КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет комп'ютерних наук та кібернетики

Кафедра прикладної статистики

Кваліфікаційна робота на здобуття ступеня бакалавра

на тему:

СЕГМЕНТАЦІЯ РИНКУ

МЕТОДАМИ КЛАСТЕРНОГО АНАЛІЗУ

Виконала студентка 4-го курсу

Шарапа Вікторія Богданівна _____

(підпис)

Науковий керівник:

доцент, кандидат фізико-математичних наук

Лівінська Ганна Володимирівна _____

(підпис)

Засвідчую, що в цій роботі немає запозичень з праць інших авторів без відповідних посилань

Студент _____

(підпис)

Роботу розглянуто й допущено до захисту на засіданні кафедри прикладної статистики

«06» червня 2022 р., протокол № 11

Завідувач кафедри

Розора І. В. _____

(підпис)

Київ – 2022

РЕФЕРАТ

Обсяг роботи: 61 сторінка, 45 ілюстрацій, 6 джерел, 1 додаток.

Ключові слова: ДАНІ, КЛАСТЕР, КЛАСТЕРИЗАЦІЯ, МЕТОД, РИНОК, СЕГМЕНТАЦІЯ.

Об’єктом дослідження у даній роботі є методи кластерного аналізу, а предметом – робота методів кластерного аналізу для сегментації ринку.

Метою роботи є сегментація ринку із використанням деяких основних методів кластерного аналізу. В якості середовища програмування було обрано Jupyter Notebook та мову програмування Python 3.9.0. Для роботи зі структурами даних було обрано бібліотеку Pandas, для кластеризації була обрана бібліотека Scikit-learn, а для візуалізації результатів були обрані бібліотеки Matplotlib та Seaborn.

Були отримані наступні результати: виконана кластеризація маркетингових даних на базі набору даних про купівельну поведінку та набору даних про покупки із різних каналів продажу. Проведено порівняльний аналіз методів, що показав ефективність лише окремих методів щодо заданих наборів даних.

Дана робота може застосовуватись для ознайомлення з темою сегментації ринку за допомогою кластеризації. Також код написаної програми може використовуватися і для обробки й кластеризації інших маркетингових датасетів.

ЗМІСТ

ВСТУП

Одна з найосновніших базових здібностей живих істот – це групування схожих об’єктів для отримання їх класифікації. Ідея сортування схожих речей за категоріями явно примітивна, оскільки, наприклад, древня людина повинна була розуміти, що безліч окремих об’єктів навколо неї наділені певними властивостями, такими як отруйність, їстівність, небезпечність та ін. Тому впевнено можна сказати, що людський мозок почав кластеризувати для себе навколишні об’єкти ще з давніх-давен.

Кластеризація (кластерний аналіз) — це групування об’єктів на основі подібності. Такий аналіз широко застосовується в інформаційних системах для пошуку закономірностей в даних.

Головною задачею кластерного аналізу є виділення необхідної кількості об’єктів схожих між собою всередині групи і таких, що максимально відрізняються від екземплярів інших кластерів. Такий аналіз широко застосовується в інформаційних системах для пошуку закономірностей в даних.

У контексті сегментації ринку кластерний аналіз — це використання математичних моделей для виявлення груп подібних клієнтів на основі пошуку найменших відмінностей серед клієнтів у кожній групі. Ці однорідні групи відомі як «архетипи клієнтів» або «персони».

Метою кластерного аналізу в маркетингу є сегментація клієнтів для досягнення більш ефективного маркетингу за допомогою персоналізації клієнтів. В результаті кластери допомагають покращити прогнозну аналітику щодо клієнтів, а також використовуються для націлювання на конкретні групи клієнтів із пропозиціями персоналізованими відповідно до їхніх побажань, потреб і переваг.

Оцінка сучасного стану об’єкта дослідження. Незважаючи на широкий спектр доступних методів групування індивідів у сегменти ринку на основі інформації багатоваріантного опитування, кластеризація залишається найпопулярнішим і найбільш широко застосовуваним методом. Сегментація ринку є однією з найбільш фундаментальних концепцій стратегічного

маркетингу. Чим краще сформовані сегменти клієнтів , тим більш успішною буде компанія на ринку. Тому якість сформованих кластерів, які досліджуються, мають вирішальне значення для успіху організації та вимагають професійного використання методів для визначення потенційно корисних кластерів клієнтів для компанії.

Актуальність роботи та підстави для її виконання. Компанії завжди хочуть знати більше про своїх клієнтів, а саме хто та чому робить покупки в їхніх магазинах, яких продуктів та послуг шукають їхні клієнти. Знання того, як інтерпретувати зібрані про клієнтів дані, може допомогти компанії вносити необхідні зміни та приймати рішення. Сегментація ринку за допомогою методів кластерного аналізу дозволяє компаніям розгортати цільові маркетингові кампанії, які пропонують найбільш привабливі продукти для кожного споживчого сегмента, що, як наслідок, сприяє збільшенню прибутків компанії та підвищенню її конкурентоспроможності на ринку.

Метою даної роботи є дослідження та використання методів кластерного аналізу для сегментації ринку.

РОЗДІЛ 1 ЗАДАЧА КЛАСТЕРИЗАЦІЇ ДАНИХ

1.1 Поняття кластеризації даних

Нині практично в усіх областях людської діяльності існує потреба у вивченні статистичних даних, що описують поведінку об’єктів, подій та явищ за якими ведеться спостереження. Однією з найбільш актуальних і практично затребуваних задач аналізу даних є задача розбиття об’єктів на відносно однорідні групи (підмножини), що називаються кластерами, а саме розбиття називається кластеризацією.

Кластер – група однорідних елементів, що характеризуються якоюсь загальною ознакою. Однорідність кластерів означає, що об’єкти, що належать одному кластеру, повинні бути схожими (близькими) відносно обраної метрики. Об’єкти з різних кластерів повинні суттєво відрізнятися. Задача розбиття даних на кластери називається задачею кластеризації даних. Також її прийнято називати таксономією, автоматичною класифікацією, групуванням об’єктів або задачею навчання без вчителя.

Застосування кластерного аналізу передбачає наступні етапи:

1) Формування вибірки для кластеризації;
2) Визначення множини характеристик, по яким будуть оцінюватися об’єкти у вибірці. За необхідності – нормування (стандартизація) значень змінних;
3) Вибір міри схожості між об’єктами;
4) Застосування методу кластерного аналізу для створення груп схожих об’єктів;
5) Представлення результатів кластеризації.

Після отримання і аналізу результатів можливе корегування обраної метрики і методу кластеризації до отримання оптимального результату.

