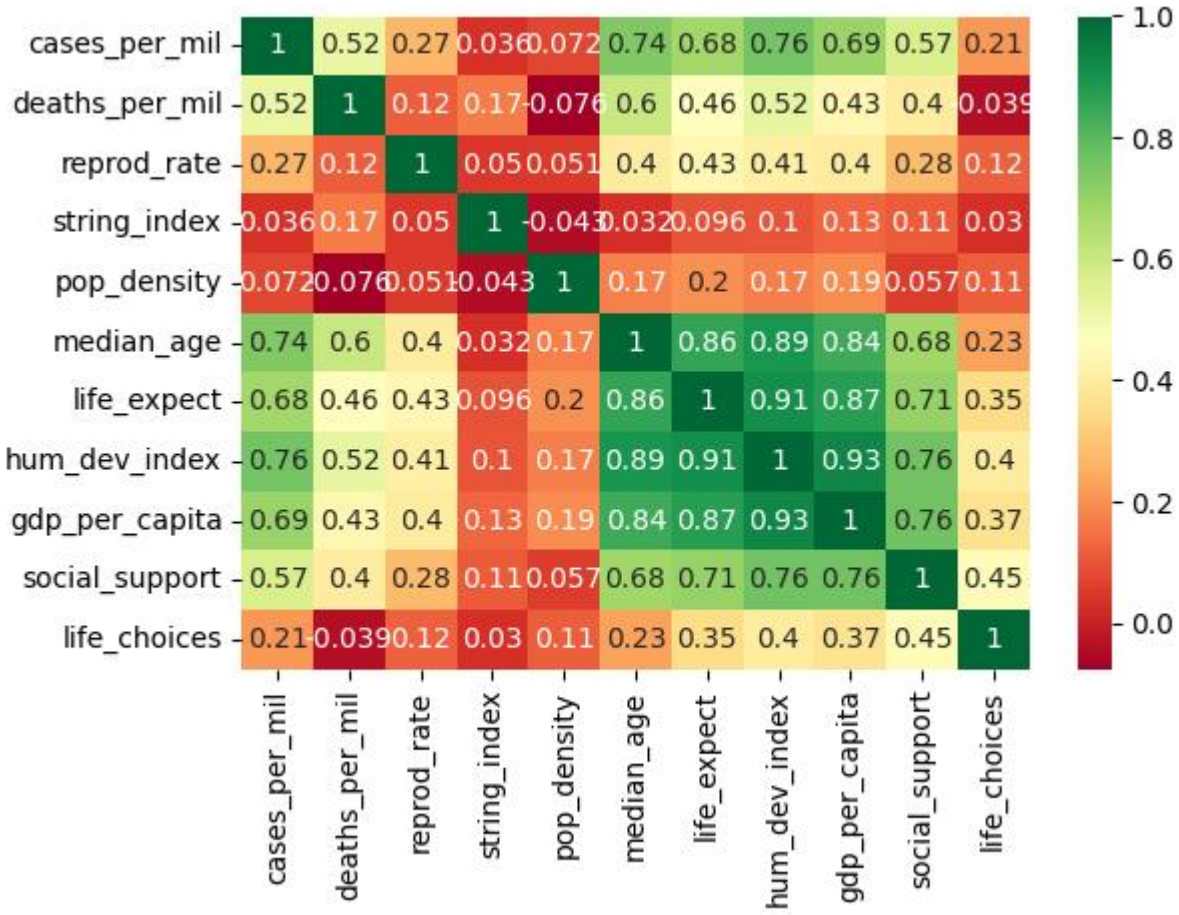
Алгоритм виконання ієрархічної кластеризації

1. Кожна точка даних розглядається як один кластер. Таким чином, кількість кластерів на початку дорівнює k, де k – кількість точок даних.

2. Новий кластер $k-1$ формується на основі об'єднання двох найближчих точок.
3. Новий кластер $k-2$ формується на основі об'єднання двох найближчих кластерів.
4. Кроки 2 та 3 повторюються поки не утвориться один великий кластер.
5. Після формування єдиного кластера використовується дендрограма для пошуку та поділу даних на оптимальну кількість кластерів.

4.3 Практичне застосування кластеризації та аналіз отриманих результатів

Перед початком виконання кластеризації було створено heatmap з кореляціями всіх полів датафрейму df_plus_happiness (мал. 4.3.1)



мал. 4.3.1 – heatmap кореляцій датафрейму df_plus_happiness

На графіку можна побачити, що поля 'Середній вік, 'Індекс розвитку людини', 'ВВП на особу', 'Соціальна підтримка', які відображають загальний стан жителів країни тісно корелюють між собою, в той час як поля 'Смертність на мільйон

осіб’, ‘Щільність населення’ та ‘Індекс жорсткості’ в цьому датасеті майже не мають кореляцій.

Порівняння K-means, DBSCAN та ієрархічної кластеризації

Склад початкового датафрейму зображено на мал. 4.3.2

	cases_per_mil	deaths_per_mil	...	social_support	life_choices
Afghanistan	4525.093	193.320	...	0.517	0.000
Albania	96104.192	1217.223	...	0.848	0.383
Algeria	5959.146	154.091	...	1.160	0.086
Argentina	202399.001	2826.152	...	1.432	0.471
Armenia	142501.604	2905.872	...	1.055	0.283

мал. 4.3.3 – склад початкового датафрейму

Для зменшення кількості вимірів використовується PCA. В обох випадках перед початком роботи використано StandartScaler для полегшення робити. Склад датафрейму після використання StandartScaler наведено на мал. 4.3.3. Приклад складу масиву після перетворення з 11 вимірів на 2 виміри за допомогою PCA наведено на мал. 4.3.4

	cases_per_mil	deaths_per_mil	...	social_support	life_choices
Afghanistan	-0.882603	-0.776977	...	-2.302100	-2.707711
Albania	-0.270508	-0.009090	...	-1.200626	-0.077186
Algeria	-0.873018	-0.806397	...	-0.162379	-2.117045
Argentina	0.439945	1.197542	...	0.742760	0.527217
Armenia	0.039603	1.257329	...	-0.511789	-0.764007

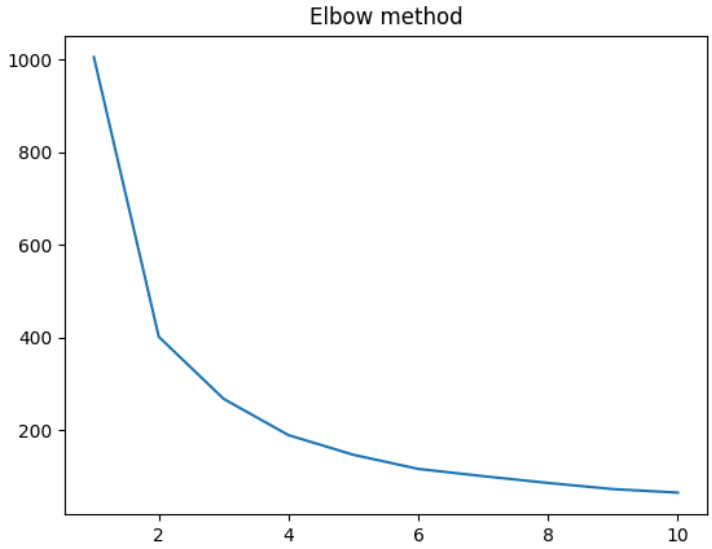
мал. 4.3.3 – склад датафрейму після використання StandartScaler

```
[[ 4.00092990e+00 -1.25722852e+00]
 [-1.92345010e-01 -3.22146229e-01]
 [ 7.04765523e-01 -8.29539852e-01]
 [-1.55782718e+00 -8.51097230e-01]
 [-3.66195607e-01 -1.22188029e+00]
 [-2.58953826e+00  1.06485334e+00]
 [-3.55532784e+00 -2.40217087e-02]
```

