

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Навчально-науковий інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу**

До захисту допущено:

Завідувач кафедри

_____ Оксана ТИМОЩУК

«__» _____ 2026 р.

Дипломна робота

на здобуття ступеня бакалавра

**за освітньо-професійною програмою «Системний аналіз і управління»
спеціальності 124 «Системний аналіз»**

**на тему: «Кластеризація регіональних профілів повітряних тривог для
виявлення режимів ризику»**

Виконала:

студентка IV курсу, групи КА-23

Кабанова Марина Ігорівна

Керівник:

Професор кафедри ММСА, д.т.н., проф.

Кузнєцова Наталія Володимирівна

Консультант з економічного розділу:

Доцент кафедри ТПЕ, к.е.н.

Рощина Надія Василівна

Консультант з нормоконтролю:

Доцент кафедри ММСА, к.ф.-м.н.

Статкевич Віталій Михайлович

Рецензент:

Професор кафедри ІІІ, д.т.н., проф.

Данилов Валерій Якович

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.

Студентка _____

Київ – 2026 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 124 «Системний аналіз»

Освітньо-професійна програма «Системний аналіз і управління»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Оксана ТИМОЩУК

« 27 » _____ 05 _____ 2026 р.

ЗАВДАННЯ
на дипломну роботу студенту
Кабановій Марині Ігорівні

1. Тема роботи «Кластеризація регіональних профілів повітряних тривог для виявлення режимів ризику», керівник роботи Кузнєцова Наталія Володимирівна, доктор технічних наук, професор, затверджені наказом по університету від «27» травня 2026 р. № 1944-с
2. Термін подання студентом роботи: 10.06.2026
3. Вихідні дані до роботи: науково-методична література за темою роботи; помісячні регіональні дані про повітряні тривоги в Україні за 2023–2025 роки; дані про інфраструктурні події; адміністративно-територіальні дані України; програмні засоби Python.
4. Зміст роботи: теоретико-методичні основи аналізу та кластеризації часових рядів регіонального ризику; формування інформаційної бази, побудова ознак і вибір конфігурації кластеризації; аналіз результатів кластеризації, міжрічних переходів і просторового розподілу режимів ризику; функціонально-вартісний аналіз програмного продукту.
5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): порівняння методів кластеризації, інтерпретація режимів ризику, аналіз міжрічних переходів і просторова візуалізація результатів, презентація.
6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Функціонально-вартісний аналіз	Рощина Н.В., доц., к.е.н.		

7. Дата видачі завдання 20.09.2025

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Затвердження теми бакалаврської дипломної роботи (БДР), опанування ДСТУ 3008:2015	20.09.2025-26.09.2025	виконано
2	Огляд літератури з аналізу часових рядів, кластеризації, критеріїв стабільності та просторового аналізу	01.10.2025-26.10.2025	виконано
3	Збір, очищення та агрегування даних про повітряні тривоги й інфраструктурні події	27.10.2025-09.11.2025	виконано
4	Побудова ознак і формування річних багатовимірних профілів типу регіон-рік	10.11.2025-16.11.2025	виконано
5	Реалізація та порівняння методів кластеризації часових профілів, вибір фінальної конфігурації за критерієм Stadion; апробація проміжних результатів на конференціях	17.11.2025-18.02.2026	виконано
6	Інтерпретація режимів ризику, розрахунок річного індексу ризику, аналіз переходів і просторових закономірностей	19.02.2026-25.04.2026	виконано
7	Оформлення пояснювальної записки, джерел, додатків і презентації; підготовка матеріалів до міжнародної конференції	26.04.2026-31.05.2026	виконано
8	Передзахист БДР	05.06.2026	виконано

Студентка _____

Марина КАБАНОВА

Керівник _____

Наталія КУЗНЄЦОВА

РЕФЕРАТ

Дипломна робота: 139 с., 21 рис., 16 таб., 1 дод., 36 джерела.

ПОВІТРЯНІ ТРИВОГИ, РЕГІОНАЛЬНИЙ РИЗИК, ЧАСОВІ РЯДИ, КЛАСТЕРИЗАЦІЯ, DTW, SOFT-DTW, STADION, KAPLAN-MEIER, КОРЕЛЯЦІЯ СПІРМЕНА, ІНДЕКС МОРАНА, PYTHON.

Об'єкт дослідження – регіональні режими ризику в Україні в умовах повномасштабної війни.

Предмет дослідження – методи побудови, кластеризації та інтерпретації багатовимірних річних профілів регіонів для виявлення типових режимів регіонального ризику.

Мета роботи – розробка підходу до кластеризації річних регіональних профілів повітряних тривог для виявлення режимів ризику в Україні у 2023–2025 роках, та аналіз міжрічних переходів регіонів між цими режимами та їхнього просторового розподілу.

Методи дослідження – аналіз багатовимірних часових рядів, кластеризація часових профілів, метод k-середніх для часових рядів, метод м'якого динамічного викривлення часу, критерій стабільності Stadion, матриці міжрічних переходів, оцінка Каплана-Мейєра, кореляція Спірмена, глобальний індекс Морана, просторовий аналіз та візуалізація даних.

Результати роботи – виділено та інтерпретовано п'ять різних режимів ризику. Запропоновано річний індекс ризику, побудовано матриці міжрічних переходів, оцінено час до першого переходу регіону в більш ризиковий режим і виконано просторову інтерпретацію результатів.

Шляхи подальшого розвитку предмета дослідження – розширення часового періоду спостереження, доповнення новими ознаками регіональної вразливості, перевірка стійкості результатів.

Результати роботи можуть бути використані для моніторингу безпекової ситуації, переходів режимів та подальших досліджень регіональної стійкості.

ABSTRACT

Thesis: 139 pages, 21 figures, 16 tables, 1 appendix, 36 references.

AIR-RAID ALERTS, REGIONAL RISK, TIME SERIES, CLUSTERING, DTW, SOFT-DTW, STADION, KAPLAN-MEIER, SPEARMAN CORRELATION, MORAN'S INDEX, PYTHON.

Object of study – regional risk regimes in Ukraine under full-scale war conditions.

Research subject – methods for constructing, clustering, and interpreting multidimensional annual regional profiles to identify typical regional risk regimes.

Aim of the thesis – to develop an approach to clustering annual regional air-raid alert profiles in order to identify risk regimes in Ukraine in 2023–2025, as well as to analyse interannual transitions of regions between these regimes and their spatial distribution.

The research methods include multidimensional time series analysis, clustering of temporal profiles, time series k-means, Soft Dynamic Time Warping, the Stadion stability criterion, interannual transition matrices, Kaplan-Meier estimation, Spearman correlation, global Moran's Index, spatial analysis and data visualization.

The results of the thesis include the identification and interpretation of five different risk regimes. An annual risk index was proposed, interannual transition matrices were constructed, the time to the first transition of a region to a higher-risk regime was estimated, and a spatial interpretation of the results was carried out.

Further development of the research subject may include extending the observation period, adding new indicators of regional vulnerability, and testing the stability of the results.

The results of the thesis can be used for monitoring the security situation, analysing transitions between risk regimes, and supporting further studies of regional resilience.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 МЕТОДИКИ КЛАСТЕРИЗАЦІЇ ТА АНАЛІЗУ ЧАСОВИХ РЯДІВ ДЛЯ ВИЯВЛЕННЯ РЕЖИМІВ РЕГІОНАЛЬНОГО РИЗИКУ	12
1.1 Повітряні тривоги як індикатор регіонального ризику	12
1.2 Багатовимірні часові ряди як форма подання та аналізу регіональних безпекових процесів.....	12
1.3 Кластеризація часових рядів: сутність і основні підходи	14
1.4 Метрики подібності часових рядів	18
1.5 Оцінювання якості та стабільності кластеризації.....	20
1.6 Використані підходи до просторової інтерпретації результатів.....	23
1.7 Аналіз часу до події для оцінювання переходів між режимами ризику	25
Висновки до розділу 1	26
РОЗДІЛ 2 ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ БАЗИ ТА МЕТОДИКИ ВИЯВЛЕННЯ РЕЖИМІВ РИЗИКУ	29
2.1 Джерела та формування бази дослідження	29
2.1.1 Опис вхідних даних	29
2.1.2 Побудова ознак для опису інтенсивності та структури тривог	30
2.2 Описовий аналіз особливостей динаміки повітряних тривог у регіонах України у 2023–2025 роках	31
2.3 Формування багатовимірних річних профілів типу регіон-рік.....	35
2.4 Підготовка даних до кластеризації та вибір фінальної конфігурації.	36
Висновки до розділу 2	41
РОЗДІЛ 3 РЕЗУЛЬТАТИ КЛАСТЕРИЗАЦІЇ, ІНТЕРПРЕТАЦІЯ РЕЖИМІВ РИЗИКУ ТА АНАЛІЗ ЇХ ДИНАМІКИ	43

3.1 Результати фінальної кластеризації та характеристика виявлених режимів ризику.....	43
3.2 Формалізація річного індексу ризику повітряних тривог	49
3.3 Упорядкування режимів за відносною ризиковістю	51
3.4 Матриці міжрічних переходів між режимами ризику	53
3.5 Аналіз траєкторій регіонів між кластерами та рангами відносної ризиковості у 2023–2025 роках.....	55
3.6 Оцінювання часу до першого переходу регіону в більш ризиковий режим	58
3.7 Просторова інтерпретація виявлених режимів ризику.....	61
3.7.1 Картографічне представлення просторового розподілу індексів ризику	61
3.7.2 Оцінювання просторового градієнта за кореляцією Спірмена	63
3.7.3 Перевірка просторової автокореляції за глобальним індексом Морана	64
3.8 Обмеження дослідження та напрями подальшого розвитку	67
Висновки до розділу 3	69
РОЗДІЛ 4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ	71
4.1 Постановка задачі проектування.....	72
4.2 Обґрунтування функцій програмного продукту.....	73
4.3 Обґрунтування системи параметрів програмного продукту	77
4.4 Аналіз експертного оцінювання параметрів	80
4.5 Аналіз рівня якості варіантів реалізації функцій	84
4.6 Економічний аналіз варіантів розробки програмного продукту	85
4.7 Вибір кращого варіанту ПП техніко-економічного рівня	91

Висновки до розділу 4	92
ВИСНОВКИ.....	93
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	98
ДОДАТОК А. ЛІСТИНГ ПРОГРАМНОГО ПРОДУКТУ	103
ДОДАТОК Б. ГРАФІЧНІ МАТЕРІАЛИ.....	139

ВСТУП

Повномасштабна війна створила для України тривале середовище безпекової, соціальної та економічної нестабільності, яке неоднаково проявляється на рівні регіонів. Одним із найбільш регулярних і кількісно спостережуваних проявів цієї нестабільності є повітряні тривоги. Вони впливають на функціонування транспорту, підприємств, освітніх установ, сфери послуг і повсякденну поведінку населення, а тому можуть розглядатися як індикатор регіонального ризику в умовах війни. Повітряні тривоги, відключення та воєнні загрози мають відчутний операційний вплив на регіональну економічну активність і локальну стійкість [3–6].

Разом із тим для оцінювання такого ризику між різними регіонами недостатньо аналізувати лише сумарну кількість або загальну тривалість тривог. Регіони можуть мати близький загальний рівень тривожного навантаження, але істотно відрізнятися за характером його формування впродовж року: частотою тривог, середньою тривалістю епізодів, співвідношенням нічних і денних тривог, а також зв'язком із подіями ураження інфраструктури. Тому для більш змістовного аналізу доцільно розглядати не окремі агреговані показники, а багатовимірні часові профілі. Така логіка узгоджується із загальними підходами аналізу часових рядів і кластеризації часових даних, де ключове значення має не лише рівень показників, а й форма їх зміни в часі [7].

В межах цієї роботи кожна пара регіон-рік розглядається як окремий стан регіональної системи ризику, поданий у вигляді багатовимірного річного часового профілю. Така постановка дозволяє перейти від статичного порівняння регіонів до динамічного опису типових режимів ризику, які можуть повторюватися в різних регіонах і змінюватися в часі. Це особливо важливо в умовах війни, коли стан регіону не є фіксованим, а змінюється під впливом зовнішніх загроз, руйнувань інфраструктури та зміни інтенсивності бойових дій.

У роботі регіон розглядається як система, яка реагує на зовнішні загрози. Повітряні тривоги тут виступають показником того, як саме воєнні чинники змінюють стан цієї системи. Відповідно, предметом аналізу є не окрема тривога, а річний режим функціонування регіональної системи ризику, описаний набором взаємопов'язаних показників, зокрема частотою тривоги, їхньою тривалістю, добовою структурою та інфраструктурним компонентом.

Формально річний профіль регіону r у році y можна подати як багатовимірний часовий ряд $X_{r,y} = (x_{r,y,1}, x_{r,y,2}, \dots, x_{r,y,12})$. Після застосування алгоритму кластеризації кожному такому профілю присвоюється мітка кластера $z_{r,y} \in \{1, \dots, K\}$, яка після інтерпретації розглядається як номер режиму ризику.

Розуміння такого профілю дає змогу виявляти не тільки регіони з вищою або нижчою інтенсивністю тривоги, а й різні типи ризикової поведінки: стабільно помірні, інтенсивні, нічно зміщені, інфраструктурно напружені або перехідні режими. Подальший аналіз переходів між цими режимами дозволяє простежити, чи зберігає регіон свій стан, чи переходить до більш напруженої або менш напруженої конфігурації ризику.

Для інтерпретації та змістовного впорядкування отриманих режимів запропоновано річний індекс ризику повітряних тривоги, який поєднує інтенсивність тривожного навантаження та структурну напруженість режиму. Цей індекс не використовується для формування кластерів, а застосовується після кластеризації для інтерпретації та ранжування вже виявлених режимів за відносною ризиковістю.

Елементом новизни роботи є формалізація підходу до опису річних режимів регіонального ризику на основі багатовимірних профілів повітряних тривоги. На відміну від аналізу окремих агрегованих показників, у роботі кожен об'єкт типу регіон-рік подається як часовий профіль, кластеризується у режим ризику, а після кластеризації додатково характеризується річним індексом ризику. Крім того, міжрічна динаміка регіонів може бути описана через переходи між режимами, а також через індикатор першого переходу до

режиму з вищим рангом ризиковості. Це дозволяє не лише виділити типові режими, а й впорядкувати їх за відносною ризиковістю, проаналізувати міжрічні переходи регіонів і оцінити час до першого погіршення режиму ризику.

Метою роботи є узагальнення теоретичних засад і представлення результатів дослідження, присвяченого розробці підходу до кластеризації річних регіональних профілів повітряних тривог для виявлення режимів ризику в Україні у 2023–2025 роках. Окрему увагу приділено вибору критерію якості кластеризації, що фокусується на стабільності отриманого розбиття, упорядкуванню виявлених кластерів за відносною ризиковістю, аналізу міжрічних переходів між режимами та просторовій інтерпретації результатів.

Для досягнення поставленої мети у роботі розв’язуються такі завдання: проаналізувати повітряні тривоги як індикатор регіонального ризику; узагальнити теоретичні підходи до аналізу багатовимірних часових рядів і кластеризації часових профілів; сформулювати систему ознак, що характеризують інтенсивність і структуру тривог; побудувати річні профілі типу регіон-рік; порівняти альтернативні алгоритми кластеризації часових рядів; обрати фінальну конфігурацію за критеріями якості та стабільності; запропонувати індекс для впорядкування режимів за відносною ризиковістю; інтерпретувати виділені режими ризику; проаналізувати переходи між ними; формалізувати подію першого переходу до більш ризикового режиму та оцінити час до її настання; виконати просторову інтерпретацію отриманих результатів.

Отримані результати можуть бути використані для подальших досліджень і моніторингу змін безпекової ситуації в регіонах, виявлення переходів до більш напружених режимів, а також як аналітична основа для підтримки рішень щодо регіонального планування, відновлення та пріоритезації уваги до вразливих територій.

РОЗДІЛ 1 МЕТОДИКИ КЛАСТЕРИЗАЦІЇ ТА АНАЛІЗУ ЧАСОВИХ РЯДІВ ДЛЯ ВИЯВЛЕННЯ РЕЖИМІВ РЕГІОНАЛЬНОГО РИЗИКУ

1.1 Повітряні тривоги як індикатор регіонального ризику

В умовах повномасштабної війни повітряні тривоги можна розглядати як один із найбільш регулярних сигналів небезпеки, що відображає рівень зовнішньої загрози та супроводжує порушення нормального функціонування регіональних систем [4]. На відміну від багатьох інших воєнних показників, інформація про тривоги має виражений часовий характер, фіксується систематично і може бути агрегована на регіональному рівні [9]. Саме це робить її придатною для побудови порівнюваних часових профілів.

Разом з тим, повітряні тривоги є лише одним із проявів воєнного ризику, тому мають розглядатися разом із відповідними даними про ураження інфраструктури, порушення роботи сервісів та інші наслідки війни [9]. Тривоги виступають показником того, як війна впливає на життя та операційну діяльність регіону: через переривання роботи, логістичні обмеження, перебої з електроенергією та зниження економічної активності [3]. Цей підхід краще пояснює економічні втрати та допомагає оцінити стійкість територій під час війни [1].

1.2 Багатовимірні часові ряди як форма подання та аналізу регіональних безпекових процесів

Безпекові процеси мають виражену часову природу: змінюються нерівномірно, містять пікові періоди, хвилі загострення, локальні зсуви та

сезоноподібні коливання. Тому аналіз лише підсумкових значень часто втрачає важливу інформацію про структуру динаміки. У випадку повітряних тривог однакова річна тривалість або подібна кількість епізодів можуть відповідати різним режимам: стабільному фону, короткому періоду різкого загострення або частим коливанням упродовж року [1; 2].

Для змістовного опису безпекової ситуації недостатньо лише одного показника. Регіональний режим ризику формується одночасно кількома характеристиками: частотою тривог, середньою тривалістю епізодів, часовою структурою за добою, та зв'язком із подіями ураження інфраструктури [1; 2]. Тому в роботі використовується подання даних у вигляді багатовимірних часових рядів.

Нехай спостерігається d змінних часового ряду протягом T моментів часу. Тоді багатовимірний часовий ряд можна подати як послідовність векторів $X_t = (x_t, x_{2t}, \dots, x_{dt})^T, t = 1, \dots, T$. Таке подання використовується, коли потрібно аналізувати не одну ізольовану змінну, а спільну динаміку кількох взаємопов'язаних показників [7; 9].

Для задачі аналізу повітряних тривог це має принципове значення. Сумарні річні значення дозволяють оцінити загальне навантаження, але не показують, як саме воно формувалося протягом року. Наприклад, близька сумарна тривалість тривог може відповідати або стабільно напруженому фону, або короткому періоду різкого загострення. Аналогічно, подібна кількість тривог не означає однаковий режим, якщо в одному регіоні вони відбуваються переважно вдень, а в іншому вночі. Саме такі відмінності визначають реальну складність адаптації до ризику.

У межах цієї роботи часовий вимір є ключовим, оскільки об'єктом аналізу виступає не регіон загалом, а його стан у конкретний рік. Це означає, що кожна пара регіон-рік подається як окремий багатовимірний річний часовий профіль, де для кожного такого об'єкта формується річний профіль довжини 12 місяців, описаний набором ознак.

Нехай r позначає регіон, y – рік, $m = 1, \dots, 12$ – місяць року, а d – кількість

ознак, що описують тривожний режим. Тоді місячний стан регіону можна подати як вектор:

$$x_{r,y,m} = (x_{r,y,m}^1, x_{r,y,m}^2, \dots, x_{r,y,m}^d),$$

де $x_{r,y,m}^i$ – значення i -ої ознаки для регіону r у місяці m року y . Багатовимірний річний профіль регіону задається як послідовність таких місячних векторів:

$$X_{r,y} = (x_{r,y,1}, x_{r,y,2}, \dots, x_{r,y,12})$$

У такій постановці один і той самий регіон у різні роки може розглядатися як різний стан системи. Це дає змогу аналізувати не лише відмінності між регіонами, а й зміну режимів ризику в часі.

1.3 Кластеризація часових рядів: сутність і основні підходи

Кластерний аналіз є одним із базових методів багатовимірного статистичного аналізу, що використовується для групування об'єктів у відносно однорідні групи на основі певної міри подібності [34]. Кластеризація розглядається як метод навчання без учителя (unsupervised learning).

На відміну від класичної кластеризації статичних даних, у випадку часових рядів ключову роль відіграє не лише величина показників, а форма та характер їх зміни у часі [7]. Формально задачу кластеризації часових рядів можна описати наступним чином.

Нехай задано множину N багатовимірних часових рядів $X = \{X_1, X_2, \dots, X_N\}$, $X_i \in R^{d \times T}$, де T – довжина часового ряду, d – кількість вимірів або ознак. Потрібно знайти розбиття множини X на K кластерів $C =$

$\{C_1, C_2, \dots, C_K\}$, $K < N$, яке мінімізує сумарну внутрішньокластерну міру невідповідності відповідно до обраної метрики відстані.

Цей підхід використовується для виявлення закономірностей у наборах часових рядів і може бути корисним у різних прикладних задачах, зокрема для детекції аномалій, аналізу динаміки змін, пошуку повторюваних патернів, прогнозування та формування рекомендацій [7].

Кластеризація є інструментом, який дозволяє перейти від аналізу окремих показників до виявлення внутрішньої структури даних. У задачах регіонального аналізу безпеки це означає можливість виділити не просто регіони з вищими або нижчими значеннями окремих характеристик, а типові конфігурації ризику, що повторюються в даних.

У межах цієї роботи результат кластеризації інтерпретується як належність річного профілю до певного режиму ризику. Якщо $X_{r,y}$ – річний профіль регіону r у році y , то результат роботи алгоритму кластеризації можна подати як:

$$f(X_{r,y}) = z_{r,y}, \quad z_{r,y} \in \{1, \dots, K\},$$

де $z_{r,y}$ – мітка кластера, що після інтерпретації розглядається як номер режиму ризику, а K – кількість виділених режимів.

Оскільки в роботі використовується жорстка кластеризація, кожен об'єкт типу регіон-рік належить лише до одного кластера. Належність об'єкта до режиму k можна записати через індикаторну функцію:

$$I^k_{r,y} = \begin{cases} 1, & z_{r,y} = k \\ 0, & z_{r,y} \neq k \end{cases}$$

У літературі виокремлюють три базові підходи до кластеризації часових рядів: на основі форми (shape-based), на основі ознак (feature-based) та на основі моделей (model-based). Підхід на основі форми працює безпосередньо

з вихідними рядами та порівнює їхні часові профілі; підхід на основі ознак передбачає попереднє перетворення рядів у вектори статистичних характеристик; модельний підхід ґрунтується на апроксимації кожного ряду параметричною моделлю та подальшому групуванні за параметрами [7].

Для нашого дослідження найбільш доцільним є саме підхід на основі форми. Його перевага полягає в тому, що він дозволяє зберегти річний профіль у початковій часовій формі та безпосередньо порівнювати режими за динамікою. Це відповідає змісту поставленої задачі, адже метою є не звести кожен регіон до кількох агрегатів, а виявити повторювані типи річної поведінки повітряних тривог.

Серед роздільних алгоритмів кластеризації в роботі розглядається сімейство центроїдних методів. У класичному варіанті метод k -середніх поділяє множину з n об'єктів на k кластерів так, щоб мінімізувати суму квадратів відхилень об'єктів від центрів відповідних кластерів. При цьому центром кластера є арифметичне середнє об'єктів, що належать до нього. У більш загальному вигляді для методів цього типу цільову функцію можна записати як суму мір несхожості між об'єктами та прототипами кластерів:

$$J = \sum_{j=1}^k \sum_{x_i \in C_j}^n D(x_i, c_j),$$

де C_j – множина об'єктів j -го кластера; c_j – прототип кластера; $D(x_i, c_j)$ – міра несхожості між об'єктом і прототипом.

Як альтернативний варіант у роботі також розглядається агломеративна кластеризація, що є типом ієрархічної кластеризації. Вона дотримується підходу “знизу вгору”, де кожна точка даних починається як окремий кластер і поступово об'єднується з іншими на основі подібності. Об'єднання продовжується доти, доки всі точки не утворюють один кластер або не залишиться певна кількість кластерів.

На відміну від кластеризації часових рядів, ця кластеризація застосовувалася до векторизованих річних профілів, тобто до двовимірного подання даних після перетворення матриці в плоский вектор. У такому вигляді агломеративна кластеризація є корисною для порівняння, але меншою мірою зберігає специфіку часової структури [7].

У межах цієї роботи основну увагу приділено методам, які дозволяють порівнювати річні профілі як багатовимірні часові ряди та водночас зберігати інформацію не лише про форму, а й про відносний рівень показників. Це є важливим, оскільки режим ризику визначається не лише характером зміни показників упродовж року, а й загальною інтенсивністю тривожного навантаження, середньою тривалістю епізодів, добовою структурою та інфраструктурним компонентом.

Окремо як інший можливий підхід на основі форми можна розглядати алгоритм K-Shape, який орієнтований на групування часових рядів за подібністю нормалізованої форми. Його перевагою є здатність виділяти близькі характеристики динаміки навіть за наявності відмінностей у масштабі [10]. Водночас у межах цієї роботи K-Shape не було включено до фінального експериментального порівняння, оскільки для поставленої задачі важливо зберегти не лише форму профілю, а й відносний рівень ризику та календарну інтерпретацію місячних значень. Надмірне зосередження лише на нормалізованій формі могло б об'єднувати регіон-рік об'єкти зі схожою динамікою, але різним рівнем фактичного навантаження, що ускладнило б змістовну інтерпретацію режимів ризику.

Отже, у роботі зіставляються класичні векторні алгоритми із методами, які спеціально адаптовані для аналізу часових рядів. Це дозволяє перевірити, чи є часова динаміка визначальною для наших даних та обґрунтовано показати, який підхід краще відповідає змісту задачі виявлення річних режимів ризику.

1.4 Метрики подібності часових рядів

Ключовим елементом кластеризації часових рядів є вибір метрики подібності між профілями. Саме вона визначає, які об'єкти вважатимуться схожими, а отже, які режими будуть виділені в даних.

Найпростішою мірою є евклідова відстань, яка порівнює значення поелементно в однакових точках часу. Такий підхід добре працює тоді, коли часові профілі мають жорстко узгоджену вісь часу і змістовно важливо, в якому саме місяці року відбулося зростання чи спад.

Евклідова відстань вимірює поелементну різницю між двома векторами однакової довжини. У кластеризації k -середніх вона використовується для визначення близькості об'єкта до центроїда кластера [11]:

$$d_E = \sqrt{\sum_{k=1}^P (x_{ik} - v_{jk})^2},$$

де x_{ik} та v_{jk} задані P -розмірні вектори.

В межах поставленої задачі календарна вісь є змістовно важливою, тому евклідова відстань використовується як базовий контрольний варіант. Водночас вона є чутливою до зсувів піків у часі, наприклад, два профілі можуть бути подібними за загальною формою, але отримати велику відстань лише через те, що пік припав на сусідні місяці.

Для врахування такої ситуації широко використовується метод динамічного викривлення часу (Dynamic Time Warping, DTW), який дозволяє еластично вирівнювати часові послідовності шляхом пошуку оптимального шляху зіставлення між їхніми точками. На відміну від евклідової відстані,

DTW порівнює не тільки значення в однакові моменти часу, а й допускає локальні стискання або розтягування часової осі, якщо це дає змістовно кращу відповідність між профілями. Саме тому DTW широко використовується у сучасних прикладних задачах як стандартна метрика для часових рядів, де схожа динаміка може проявлятися з невеликим часовим зсувом. Формальне визначення відстані DTW задається наступною формулою [12]:

$$DTW_q(A, B) = \min_{\pi \in P} \left(\sum_{(i,j) \in \pi} d(a_i, b_j)^q \right)^{\frac{1}{q}},$$

де P – множина допустимих шляхів вирівнювання; $d(a_i, b_j)$ – відстань між точками a_i з ряду A та b_j з ряду B , та сума береться по всіх точках (i, j) в оптимальному шляху вирівнювання з P .

Подальшим розвитком DTW є метод м'якого динамічного викривлення часу (Soft Dynamic Time Warping, Soft-DTW). На відміну від класичного методу динамічного викривлення часу, у цьому підході замість жорсткого мінімуму використовується згладжений мінімум (soft-min) по множині можливих шляхів, заданий параметром γ . За малих значень γ міра подібності наближається до класичного DTW, тоді як за більших значень вирівнювання стає м'якшим і менш чутливим до локальних відмінностей між часовими профілями. Саме тому вибір γ має безпосередній вплив на те, наскільки жорстко або гнучко порівнюються річні часові профілі.

Для даної роботи це важливо, оскільки річні профілі повітряних тривог можуть містити локальні коливання, але при цьому зберігати схожу загальну форму динаміки.

Метод Soft-DTW як згладжений аналог DTW визначається за наступною формулою [13]:

$$softDTW_{\gamma}(A, B) = \min_{\pi \in P} \sum_{(i,j) \in \pi} d(a_i, b_j)^2,$$

де $\min^\gamma(a_i, \dots, a_n) = -\gamma \log \sum_i e^{-a_i/\gamma}$, а γ – коефіцієнт згладжування gamma ($\gamma > 0$).

Таким чином, у роботі порівнюються три підходи до вимірювання подібності між річними профілями: евклідова відстань як базове порівняння без часового вирівнювання, DTW як еластичне вирівнювання з урахуванням локальних зсувів, і Soft-DTW як згладжений варіант DTW, придатний для кластеризації часових рядів. Таке порівняння є необхідним, оскільки для профілів повітряних тривог наперед не можна однозначно стверджувати, який саме тип подібності найкраще відображає реальні відмінності між режимами ризику.

1.5 Оцінювання якості та стабільності кластеризації

Оцінювання якості кластеризації у даному дослідженні є необхідним, оскільки у задачі відсутня еталонна розмітка (ground truth), тому вибір моделі повинен спиратися на внутрішні критерії якості та стабільності.

Один із найвідоміших внутрішніх критеріїв є коефіцієнт силуету (Silhouette Score). Він порівнює середню відстань від об'єкта до елементів власного кластера з найменшою середньою відстанню до іншого кластера і таким чином відображає співвідношення між внутрішньокластерною компактністю та міжкластерним відокремленням. Для кожної точки даних з усього набору обчислюється за наступним чином [14]:

$$s(i) = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}} \in [-1, 1],$$

де $a(i)$ – середня відстань від точки i до власного кластера; $b(i)$ – мінімальна

середня відстань до точок іншого кластера.

Значення, близьке до 1, свідчить про добре відокремлений кластер, тоді як значення, близьке до -1, – про погіршення кластеризації.

Однак цей критерій якості також має певні обмеження. По-перше, коефіцієнт силуету занижує кількість кластерів у випадку слабого або помірного розподілу кластерів, недооцінюючи справжню структуру. По-друге, він гірше відображає якість розбиття для неопуклих або складно розподілених у метричному просторі кластерів. Крім того, за незбалансованих розмірів кластерів його значення може зміщуватися в бік великих груп [14].

У задачі кластеризації річних регіональних профілів повітряних тривог коефіцієнт силуету не був обраний як основний критерій відбору фінальної моделі. Причина полягає в тому, що його значення залежить від використаної міри несхожості. У задачі, де порівнюються різні уявлення про подібність часових рядів (евклідова відстань, DTW і Soft-DTW) силует оцінює якість розбиття всередині обраної геометрії, але не дає повністю нейтральної основи для вибору між різними типами часової подібності [14].

З огляду на це в роботі основний акцент зроблено не на компактності кластерів, а на стабільності знайденої структури. Для цього використано критерій Stadion, запропонований F. Forest та співавторами для валідації кластеризації часових рядів [15]. Його логіка базується на ідеї *invariance-guided stability*: хороша кластеризація має не лише давати внутрішньо узгоджене розбиття, а й зберігати свою структуру під малими змістовно допустимими збуреннями даних. Для часових рядів такими збуреннями можуть бути невеликий шум, локальні часові зсуви чи м'які зміни масштабу. Саме така логіка добре відповідає задачі аналізу річних профілів повітряних тривог, де природними є дрібні коливання та зсуви окремих піків.

У цьому підході поєднуються дві складові: міжкластерна стійкість (*between-cluster stability*, $Stab_B$) – наскільки відтворюється саме розбиття під збуреннями; внутрішньо кластерна стійкість (*within-cluster stability*, $Stab_W$) – чи не зберігається всередині кластерів прихована підструктура, яка свідчить

про їхню внутрішню неоднорідність. Підсумковий показник інтерпретується як компроміс між цими двома аспектами [15]:

$$\begin{aligned}
 Stadion(C_K, \Omega) &= Stab_B(C_K) - Stab_W(C_K, \Omega), \\
 Stab_B(A, X, C_K, K) &= \frac{1}{D} \sum_{d=1}^D s(C_K, A(X_d, K)), \\
 Stab_W(A, X, C_K, K, \Omega) &= \sum_{k=1}^K \left(\frac{1}{|\Omega|} \sum_{K' \in \Omega} Stab_B(A, C_k, Q_{K'}^{(k)}, K') \right) \times \frac{N_k}{N}
 \end{aligned}$$

Для $\{X_1, X_2, \dots, X_D\}$, D – це кількість збурених версій вихідного набору даних X . Міжкластерна стійкість алгоритму A з параметром K оцінює математичне очікування за допомогою емпіричного середнього значення подібності s між еталонною кластеризацією $C_K = A(X, K)$ та кластеризаціями збурених наборів даних. Формально, $\Omega \subset N^*$ це множина можливих кількостей підкластерів, на які перевіряється кожен уже знайдений кластер під час обчислення $Stab_W$, де k -й кластер у еталонній кластеризації позначається як C_K , його кількість елементів – як N_K , а $Q_{K'}^{(k)} = (C_k, K')$ позначає розбиття C_k на K' кластерів. При цьому, показник $Stab_B(C_K)$ має бути максимізований, тоді як $Stab_W(C_K, \Omega)$ – мінімізований.

В межах цього дослідження використано адаптацію цього критерію із малими шумовими збуреннями значень ознак. Для кожної кандидатної конфігурації спочатку виконується кластеризація вихідного стандартизованого масиву річних профілів. Далі формується серія збурених копій цього масиву шляхом додавання малого адитивного шуму до значень ознак. Після цього кластеризація повторюється з тією самою конфігурацією параметрів, а отримані розбиття порівнюються з базовим.

У даній задачі календарна вісь має предметний зміст, де кожна координата ряду відповідає конкретному місяцю року. Саме тому в межах

оцінювання стабільності не використовуються збурення, що переставляють місяці, зсувають піки або змінюють часовий порядок профілю. Подібні перетворення порушували б календарну інтерпретацію річного режиму. Натомість малі шумові збурення дозволяють перевірити, чи не є знайдена структура випадковим наслідком локальних варіацій значень, не руйнуючи зміст об'єкта типу регіон-рік.

У реалізації бакалаврської роботи міжкластерна стабільність оцінюється за узгодженістю базового та збурених розбиттів, а внутрішньокластерна через додаткову перевірку можливого підкластерування всередині вже знайдених кластерів. Таким чином, критерій Stadion у роботі використовується як міра балансу між відтворюваністю кластерної структури та відсутністю вираженої прихованої неоднорідності всередині кластерів. Відповідно, фінальна конфігурація кластеризації обирається не лише за якістю розділення, а передусім за стійкістю отриманих режимів.