Цілі кластеризації даних:

- Розбиття вибірки на групи схожих об’єктів для полегшення розуміння кластерної структури, що спрощує обробку даних і

прийняття рішень із застосуванням свого методу аналізу до кожного кластера;

- Скорочення об’єму даних, після чого залишається по одному або декілька типових представників від кожного класу. В таких задачах найважливіше забезпечити високий ступінь збіжності об’єктів всередині кожного кластеру, а кластерів може бути скільки завгодно.
- Виділення нетипових об’єктів, аномалій та викидів, для визначення новизни кластерів або їх кількості. Найбільший інтерес викликають окремі об’єкти, які не вписуються ні до одного кластеру.

Задача кластеризації відноситься до статистичної обробки, а також до широкого класу задач навчання без вчителя. Безсумнівною перевагою кластерного аналізу є те, що він дозволяє проводити розбиття об’єктів не по одному параметру, а по цілому набору ознак. Окрім того, кластерний аналіз на відміну від більшості математично-статистичних методів не накладає ніяких обмежень на вид об’єктів, що розглядаються, і дозволяє розглядати множину вихідних даних будь-якої природи.

1.2 Формальні означення

Введемо означення тих понять, якими будемо оперувати:

Об’єкт – елементарна група даних, з якою працюють алгоритми кластеризації. Для кожного об’єкта визначаються параметри, які описують його і які об’єднуються у вектор характеристик $x = (x_1, \dots, x_m)$, де m – розмірність простору характеристик, а компонента x_i є окремою характеристикою об’єкта (кількісною чи якісною).

Міру схожості двох об’єктів $d(x_i, x_j)$ обрахованою бо заданій метриці будемо називати *відстанню* між об’єктами, де x_i, x_j – елементи множини.

Кластер – підмножина схожих один на одного об’єктів.

Кластеризація – розподіл множини вхідних векторів на групи (кластери) по ступеню «подібності» один на одного.

Розбиття – сукупність класів, що є не пустими і не перетинаються; одна із найпопулярніших кластерних структур, що досить часто застосовується при аналізі даних про схожість між об'єктами.

1.3 Постановка задачі кластеризації

Нехай X – множина об'єктів, а $Y \subset \mathbb{N}$ - множина ідентифікаторів (міток) кластерів. На множині X задана функція відстані між об'єктами $d(x_i, x_j)$. Задана скінченна вибірка об'єктів $X^n = \{x_1, \ldots, x_n\} \subset X$. Необхідно розбити вибірку на підмножини (кластери), що не перетинаються, тобто кожному об'єкту $x_i \in X^n$ поставити у відповідність мітку (номер кластера) $y_i \in Y$ таким чином, щоб об'єкти всередині кожного кластера були близькими відносно метрики $d(x_i, x_j)$, а об'єкти з різних кластерів суттєво відрізнялися.

Алгоритм кластеризації $a: X \to Y$ – це функція, яка будь-якому об'єкту $x \in X$ ставить у відповідність номер кластера $y \in Y$. Множина Y в деяких випадках відома наперед, але частіше ставиться задача визначити оптимальну кількість кластерів, з точки зору того чи іншого критерія кластеризації.

1.4 Вхідні дані

В задачах кластерного аналізу вхідні дані зазвичай представлені у вигляді таблиці (матриці X), рядки якої представляють результати вимірювання одної з m ознак для кожного з n досліджуваних об'єктів (кількість стовпчиків).

В конкретних випадках може представляти інтерес як групування об'єктів, так і групування ознак.

Значення ознак (елементи матриці X) можуть бути різних типів: кількісні, якісні та рангові (ординальні, порядкові, використання яких в арифметичних операціях зазвичай є некоректним).

В задачах кластеризації ознаки не завжди є кількісними.

Часто в якості одної з характеристик об'єкту є наявність чи відсутність певної властивості (особливості). Якісні дані зазвичай невпорядковані за виключенням так званих бінарних змінних, які зазвичай позначаються числами

«0» та «1» та які часом можна вважати впорядкованими. Проте, в більшості випадків вважати їх кількісними змінними не можна.

Звичайно, наявність різних типів даних робить аналіз складнішим, зокрема тому, що в кластерному аналізі важливим є спосіб вимірювання «відстані» між об’єктами як міри подібності цих елементів, і тип цієї міри, що використовується, залежить від типу спостережуваних даних. Якщо наявні дані різних типів, їх намагаються звести до одного типу.

Окрім однотипності даних, бажано, щоб всі дані в таблиці були виміряні в одній шкалі.

Для зведення даних до одного масштабу проводиться так звана стандартизація (нормалізація) змінних, наприклад, діленням кожного зі значень ознак на максимальне значення цієї ознаки у вибірці.

1.5 Виділення вектора характеристик

Для початку необхідно обрати характеристики об’єктів, які будуть використані в процесі кластеризації. Ними можуть бути як кількісні характеристики так і якісні характеристики.

Найчастіше працюють із кількісними характеристиками, так як для них можна застосовувати більшу кількість метрик.

Коли простір характеристик є досить великим, процес кластеризації проходить досить повільно, і його результати не завжди прийнятні. Тому при великій розмірності простору характеристик, потрібно постаратися зменшити цю розмірність, залишивши найбільш важливі характеристики об’єктів.

Отриманий набір характеристик кожного об’єкту необхідно стандартизувати (нормувати), для кращих результатів. Стандартизувати вектор означає привести його до фіксованого розміру. Характеристики нормованого вектора будуть лежати всередині фіксованого відрізка, наприклад, це можуть бути такі відрізки: [0, 1] або [-1 , 1] в залежності від задачі, яку ми розглядаємо. Сама ж стандартизація не є обов’язковою.

РОЗДІЛ 2 МЕТОДИ КЛАСТЕРНОГО АНАЛІЗУ

2.1 Порівняльний аналіз ієрархічних та неієрархічних методів

Методів кластерного аналізу існує досить багато. Всі їх можна розділити на ієрархічні та неієрархічні методи.

1) Ієрархічні методи. Ці методи є ефективними за невеликої кількості спостережень, а ієрархічна кластеризація – це послідовність розбиттів, в якій кожне розбиття вкладається в наступне розбиття в послідовності. Ієрархічні методи поділяються на агломеративні (ті, що об’єднують) та дивізивні (ті, що розділяють).
2) Неієрархічні (ітеративні) методи (зокрема, метод k-середніх). Застосовуються при великій кількості спостережень. Кластери формуються виходячи з умов розбиття, що задаються, і які можуть бути змінені користувачем для досягнення бажаної якості. Ці методи можуть призвести до утворення кластерів, що перетинаються, коли один об’єкт може належати одночасно кільком кластерам. Неієрархічні методи є більш стійкими відносно шумів та викидів, некоректного вибору метрики, включенню незначущих змінних в набір даних, що бере участь в кластеризації. Проте є й мінуси. Аналітик повинен наперед визначити кількість кластерів, кількість ітерацій або правило зупинки, а також деякі інші параметри кластеризації.