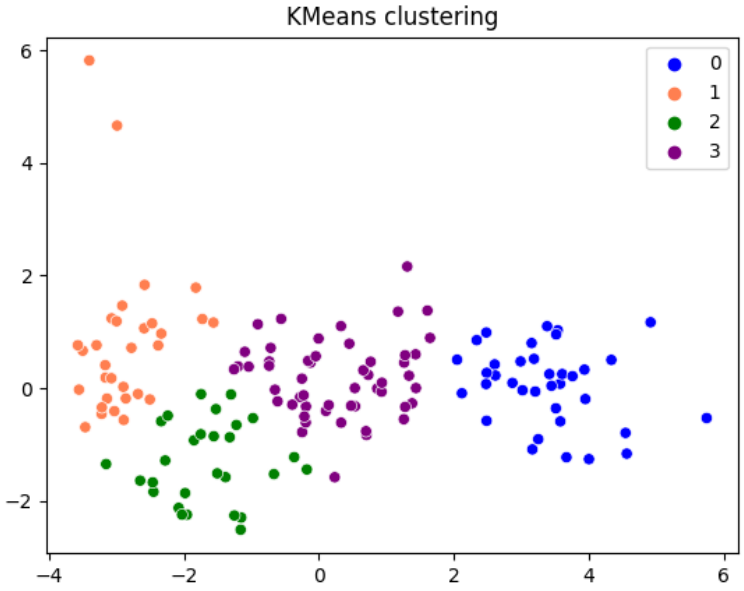
Мал. 4.3.4 – склад масиву після використання PCA

Спочатку виконаємо K-means кластеризацію. Першим кроком буде визначити кількість кластерів за допомогою методу ліктя. Графік методу ліктя для цього датафрейму зображено на мал. 4.3.5. З нього можна зробити висновок, що

оптимальна кількість кластерів – 4. Отже, в метод Kmeans передаємо цей результат та виводимо остаточний графік (мал. 4.3.6) Також в консоль виводимо результат кластеризації(мал. 4.3.7).



мал. 4.3.5, графік методу ліктя

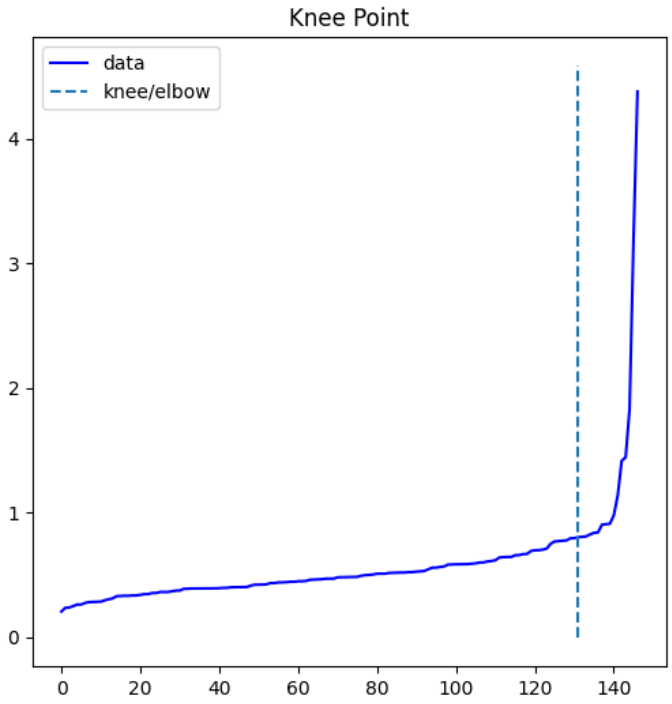


мал. 4.3.6, результат роботи k-means кластеризації

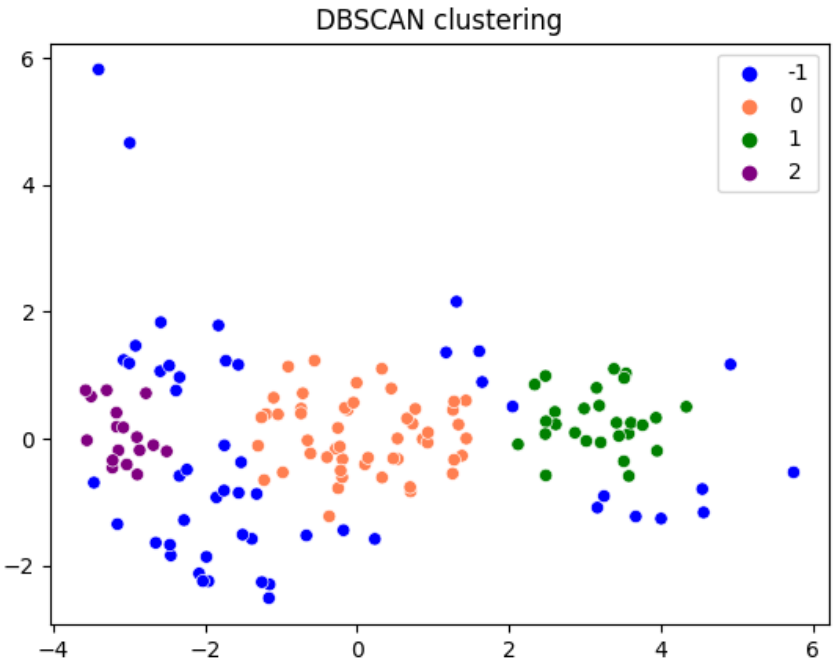
```
KMEANS clustering
[3 0 0 2 2 1 1 0 1 0 0 1 3 0 0 2 0 2 2 3 3 0 3 1 3 3 2 0 2 3 0 2 2 1 0 0 0
 0 1 3 1 1 0 3 2 1 3 2 0 3 3 0 1 2 1 0 0 0 0 1 1 2 0 1 0 0 3 0 0 0 0 2 0 3
 3 0 2 1 3 3 0 3 1 3 0 0 2 0 2 0 3 0 0 0 1 1 0 3 3 2 1 3 2 0 2 0 2 1 1 2 2
 3 0 3 2 3 1 2 1 3 0 1 3 1 0 1 1 3 0 0 3 0 3 0 2 3 2 1 1 1 2 0 0 0 3 3 3]
Counter({0: 52, 3: 36, 1: 31, 2: 28})
```

Мал. 4.3.7, консольний вивід результату роботи k-means кластеризації

Другий крок - кластеризація за допомогою DBSCAN. Спочатку знайдемо eps за допомогою методу коліна (мал. 4.3.8). З графіку та виводу в консоль можна зробити висновок, що оптимальний $eps = 0.8$. Результат роботи кластеризації DBSCAN та консольний вивід результатів зображено на мал. 4.3.9-4.3.10