1.6 Використані підходи до просторової інтерпретації результатів

Після виділення режимів ризику важливо з'ясувати, як вони розподілені у просторі. Це дозволяє зрозуміти, чи є виявлені режими випадковими, чи вони мають просторові закономірності, наприклад концентруються в сусідніх регіонах, чи пов'язані їх відмінності з географічним положенням регіонів. Така інтерпретація робить результати кластеризації більш змістовними та придатними для подальшого аналізу.

Рангова кореляція Спірмена використовується для визначення сили та напрямку зв'язку між двома змінними на основі їх рангів. Її значення змінюється в межах від -1 до +1, де додатні значення свідчать про позитивний монотонний зв'язок, від'ємні – про негативний, а значення, близькі до нуля, вказують на відсутність вираженого монотонного зв'язку [16]:

$$p = 1 - \frac{6\sum d^2}{n(n^2 - 1)},$$

де d – різниця між рангами відповідних значень; n – загальна кількість спостережень.

Для поставленої задачі, цей показник можна використовувати для перевірки зв'язку між інтегральними характеристиками ризику та географічними координатами регіонів.

Більш спеціалізованим інструментом є глобальний індекс Морана (Global Moran's I). Він дозволяє перевірити, чи мають сусідні об'єкти схожі значення досліджуваного показника. Додатні та статистично значущі значення Moran's I свідчать про просторове групування подібних значень, від'ємні – про просторове чергування, а значення, близькі до нуля, вказують на відсутність вираженої просторової структури [8]:

$$I = \frac{n}{S_0} \frac{\sum_{i=1}^n \sum_{j=1}^n w_{i,j} z_i z_j}{\sum_{i=1}^n z_i^2},$$

$$S_0 = \sum_{i=1}^n \sum_{j=1}^n w_{i,j},$$

де z_i – це відхилення значення атрибута для об'єкта i від його середнього значення ($x_i - X$); $w_{i,j}$ – просторова вага між об'єктами i та j ; n дорівнює загальній кількості об'єктів, а S_0 – сума всіх просторових ваг.

У межах цієї роботи цей показник дає змогу оцінити, чи зосереджуються подібні за рівнем ризику регіони поруч, чи їхній просторовий розподіл не має чіткої закономірності.

1.7 Аналіз часу до події для оцінювання переходів між режимами ризику

В задачах, де важливо оцінити не лише факт настання певної події, а й момент її першої появи, застосовується аналіз часу до події (Survival Analysis). Для кожного об'єкта визначається тривалість спостереження до настання події або до завершення періоду аналізу. Якщо подія не настала до кінця періоду, таке спостереження вважається цензурованим (censored data) [17].

У загальному випадку нехай T – невід'ємна випадкова величина, що позначає час до настання події. Тоді функція виживання визначається як ймовірність того, що подія не відбулася до моменту часу t [17]:

$$S(t) = P(T > t),$$

Одним із класичних непараметричних методів оцінювання функції виживання є оцінка Каплана-Мейєра (Kaplan-Meier). Її перевага полягає в тому, що вона не потребує припущення про конкретний розподіл часу до події та дозволяє враховувати цензуровані спостереження [18].

Нехай $t_1 < t_2 < \dots < t_k$ моменти часу, у які відбувається принаймні одна подія, d_j – кількість подій у момент часу t_j , а n_j – кількість об'єктів у групі ризику настання події безпосередньо перед моментом t_j . Тоді оцінка Каплана-Мейєра виглядає наступним чином [18]:

$$\hat{s}(t) = \prod_{t_j \leq t} \left(1 - \frac{d_j}{n_j}\right),$$

Для задачі аналізу регіональних режимів ризику подією може бути визначено перший перехід регіону до режиму з вищим рангом відносної ризиковості. Оскільки в роботі аналізуються дані за 2023–2025 роки, то час вимірюється в кількості міжрічних переходів з 2023 до 2024 та з 2024 до 2025 років.

Якщо до кінця періоду спостереження регіон не перейшов до більш ризикового режиму, це не означає, що такий перехід для нього неможливий. Це означає лише те, що в межах доступних даних за 2023–2025 роки подія не була зафіксована, та могла б відбутися пізніше за довшого періоду спостереження.

Окрім оцінки Каплана-Мейєра, в аналізі часу до події використовують також логарифмічний ранговий тест і регресію Кокса. Логарифмічний ранговий тест застосовується для порівняння функцій виживання між кількома групами [19]. Регресія Кокса використовується для оцінювання впливу пояснювальних змінних на інтенсивність настання події [20]. У цій роботі ці методи не використовуються, оскільки аналіз має обмежений часовий горизонт і спрямований не на побудову регресійної моделі, а на емпіричне узагальнення часу до першого погіршення режиму ризику.

Таким чином, оцінка Каплана-Мейєра є доречною для цієї задачі як інструмент описового аналізу міжрічної динаміки. Вона дозволяє узагальнити, яка частка регіонів зберігає стан без переходу до більш ризикового режиму після кожного міжрічного кроку, і тим самим доповнює матриці переходів та аналіз індивідуальних траєкторій.

Висновки до розділу 1

У першому розділі було обґрунтовано теоретичні засади дослідження регіональних профілів повітряних тривог в Україні. Показано, що повітряні

тривоги доцільно розглядати як один із спостережуваних індикаторів регіонального ризику в умовах повномасштабної війни, оскільки вони відображають інтенсивність зовнішніх загроз і супроводжують порушення нормального функціонування регіональних систем. Водночас встановлено, що для змістовного міжрегіонального порівняння недостатньо використовувати лише агреговані річні показники, оскільки близькі сумарні значення можуть відповідати різним режимам динаміки.

Обґрунтовано доцільність подання об'єктів дослідження у вигляді багатовимірних часових рядів типу регіон-рік, що дозволяє зберегти інформацію не лише про рівень показників, а й про форму їх зміни в часі. Показано, що така постановка є методично коректною для задачі виявлення типових річних режимів ризику та аналізу переходів між ними.

У розділі також узагальнено основні підходи до кластеризації часових рядів, метрики подібності часових рядів, насамперед евклідову відстань, DTW і Soft-DTW, а також критерії оцінювання якості та стабільності кластеризації. Показано, що в умовах відсутності еталонної розмітки особливе значення має не лише компактність кластерів, а й стійкість отриманої структури до малих збурень даних.

Окремо обґрунтовано вибір методів, включених до подальшого експериментального порівняння. Показано, що для поставленої задачі найбільш доречними є підходи, які дозволяють зберегти як часову форму річного профілю, так і відносний рівень показників ризику. Тому основну увагу в практичній частині приділено Euclidean K-Means, DTW-KMeans, SoftDTW-KMeans та агломеративній кластеризації. K-Shape розглянуто як методичну альтернативу для кластеризації на основі форми, однак не включено до фінального експериментального порівняння через його більшу орієнтацію на нормалізовану форму ряду, що меншою мірою відповідає потребі зберегти рівневу та календарну інтерпретацію річних режимів ризику.

Крім того, окреслено підходи до просторової інтерпретації результатів

кластеризації, що дає можливість перейти від формального групування профілів до змістовного аналізу територіальної організації ризику. Отже, перший розділ формує теоретичне і методичне підґрунтя для подальшого аналізу даних, побудови річних профілів та виділення режимів ризику у практичній частині дослідження.

РОЗДІЛ 2 ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ БАЗИ ТА МЕТОДИКИ ВИЯВЛЕННЯ РЕЖИМІВ РИЗИКУ

Програмна реалізація дослідження виконана мовою програмування Python у середовищі розробки Visual Studio Code. Для обробки даних, кластеризації часових рядів, просторового аналізу та оцінювання часу до події використано бібліотеки pandas, numpy, scikit-learn, tslearn, GeoPandas, PySAL, esda, matplotlib, scipy та lifelines [21–29].

2.1 Джерела та формування бази дослідження

2.1.1 Опис вхідних даних

Основою дослідження є помісячні регіональні дані за період з січня 2023 року по грудень 2025 року. До аналізу не включено 2022 рік, оскільки початковий етап повномасштабного вторгнення характеризувався різкою зміною умов функціонування регіонів і суттєвою нерівномірністю даних, що ускладнює коректне порівняння з наступними роками.

Основними джерелами даних є сервіс Alerts.in.ua [9], з якого отримано інформацію про кількість, тривалість і часовий розподіл повітряних тривог, а також база ACLED [30], з якої використано відомості про події, пов'язані з ураженням цивільної та критичної інфраструктури. В межах роботи ці джерела поєднуються на помісячному регіональному рівні.

Початковою одиницею спостереження є запис формату регіон-місяць. Для кожного такого запису наявні показники, що характеризують інтенсивність повітряних тривог, їхній часовий розподіл і супровідні інфраструктурні події. Надалі ці помісячні дані використовуються для побудови річних профілів типу регіон-рік, які і є основними об'єктами

кластеризації.

Дослідження не включає Автономну Республіку Крим, місто Севастополь, та Луганську область, оскільки через тимчасову окупацію цих регіонів дані частково відсутні, що могло б спотворювати результати кластерного аналізу.

Таким чином, на початковому етапі формується помісячна регіональна база даних, яка надалі перетворюється на багатовимірні річні профілі для виявлення типових режимів ризику.

2.1.2 Побудова ознак для опису інтенсивності та структури тривог

Після зведення вихідних даних у помісячну регіональну таблицю було виконано базову перевірку якості: усунено дублікати, перевірено пропуски, а також приведено змінні до числового формату. Для відносних показників, що мають природне обмеження в інтервалі від 0 до 1, додатково контролювалися граничні значення.

На основі вхідних даних сформовано систему похідних ознак, які дозволяють описувати не лише загальний рівень тривог, а й структуру ризикового режиму. Також було сформовано нові агреговані ознаки, що краще відображають інтенсивність і структуру тривог для порівняння характеру:

- кількість тривог на день (`alert_count_per_day`) – відношення кількості тривог за місяць до кількості календарних днів у цьому місяці. Цей показник характеризує середню частоту тривог і дає змогу порівнювати між собою місяці різної тривалості;
- тривалість тривог за день (`alert_hours_per_day`) – відношення сумарної тривалості тривог за місяць до кількості днів. На відміну від простої кількості тривог, ця змінна відображає загальне часове навантаження на регіон;

- середня тривалість однієї тривоги (`hours_per_alerts`) – відношення сумарної тривалості тривог до їх кількості. Вона дозволяє відрізнити режими з частими короткими тривогами від режимів з меншою кількістю, але довгими епізодами;
- частка нічних тривог (`night_alerts_proportion`) – відношення кількості тривог у часовому інтервалі 22:00–06:00 до загальної кількості тривог за місяць. Ця змінна відображає структурну напруженість режиму, оскільки нічні тривоги мають інший вплив на повсякденне функціонування населення та інфраструктури;
- частка тривог у денні операційні години (`operational-hours_alert_share`) – характеризує частину тривог, що припадає на часовий проміжок найбільшої ділової та соціальної активності (10:00–21:00), і використовується для опису потенційного операційного навантаження;
- інфраструктурний компонент тривог (`infr_attacks_per_alert`) – відношення кількості подій, пов’язаних з ураженням інфраструктури, до кількості тривог у відповідному місяці. Цей показник дозволяє врахувати, наскільки режим тривог супроводжувався подіями з підвищеною інфраструктурною напруженістю.

2.2 Описовий аналіз особливостей динаміки повітряних тривог у регіонах України у 2023–2025 роках

Було розглянуто дві групи характеристик: показники інтенсивності тривожного навантаження та показники його часової структури тривог.

На рисунку 2.1 відображено середню місячну динаміку кількості тривог на день (`alert_count_per_day`) і тривалості тривог за день (`alert_hours_per_day`) в Україні за 2023–2025 роки. На рисунку 2.2 показано зміну частки нічних

тривог (night_share), частки тривог у денні операційні години (mall_share) та кількості інфраструктурних атак на одну тривогу (infr_attacks_per_alert). Такий поділ дозволяє окремо проаналізувати загальний рівень навантаження та зміну його внутрішньої структури.

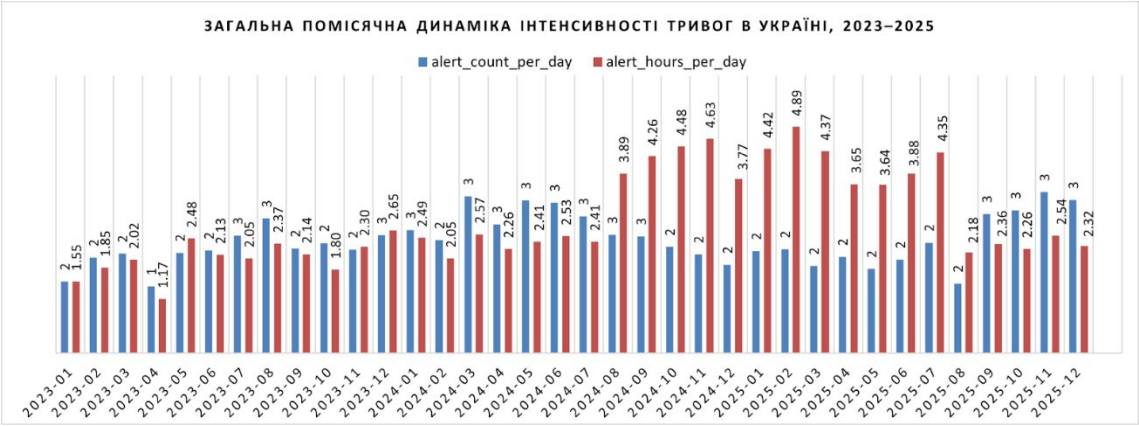


Рисунок 2.1 – Загальна помісячна динаміка інтенсивності тривог в Україні, 2023–2025

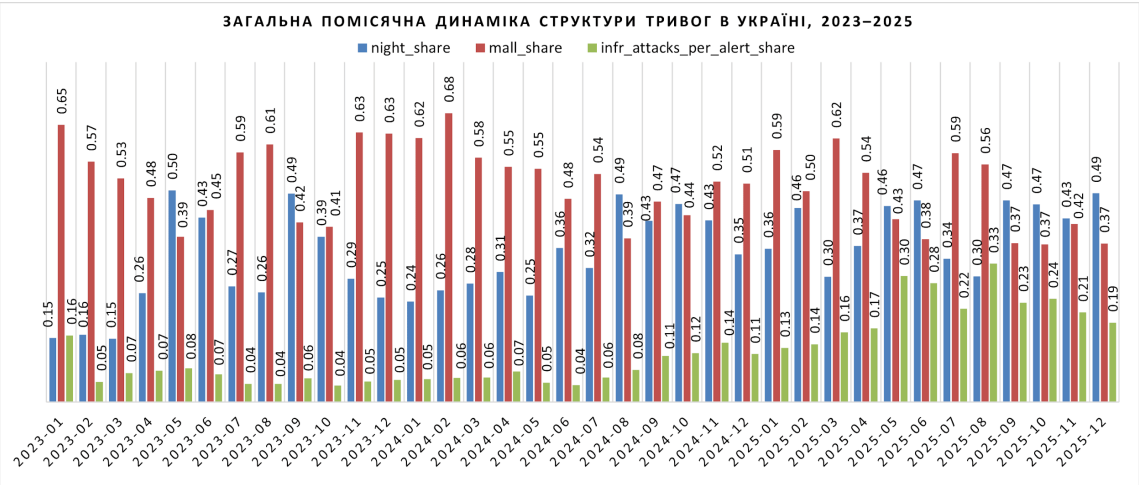


Рисунок 2.2 – Загальна помісячна динаміка структури тривог в Україні, 2023–2025

За показниками інтенсивності протягом усього періоду спостерігається виражена хвилеподібна динаміка. У 2023 році середня кількість тривог на день змінювалася нерівномірно, де після нижчого значення у квітні вона зростала у літньо-осінній період і надалі коливалася поблизу 2–3 тривог на день. У 2024 році кількість тривог на день залишається відносно стабільною, однак помітно змінилася тривалість тривожного навантаження. Починаючи з серпня 2024 року значення alert_hours_per_day різко зросло й залишалося вищим за інші

роки рівні аж до серпня 2025 року.

Структурні показники також демонструють помітні варіації рівнів. Частка нічних тривог у 2023 році мала кілька підвищень, зокрема у весняно-літній період та у вересні, коли значення наближалося до 0.5. У 2024 році високі значення нічного компоненту знову спостерігалися в серпні–листопаді, а у 2025 році – у лютому, травні, червні, вересні, жовтні та грудні. Отже, нічний компонент не був випадковою локальною особливістю окремих місяців, а виступав повторюваною складовою ризикового режиму.

Показник частки тривог у денні операційні години також демонструє помітну мінливість. У 2023 році після весняного зниження його значення знову зросли влітку та наприкінці року, а в окремі місяці перевищували 0.6. У 2024 році денний компонент загалом залишався відносно високим, хоча в другій половині року його рівень дещо зменшився порівняно з початком року. У 2025 році він мав кілька виражених підйомів, зокрема у березні, липні та серпні, однак наприкінці року простежувалося поступове зниження.

У свою чергу, показник кількості інфраструктурних атак на одну тривогу у 2023 році залишався відносно невисоким і переважно не перевищував 0.08. У 2024 році значення почало поступово зростати, особливо в другій половині року. Найбільш виражене посилення спостерігалось у 2025 році, коли показник утримувався на вищому рівні майже протягом усього року, а локального максимуму досяг у серпні.

Для оцінювання міжрегіональної неоднорідності було також побудовано коробкові графіки за роками (рис. 2.3). Вони показують, що у 2024–2025 роках зросли як середні рівні окремих показників, так і розкид значень між регіонами. Це особливо помітно для частоти тривог на день та середньої тривалості однієї тривоги, де верхні хвости розподілів показників регіонів у 2024 і 2025 роках стали значно виразнішими. Для нічного компоненту також простежується поступове зміщення медіани вгору, тоді як, найбільше посилення міжрегіональної неоднорідності спостерігається для показника кількості інфраструктурних атак на одну тривогу, особливо у 2025 році.

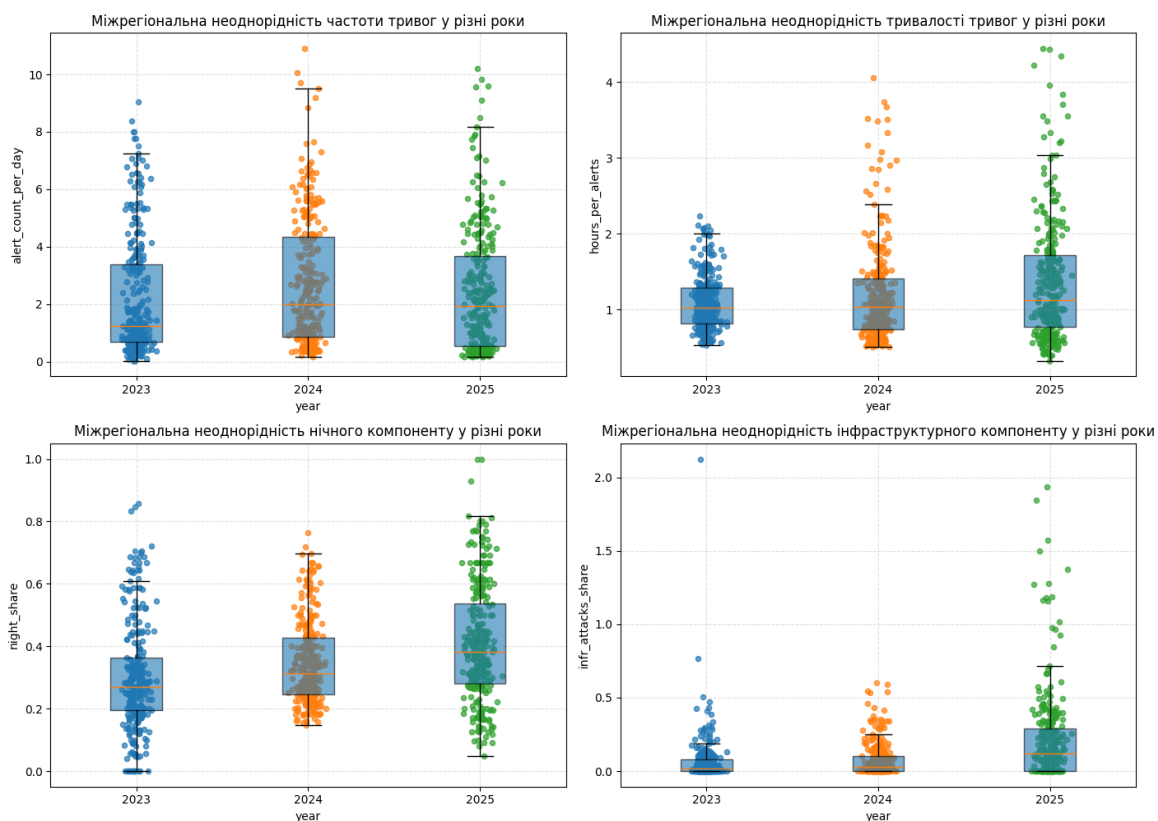


Рисунок 2.3 – Міжрегіональна неоднорідність показників інтенсивності та структури повітряних тривог у 2023–2025 роках.

Таким чином, проведений описовий аналіз показав, що у 2023–2025 роках динаміка повітряних тривог в Україні була неоднорідною як у часовому, так і в регіональному вимірі (рис. 2.4). Відмінності між регіонами формувалися не лише за загальною інтенсивністю тривог, а й за структурою режиму, насамперед за часовим компонентом і зв'язком тривог з інфраструктурними атаками. Це стало підставою для переходу до постановки, у якій основною одиницею аналізу є не регіон як такий, а пара регіон-рік. Такий підхід дозволяє фіксувати зміну стану регіональної системи в часі та надалі аналізувати переходи між виявленими режимами ризику.

region_eng	alert_number_pd	alert_hours_pd	hours_per_alert	night_share	infr_attacks_per_alert_share
Dnipropetrovsk	6.11	5.26	0.86	0.28	0.17
Kharkiv	5.99	6.25	1.04	0.26	0.20
Donetsk	5.51	9.23	1.68	0.28	0.51
Zaporizhzhia	5.35	5.69	1.06	0.27	0.12
Sumy	4.62	7.28	1.58	0.22	0.27
Mykolaiv	3.78	3.17	0.84	0.30	0.06
Kherson	3.64	3.16	0.87	0.31	0.30
Poltava	3.46	4.40	1.27	0.27	0.03
Kirovohrad	3.35	3.48	1.04	0.31	0.01
Chernihiv	2.64	3.83	1.45	0.29	0.15
Kyiv	1.98	2.84	1.43	0.36	0.07
Odesa	1.97	1.73	0.88	0.37	0.11
Cherkasy	1.87	2.12	1.13	0.33	0.05
Kyiv city	1.24	1.43	1.15	0.43	0.22
Vinnytsia	1.05	1.44	1.37	0.48	0.05
Zhytomyr	1.02	1.64	1.61	0.48	0.03
Khmelnysky	0.79	1.04	1.32	0.52	0.06
Volyn	0.77	0.95	1.24	0.35	0.01
Lviv	0.68	0.69	1.01	0.36	0.07
Rivne	0.63	0.74	1.17	0.41	0.01
Ternopil	0.58	0.67	1.15	0.41	0.02
Chernivtsi	0.52	0.65	1.25	0.39	0.03
Ivano-Frankivsk	0.52	0.52	1.00	0.36	0.07
Zakarpattia	0.45	0.49	1.10	0.32	0.00

Рисунок 2.4 – Сукупні показники інтенсивності та структури повітряних тривог за регіонами у 2023–2025 роках

2.3 Формування багатовимірних річних профілів типу регіон-рік

Для задачі кластеризації в підсумку було відібрано п'ять ознак, які формують фінальний річний профіль кожного об'єкта: `alert_count_per_day`, `hours_per_alerts`, `night_alerts_proportion`, `operational-hours_alert_share`, `infr_attacks_per_alert`. Ознака `alert_hours_per_day` не була включена, оскільки вона функціонально пов'язана з `alert_count_per_day` та `hours_per_alerts`. Тому в роботі вона використовувалася переважно для описового аналізу та подальшої

інтерпретації результатів, але не як окрема вісь кластеризації.

Таким чином, для кожного об'єкта типу регіон-рік було сформовано масив $X_{r,y} \in \mathbb{R}^{12 \times 5}$, де 12 відповідає кількості місяців у році, а 5 – кількості ознак, відібраних для кластеризації. Після об'єднання всіх років було сформовано єдиний масив розмірності $72 \times 12 \times 5$: $X \in \mathbb{R}^{72 \times 12 \times 5}$.

Саме цей набір використовується під час побудови багатовимірних часових рядів для кластеризації. Його вибір зумовлений тим, що він одночасно відображає три важливі аспекти режиму, а саме інтенсивність тривоги, часову структуру тривожного навантаження, зв'язок із подіями інфраструктурного ураження (рис. 2.5).

	region_eng	period_month	alert_count_per_day	hours_per_alerts	night_alerts_proportion	operational-hours_alert_share	infr_attacks_per_alert
0	Cherkasy	2023-01-31	0.870968	1.356667	0.259259	0.629630	0.000000
1	Cherkasy	2023-02-28	1.178571	1.356667	0.212121	0.575758	0.000000
2	Cherkasy	2023-03-31	1.129032	1.209714	0.171429	0.542857	0.000000
3	Cherkasy	2023-04-30	0.333333	0.880000	0.200000	0.500000	0.200000
4	Cherkasy	2023-05-31	1.225806	1.812895	0.552632	0.368421	0.000000
...	...	...	...	...	...	...	...
859	Zhytomyr	2025-08-31	0.741935	1.652174	0.347826	0.565217	0.000000
860	Zhytomyr	2025-09-30	1.033333	0.934516	0.516129	0.387097	0.129032
861	Zhytomyr	2025-10-31	1.000000	0.934516	0.516129	0.387097	0.000000
862	Zhytomyr	2025-11-30	1.133333	0.690000	0.617647	0.235294	0.000000
863	Zhytomyr	2025-12-31	1.548387	0.975625	0.437500	0.479167	0.041667

Рисунок 2.5 – Приклад записів ознак відібраних для кластеризації

2.4 Підготовка даних до кластеризації та вибір фінальної конфігурації

Оскільки побудовані ознаки мають різні одиниці вимірювання та різні числові масштаби, перед кластеризацією їх необхідно привести до вигляду можливого до порівняння. Без цього змінні з більшими абсолютними значеннями могли б непропорційно впливати на результат групування.

У роботі використовується глобальна стандартизація ознак по всьому об'єднаному набору річних профілів за допомогою StandardScaler з бібліотеки scikit-learn, Python. Це означає, що стандартизація виконується не окремо для

кожного регіону і не окремо для кожного року, а для всіх об'єктів разом. Це важливо, оскільки такий підхід зберігає можливість порівнювати не лише форму динаміки, а й відносний рівень ознак у спільному просторі всіх об'єктів [21].

Якби кожен річний профіль масштабувався окремо, модель переважно порівнювала б форму динаміки, але частково втрачала б інформацію про те, які регіони та роки мають вищий або нижчий загальний рівень навантаження. Для поставленої задачі це небажано, оскільки режим ризику визначається не лише характером зміни показників упродовж року, а й їхнім відносним рівнем.

Після стандартизації було проведено порівняння кількох конфігурацій кластеризації часових рядів за адаптованим критерієм стабільності Stadion, що є різницею між міжкластерною стабільністю (StabB) та внутрішньокластерною стабільністю (StabW) [15].

Міжкластерна стабільність показує, наскільки добре базове розбиття відтворюється після малих збурень даних. Якщо після додавання невеликого шуму алгоритм формує подібні кластери, це свідчить про стійкість знайденої структури. Тоді як, внутрішньокластерна стабільність використовується для перевірки того, чи не містять уже знайдені кластери стабільної прихованої підструктури. Якщо всередині кластера можна стабільно виділити додаткові підкластери, це означає, що початкове розбиття може бути недостатньо деталізованим або внутрішньо неоднорідним [15].

Для коректного порівняння всі алгоритми оцінювалися в однаковій експериментальній схемі. Однаковими для всіх методів були: вхідний стандартизований масив $X \in \mathbb{R}^{72 \times 12 \times 5}$, набір ознак, спосіб попередньої обробки, критерій оцінювання Stadion, схема перевірки стійкості через малі шумові збурення та обмеження на мінімальний розмір кластера. Водночас, для кожного алгоритму варіювалися його власні гіперпараметри, зокрема кількість кластерів та параметри, специфічні для відповідного методу.

Під час оцінювання стабільності використовувалися наступні параметри: $\text{eps} = (0.02, 0.03, 0.04)$, $\text{Omega} = (2, 3)$, $\text{D_between} = 12$, $\text{D_within} = 6$.

Ці параметри не є параметрами фінальної кластерної моделі, а задають протокол перевірки стабільності. Параметр `eps` визначає рівень малого шумового збурення стандартизованих даних. Використання кількох значень дозволяє не робити висновок на основі лише одного випадково обраного рівня шуму, а оцінювати стійкість конфігурації для кількох близьких рівнів збурення.

Параметр `D_between` визначає кількість збурених копій даних, які використовуються для оцінювання міжкластерної стабільності. Тобто для кожної конфігурації алгоритм багаторазово запускався на збурених версіях вихідного масиву, а отримані розбиття порівнювалися з базовим розбиттям. Параметр `D_within` визначає кількість повторень для оцінювання внутрішньокластерної стабільності, що є обчислювально складнішим за `D_between`, адже потребує додаткового підкластерування всередині кожного вже знайденого кластера [15].

Множина `Omega` задає, на яку кількість підкластерів перевіряється можливе внутрішнє розбиття кожного кластера під час обчислення `StabW` [15]. Тобто, для кожного знайденого кластера перевіряється, чи можна стабільно розділити його ще на 2 або 3 підкластери. Оскільки вибірка містить 72 об'єкти, така множина є достатньою для перевірки прихованої підструктури без надмірного дроблення кластерів.

Також під час відбору конфігурацій було задано мінімально допустимий розмір кластера не менше двох об'єктів. Це обмеження потрібне для того, щоб уникати конфігурацій, у яких окремі режими зводяться до одиничних випадків і погано піддаються змістовній інтерпретації.

Для забезпечення відтворюваності експериментів було зафіксовано базове значення генератора випадкових чисел `base_seed = 42`. Це дозволяє повторити процес збурення даних, ініціалізації алгоритмів і порівняння конфігурацій. Таким чином, фінальна модель обиралася не за результатом одного запуску, а за повторюваним стабільнісним протоколом.

Для `Soft-DTW` параметр `gamma` не задавався довільно. Його попередня

оцінка обчислювалася на об'єднаному масиві річних профілів за допомогою функції `tslearn.metrics.gamma_soft_dtw` [23]. Обчислення повторювалося для кількох значень `random_state`, а як фінальне значення для подальшого перебору конфігурацій бралася медіана отриманих оцінок. Такий підхід зменшує залежність результатів від одного випадкового запуску.

На етапі вибору фінальної конфігурації було порівняно кілька груп методів, які відображають різні припущення щодо подібності річних профілів. Euclidean K-Means використано як базовий варіант, що порівнює значення показників у відповідних місяцях без еластичного вирівнювання. DTW-KMeans включено для перевірки того, чи покращує якість кластеризації врахування локальних часових зсувів у динаміці профілів. SoftDTW-KMeans розглянуто як згладжений варіант DTW, що дозволяє більш м'яко порівнювати річні профілі та контролювати ступінь гнучкості через параметр `gamma`. Як додатковий контрольний підхід використано агломеративну кластеризацію для векторизованого подання профілів.

Після цього для кожної групи методів було виконано сітковий перебір параметрів (`grid search`). Для кожної кандидатної конфігурації спочатку будувалося базове кластерне розбиття, далі створювалися малі шумові збурення стандартизованого масиву, після чого повторно виконувалася кластеризація та оцінювалася стабільність отриманих розбиттів. Підсумкове значення Stadion усереднювалося за різними рівнями збурення, що дозволило порівнювати конфігурації за їхньою загальною стійкістю.

За результатами порівняння альтернативних конфігурацій моделей кластеризації, найкраще значення критерію Stadion продемонструвала конфігурація SoftDTW-KMeans з такими параметрами: `n_clusters = 5`, `n_init = 2`, `max_iter_barycenter = 30`, `gamma = 270.34`.

Вона була обрана як фінальна для побудови річних режимів ризику (табл. 2.1, рис. 2.6). Для неї середнє значення критерію становило 0.078717, стандартне відхилення – 0.01483, а мінімальний розмір кластера – 6.

Таблиця 2.1 – Підсумкова таблиця найкращих конфігурацій для алгоритмів.

Метод	Найкраща конфігурація	Stadion_mean	Stadion_std	Мінімальний розмір кластеру	StabB_mean	StabW_mean
SoftDTW-KMeans	'n_clusters': 5, 'n_init': 2, 'max_iter_barycenter': 30, 'gamma': 270.34	0.079	0.015	6	0.957	0.878
Euclid-KMeans	'n_clusters': 4, 'n_init': 10, 'max_iter': 300, 'tol': 0.0001, 'init': 'k-means++', 'random_state': 42	0.058	0.036	9	1	0.942
DTW-KMeans	'n_clusters': 4, 'n_init': 5, 'max_iter_barycenter': 30	0.037	0.012	5	0.95	0.914
Agglomerative	'n_clusters': 4, 'linkage': 'ward'	-0.128	0.029	10	0.846	0.973

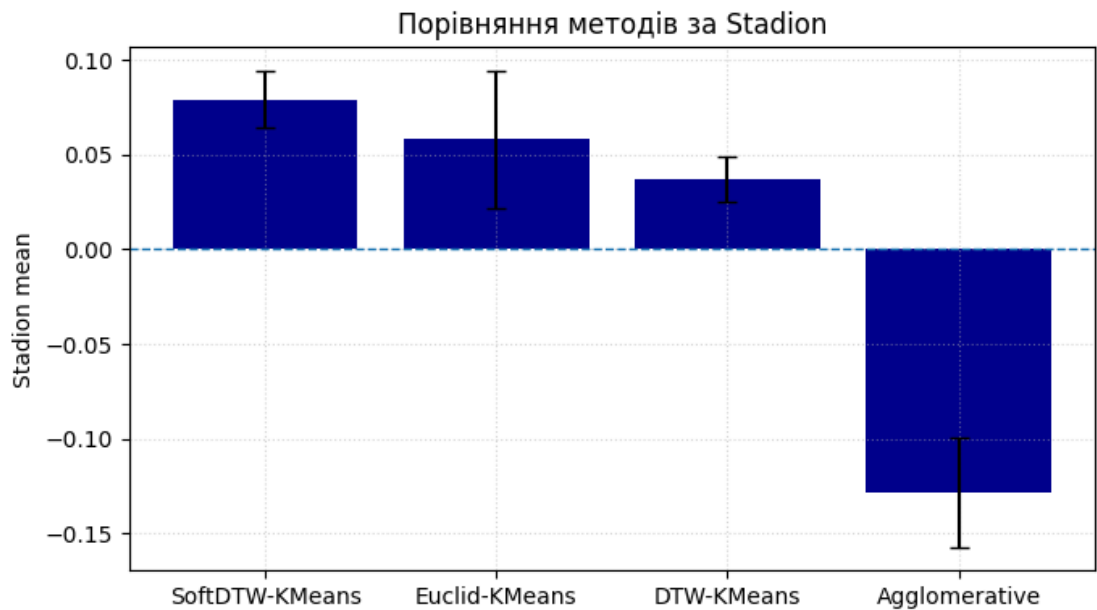


Рисунок 2.6 – Порівняння фінальних конфігурацій різних методів за Stadion mean.