Якщо немає ніяких припущень щодо кількості кластерів, рекомендують використовувати ієрархічні алгоритми кластерного аналізу. Однак, якщо об’єм вибірки не дозволяє цього зробити, то можна провести ряд експериментів з різною кількістю кластерів, наприклад, почати розбиття сукупності даних з двох груп і, поступово збільшуючи їхню кількість, порівнювати результати. За рахунок такої «варіації» результатів досягається досить велика гнучкість кластеризації.

Ієрархічні методи, на відміну від неієрархічних, не визначають кількість кластерів, а будують повне дерево вкладених кластерів. Труднощі, що можуть

виникнути при використанні цих методів – обмеженість щодо об’єму набору даних, вибір міри близькості, негнучкість отриманих класифікацій. Перевагою цих методів є їхня наочність і можливість отримати детальне уявлення про структуру даних.

2.2 Методи ієрархічної кластеризації

З усіх методів кластерного аналізу найбільш вживаними та поширеними є ієрархічні методи, так як вони є найбільш універсальними та точними. До того ж особливо добре працюють тоді, коли за своєю природою дані мають ієрархічну структуру.

До недоліків ієрархічних методів варто віднести громіздкість їх обчислювальної реалізації. На кожному кроці методи потребують обрахування матриці відстаней, а отже, великих об’ємів пам’яті (порядок просторової складності – $O(n^2)$, де n – кількість об’єктів) і великої кількості часу (порядок просторової складності $-\ O(n^3)$).

Етапи агломеративного кластерного аналізу:

1) На підготовчому етапі за потреби проводиться стандартизація (нормування) даних.
2) До початку процедури агломерації необхідно визначити як буде вимірюватися міра схожості об’єктів («відстань»). На основі відстаней між елементами будуються відстані між кластерами.
3) Вибір методу кластеризації.
4) Кожен елемент утворює спочатку свій окремий кластер.
5) На першому кроці аналізу два «найближчі» кластери об’єднуються в один.
6) Процес аналізу триває доти, доки не залишиться лише один кластер.
7) Приймається рішення про кількість кластерів.
8) Оцінка якості кластеризації.

Якщо базова матриця подібності має розмірність $n \times n$ (тобто наявні n об'єктів), то повністю процес кластеризації завершується за $n - 1$ кроків. В результаті всі об'єкти будуть об'єднані в один кластер.

На етапі, коли кожен об'єкт представляє собою окремий кластер, необхідно порівнювати групи елементів по ступеню подібності. Для порівняння груп елементів також використовується функція «відстані», яка визначає назву методу кластеризації.

Під методом агломеративної ієрархічної кластеризації розуміють спосіб обчислення відстаней між кластерами. Кластерний аналіз пропонує широкий вибір таких методів. Наведемо лише декілька з них:

1) Метод дальнього сусіда. Подібність двох кластерів визначається як максимум відстані між двома точками, що належать цим двом кластерам.

$$d(C_1,\ C_2) = \max_{i,j:x_i \in C_1, x_j \in C_2} d(x_i, x_j).$$

Його перевагою є те, що цей метод добре працює для розділення кластерів, коли є шум між кластерами.

Недоліки методу: метод тяжіє до утворення компактних сферичних кластерів подібного розміру, проте не враховує структуру кластера, має тенденцію розділяти великі кластери, є чутливим до викидів.

2) Метод Варда. Відстань у методі Варда обчислюється як сума квадратів відстаней між x_i та x_j:

$$d(C_1,\ C_2) = \sum_{i,j:x_i \in C_1, x_j \in C_2} \frac{\left(d(x_i, x_j)\right)^2}{|C_1| * |C_2|}$$

Всі можливі пари для кожного кластеру комбінуються та обчислюється сума квадратів евклідових відстаней між парами всередині кожного кластера, а потім сумується по всіх кластерах. Вибирається комбінація, яка дає найменшу суму квадратів (в один

кластер об’єднуються ті кластери, які дають найменший приріст дисперсії).

Це один з найпопулярніших методів поруч з методом середнього зв’язку. Вважається дуже ефективним, але тяжіє до утворення кластерів малого розміру.

Цей метод використовує підхід аналізу дисперсій для визначення відстаней між кластерами. Тобто мінімізує суму квадратів будь-яких двох гіпотетичних кластерів, які можуть бути сформовані на кожному кроці.

Перевагою є те, що цей метод добре працює при розділенні кластерів за наявності шумів між кластерами.

Недоліки: тяжіє до утворення компактних сферичних кластерів приблизно однакового розміру, також до утворення кластерів малого розміру, а ще є чутливим до викидів.

2.3 Неієрархічні методи кластеризації

Суть неієрархічних методів кластеризації полягає в тому, що процес кластеризації починається із задання деяких початкових умов (кількість кластерів, кількість ітерацій або правило зупинки, деякі інші параметри кластеризації).

В неієрархічних методах існує проблема визначення кількості кластерів. В загальному випадку їх кількість може бути невідома. Не всі методи потребують задання кількості кластерів із самого початку, проте дозволяють, використовуючи декілька алгоритмів, змінюючи кількість кластерів, що утворюється, або встановлений поріг близькості для об’єднання об’єктів в кластери, досягати найкращого розбиття за заданим критерієм якості.

Обчислювальні процедури більшості неієрархічних методів кластеризації зводяться до виконання наступних дій:

1) Вибір кількості кластерів, на які повинен бути розбитий набір даних, задання початкового розбиття об’єктів та визначення центрів ваги кластерів.

2) Відповідно до обраної міри близькості визначення вмісту кожного кластеру.
3) Після повного розподілу всіх об’єктів по кластерах здійснюється перерахунок центрів ваги нових кластерів.

Кроки 2) і 3) повторюються доти, доки наступна ітерація не дасть такий самий перерахунок центрів ваги нових кластерів.

Метод k-середніх найбільш простий, але в той же час досить неточний метод кластеризації. Він розбиває множину елементів векторного простору на заздалегідь відому кількість кластерів. Дія методу така, що він прагне мінімізувати середньоквадратичне відхилення на точках кожного кластеру. Основна ідея полягає в тому, що на кожній ітерації перераховується центр ваги для кожного кластеру, отриманого на попередньому кроці, потім вектори розбиваються на кластери знову у відповідності з тим, який із нових центрів виявився ближчим по обраній метриці. Алгоритм завершується, коли на якійсь ітерації не відбувається ніяких змін кластерів.