мал.4.3.8, графік методу коліна

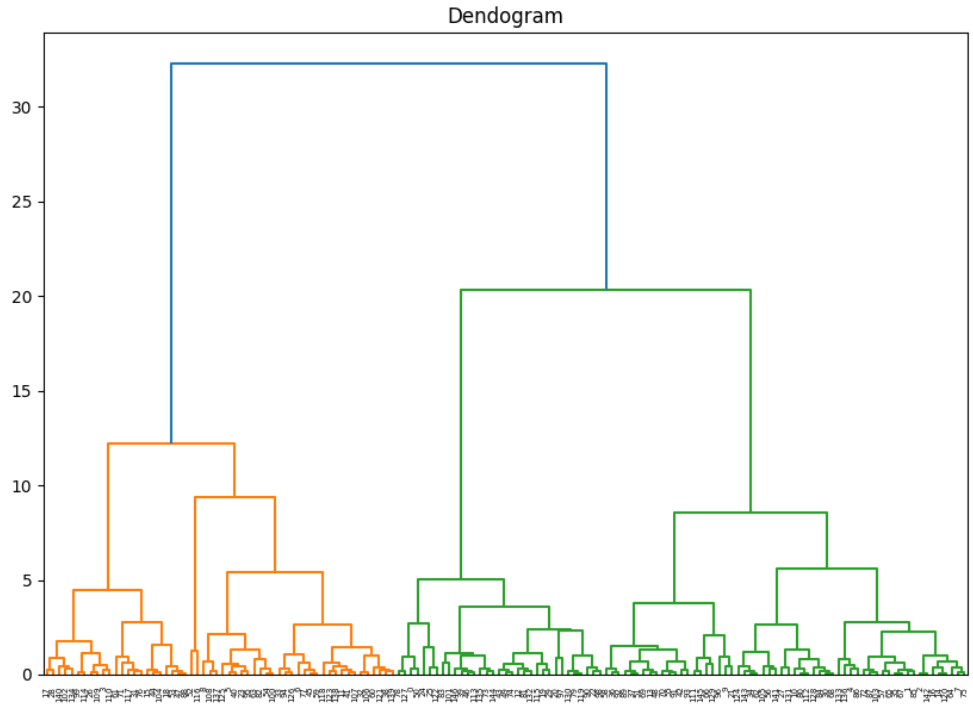


Мал. 4.3.9, результат роботи DBSCAN

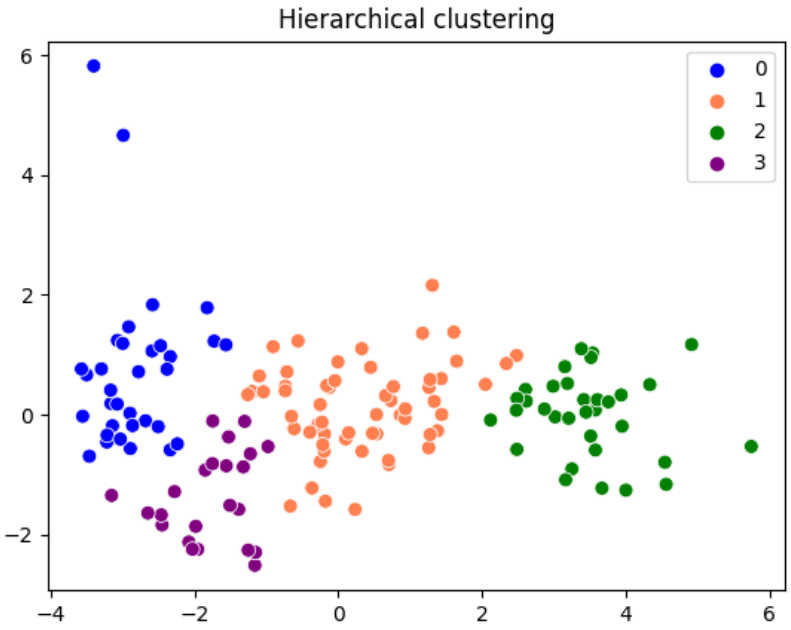
```
[-1  0  0 -1  0 -1  2  0 -1 -1  0  2  1  0  0 -1  0  0 -1  1 -1 -1  1 -1
 -1 -1 -1  0  0  1  0 -1 -1  2  0  0  0  0  2  1 -1  2  0  1 -1  2  1 -1
  0  1 -1  0 -1 -1 -1  0  0  0  0  2  2 -1  0 -1  0  0 -1  0  0  0  0 -1
  0  1  1  0 -1  2 -1  1  0  1 -1  1  0  0 -1  0 -1  0  1  0  0  0  2 -1
 -1  1  1 -1 -1  1  0  0 -1  0 -1  2 -1 -1 -1  1  0  1 -1  1 -1 -1 -1  1
  0  2 -1  2  0  2  2 -1  0 -1  1  0  1 -1 -1  1 -1 -1  2  2 -1  0  0  0
  1  1  1]
Counter({-1: 52, 0: 51, 1: 27, 2: 17})
```

Мал. 4.3.10, консольний вивід результату роботи DBSCAN

Третій метод кластеризації – ієрархічний. Спочатку створюємо та виводи дендрограму інформації з датафрейму (мал. 4.3.11). Можна побачити, що оптимальною кількістю кластерів для поділу є 4 кластерів. Отже, виводимо вхідний масив даних у форматі scatterplot з поділом на кластери (мал. 4.3.12). Також в консоль виводимо результати кластеризації (мал. 4.3.13)



мал. 4.3.11, дендрограма кластеризації



мал. 4.3.12, результат ієрархічної кластеризації

```
Hierarchical clustering
[2 1 1 3 1 0 0 1 0 1 1 0 2 1 1 3 1 3 3 2 2 1 2 0 2 2 3 1 3 2 1 3 0 0 1 1 1
 1 0 2 0 0 1 2 3 0 2 3 1 2 2 1 0 3 0 1 1 1 1 0 0 3 1 0 1 1 1 1 1 1 3 1 2
 2 1 3 0 2 2 1 2 0 2 1 1 1 1 3 1 2 1 1 1 0 0 1 2 2 3 0 2 3 1 3 1 0 0 0 3 3
 1 1 2 3 2 0 3 0 2 1 0 2 0 1 0 0 2 1 1 2 1 2 1 3 2 1 0 0 0 3 1 1 1 2 1 2]
Counter({1: 58, 2: 33, 0: 33, 3: 23})
```

мал. 4.3.13, консольний вивід результату роботи HCA

4.4 Аналіз та порівняння ефективності методів інтелектуального аналізу

В таблиці 4.3.1 наведено середні значення кожного поля, які використовувались для аналізу.

Таблиця 4.3.1 – середні показники полів

Опис поля	Середнє значення
К-ть випадків на мільйон осіб	136576.35
К-ть смертей на мільйон осіб	1229.34
Швидкість розповсюдження хвороби	2.27
Індекс жорсткості	84.95
Щільність населення	255.09
Середній вік населення	31.08
Ймовірна тривалість життя	73.15
Індекс людського розвитку	0.73
ВВП на одну особу	0.9
Оцінка соціальної підтримки населення	1.2
Оцінка свободи приймати рішення	0.39

Під час K-means кластеризації було створено 4 кластери. В таблиці 4.3.2 наведено результат роботи кластеризації, приклади країн, що були визначені в кожний кластер, наведено спільні характеристики кожного кластеру. Повна таблиця з розподіленням країн в кластери знаходяться в Додатку В, таблиця 1.

Таблиця 4.3.2 – результати K-means кластеризації

№ кластеру	Приклади країн	Спільні характеристики
0	Австралія, Австрія, Канада, Франція, Німеччина, Об'єднані Арабські Емірати, Великобританія, США	К-ть випадків > середнього п. К-ть смертей < середнього п. Розповсюдження ≈ середньому п. Індекс жорсткості ≈ середньому п. Щільність населення < середнього п. Середній вік > середнього п. Ймов. трив. життя > середнього п. Індекс розвитку > середнього п. ВВП на особу > середнього п. Рівень соц. підтримки > середнього п. Рівень свободи вибору > середнього п.
1	Азербайджан, Білорусь, Єгипет, Казахстан, Саудівська Аравія, Шрі-Ланка, Тайвань, Таджикистан, Таїланд.	К-ть випадків < середнього п. К-ть смертей < середнього п. Розповсюдження ≈ середньому п. Індекс жорсткості < середнього п. Щільність населення < середнього п. Середній вік > середнього п. Ймов. трив. життя ≈ середньому п. Індекс розвитку ≈ середньому п. ВВП на особу ≈ середньому п. Рівень соц. підтримки ≈ середньому п. Рівень свободи вибору < середнього п.
2	Афганістан, Бенін, Чад, Ефіопія, Гамбія, Мадагаскар, Руанда, Сирія, Замбія, Зімбабве	К-ть випадків < середнього п. К-ть смертей < середнього п. Розповсюдження < середнього п. Індекс жорсткості ≈ середньому п. Щільність населення < середнього п. Середній вік < середнього п. Ймов. трив. життя < середнього п. Індекс розвитку < середнього п. ВВП на особу < середнього п. Рівень соц. підтримки < середнього п. Рівень свободи вибору < середнього п.
3	Аргентина, Вірменія, Болгарія, Кіпр, Грузія,	К-ть випадків > середнього п. К-ть смертей > середнього п.