Вибір SoftDTW-KMeans є змістовно узгодженим із поставленою задачею. Оскільки об'єктом аналізу є багатовимірні річні профілі, важливо враховувати не лише рівень ознак у фіксованих місяцях, а й загальну форму

динаміки. Цей метод показав найкраще значення інтегрального критерію Stadion серед розглянутих кандидатів і водночас забезпечив змістовно інтерпретовану кластерну структуру. Порівняно з Euclidean K-Means, SoftDTW-KMeans краще враховує можливі локальні відмінності у формі річних профілів. Порівняно з DTW-KMeans, згладжений варіант дозволяє уникнути надмірно жорсткого вибору одного оптимального шляху вирівнювання. Агломеративна кластеризація була корисною як додатковий контрольний підхід, однак її застосування до векторизованих профілів меншою мірою відповідає часовій природі задачі.

Висновки до розділу 2

У другому розділі було сформовано інформаційну та методичну основу для практичної реалізації запропонованого підходу до кластеризації регіональних профілів повітряних тривог. На основі даних за 2023–2025 роки було сформовано систему показників, що описують інтенсивність і структуру тривог, а також підготовлено фінальний набір ознак для побудови річних багатовимірних профілів. Проведений описовий аналіз показав, що динаміка повітряних тривог є неоднорідною як між різними регіонами, так і між роками. Регіони відрізняються не лише за загальною інтенсивністю тривог, а й за середньою тривалістю епізодів, добовою структурою та інфраструктурним компонентом ризику.

Для подальшої кластеризації кожен пару регіон-рік було подано у вигляді багатовимірного часового профілю довжиною 12 місяців. До фінального набору ознак було включено показники, які відображають частоту тривог, середню тривалість одного епізоду, частку нічних тривог, частку тривог у денні операційні години та кількість інфраструктурних атак на одну тривогу. Такий підхід дозволив розглядати кожен регіон у конкретному році

як окремий стан регіональної системи ризику.

Після стандартизації даних виконано порівняння кількох конфігурацій кластеризації: Euclidean K-Means, DTW-KMeans, SoftDTW-KMeans та агломеративної кластеризації. Оцінювання здійснювалося з урахуванням критерію стабільності Stadion, оскільки в задачі відсутня еталонна розмітка, а тому важливо було обрати не лише формально якісне, а й стійке до малих збурень розбиття.

За результатами порівняння фінальною обрано конфігурацію SoftDTW-KMeans із п'ятьма кластерами. Метод м'якого динамічного викривлення часу дозволяє враховувати подібність річних профілів з урахуванням можливих локальних відмінностей у динаміці, але водночас не зводить аналіз лише до жорсткого поелементного порівняння місячних значень. Отримана конфігурація стала основою для подальшої інтерпретації виявлених режимів ризику, аналізу їхньої міжрічної динаміки та просторового розподілу.

РОЗДІЛ 3 РЕЗУЛЬТАТИ КЛАСТЕРИЗАЦІЇ, ІНТЕРПРЕТАЦІЯ РЕЖИМІВ РИЗИКУ ТА АНАЛІЗ ЇХ ДИНАМІКИ

3.1 Результати фінальної кластеризації та характеристика виявлених режимів ризику

За результатами фінальної кластеризації методом SoftDTW-KMeans було виділено п'ять кластерів. Їх загальні розміри та розподіл за роками наведено в таблиці 3.1 та рисунку 3.1.

Таблиця 3.1 – Розміри отриманих кластерів та розподіл за роками

Кластер	Кількість об'єктів	2023	2024	2025
0	9	0	0	9
1	33	17	11	5
2	6	0	0	6
3	18	7	9	2
4	6	0	4	2

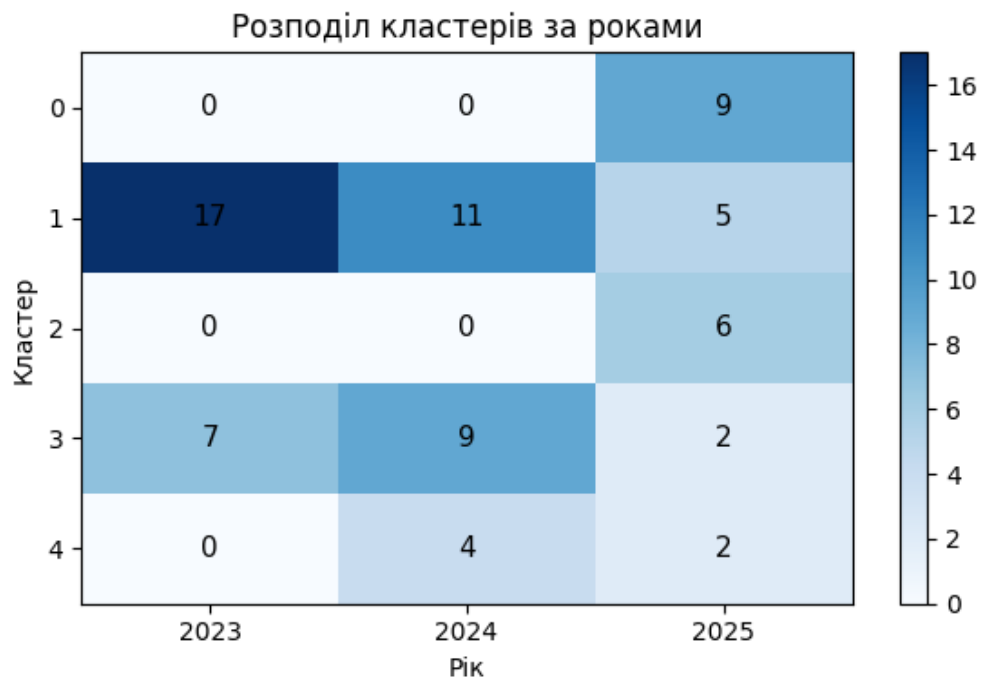


Рисунок 3.1 – Розподіл кластерів за роками

Найбільшим є кластер 1, до якого належать 33 об'єкти, що свідчить про

наявність домінуючого річного режиму, який повторюється в даних найчастіше. Водночас кластери 2 і 4 мають найменше об'єктів, лише по 6, тобто відповідають більш специфічним режимам, які спостерігаються рідше.

Отриманий розподіл показує, що структура режимів ризику в досліджуваний період була неоднорідною за роками. У 2023 році спостерігалися лише два режими – кластери 1 і 3. У 2024 році до них додався кластер 4. Лише у 2025 році в даних з'явилися всі п'ять режимів, зокрема кластери 0 і 2, які не спостерігалися у попередні роки.

Таким чином, результати спільної кластеризації річних профілів за 2023–2025 роки свідчать про наявність вираженого міжрічного ефекту. Структура режимів ризику 2023–2025 роках не залишилась сталою, а у 2025 році стала менш однорідна. Це підтверджує доцільність аналізу не лише окремих регіонів, а й переходів між режимами в часі.

Для інтерпретації отриманих кластерів було проаналізовано їх узагальнені середні профілі за п'ятьма ознаками (рис. 3.2–3.3) та середню помісячну динаміку ознак у межах кожного кластера в різні роки (рис. 3.4). Загальні середні значення дозволяють проаналізувати відносне положення кластера в просторі ознак, а помісячні профілі – форму динаміки протягом року.

Результати показали, що режими ризику істотно відрізняються не лише за інтенсивністю тривожного навантаження, а й за його внутрішньою структурою. Найвищу середню частоту тривог мають кластери 2 ($\text{alert_count_per_day} = 4.81$) та 3 ($\text{alert_count_per_day} = 4.97$). Також кластер 2 характеризується найбільшою середньою тривалістю однієї тривоги ($\text{hours_per_alerts} = 1.76$) та найвищим навантаженням на інфраструктуру ($\text{infr_attacks_per_alert} = 0.55$). Кластери 0 і 4, навпаки, характеризуються низькою частотою тривог ($\text{alert_count_per_day} < 1$), але підвищеною часткою нічних епізодів. Кластер 1 займає проміжне, відносно помірне положення за більшістю ознак.



Рисунок 3.2 – Узагальнені середні профілі кластерів у вихідному масштабі

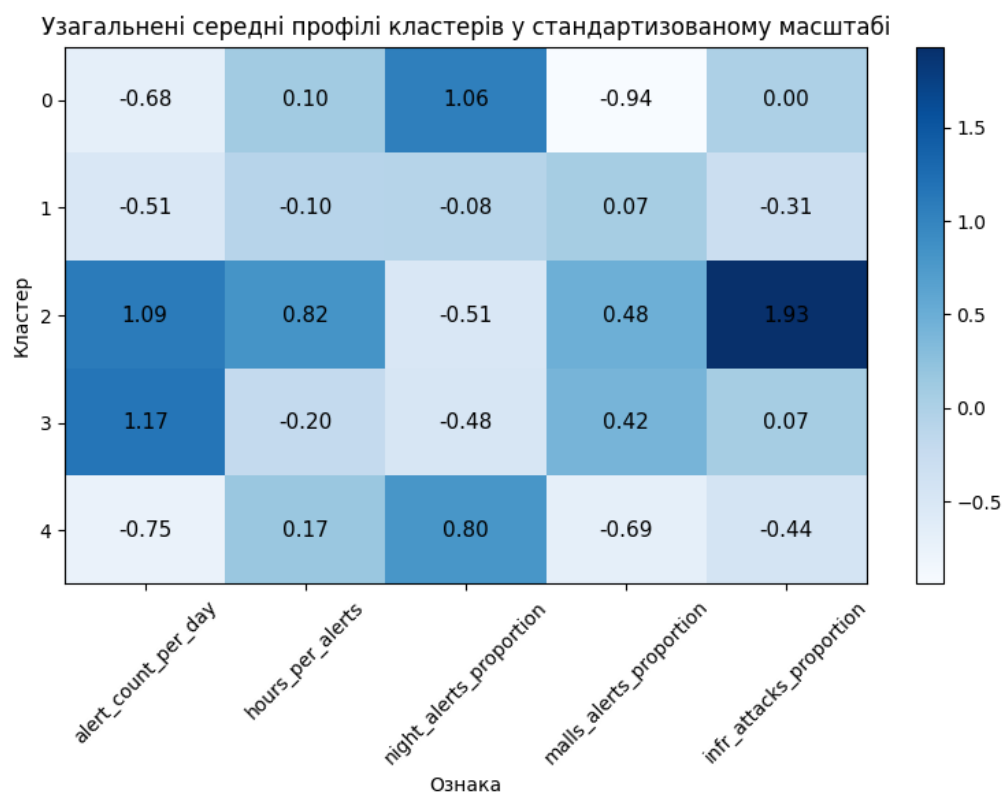


Рисунок 3.3 – Узагальнені середні профілі кластерів у стандартизованому масштабі

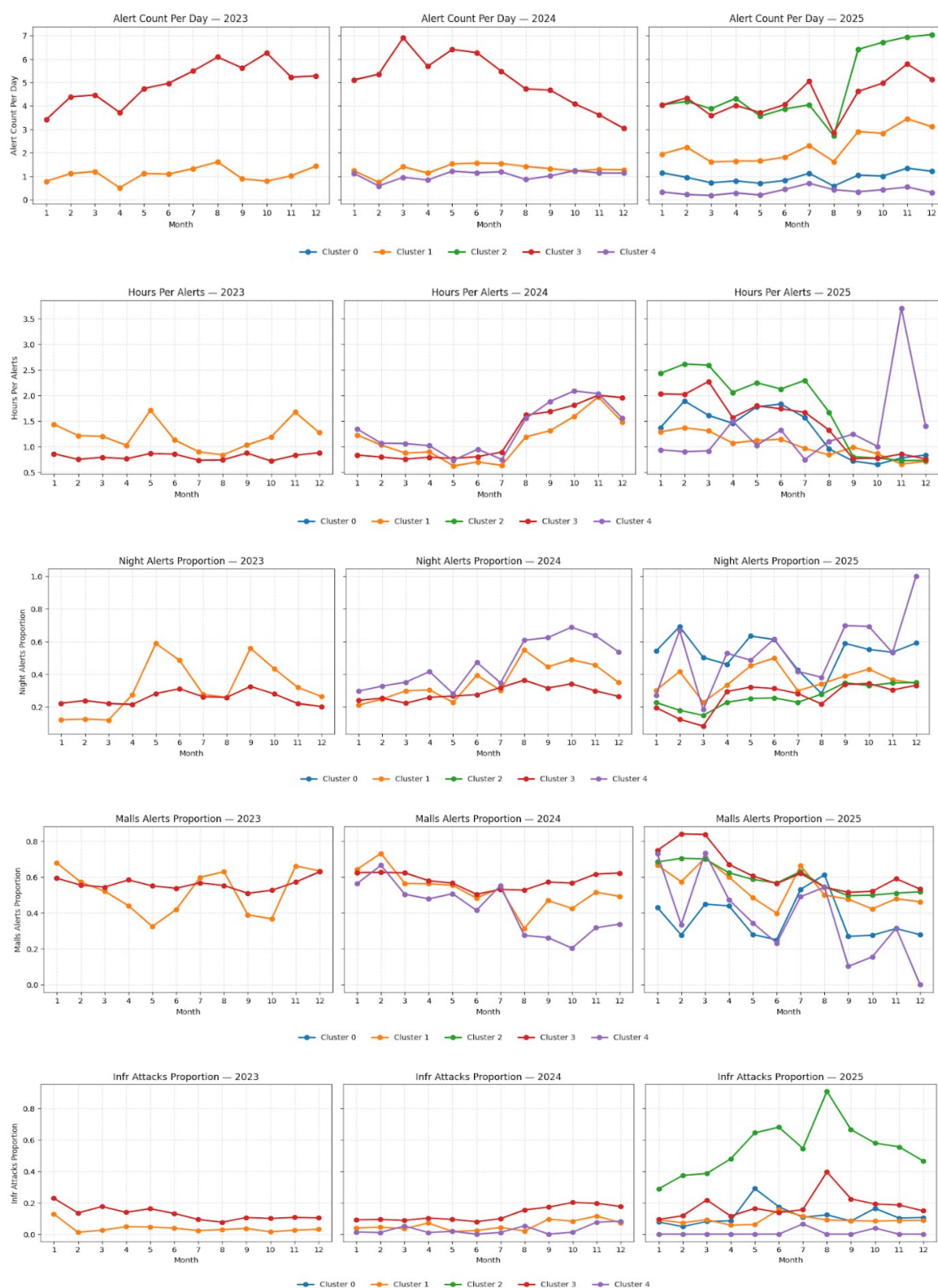


Рисунок 3.4 – Середня помісячна динаміка ознак у межах кожного кластера в різні роки (у вихідному масштабі)

На основі узагальнених і помісячних профілів кластери можуть бути інтерпретовані як наступні режими ризику.

1. Кластер 0 – нічно зміщений режим із низькою частотою та

помірним інфраструктурним ризиком

- Кластер 0 з'являється тільки у 2025 році й характеризується нижчою частотою, але найвищою часткою нічних епізодів та нижчою часткою тривог у денному операційному інтервалі. Середня тривалість однієї тривоги тут є дещо вищою, ніж у базовому кластері 1, а інфраструктурний компонент близький до середнього і перевищує значення відносно схожого кластера 4.
- Аналізуючи щомісячну динаміку за 2025 рік, можна побачити помітне зменшення тривалості тривог під кінець року.

2. Кластер 1 – базовий відносно помірний режим ризику

- Кластер 1 є серед з найбільшим за кількістю об'єктів, і характеризується порівняно невисокою частотою та помірною середньою тривалістю тривог. Для кластеру характерними є середня частка нічних тривог і низький інфраструктурний компонент. Частка тривог у денному операційному інтервалі є дещо вищою, ніж у кластерів 0 і 4. Помісячна динаміка є відносно рівною, з помірними змінами інтенсивності протягом року.
- Тому цей кластер доцільно інтерпретувати як базовий або фоновий режим з нижчим відносним рівнем ризику. У 2023-2024 роках цей режим був домінуючим, а у 2025 році його частка зменшилася.

3. Кластер 2 – найбільш напружений інтенсивно-інфраструктурний режим

- Кластер 2 має дуже високу частоту тривог, найбільшу середню тривалість одного епізоду та найвищий інфраструктурний компонент. Водночас частка нічних тривог у ньому не є максимальною.
- Помісячна динаміка кластеру показує, що в другій половині року посилюється навантаження за кількістю тривог, водночас із помітним зменшенням середньої тривалості однієї події.
- Кластер 2 з'являється лише у 2025 році.

4. Кластер 3 – інтенсивний режим частих та відносно коротких

денних тривог

- Кластер 3 характеризується дуже високою середньою частотою тривог при найнижчій порівняно з іншими кластерами середній тривалості одного епізоду. Частка нічних тривог у ньому близька до кластеру 2 та є нижчою за кластери 0 і 4. Тоді як, частка тривог у денному операційному інтервалі, навпаки, є підвищеною. Інфраструктурний компонент тут не є максимальним, але перевищує значення базового кластера 1.
- Цей режим був одним із двох основних у 2023 році, зберігався у 2024 році і частково залишився у 2025 році.

5. Кластер 4 – знижено інтенсивний, але структурно напружений режим

- Кластер 4 з'являється з 2024 року та є близьким до кластера 0 за низькою частотою тривог і підвищеною часткою нічних епізодів. Однак має значно нижчий та, водночас, найменший інфраструктурний компонент.
- Його можна розглядати як окремий режим структурної напруженості, де ризик формується не стільки через високу інтенсивність тривог, а через несприятливий часовий розподіл.

Таким чином, отримані кластери показують, що режими ризику відрізняються щонайменше за трьома вимірами: за інтенсивністю навантаження, за часовою структурою, зокрема нічним або денним зміщенням, та за інфраструктурним компонентом. Це означає, що ризиковість річного профілю не зводиться до однієї ознаки. Один режим може бути напруженим через дуже високу частоту тривог, інший – через довші епізоди та зв'язок з інфраструктурними подіями, або через несприятливу нічну структуру навіть за відносно нижчої загальної частоти.

3.2 Формалізація річного індексу ризику повітряних тривог

У результаті кластеризації кожен об'єкт типу регіон-рік було віднесено до одного з п'яти кластерів. У межах цієї роботи кластер розуміється як прямий результат алгоритму групування, тоді як режим ризику є змістовною інтерпретацією цього кластера з точки зору регіонального навантаження в умовах війни. Оскільки числові мітки кластерів не мають природного порядку, після кластеризації було виконано додаткове впорядкування режимів за відносною ризиковістю. Це впорядкування не змінює результатів кластеризації і не створює нових груп, а дозволяє інтерпретувати вже виявлені режими від менш напружених до більш напружених.

Для цього після виконання кластеризації було побудовано допоміжний річний індекс ризику повітряних тривог на рівні кожного об'єкта типу регіон-рік. Спочатку для кожного об'єкта були розраховані річні агрегати: сумарна кількість тривог, сумарна тривалість тривог, кількість нічних тривог, кількість тривог у денні операційні години, кількість інфраструктурних атак і загальна кількість днів у відповідному році. На їх основі було побудовано річні показники для регіону r у році y :

$$\begin{aligned} alert_count_per_day_{r,y} &= \frac{alerts_total_{r,y}}{days_total_{r,y}}, \\ alert_hours_per_day_{r,y} &= \frac{alerts_hours_total_{r,y}}{days_total_{r,y}}, \\ night_alerts_proportion_{r,y} &= \frac{night_alerts_total_{r,y}}{alerts_total_{r,y}}, \\ infra_attacks_per_alert_{r,y} &= \frac{infra_attacks_total_{r,y}}{alerts_total_{r,y}}, \end{aligned}$$

Оскільки ці показники мають різні одиниці вимірювання та числові масштаби, перед агрегуванням було виконано стандартизацію:

$$Z(v_{r,y}) = \frac{v_{r,y} - \mu_v}{\sigma_v}$$

де $v_{r,y}$ – значення певної річної ознаки для регіону r у році y , μ_v – середнє значення цієї ознаки за всіма об'єктами, σ_v — стандартне відхилення.

Далі було сформовано дві складові річного індексу ризику:

- 1) інтенсивність навантаження (*exposure_score*) – відображає, наскільки часто виникали тривоги та скільки часу вони займали в середньому за день:

$$E_{r,y} = \frac{1}{2} \times (Z(alert_count_per_day_{r,y}) + Z(alert_hours_per_day_{r,y})),$$

- 2) структурна напруженість режиму (*structural_score*) – відображає нічне зміщення тривог і зв'язок тривожного режиму з інфраструктурними атаками.

$$S_{r,y} = \frac{1}{2} \times (Z(night_alerts_proportion_{r,y}) + Z(infra_attacks_per_alert_{r,y})),$$

На основі цих двох складових було розраховано річний індекс ризику повітряних тривог (*annual_risk_index*) як їхнє середнє значення:

$$ARI_{r,y} = \frac{1}{2} \times (E_{r,y} + S_{r,y}),$$

де $ARI_{r,y}$ – річний індекс ризику для регіону r у році y , $E_{r,y}$ – показник інтенсивності навантаження, $S_{r,y}$ – показник структурної напруженості.

Важливо зауважити, що показник частки тривог у денні операційні години (*operational-hours_alert_share*) використовувався у фінальному наборі ознак для кластеризації, однак не був включений до базової версії структурної

складової індексу, оскільки цей індекс має відобразити ті структурні характеристики, які найчіткіше інтерпретуються як додаткове навантаження на регіон, а не описати весь добовий розподіл тривог. Саме тому до нього було включено нічний та інфраструктурний компоненти. На відміну від тривог в операційні години, нічні тривоги розглядаються як більш несприятливий часовий компонент, оскільки вони порушують період відпочинку та сну, що пов'язується з підвищеним психофізіологічним навантаженням.

3.3 Упорядкування режимів за відносною ризиковістю

Для переходу від рівня окремих об'єктів типу регіон-рік до рівня режимів ризику для кожного кластера були обчислені медіанні значення відповідних показників:

$$\begin{aligned} exposure_median_k &= median\{E_{r,y} : z_{r,k} = k\}, \\ structural_median_k &= median\{S_{r,y} : z_{r,k} = k\}, \\ annual_risk_index_median_k &= median\{ARI_{r,y} : z_{r,k} = k\}, \end{aligned}$$

де $z_{r,y}$ – мітка кластера, до якого алгоритм відніс регіон r у році y ; k – номер відповідного кластера.

Медіана була обрана тому, що вона є стійкішою до окремих екстремальних значень, ніж середнє значення. Після цього кластери було впорядковано за зростанням $annual_risk_index_median_k$, а отриманий порядок було використано для присвоєння кожному кластеру рангу відносної ризиковості $cluster_severity_rank_k$ (табл. 3.2, рис. 3.5). Менший ранг відповідає менш напруженому режиму, а більший – більш напруженому.

Таблиця 3.2 – Упорядкування кластерів за показниками режиму ризику (від меншого до більшого)

Кластер	Ранг ризикованості	exposure median	structural median	annual_risk_index_median
1	0	-0.714	-0.298	-0.438
4	1	-0.769	0.442	-0.114
0	2	-0.763	0.837	0.054
3	3	1.086	-0.461	0.332
2	4	1.633	0.446	0.971

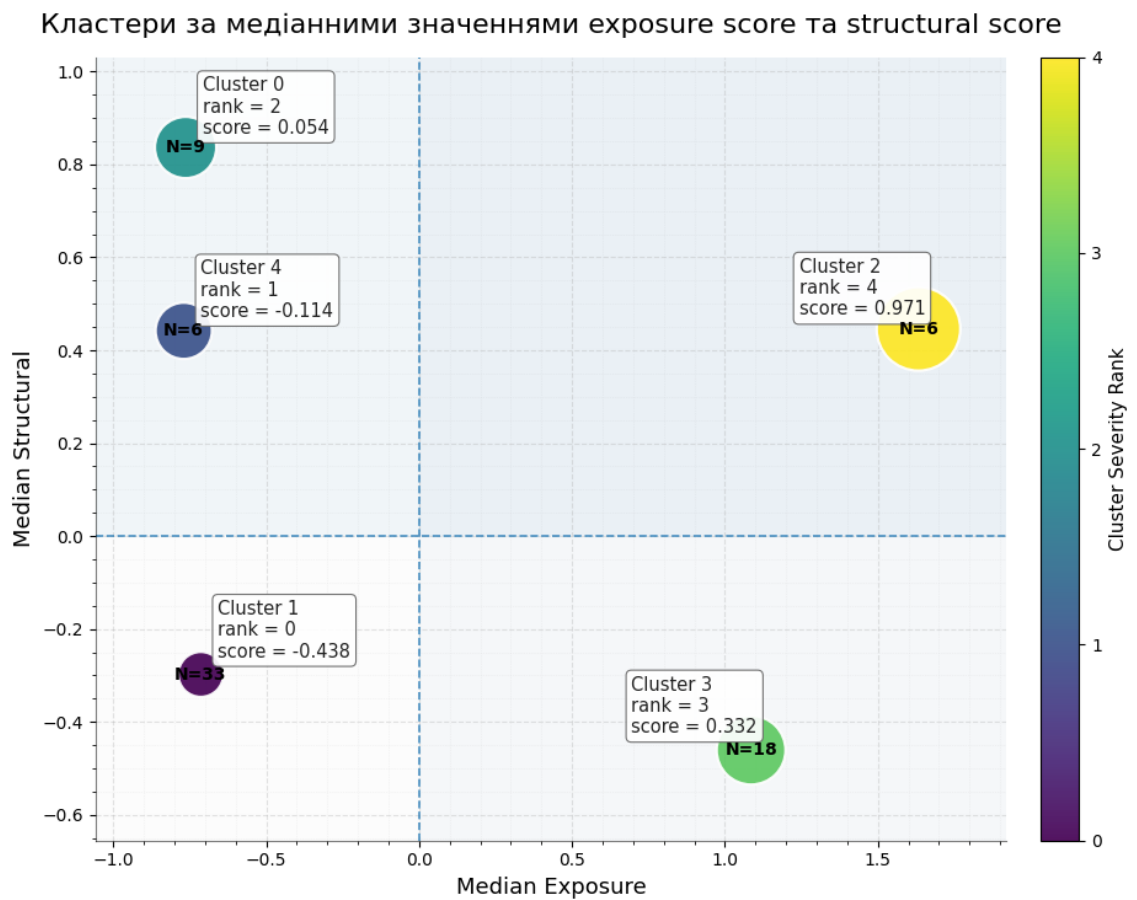


Рисунок 3.5 – Упорядкування кластерів за показниками режиму ризику (від меншого до більшого)

За результатами розрахунку медіанного значення annual_risk_index для кожного кластера режими впорядковуються від менш до більш ризикових у такій послідовності (cluster_severity_rank): 1, 4, 0, 3, 2. Найменш ризиковим є кластер 1, який відповідає базовому відносно помірному режиму. Найбільш напруженим є кластер 2, у якому поєднуються висока інтенсивність

навантаження та структурна напруженість режиму. Проміжні режими відрізняються тим, що ризиковість у них формується різними способами: або за рахунок більшої нічної та структурної напруженості, або за рахунок вищих показників інтенсивності.

3.4 Матриці міжрічних переходів між режимами ризику

Для аналізу міжрічної динаміки регіонів було побудовано матриці переходів між режимами ризику [31; 32]. Вони відображають, як змінювався кластер кожного регіону між сусідніми роками, а саме між 2023 і 2024 роками та між 2024 і 2025 роками. Матриці подано у двох формах: у вигляді абсолютної кількості переходів між кластерами (табл. 3.3) та у вигляді часток переходів за кожним вихідним кластером (табл. 3.4).

Таблиця 3.3 – Загальна кількість переходів між кластерами

З кластеру	0	1	2	3	4
До кластеру					
1	5	14	1	2	6
3	0	2	5	9	0
4	4	0	0	0	0

Таблиця 3.4 – Частки переходів між кластерами

З кластеру	0	1	2	3	4
До кластеру					
1	0.179	0.5	0.036	0.071	0.214
3	0	0.125	0.312	0.562	0
4	1	0	0	0	0

Якщо $z_{r,y}$ позначає режим ризику регіону r у році y , то перехід між двома послідовними роками задається впорядкованою парою: $(z_{r,y}, z_{r,y+1})$.

Кількість переходів із режиму a у році y до режиму b у році $y+1$ визначається наступним чином [32]:

$$n_{ab}^{(y,y+1)} = \sum_{r=1}^N I(z_{r,y} = a, z_{r,y+1} = b)$$

де N – кількість регіонів, для яких наявні спостереження в обох послідовних роках, а I – індикаторна функція, що дорівнює 1, якщо умова виконується, і 0 – в іншому випадку.

На основі цих значень формується матриця кількостей переходів:

$$N^{(y,y+1)} = [n_{ab}^{(y,y+1)}]_{K \times K},$$

де K – кількість кластерів (режимів ризику).

Крім абсолютних кількостей, для кожного вихідного режиму було обчислено емпіричні частки переходів. Частка переходів із режиму a до режиму b визначається як відношення кількості регіонів, які перейшли з режиму a до режиму b , до загальної кількості регіонів, що перебували в режимі a у початковому році:

$$p_{ab}^{(y,y+1)} = \frac{\sum_{r=1}^N I(z_{r,y} = a, z_{r,y+1} = b)}{\sum_{r=1}^N I(z_{r,y} = a)}$$

Для кожного вихідного режиму сума таких часток за всіма можливими режимами переходу дорівнює одиниці:

$$\sum_{b=1}^K p_{ab} = 1$$

Програмно розрахунок матриці абсолютних кількостей переходів N реалізовано через процедуру крос-табуляції впорядкованих пар станів із використанням `pandas.crosstab`. Для розрахунку часток переходів

(ймовірностей, P) використано нормування значень за індексом рядків (параметр `normalize="index"`), що автоматично забезпечило умову, щоб сума часток одиниці для кожного вихідного режиму була рівна одиниці.

Важливо враховувати, що в матрицях вихідними є лише ті кластери, які були присутні в початковому році відповідного інтервалу. У 2023 році спостерігалися лише кластери 1 і 3, у 2024 році – кластери 1, 3 і 4, тоді як кластери 0 і 2 з'явилися лише у 2025 році як результати переходів, а не як вихідні стани. Саме тому у таблицях відсутні рядки для кластерів 0 і 2.

За результатами матриць переходів видно, що найбільш поширеними є переходи: $1 \rightarrow 1$, $3 \rightarrow 3$, $1 \rightarrow 4$, $1 \rightarrow 0$, $3 \rightarrow 2$ та $4 \rightarrow 0$. Для кластера 1 половина переходів зберігає той самий режим у наступному році, тоді як решта переходів розподіляється між кластерами 4, 0, 3 і 2. Для кластера 3 найбільш типовим також є збереження режиму, однак суттєва частка об'єктів переходить до найбільш напруженого кластера 2. Водночас, кластер 4 демонструє іншу поведінку, з переходом усіх регіонів із цього кластера у 2024 році до кластера 0 у 2025 році.

3.5 Аналіз траєкторій регіонів між кластерами та рангами відносної ризиковості у 2023–2025 роках

Для детальнішого аналізу міжрічної динаміки було побудовано таблицю траєкторій регіонів між кластерами у 2023–2025 роках, а також зміну їхніх рангів відносної ризикованості (табл. 3.5, рис. 3.6). Така таблиця дозволяє простежити не лише факт зміни кластера, а й напрям зміни режиму до більш або менш напруженого стану.

Аналіз таблиці траєкторії показує, що у 2023–2024 роках переважала стабільність режимів, де 18 з 24 регіонів (75%) не змінили ранг ризикованості, тоді як 6 регіонів (25%) перейшли до більш напружених режимів. Переходів

до менш ризикованих режимів для цієї траєкторії не зафіксовано. Більшість регіонів або залишалися в базовому кластері 1, або зберігали належність до інтенсивного кластера 3.

Для переходу 2024 до 2025 року структура змін стала більш вираженою: 17 з 24 регіонів (70.8%) перейшли до більш ризикованих режимів, тоді як 5 регіонів (20.8%) не змінили ранг, а лише 2 регіони (8.3%) продемонстрували зниження ризиковості.

Таблиця 3.5 – Траєкторії регіонів між кластерами за 2023–2025 роки та ранги відносної ризикованості

Регіон	Кластер 2023	Кластер 2024	Кластер 2025	Ранг 2023	Ранг 2024	Ранг 2025	Напрямок зміни ризиковості з 2023 до 2024 року	Напрямок зміни ризиковості з 2024 до 2025 року
Cherkasy	1	1	1	0	0	0	без змін	без змін
Chernihiv	1	1	2	0	0	4	без змін	підвищення
Chernivtsi	1	1	4	0	0	1	без змін	підвищення
Dnipropetrovsk	3	3	2	3	3	4	без змін	підвищення
Donetsk	3	3	2	3	3	4	без змін	підвищення
Ivano-Frankivsk	1	1	0	0	0	2	без змін	підвищення
Kharkiv	3	3	2	3	3	4	без змін	підвищення
Kherson	3	3	2	3	3	4	без змін	підвищення
Khmelnitsky	1	4	0	0	1	2	підвищення	підвищення
Kirovohrad	1	3	1	0	3	0	підвищення	зниження
Kyiv	1	1	0	0	0	2	без змін	підвищення
Kyiv city	1	1	0	0	0	2	без змін	підвищення
Lviv	1	1	0	0	0	2	без змін	підвищення
Mykolaiv	3	3	1	3	3	0	без змін	зниження
Odesa	1	1	1	0	0	0	без змін	без змін
Poltava	1	3	3	0	3	3	підвищення	без змін
Rivne	1	1	0	0	0	2	без змін	підвищення
Sumy	3	3	2	3	3	4	без змін	підвищення
Ternopil	1	4	0	0	1	2	підвищення	підвищення
Vinnitsia	1	4	0	0	1	2	підвищення	підвищення
Volyn	1	1	1	0	0	0	без змін	без змін
Zakarpattia	1	1	4	0	0	1	без змін	підвищення
Zaporizhzhia	3	3	3	3	3	3	без змін	без змін
Zhytomyr	1	4	0	0	1	2	підвищення	підвищення

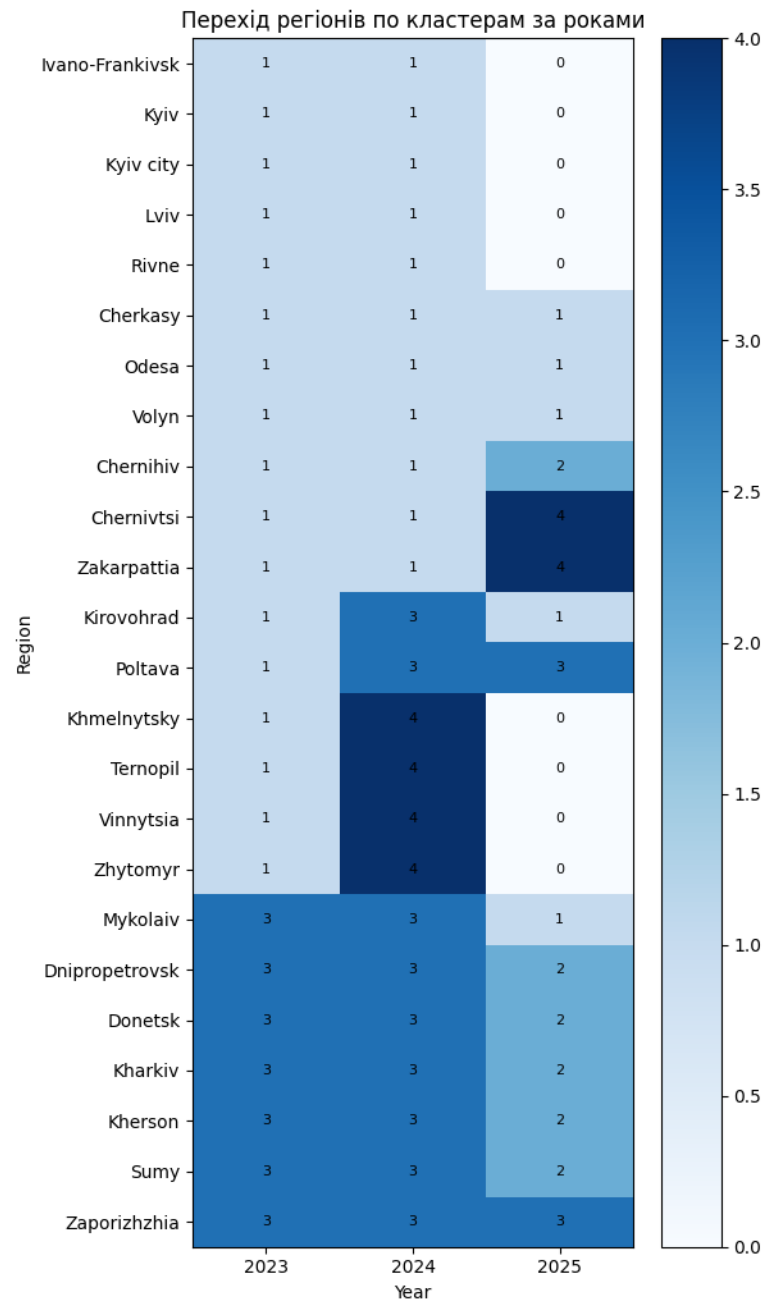


Рисунок 3.6 – Перехід регіонів по кластерам за роками

Серед індивідуальних траєкторій найбільш типовими є $1 \rightarrow 1 \rightarrow 0$, $3 \rightarrow 3 \rightarrow 2$, $1 \rightarrow 4 \rightarrow 0$ та $1 \rightarrow 1 \rightarrow 1$. Траєкторія $1 \rightarrow 1 \rightarrow 0$ відображає збереження базового режиму впродовж 2023–2024 років із подальшим переходом до більш структурно напруженого нічно зміщеного режиму у 2025 році. Траєкторія $3 \rightarrow 3 \rightarrow 2$, навпаки, характеризує збереження інтенсивного режиму з подальшим переходом до найбільш напруженого інтенсивно-інфраструктурного режиму. Траєкторія $1 \rightarrow 4 \rightarrow 0$ відображає поступове зміщення від базового режиму до структурно більш напружених станів.