Після отримання результатів кластерного аналізу методом k-середніх потрібно перевірити правильність кластеризації (тобто оцінити, наскільки кластери відрізняються один від одного). Для цього обраховуються середні значення для кожного кластера. За хорошої кластеризації ми повинні отримати такі середні значення, що будуть сильно відрізнятися для всіх вимірів, або принаймні хоча б для їх більшої частини.

Кластерний аналіз методом k-середніх зазвичай використовується у випадках, коли є дуже великі набори даних. Часом цьому методу надають перевагу через те, що він дозволяє об’єктам рухатись з одного кластера в інший, що неможливо в ієрархічних методах.

Недоліки методу:

- Дуже чутливий до викидів, які можуть спотворювати середнє.
- Метод може повільно працювати на великих об’ємах даних.
- Часто важко визначити, скільки треба було б сформувати кластерів.

Одним із можливих способів подолання недоліків є використати спочатку ієрархічний алгоритм, щоб визначити кількість кластерів та їх центри, а потім застосувати цю інформацію для проведення неієрархічної процедури.

Переваги методу:

- Простота використання.
- Швидкість використання.
- Зрозумілість і прозорість методу.

2.4 Вибір оптимальної кількості кластерів

Одним із важливих питань при розв'язанні задач кластеризації є вибір необхідної кількості кластерів. В деяких випадках це число може бути задане апріорно, однак в загальному випадку це число визначається в процесі розбиття множин на різну кількість кластерів.

Є кілька шляхів для визначення оптимальної кількості кластерів, деякі дещо неформальні та суб'єктивні, деякі більш формальні. Наведемо один неформальний, але зручний метод визначення кількості кластерів для методів ієрархічної кластеризації.

При проведенні ієрархічного кластерного аналізу, результати зручно зображувати у вигляді дендрограми, яка демонструє, які кластери були з'єднані на якому кроці аналізу та дистанцію між кластерами (висота) від одного кроку до іншого. На одній зі стадій кластери, що є відносно близькими та були з'єднані, на наступній стадії з'єднані кластери виявилися відносно далеко один від одного. З цього випливає, що оптимальною кількістю кластерів може бути та кількість, що представлена безпосередньо до великого стрибка у відстані між кластерами. Це простіше зрозуміти, дивлячись на дендрограму.

Найкраща кількість кластерів – це кількість вертикальних ліній в дендрограмі, що перетинаються горизонтальними лініями, відстань між якими є перевернутою максимальною вертикальною відстанню без перетину кластеру. На Рис. 2.4.1 найкращим числом кластерів є чотири, оскільки червона

горизонтальна лінія в дендрограмі покриває максимальну вертикальну дистанцію AB.

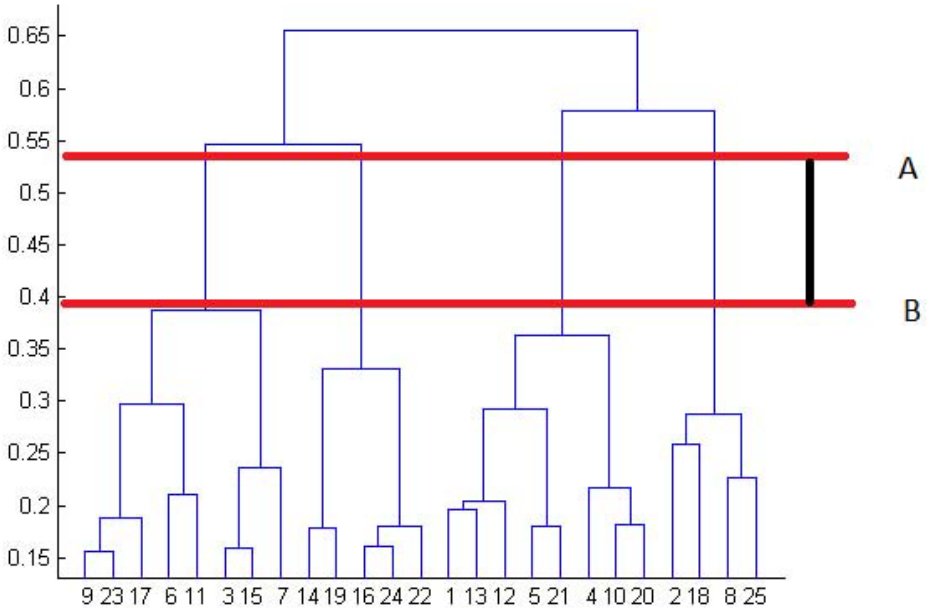

Рис. 2.4.1 – Дендрограма. Оптимальна кількість кластерів – 4

Аналізуючи розбиття на різне число кластерів, можна визначити ту кількість кластерів, при якій кластери розділені найкращим чином. Таку кількість будемо називати оптимальною для заданої вибірки даних. Визначити цю кількість можна за допомогою візуалізації або іншими різними методами.

Розглянемо найбільш вживані методи визначення оптимальної кількості кластерів для методу k-середніх: метод ліктя та метод силуету(окрім них існує більше тридцяти інших показників та методів визначення оптимальної кількості кластерів).

Метод ліктя – це найвідоміший метод, що базується на підрахунку та зображенні внутрішньокластерної суми квадратів відхилень для різної кількості кластерів. Спостерігаючи за зміною кута нахилу отриманої ламаної від стрімкого спадання до більш повільного (лікоть), можна визначити оптимальну кількість кластерів. Зауважимо, що часом цей метод є неоднозначним.

Даний метод розглядає загальну внутрішньокластерну суму квадратів як функцію кількості кластерів. Необхідно вибрати кількість кластерів таким

чином, щоб додавання ще одного кластеру не давало суттєвого покращення загальної внутрішньокластерної суми квадратів.

Алгоритм визначення оптимальної кількості кластерів за методом ліктя:

1) Провести алгоритм кластеризації (наприклад, методом k-середніх) для різних значень k. Наприклад, для кількості кластерів k від одного до десяти.
2) Для кожного значення k обчислюється внутрішньокластерна сума квадратів (WSS): $WSS = \sum_{i=1}^{k} W(C_i)$, де $W(C_i)$ – сума квадратів в кластері C_i.
3) Зображується ламана WSS залежно від кількості кластерів k.
4) Позиція зміни кута нахилу ламаної (лікоть) на отриманому графіку розглядається як оптимальна кількість кластерів.