	Греція, Молдова, Польща, Росія, Туреччина, **Україна**, Уругвай	Розповсюдження ≈ середньому п. Індекс жорсткості ≈ середньому п. Щільність населення ≈ середньому п. Середній вік > середнього п. Ймов. трив. життя ≈ середньому п. Індекс розвитку ≈ середньому п. ВВП на особу ≈ середньому п. Рівень соц. підтримки ≈ середньому п. Рівень свободи вибору < середнього п.

Під час DBSCAN кластеризації було створено 4 кластери, де кластер зі значенням -1 вказує на викиди (англ. outliers) серед всього масиву інформації. В таблиці 4.3.3 наведено результат роботи кластеризації, приклади країн, що були визначені в кожний кластер, наведено спільні характеристики кожного кластеру. Повна таблиця з розподіленням країн в кластери знаходяться в Додатку В, таблиця 2.

Таблиця 4.3.3 – результат роботи DBSCAN кластеризації

№ кластеру	Назва країни	Спільні характеристики
0	Албанія, Вірменія, Білорусь, Колумбія, , Єгипет, Індія, Мексика, Монголія, Саудівська Аравія, Південна Африка, Шрі-Ланка, Тайвань, Таїланд, Узбекистан, Венесуела, В'єтнам	К-ть випадків < середнього п. К-ть смертей < середнього п. Розповсюдження ≈ середньому п. Індекс жорсткості > середнього п. Щільність населення > середнього п. Середній вік ≈ середньому п. Ймов. трив. життя ≈ середньому п. Індекс розвитку ≈ середньому п. ВВП на особу ≈ середньому п. Рівень соц. підтримки < середнього п Рівень свободи вибору ≈ середньому п.
1	Бенін, Буркіна-Фасо, Ефіопія, Гамбія, Гана, Гвінея, Лесото, Ліберія, Малаві, Малі, Мавританія, Мозамбік, Нігер, Нігерія, Пакистан, Руанда, Сенегал, Сьєрра-Леоне, Сомалі, Танзанія, Того, Уганда, Ємен, Замбія, Зімбабве	К-ть випадків < середнього п. К-ть смертей < середнього п. Розповсюдження < середнього п. Індекс жорсткості < середнього п. Щільність населення < середнього п. Середній вік < середнього п. Ймов. трив. життя < середнього п.п. Індекс розвитку < середнього п. ВВП на особу < середнього п. Рівень соц. підтримки < середнього п. Рівень свободи вибору < середньому п.

2	Австрія, Бельгія, Данія, Естонія, Франція, Німеччина, Ірландія, Ізраїль, Люксембург, Нідерланди, Португалія, Південна Корея, Іспанія, Швеція, Швейцарія, Велика Британія, США	К-ть випадків > середнього п. К-ть смертей > середнього п. Розповсюдження > середнього п. Індекс жорсткості ≈ середньому п. Щільність населення < середнього п. Середній вік > середнього п. Ймов. трив. життя > середнього п. Індекс розвитку ≈ середньому п. ВВП на особу > середнього п. Рівень соц. підтримки > середнього п. Рівень свободи вибору > середнього п.
-1 (outliers)	Афганістан, Аргентина, Австралія, Канада, Фінляндія, Грузія, Греція, Гаїті, Гонконг, Угорщина, Ісландія, Італія, Норвегія, Перу, Польща, Росія, Сирія, Туніс, Туреччина, **Україна**, Об'єднані Арабські Емірати, Уругвай	К-ть випадків > середнього п. К-ть смертей > середнього п. Розповсюдження ≈ середньому п. Індекс жорсткості ≈ середньому п. Щільність населення ≈ середньому п. Середній вік > середнього п. Ймов. трив. життя ≈ середньому п. Індекс розвитку ≈ середньому п. ВВП на особу ≈ середньому п. Рівень соц. підтримки ≈ середньому п. Рівень свободи вибору ≈ середньому п.

Під час виконання ієрархічної кластеризації було створено 4 кластери. В таблиці 4.3.4 наведено результат роботи кластеризації, приклади країн, що були визначені в кожний кластер, наведено спільні характеристики кожного кластеру. Повна таблиця з розподіленням країн в кластери знаходяться в Додатку В, таблиця 3.

Таблиця 4.3.4 – результат роботи ієрархічної кластеризації

№ кластеру	Назва країни	Спільні характеристики
0	Австралія, Австрія, Канада, Кіпр, Данія, Естонія, Фінляндія, Франція, Німеччина, Ісландія, Ірландія, Польща, Об'єднані Арабські Емірати, Великобританія, США	К-ть випадків > середнього п. К-ть смертей > середнього п. Розповсюдження > середнього п. Індекс жорсткості ≈ середньому п. Щільність населення < середнього п. Середній вік > середнього п. Ймов. трив. життя > середнього п. Індекс розвитку > середнього п. ВВП на особу > середнього п Рівень соц. підтримки ≈ середньому п. Рівень свободи вибору > середнього п

1	Албанія, Азербайджан, Камбоджа, Китай, Єгипет, Індія, Індонезія, Іран, Ірак, Казахстан, Монголія, Руанда, Саудівська Аравія, Таджикистан, Таїланд, Туніс, **Україна**, Узбекистан, Венесуела, В'єтнам, Замбія	К-ть випадків < середнього п. К-ть смертей ≈ середньому п. Розповсюдження ≈ середньому п. Індекс жорсткості ≈ середньому п. Щільність населення < середнього п. Середній вік > середнього п. Ймов. трив. життя > середнього п. Індекс розвитку < середнього п. ВВП на особу ≈ середньому п. Рівень соц. підтримки < середнього п. Рівень свободи вибору ≈ середньому п.
2	Афганістан, Бенін, Буркіна-Фасо, Бурунді, Камерун, Чад, Коморські острови, Ефіопія, Гамбія, Гана, Мавританія, Мозамбік, Сьєрра-Леоне, Сомалі, Південний Судан, Сирія, Танзанія, Того, Уганда, Ємен, Зімбабве	К-ть випадків < середнього п. К-ть смертей < середнього п. Розповсюдження < середнього п. Індекс жорсткості < середнього п. Щільність населення < середнього п. Середній вік < середнього п. Ймов. трив. життя < середнього п.п. Індекс розвитку < середнього п. ВВП на особу < середнього п. Рівень соц. підтримки < середнього п. Рівень свободи вибору < середньому п.
3	Аргентина, Боснія і Герцеговина, Бразилія, Болгарія, Чилі, Колумбія, Хорватія, Грузія, Греція, Угорщина, Італія, Латвія, Литва, Перу, Румунія, Росія, Сербія, Словаччина, Туреччина, Уругвай	К-ть випадків > середнього п. К-ть смертей > середнього п. Розповсюдження ≈ середньому п. Індекс жорсткості ≈ середньому п. Щільність населення ≈ середньому п. Середній вік > середнього п. Ймов. трив. життя ≈ середньому п. Індекс розвитку ≈ середньому п. ВВП на особу ≈ середньому п. Рівень соц. підтримки ≈ середньому п. Рівень свободи вибору < середнього п.

Аналіз результатів кластеризації

Кластери в Kmeans кластеризації поділені найбільшим чином за показниками смертності та захворюваності, також різниця між кластерами 1 та 2, що мають приблизно однакові показники смертності та захворюваності, полягає в тому, що країни в другому кластері мають менше свобод, наприклад свободи вибори та можливості отримувати соціальну підтримку.

Після аналізу результатів DBSCAN кластеризації можна зробити висновок, що до кластеру 0 відносяться країни з низьким рівнем смертності та захворюваності та з середнім рівнем загальних показників якості життя населення, наприклад середній вік населення та ВВП на особу. В кластері 1 знаходяться країни з низькими показниками захворюваності та смертності, але в той час ці країни мають низькі показники якості життя населення. До кластеру 2 належать країни з розвиненою економікою та високим рівнем життя, в той самий час в цих країнах спостерігаються великі рівні захворюваності та смертності.