Таким чином, міжрічна динаміка має асиметричний характер. Перехід від 2023 до 2024 року переважно відображає збереження вже сформованих режимів, тоді як перехід від 2024 до 2025 року значно частіше пов'язаний із входженням регіонів у більш напружені режими ризику. Отже, траєкторії підтверджують, що одна й та сама область у різні роки може перебувати в різних станах регіонального ризику, що й обґрунтовує розгляд регіону як системи зі змінним режимом функціонування.

3.6 Оцінювання часу до першого переходу регіону в більш ризиковий режим

Після побудови матриць міжрічних переходів і аналізу індивідуальних траєкторій регіонів було додатково оцінено час до першого переходу в більш ризиковий режим. Такий підхід дозволяє узагальнити не лише напрям зміни режимів, а й те, на якому саме міжрічному переході для регіону вперше було зафіксовано погіршення.

В межах цього дослідження подією вважається перший перехід регіону до режиму з вищим рангом відносної ризиковості порівняно з попереднім роком. Нехай $z_{r,y}$ позначає ранг режим ризику регіону r у році y , а $s(z_{r,y})$ – ранг відносної ризиковості відповідного режиму. Тоді індикатор погіршення режиму ризику між двома послідовними роками можна задати як:

$$e_r^{(y,y+1)} = \begin{cases} 1, & s_{r,y+1} > s_{r,y} \\ 0, & s_{r,y+1} \leq s_{r,y} \end{cases}$$

де $e_r^{(y,y+1)} = 1$ означає, що регіон перейшов до більш ризикового режиму порівняно з попереднім роком, а $e_r^{(y,y+1)} = 0$ свідчить про відсутність погіршення.

Для кожного регіону визначався час до першого погіршення T_r , тобто

номер першого міжрічного переходу, на якому було зафіксовано перехід до режиму з вищим рангом ризиковості. Для трьохрічного періоду спостереження можливі два переходи: між 2023 і 2024 та 2024 і 2025 роками. Якщо перше погіршення відбулося під час переходу з 2023 до 2024 року, то $T_r = 1$. Якщо під час переходу з 2024 до 2025 або взагалі не сталося до кінця періоду, то $T_r = 2$.

Для оцінювання емпіричної ймовірності залишитися без переходу до більш ризикового режиму було використано оцінку Каплана-Мейєра [18]. У цьому контексті функція виживання $\hat{S}(t)$ інтерпретується як функція збереження стану без погіршення, тобто як ймовірність того, що після t міжрічних переходів регіон ще не перейшов до режиму з вищим рангом ризиковості.

Для практичної реалізації було використано функцію `KaplanMeierFitter` з бібліотеки `lifelines`, та для кожного регіону було сформовано таблицю, у якій зазначено час до першого погіршення, індикатор події та траєкторію рангів ризиковості протягом 2023–2025 років [29]. Узагальнений розподіл регіонів за часом до першого переходу в більш ризиковий режим наведено в таблиці 3.6.

Таблиця 3.6 – Узагальнений розподіл регіонів за часом до першого переходу в більш ризиковий режим

Час до події	Індикатор події	Кількість регіонів	Інтерпретація
1	1	6	Перше погіршення відбулося після переходу з 2023 до 2024 року
2	0	5	Погіршення не відбулося до кінця 2025 року
2	1	13	Перше погіршення відбулося після переходу з 2024 до 2025 року

На основі цих даних було побудовано криву Каплана-Мейєра, яка показує зміну емпіричної ймовірності залишитися без переходу до більш ризикового режиму після кожного міжрічного переходу (рис. 3.7). У таблиці 3.7 наведено відповідні числові значення функції.

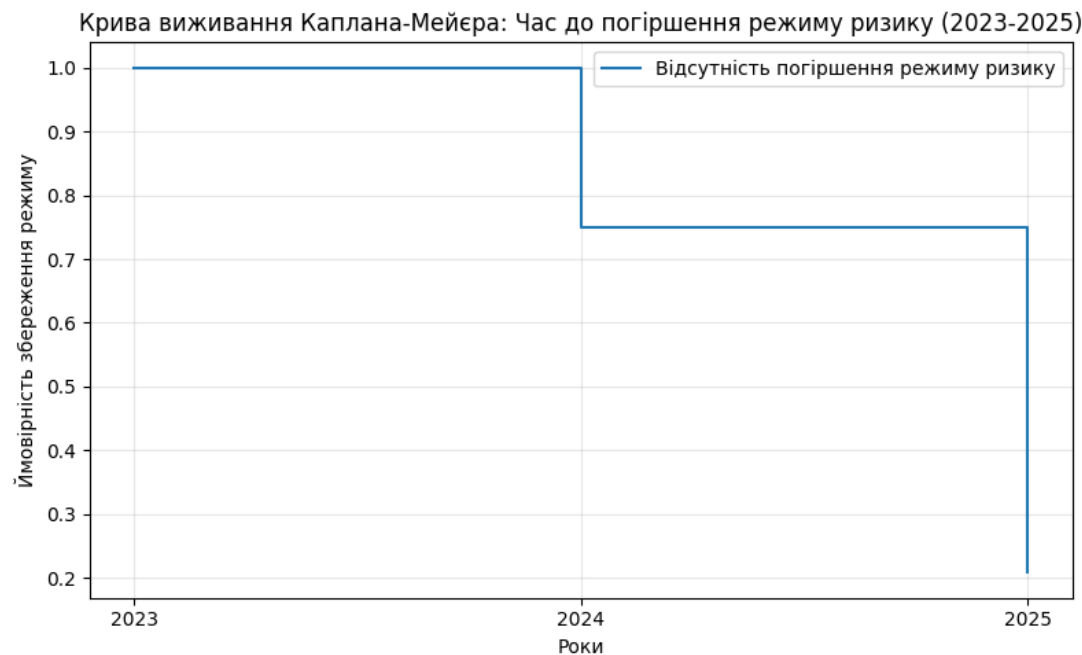


Рисунок 3.7 – Крива виживання Каплана-Мейєра для оцінки часу до погіршення режиму ризику, 2023–2025

Таблиця 3.7 – Оцінені значення функції збереження режиму ризику без погіршення

Кількість міжрічних переходів	Ймовірність залишитися без погіршення	Момент часу
0	1.00	Початок спостереження, 2023 рік
1	0.75	Перехід з 2023 до 2024 року
2	0.21	Перехід з 2024 до 2025 року

З отриманих результатів видно, що після першого міжрічного переходу значення функції становить $\hat{S}(1) = 0.75$, тобто приблизно 75% регіонів залишилися без погіршення. Тоді як, після другого переходу значення знизилося до $\hat{S}(2) = 0.21$, тобто приблизно 21% регіонів не мали погіршення до кінця періоду спостереження. Водночас для 5 регіонів перехід до більш ризикового режиму не був зафіксований до кінця періоду спостереження, тому вони розглядаються як цензуровані спостереження.

Таким чином, оцінювання часу до першого переходу в більш ризиковий режим доповнює аналіз матриць переходів і траєкторій регіонів. Якщо матриці переходів показують напрям і кількість змін між режимами, то оцінка Каплана-Мейєра узагальнює, як швидко регіони втрачають стан без погіршення. Отримане зниження функції після другого міжрічного переходу узгоджується з попередніми результатами та свідчить про посилення динаміки переходів до більш напружених режимів ризику у 2025 році.

3.7 Просторова інтерпретація виявлених режимів ризику

3.7.1 Картографічне представлення просторового розподілу індексів ризику

Із метою аналізу та перевірки того, чи мають виявлені річні режими та допоміжні індекси географічні закономірності, було виконано додаткову просторову інтерпретацію розподілу показників `annual_risk_index`, `exposure_score`, `structural_score` та `cluster_severity_rank` на рівні регіонів окремо для 2023, 2024 і 2025 років (рис. 3.8).

Картографічна візуалізація показала, що інтегральний річний ризик (`annual_risk_index`) у всі три роки мав виражений західно-східний характер. У 2023 році нижчі значення спостерігалися переважно у західних та частині центральних областей, тоді як вищі були зосереджені у східних і південно-східних регіонах. У 2024 році ця поведінка збереглася та стала ще більш виразною. У 2025 році просторовий контраст залишився помітним, проте розподіл став більш неоднорідним, що узгоджується з результатами кластерного аналізу, де саме у цьому році було виявлено найбільшу різноманітність режимів ризику.

Подібна просторова карта спостерігалася і для показника `exposure_score`,

який характеризує інтенсивність режиму. У всі роки для нього також виявлено чітке зростання значень від західних областей до східних. Натомість для structural_score просторові відмінності були менш вираженими і не формували таких чітких груп сусідніх регіонів і не зберігалися стабільно в часі, як у випадку exposure_score.

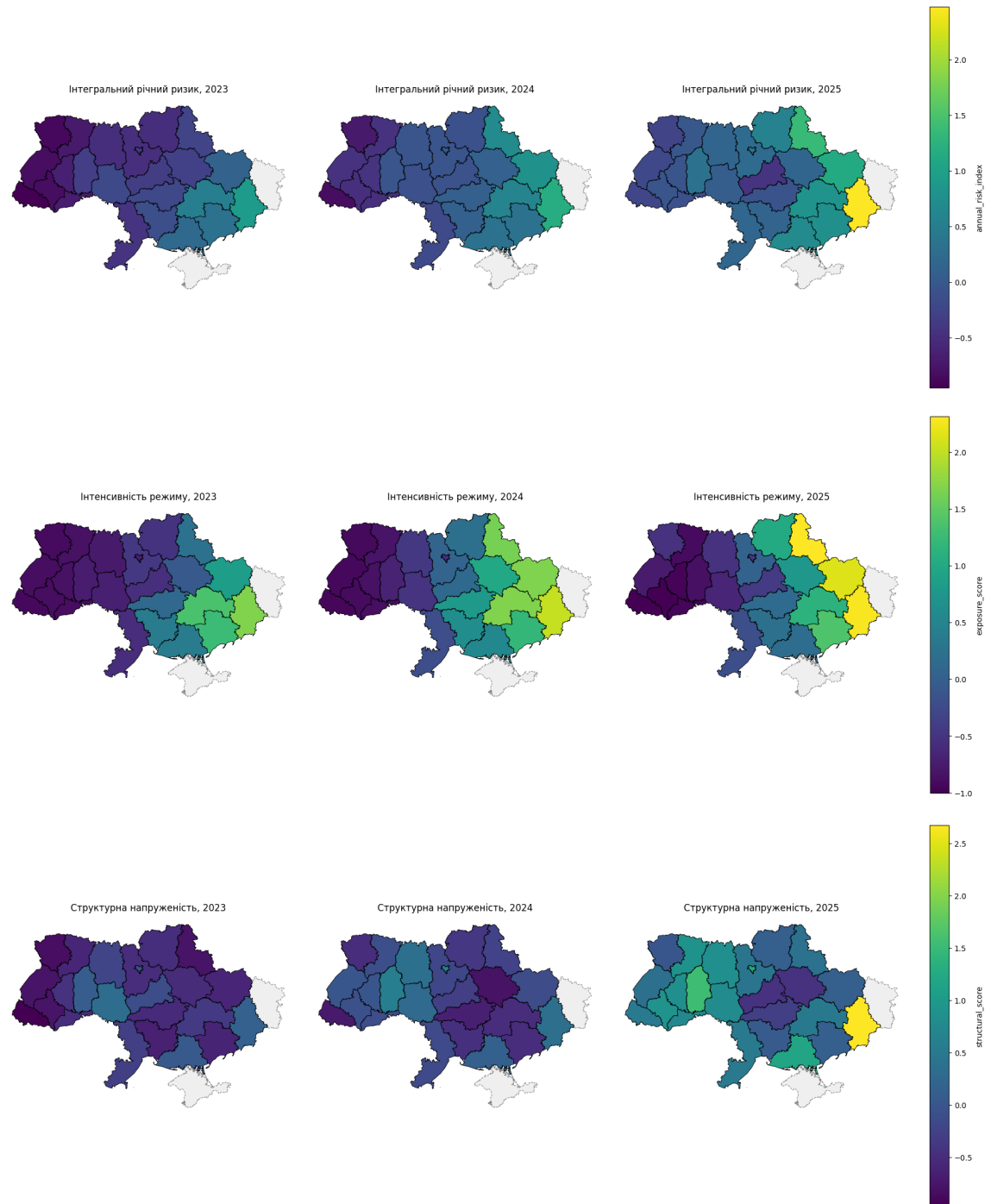


Рисунок 3.8 – Картографічна візуалізація просторового розподілу показників annual risk index, exposure score та structural score

Візуалізація cluster_severity_rank додатково показала чіткі групи

розподілу рівнів у 2023 та 2024 роках, та більш диференційованим полем у 2025 році (рис. 3.9).

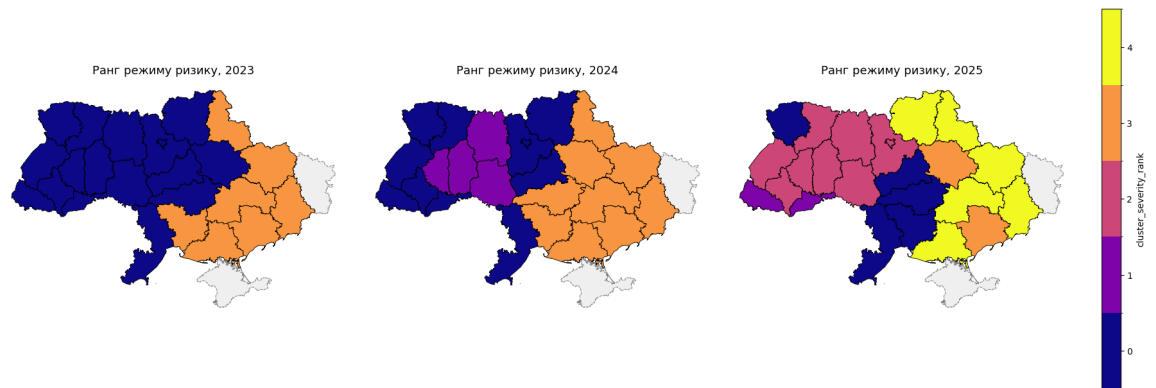


Рисунок 3.9 – Картографічна візуалізація просторового розподілу показника cluster severity rank

3.7.2 Оцінювання просторового градієнта за кореляцією Спірмена

Для оцінювання наявності великомасштабного просторового градієнта було використано коефіцієнт рангової кореляції Спірмена. У межах цієї роботи він застосовується для перевірки монотонного зв'язку між значеннями інтегрального індексу ризику та географічними координатами центрів регіонів (табл. 3.8). Такий підхід дозволяє оцінити, чи пов'язане зростання або зниження ризику із західно-східним або північно-південним положенням регіону.

Координати центрів регіонів і коефіцієнти рангової кореляції було визначено на основі адміністративних меж України першого рівня [33]. Окремо обчислювалася кореляція між значеннями `annual_risk_index` та довготою регіону, а також між `annual_risk_index` та широтою регіону. Розрахунок виконано за допомогою функції `scipy.stats.spearmanr` [27].

Знайдене додатне значення кореляції з довготою означає, що зі зміщенням на схід значення ризику мають тенденцію зростати, тоді як від'ємне значення вказує на вищі значення ризику в західному напрямку.

Додатне значення кореляції з широтою свідчить про тенденцію до зростання ризику в північному напрямку, а від’ємне у південному. Значення, близькі до нуля, вказують на відсутність вираженого монотонного просторового градієнта. У роботі результат вважався статистично значущим, якщо р-значення ($p\text{-value}$) < 0.05 .

Таблиця 3.8 – Коефіцієнти рангової кореляції Спірмена для показника інтегрального ризику за роками

year	value_col	rho_lon	p_lon	rho_lat	p_lat
2023	annual_risk_index	0.897	2.826744e-09	-0.45	0.028
2024	annual_risk_index	0.9367	1.729400e-11	-0.183	0.393
2025	annual_risk_index	0.764	1.373451e-05	-0.07	0.747

Отримані результати показали сильний і статистично значущий позитивний зв’язок із довготою в усі три роки. Найвищі значення кореляції спостерігалися у 2023–2024 роках, тоді як у 2025 році зв’язок залишився статистично значущим, але став дещо слабшим. Це означає, що інтегральний ризик у досліджуваній період загалом зростав у напрямі із заходу на схід.

Водночас, зв’язок із широтою виявився слабшим і нестійким. У 2023 році спостерігався помірний негативний ефект, тоді як у 2024–2025 роках він статистично не підтвердився.

Отже, річний ризик відрізняється передусім між західними та східними регіонами, а не між південними та північними.

3.7.3 Перевірка просторової автокореляції за глобальним індексом Морана

Для перевірки того, чи мають сусідні регіони подібні значення інтегрального індексу ризику було розраховано глобальний індекс Морана

(Global Moran's I) (табл. 3.9). На відміну від кореляції Спірмена, яка оцінює загальний напрямний градієнт, Moran's I враховує просторову суміжність регіонів і дозволяє перевірити наявність просторової автокореляції [8].

Таблиця 3.9 – Global Moran's I для показників у різні роки

year	moran_I	p_sim	EI
annual_risk_index			
2023	0.757	0.001	-0.043
2024	0.685	0.001	-0.043
2025	0.495	0.001	-0.043
exposure_score			
2023	0.813	0.001	-0.043
2024	0.833	0.001	-0.043
2025	0.775	0.001	-0.043
structural_score			
2023	0.202	0.056	-0.043
2024	0.118	0.121	-0.043
2025	0.064	0.199	-0.043

Додатне статистично значуще значення індексу означає, що регіони з подібним рівнем ризику мають тенденцію зосереджуватися поруч. Від'ємне значення вказувало б на просторове чергування високих і низьких значень, а значення, близьке до нуля, на відсутність вираженої просторової структури.

Для розрахунку було використано функцію `esda.moran.Moran` [28]. Матриця просторових ваг задає, які регіони вважаються сусідами під час розрахунку просторової автокореляції. У роботі використано критерій суміжності Queen функцією `libpysal.weights.Queen`, за яким регіони є сусідніми, якщо мають спільну межу або точку [28]. Після побудови матриці сусідства вона була рядково стандартизована. Це означає, що ваги сусідів для кожного регіону нормуються так, щоб їх сума дорівнювала одиниці. Завдяки

цьому регіони з більшою кількістю сусідів не мають непропорційно більшого впливу на розрахунок індексу.

В отриманих результатах I відповідає фактичному значенню глобального індексу Морана, EI – очікуваному значенню індексу за нульової гіпотези випадкового просторового розподілу, p_sim – симуляційному р-значенню, отриманому на основі перестановок значень між регіонами. Якщо фактичне значення I є додатним і статистично значущим, це свідчить про наявність позитивної просторової автокореляції, тобто просторового групування подібних значень досліджуваного показника [28]. У роботі результат вважався статистично значущим, якщо p -значення < 0.05 .

Для `annual_risk_index` в усі роки отримано статистично значущу позитивну просторову автокореляцію (рис. 3.10). Це свідчить про те, що області з близьким рівнем інтегрального ризику мають тенденцію просторово групуватися. Водночас зниження значення Moran's I у 2025 році вказує на послаблення просторової компактності та зростання внутрішньої неоднорідності просторової структури ризику.

Для показника інтенсивності режиму (`exposure_score`) спостерігається сильніша просторова автокореляція, із значеннями Moran's $I > 0.7$ та $p = 0.001$ для всіх років. Водночас, для `structural_score`, навпаки, глобальна просторова автокореляція виявилася слабкою. Це означає, що інтенсивність режиму має найвиразніший просторовий розподіл серед усіх складових системи. Натомість структурна напруженість проявляється більш локально і нерівномірно.

Оскільки в кожному році аналізувалася однакова кількість регіонів, очікуване значення EI залишається однаковим і дорівнює приблизно -0.043.

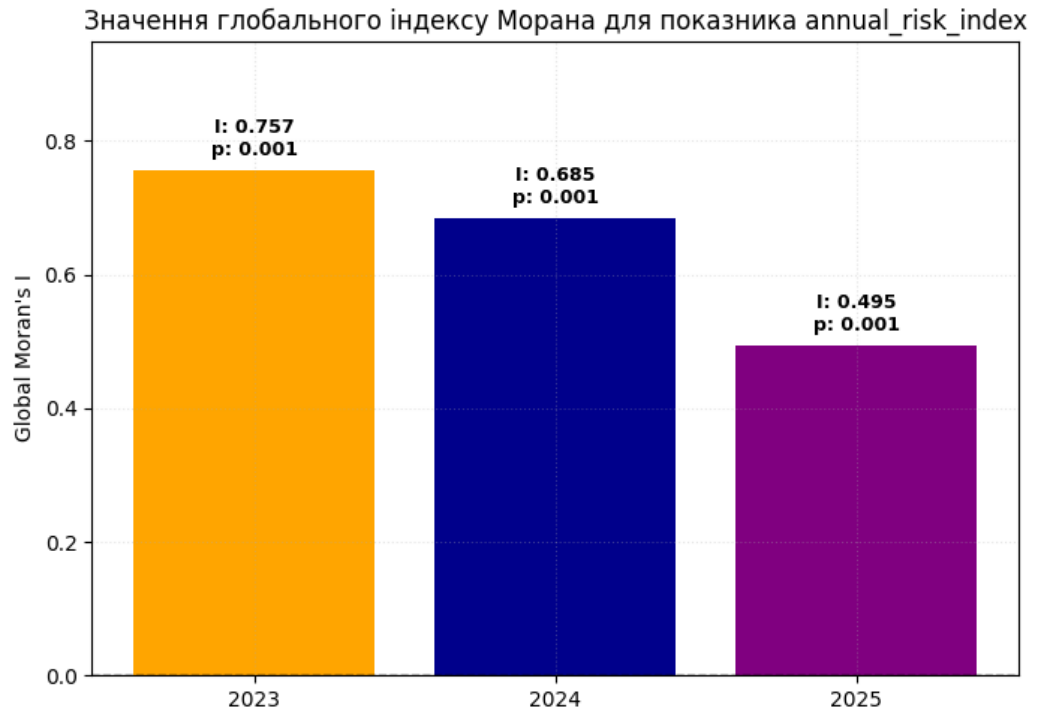


Рисунок 3.10 – Значення глобального індексу Морана для показника annual_risk_index у різні роки

3.8 Обмеження дослідження та напрями подальшого розвитку

Проведене дослідження має кілька обмежень, які необхідно враховувати під час інтерпретації отриманих результатів.

По-перше, обраний часовий період охоплює лише 2023–2025 роки. Цей період дозволяє простежити основні зміни регіональних режимів ризику після першого року повномасштабної війни, однак він залишається відносно коротким для довгострокового аналізу. Через це оцінювання часу до першого переходу регіону в більш ризиковий режим має переважно описовий характер. Оцінка Каплана-Мейєра у цій роботі використовується не як модель прогнозування, а як спосіб узагальнити, яка частка регіонів залишалася без погіршення після кожного міжрічного переходу.

По-друге, наступним обмеженням є неповнота даних для окремих територій. Через тимчасову окупацію та обмежену доступність інформації до

аналізу не було включено Автономну Республіку Крим, місто Севастополь та Луганську область. Таким чином, результати не описують повну картину для всієї території України.

Також важливо враховувати, що повітряні тривоги є лише одним із можливих індикаторів регіонального ризику. Вони добре відображають регулярність і часову структуру загроз, однак не охоплюють усі аспекти воєнного впливу на регіони. Наприклад втрати населення, руйнування та економічні збитки, міграційні процеси, стан енергетичної інфраструктури та близькість до лінії фронту можуть мати додаткове значення для оцінювання ризику. У цій роботі частково враховано інфраструктурний компонент, проте повна оцінка регіональної вразливості потребує ширшого набору даних.

Важливо зазначити, що кластеризація не встановлює причинно-наслідкових зв'язків. Вона дозволяє виявити групи регіон-рік об'єктів зі схожими часовими профілями, але не пояснює безпосередньо, чому певний регіон перейшов до більш або менш ризикового режиму. Тому отримані кластери доцільно розглядати як аналітичні режими тривожного навантаження, а не як остаточну оцінку безпекового стану регіону. Крім того, результати залежать від вибору ознак, способу стандартизації, метрики подібності та кількості кластерів [7].

Напрямом подальшого розвитку дослідження є розширення часового горизонту після появи нових даних та доповнення новими ознаками, такими як економічної активності, демографічних змін, енергетичних відключень та близькості до зон підвищеної небезпеки. Окрім цього, можна також провести ширший аналіз чутливості результатів до вибору ознак, кількості кластерів і параметрів моделей.

Практичним продовженням роботи може стати створення аналітичної системи або інтерактивної панелі моніторингу, яка дозволяла б відстежувати зміну регіональних режимів ризику в часі та виявляти переходи до більш напружених станів.

Висновки до розділу 3

У третьому розділі було представлено результати фінальної кластеризації річних регіональних профілів повітряних тривог та виконано їхню змістовну інтерпретацію. На основі обраної конфігурації SoftDTW-KMeans було виділено п'ять типових режимів ризику, що відрізняються за частотою тривог, середньою тривалістю епізодів, часткою нічних тривог, навантаженням у денні операційні години та інфраструктурним компонентом.

Показано, що структура цих режимів не була сталою впродовж 2023–2025 років. У різні роки регіони переходили між кластерами, а у 2025 році кластерний розподіл виявився більш різноманітним.

Для змістовної інтерпретації отриманих кластерів було запропоновано річний індекс ризику повітряних тривог, який поєднує інтенсивність тривожного навантаження та структурну напруженість режиму. Важливо, що цей індекс не використовувався для формування кластерів, а був розрахований після кластеризації для кожного об'єкта типу регіон-рік. Далі для кожного кластера було визначено медіанне значення річного індексу ризику, і саме за зростанням цієї медіани кластерам присвоєно ранги відносної ризиковості. Це дозволило перейти від формальних номерів кластерів до інтерпретованої шкали режимів, від менш до більш ризикових.

На основі такого впорядкування було побудовано матриці міжрічних переходів, проаналізовано індивідуальні траєкторії регіонів і визначено напрями зміни ризиковості між 2023, 2024 і 2025 роками. При цьому погіршенням режиму вважався перехід регіону до кластера з вищим рангом відносної ризиковості, а не будь-яке локальне збільшення окремого числового показника. Такий підхід дозволив аналізувати саме зміну типового режиму ризику в часі.

Додатково було формалізовано подію першого переходу регіону в більш

ризиковий режим і виконано оцінювання часу до її настання за допомогою оцінки Каплана-Мейєра. Результати показали, що після першого міжрічного переходу 75% регіонів залишалися без погіршення, тоді як після другого переходу ця частка знизилася приблизно до 21%. Це свідчить про посилення динаміки переходів до більш напружених режимів ризику в межах досліджуваного періоду.

Просторова інтерпретація результатів показала, що виявлені режими та інтегральні показники ризику мають виражену географічну закономірність. Картографічний аналіз і кореляція Спірмена виявили стійкий західно-східний градієнт ризику, тоді як оцінювання глобального індексу Морана підтвердило наявність статистично значущої позитивної просторової автокореляції для річного індексу ризику. Це означає, що регіони з подібними рівнями інтегрального ризику мають тенденцію просторово групуватися.

Отримані результати підтверджують доцільність використання багатовимірних річних профілів, методів кластеризації часових рядів і просторового аналізу для дослідження регіональних режимів ризику в умовах війни. Практичне значення роботи полягає в можливості застосування такого підходу для моніторингу змін безпекової ситуації, виявлення переходів до більш напружених режимів і формування аналітичної основи для підтримки рішень у сфері регіонального планування та відновлення.

РОЗДІЛ 4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ

У цьому розділі проводиться функціонально-вартісний аналіз програмного продукту, розробленого для кластеризації річних регіональних профілів повітряних тривог та виявлення типових режимів ризику в Україні у 2023–2025 роках.

Під час проєктування програмного продукту, враховувалися його функціональність, обчислювальна ефективність, зручність використання для дослідницького аналізу, можливість відтворення результатів та економічна доцільність реалізації. Розроблений програмний продукт має забезпечувати обробку вхідних даних, формування багатовимірних часових профілів типу регіон-рік, застосування методів кластеризації часових рядів, оцінювання якості та стабільності отриманого розбиття, а також побудову таблиць, графіків і карт для подальшої інтерпретації результатів.

Для обґрунтування вибору найкращого варіанту реалізації використовується апарат функціонально-вартісного аналізу (ФВА). Це методологія, що дозволяє оцінити співвідношення між корисністю програмного продукту, його технічними характеристиками та витратами на створення. У межах цього розділу розглядаються альтернативні варіанти реалізації окремих функціональних компонентів програмного продукту, після чого обирається варіант із найкращим техніко-економічним рівнем. Функціонально-вартісний аналіз виконується за такими етапами:

- 1) постановка задачі проєктування;
- 2) обґрунтування основних функцій програмного продукту;
- 3) вибір ключових параметрів, здійснення експертного оцінювання їхньої вагомості;
- 4) аналіз рівня якості альтернативних варіантів реалізації;
- 5) економічне оцінювання витрат на реалізацію кожного з можливих

варіантів;

б) вибір оптимального варіанту програмного продукту.

4.1 Постановка задачі проектування

У межах цієї роботи функціонально-вартісний аналіз використовується для техніко-економічного обґрунтування доцільності розробки програмного продукту, призначеного для аналізу річних регіональних профілів повітряних тривог. Програмний продукт реалізує підхід до виявлення типових режимів ризику в Україні у 2023–2025 роках на основі багатовимірних часових профілів.

Основне призначення програмного продукту полягає у підтримці дослідницького аналізу безпекових даних на регіональному рівні. Він має забезпечувати завантаження та попередню обробку даних, розрахунок ознак інтенсивності й структури повітряних тривог, формування річних профілів типу регіон-рік, застосування алгоритмів кластеризації часових рядів, оцінювання якості та стабільності кластерного розбиття, а також візуалізацію отриманих результатів.

Користувачем такого програмного продукту може бути дослідник, аналітик або фахівець, який працює з регіональними безпековими, соціально-економічними чи просторовими даними. Результатами роботи програмного продукту є кластерні мітки для об'єктів типу регіон-рік, характеристики виявлених режимів ризику, показники якості та стабільності кластеризації, таблиці переходів між режимами, а також графіки й карти для просторової інтерпретації результатів.

До програмного продукту висуваються такі основні вимоги:

- 1) можливість роботи на персональному комп'ютері зі стандартним набором програмних засобів;

- 2) підтримка обробки табличних даних і формування багатовимірних часових профілів;
- 3) реалізація кількох алгоритмів кластеризації та можливість порівняння їх результатів;
- 4) забезпечення відтворюваності обчислень за рахунок фіксації параметрів і випадкових станів;
- 5) достатня швидкість обробки даних для проведення серії експериментів;
- 6) можливість подальшого розширення коду, зокрема додавання нових ознак, метрик якості або способів візуалізації;
- 7) зрозумілість структури програмного коду для подальшого використання в дослідницьких або прикладних задачах;
- 8) мінімізація витрат на розробку та подальше використання програмного продукту.

4.2 Обґрунтування функцій програмного продукту

Головна функція F_0 полягає у створенні програмного продукту, який у свою чергу реалізує підхід виявлення річних режимів ризику на основі кластеризації річних регіональних профілів повітряних тривог.

На основі головної функції, можна виділити наступні складові функціональні підсистеми, підфункції, які забезпечують реалізацію ключових задач дослідження:

- 1) F_1 – вибір мови програмування;
- 2) F_2 – вибір середовища розробки;
- 3) F_3 – вибір способу реалізації алгоритмів обробки, кластеризації та аналізу даних.

Кожна з цих функцій має декілька варіантів реалізації:

Функція F_1 :

а) Python;

б) R;

в) C++.

Функція F_2 :

а) Google Colaboratory;

б) Visual Studio Code;

в) Jupyter Notebook.

Функція F_3 :

а) використання готових бібліотек для обробки даних, кластеризації та візуалізації;

б) повна власна реалізація алгоритмів і допоміжних процедур;

в) комбінований підхід: використання готових бібліотек для базових алгоритмів та власних функцій для спеціалізованих етапів аналізу.

Варіанти реалізації основних функцій наведені у морфологічній карті системи (рис. 4.1).