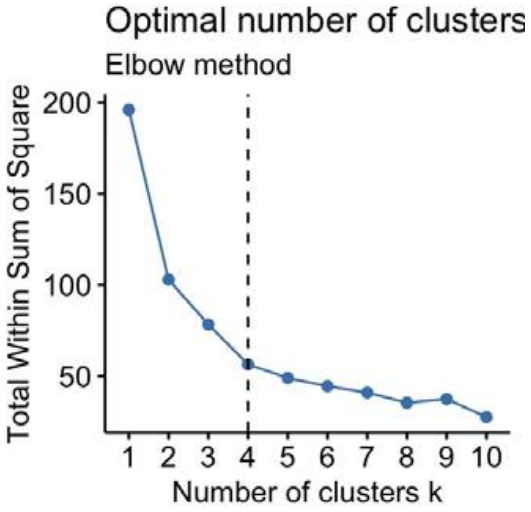

Рис. 2.4.2 – Визначення оптимальної кількості кластерів за допомогою методу ліктя

Альтернативним методом до методу ліктя є метод середнього силуету.

Це також метод пошуку оптимальної кількості кластерів, інтерпретації та перевірки узгодженості всередині кластерів даних. Метод силуету обраховує коефіцієнти силуету кожної точки, яку вимірюють, тобто наскільки точка схожа на свій власний кластер у порівнянні з іншими кластерами.

Значення силуету знаходиться в межах від мінус одиниці до одиниці, де високе значення вказує на те, що об’єкт добре відповідає своєму власному

кластеру і погано відповідає сусіднім. Якщо більшість об’єктів мають високе значення, то конфігурація кластеризації підходить. Якщо ж багато точок мають низьке або й від’ємне значення, то в конфігурації кластеризації може бути надто багато або надто мало кластерів. Відповідно, значення «+1» силуету є ідеальним, значення «-1» є найгіршим. Більше значення відповідає кращій конфігурації кластерів.

Метод визначає значення силуету спостережень для різних значень k (кількості кластерів), зображує отримані значення на діаграмі та визначає оптимальну кількість кластерів як те значення k, для якого значення середнього силуету є найбільшим.

Математично для кожного S_i-го об’єкта ширина силуету обчислюється таким чином:

1) Для кожного i-го об’єкта порахувати середню несхожість (відстань) a_i між i-тим об’єктом та всіма іншими точками кластера, якому він належить.
2) Для всіх інших кластерів C, яким не належить i-й об’єкт, порахувати середню несхожість $d(i, C)$ для всіх об’єктів з C. Найменше з цих значень визначає величину $b_i = \min_C d(i, C)$. Ця величина розглядається як відстань між i-тим об’єктом та найближчим «сусідом» – найближчим кластером, якому i-й об’єкт не належить.
3) Ширина силуету i-го об’єкта визначається за формулою: $S_i = \frac{b_i - a_i}{\max(a_i, b_i)}$.

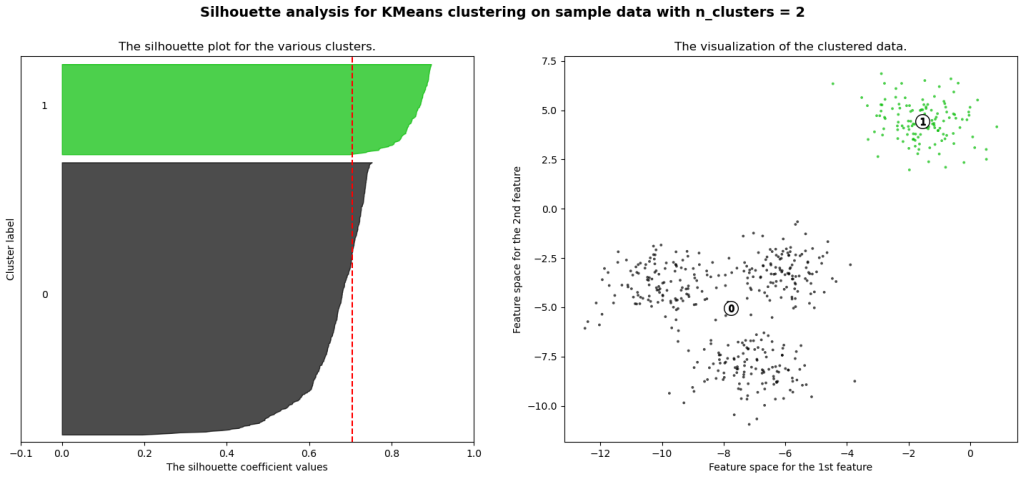

Рис. 2.4.3 – Ліворуч - список кластерів кожної точки даного кластера. Чорна область – S-значення для об'єктів, що належать кластеру 0, зелена – для об'єктів, що належать кластеру 1. Червона пунктирна лінія позначає середнє значення S для кластерів, що розглядаються. Значення приблизно рівне. Праворуч – візуалізація приналежності точок кластерам

Щоб обрати адекватну кількість кластерів, маючи подібне зображення, бажано дотримуватися таких пунктів:

- Середнє значення має бути максимально близьким до одиниці;
- Зображення кожного кластера має бути якомога вище, ніж середнє значення;
- Ширина зображень для кластерів має бути якомога рівномірнішою.

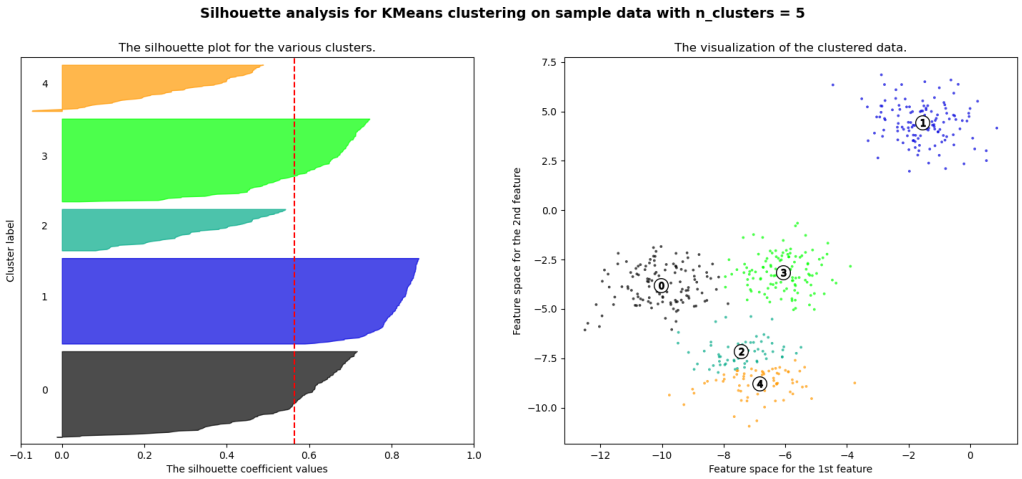

Рис. 2.4.4 – Цей граік S-значень демонструє небажану кількість кластерів, оскільки для деяких кластерів S-значення менше, ніж середнє. Окрім того, області мають дуже різну ширину