Проаналізувавши результати ієрархічної кластеризації можна зробити висновок, що кластеризація відбувалася за схожим принципом с Kmeans кластеризацією. Країни більшим чином поділені за показниками смертності/захворюваності, але кластери 0 та 3 з показниками великої захворюваності та смертності відрізняються в загальному рівні життя, оскільки показники як ВВП на особу та ймовірна довжина життя перевищують середні показники. Також схожість між результатами Kmeans та ієрархічної кластеризації можна побачити на графіках. Оскільки DBSCAN кластеризація відбувається на основі щільності, то за його результатами можна побачити, що майже половина даних була позначена як викид.

Порівняння методів кластеризації та їх ефективності

До переваг виконання Kmeans кластеризації можна віднести його простоту та швидкість виконання. Метод k-середніх більш зручний для кластеризації великої кількості спостережень, ніж ієрархічна кластеризація, оскільки в HCA використовуються дендрограми, які можуть стати перевантаженими через велику кількість даних. Серед недоліків методу k-середніх можна виділити відсутність відкидання шуму (як наприклад в DBSCAN кластеризації) через що середнє значення може викривлюватись. Також результат кластеризації сильно залежить від випадкових початкових позицій центрів кластерів, а також кількість самих кластерів заздалегідь визначається дослідником, на відміну від двох інших методів.

До переваг DBSCAN кластеризації можна віднести те, що не потрібно завчасно самостійно визначати кількість кластерів. Виявлення викидів є надійним для виявлення аномалій та роботи з ними. DBSCAN потребує лише два параметри (ε та $minPts$) і є здебільшого нечутливим до впорядкування точок. Серед недоліків можна виділити неможливість виконати кластеризацію великих даних з великим перепадом щільностей, оскільки неможливо підібрати оптимальні значення параметрів, яке б відповідало різним кластерам. З такою проблемою ми зіткнулися і в цій курсовій роботі, оскільки майже половина даних була позначена як шум.

До переваг ієрархічної кластеризації можна віднести те, що як і в DBSCAN кластеризації, не потрібно завчасно самостійно визначати кількість кластерів. Завдяки цьому під час ієрархічної кластеризації можна зупинитися на будь-якій кількості кластерів. Основним мінусом використання ієрархічної кластеризації є її неможливість працювати на великих масивах даних, оскільки тоді дендрограма може стати перевантаженою.

ВИСНОВКИ

В результаті виконання курсової роботи було створено та проаналізовано дані про захворюваність на COVID-19. В якості методу інтелектуального аналізу даних було використано кластерний аналіз, а саме кластеризацію методом k-середніх, кластеризацію засновану на щільності DBSCAN та ієрархічну кластеризацію.

На основі детального опису та проведеного аналізу предметної області інтелектуального аналізу даних про захворюваність на COVID-19 та даних про рівень життя в визначених країнах було отримано залежність від якості життя та кількості випадків хвороби. Оскільки країни з розвиненою економікою мають високий рівень захворюваності можна зробити висновок, що країни з кращим рівнем життя мають вищу статистику, оскільки ці країни мають змогу швидше та ефективніше збирати інформацію про перетік хвороб. Також можна зазначити, що вірогідно країни з низьким рівнем якості життя мають низьку статистику щодо захворюваності та смертності, тому що в цих країнах тестування на ковід не виконувалися в великому обсязі, тому не було виявлено багато випадків хвороби.

Отже, поставлені задачі були виконані, а також планується продовження аналізу вхідної бази даних для виявлення більшої кількості залежностей між показниками якості життя та перебігом пандемії. Для реалізації поставленої задачі було використано мову програмування Python3.8 та бібліотеки NumPy, Pandas, Matplotlib та Sklearn. Середа розробки - PyCharm

ПЕРЕЛІК ПОСИЛАНЬ

1. “Кластерний Аналіз — Вікіпедія.” *Кластерний Аналіз — Вікіпедія*, uk.wikipedia.org, 2020, URL: t.ly/h5Ruy
2. “DBSCAN — Вікіпедія.” *DBSCAN — Вікіпедія*, uk.wikipedia.org, 2022, URL: https://uk.wikipedia.org/wiki/DBSCAN.
3. “Ієрархічна Кластеризація — Вікіпедія.” *Ієрархічна Кластеризація — Вікіпедія*, uk.wikipedia.org, 2022, URL: t.ly/V70t.
4. “Кластеризація Методом к–Середніх — Вікіпедія.” *Кластеризація Методом к–Середніх — Вікіпедія*, t.ly, 0 0 2022, URL: t.ly/d6H_.
5. “Clustering in Machine Learning - GeeksforGeeks.” *GeeksforGeeks*, www.geeksforgeeks.org, 15 Jan. 2018, URL: https://www.geeksforgeeks.org/clustering-in-machine-learning/.
6. “Pandas Documentation — Pandas 1.4.2 Documentation.” *Pandas Documentation — Pandas 1.4.2 Documentation*, pandas.pydata.org, B2022, URL: https://pandas.pydata.org/docs/.
7. “Matplotlib Documentation — Matplotlib 3.5.2 Documentation.” *Matplotlib Documentation — Matplotlib 3.5.2 Documentation*, matplotlib.org, URL: https://matplotlib.org/stable/index.html.
8. “Documentation Scikit-Learn: Machine Learning in Python — Scikit-Learn 0.15-Git Documentation.” *Documentation Scikit-Learn: Machine Learning in Python — Scikit-Learn 0.15-Git Documentation*, scikit-learn.org, URL: https://scikit-learn.org/0.15/documentation.html.

ДОДАТОК А ТЕКСТИ ПРОГРАМНОГО КОДУ

Тексти програмного коду кластеризації датасету за допомогою Kmeans, DBSCAN та ієрархічної кластеризації

(Найменування програми (документа))

SSD

(Вид носія даних)

5 арк

(Обсяг програми (документа), арк.,

студента групи ІТ-03 ІІ курсу

Цуканової Марини Сергіївни

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.decomposition import PCA
from sklearn.manifold import TSNE
from sklearn.neighbors import NearestNeighbors
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans, DBSCAN, AgglomerativeClustering
import scipy.cluster.hierarchy as shc
from kneed import KneeLocator
from collections import Counter

#----------- Step 1. Обробка вхідного датасету ----------------------
corona_dataset_csv = pd.read_csv("owid-covid-data.csv", header=[0])

lst_columns = ['continent', 'total_cases', 'total_deaths', 'date', 'iso_code',
'new_cases', 'new_cases_smoothed','new_deaths', 'new_deaths_smoothed',
               'new_cases_per_million', 'new_cases_smoothed_per_million',
'new_deaths_per_million', 'new_deaths_smoothed_per_million',
               'icu_patients', 'icu_patients_per_million', 'hosp_patients',
'hosp_patients_per_million',
               'weekly_icu_admissions', 'weekly_icu_admissions_per_million',
'weekly_hosp_admissions',
               'weekly_hosp_admissions_per_million', 'total_tests', 'new_tests',
'total_tests_per_thousand',
               'new_tests_per_thousand', 'new_tests_smoothed',
'new_tests_smoothed_per_thousand',
               'positive_rate', 'tests_per_case', 'tests_units',
'total_vaccinations', 'people_vaccinated',
               'people_fully_vaccinated', 'total_boosters', 'new_vaccinations',
'new_vaccinations_smoothed',
               'total_vaccinations_per_hundred',
'people_fully_vaccinated_per_hundred',
               'total_boosters_per_hundred',
'new_vaccinations_smoothed_per_million', 'new_people_vaccinated_smoothed',
               'new_people_vaccinated_smoothed_per_hundred', 'population',
'aged_65_older', 'aged_70_older',
               'gdp_per_capita', 'cardiovasc_death_rate', 'diabetes_prevalence',
'female_smokers', 'male_smokers',
               'handwashing_facilities', 'hospital_beds_per_thousand',
'excess_mortality_cumulative_absolute',
               'excess_mortality_cumulative', 'excess_mortality',
'excess_mortality_cumulative_per_million',
               'people_vaccinated_per_hundred', 'extreme_poverty']
lst_rows = ['Africa', 'Europe', 'Upper middle income', 'South America', 'World',
'North America', 'Asia', 'North Korea']