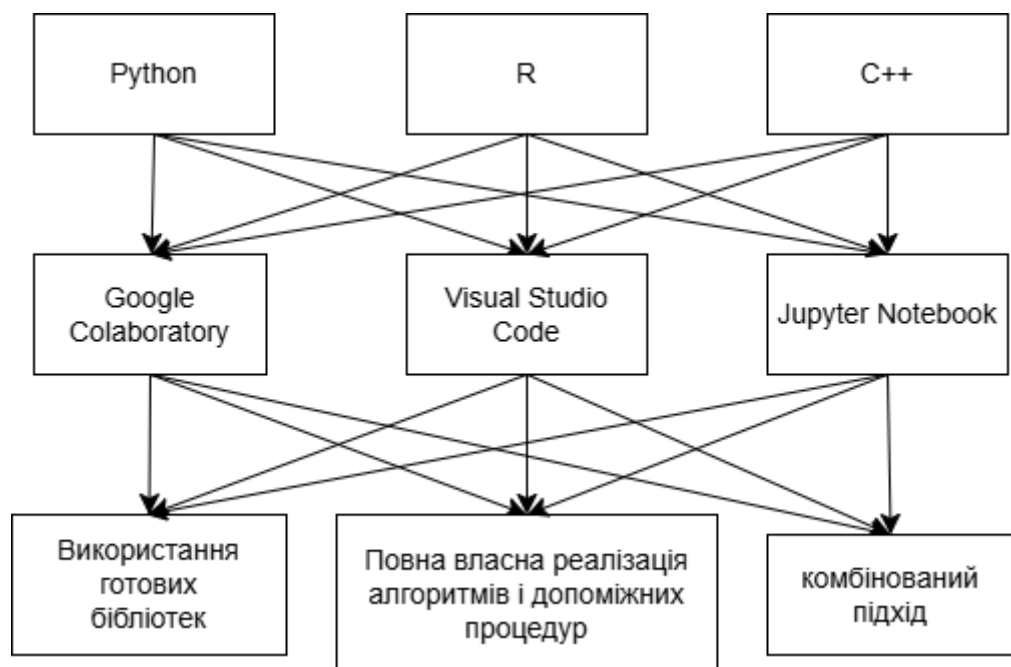


Рисунок 4.1 Морфологічна карта системи

Морфологічна карта відображає множину можливих варіантів реалізації

основних функцій програмного продукту. Для подальшого відбору найбільш доцільних варіантів побудовано позитивно-негативну матрицю, наведену в таблиці 4.1.

Таблиця 4.1 Позитивно-негативна матриця

Функції	Варіанти реалізації	Переваги	Недоліки
F_1	а) Python	Простіша розробка, має широкий набір бібліотек для обробки даних, методів машинного навчання, кластеризації часових рядів, статистичного аналізу та візуалізації.	Потребує встановлення та узгодження версій бібліотек, для окремих обчислень може поступатися компільованим мовам за швидкістю.
	б) R	Має зручні засоби статистичного аналізу, побудови графіків і роботи з табличними даними.	Менш зручний для інтеграції різних етапів аналізу часових рядів, кластеризації, просторової обробки та подальшого розширення програмного продукту.
	в) C++	Забезпечує високу швидкодію та контроль над використанням пам'яті.	Вимагає значно більшого часу на розробку, має складніший синтаксис і меншу кількість готових інструментів для аналізу даних.
F_2	а) Google Colaboratory	Не потребує складного локального налаштування, дозволяє запускати код у браузері, можливість збереження файлів на Google Drive.	Залежить від інтернет-з'єднання, має обмеження за часом сесії, потребує повторного підключення файлів і може бути менш зручним для довгострокового відтворення результатів.
	б) Visual Studio Code	Дозволяє працювати локально, зберігати стабільну структуру проекту, контролювати файли, середовище, результати експериментів, та потребує менших обчислювальних потужностей.	Потребує попереднього налаштування середовища та встановлення необхідних бібліотек.

Продовження таблиці 4.1

Функції	Варіанти реалізації	Переваги	Недоліки
F_2	в) Jupyter Notebook	Зручний для поетапного дослідницького аналізу, перевірки проміжних результатів, побудови графіків і документування експериментів.	Менш зручний для підтримки великої структури проєкту та повторного запуску повного циклу аналізу у вигляді окремого програмного продукту.
F_3	а) Використання готових бібліотек	Пришвидшує розробку, зменшує кількість помилок, дозволяє використовувати перевірені реалізації алгоритмів.	Не всі спеціалізовані етапи дослідження можуть бути повністю реалізовані.
	б) Повна власна реалізація алгоритмів	Дає повний контроль над усіма етапами обчислень і дозволяє адаптувати алгоритми під конкретну задачу.	Є трудомісткою, підвищує ймовірність помилок і потребує значно більшого часу на тестування та перевірку коректності результатів.
	в) Комбінований підхід	Дозволяє поєднати надійність готових бібліотек із гнучкістю власних функцій для спеціалізованих етапів.	Потребує узгодження результатів, отриманих за допомогою різних бібліотек і власних функцій.

На основі аналізу позитивно-негативної матриці можна зробити висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто відкинути, тому що вони не відповідають поставленим вимогам задачі.

Для функції F_1 найкращим варіантом є Python, оскільки він забезпечує достатню швидкість розробки, має широкий набір бібліотек для роботи з табличними даними, часовими рядами, машинним навчанням, кластеризацією та візуалізацією. Варіант із використанням R є можливим, однак менш зручним для комплексної реалізації всіх етапів дослідження. Використання C++ є недоцільним, оскільки для цієї задачі критичною є не максимальна швидкодія, а гнучкість, відтворюваність і швидкість дослідницької розробки.

Для функції F_2 можливими кращими варіантами є Visual Studio Code та Jupyter Notebook. Visual Studio Code більше підходить для оформлення роботи

у вигляді повноцінного програмного продукту, де код поділений на окремі файли і його зручно змінювати або доповнювати. Jupyter Notebook зручніший на етапі дослідження, коли потрібно послідовно запускати окремі частини коду, переглядати таблиці, перевіряти результати кластеризації та будувати візуалізації. Google Colaboratory може бути корисним для швидкого запуску коду, однак через залежність від інтернет-з'єднання та обмеження середовища він є менш надійним як основний варіант реалізації програмного продукту.

Для функції F_3 найбільш доцільним є комбінований підхід. У межах роботи базові операції обробки даних, кластеризації та візуалізації доцільно виконувати за допомогою готових бібліотек, оскільки це зменшує трудомісткість розробки та підвищує надійність результатів. Водночас окремі етапи, пов'язані з формуванням річних профілів, розрахунком стабільності, упорядкуванням режимів за відносною ризиковістю та побудовою спеціалізованих таблиць, потребують власних функцій. Тому повністю бібліотечний підхід є недостатньо гнучким, а повна власна реалізація всіх алгоритмів надмірно трудомісткою.

Таким чином, розглядаються наступні варіанти реалізації програмного продукту:

$$F_1a - F_2b - F_3b$$

$$F_1a - F_2b - F_3b$$

4.3 Обґрунтування системи параметрів програмного продукту

Для оцінювання коефіцієнта технічного рівня обрана система параметрів, яка найбільш точно характеризує ефективність та функціональність програмної реалізації. На основі даних розглянутих вище були обрані наступні параметри:

- X_1 – час попередньої обробки та формування річних профілів;

- X2 – час виконання кластеризації та оцінювання стабільності;
- X3 – обсяг оперативної пам'яті;
- X4 – потенційний обсяг програмного коду.

Гірші, середні та кращі значення параметрів обрано з урахуванням вимог замовника до програмного продукту, умов його використання та характеру поставленої задачі. Значення наведено в таблиці 4.2.

Таблиця 4.2 Основні параметри програмного продукту

Назва параметра	Позначення	Одиниці виміру	Значення параметра		
			гірші	середні	кращі
Час попередньої обробки та формування річних профілів	X1	с	60	30	10
Час виконання кластеризації та оцінювання стабільності	X2	с	3600	1800	600
Обсяг оперативної пам'яті	X3	Мб	2048	1024	512
Потенційний обсяг програмного коду	X4	к-сть рядків	3000	2500	1800

За даними табл. 4.2 будемо графічні характеристики параметрів (рис. 4.2–4.5).

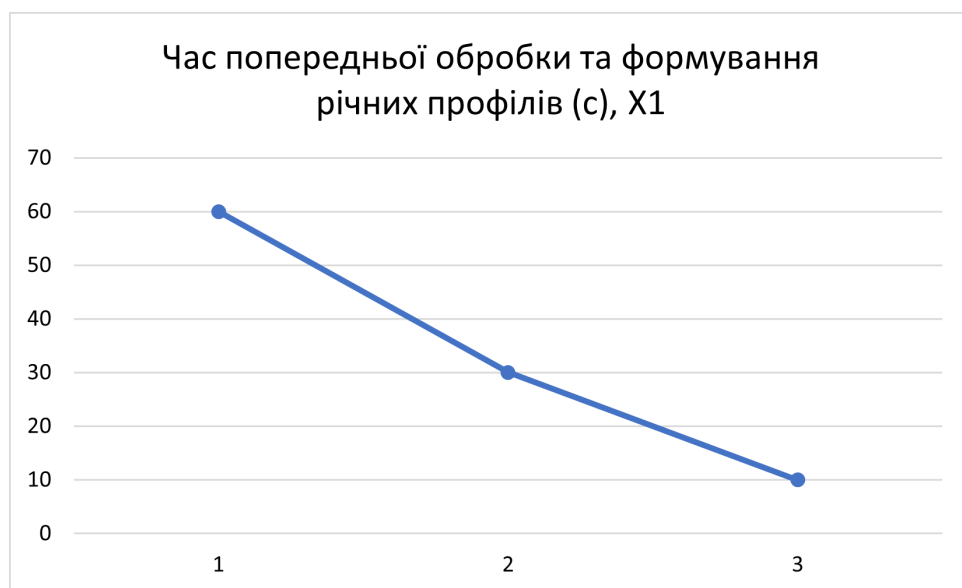


Рисунок 4.2 – Час попередньої обробки та формування річних профілів (с), X1

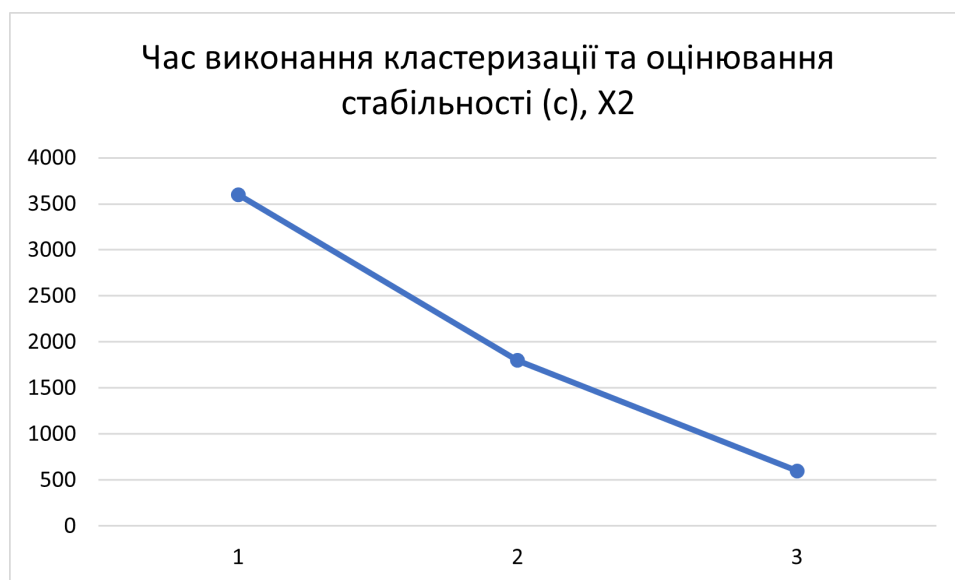


Рисунок 4.3 – Час виконання кластеризації та оцінювання стабільності (с), X2

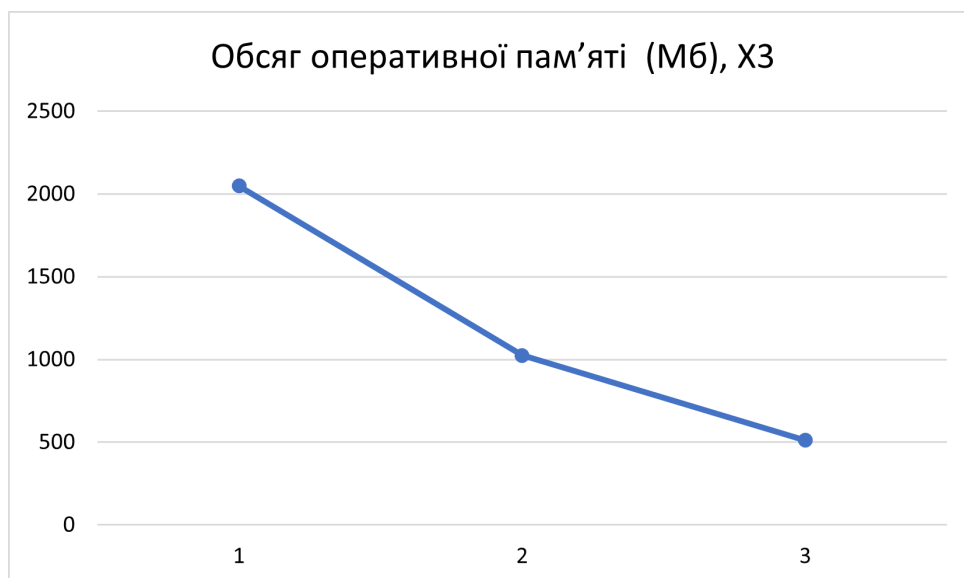


Рисунок 4.4 – Обсяг оперативної пам'яті (Мб), X3

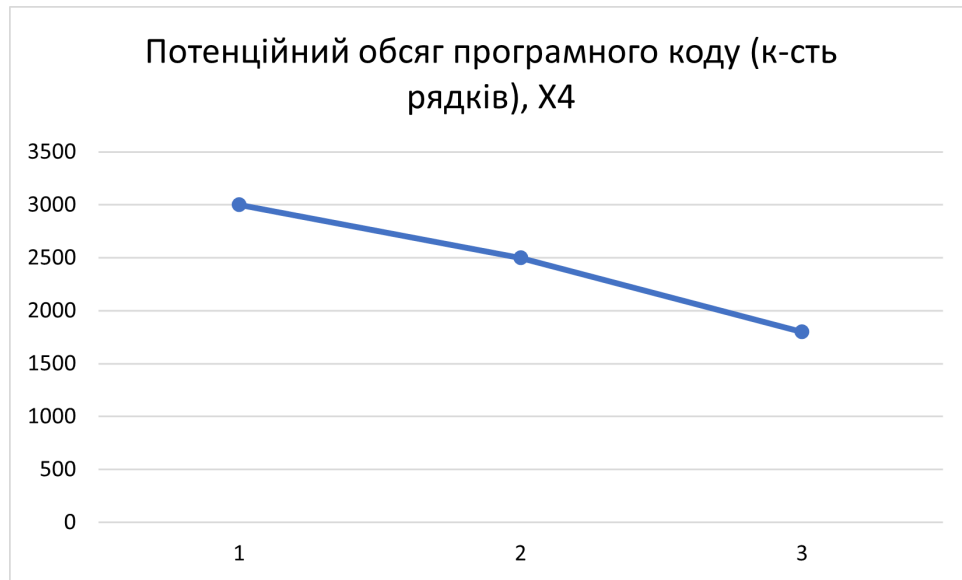


Рисунок 4.5 – Потенційний обсяг програмного коду (к-ість рядків), X4

4.4 Аналіз експертного оцінювання параметрів

Після обговорення й аналізу кожен експерт оцінює ступінь важливості кожного параметру для конкретно поставленої цілі – кластеризація річних регіональних профілів повітряних тривог та виявлення типових режимів ризику в Україні у 2023–2025 роках.

Значимість кожного параметра визначається за допомогою методу попарного порівняння. Оцінювання здійснюється експертною комісією у складі 7 фахівців. Визначення коефіцієнтів значимості має наступний вигляд:

- визначення рівня значимості параметра шляхом присвоєння різних рангів;
- перевірка коректності експертних оцінок для їх подальшого використання;
- визначення оцінки попарного пріоритету параметрів;
- обробка отриманих результатів і розрахунок коефіцієнтів значимості.

Підсумкові результати експертного ранжування представлені в таблиці 4.3.

Таблиця 4.3 – Результати ранжування параметрів

Параметр	Ранг параметра за оцінкою експерта							Сума рангів, R_i	Відхилення, Δ_i	Δ_i^2
	1	2	3	4	5	6	7			
X1	1	1	2	2	2	2	3	13	-4.5	20.25
X2	2	2	1	1	1	1	1	9	-8.5	72.25
X3	3	3	3	3	3	4	2	21	3.5	12.25
X4	4	4	4	4	4	3	4	27	9.5	90.25
Разом	10	10	10	10	10	10	10	70	0	195

Для перевірки степені достовірності експертних оцінок, визначимо наступні параметри:

- 1) сума рангів кожного з параметрів і загальна сума рангів:

$$R_i = \sum_{j=1}^N r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 70,$$

де r_{ij} – ранг i -го параметра, визначеним j -м експертом, N – число експертів, n – кількість параметрів.

- 2) середня сума рангів:

$$T = \frac{1}{n} R_{ij} = 17.5$$

- 3) відхилення суми рангів кожного параметра від середньої суми рангів:

$$\Delta_i = R_i - T$$

Сума відхилень за всіма параметрами повинна дорівнювати 0.

- 4) загальна сума квадратів відхилення:

$$S = \sum_{i=1}^N \Delta_i^2 = 195$$

5) коефіцієнт узгодженості (конкордації):

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 \times 195}{7^2(4^3 - 4)} = 0.796 > W_k = 0.67$$

Коефіцієнт конкордації може лежати в інтервалі $0 \leq W \leq 1$. У разі повної узгодженості поглядів експертів коефіцієнт $W = 1$, тоді як, за більшої розбіжності поглядів експертів, значення W буде меншим. Розрахункове значення W порівнюють з нормативним W_k . Якщо $W > W_k$, то визначені дані придатні до використання.

У нашому випадку знайдений коефіцієнт узгодженості перевищує нормативний, що дорівнює $W_k = 0.67$, тому отримані результати можна вважати надійними. Скориставшись результатами ранжирування, проведемо попарне порівняння всіх параметрів і результати занесемо у таблицю 4.4.

Числове значення, що визначає ступінь переваги параметра X_i над X_j визначається наступним чином:

$$a_{ij} = \begin{cases} 1.5, & X_i > X_j, \\ 1, & X_i = X_j, \\ 0.5, & X_i < X_j \end{cases}$$

З отриманих числових значень коефіцієнтів переваги складемо квадратну матрицю вагомості параметрів $A = \|a_{ij}\|$. Для кожного параметра розрахунок вагомості K_{bi} робимо наступним чином:

$$K_{bi} = \frac{b_i}{\sum_{i=1}^n b_i},$$

$$b_i = \sum_{j=1}^N a_{ij}$$

Таблиця 4.4 – Попарне порівняння параметрів

Пари параметрів	Експерти							Підсумкова оцінка	Числове значення (X_{ij})
	1	2	3	4	5	6	7		
X1 та X2	>	>	<	<	<	<	<	<	0.5
X1 та X3	>	>	>	>	>	>	<	>	1.5
X1 та X4	>	>	>	>	>	>	>	>	1.5
X2 та X3	>	>	>	>	>	>	>	>	1.5
X2 та X4	>	>	>	>	>	>	>	>	1.5
X3 та X4	>	>	>	>	>	<	>	>	1.5

Відносні оцінки розраховуються декілька разів доки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються наступним чином:

$$K_{bi} = \frac{b'_i}{\sum_{i=1}^n b'_i},$$

$$b'_i = \sum_{j=1}^N a_{ij} b_{ij},$$

де b_i – вагомість i -го параметра за результатами оцінок всіх експертів. Визначається як сума значень коефіцієнтів переваги даних усіма експертами по i -му параметру. Розрахунок вагомості параметрів наведено в табл. 4.5.

З таблиці 4.5 видно, що різниця значень коефіцієнтів вагомості на першій і другій ітерації не перевищує 2%, тому проведення більшої кількості ітерацій не потрібно.

Таблиця 4.5 – Розрахунок вагомості параметрів

Параметри X_i	Параметри X_i				Перша ітерація		Друга ітерація	
	X1	X2	X3	X4	b_i	K_{bi}	b'_i	K'_{bi}
X1	1	0.5	1.5	1.5	4.5	0.281	16.25	0.275
X2	1.5	1	1.5	1.5	5.5	0.344	21.25	0.36
X3	0.5	0.5	1	1.5	3.5	0.219	12.25	0.208
X4	0.5	0.5	0.5	1	2.5	0.156	9.25	0.157
Всього					16	1	59	1

4.5 Аналіз рівня якості варіантів реалізації функцій

Визначимо рівень якості кожного з відібраних варіантів реалізації програмного продукту. Оцінювання рівня якості здійснюється за параметрами: X1 (час попередньої обробки та формування річних профілів), X2 (час виконання кластеризації та оцінювання стабільності), X3 (обсяг оперативної пам'яті), X4 (потенційний обсяг програмного коду). Для всіх обраних параметрів менше абсолютне значення відповідає кращому рівню реалізації, оскільки зменшення часу виконання, обсягу використаної пам'яті та обсягу програмного коду покращує ефективність і зручність підтримки програмного продукту. При цьому бальна оцінка параметрів визначається з урахуванням гірших, середніх і кращих значень, наведених у таблиці 4.2.

Коефіцієнт технічного рівня для кожного варіанта реалізації програмного продукту розраховується наступним чином:

$$K_K(j) = \sum_{i=1}^n K_{bi} B_{ij},$$

де n – кількість параметрів, K_{bi} – коефіцієнт вагомості коефіцієнт вагомості i -

го параметра, B_i – оцінка i -го параметра в балах.

Результати розрахунку показників рівня якості наведено в таблиці 4.6.

Таблиця 4.6 – Розрахунок показників рівня якості варіантів реалізації основних функцій програмного продукту

Основні функції	Варіанти реалізації	Параметри, які беруть участь у реалізації функцій	Абсолютне значення параметру	Оцінка параметра в балах, B_i	Коефіцієнт вагомості, K_{bi}	Коефіцієнт якості
F1	а	X1	20 с	8	0.275	2.2
F2	б	X2	900 с	8	0.36	2.88
	б	X3	1024 Мб	7	0.208	1.456
F2	в	X2	1300 с	7	0.36	2.52
	в	X3	1500 Мб	6	0.208	1.248
F3	в	X4	2100 рядків	7	0.157	1.099

За даними таблиці 4.6 визначаємо рівень якості кожного з варіантів:

$$K_K = K_{TY}[F_{1k}] + K_{TY}[F_{2k}] + \dots + K_{TY}[F_{zk}],$$

$$K_{K1} = 2.2 + 2.88 + 1.456 + 1.099 = 7.635$$

$$K_{K2} = 2.2 + 2.52 + 1.248 + 1.099 = 7.067$$

Із розрахунків кращим є перший варіант, для якого коефіцієнт якості має більше значення.

4.6 Економічний аналіз варіантів розробки програмного продукту

Для визначення вартості розробки програмного продукту спочатку проведемо розрахунок трудомісткості для кожного з обраних варіантів

реалізації. Для обох варіантів виділено два окремих завдання:

- 1) розробка проєкту програмного продукту;
- 2) розробка алгоритмічного модуля для обробки даних, кластеризації, оцінювання стабільності та побудови аналітичних результатів.

Завдання 1 за ступенем новизни належить до групи Б, зі ступенем складності 1. Завдання 2 за ступенем новизни належить до групи В, а за ступенем складності – до групи 3.

Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань. Загальна трудомісткість обчислюється як:

$$T_o = T_p \cdot K_{\Pi} \cdot K_{СК} \cdot K_M \cdot K_{СТ} \cdot K_{СТ.М} ,$$

де T_p – трудомісткість розробки програмного продукту; K_{Π} – поправочний коефіцієнт; $K_{СК}$ – коефіцієнт на складність вхідної інформації; K_M – коефіцієнт рівня мови програмування; $K_{СТ}$ – коефіцієнт використання стандартних модулів і прикладних програм; $K_{СТ.М}$ – коефіцієнт стандартного математичного забезпечення.

Для першого завдання, виходячи із норм часу для завдань розрахункового характеру ступеню новизни Б та першої групи складності алгоритму, трудомісткість дорівнює: $T_p = 30$ людино-днів. Поправочний коефіцієнт, який враховує вид нормативно-довідкової інформації для першого завдання: $K_{\Pi} = 1.08$. Поправочний коефіцієнт, який враховує складність контролю вхідної та вихідної інформації для завдань: $K_{СК} = 1$. Оскільки при розробці першого завдання використовуються стандартні модулі, врахуємо це за допомогою коефіцієнта $K_{СТ} = 0.7$. Візьмемо $K_M = 0.8$ та $K_{СТ.М} = 1$. Тоді загальна трудомісткість програмування першого завдання дорівнює:

$$T_1 = 30 \cdot 1.08 \cdot 1 \cdot 0.8 \cdot 0.7 \cdot 1 = 18.14 \text{ людино-днів.}$$

Для другого завдання, із ступенем новизни В та третім рівнем складності: $T_p = 22$ людино-дні, $K_{\Pi} = 1$, $K_{CK} = 1$, $K_M = 0.8$ та $K_{CT.M} = 1$. Для варіанту I: $K_{CT} = 0.7$, а для варіанту II: $K_{CT} = 0.8$. Тоді:

$$T_{2,I} = 22 \cdot 1 \cdot 1 \cdot 0.8 \cdot 0.7 \cdot 1 = 12.32 \text{ людино-днів.}$$

$$T_{2,II} = 22 \cdot 1 \cdot 1 \cdot 0.8 \cdot 0.8 \cdot 1 = 14.08 \text{ людино-днів.}$$

Складаємо трудомісткість відповідних завдань для кожного з обраних варіантів реалізації програми, щоб отримати їх трудомісткість:

$$T_I = (18.14 + 12.32) \cdot 8 = 243.68 \text{ людино-годин.}$$

$$T_{II} = (18.14 + 14.08) \cdot 8 = 257.76 \text{ людино-годин.}$$

У розробці бере участь один програміст з окладом 26000 грн. та один аналітик даних з окладом 28000. Визначимо середню зарплату за годину:

$$C_q = \frac{M}{T_m \cdot t},$$

де M – місячний оклад працівників; T_m – кількість робочих днів тиждень; t – кількість робочих годин в день. Тоді:

$$C_q = \frac{26000 + 28000}{2 \cdot 21 \cdot 8} = 160.71 \text{ грн/год.}$$

Далі розрахуємо заробітну плату:

$$C_{зп} = C_q \cdot T_i \cdot K_d,$$

де C_q – величина погодинної оплати праці програміста; T_i – трудомісткість відповідного завдання; K_d – норматив, який враховує додаткову заробітну

плату;

Зарплата розробників за варіантами становить:

$$C_{зп, I} = 160.71 \cdot 243.68 \cdot 1.2 = 46995.43 \text{ грн.}$$

$$C_{зп, II} = 160.71 \cdot 257.76 \cdot 1.2 = 49710.86 \text{ грн.}$$

Відрахування на єдиний соціальний внесок становить 22%:

$$C_{від, I} = 46995.43 \cdot 0.22 = 10338.99 \text{ грн.}$$

$$C_{від, II} = 49710.86 \cdot 0.22 = 10936.39 \text{ грн.}$$

Тепер визначимо витрати на оплату однієї машино-години (C_M). Так як одна ЕОМ обслуговує одного програміста з окладом 26000 грн., з коефіцієнтом зайнятості 0.2, то для однієї машини отримаємо:

$$C_r = 12 \cdot M \cdot K_3 = 12 \cdot \frac{26000 + 28000}{2} \cdot 0.2 = 64800 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{зп} = C_r(1 + K_3) = 64800 \cdot (1 + 0.2) = 77760 \text{ грн.}$$

Відрахування на соціальний внесок:

$$C_{від} = C_{зп} \cdot 0.22 = 77760 \cdot 0.22 = 17107.2 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 10000 грн.

$$C_A = K_{тм} \cdot K_A \cdot C_{пр} = 1.4 \cdot 0.25 \cdot 10000 = 3500 \text{ грн.,}$$

де $K_{\text{тм}}$ – коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача; K_A – річна норма амортизації; $C_{\text{пр}}$ – договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$$C_P = K_{\text{тм}} \cdot K_P \cdot C_{\text{пр}} = 1.4 \cdot 0.08 \cdot 10000 = 1120 \text{ грн.},$$

де K_P – відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$T_{\text{ЕФ}} = (D_K - D_B - D_C - D_P) \cdot t_z \cdot K_B = (365 - 104 - 11 - 16) \cdot 8 \cdot 0.8 = 1497.6 \text{ години},$$

де D_K – календарна кількість днів у році; D_B, D_C – відповідно кількість вихідних та святкових днів; D_P – кількість днів планових ремонтів устаткування; t – кількість робочих годин у день; K_B – коефіцієнт використання приладу у часі протягом зміни.

Витрати на оплату електроенергії розраховуємо наступним чином:

$$C_{\text{ЕЛ}} = T_{\text{ЕФ}} \cdot N_C \cdot K_z \cdot C_{\text{ЕН}} = 1497.6 \cdot 0.2 \cdot 0.7 \cdot 13.64 = 2859.82 \text{ грн.},$$

де N_C – середньо-споживча потужність приладу; K_z – коефіцієнтом зайнятості приладу; $C_{\text{ЕН}}$ – тариф за 1 КВт-годин електроенергії.

Розраховуємо накладні витрати:

$$C_H = C_{\text{пр}} \cdot 0.67 = 10000 \cdot 0.67 = 6700 \text{ грн.},$$

Тоді, річні експлуатаційні витрати будуть:

$$C_{\text{ЕКС}} = C_{\text{ЗП}} + C_{\text{ВІД}} + C_{\text{А}} + C_{\text{Р}} + C_{\text{ЕЛ}} + C_{\text{Н}} = 77760 + 17107.2 + 3500 + 1120 + 2859.82 + 6700 = 109047.02 \text{ грн.},$$

Собівартість однієї машино-години ЕОМ:

$$C_{\text{МГ}} = \frac{C_{\text{ЕКС}}}{T_{\text{ЕФ}}} = \frac{109047.02}{1497.6} = 72.81 \text{ грн.},$$

Оскільки в даному випадку всі роботи, які пов'язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$\begin{aligned} C_{\text{М}} &= C_{\text{МГ}} \cdot T, \\ C_{\text{М, I}} &= 72.81 \cdot 243.68 = 17743.44 \text{ грн.}, \\ C_{\text{М, II}} &= 72.81 \cdot 257.76 = 18768.67 \text{ грн.}, \end{aligned}$$

Накладні витрати складають 67% від заробітної плати:

$$\begin{aligned} C_{\text{Н}} &= C_{\text{ЗП}} \cdot 0.67, \\ C_{\text{Н, I}} &= 46995.43 \cdot 0.67 = 31486.94 \text{ грн.}, \\ C_{\text{Н, II}} &= 49710.86 \cdot 0.67 = 33306.27 \text{ грн.}, \end{aligned}$$

Отже, вартість розробки ПП за варіантами становить:

$$\begin{aligned} C_{\text{ПП}} &= C_{\text{ЗП}} + C_{\text{ВІД}} + C_{\text{М}} + C_{\text{Н}}, \\ C_{\text{ПП, I}} &= 46995.43 + 10338.99 + 17743.44 + 31486.94 = \\ &106564.80 \text{ грн.}, \\ C_{\text{ПП, II}} &= 49710.86 + 10936.39 + 18768.67 + 33306.27 = \\ &112722.19 \text{ грн.} \end{aligned}$$

4.7 Вибір кращого варіанту ІІІ техніко-економічного рівня

Розрахуємо коефіцієнт техніко-економічного рівня:

$$K_{\text{ТЕР}j} = \frac{K_{kj}}{C_{\Phi j}},$$

$$K_{\text{ТЕР}1} = \frac{7.635}{106564.80} = 7.165 \cdot 10^{-5},$$

$$K_{\text{ТЕР}2} = \frac{7.067}{112722.19} = 6.269 \cdot 10^{-5}$$

Таким чином, за результатами порівняння альтернатив, що залишилися після попереднього відбору, оптимальним визначено перший варіант реалізації програмного продукту. Для нього отримано вище значення показника техніко-економічного рівня якості: $K_{\text{ТЕР}1} = 7.165 \cdot 10^{-5}$.

Цей варіант реалізації програмного продукту має такі параметри:

- вибір мови програмування Python;
- використання середовища Visual Studio Code;
- вибір комбінованого підходу для реалізації алгоритмів обробки.

Такий варіант є доцільним для поставленої задачі, оскільки забезпечує зручну реалізацію програмного продукту, достатню функціональність для обробки та аналізу даних, а також краще співвідношення між якістю реалізації та витратами на розробку.

Висновки до розділу 4

У розділі було проведено функціонально-вартісний аналіз програмного продукту, призначеного для кластеризації річних регіональних профілів повітряних тривог та аналізу режимів ризику. У межах аналізу визначено основні функції програмного продукту, розглянуто можливі варіанти їх реалізації та обґрунтовано систему параметрів, що характеризують якість розробки.

На основі експертного оцінювання було розраховано коефіцієнти вагомості параметрів, оцінено рівень якості альтернативних варіантів реалізації та проведено економічний розрахунок витрат на розробку. Отримані результати дали змогу порівняти варіанти не лише за функціональними характеристиками, а й за їх техніко-економічною доцільністю.

За результатами функціонально-вартісного аналізу доцільним обрано перший варіант реалізації програмного продукту: мова програмування – Python, середовище розробки – Visual Studio Code, спосіб реалізації алгоритмів обробки даних – комбінований підхід. Саме цей варіант має вищий коефіцієнт якості та кращий техніко-економічний рівень, тому обраний для реалізації поставленої задачі.

ВИСНОВКИ

У рамках цієї дипломної роботи було досліджено задачу виявлення річних режимів регіонального ризику в Україні на основі динаміки повітряних тривог у 2023–2025 роках.

Актуальність роботи зумовлена тим, що в умовах повномасштабної війни регіони України перебувають у різних режимах безпекового навантаження, які не можна повністю описати лише загальною кількістю або сумарною тривалістю повітряних тривог. Для змістовного аналізу важливо враховувати не тільки рівень тривожного навантаження, а і його часову структуру, середню тривалість епізодів, добовий розподіл і зв'язок із подіями інфраструктурного ураження.

На відміну від підходу, за якого регіони порівнюються лише за окремими агрегованими показниками, у цій роботі кожен пар регіон-рік подано як окремий стан регіональної системи ризику. Такий стан описано багатовимірним річним часовим профілем, що складається з 12 місячних спостережень і набору взаємопов'язаних ознак. Це дозволило перейти від статичного порівняння регіонів до аналізу типових режимів ризику та їхньої зміни в часі.

У теоретичній частині роботи було обґрунтовано доцільність використання багатовимірних часових рядів для опису регіональних профілів повітряних тривог. Розглянуто основні підходи до кластеризації часових профілів, метрики подібності часових рядів, зокрема евклідову відстань, метод динамічного викривлення часу (DTW) та його згладжену версію (Soft-DTW), а також критерії оцінювання якості й стабільності кластеризації. Окремо розглянуто підходи до просторової інтерпретації результатів та аналізу часу до події, що надалі використано для оцінювання переходів регіонів між режимами ризику.