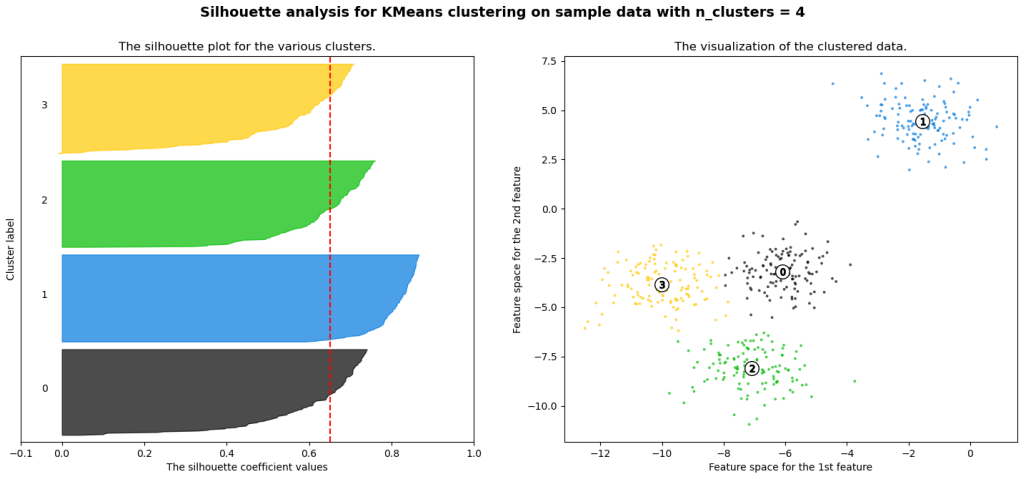

Рис. 2.4.5 – Оптимальна кількість кластерів – чотири

2.5 Перевірка якості кластеризації

Після створення кластерного розв'язку зазвичай виникає запитання, наскільки цей розв'язок стійкий і статистично значимий. Тут існує емпіричне правило – стійке групування повинне зберігатися при зміні методів кластеризації: наприклад, якщо результати ієрархічного кластерного аналізу мають частку співпадінь більше 70% з групуванням по методу k-середніх, то припущення про стійкість приймається.

Проблема оцінки якості кластеризації є проблемою, як мінімум, з двох причин:

- Не існує оптимального алгоритму кластеризації (теорема Клейнберга);
- Багато алгоритмів кластеризації не здатні визначити справжню кількість кластерів в даних. Найчастіше кількість кластерів подається на вхід алгоритму та підбирається кількома запусками алгоритму.

Варто відмітити, що результат кластеризації залежить від застосованого методу, метрики, нормалізації (стандартизації) значень ознак. При використанні різних методів (метрик, стандартизації) кластерного аналізу для одного набору даних можуть бути отримані різні варіанти розбиття. Суттєвий вплив на

кластерну структуру виявляють набір ознак, за яким проводиться класифікація, та тип обраного алгоритму.

Існує думка, що не існує правильних чи неправильних результатів кластеризації, оскільки, за означенням, це метод навчання «без вчителя». Все визначається відповідністю отриманого розв'язку поставленій задачі, яка на практиці часто зводиться просто до того, щоб приблизно оцінити, на скільки груп доцільно розділити дані. При цьому ступінь цієї відповідності зазвичай є суб'єктивним.

Для зняття невизначеності із отриманого результату використовуються деякі характеристики для кількісної оцінки результатів кластеризації.

Виділяють такі дві групи методів оцінки якості кластеризації.

- Зовнішні. Ці методи базуються на порівнянні результату кластеризації з апріорі відомим розподілом на класи.
- Внутрішні. Характеризують якість кластеризації лише по інформації в самих даних.

Внутрішні міри якості кластеризації виписуються за допомогою відповідних функціоналів якості. Найкращим за обраним функціоналом вважають таке розбиття, при якому досягається його екстремальне (мінімальне чи максимальне) значення.

Ці міри якості оцінюють якість структури кластерів, спираючись лише безпосередньо на неї, не використовуючи зовнішньої інформації.

Критерій Калінські-Харабаша, який інколи називають критерієм відношення дисперсії використовується для пошуку оптимального значення кількості кластерів. Він визначається наступним чином:

$$VRC_k = \frac{SS_B}{SS_w} \times \frac{(n-k)}{(k-1)},$$

де k – кількість кластерів, n – кількість вибірок, SS_B – це похибка суми квадратів між групами, а SS_w – похибка суми квадратів усередині групи. Тоді, чим значення SS_w менше, а SS_B більше, тобто чим більше значення індексу - тим кластеризація краща. Є внутрішнім методом оцінки якості кластеризації.

Індекс Девіса-Болдіна також є внутрішнім методом оцінки якості кластеризації. Нехай $R_{i,j}$ – міра того, наскільки якісною є кластеризація. Ця міра повинна враховувати $M_{i,j}$ розділення між і-им і j-им кластерами, яке в ідеалі повинне бути якомога більшим, та S_i – розкид всередині кластера і, який повинен бути якомога меншим. Тоді індекс Девіса-Болдіна визначається як відношення S_i та $M_{i,j}$ за умови, що такі властивості зберігаються:

1) $R_{i,j} \geq 0$
2) $R_{i,j} = R_{j,i}$
3) Коли $S_j \geq S_k$ і $M_{i,j} = M_{i,k}$, тоді $R_{i,j} > R_{i,k}$
4) Коли $S_j = S_k$ і $M_{i,j} \leq M_{i,k}$, тоді $R_{i,j} > R_{i,k}$

Тоді чим нижче значення індексу, тим краще розділені кластери і тим краща «герметичність» всередині кластерів.

Тоді $R_{i,j} = \frac{S_j + S_j}{M_{i,j}}$, а сам індекс Девіса-Болдіна визначається як:

$$DB = \frac{1}{n}\sum_{i=1}^{n} \max_{j \neq i} R_{i,j}.$$

РОЗДІЛ 3 ЗАСТОСУВАННЯ КЛАСТЕРНОГО АНАЛІЗУ ДЛЯ СЕГМЕНТАЦІЇ РИНКУ

Ми розглянемо методи кластеризації для сегментації ринку, використовуючи вибіркові дані про клієнтів із бази даних продовольчої фірми iFood, що займається доставкою їжі у Бразилії. Сегментація ринку - це поділ клієнтів на групи, які відображають схожість клієнтів у кожному кластері.