#очищуємо датасет від неподрібних колонок та строк
corona_dataset_clear = corona_dataset_csv.drop(lst_columns, axis=1)
corona_dataset_clear.set_index('location', inplace=True)
corona_dataset_clear.drop(lst_rows, axis=0, inplace=True)

corona_dataset_aggregated = corona_dataset_clear.groupby(['location']).max()
corona_dataset_aggregated =
corona_dataset_aggregated[corona_dataset_aggregated['total_cases_per_million'].n
otna()]
corona_dataset_aggregated =
corona_dataset_aggregated[corona_dataset_aggregated['total_deaths_per_million'].
notna()]
corona_dataset_aggregated.to_csv("output.csv")

#----------- Step 2. Обробка другого датасету -----------------------------
```

```
world_happiness_csv = pd.read_csv("worldwide_happiness_report.csv")
useless_columns = ['Overall rank', 'Generosity', 'Perceptions of corruption',
'Score', 'Healthy life expectancy']
happiness_aggregated = world_happiness_csv.drop(useless_columns, axis=1)
happiness_aggregated.set_index('Country or region', inplace=True)

#----------- Step 3. З'єднання двох датасетів -------------------------------
df_plus_happiness = corona_dataset_aggregated.join(happiness_aggregated,
how='inner')
df_plus_happiness = df_plus_happiness.fillna(df_plus_happiness.mean())
# df_plus_happiness.reset_index(inplace=True)
# df_plus_happiness.rename(columns={'index':'location'}, inplace=True)
df_plus_happiness.rename(columns = {'total_cases_per_million':'cases_per_mil',
                                    'total_deaths_per_million':'deaths_per_mil',
                                    'reproduction_rate':'reprod_rate',
                                    'stringency_index':'string_index',
                                    'population_density':'pop_density',
                                    'life_expectancy':'life_expect',
                                    'human_development_index':'hum_dev_index',
                                    'GDP per capita':'gdp_per_capita',
                                    'Social support':'social_support',
                                    'Freedom to make life
choices':'life_choices'}, inplace=True)
df_plus_happiness.to_csv("output_w_anomalies.csv")

#----------- Кореляції в датасеті -------------------------------------------

corrmat = df_plus_happiness.corr()
top_corr_features = corrmat.index
fig =
sns.heatmap(df_plus_happiness[top_corr_features].corr(),annot=True,cmap="RdYlGn"
)
fig.figure.tight_layout()
plt.savefig('correlation.png')
plt.show()


def pca(dataset):
    pca_2 = PCA(n_components=2)
    pca_2_result = pca_2.fit_transform(dataset)
    return pca_2_result

def tsne(dataset):
    tsne_em = TSNE(n_components=2, perplexity=30.0, n_iter=1000,
verbose=1).fit_transform(dataset)
    # cluster.tsneplot(score=tsne_em)
    # print(tsne_em)
    return tsne_em

def kneelocator(dataset):
    neighbors = NearestNeighbors(n_neighbors=5).fit(dataset)
    distance, index = neighbors.kneighbors(dataset)
    distance = np.sort(distance[:, 4], axis=0)
    i = np.arange(len(distance))
    knee = KneeLocator(i, distance, S=1, curve='convex', direction='increasing',
interp_method='polynomial')
    knee.plot_knee()
    plt.show()

    result = distance[knee.knee]
    print(distance[knee.knee])
    return result
```

```
def dbscan_clustering(dataset, knee):
    clusters = DBSCAN(eps=0.8, min_samples=17).fit(dataset)
    print(clusters.labels_)
    print(Counter(clusters.labels_))
    labels = clusters.labels_

    sns.scatterplot(data=dataset, x=dataset[:, 0], y=dataset[:, 1], hue=labels,
palette=['blue', 'coral', 'green', 'purple'], legend='full')
    plt.title('DBSCAN clustering')
    plt.show()

def kmeans_clustering(dataset):
    result = []
    for k in range(1, 11):
        kmeans = KMeans(n_clusters=k, init='k-means++')
        kmeans.fit(dataset)
        result.append(kmeans.inertia_)
    plt.plot(range(1, 11), result)
    plt.title('Elbow method')
    plt.show()

    kmeans = KMeans(n_clusters=4)
    label = kmeans.fit_predict(dataset)
    print(label)
    print(Counter(label))
    sns.scatterplot(data = dataset, x=dataset[:, 0], y=dataset[:, 1], hue=label,
palette=['blue', 'coral', 'green', 'purple'], legend='full')
    plt.title('KMeans clustering')
    plt.show()
    return label

def hierarchical_clustering(dataset):
    plt.figure(figsize=(10, 7))
    plt.title("Dendogram")
    dend = shc.dendrogram(shc.linkage(dataset, method='ward'))
    plt.show()
    clusters = AgglomerativeClustering(n_clusters=4, affinity='euclidean',
linkage='ward')
    label = clusters.fit_predict(dataset)
    sns.scatterplot(data=dataset, x=dataset[:, 0], y=dataset[:, 1],
hue=clusters.labels_,
                palette=['blue', 'coral', 'green', 'purple'], legend='full',
s=50)
    plt.title('Hierarchical clustering')
    plt.show()
    print(clusters.labels_)
    print(Counter(clusters.labels_))
    return label



if __name__=="__main__":
    df_plus_happiness[df_plus_happiness.columns] =
StandardScaler().fit_transform(df_plus_happiness)
    df_plus_happiness.to_csv('df_standard.csv')
    #DBSCAN PCA
    print('DBSCAN clustering')
    pca_dataset_dbscan = pca(df_plus_happiness)
    knee_tsne = kneelocator(pca_dataset_dbscan)
    dbscan_clustering(pca_dataset_dbscan, knee_tsne)
    #KMEANS PCA
    print('KMEANS clustering')
    pca_dataset_kmeans = pca(df_plus_happiness)
```

```
    labels_pca = kmeans_clustering(pca_dataset_kmeans)
    #HCA PCA
    print('Hierarchical clustering')
    pca_dataset_hca = pca(df_plus_happiness)
    labels_hca = hierarchical_clustering(pca_dataset_hca)



import pandas as pd

corona_dataset_csv = pd.read_csv("owid-covid-data.csv", header=[0])
lst_columns = ['continent', 'total_cases', 'total_deaths', 'date', 'iso_code',
'new_cases', 'new_cases_smoothed','new_deaths', 'new_deaths_smoothed',
               'new_cases_per_million', 'new_cases_smoothed_per_million',
'new_deaths_per_million', 'new_deaths_smoothed_per_million',
               'icu_patients', 'icu_patients_per_million', 'hosp_patients',
'hosp_patients_per_million',
               'weekly_icu_admissions', 'weekly_icu_admissions_per_million',
'weekly_hosp_admissions',
               'weekly_hosp_admissions_per_million', 'total_tests', 'new_tests',
'total_tests_per_thousand',
               'new_tests_per_thousand', 'new_tests_smoothed',
'new_tests_smoothed_per_thousand',
               'positive_rate', 'tests_per_case', 'tests_units',
'total_vaccinations', 'people_vaccinated',
               'people_fully_vaccinated', 'total_boosters', 'new_vaccinations',
'new_vaccinations_smoothed',
               'total_vaccinations_per_hundred',
'people_fully_vaccinated_per_hundred',
               'total_boosters_per_hundred',
'new_vaccinations_smoothed_per_million', 'new_people_vaccinated_smoothed',
               'new_people_vaccinated_smoothed_per_hundred', 'population',
'aged_65_older', 'aged_70_older',
               'gdp_per_capita', 'cardiovasc_death_rate', 'diabetes_prevalence',
'female_smokers', 'male_smokers',
               'handwashing_facilities', 'hospital_beds_per_thousand',
'excess_mortality_cumulative_absolute',
               'excess_mortality_cumulative', 'excess_mortality',
'excess_mortality_cumulative_per_million',
               'people_vaccinated_per_hundred', 'extreme_poverty']
lst_rows = ['Africa', 'Europe', 'Upper middle income', 'South America', 'World',
'North America', 'Asia', 'North Korea']

corona_dataset_clear = corona_dataset_csv.drop(lst_columns, axis=1)
corona_dataset_clear.set_index('location', inplace=True)
corona_dataset_clear.drop(lst_rows, axis=0, inplace=True)

corona_dataset_aggregated = corona_dataset_clear.groupby(['location']).max()
corona_dataset_aggregated =
corona_dataset_aggregated[corona_dataset_aggregated['total_cases_per_million'].n
otna()]
corona_dataset_aggregated =
corona_dataset_aggregated[corona_dataset_aggregated['total_deaths_per_million'].
notna()]