На основі вхідних даних за 2023–2025 роки сформовано систему ознак,

яка описує інтенсивність і структуру повітряних тривог. До фінального набору ознак для кластеризації включено частоту тривог, середню тривалість одного епізоду, частку нічних тривог, частку тривог у денні операційні години та інфраструктурний компонент. У підсумку кожен об'єкт типу регіон-рік було подано як річний профіль розмірності 12×5 , а загальний масив для кластеризації мав розмірність $72 \times 12 \times 5$. Такий підхід дозволив описати регіональний ризик не одним узагальненим числом, а через кілька взаємопов'язаних характеристик, що разом відображають режим функціонування регіону в умовах воєнної загрози.

Для виявлення типових режимів ризику було порівняно кілька підходів: Euclidean K-Means, DTW-KMeans, SoftDTW-KMeans та агломеративну кластеризацію для векторизованих профілів. Оскільки в задачі відсутня еталонна розмітка режимів ризику, основний акцент для вибору фінальної конфігурації зроблено не лише на компактності кластерів, а на стійкості отриманого розбиття. Для цього використано адаптований критерій стабільності Stadion, який дозволив оцінити, наскільки кластерна структура зберігається за малих допустимих збурень даних. На відміну від простішого вибору за одним внутрішнім показником якості, такий підхід дав змогу обрати конфігурацію, яка є більш надійною для інтерпретації та менш залежною від випадкових локальних варіацій у даних.

За результатами порівняння фінальною було обрано конфігурацію SoftDTW-KMeans з п'ятьма кластерами, параметром $\gamma = 270.34$ та середнім значенням критерію Stadion 0.078717. Саме цей метод виявився найбільш узгодженим із поставленою задачею, оскільки дозволяє враховувати локальні відмінності у формі річних профілів і водночас зберегти можливість змістовно інтерпретувати отримані групи.

У результаті фінальної кластеризації було виділено п'ять типових режимів ризику, які відрізняються за частотою тривог, середньою тривалістю епізодів, добовою структурою та інфраструктурним навантаженням. Найбільшим виявився кластер 1, до якого належать 33 об'єкти, тоді як

кластери 2 і 4 містять лише по 6 об'єктів і відповідають більш специфічним режимам. Отриманий розподіл показав, що регіональна ситуація не була сталою впродовж 2023–2025 років: у 2023 році спостерігалися лише два режими, у 2024 році додався ще один, а у 2025 році в даних були представлені всі п'ять режимів. Це свідчить про зростання різноманітності регіональних станів ризику у 2025 році.

Для того щоб інтерпретувати отримані кластери, було запропоновано річний індекс ризику повітряних тривог. Він поєднує дві складові: інтенсивність тривожного навантаження та структурну напруженість режиму. Інтенсивність відображає частоту тривог і загальне часове навантаження, а структурна складова враховує нічний та інфраструктурний компоненти. Важливо, що цей індекс не використовувався для формування кластерів, а був розрахований після кластеризації для кожного об'єкта типу регіон-рік.

Далі для кожного кластера було визначено медіанне значення річного індексу ризику. Саме за зростанням цього медіанного значення кластерам було присвоєно ранги відносної ризиковості від 0 до 4. Такий підхід дозволив перейти від формальних номерів кластерів, які самі по собі не мають порядку, до змістовної шкали режимів, від менш до більш ризикових. Таким чином, у роботі встановлено відповідність між отриманими кластерами та рівнями відносної ризиковості, що зробило результати кластеризації придатними для подальшого аналізу динаміки регіонів.

На основі впорядкованих режимів було проаналізовано міжрічні переходи регіонів у 2023–2025 роках. Було побудовано матриці переходів, розглянуто індивідуальні траєкторії регіонів і визначено напрями зміни ризиковості. При цьому погіршенням режиму вважався перехід регіону до кластера з вищим рангом відносної ризиковості порівняно з попереднім роком.

Додатково було формалізовано подію першого переходу регіону до більш ризикового режиму та виконано оцінювання часу до її настання за допомогою оцінки Каплана-Мейєра. Отримані результати показали, що після першого міжрічного переходу 75% регіонів залишалися без погіршення, тоді

як після другого переходу ця частка знизилася приблизно до 21%. Це свідчить про посилення переходів до більш напружених режимів ризику у 2025 році.

Просторова інтерпретація результатів показала, що виявлені режими та інтегральний річний ризик мають виражену територіальну закономірність. Картографічний аналіз і кореляція Спірмена показали сильний і статистично значущий зв'язок інтегрального ризику з довготою в усі три роки, що вказує на виражений напрям зростання ризику із заходу на схід. Водночас зв'язок із широтою був менш стійким і не підтверджувався статистично для всіх років. Оцінювання глобального індексу Морана додатково підтвердило наявність позитивної просторової автокореляції для річного індексу ризику, тобто регіони з подібними рівнями ризику мають тенденцію просторово групуватися.

У межах роботи було реалізовано програмний продукт мовою Python, який охоплює всі основні етапи дослідження: підготовку та агрегацію даних, формування річних багатовимірних профілів, стандартизацію ознак, порівняння альтернативних алгоритмів кластеризації, оцінювання стабільності, побудову фінальної кластерної моделі, розрахунок річного індексу ризику, післякластерне впорядкування режимів, аналіз переходів, оцінку Каплана-Мейєра та просторову візуалізацію результатів. Це забезпечує відтворюваність дослідження та можливість подальшого аналізу після появи нових даних.

Науково-практична цінність роботи полягає в тому, що запропонований підхід поєднує кластеризацію багатовимірних річних профілів, вибір стійкої конфігурації за адаптованим критерієм Stadion та післякластерне впорядкування режимів за річним індексом ризику. На відміну від простого ранжування регіонів за кількістю або тривалістю тривог, такий підхід дозволяє виділяти типові режими ризику, встановлювати їхню відповідність рівням відносної ризиковості, аналізувати переходи регіонів між режимами та перевіряти просторову структуру отриманих результатів. Це може бути корисним для моніторингу безпекової ситуації та підтримки подальших

аналітичних досліджень у сфері регіональної стійкості, відновлення та планування.

Подальший розвиток роботи може бути пов'язаний із розширенням часового періоду, додаванням нових ознак, пов'язаних із типами загроз, руйнуваннями інфраструктури, економічною активністю та демографічними змінами. За наявності довшого часового ряду також можна глибше дослідити переходи між режимами, побудувати моделі факторів погіршення ризику та перевірити стійкість отриманих кластерів на нових даних.

Отже, у роботі було досягнуто поставленої мети: розроблено та реалізовано підхід до кластеризації річних регіональних профілів повітряних тривог для виявлення режимів ризику в Україні. Запропонована методика поєднує аналіз багатовимірних часових рядів, вибір стійкої конфігурації кластеризації на основі адаптованого критерію Stadion, післякластерне впорядкування режимів за річним індексом ризику, аналіз міжрічних переходів і просторову перевірку результатів. Це дозволило отримати не лише формальне розбиття регіонів на групи, а більш змістовне уявлення про те, як змінювалися регіональні режими ризику в Україні в умовах війни в 2023–2025 роках.

Отримані результати роботи були опубліковані в статті [1] та апробовані на Всеукраїнській конференції [2] та міжнародних конференціях [35; 36].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Кабанова М. І., Мельничук В. Е., Семенченко Н. І. Воєнні ризики та регіональна економічна стійкість України: повітряні тривоги, атаки на інфраструктуру та релокація бізнесу. Сталий розвиток економіки. 2025. № 6(57). С. 155–163. DOI: 10.32782/2308-1988/2025-57-21.
2. Кабанова М. І., Кузнєцова Н. В. Кластеризація регіонів України за динамікою повітряних тривог методом k-середніх для часових рядів (DTW/Soft-DTW) у контексті релокацій бізнесу. Системні науки та інформатика : збірка доповідей IV науково-практичної конференції, Київ, 1–5 грудня 2025 р. Київ : НН ІПСА КПІ ім. Ігоря Сікорського, 2025. С. 109–116.
3. ACAPS. Impact of conflict on the Ukrainian economy: implications for humanitarian needs. 2023. URL: https://www.acaps.org/fileadmin/Data_Product/Main_media/20230213_acaps_the_matic_report_of_the_impact_of_conflict_on_the_ukrainian_economy.pdf
4. Cedos. A year of full-scale war in Ukraine: thoughts, feelings, actions. 2023. URL: <https://cedos.org.ua/wp-content/uploads/layout.-one-year-eng.pdf>
5. World Bank, Government of Ukraine, European Commission, United Nations. Ukraine Fifth Rapid Damage and Needs Assessment (RDNA5): February 2022 – December 2025. URL: <https://documents.worldbank.org/en/publication/documents-reports/documentdetail/099022026094036395>
6. Melnykovska I., Wilson Sokhey S. The local and regional dimension of Ukraine's resilience during Russia's full-scale invasion: an introduction. Post-Soviet Affairs. 2025. Vol. 41, No. 5. P. 401–410. DOI: 10.1080/1060586X.2025.2545626.
7. Aghabozorgi S., Shirkhorshidi A. S., Wah T. Y. Time-series clustering: a decade review. Information Systems. 2015. Vol. 53. P. 16–38. DOI: 10.1016/j.is.2015.04.007.
8. Moran P. A. P. Notes on continuous stochastic phenomena. Biometrika.

1950. Vol. 37, No. 1–2. P. 17–23. DOI: 10.1093/biomet/37.1-2.17.

9. Alerts.in.ua. Alerts.in.ua API documentation. URL: <https://devs.alerts.in.ua/>

10. Paparrizos J., Gravano L. K-Shape: Efficient and Accurate Clustering of Time Series. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. 2015. P. 1855–1870. DOI: 10.1145/2723372.2737793.

11. Liao T. W. Clustering of time series data: a survey. Pattern Recognition. 2005. Vol. 38, No. 11. P. 1857–1874. DOI: 10.1016/j.patcog.2005.01.025.

12. Sakoe H., Chiba S. Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1978. Vol. 26, No. 1. P. 43–49. DOI: 10.1109/TASSP.1978.1163055.

13. Cuturi M., Blondel M. Soft-DTW: a differentiable loss function for time-series. Proceedings of the 34th International Conference on Machine Learning. PMLR, 2017. Vol. 70. P. 894–903.

14. Rousseeuw P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics. 1987. Vol. 20. P. 53–65. DOI: 10.1016/0377-0427(87)90125-7.

15. Forest F., Mourer A., Lebbah M., Azzag H., Lacaille J. An invariance-guided stability criterion for time series clustering validation. 2020 25th International Conference on Pattern Recognition (ICPR). 2021. P. 9296–9303. DOI: 10.1109/ICPR48806.2021.9412020.

16. Spearman C. The proof and measurement of association between two things. The American Journal of Psychology. 1904. Vol. 15, No. 1. P. 72–101. DOI: 10.2307/1412159.

17. Hosmer D. W., Lemeshow S., May S. Applied Survival Analysis: Regression Modeling of Time-to-Event Data. 2nd ed. Hoboken: Wiley-Interscience, 2008. 416 p.

18. Kaplan E. L., Meier P. Nonparametric estimation from incomplete observations. Journal of the American Statistical Association. 1958. Vol. 53, No. 282. P. 457–481. DOI: 10.1080/01621459.1958.10501452.

19. Mantel N. Evaluation of survival data and two new rank order statistics arising in its consideration. *Cancer Chemotherapy Reports*. 1966. Vol. 50, No. 3. P. 163–170.
20. Cox D. R. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*. 1972. Vol. 34, No. 2. P. 187–202.
21. Pedregosa F. et al. Scikit-learn: machine learning in Python. *Journal of Machine Learning Research*. 2011. Vol. 12. P. 2825–2830. DOI: 10.1111/j.2517-6161.1972.tb00899.x.
22. Harris C. R. et al. Array programming with NumPy. *Nature*. 2020. Vol. 585. P. 357–362. DOI: 10.1038/s41586-020-2649-2.
23. Tavenard R., Faouzi J., Vandewiele G., Divo F., Androz G., Holtz C. et al. Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*. 2020. Vol. 21, No. 118. P. 1–6.
24. The pandas development team. pandas-dev/pandas: Pandas. Zenodo, 2023. DOI: 10.5281/zenodo.3509134.
25. Jordahl K. et al. GeoPandas: Python tools for geographic data. Zenodo, 2020. DOI: 10.5281/zenodo.3946761.
26. Hunter J. D. Matplotlib: a 2D graphics environment. *Computing in Science & Engineering*. 2007. Vol. 9, No. 3. P. 90–95. DOI: 10.1109/MCSE.2007.55.
27. Virtanen P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*. 2020. Vol. 17. P. 261–272. DOI: 10.1038/s41592-019-0686-2.
28. Rey S. J., Anselin L. PySAL: a Python library of spatial analytical methods. *The Review of Regional Studies*. 2007. Vol. 37, No. 1. P. 5–27.
29. Davidson-Pilon C. lifelines: survival analysis in Python. *Journal of Open Source Software*. 2019. Vol. 4, No. 40. Article 1317. DOI: 10.21105/joss.01317.
30. ACLED. ACLED's methodology on Ukraine civilian infrastructure automated tags. URL: <https://acleddata.com/methodology/acleds-methodology-ukraine-civilian-infrastructure-automated-tags>

31. Quah D. Empirical cross-section dynamics in economic growth. *European Economic Review*. 1993. Vol. 37, No. 2–3. P. 426–434. DOI: 10.1016/0014-2921(93)90031-5.
32. Chen Y.-C. Lecture 6: Inference on Markov Chain. STAT 516. University of Washington, 2018. URL: https://faculty.washington.edu/yenchic/18A_stat516/Lec6_MC_Inf.pdf
33. Ukraine - Subnational Administrative Boundaries (COD-AB). Humanitarian Data Exchange. URL: <https://data.humdata.org/dataset/cod-ab-ukr>.
34. Everitt B. S., Landau S., Leese M., Stahl D. *Cluster Analysis*. 5th ed. Chichester : Wiley, 2011. 330 p. DOI: 10.1002/9780470977811.
35. Kabanova M., Melnychuk V. Air-raid alert dynamics in Ukraine: a data-driven typology for wartime urban management. Моделювання та прогнозування економічних процесів : збірник тез доповідей XIX Міжнар. наук.-практ. конф., м. Київ, 4 грудня 2025 р. Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2025. С. 256–258.
36. Kabanova M., Kuznietsova N., Seebauer M. Regional Air-Raid Alert Risk Regimes in Ukraine: Management Implications under Uncertainty. *Economic Transformations in the Global World : Proceedings of the International Scientific Conference, Varna Free University “Chernorizets Hrabar”, Varna, Bulgaria, 12–13 June 2026*. 10 p.

НАУКОВІ ДОСЯГНЕННЯ

1. Кабанова М. І., Мельничук В. Е., Семенченко Н. І. Воєнні ризики та регіональна економічна стійкість України: повітряні тривоги, атаки на інфраструктуру та релокація бізнесу. Сталий розвиток економіки. 2025. № 6(57). С. 155–163. DOI: 10.32782/2308-1988/2025-57-21.

2. Кабанова М. І., Кузнєцова Н. В. Кластеризація регіонів України за динамікою повітряних тривог методом k-середніх для часових рядів (DTW/Soft-DTW) у контексті релокацій бізнесу. Системні науки та інформатика : збірка доповідей IV науково-практичної конференції, Київ, 1–5 грудня 2025 р. Київ : НН ІПСА КПІ ім. Ігоря Сікорського, 2025. С. 109–116.

3. Kabanova M., Melnychuk V. Air-raid alert dynamics in Ukraine: a data-driven typology for wartime urban management. Моделювання та прогнозування економічних процесів : збірник тез доповідей XIX Міжнар. наук.-практ. конф., м. Київ, 4 грудня 2025 р. Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2025. С. 256–258.

4. Kabanova M., Kuznietsova N., Seebauer M. Regional Air-Raid Alert Risk Regimes in Ukraine: Management Implications under Uncertainty. Economic Transformations in the Global World : Proceedings of the International Scientific Conference, Varna Free University “Chernorizets Hrabar”, Varna, Bulgaria, 12–13 June 2026. 10 p

ДОДАТОК А. ЛІСТИНГ ПРОГРАМНОГО ПРОДУКТУ

```

import numpy as np
import pandas as pd
import geopandas as gpd
import matplotlib as mpl
from matplotlib import cm
import matplotlib.pyplot as plt
import seaborn as sns
from pathlib import Path
import math
import time
from itertools import product

import tslearn
from tslearn.clustering import TimeSeriesKMeans
from tslearn.metrics import gamma_soft_dtw
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans, AgglomerativeClustering

from scipy.stats import spearmanr
from libpysal.weights import Queen
from esda.moran import Moran
from lifelines import KaplanMeierFitter

path = Path("data/alerts_regions_data.xlsx")

data = pd.DataFrame(pd.read_excel(path))
data.dropna(inplace=True)
data.isnull().sum()

for column in ['night_alerts_count_month (22:00-06:00)',
               'alerts_shool_count (08:00-15:00)',
               'alerts_malls_count (10:00-21:00)']:
    err = data[column] > data['alerts_count_month']
    err_idx = data.index[err]

    for idx in err_idx:
        print(f'ERROR! in index {idx} (column: {column}):')
        print('-' * 40)
    if err.all() == False:

```

```

print(f'There is no errors for: {column}')

# DATA PREPROCESSING
data = data.rename(columns={
    'alerts_shool_count (08:00-15:00)': 'alerts_school_count (08:00-15:00)'
})
idx = data.index[data['region_eng'].isin(['Autonomous Republic of Crimea', 'Luhansk'])]
data = data.drop(index = idx).drop(columns=['region_ukr', 'alert_duration_month'])

value_start = data["period"].str.split('-').str[0].str.strip()
value_end = data["period"].str.split('-').str[1].str.strip()
data.insert(loc=2,
            column = 'period_start',
            value = value_start)

data.insert(loc=3,
            column = 'period_end',
            value = value_end)

data["period_start"] = pd.to_datetime(data["period_start"], format="%d/%m/%Y")
data["period_end"] = pd.to_datetime(data["period_end"], format="%d/%m/%Y")
data.drop(columns=['year'], inplace=True)
data.drop(columns=['month'], inplace=True)

value_month = data['period_start'].dt.to_period('M')
data.insert(loc=4,
            column = 'period_month',
            value = value_month)

pd.set_option('future.no_silent_downcasting', True)
data['border_ru_flag'] = data['border_ru_flag'].replace({'Yes': 1, 'No': 0}).astype("float")
data['border_b_flag'] = data['border_b_flag'].replace({'Yes': 1, 'No': 0}).astype("float")

days = data['period_month'].dt.to_timestamp().dt.days_in_month

## Add new features
data['alert_hours_per_day'] = data['alert_hours_month'] / days
data['alert_count_per_day'] = data['alerts_count_month'] / days
data['hours_per_alerts'] = (data['alert_hours_month'] / data['alerts_count_month']).replace([np.inf, -np.inf],
np.nan).fillna(0.0) # hours per alert
al = data['alerts_count_month']
data['total_infrastructure_attacks'] = data['educational_infrastructure_attacks_count'] +

```

```
data['healthcare_infrastructure_attacks_count'] + data['energy_infrastructure_attacks_count'] +
data['residential_infrastructure_attacks_count']
```

```
data['night_alerts_proportion'] = data['night_alerts_count_month (22:00-06:00)'].div(al).where(al.ne(0), 0)
data['school_alerts_proportion'] = data['alerts_school_count (08:00-15:00)'].div(al).where(al.ne(0), 0)
data['operational-hours_alert_share'] = data['alerts_malls_count (10:00-21:00)'].div(al).where(al.ne(0), 0)
```

```
data['fatalities_per_alert'] = data['civilian_fatalities_count'].div(al).where(al.ne(0), 0)
```

```
data['malls_non_sch_alerts_count'] = np.abs(data['alerts_malls_count (10:00-21:00)'] -
data['alerts_school_count (08:00-15:00)'])
data['malls_non_sch_alerts_proportion'] = data['malls_non_sch_alerts_count'].div(al).where(al.ne(0), 0)
```

```
## To make the proportion between 0 and 1
for c in ['night_alerts_proportion', 'school_alerts_proportion',
'operational-hours_alert_proportion', 'malls_non_sch_alerts_proportion']:
    bad_mask = ~data[c].between(0, 1, inclusive='both')
    if bad_mask.any():
        print(f'\nError: {c}')
        print(data.loc[bad_mask, c]) # display index and the value
        data[c] = data[c].clip(lower=0.0, upper=1.0)
```

TIME SERIES CLUSTERING

1. Місячна агрегація по тривогах і ударах

```
alerts_month = (
    data
    .groupby(['region_eng', 'period_month'], as_index=False)
    .agg(
        alert_hours_month=('alert_hours_month', 'sum'),
        alerts_count_month=('alerts_count_month', 'sum'),
        night_alerts_count_month=('night_alerts_count_month (22:00-06:00)', 'sum'),
        total_infrastructure_attacks=('total_infrastructure_attacks', 'sum'),
        civilian_fatalities_count=('civilian_fatalities_count', 'sum'),
        educational_infrastructure_attacks_count=('educational_infrastructure_attacks_count', 'sum'),
        healthcare_infrastructure_attacks_count=('healthcare_infrastructure_attacks_count', 'sum'),
        energy_infrastructure_attacks_count=('energy_infrastructure_attacks_count', 'sum'),
        residential_infrastructure_attacks_count=('residential_infrastructure_attacks_count', 'sum'),
        malls_alerts_count_month=('alerts_malls_count (10:00-21:00)', 'sum'),
        school_alerts_count_month=('alerts_school_count (08:00-15:00)', 'sum'),
        malls_non_sch_alerts_count_month=('malls_non_sch_alerts_count', 'sum'),
    )
)
```

```

# 2. Інтенсивність у перерахунку на день / подію
days = alerts_month['period_month'].dt.to_timestamp().dt.days_in_month
alerts_month['alert_hours_per_day'] = alerts_month['alert_hours_month'] / days
alerts_month['alert_count_per_day'] = alerts_month['alerts_count_month'] / days
alerts_month['hours_per_alerts'] = (
    alerts_month['alert_hours_month']
    / alerts_month['alerts_count_month'].replace(0, np.nan)
)

# 3. Структурні пропорції
total_alerts = alerts_month['alerts_count_month'].replace(0, np.nan)

alerts_month['night_alerts_proportion'] = (
    alerts_month['night_alerts_count_month'] / total_alerts
)

alerts_month['operational-hours_alert_share'] = (
    alerts_month['malls_alerts_count_month'] / total_alerts
)

alerts_month['school_alerts_proportion'] = (
    alerts_month['school_alerts_count_month'] / total_alerts
)

alerts_month['fatalities_proportion'] = (
    alerts_month['civilian_fatalities_count'] / total_alerts
)

alerts_month['infra_attacks_per_alert'] = (
    alerts_month['total_infrastructure_attacks'] / total_alerts
)

# після розрахунку інтенсивностей/пропорцій
alerts_month["hours_per_alerts"] = alerts_month["hours_per_alerts"].replace([np.inf, -np.inf],
np.nan).fillna(0.0)

for c in ["night_alerts_proportion", "operational-hours_alert_share", "school_alerts_proportion",
        "fatalities_proportion", "infra_attacks_per_alert"]:
    alerts_month[c] = alerts_month[c].replace([np.inf, -np.inf], np.nan).fillna(0.0)

df = alerts_month.copy()

```

```

df['period_month'] = df['period_month'].dt.to_timestamp('M')

# -----
# 0) Helpers: months + tensor build
# -----

def canonical_months(year: int) -> pd.DatetimeIndex:
    # month-end timestamps Jan..Dec
    return pd.date_range(f'{year}-01-31', f'{year}-12-31', freq="ME")

def build_X_raw_one_year(
    df: pd.DataFrame,
    year: int,
    features: Sequence[str],
    aggfunc="mean",
    regions_ref: Optional[List[str]] = None,
) -> Tuple[np.ndarray, List[str]]:
    """
    Returns X_raw: (n_regions, 12, n_features) for one year, WITHOUT per-series scaling.
    Aligns months to Jan..Dec end-of-month.
    """
    d = df.copy()

    # normalize period_month to month-end Timestamp
    if isinstance(d["period_month"].dtype, pd.PeriodDtype):
        d["period_month"] = d["period_month"].dt.to_timestamp("M")
    else:
        d["period_month"] = pd.to_datetime(d["period_month"]).dt.to_period("M").dt.to_timestamp("M")

    months = canonical_months(year)
    d = d[(d["period_month"] >= months.min()) & (d["period_month"] <= months.max())].copy()

    # Determine base region order
    if regions_ref is None:
        regions = sorted(d["region_eng"].dropna().unique().tolist())
    else:
        regions = list(regions_ref)

    # Build feature pivots aligned on regions x months
    mats = []
    for f in features:
        pt = (pd.pivot_table(d, index="region_eng", columns="period_month", values=f, aggfunc=aggfunc)
              .reindex(index=regions, columns=months))

```

```

        .replace([np.inf, -np.inf], np.nan))
    # IMPORTANT: choose imputation policy
    # Here: fill missing with median
    #pt = pt.fillna(pt.mean())
    mats.append(pt.values)

X = np.stack(mats, axis=2) # (n_regions, 12, n_features)
return X, regions

def fit_global_scaler(X_list: List[np.ndarray]) -> StandardScaler:
    Z = np.concatenate([X.reshape(-1, X.shape[-1]) for X in X_list], axis=0)
    return StandardScaler().fit(Z)

def apply_global_scaler(X: np.ndarray, scaler: StandardScaler) -> np.ndarray:
    Z = scaler.transform(X.reshape(-1, X.shape[-1]))
    return Z.reshape(X.shape)

# -----
# 1) Perturbation (noise-only)
# -----
def perturb_noise_only(X: np.ndarray, eps: float, rng: np.random.Generator) -> np.ndarray:
    # after StandardScaler, eps is in std units
    return X + rng.normal(0.0, eps, size=X.shape)

def perturb_bank(eps: float):
    return [("noise", lambda X, rng: perturb_noise_only(X, eps, rng))]

# -----
# 2) Clustering wrappers
# -----
def flatten_2d(X3: np.ndarray) -> np.ndarray:
    n, t, d = X3.shape
    return X3.reshape(n, t * d)

def fit_agglomerative(X2: np.ndarray, K: int, linkage: str = "ward") -> np.ndarray:
    if linkage == "ward":
        return AgglomerativeClustering(n_clusters=K, linkage="ward").fit_predict(X2)
    try:
        return AgglomerativeClustering(n_clusters=K, linkage=linkage, metric="euclidean").fit_predict(X2)
    except TypeError:
        return AgglomerativeClustering(n_clusters=K, linkage=linkage, affinity="euclidean").fit_predict(X2)

```

```

def fit_labels(method: str, X: np.ndarray, cfg: Dict) -> np.ndarray:
    K = int(cfg["n_clusters"])

    if method == "DTW-KMeans":
        model = TimeSeriesKMeans(
            n_clusters=K, metric="dtw",
            n_init=int(cfg.get("n_init", 3)),
            max_iter_barycenter=int(cfg.get("max_iter_barycenter", 20)),
            random_state=int(cfg.get("random_state", 42)),
        )
        return model.fit_predict(X)

    if method == "SoftDTW-KMeans":
        gamma = float(cfg["gamma"])
        model = TimeSeriesKMeans(
            n_clusters=K, metric="softdtw",
            metric_params={"gamma": gamma},
            n_init=int(cfg.get("n_init", 3)),
            max_iter_barycenter=int(cfg.get("max_iter_barycenter", 20)),
            random_state=int(cfg.get("random_state", 42)),
        )
        return model.fit_predict(X)

    if method == "Euclid-KMeans":
        X2 = flatten_2d(X)
        model = KMeans(
            n_clusters=K,
            init=str(cfg.get("init", "k-means++")),
            n_init=int(cfg.get("n_init", 20)),
            max_iter=int(cfg.get("max_iter", 300)),
            tol=float(cfg.get("tol", 1e-4)),
            random_state=int(cfg.get("random_state", 42)),
        )
        return model.fit_predict(X2)

    if method == "Agglo":
        X2 = flatten_2d(X)
        return _fit_agglomerative(X2, K=K, linkage=str(cfg.get("linkage", "ward")))

    raise ValueError(f"Unknown method: {method}")

# -----

```

```

# 3) Stadion (Forest et al. Eq. (6)-(8)) — strict  $|\Omega|$ 
# -----
def min_cluster_size(labels: np.ndarray, K: int) -> int:
    sizes = np.bincount(labels.astype(int), minlength=K)
    return int(sizes.min())

def stab_between_fixed_ref(method: str, X: np.ndarray, cfg: Dict, ref_labels: np.ndarray,
                           D: int, seed: int, eps: float) -> float:
    """
    Eq. (6): mean over D perturbed datasets vs fixed reference partition.
    """
    rng = np.random.default_rng(seed)
    pert = perturb_bank(eps)
    sims = []
    for _, pfun in pert:
        for _ in range(D):
            Xp = pfun(X, rng)
            lab = fit_labels(method, Xp, cfg)
            sims.append(adjusted_rand_score(ref_labels, lab))
    return float(np.mean(sims)) if sims else 0.0

def stab_within_fixed_ref(method: str, X: np.ndarray, cfg: Dict, ref_labels: np.ndarray,
                           Omega=(2,3), D: int = 10, seed: int = 43, eps: float = 0.03) -> float:
    """
    Eq. (7) strictly:
    - divisor is  $|\Omega|$  (constant)
    - weight  $N_k/N$ 
    - if  $N_k < K_0$  then that  $K_0$  contributes 0 (term absent -> treated as 0)
    """
    rng = np.random.default_rng(seed)
    pert = perturb_bank(eps)

    K = int(cfg["n_clusters"])
    N = X.shape[0]
    Omega = tuple(int(k0) for k0 in Omega)
    denom = float(len(Omega)) if len(Omega) else 1.0

    total = 0.0
    for k in range(K):
        idx = np.where(ref_labels == k)[0]
        Nk = len(idx)
        if Nk == 0:

```

```

        continue

    wk = Nk / float(N)
    Xk = X[idx]

    sum_K0 = 0.0
    for K0 in Omega:
        if K0 < 2 or Nk < K0:
            continue # contributes 0

        cfg_in = dict(cfg)
        cfg_in["n_clusters"] = int(K0)

        #  $Q_{\{K0\}^{\{k\}}} = A(Ck, K0)$ 
        ref_in = fit_labels(method, Xk, cfg_in)

        sims = []
        for _, pfun in pert:
            for _ in range(D):
                Xkp = pfun(Xk, rng)
                lab_in = fit_labels(method, Xkp, cfg_in)
                sims.append(adjusted_rand_score(ref_in, lab_in))

        sum_K0 += float(np.mean(sims)) if sims else 0.0

    total += wk * (sum_K0 / denom)

return float(total)

def stadion(method: str, X: np.ndarray, cfg: Dict,
            Omega=(2,3), D_between=20, D_within=10, eps=0.03,
            base_seed=42, min_cluster_size_constraint: Optional[int] = 2) -> Dict:
    """
    Eq. (8): Stadion = StabB - StabW, same reference CK used for both. (Forest et al.)
    """
    ref = fit_labels(method, X, cfg)
    K = int(cfg["n_clusters"])
    min_sz = min_cluster_size(ref, K)

    if min_cluster_size_constraint is not None and min_sz < int(min_cluster_size_constraint):
        return {
            "stadion": float("-inf"),

```

```

        "stab_between": 0.0,
        "stab_within": 0.0,
        "min_cluster_size": int(min_sz),
        "labels": ref
    }

sb = stab_between_fixed_ref(method, X, cfg, ref, D=D_between, seed=base_seed, eps=eps)
sw = stab_within_fixed_ref(method, X, cfg, ref, Omega=Omega, D=D_within,
                           seed=base_seed+1, eps=eps)

return {
    "stadion": float(sb - sw),
    "stab_between": float(sb),
    "stab_within": float(sw),
    "min_cluster_size": int(min_sz),
    "labels": ref
}

# -----
# 4) Grid search over configs (eps-robust)
# -----

def grid_search(method: str, X: np.ndarray, param_grid: Dict,
                eps_grid=(0.02, 0.03, 0.04),
                Omega=(2,3), D_between=15, D_within=8,
                base_seed=42, min_cluster_size_constraint: Optional[int] = 2):
    keys = list(param_grid.keys())
    rows = []
    best = None

    for eps in eps_grid:
        for combo in product(*[param_grid[k] for k in keys]):
            cfg = dict(zip(keys, combo))
            out = stadion(method, X, cfg,
                          Omega=Omega, D_between=D_between, D_within=D_within,
                          eps=float(eps), base_seed=base_seed,
                          min_cluster_size_constraint=min_cluster_size_constraint)
            row = {"method": method, "eps": float(eps), **cfg,
                  "stadion": out["stadion"],
                  "stab_between": out["stab_between"],
                  "stab_within": out["stab_within"],
                  "min_cluster_size": out["min_cluster_size"]}
            rows.append(row)
            if best is None or row["stadion"] > best["stadion"]:

```

```

        best = row

    df = pd.DataFrame(rows).sort_values(["stadium", "min_cluster_size"], ascending=[False,
False]).reset_index(drop=True)

    # robust aggregation across eps
    group_cols = ["method"] + keys
    agg = (df.groupby(group_cols)
           .agg(stadium_mean=("stadium", "mean"),
                stadium_std=("stadium", "std"),
                sb_mean=("stab_between", "mean"),
                sw_mean=("stab_within", "mean"),
                min_cluster_size_min=("min_cluster_size", "min"))
           .reset_index()
           .sort_values(["stadium_mean", "min_cluster_size_min"], ascending=[False, False])
           .reset_index(drop=True))
    best_cfg = {k: agg.loc[0, k] for k in keys}
    return best_cfg, df, agg

# -----
# 5) MAIN: Build pooled X_pool and run selection
# -----
FEATURES = [
    "alert_count_per_day",
    "hours_per_alerts",
    "night_alerts_proportion",
    "operational-hours_alert_share",
    "infra_attacks_per_alert",
]

YEARS = [2023, 2024, 2025]
BASE_SEED = 42

X_raw_by_year = {}
regions_ref = None
for y in YEARS:
    Xy_raw, regions = build_X_raw_one_year(df, y, FEATURES, regions_ref=regions_ref)
    if regions_ref is None:
        regions_ref = regions
    X_raw_by_year[y] = Xy_raw

# Global scaler across ALL YEARS (risk-level + shape)