У роботі ми здійснюємо поділ клієнтів на сегменти для оптимізації значущості кожного клієнта для бізнесу. Це буде корисним для модифікації продуктів відповідно до конкретних потреб і поведінки клієнтів і, окрім того, допоможе бізнесу задовольнити потреби кожного типу клієнтів.

В якості середовища програмування було обрано Jupyter Notebook та мову програмування Python 3.9.0. Для роботи зі структурами даних було обрано бібліотеку Pandas, для кластеризації була обрана бібліотека Scikit-learn, а для візуалізації результатів були обрані бібліотеки Matplotlib та Seaborn.

3.1 Підготовка вхідних даних

Датасет містить 2240 рядків(об'єктів) та 27 атрибутів(колонок). Атрибути можуть бути розділені на такі групи за інформацією, яку вони містять: клієнти, продукти, просування, місце.

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	MntWines	...	NumCatalogPurchases	NumStorePurchases
0	5524	1957	Graduation	Single	58138.0	0	0	04.09.2019	58	635	...	10	
1	2174	1954	Graduation	Single	46344.0	1	1	08.03.2021	38	11	...	1	
2	4141	1965	Graduation	Together	71613.0	0	0	21.08.2020	26	426	...	2	
3	6182	1984	Graduation	Together	26646.0	1	0	10.02.2021	26	11	...	0	
4	5324	1981	PhD	Married	58293.0	1	0	19.01.2021	94	173	...	3	
5	7446	1967	Master	Together	62513.0	0	1	09.09.2020	16	520	...	4	
6	965	1971	Graduation	Divorced	55635.0	0	1	13.11.2019	34	235	...	3	
7	6177	1985	PhD	Married	33454.0	1	0	08.05.2020	32	76	...	0	
8	4855	1974	PhD	Together	30351.0	1	0	06.06.2020	19	14	...	0	
9	5899	1950	PhD	Together	5648.0	1	1	13.03.2021	68	28	...	0	

10 rows × 27 columns

Рисунок 3.1.1 – Вигляд датасету

Поглянемо, які колонки входять до кожної групи.

Клієнти (характеристики клієнтів):

- ID: унікальний ідентифікатор клієнта
- Year_Birth: Рік народження клієнта

- Education: Рівень освіти клієнта
- Marital_Status: Сімейний статус клієнта
- Income: Річний дохід клієнта
- Kidhome: Кількість дітей у сім'ї клієнта
- Teenhome: Кількість підлітків у сім'ї клієнта
- Dt_Customer: Дата рєєстрації клієнта
- Recency: Кількість днів від останньої покупки клієнта
- Complain: 1, якщо клієнт скаржився за останній рік, 0 – якщо ні

Продукти (суми, що витрачена на різні продукти за останній рік):

- MntWines: Сума, витрачена на винні продукти за останній рік
- MntFruits: Сума, витрачена на фрукти за останній рік
- MntMeatProducts: Сума, витрачена на м'ясні продукти за останній рік
- MntFishProducts: Сума, витрачена на рибу за останній рік
- MntSweetProducts: Сума, витрачена на солодощі за останній рік
- MntGoldProds: Сума, витрачена на золото за останній рік

Просування (чи прийняв користувач пропозицію за кожної рекламної кампанії):

- NumDealPurchases: Кількість покупок, що здійснив клієнт зі знижкою
- AcceptedCmp1: 1, якщо клієнт прийняв акційну пропозицію в першій кампанії, 0 – інакше
- AcceptedCmp2: 1, якщо клієнт прийняв акційну пропозицію в другій кампанії, 0 – інакше
- AcceptedCmp3: 1, якщо клієнт прийняв акційну пропозицію в третій кампанії, 0 – інакше
- AcceptedCmp4: 1, якщо клієнт прийняв акційну пропозицію в четвертій кампанії, 0 – інакше

- AcceptedCmp5: 1, якщо клієнт прийняв акційну пропозицію в п’ятій кампанії, 0 – інакше
- Response: 1, якщо клієнт прийняв акційну пропозицію в останній кампанії, 0 – інакше

Місце (кількість покупок з різних каналів продажу):

- NumWebPurchases: Кількість покупок, здійснених через веб-сайт компанії
- NumCatalogPurchases: Кількість покупок, здійснений з використанням каталогу продуктів
- NumStorePurchases: Кількість покупок, здійснених в магазині
- NumWebVisitsMonth: Кількість відвідувань веб-сайту компанії за останній місяць.

Після того, як ми описали структуру датасету, ми очищуємо наші дані від пропущених значень, формуємо нові колонки, що допоможуть у подальшій кластеризації, видаляємо ті, які нам не потрібні та позбуваємось викидів, які можуть негативно впливати на якість кластеризації.

Нові сформовані колонки:

- Колонка Age, що буде вказувати на вік кожного клієнта
- Колонка Spent, що вказуватиме на суму, витрачену клієнтом на усі категорії продуктів за рік(сума усіх витрат)
- Колонка Living_With, щоб виділити у якій сімейній ситуації щодо проживання перебуває клієнт
- Колонка Children, яка вказуватиме на загальну кількість дітей у сім’ї
- Колонка Family_Size, що міститиме інформацію про розмір сім’ї кожного клієнта
- Колонка Parenthood, яка вказуватиме чи має клієнт дітей
- Модифікація колонки Education, за допомогою звуження кількості категорій до трьох.

Після очищення даних у датасеті залишилося 2202 об’єкти.

3.2 Кластеризація даних про купівельну поведінку клієнтів

У цій роботі ми будемо кластеризувати окремо два набори даних, що сформовані із початкового: один із них буде містити інформацію про купівельну поведінку клієнтів (витрати), інший - про кількість покупок із різних каналів продажу (покупки). Визначимо, яка кластеризація краще розділяє клієнтів і опишемо її.

Етапи кластеризації:

- Застосування методу ліктя, методу силуету для визначення оптимальної кількості кластерів, за потреби - використання індексів оцінки кластеризації
- Кластеризація за допомогою методу k-середніх та агломеративної кластеризації (метод Варда та метод далекого сусіда)
- Дослідження кластерів, утворених за допомогою діаграми розсіювання

Спочатку ми розглянемо дані, що містять інформацію про купівельну поведінку клієнтів (тобто дані про витрати на різні продукти). Подивимось, яку кількість кластерів нам пропонують обрати методи, вказані вище.