#=============================================================
world_happiness_csv = pd.read_csv("worldwide_happiness_report.csv")
useless_columns = ['Overall rank', 'Generosity', 'Perceptions of corruption',
'Score', 'Healthy life expectancy']
happiness_aggregated = world_happiness_csv.drop(useless_columns, axis=1)
happiness_aggregated.set_index('Country or region', inplace=True)

#=============================================================
```

```
df_plus_happiness = corona_dataset_aggregated.join(happiness_aggregated,
how='inner')
df_plus_happiness = df_plus_happiness.fillna(df_plus_happiness.mean())
print(list(df_plus_happiness.columns))
df_plus_happiness.reset_index(drop=True, inplace=True)
df_plus_happiness.rename(columns = {'total_cases_per_million':'cases_per_mil',
                                    'total_deaths_per_million':'deaths_per_mil',
                                    'reproduction_rate':'reprod_rate',
                                    'stringency_index':'string_index',
                                    'population_density':'pop_density',
                                    'life_expectancy':'life_expect',
                                    'human_development_index':'hum_dev_index',
                                    'GDP per capita':'gdp_per_capita',
                                    'Social support':'social_support',
                                    'Freedom to make life
choices':'life_choices'}, inplace=True)

print('mean')
print(df_plus_happiness.mean(), df_plus_happiness.min(),
df_plus_happiness.max())
```

ДОДАТОК Б. ОПИС ДАТАСЕТІВ

Опис складу датасету та його полів

_____

(Найменування програми (документа))

SSD

(Вид носія даних)

4 арк

(Обсяг програми (документа), арк.,

студента групи ІТ-03 ІІ курсу

Цуканової Марини Сергіївни

Таблиця 1 – Таблиця датасету owid-covid-data

Номер поля	Назва поля	Опис поля	Допустимі значення
1	iso_code	Код країни	any string
2	continent	Континент розташування країни	any string
3	location	Назва країни	any string
4	date	Дата спостереження	date year-month-day
5	total_cases	К-ть підтверджених випадків захворювання	any float
6	new_cases	К-ть нових випадків	any float
7	new_cases_smoothed	К-ть нових випадків за останні 7 днів	any float
8	total_deaths	К-ть смертей за весь період	any float
9	new_deaths	К-ть нових смертей	any float
10	new_deaths_smoothed	К-ть нових смертей за останні 7 днів	any float
11	total_cases_per_million	К-ть захворювань на мільйон осіб	any float
12	new_cases_per_million	К-ть нових випадків на мільйон осіб	any float
13	new_cases_smoothed_ _per_million	К-ть нових захворювань за 7 днів на мільйон осіб	any float
14	total_deaths_per_million	К-ть смертей з мільйона осіб	any float
15	new_deaths_per_million	К-ть нових смертей на мільйон осіб	any float
16	new_deaths_smoothed_ per_million	К-ть нових смертей за тиждень на мільйон осіб	any float
17	reproduction_rate	Швидкість розповсюдження хвороби	any float
18	icu_patients	К-ть пацієнтів в відділенні інтенсивої терапії	any float
19	icu_patients_per_million	К-ть пацієнтів в відділенні інтенсивної терапії на мільйон осіб	any float
20	hosp_patients	К-ть пацієнтів в лікарні в заданий день	any float
21	hosp_patients_per_million	К-ть пацієнтів в лікарні в заданий день на мільйон осіб	any float

22	weekly_icu_admissions	К-ть нових пацієнтів в відділенні інтенсивої терапії	any float
23	weekly_icu_admissions_ per_million	К-ть нових пацієнтів в відділенні інтенсивої терапії на мільйон осіб	any float
24	weekly_hosp_admissions	К-ть нових пацієнтів в лікарні в заданий день	any float
25	weekly_hosp_admissions_ per_million	К-ть пацієнтів в лікарні в заданий день	any float
26	total_tests	К-ть зроблених тестів	any float
27	new_tests	К-ть нових тестів	any float
28	total_tests_per_thousand	К-ть тестів на тисячу осіб	any float
29	new_tests_per_thousand	К-ть нових тестів на тисячу осіб	any float
30	new_tests_smoothed	К-ть нових тестів за тиждень	any float
31	new_tests_smoothed _per_thousand	К-ть нових тестів за тиждень на тисяцу осіб	any float
32	positive_rate	К-ть позитивних тестів за тиждень	any float
33	tests_per_case	Відношення тестів до позитивного випадку захворювання	any float
34	tests_units	К-ть місць для проведення тестів	any float
35	total_vaccinations	К-ть доз для вакцинації	any float
36	people_vaccinated	К-ть вакцинованих людей	any float
37	people_fully_vaccinated	К-ть повністю вакцинованих осіб	any float
38	total_boosters	К-ть бустерних доз	any float
39	new_vaccinations	К-ть нових вакцинацій	any float
40	new_vaccinations _smoothed	К-ть нових вакцинацій за тиждень	any float
41	total_vaccinations _per_hundred	К-ть вакцинацій на сто осіб	any float
42	people_vaccinated _per_hundred	К-ть вакцинованих людей на сто осіб	any float
43	people_fully_vaccinated _per_hundred	К-ть повністю вакцинованих людей на сто осіб	any float
44	total_busters _per_hundred	К-ть бустерних доз на сто осіб	any float

45	new_vaccinations_smoothed _per_million	К-ть нових вакцинацій за тиждень на мільйон осіб	any float
46	new_people_vaccinated _smoothed	К-ть нововакцинованих осіб за тиждень	any float
47	new_people_vaccinated _smoothed_per_hundred	К-ть нововакцинованих осіб за тиждень на 100 осіб	any float
48	stringency_index	Індекс жорсткості*	any float
49	population	К-ть населення країни	any float
50	population_density	Щільність населення	any float
51	median_age	Середній вік населення	any float
52	aged_65_older	Відсоток людей старше 65 років	any float
53	aged_75_older	Відсоток людей старше 75 років	any float
54	gdp_per_capita	ВВП на людину	any float
55	extreme_poverty	Відсоток людей, що живуть за межею бідності	any float
56	cardiovasc_death_rate	Відсоток смертності від хвороб судинної системи	any float
57	diabetes_prevalence	Поширеність цукрового діабету	any float
58	female_smokers	Відсоток жінок, що палять	any float
59	male_smokers	Відсоток чоловіків, що палять	any float
60	handwashing_facilities	Відсоток населення, що має доступ до об’єктів гігієни	any float
61	hospital_beds _per_thousand	К-ть лікарняних ліжок на тисячу осіб	any float
62	life_expectancy	Ймовірна тривалість життя	any float