```

```

scaler = fit_global_scaler([X_raw_by_year[y] for y in YEARS])
X_by_year = {y: apply_global_scaler(X_raw_by_year[y], scaler) for y in YEARS}
# Build pooled objects: (region, year)
X_pool = np.concatenate([X_by_year[y] for y in YEARS], axis=0) # (N≈72, 12, F)
keys_pool = [(r, y) for y in YEARS for r in regions_ref] # same order as concat

X_pool_raw = np.concatenate([X_raw_by_year[y] for y in YEARS], axis=0)

print("X_pool shape:", X_pool.shape)
print("X_pool_raw shape:", X_pool_raw.shape)
assert X_pool.shape == X_pool_raw.shape, "Raw and standardized tensors must align."

# Soft-DTW gamma ONCE on pooled (for comparability)
gamma_seeds = [0, 1, 2, 3, 4, 42, 43, 44, 46]
gamma_vals = []

for gamma_seed in gamma_seeds:
    g = float(
        gamma_soft_dtw(
            X_pool,
            n_samples=min(400, X_pool.shape[0]),
            random_state=gamma_seed
        )
    )
    print(f'gamma_seed = {gamma_seed}, gamma = {g}')
    gamma_vals.append(g)

median_gamma = float(np.median(gamma_vals))
print("median_gamma =", median_gamma)

LL_METHODS = ["DTW-KMeans", "SoftDTW-KMeans", "Euclid-KMeans", "Agglo"]

param_grids = {
    "DTW-KMeans": {
        "n_clusters": [4, 5, 6, 7],
        "n_init": [2, 5, 10],
        "max_iter_barycenter": [30, 50],
        "random_state": [BASE_SEED],
    },
    "SoftDTW-KMeans": {
        "n_clusters": [4, 5, 6, 7],
        "n_init": [2, 5, 10],
    }
}

```

```

    "max_iter_barycenter": [30, 50],
    #"gamma": [auto_gamma], # FIXED gamma
    "gamma": [0.75*median_gamma, median_gamma, 1.5*median_gamma],
    "random_state": [BASE_SEED],
},
"Euclid-KMeans": {
    "n_clusters": [4, 5, 6, 7],
    "n_init": [10, 20],
    "max_iter": [300],
    "tol": [1e-4],
    "init": ["k-means++"],
    "random_state": [BASE_SEED],
},
"Agglo": {
    "n_clusters": [4, 5, 6, 7],
    "linkage": ["ward", "average", "complete"],
}
}

# Practical constraint for typology:
MIN_CLUSTER_SIZE = 2 # set None if you want allow singleton as "anomaly mode"

all_compare = []
best_cfgs = {}

for m in ["Euclid-KMeans", "Agglo", "DTW-KMeans", "SoftDTW-KMeans"]:
    best_cfg, df_full, df_agg = grid_search(
        m, X_pool, param_grids[m],
        eps_grid=(0.02, 0.03, 0.04),
        Omega=(2,3),
        D_between=12, D_within=6,
        base_seed=BASE_SEED,
        min_cluster_size_constraint=MIN_CLUSTER_SIZE
    )
    best_cfgs[m] = best_cfg
    all_compare.append({
        "method": m,
        "best_cfg": best_cfg,
        "stadium_mean": float(df_agg.loc[0, "stadium_mean"]),
        "stadium_std": float(df_agg.loc[0, "stadium_std"]) if not np.isnan(df_agg.loc[0, "stadium_std"]) else 0.0,
        "min_cluster_size_min": int(df_agg.loc[0, "min_cluster_size_min"]),
        "sb_mean": float(df_agg.loc[0, "sb_mean"]),
    })

```

```

        "sw_mean": float(df_agg.loc[0, "sw_mean"]),
    })

    compare_df = pd.DataFrame(all_compare).sort_values("stadion_mean",
ascending=False).reset_index(drop=True)
    display(compare_df)

    ## графік порівняння методів за Stadion
    fig, ax = plt.subplots(figsize=(7, 4))
    x = np.arange(len(compare_df))
    y = compare_df["stadion_mean"].values
    yerr = compare_df["stadion_std"].values

    ax.bar(x, y, yerr=yerr, capsize=4, color = 'darkblue')
    ax.set_xticks(x)
    ax.set_xticklabels(compare_df["method"])
    ax.set_ylabel("Stadion mean")
    ax.set_title("Method comparison by Stadion")
    ax.axhline(0, linestyle="--", linewidth=1)
    ax.grid(alpha=0.5, linestyle=":")
    plt.tight_layout()
    plt.show()

def flatten_2d(X3: np.ndarray) -> np.ndarray:
    n, t, d = X3.shape
    return X3.reshape(n, t * d)

def fit_model_and_labels(method: str, X: np.ndarray, cfg: dict):
    """
    Fit clustering model and return both model and labels.
    """
    K = int(cfg["n_clusters"])

    if method == "DTW-KMeans":
        model = TimeSeriesKMeans(
            n_clusters=K,
            metric="dtw",
            n_init=int(cfg.get("n_init", 3)),
            max_iter_barycenter=int(cfg.get("max_iter_barycenter", 20)),
            random_state=int(cfg.get("random_state", 42)),
        )
        labels = model.fit_predict(X)

```

```

return model, labels

if method == "SoftDTW-KMeans":
    model = TimeSeriesKMeans(
        n_clusters=K,
        metric="softdtw",
        metric_params={"gamma": float(cfg["gamma"])},
        n_init=int(cfg.get("n_init", 3)),
        max_iter_barycenter=int(cfg.get("max_iter_barycenter", 20)),
        random_state=int(cfg.get("random_state", 42)),
    )
    labels = model.fit_predict(X)
    return model, labels

if method == "KShape":
    model = KShape(
        n_clusters=K,
        n_init=int(cfg.get("n_init", 3)),
        max_iter=int(cfg.get("max_iter", 100)),
        tol=float(cfg.get("tol", 1e-6)),
        random_state=int(cfg.get("random_state", 42)),
    )
    labels = model.fit_predict(X)
    return model, labels

if method == "Euclid-KMeans":
    X2 = flatten_2d(X)
    model = KMeans(
        n_clusters=K,
        init=str(cfg.get("init", "k-means++")),
        n_init=int(cfg.get("n_init", 20)),
        max_iter=int(cfg.get("max_iter", 300)),
        tol=float(cfg.get("tol", 1e-4)),
        random_state=int(cfg.get("random_state", 42)),
    )
    labels = model.fit_predict(X2)
    return model, labels

if method == "Agglo":
    X2 = flatten_2d(X)
    linkage = str(cfg.get("linkage", "ward"))
    try:

```

```

        model = AgglomerativeClustering(n_clusters=K, linkage=linkage, metric="euclidean")
    except TypeError:
        model = AgglomerativeClustering(n_clusters=K, linkage=linkage, affinity="euclidean")
    labels = model.fit_predict(X2)
    return model, labels

raise ValueError(f"Unknown method: {method}")

## Fit final model on pooled standardized data
FINAL_METHOD = "SoftDTW-KMeans"
FINAL_CFG = best_cfgs[FINAL_METHOD]
print("FINAL_METHOD =", FINAL_METHOD)
print("FINAL_CFG =", FINAL_CFG)
final_model, final_labels = fit_model_and_labels(FINAL_METHOD, X_pool, FINAL_CFG)
labels_df = pd.DataFrame({
    "region": [k[0] for k in keys_pool],
    "year": [k[1] for k in keys_pool],
    "cluster": final_labels.astype(int)
}).sort_values(["region", "year"]).reset_index(drop=True)

## cluster sizes and cluster x year table
cluster_sizes = (
    labels_df["cluster"]
    .value_counts()
    .sort_index()
    .rename_axis("cluster")
    .reset_index(name="N")
)

cluster_year_counts = pd.crosstab(labels_df["cluster"], labels_df["year"])

display(cluster_sizes)
display(cluster_year_counts)

## Region trajectories and transitions
pivot = labels_df.pivot_table(
    index="region",
    columns="year",
    values="cluster",
    aggfunc="first"
).sort_index()

```

```

trajectory_df = pivot.copy()
trajectory_df["trajectory"] = trajectory_df.apply(
    lambda row: " -> ".join(
        str(int(v)) if pd.notna(v) else "NA"
        for v in [row.get(2023), row.get(2024), row.get(2025)]
    ),
    axis=1
)

display(trajectory_df)

transitions = []
for region, row in pivot.iterrows():
    y23, y24, y25 = row.get(2023, np.nan), row.get(2024, np.nan), row.get(2025, np.nan)

    if pd.notna(y23) and pd.notna(y24):
        transitions.append({
            "region": region,
            "from_year": 2023,
            "to_year": 2024,
            "from_cluster": int(y23),
            "to_cluster": int(y24)
        })

    if pd.notna(y24) and pd.notna(y25):
        transitions.append({
            "region": region,
            "from_year": 2024,
            "to_year": 2025,
            "from_cluster": int(y24),
            "to_cluster": int(y25)
        })

transitions_df = pd.DataFrame(transitions)

tm_counts = pd.crosstab(
    transitions_df["from_cluster"],
    transitions_df["to_cluster"]
)

tm_row_share = pd.crosstab(
    transitions_df["from_cluster"],

```

```

    transitions_df["to_cluster"],
    normalize="index"
).round(3)

print('Transitions per Region')
display(transitions_df.head())

print('\nTransitions Number per Cluster')
display(tm_counts)

print('\nTransitions Share per Cluster')
display(tm_row_share)

## heatmap cluster × year
ct = pd.crosstab(labels_df["cluster"], labels_df["year"])

fig, ax = plt.subplots(figsize=(6, 4))
im = ax.imshow(ct.values, aspect="auto", cmap='YlOrBr')

ax.set_xticks(range(ct.shape[1]))
ax.set_xticklabels(ct.columns)
ax.set_yticks(range(ct.shape[0]))
ax.set_yticklabels(ct.index)

for i in range(ct.shape[0]):
    for j in range(ct.shape[1]):
        ax.text(j, i, str(ct.iloc[i, j]), ha="center", va="center", size = 11)

ax.set_xlabel("Year")
ax.set_ylabel("Cluster")
ax.set_title("Cluster distribution by year")
plt.colorbar(im, ax=ax)
plt.tight_layout()
plt.show()

# Summary profiles + monthly profiles
## Summary profiles (standardized and raw)
## mean values over all objects and all months in each cluster.

profiles_std = []
profiles_raw = []

```

```

for c in sorted(np.unique(final_labels)):
    idx = np.where(final_labels == c)[0]

    Xc_std = X_pool[idx]    # standardized
    Xc_raw = X_pool_raw[idx] # raw

    prof_std = Xc_std.mean(axis=(0, 1))
    prof_raw = Xc_raw.mean(axis=(0, 1))

    profiles_std.append([int(c), len(idx), *prof_std.tolist()])
    profiles_raw.append([int(c), len(idx), *prof_raw.tolist()])

profiles_std_df = pd.DataFrame(
    profiles_std, columns=["cluster", "N", *FEATURES]
).sort_values("cluster")

profiles_raw_df = pd.DataFrame(
    profiles_raw, columns=["cluster", "N", *FEATURES]
).sort_values("cluster")

print('Summary Profiles mean (standardized data)')
display(profiles_std_df)

print('\nSummary Profiles mean (raw data)')
display(profiles_raw_df)
## Heatmap summary profiles
def plot_profiles_heatmap(profiles_df: pd.DataFrame, feature_cols, title="Cluster profiles (standardized)"):
    mat = profiles_df.set_index("cluster")[feature_cols]

    fig, ax = plt.subplots(figsize=(8, 6))
    im = ax.imshow(mat.values, aspect="auto", cmap='YlOrBr')

    ax.set_xticks(range(len(feature_cols)))
    ax.set_xticklabels(feature_cols, rotation=45)
    ax.set_yticks(range(mat.shape[0]))
    ax.set_yticklabels(mat.index)

    for i in range(mat.shape[0]):
        for j in range(mat.shape[1]):
            ax.text(j, i, f"{mat.iloc[i, j]:.2f}", ha="center", va="center", size = 11)

    ax.set_xlabel("Feature")

```

```

ax.set_ylabel("Cluster")
ax.set_title(title)
plt.colorbar(im, ax=ax)
plt.tight_layout()
plt.show()

plot_profiles_heatmap(
    profiles_std_df,
    FEATURES,
    title="Summary cluster profiles mean (standardized)"
)
plot_profiles_heatmap(
    profiles_raw_df,
    FEATURES,
    title="Summary cluster profiles mean (raw)"
)

## Risk Score and Risk Rank
# =====

# STEP 1. Annual region-year metrics
# Рахуємо річні показники для кожної пари region-year
# =====

annual_metrics_df = (
    df.assign(year=df["period_month"].dt.year)
    .groupby(["region_eng", "year"], as_index=False)
    .agg(
        alert_hours_total=("alert_hours_month", "sum"),
        alerts_total=("alerts_count_month", "sum"),
        night_alerts_total=("night_alerts_count_month", "sum"),
        malls_alerts_total=("malls_alerts_count_month", "sum"),
        infra_attacks_total=("total_infrastructure_attacks", "sum"),
        days_total=("period_month", lambda s: pd.to_datetime(s).dt.days_in_month.sum())
    )
    .rename(columns={"region_eng": "region"})
)

den = annual_metrics_df["alerts_total"].replace(0, np.nan)

annual_metrics_df["alert_count_per_day"] = annual_metrics_df["alerts_total"] /
annual_metrics_df["days_total"]
annual_metrics_df["alert_hours_per_day"] = annual_metrics_df["alert_hours_total"] /

```

```

annual_metrics_df["days_total"]

annual_metrics_df["hours_per_alerts"] = annual_metrics_df["alert_hours_total"] / den
annual_metrics_df["night_alerts_proportion"] = annual_metrics_df["night_alerts_total"] / den
annual_metrics_df["operational-hours_alert_share"] = annual_metrics_df["malls_alerts_total"] / den
annual_metrics_df["infra_attacks_per_alert"] = annual_metrics_df["infra_attacks_total"] / den


for c in ["hours_per_alerts", "night_alerts_proportion", "operational-hours_alert_share",
"infra_attacks_per_alert"]:
    annual_metrics_df[c] = annual_metrics_df[c].replace([np.inf, -np.inf], np.nan).fillna(0.0)


display(annual_metrics_df.head())


INDEX_FEATURES = [
    "alert_count_per_day",
    "alert_hours_per_day",
    "night_alerts_proportion",
    "operational-hours_alert_share",
    "infra_attacks_per_alert"
]


index_scaler = StandardScaler()
Z = index_scaler.fit_transform(annual_metrics_df[INDEX_FEATURES])


annual_index_df = annual_metrics_df.copy()
z_cols = [f"z_{c}" for c in INDEX_FEATURES]
annual_index_df[z_cols] = Z


## інтенсивність навантаження (як часто тривоги, скільки часу вони займають загалом)
annual_index_df["exposure_score"] = annual_index_df[
    ["z_alert_count_per_day", "z_alert_hours_per_day"]
].mean(axis=1)


## структурна напруженість режиму (наскільки тривоги зміщені в ніч, наскільки пов'язані з
інфраструктурним компонентом)
annual_index_df["structural_score"] = annual_index_df[
    ["z_night_alerts_proportion", "z_infra_attacks_per_alert"]
].mean(axis=1)


## середнє цих двох осей
annual_index_df["annual_risk_index"] = annual_index_df[
    ["exposure_score", "structural_score"]
].mean(axis=1)

```

```

# hours_per_alerts лишаємо як додаткову характеристику
duration_scaler = StandardScaler()
annual_index_df["z_hours_per_alerts"] = duration_scaler.fit_transform(
    annual_index_df[["hours_per_alerts"]]
)

display(
    annual_index_df[
        [
            "region", "year",
            "alert_count_per_day", "alert_hours_per_day",
            "hours_per_alerts", "night_alerts_proportion", "infra_attacks_per_alert",
            "exposure_score", "structural_score",
            "structural_score_alt",
            "annual_risk_index"
        ]
    ].head()
)

# =====
# STEP 3. Merge annual indices into labels_df
# =====

for c in ["exposure_score", "structural_score", "structural_score_alt", "annual_risk_index",
"annual_risk_index_alt", "cluster_severity_rank"]:
    if c in labels_df.columns:
        labels_df = labels_df.drop(columns=[c])

labels_df = labels_df.merge(
    annual_index_df[["region", "year", "exposure_score", "structural_score",
"structural_score_alt", "annual_risk_index", "annual_risk_index_alt"]],
    on=["region", "year"],
    how="left",
    validate="one_to_one"
)

# =====
# STEP 4. Order clusters by median annual_risk_index
# =====

```

```

cluster_order_df = (
    labels_df.groupby("cluster", as_index=False)
    .agg(
        N=("region", "count"),
        median_exposure=("exposure_score", "median"),
        median_structural=("structural_score", "median"),
        median_structural_alt=("structural_score_alt", "median"),
        cluster_order_score=("annual_risk_index", "median"),
        cluster_order_score_alt=("annual_risk_index_alt", "median")
    )
    .sort_values(["cluster_order_score", "median_exposure", "median_structural"])
    .reset_index(drop=True)
)

cluster_order_df["cluster_severity_rank"] = range(len(cluster_order_df))

cluster_rank_map = dict(
    zip(cluster_order_df["cluster"], cluster_order_df["cluster_severity_rank"])
)

labels_df["cluster_severity_rank"] = labels_df["cluster"].map(cluster_rank_map)

cluster_order_plot = cluster_order_df.sort_values("cluster_severity_rank").reset_index(drop=True)

# Масштаб розміра точок по cluster_order_score
score_min = cluster_order_plot["cluster_order_score"].min()
score_max = cluster_order_plot["cluster_order_score"].max()
i
cluster_order_plot["marker_size"] = 700 + 1700 * (
    (cluster_order_plot["cluster_order_score"] - score_min) / (score_max - score_min)
)

# -----
# Figure
# -----
fig, ax = plt.subplots(figsize=(10, 7.5))
fig.patch.set_facecolor("white")
ax.set_facecolor("#fcfcfc")

# Axis limits with padding
x_pad = 0.12 * (cluster_order_plot["median_exposure"].max() -
cluster_order_plot["median_exposure"].min())

```

```

y_pad = 0.15 * (cluster_order_plot["median_structural"].max() -
cluster_order_plot["median_structural"].min())

x_min = cluster_order_plot["median_exposure"].min() - x_pad
x_max = cluster_order_plot["median_exposure"].max() + x_pad
y_min = cluster_order_plot["median_structural"].min() - y_pad
y_max = cluster_order_plot["median_structural"].max() + y_pad

ax.set_xlim(x_min, x_max)
ax.set_ylim(y_min, y_max)

# -----
# Quadrant shading
# -----
ax.axvspan(x_min, 0, ymin=(0 - y_min)/(y_max - y_min), ymax=1, alpha=0.05)
ax.axvspan(0, x_max, ymin=(0 - y_min)/(y_max - y_min), ymax=1, alpha=0.03)
ax.axhspan(y_min, 0, xmin=(0 - x_min)/(x_max - x_min), xmax=1, alpha=0.03)
ax.axhspan(0, y_max, xmin=(0 - x_min)/(x_max - x_min), xmax=1, alpha=0.05)

# -----
# Grid
# -----
ax.grid(True, which="major", linestyle="--", linewidth=0.8, alpha=0.35)
ax.minorticks_on()
ax.grid(True, which="minor", linestyle=":", linewidth=0.5, alpha=0.18)

# Zero lines
ax.axhline(0, color="#1f77b4", linestyle="--", linewidth=1.2, alpha=0.8)
ax.axvline(0, color="#1f77b4", linestyle="--", linewidth=1.2, alpha=0.8)

# -----
# Scatter
# -----
cmap = plt.cm.viridis
norm = mpl.colors.Normalize(
    vmin=cluster_order_plot["cluster_severity_rank"].min(),
    vmax=cluster_order_plot["cluster_severity_rank"].max()
)

sc = ax.scatter(
    cluster_order_plot["median_exposure"],
    cluster_order_plot["median_structural"],

```

```

s=cluster_order_plot["marker_size"],
c=cluster_order_plot["cluster_severity_rank"],
cmap=cmap,
norm=norm,
edgecolor="white",
linewidth=1.6,
alpha=0.92,
zorder=3
)

# -----
# Labels
# -----
for _, row in cluster_order_plot.iterrows():
    x = row["median_exposure"]
    y = row["median_structural"]

    # smarter offset depending on point position
    dx = 10 if x <= 0 else -70
    dy = 10 if y <= 0 else 8

    label = (
        f'Cluster {int(row['cluster'])}\n'
        f'rank = {int(row['cluster_severity_rank'])}\n'
        f'score = {row['cluster_order_score']:.3f}'
    )

    ax.annotate(
        label,
        (x, y),
        xytext=(dx, dy),
        textcoords="offset points",
        fontsize=10.5,
        color="#222222",
        bbox=dict(
            boxstyle="round,pad=0.3",
            fc="white",
            ec="#666666",
            lw=0.8,
            alpha=0.92
        ),
        zorder=4

```

```

    )

for _, row in cluster_order_plot.iterrows():
    ax.text(
        row["median_exposure"],
        row["median_structural"],
        f"N={int(row['N'])}",
        ha="center",
        va="center",
        fontsize=10,
        weight="bold",
        color="black",
        zorder=5
    )

ax.set_title(
    "Clusters by median exposure and median structural values",
    fontsize=16,
    #weight="bold",
    pad=14
)

ax.set_xlabel("Median Exposure", fontsize=13)
ax.set_ylabel("Median Structural", fontsize=13)

# cleaner spines
ax.spines["top"].set_visible(False)
ax.spines["right"].set_visible(False)
ax.spines["left"].set_alpha(0.5)
ax.spines["bottom"].set_alpha(0.5)

# Colorbar
cbar = plt.colorbar(sc, ax=ax, pad=0.03)
cbar.set_label("Cluster Severity Rank", fontsize=11)
cbar.set_ticks([0, 1, 2, 3, 4])

plt.tight_layout()
plt.show()

df2 = cluster_order_df.sort_values("cluster_severity_rank")

values = df2["cluster_order_score"].values

```

```

# нормалізація (0–1)
norm = plt.Normalize(values.min(), values.max())

cmap = cm.get_cmap("YlOrBr")

colors = cm.YlOrBr(np.linspace(0.3, 0.9, len(df2)))

plt.figure(figsize=(8, 5))
plt.barh(
    y=df2["cluster"].astype(str),
    width=values,
    color=colors
)

plt.xlabel("Median annual risk index")
plt.ylabel("Cluster")

# =====
# STEP 5. Rebuild transition direction from cluster order
# =====

for c in ["from_cluster_severity_rank", "to_cluster_severity_rank", "direction_by_index"]:
    if c in transitions_df.columns:
        transitions_df = transitions_df.drop(columns=[c])

transitions_df["from_cluster_severity_rank"] = transitions_df["from_cluster"].map(cluster_rank_map)
transitions_df["to_cluster_severity_rank"] = transitions_df["to_cluster"].map(cluster_rank_map)

transitions_df["direction_by_index"] = np.where(
    transitions_df["to_cluster_severity_rank"] > transitions_df["from_cluster_severity_rank"],
    "up",
    np.where(
        transitions_df["to_cluster_severity_rank"] < transitions_df["from_cluster_severity_rank"],
        "down",
        "same"
    )
)

## перехід у кластер з вищою типовою медіаною annual_risk_index
display(transitions_df.head())

pivot = labels_df.pivot_table(

```

```

        index="region",
        columns="year",
        values="cluster",
        aggfunc="first"
    )

    pivot = pivot.sort_values(list(pivot.columns))

    fig, ax = plt.subplots(figsize=(6, 0.35 * len(pivot) + 2))
    im = ax.imshow(pivot.values, aspect="auto", cmap = 'YlOrBr')

    ax.set_xticks(range(pivot.shape[1]))
    ax.set_xticklabels(pivot.columns)
    ax.set_yticks(range(pivot.shape[0]))
    ax.set_yticklabels(pivot.index)

    for i in range(pivot.shape[0]):
        for j in range(pivot.shape[1]):
            ax.text(j, i, str(int(pivot.iloc[i, j])), ha="center", va="center", fontsize=8)

    ax.set_xlabel("Year")
    ax.set_ylabel("Region")
    ax.set_title("Region trajectories across years")
    plt.colorbar(im, ax=ax)
    plt.tight_layout()
    plt.show()

# Longitude, Latitude (Soft-DTW)
labels_df_softdtw = labels_df.copy()
region_coords_df = pd.read_excel("data/ukr_admin_boundaries_regions.xlsx")
region_coords_df.head()
map_reg_names = { 'Cherkasy': 'Cherkaska',
                  'Chernihiv': 'Chernihivska',
                  'Chernivtsi': 'Chernivetska',
                  'Dnipropetrovsk': 'Dnipropetrovska',
                  'Donetsk': 'Donetska',
                  'Ivano-Frankivsk': 'Ivano-Frankivska',
                  'Kharkiv': 'Kharkivska',
                  'Kherson': 'Khersonska',
                  'Khmelnysky': 'Khmelnyska',
                  'Kirovohrad': 'Kirovohradska',
                  'Kyiv city': 'Kyiv',

```

```

'Kyiv': 'Kyivska',
'Lviv': 'Lvivska',
'Mykolaiv': 'Mykolaivska',
'Odesa': 'Odeska',
'Poltava': 'Poltavska',
'Rivne': 'Rivnenska',
'Sumy': 'Sumska',
'Ternopil': 'Ternopilska',
'Vinnytsia': 'Vinnytska',
'Volyn': 'Volynska',
'Zakarpattia': 'Zakarpatska',
'Zaporizhzhia': 'Zaporizka',
'Zhytomyr': 'Zhytomyrska'
}

labels_df_softdtw["region"] = labels_df_softdtw["region"].map(map_reg_names)

# merge with final labels
map_df = labels_df_softdtw.merge(region_coords_df, on="region", how="left")

# перевірка
print(map_df.shape)

# чи всі координати підтягнулися
missing_coords = map_df[map_df["lat"].isna() | map_df["lon"].isna()]
missing_coords

# geojson
gdf = gpd.read_file("data/ukr_admin1.geojson")
adm1_gdf = gdf[["adm1_name", "geometry"]].copy()
adm1_gdf = adm1_gdf.rename(columns={"adm1_name": "region"})
adm1_gdf.head()
adm1_gdf = adm1_gdf[adm1_gdf["region"].isin(map_df["region"].unique())].copy()

print(adm1_gdf.shape)
print(sorted(adm1_gdf["region"].unique()))
choro_gdf = adm1_gdf.merge(
    map_df,
    on="region",
    how="inner",
    validate="one_to_many"
)

print(choro_gdf.shape)
choro_gdf.head()

```

```

# Choropleth map
import matplotlib as mpl

def plot_choropleth_by_year(gdf_yearly, value_col, title_prefix, cmap="viridis"):
    years = sorted(gdf_yearly["year"].unique())
    fig, axes = plt.subplots(1, len(years), figsize=(6 * len(years), 7), constrained_layout=True)

    if len(years) == 1:
        axes = [axes]

    vmin = gdf_yearly[value_col].min()
    vmax = gdf_yearly[value_col].max()
    norm = mpl.colors.Normalize(vmin=vmin, vmax=vmax)

    for ax, year in zip(axes, years):
        sub = gdf_yearly[gdf_yearly["year"] == year].copy()

        sub.plot(
            column=value_col,
            cmap=cmap,
            linewidth=0.6,
            edgecolor="black",
            legend=False,
            norm=norm,
            ax=ax
        )

        ax.set_title(f"{title_prefix}, {year}")
        ax.set_axis_off()

    sm = mpl.cm.ScalarMappable(cmap=cmap, norm=norm)
    sm._A = []
    cbar = fig.colorbar(sm, ax=axes, fraction=0.03, pad=0.02)
    cbar.set_label(value_col)

    plt.show()

plot_choropleth_by_year(
    choro_gdf,
    value_col="annual_risk_index",
    title_prefix="Інтегральний річний ризик"
)

```

```

plot_choropleth_by_year(
    choro_gdf,
    value_col="exposure_score",
    title_prefix="Інтенсивність режиму"
)

plot_choropleth_by_year(
    choro_gdf,
    value_col="structural_score",
    title_prefix="Структурна напруженість"
)

def plot_choropleth_rank_by_year(gdf_yearly, value_col="cluster_severity_rank",
                                title_prefix="Ранг режиму ризику", cmap="plasma"):
    years = sorted(gdf_yearly["year"].unique())
    fig, axes = plt.subplots(1, len(years), figsize=(6 * len(years), 7), constrained_layout=True)

    if len(years) == 1:
        axes = [axes]

    vmin = int(gdf_yearly[value_col].min())
    vmax = int(gdf_yearly[value_col].max())
    boundaries = np.arange(vmin - 0.5, vmax + 1.5, 1)
    cmap_obj = plt.get_cmap(cmap, vmax - vmin + 1)
    norm = mpl.colors.BoundaryNorm(boundaries, cmap_obj.N)

    for ax, year in zip(axes, years):
        sub = gdf_yearly[gdf_yearly["year"] == year].copy()

        sub.plot(
            column=value_col,
            cmap=cmap_obj,
            linewidth=0.6,
            edgecolor="black",
            legend=False,
            norm=norm,
            ax=ax
        )

        ax.set_title(f"{title_prefix}, {year}")
        ax.set_axis_off()

```

```

sm = mpl.cm.ScalarMappable(cmap=cmap_obj, norm=norm)
sm._A = []
cbar = fig.colorbar(sm, ax=axes, fraction=0.03, pad=0.02)
cbar.set_label(value_col)
cbar.set_ticks(range(vmin, vmax + 1))

plt.show()

plot_choropleth_rank_by_year(choro_gdf)

# Spearman 3 lon/lat
tmp = choro_gdf.copy()
tmp["rep_point"] = tmp.geometry.representative_point()
tmp["lon"] = tmp["rep_point"].x
tmp["lat"] = tmp["rep_point"].y

# 3. Spearman
def spatial_gradient_by_year(gdf_yearly, value_col):
    rows = []

    for year in sorted(gdf_yearly["year"].unique()):
        sub = gdf_yearly[gdf_yearly["year"] == year].copy()

        rho_lon, p_lon = spearmanr(sub[value_col], sub["lon"])
        rho_lat, p_lat = spearmanr(sub[value_col], sub["lat"])

        rows.append({
            "year": year,
            "value_col": value_col,
            "rho_lon": rho_lon,
            "p_lon": p_lon,
            "rho_lat": rho_lat,
            "p_lat": p_lat
        })

    return pd.DataFrame(rows)

spatial_gradient_df = spatial_gradient_by_year(tmp, value_col="annual_risk_index")
spatial_gradient_df.round(3)

# Global Moran's I

```

```

def compute_global_moran_by_year(gdf_yearly, value_col, permutations=999):
    results = []

    for year in sorted(gdf_yearly["year"].unique()):
        sub = gdf_yearly[gdf_yearly["year"] == year].copy()

        sub = sub.dropna(subset=[value_col, "geometry"]).reset_index(drop=True)

        w = Queen.from_dataframe(sub, use_index=False)
        w.transform = "r" # row-standardized weights

        y = sub[value_col].astype(float).values

        mi = Moran(y, w, permutations=permutations)

        results.append({
            "year": year,
            "value_col": value_col,
            "moran_I": mi.I,
            "p_sim": mi.p_sim,
            "EI": mi.EI
        })

    return pd.DataFrame(results)

moran_risk = compute_global_moran_by_year(choro_gdf, "annual_risk_index")
moran_exposure = compute_global_moran_by_year(choro_gdf, "exposure_score")
moran_structural = compute_global_moran_by_year(choro_gdf, "structural_score")

print("Global Moran's I: annual_risk_index")
display(moran_risk.round(3))

print("Global Moran's I: exposure_score")
display(moran_exposure.round(3))

print("Global Moran's I: structural_score")
display(moran_structural.round(3))

bar_colors = ['orange', 'darkblue', 'purple']
fig, ax = plt.subplots(figsize=(7, 5))
bars = ax.bar(moran_risk['year'], moran_risk['moran_I'],
              color=bar_colors)
labels = [f"I: {i:.3f}\np: {p:.3f}" for i, p in zip(moran_risk['moran_I'], moran_risk['p_sim'])]

```

```

ax.bar_label(bars, labels=labels, padding=5, fontsize=9, fontweight='bold')
ax.set_xticks(moran_risk['year'])
ax.set_ylabel("Global Moran's I")
ax.set_title("Global Moran's I for Annual Risk Index by year")
ax.set_ylim(0, 0.95)
ax.axhline(0, linestyle="--", linewidth=1, color='black')
ax.grid(alpha=0.3, linestyle=":")
plt.tight_layout()
plt.show()

```

Survival Function

$S(t)$ = ймовірність того, що подія ще не настала до часу t

$S(t)$ = ймовірність того, що регіон ще не перейшов до більш ризикового режиму після t міжрічних переходів.

```

labels_with_ranks = labels_df.copy()
labels_with_ranks = labels_with_ranks[['region', 'year', 'cluster',
                                       'annual_risk_index', 'cluster_severity_rank']]
labels_with_ranks

labels_with_ranks = labels_with_ranks.sort_values(["region", "year"])

survival_rows = []

for region, g in labels_with_ranks.groupby("region"):
    g = g.sort_values("year")
    years = g["year"].to_list()
    ranks = g["cluster_severity_rank"].to_list()
    clusters = g["cluster"].to_list()

    event = 0
    duration = len(years) - 1 # максимум 2 переходи: 2023->2024 і 2024->2025
    event_year = None

    # перше погіршення відносно попереднього року
    for i in range(1, len(ranks)):
        if ranks[i] > ranks[i - 1]:
            event = 1
            duration = i
            event_year = years[i]

```

```

break

survival_rows.append({
    "region": region,
    "duration": duration,    # кількість міжрічних переходів до ПЕРШОГО зафіксованого
погіршення режиму ризику
    "event": event,         # бінарний індикатор: 1 = хоча б одне перше погіршення було зафіксовано
    "event_year": event_year, # рік коли відбулося погіршення ризику
    "rank_2023": ranks[0],
    "rank_2024": ranks[1] if len(ranks) > 1 else np.nan,
    "rank_2025": ranks[2] if len(ranks) > 2 else np.nan,
    "cluster_2023": clusters[0],
    "cluster_2024": clusters[1] if len(clusters) > 1 else np.nan,
    "cluster_2025": clusters[2] if len(clusters) > 2 else np.nan,
    "trajectory_rank": " -> ".join(map(str, ranks)),
    "trajectory_cluster": " -> ".join(map(str, clusters))
})

surv_df = pd.DataFrame(survival_rows)

display(surv_df.head())

print("\nкількість погіршень ризику:", surv_df["event"].value_counts())

summary_surv = (
    surv_df
    .assign(
        category=np.where(
            surv_df["event"] == 1,
            "Наявність першого погіршення ризику",
            "Без погіршень"
        )
    )
    .groupby(["duration", "event", "category"])
    .agg(n_regions=("region", "count"))
    .reset_index()
)

display(summary_surv)

kmf = KaplanMeierFitter()

```

```

plt.figure(figsize=(8, 5))

# durations 1 = подія сталася на першому переході 2023->2024
# durations 2 = подія сталася на другому переході 2024->2025
# duration = 2, event = 0 == погіршення не сталося до кінця періоду

# event = 1: погіршення було зафіксовано
# event = 0: погіршення не було зафіксовано

kmf.fit(
    durations=surv_df["duration"],
    event_observed=surv_df["event"], # індикатор першого погіршення. -- The occurrence of the event of
interest (e.g., death, failure).
    label="Відсутність погіршення режиму ризику"
)

kmf.plot_survival_function(ci_show=False)

plt.title('Kaplan–Meier крива виживання: Час до погіршення режиму ризику (2023-2025)')
plt.xlabel('Роки')
plt.ylabel('Ймовірність збереження режиму')
plt.xticks([0, 1, 2], ["2023", "2024", "2025"])
plt.grid(alpha=0.3)
plt.tight_layout()
plt.show()

survival_table = kmf.survival_function_.reset_index()
survival_table.columns = ["duration", "survival_probability"]

display(survival_table)

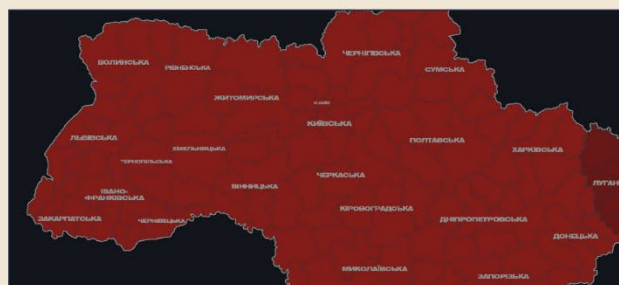
```

КЛАСТЕРИЗАЦІЯ РЕГІОНАЛЬНИХ ПРОФІЛІВ ПОВІТРЯНИХ ТРИВОГ ДЛЯ ВИЯВЛЕННЯ РЕЖИМІВ РИЗИКУ

02

Актуальність дослідження

Аналізувати не окремі агреговані показники, а **річні багатовимірні профілі регіонів**, які враховують інтенсивність, тривалість, добову структуру та інфраструктурний компонент тривоги.