63	human_development _index	Індекс людського розвитку**	any float
64	excess_mortality _cumulative_absolute	Кумулятивна різниця зареєстрованою к-тю смертей і прогнозованою за однаковий період	any float
65	excess_mortality _cumulative	Відсоткова різниця між зареєстрованою к-тю смертей і кумулятивною прогнозованою за однаковий період	any float
66	excess_mortality	Відсоткова різниця між зареєстрованою к-тю смертей і прогнозованою за однаковий період	any float

67	excess_mortality _cumulative_per_million	Кумулятивна різниця зареєстрованою к-тю смертей і прогнозованою за однаковий період на мільйон осіб	any float

*Індекс жорсткості розраховується за 9ма параметрами, такими як закриття шкіл, закриття робочих місць, скасування культурних подій, заборона суспільних зборів, закриття громадського транспорту, необхідність залишатися вдома, кампанії з освіти громадян, заборона переміщень всередині країни та контроль закордонних переміщень.

**Індекс людського розвитку розраховується за 3ма параметрами: ймовірна тривалість життя, якість освіти та ВВП на людину.

ДОДАТОК В. ТАБЛИЦІ З РЕЗУЛЬТАТИ РОБОТИ

Таблиці з результатами Kmeans, DBSCAN та ієрархічної кластеризації

(Найменування програми (документа))

SSD

(Вид носія даних)

3 арк

(Обсяг програми (документа), арк.,

студента групи ІТ-03 ІІ курсу

Цуканової Марини Сергіївни

Таблиця 1. Результат Kmeans кластеризації

№ кластеру	К-ть елем. кластеру	Назва країни
0	31	Австралія, Австрія, Бахрейн, Бельгія, Канада, Данія, Естонія, Фінляндія, Франція, Німеччина, Гонконг, Ісландія, Ірландія, Ізраїль, Японія, Люксембург, Мальта, Нідерланди, Нова Зеландія, Норвегія, Португалія, Катар, Сінгапур, Словенія, Південна Корея, Іспанія, Швеція, Швейцарія, Об’єднані Арабські Емірати, Великобританія, США
1	57	Албанія, Алжир, Азербайджан, Бангладеш, Білорусь, Бутан, Болівія, Ботсвана, Камбоджа, Китай, Коста-Ріка, Домініканська Республіка, Еквадор, Єгипет, Сальвадор, Габон, Гватемала, Гондурас, Індія, Індонезія, Іран, Ірак, Ямайка, Йорданія , Казахстан, Косово, Кувейт, Киргизстан, Лаос, Ліван, Лівія, Малайзія, Маврикій, Мексика, Монголія, Марокко, М'янма, Намібія, Непал, Нікарагуа, Парагвай, Філіппіни, Саудівська Аравія, Південна Африка, Шрі-Ланка, Тайвань, Таджикистан, Таїланд, Туніс, Узбекистан, Венесуела, В'єтнам
2	36	Афганістан, Бенін, Буркіна-Фасо, Бурунді, Камерун, Центральноафриканська Республіка, Чад, Коморські острови, Ефіопія, Гамбія, Гана, Гвінея, Гаїті, Кенія, Лесото, Ліберія, Мадагаскар, Малаві, Малі, Мавританія, Мозамбік, Нігер, Нігерія, Пакистан , Руанда, Сенегал, Сьєрра-Леоне, Сомалі, Південний Судан, Сирія, Танзанія, Того, Уганда, Ємен, Замбія, Зімбабве
3	28	Аргентина, Вірменія, Боснія і Герцеговина, Бразилія, Болгарія, Чилі, Колумбія, Хорватія, Кіпр, Грузія, Греція, Угорщина, Італія, Латвія, Литва, Молдова, Чорногорія, Північна Македонія, Панама, Перу, Польща, Румунія, Росія, Сербія , Словаччина, Туреччина, Україна, Уругвай

Таблиця 2. Результат DBSCAN кластеризації

№ кластеру	К-ть елем кластеру	Назва країни
0	51	Албанія, Алжир, Вірменія, Азербайджан, Білорусь, Бутан, Болівія, Ботсвана, Бразилія, Китай, Колумбія,

		Коста-Ріка, Домініканська Республіка, Еквадор, Єгипет, Сальвадор, Габон, Гватемала, Гондурас, Індія, Індонезія, Іран, Ірак, Ямайка , Йорданія, Казахстан, Косово, Кувейт, Киргизстан, Лаос, Ліван, Лівія, Малайзія, Маврикій, Мексика, Монголія, Марокко, М'янма, Намібія, Непал, Панама, Парагвай, Філіппіни, Саудівська Аравія, Південна Африка, Шрі-Ланка, Тайвань, Таїланд, Узбекистан, Венесуела, В'єтнам
1	27	Бенін, Буркіна-Фасо, Камерун, Коморські острови, Ефіопія, Гамбія, Гана, Гвінея, Лесото, Ліберія, Малаві, Малі, Мавританія, Мозамбік, Нігер, Нігерія, Пакистан, Руанда, Сенегал, Сьєрра-Леоне, Сомалі, Танзанія, Того, Уганда, Ємен, Замбія, Зімбабве
2	17	Австрія, Бельгія, Данія, Естонія, Франція, Німеччина, Ірландія, Ізраїль, Люксембург, Нідерланди, Португалія, Південна Корея, Іспанія, Швеція, Швейцарія, Велика Британія, США
-1 (outliers)	52	Афганістан, Аргентина, Австралія, Бахрейн, Бангладеш, Боснія і Герцеговина, Болгарія, Бурунді, Камбоджа, Канада, Центральноафриканська Республіка, Чад, Чилі, Хорватія, Кіпр, Фінляндія, Грузія, Греція, Гаїті, Гонконг, Угорщина, Ісландія, Італія , Японія, Кенія, Латвія, Литва, Мадагаскар, Мальта, Молдова, Чорногорія, Нова Зеландія, Нікарагуа, Північна Македонія, Норвегія, Перу, Польща, Катар, Румунія, Росія, Сербія, Сінгапур, Словаччина, Словенія, Південний Судан, Сирія, Таджикистан, Туніс, Туреччина, Україна, Об’єднані Арабські Емірати, Уругвай

Таблиця 3. Результат ієрархічної кластеризації

№ кластеру	К-ть елем кластеру	Назва країни
0	58	Австралія, Австрія, Бахрейн, Бельгія, Канада, Кіпр, Данія, Естонія, Фінляндія, Франція, Німеччина, Гонконг, Ісландія, Ірландія, Ізраїль, Японія, Люксембург, Мальта, Нідерланди, Нова Зеландія, Норвегія, Польща, Португалія, Катар, Сінгапур, Словенія, Південна Корея, Іспанія, Швеція, Швейцарія, Об'єднані Арабські Емірати, Великобританія, США
1	33	Албанія, Алжир, Вірменія, Азербайджан, Бангладеш, Білорусь, Бутан, Болівія, Ботсвана, Камбоджа, Китай,

		Коста-Ріка, Домініканська Республіка, Еквадор, Єгипет, Сальвадор, Габон, Гватемала, Гондурас, Індія, Індонезія, Іран, Ірак, Ямайка , Йорданія, Казахстан, Кенія, Косово, Кувейт, Киргизстан, Лаос, Ліван, Лівія, Малайзія, Маврикій, Мексика, Молдова, Монголія, Марокко, М'янма, Намібія, Непал, Нікарагуа, Парагвай, Філіппіни, Руанда, Саудівська Аравія, Південна Африка , Шрі-Ланка, Тайвань, Таджикистан, Таїланд, Туніс, Україна, Узбекистан, Венесуела, В'єтнам, Замбія
2	33	Афганістан, Бенін, Буркіна-Фасо, Бурунді, Камерун, Центральноафриканська Республіка, Чад, Коморські острови, Ефіопія, Гамбія, Гана, Гвінея, Гаїті, Лесото, Ліберія, Мадагаскар, Малаві, Малі, Мавританія, Мозамбік, Нігер, Нігерія, Пакистан, Сенегал , Сьєрра-Леоне, Сомалі, Південний Судан, Сирія, Танзанія, Того, Уганда, Ємен, Зімбабве
3	23	Аргентина, Боснія і Герцеговина, Бразилія, Болгарія, Чилі, Колумбія, Хорватія, Грузія, Греція, Угорщина, Італія, Латвія, Литва, Чорногорія, Північна Македонія, Панама, Перу, Румунія, Росія, Сербія, Словаччина, Туреччина, Уругвай