Мета роботи:

Розробити підхід до кластеризації річних регіональних профілів повітряних тривог для виявлення режимів ризику в Україні у 2023–2025 роках, та проаналізувати міжрічні переходи регіонів між цими режимами та їхній просторовий розподіл.

Об’єкт дослідження:

Регіональні режими ризику в Україні в умовах повномасштабної війни.

Предмет дослідження:

Методи побудови, кластеризації та інтерпретації багатовимірних річних профілів регіонів.

Інструменти:

Python (pandas, tslearn, GeoPandas, lifelines, esda)

Ключові завдання:

- 1. Сформувати ознаки для опису профілів тривог.
- 2. Побудувати річні профілі «регіон-рік».
- 3. Порівняти алгоритми кластеризації.
- 4. Обрати фінальну конфігурацію за критерієм стабільності.
- 5. Інтерпретувати режими ризику та проаналізувати переходи.
- 6. Виконати просторову інтерпретацію результатів.

Методи:

Аналіз багатовимірних часових рядів, метод k-середніх для часових рядів, метод м’якого динамічного викривлення часу (SoftDTW), критерій стабільності Stadion, матриці міжрічних переходів, оцінка Каплана-Мейєра, кореляція Спірмена, глобальний індекс Морана, просторовий аналіз та візуалізація даних.

ДАНІ ТА ФОРМУВАННЯ ОЗНАК

Джерела даних:

- Alerts.in.ua – кількість, тривалість, час доби тривог
- ACLED – події ураження цивільної та критичної інфраструктури

Масштаб дослідження:

24	3	72	12×5
регіони	роки (2023–2025)	об’єкти «регіон-рік» (24 регіони × 3 роки)	розмір одного профілю (12 місяців × 5 ознак)

Фінальні ознаки для кластеризації:

інтенсивність	- alert_count_per_day - hours_per_alerts	Частота тривог – кількість тривог на день Середня тривалість однієї тривоги (год)
структура	- night_alerts_proportion - operational_hours_alert_share	Частка нічних тривог (22:00–06:00) Частка тривог у денні операційні години (10:00–21:00)
інфраструктура	infr_attacks_per_alert	Кількість інфраструктурних атак на одну тривогу

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

05

МЕТОДИ ДОСЛІДЖЕННЯ

Фінальний набір для кластеризації:

регіон-рік як багатовимірний часовий профіль

$$X \in R^{72 \times 12 \times 5}$$

72 = об'єкти «регіон-рік»;
12 = кількість місяців року;
5 = кількість ознак

Результат кластеризації:

$$f(X_{r,y}) = z_{r,y}, \quad z_{r,y} \in \{1, \dots, K\}$$

z – мітка кластера, що після інтерпретації розглядається як номер режиму ризику;
 K – кількість виділених режимів

Мета кластеризації:

згрупувати річні профілі за подібністю динаміки та інтерпретувати отримані групи як режими ризику.

Умови експерименту:

один $X_{\text{pool}} = 72 \times 12 \times 5$; однакова стандартизація; єдиний критерій Stadion; grid search + шумові збурення $\text{eps} = 0.02, 0.03, 0.04$.

Порівнювані підходи

01

Euclidean K-Means

базове порівняння профілів без часового вирівнювання

02

DTW-KMeans

врахування локальних часових зсувів у динаміці

03

SoftDTW-KMeans

м'яке вирівнювання часових профілів

04

Agglomerative

ієрархічне порівняння векторизованих профілів

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

06

КРИТЕРІЙ ВИБОРУ ФІНАЛЬНОЇ КОНФІГУРАЦІЇ

Критерій вибору: Stadion

• Чому потрібен критерій стабільності?

Оскільки еталонної розмітки даних немає, модель обиралась не за точністю, а за стійкістю розбиття до малих шумових збурень стандартизованих даних.

• Шумове збурення даних:

$$X^{(\varepsilon)} = X + \varepsilon\eta, \quad \eta \sim N(0, 1)$$

X – стандартизований масив річних профілів (72, 12, 5);
 ε – сила шумового збурення;
 η – випадковий шум зі стандартного нормального розподілу.

• У роботі перевірялись:

$$\varepsilon \in \{0.02, 0.03, 0.04\}$$

$$\text{Stadion} = \text{StabB} - \text{StabW}$$

StabB

наскільки стабільно відтворюється розбиття після збурень (between-cluster stability);

StabW

наскільки стабільно окремі кластери можуть бути додатково розбиті всередині (within-cluster stability);

• Інтерпретація:

більший Stadion = стійкіше та менш випадкове розбиття. Якщо після малих змін даних модель дає подібне розбиття, то така конфігурація вважається стабільнішою.

GRID SEARCH ТА ФІНАЛЬНА МОДЕЛЬ

Підсумкова таблиця найкращих конфігурацій (grid search):

Метод	Кількість кластерів	Stadion_mean	Stadion_std	StabB_mean	StabW_mean
SoftDTW-KMeans	5	0.079	0.015	0.957	0.878
Euclid-KMeans	4	0.058	0.036	1	0.942
DTW-KMeans	4	0.037	0.012	0.95	0.914
Agglomerative	4	-0.128	0.029	0.846	0.973

Фінальна модель:

SoftDTW-KMeans:
n_clusters = 5; n_init = 2; max_iter_barycenter = 30; gamma = 270.34

SoftDTW-KMeans має найбільше середнє значення Stadion, тому обрана як фінальна модель.

ВИЯВЛЕНІ РЕЖИМИ РИЗИКУ

Ризиковість режиму не зводиться до однієї ознаки: кластери відрізняються за інтенсивністю, часовою структурою та інфраструктурним компонентом.

- Кластер 0** – нічно зміщений режим із помірним інфраструктурним компонентом:
- нижча частота;
 - найвища частка нічних тривог;
 - помірний інфраструктурний ризик.

Cluster	Alert count per day	Avg. hours per alert	Night alert share	Operational-hours alert share	Infra. attacks per alert
0	0.948	1.287	0.534	0.366	0.12
1	1.326	1.159	0.338	0.522	0.052
2	4.808	1.755	0.264	0.587	0.549
3	4.969	1.09	0.27	0.577	0.135
4	0.808	1.33	0.49	0.405	0.022

Узагальнені середні профілі кластерів у вихідному масштабі

- Кластер 1** – базовий відносно помірний режим:
- невисока частота;
 - помірна тривалість;
 - низький інфраструктурний компонент.

- Кластер 2** – найвищий інтенсивно-інфраструктурний режим:
- висока частота тривог;
 - найбільша середня тривалість однієї тривоги;
 - найсильніший інфраструктурний компонент.

- Кластер 3** – інтенсивний режим частих коротших тривог:
- найвища частота тривог;
 - коротша середня тривалість;
 - вищий денний/операційний компонент.

- Кластер 4** – низькоінтенсивний структурно напружений режим:
- низька інтенсивність;
 - висока частка нічних та денних тривог;
 - найменший інфраструктурний компонент.

РОЗПОДІЛ КЛАСТЕРІВ ЗА РОКАМИ

Структура режимів ризику у 2023–2025 роках не залишалася сталою, а у 2025 році стала більш неоднорідною. Найбільшим є кластер 1 (33 об’єкти), що свідчить про наявність домінуючого річного режиму, який повторюється в даних найчастіше.

Cluster	2023	2024	2025
0	0	0	9
1	17	11	5
2	0	0	6
3	7	9	2
4	0	4	2

- **2023:** спостерігалися лише два режими – кластери 1 і 3.
- **2024:** до них додався кластер 4.
- **2025:** присутні всі п’ять режимів, включаючи кластери 0 та 2.



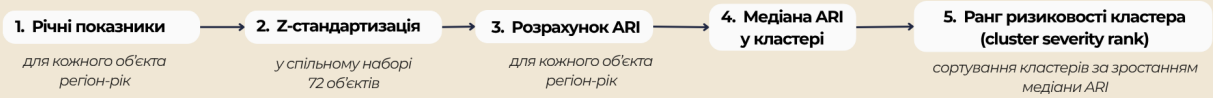
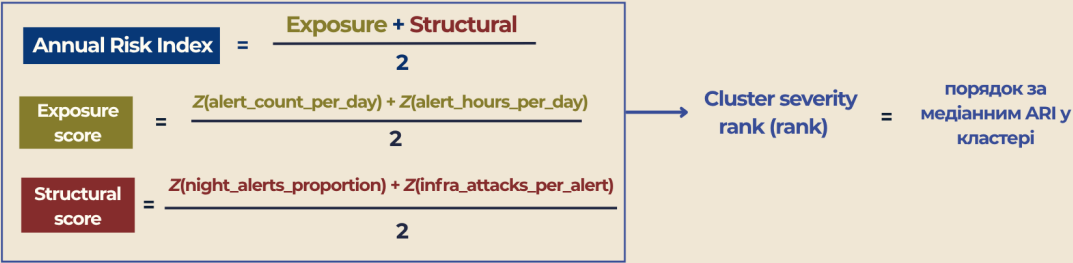
Розподіл кластерів за роками

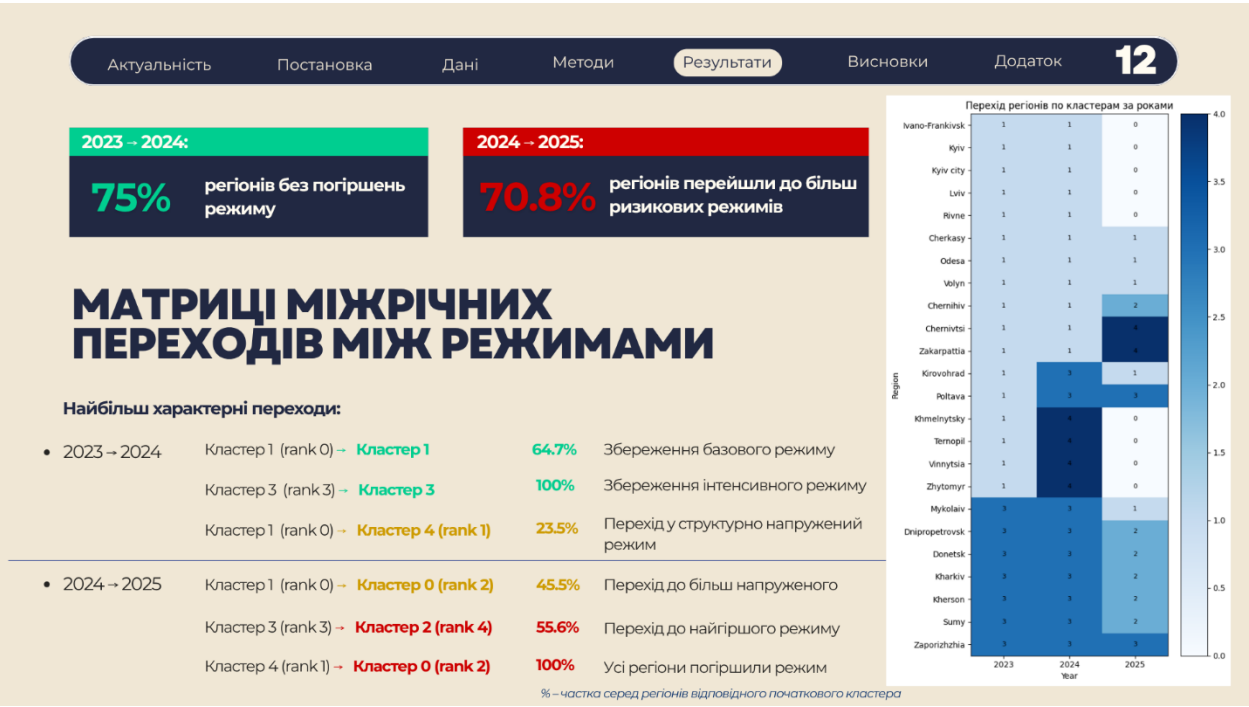
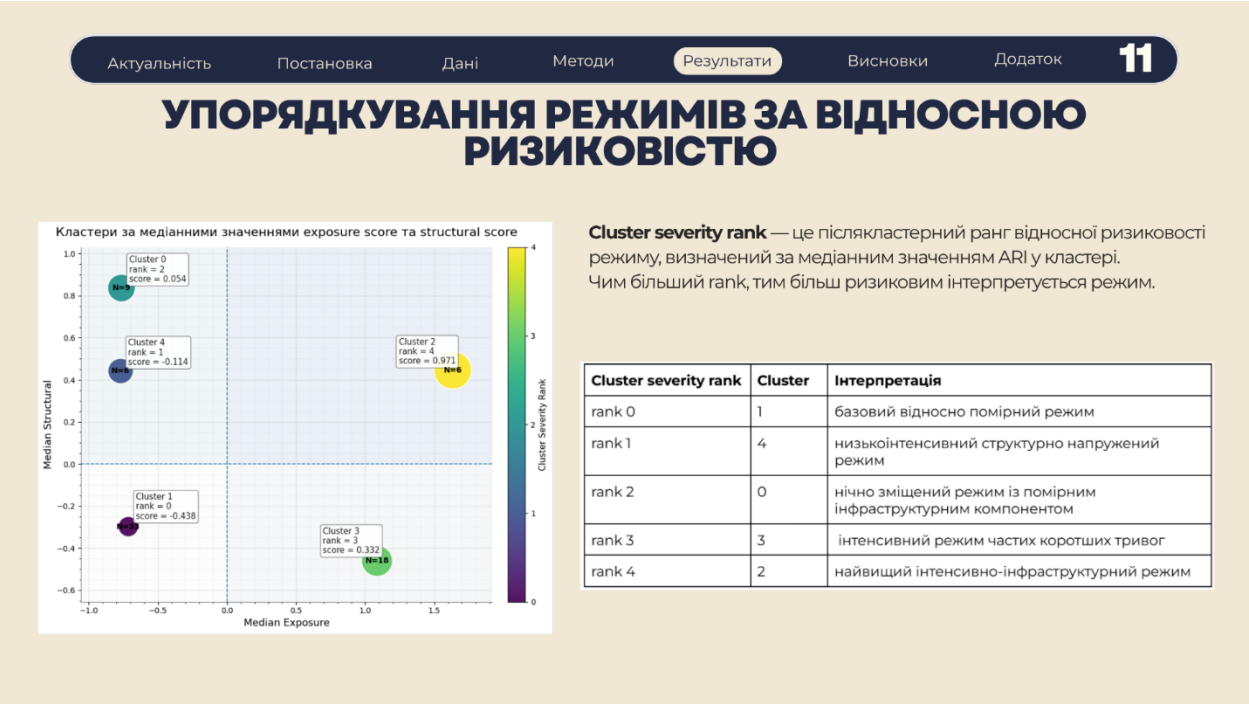
Розмір кластерів за всі роки

АВТОРСЬКИЙ ПОКАЗНИК: РІЧНИЙ ІНДЕКС РИЗИКУ

Річний індекс ризику (annual risk index, ARI) – інструмент інтерпретації та впорядкування режимів за відносною ризиковістю після кластеризації. Використаний для надання кластерам змістовного порядку. Індекс складається з двох блоків:

- **exposure score** – інтенсивність тривожного навантаження;
- **structural score** – структурна напруженість режиму.





Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

13

ЧАС ДО ПЕРШОГО ПОГІРШЕННЯ РЕЖИМУ

Метод:
оцінка Каплана-Мейєра

$$\hat{s}(t) = \prod_{t_j \leq t} \left(1 - \frac{d_j}{n_j}\right)$$

t_j – момент часу, коли відбулась;
 d_j – кількість подій у момент часу t_j ;
 n_j – кількість об'єктів у групі ризику настання події безпосередньо перед моментом t_j .

Подія:
перший перехід регіону до кластера з вищим рангом ризиковості.

Функція виживання $S(t)$:
частка регіонів, які до моменту t не мали погіршення режиму.

Важливо:
це не точковий прогноз для окремої області, а узагальнена оцінка динаміки переходів у групі регіонів за 2023–2025 роки.

Крива виживання Каплана-Мейєра: Час до погіршення режиму ризику (2023-2025)

Основні результати:

- 6 регіонів мали погіршення режиму після першого переходу;
- 13 – уперше погіршили режим на другому переході;
- $S(2)=0.21$ – приблизно 21% регіонів не мали першого погіршення до кінця періоду спостереження.

$S(0) = 1.00$

$S(1) = 0.75$

$S(2) = 0.21$

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

14

ПРОСТОРОВА ІНТЕРПРЕТАЦІЯ РЕЗУЛЬТАТІВ

Рис 1 – Просторовий розподіл показника *annual risk index*

Рис 2 – Просторовий розподіл показника *exposure score*

Рис 3 – Просторовий розподіл показника *structural score*

Рис 4 – Просторовий розподіл показника *cluster severity rank*

- cluster severity rank:** видно чіткі групи розподілу рівнів у 2023 та 2024 роках, та більш диференційоване поле у 2025 році;
- annual risk index:** нижчі значення спостерігалися переважно у західних та частині центральних областей, тоді як вищі були зосереджені у східних і південно-східних регіонах
- 2025: розподіл став більш неоднорідним

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

15

Коефіцієнти рангової кореляції Спірмена

для перевірки монотонного зв'язку між значеннями інтегрального індексу ризику (annual risk index) та географічними координатами центрів регіонів

year	rho_lon	p_lon	rho_lat	p_lat
2023	0.897	2.8e-09	-0.45	0.028
2024	0.937	1.7e-11	-0.183	0.393
2025	0.764	1.4e-05	-0.07	0.747

Результати:

- 2023–2024: $\rho_{lon} > 0$ і $p\text{-value} < 0.05$ – статистично значущий градієнт Захід–Схід, але дещо слабший у 2025;
- зв'язок із широтою статистично значущий лише у 2023: $\rho_{lat} = -0.45$, $p = 0.028$;

Глобальний індекс Морана (Global Moran's I)

для перевірки наявності просторової автокореляції – чи регіони з подібним рівнем ризику мають тенденцію зосереджуватися поруч

year	annual_risk_index	exposure_score	structural_score
2023	0.757	0.813	0.202
2024	0.685	0.833	0.118
2025	0.495	0.775	0.064

Результати:

- статистично значуща позитивна просторова автокореляція для exposure score та annual risk index ($p_{sim} = 0.001$);
- слабка для structural score й статистично не підтверджена ($p_{sim} > 0.05$);
- слабший зв'язок у 2025 році

ПРОСТОРОВА ІНТЕРПРЕТАЦІЯ РЕЗУЛЬТАТІВ

Ключові результати:

- Річний ризик відрізняється передусім між західними та східними регіонами, а не між південними та північними.
- Найвищі значення кореляції у 2023–2024 роках, із послабленням у 2025 році.
- Global Moran's I підтвердив, що регіони з подібним рівнем ризику мають тенденцію зосереджуватися поруч.
- structural score: просторові відмінності були менш вираженими і не формували таких чітких груп сусідніх регіонів.
- cluster severity rank: 2023–2024 – виразні групи розподілу рівнів, а 2025 – більш диференційоване поле.

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

16

ВИСНОВКИ: ОСНОВНІ РЕЗУЛЬТАТИ

1. Сформовано 72 річні профілі регіон-рік розмірності 12 × 5.

2. За критерієм Stadion обрано фінальну модель SoftDTW-KMeans, K=5.

3. Виділено та інтерпретовано п'ять режимів регіонального ризику, що відрізняються за інтенсивністю, добовою структурою та інфраструктурним компонентом.

4. Побудовано річний індекс ризику та ранги відносної ризиковості кластерів для впорядкування режимів.

5. Проаналізовано міжрічні переходи, час до першого погіршення та просторову структуру режимів ризику.

6. Виявлено зростання неоднорідності у 2025 році.

7. Виявлено посилення переходів до більш ризикових режимів у 2024 → 2025.

8. Підтверджено наявність Західно-Східного градієнта ризику та позитивну просторову автокореляцію для річного індексу ризику.

ВИСНОВКИ: НОВИЗНА РОБОТИ

На відміну від окремого підрахунку кількості або тривалості повітряних тривог, запропонований підхід дозволяє виявляти типові режими ризику, оцінювати їхню відносну ризиковість, міжрічну динаміку та просторову структуру.

01

1. Формалізовано об'єкт **регіон-рік** як багатовимірний часовий профіль 12×5 – кожен регіон у кожному році як окремий стан системи ризику.

02

2. Побудовано систему ознак, що поєднує **інтенсивність, добову структуру та інфраструктурний компонент тривоги**.

03

3. Адаптовано підхід вибору фінальної моделі за критерієм стабільності: **grid search + Stadion + шумові збурення**. Обраний критерій стабільності дозволяє оцінити не лише те, чи зберігається кластерне розбиття після малих збурень даних, а й чи не містять отримані кластери стабільної прихованої підструктури. Якщо після невеликого шуму регіони залишаються в подібних групах, це означає, що кластеризація не є випадковою реакцією на дрібні коливання даних.

04

4. Запропоновано післякластерне впорядкування за авторським **річним індексом ризику** та **cluster severity rank**, що дозволило аналізувати не лише належність регіону до групи, а й погіршення або покращення його режиму.

05

5. Формалізовано **аналіз динаміки**: матриці переходів + час до першого погіршення режиму. Це дозволило перейти від порівняння окремих річних карт до аналізу траєкторій регіонів між режимами ризику. Матриці переходів показують напрям рівня ризику в часі, а оцінка Каплана-Мейєра узагальнює час до першого погіршення.

06

6. Виконано **просторову перевірку результатів** через Spearman + Moran's I. Це дозволило оцінити не лише розподіл режимів, а й закономірність їх просторової концентрації.

ПРАКТИЧНА ЦІННІСТЬ та НАПРЯМИ ПОДАЛЬШОГО РОЗВИТКУ

Результати роботи можуть бути використані для:

- 01 – аналізу регіональних відмінностей воєнного ризику;
- 02 – моніторингу змін безпекової ситуації в регіонах;
- 03 – виявлення переходів до більш напружених режимів;
- 04 – порівняння регіонів не лише за рівнем тривоги, а й за типом ризикового профілю;
- 05 – підтримки досліджень регіональної стійкості, планування та відновлення.

Напрями подальшого розвитку:

- розширення часового періоду;
- додавання нових ознак вразливості;
- перевірка стійкості результатів на нових даних.
- доповнення економічними, демографічними та інфраструктурними показниками.

ПУБЛІКАЦІЇ ЗА ТЕМОЮ РОБОТИ

1) стаття:

Кабанова М. І., Мельничук В. Е., Семенченко Н. І. Воєнні ризики та регіональна економічна стійкість України: повітряні тривоги, атаки на інфраструктуру та релокація бізнесу. Сталий розвиток економіки. 2025. № 6(57). С. 155–163. DOI: 10.32782/2308-1988/2025-57-21.

2) конференція:

Кабанова М. І., Кузнєцова Н. В. Кластеризація регіонів України за динамікою повітряних тривог методом k-середніх для часових рядів (DTW/Soft-DTW) у контексті релокацій бізнесу. Системні науки та інформатика : збірка доповідей IV науково-практичної конференції, Київ, 1–5 грудня 2025 р. Київ : НН ІПСА КПІ ім. Ігоря Сікорського, 2025. С. 109–116.

3) конференція:

Kabanova M., Melnychuk V. Air-raid alert dynamics in Ukraine: a data-driven typology for wartime urban management. Моделювання та прогнозування економічних процесів : збірник тез доповідей XIX Міжнар. наук.-практ. конф., м. Київ, 4 грудня 2025 р. Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2025. С. 256–258.

4) у друці:

Kabanova M., Kuznietsova N., Seebauer M. Regional Air-Raid Alert Risk Regimes in Ukraine: Management Implications under Uncertainty. Economic Transformations in the Global World : Proceedings of the International Scientific Conference, Varna Free University "Chernorizets Hrabar", Varna, Bulgaria, 12–13 June 2026. 10 p.

Навчально-науковий інститут прикладного системного аналізу (НН ІПСА),
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» (НТУУ КПІ)

ДЯКУЮ ЗА УВАГУ!



ДОДАТОК: ФОРМУЛИ ТА ПОЯСНЕННЯ

МОЖЛИВІ ПИТАННЯ

Чому не було взято 2022 рік?

2022 рік був початковим етапом повномасштабного вторгнення, коли умови були різко нестабільними, а дані та поведінка регіонів могли бути менш порівнюваними з наступними роками.

Чому не включено Крим, Севастополь і Луганську область?

Через тимчасову окупацію та неповноту доступних даних для цих територій, їх включення могло б спотворити результати кластеризації, тому вони були виключені для коректного порівняння між регіонами.

Чому обрано частку нічних тривог та часовий інтервал 22:00–06:00?

Нічні тривоги відображають не лише інтенсивність загрози, а й структурну напруженість режиму, бо вони припадають на період відпочинку і мають інший вплив на населення та функціонування регіону. Тому це важлива структурна ознака, яка доповнює показники інтенсивності та тривалості.

Інтервал 22:00–06:00 обрано згідно зі статтею 54 Кодексу законів про працю України, де нічним вважається час з 10 години вечора до 6 години ранку.

Чому обрано операційні години 10:00–21:00?

Цей показник трактувався як ознака операційного навантаження, тобто часу, коли активні торгівля, послуги, мобільність населення та частина економічної діяльності.

Також цей інтервал узгоджується з підходом, використаним в аналітиці Української ради торгових центрів щодо впливу повітряних тривог на роботу ТЦ.

Як інтерпретувати невелике значення Stadion?

Це відносний критерій для порівняння конфігурацій. У нашому випадку всі методи працюють із складними реальними даними без еталонної розмітки, тому ідеально високого значення очікувати не варто. У роботі SoftDTW-KMeans мав найвище середнє значення Stadion серед перевірених моделей і дав змістовно інтерпретовані режими ризику, тому був обраний як фінальна конфігурація.

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

23

Які кластери та відповідні ранги ризикованості для регіонів за 2023–2025 роки?

Регіон	Кластер 2023	Кластер 2024	Кластер 2025	Ранг 2023	Ранг 2024	Ранг 2025	Напрямок зміни ризиковості з 2023 до 2024 року	Напрямок зміни ризиковості з 2024 до 2025 року
Cherkasy	1	1	1	0	0	0	без змін	без змін
Chernihiv	1	1	2	0	0	4	без змін	підвищення
Chernivtsi	1	1	4	0	0	1	без змін	підвищення
Dnipropetrovsk	3	3	2	3	3	4	без змін	підвищення
Donetsk	3	3	2	3	3	4	без змін	підвищення
Ivano-Frankivsk	1	1	0	0	0	2	без змін	підвищення
Kharkiv	3	3	2	3	3	4	без змін	підвищення
Kherson	3	3	2	3	3	4	без змін	підвищення
Khmelnytsky	1	4	0	0	1	2	підвищення	підвищення
Kirovohrad	1	3	1	0	3	0	підвищення	зниження
Kyiv	1	1	0	0	0	2	без змін	підвищення
Kyiv city	1	1	0	0	0	2	без змін	підвищення
Lviv	1	1	0	0	0	2	без змін	підвищення
Mykolaiv	3	3	1	3	3	0	без змін	зниження
Odesa	1	1	1	0	0	0	без змін	без змін
Poltava	1	3	3	0	3	3	підвищення	без змін
Rivne	1	1	0	0	0	2	без змін	підвищення
Sumy	3	3	2	3	3	4	без змін	підвищення
Ternopil	1	4	0	0	1	2	підвищення	підвищення
Vinnitsia	1	4	0	0	1	2	підвищення	підвищення
Volyn	1	1	1	0	0	0	без змін	без змін
Zakarpattia	1	1	4	0	0	1	без змін	підвищення
Zaporizhzhia	3	3	3	3	3	3	без змін	без змін
Zhytomyr	1	4	0	0	1	2	підвищення	підвищення

Актуальність

Постановка

Дані

Методи

Результати

Висновки

Додаток

24

МОЖЛИВІ ПИТАННЯ

Чому DTW / Soft-DTW, а не Euclidean K-Means?

Евклідова відстань (Euclidean K-Means)

- порівнює значення в однакових місяцях;
- є чутливою до зсувів піків у часі, наприклад, два профілі можуть бути подібними за загальною формою, але отримати велику відстань лише через те, що пік припав на сусідні місяці;
- використаний як базовий контрольний підхід.

$$d_E = \sqrt{\sum_{k=1}^P (x_{ik} - v_{jk})^2},$$

x_{ik} та v_{jk} – задані P -розмірні вектори.

Метод динамічного викривлення часу (DTW-KMeans)

- дозволяє локальні стискання або розтягування часових профілів;
- корисний, якщо схожі піки виникають у сусідні місяці.

$$DTW_q(A, B) = \min_{\pi \in P} \left(\sum_{(i,j) \in \pi} d(a_i, b_j)^q \right)^{\frac{1}{q}},$$

P – множина допустимих шляхів вирівнювання;
 $d(a_i, b_j)$ – відстань між точками a_i з ряду A та b_j з ряду B , та сума береться по всіх точках (i, j) в оптимальному шляху вирівнювання з P .

Метод м'якого динамічного викривлення часу (SoftDTW-KMeans)

- згладжена версія DTW, регульована параметром γ ;
- менш "жорстко" обирає один шлях вирівнювання;
- за малих значень γ міра подібності наближається до класичного DTW, тоді як за більших значень вирівнювання стає м'якшим і менш чутливим до локальних відмінностей між часовими профілями.

$$\text{softDTW}_\gamma(A, B) = \min_{\pi \in P} \sum_{(i,j) \in \pi} d(a_i, b_j)^2,$$

де $\min_\gamma(a_i, \dots, a_n) = -\gamma \log \sum_i e^{-a_i/\gamma}$ γ – коефіцієнт згладжування $\gamma > 0$

Що таке критерій Stadion?

Ідея: хороша кластеризація має:

- зберігати подібне розбиття після малих допустимих збурень даних;
- не містити всередині кластерів стабільної прихованої підструктури.

У цьому підході поєднуються **дві складові**:

- міжкластерна стійкість (between-cluster stability, $StabB$) – чи відтворюється базове розбиття після малих збурень;
- внутрішньо кластерна стійкість (within-cluster stability, $StabW$) – чи можна стабільно розділити вже знайдені кластери на підкластери.

Добре відповідає задачі аналізу річних профілів повітряних тривог, де можливі дрібні коливання значень ознак, але календарний порядок місяців має зберігатися.

Як оцінювалася стабільність у роботі:

1. На стандартизованому масиві $72 \times 12 \times 5$ будувалося базове розбиття.
2. Формувалася серія збурених копій цього масиву шляхом додавання малого адитивного шуму до значень ознак: 0.02, 0.03, 0.04.
3. Кластеризація повторювалася для збурених даних з тією самою конфігурацією.
4. Отримані розбиття порівнювалися з базовим та підсумкове значення Stadion усереднювалося за кількома рівнями збурення.

Параметри протоколу перевірки стабільності:

- $eps = \{0.02, 0.03, 0.04\}$ – рівень малого шумового збурення;
- $D_between = 12$ – кількість повторень для оцінки $StabB$;
- $D_within = 6$ – кількість повторень для оцінки $StabW$;
- $Omega = \{2, 3\}$ – перевірка можливого поділу кластера на 2 або 3 підкластери під час обчислення $StabW$;
- $base_seed = 42$ – фіксація випадковості для відтворюваності;
- $min_cluster_size = 2$.

$$Stadion(C_K, \Omega) = Stab_B(C_K) - Stab_W(C_K, \Omega),$$

$$Stab_B(A, X, C_K, K) = \frac{1}{D} \sum_{d=1}^D s(C_K, A(X_d, K)),$$

$$Stab_W(A, X, C_K, K, \Omega) = \sum_{k=1}^K \left(\frac{1}{|\Omega|} \sum_{K' \in \Omega} Stab_B(A, C_K, Q_K^{(k)}, K') \right) \times \frac{N_k}{N}$$

- Для кожної кандидатної конфігурації спочатку будувалося базове розбиття $C_K = A(X, K)$ на вихідному стандартизованому масиві X .
- $StabB$ оцінює, наскільки це базове розбиття відтворюється після малих збурень даних. Для цього формувалися збурені копії X_d , кластеризація повторювалася з тією самою конфігурацією, а отримані розбиття порівнювалися з C_K за мірою подібності s . У реалізації s – це Adjusted Rand Index, бо він порівнює структуру розбиттів, не залежить від номерів кластерів і враховує випадковий збіг.
- $StabW$ перевіряє, чи немає всередині знайдених кластерів стабільної прихованої підструктури. Для кожного кластера $C_K \setminus \{k\}$ перевірялося його можливе розбиття на K' підкластерів, де $K' \in \Omega$. У роботі поділ перевірявся на 2 або 3 підкластери.
- При цьому, показник $StabB$ має бути максимізований, тоді як $StabW$ – мінімізований.

МОЖЛИВІ ПИТАННЯ

Що таке рангова кореляція Спірмена?

- для визначення сили та напрямку зв'язку між ARI та координатами центроїдів на основі їх рангів
- значення в межах від -1 до +1, де близьке до нуля, вказує на відсутність вираженого монотонного зв'язку.
- значення кореляції з довготою (ρ_{lon}) > 0 – означає зростання ризику на схід; (ρ_{lon}) < 0 – на захід;
- значення кореляції з широтою (ρ_{lat}) > 0 – свідчить про тенденцію до зростання ризику в північному напрямку; (ρ_{lat}) < 0 – у південнішому;
- статистична значущість перевіряється через p-значення (p-value) < 0.05 .

$$p = 1 - \frac{6 \sum d^2}{n(n^2 - 1)},$$

d – різниця між рангами відповідних значень;
 n – загальна кількість спостережень.

Що таке глобальний індекс Морана?

- перевіряє, чи сусідні регіони мають схожі значення досліджуваного показника;
- додатні та статистично значущі значення Moran's I свідчать про просторове групування подібних значень, від'ємні – про просторове чергування, а значення, близькі до нуля, вказують на відсутність вираженої просторової структури;

z_i – це відхилення значення атрибута для об'єкта i від його середнього значення $(x_i - \bar{x})$;
 w_{ij} – просторова вага між об'єктами i та j ;
 n – загальна кількість об'єктів;
 S_0 – сума всіх просторових ваг.

$$I = \frac{n}{S_0} \frac{\sum_{i=1}^n \sum_{j=1}^n w_{ij} z_i z_j}{\sum_{i=1}^n z_i^2},$$

$$S_0 = \sum_{i=1}^n \sum_{j=1}^n w_{ij},$$