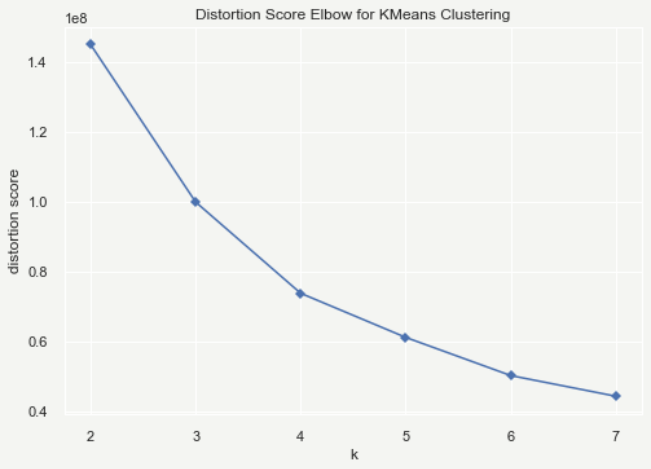

Рисунок 3.2.1 - Метод ліктя для визначення кількості кластерів. Дані про витрати

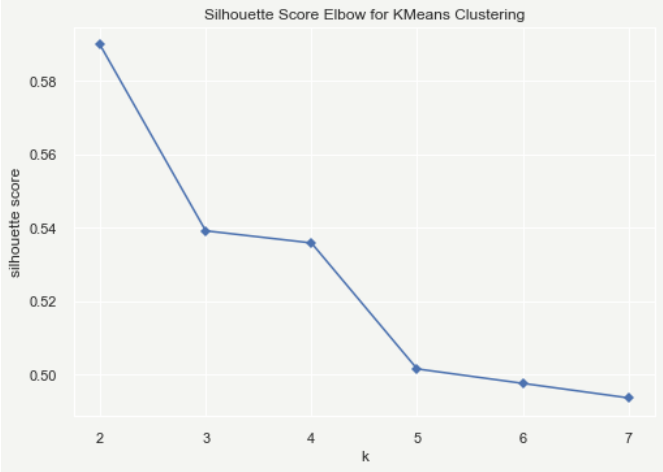

Рисунок 3.2.2 - Метод силуету для визначення кількості кластерів. Дані про витрати

Метод ліктя вказує, що чотири або три кластери - можлива оптимальна кількість кластерів для цих даних, а метод силуету пропонує обрати нам кількість кластерів рівну двом.

Варто поглянути на результати інших методів пошуку оптимальної кількості кластерів, щоб вибір k був достатньо обгрунтованим. Скористаємося двома індексами оцінки якості кластеризації.

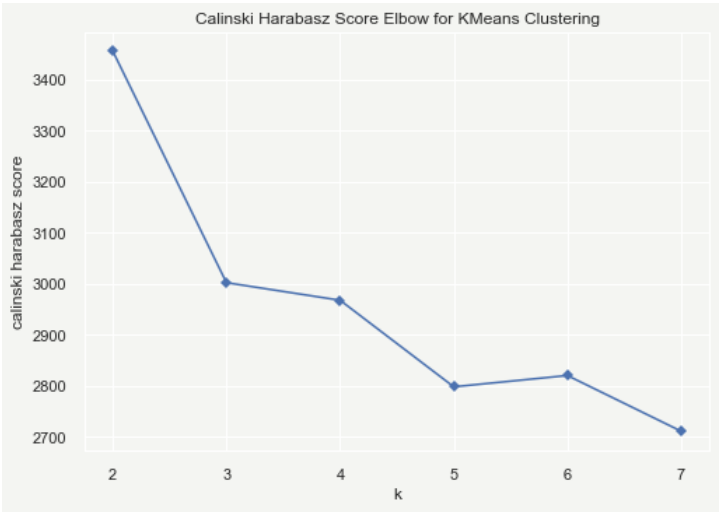

Рисунок 3.2.3 - Індекс Калінські-Харабаша для методу k-середніх. Дані про витрати

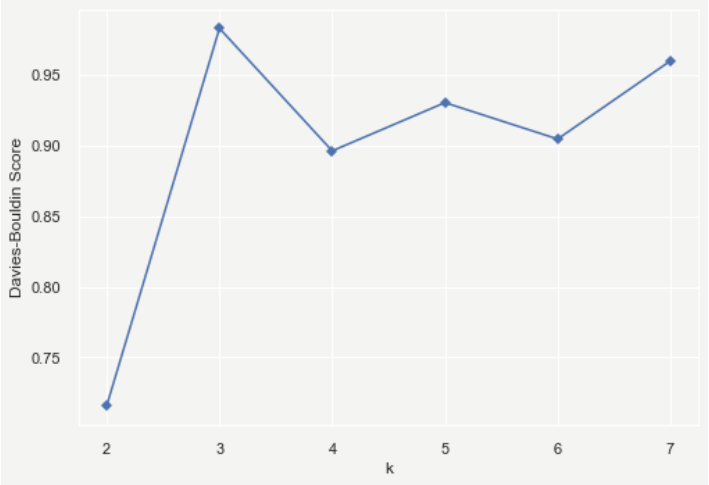

Рисунок 3.2.4 - Індекс Девіса-Болдіна для методу k-середніх. Дані про витрати

Обидва графіки показують, що найкращий вибір для k рівний двом (індекс Калінські-Харабаша повинен бути найвищим, а Девіса-Болдіна найнижчим). Тому для методу k-середніх будемо ділити дані на два кластери.

Також поглянемо, яку кількість кластерів запропонує обрати дендрограма для методів агломеративної кластеризації.

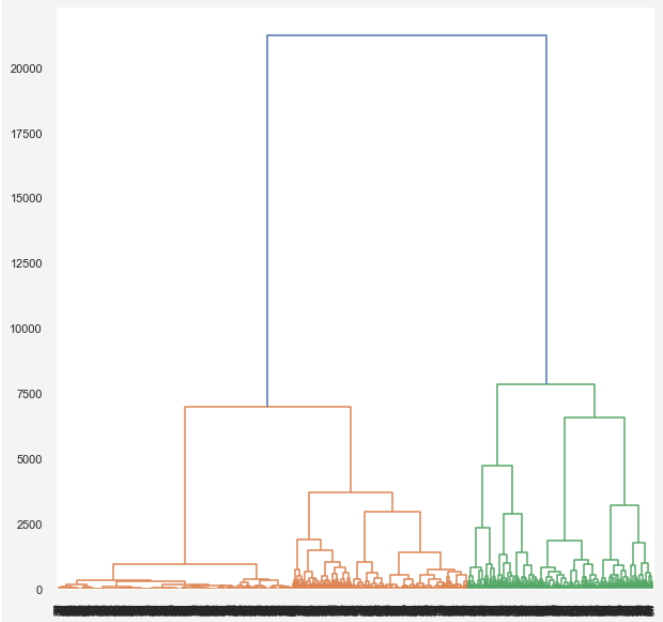

Рисунок 3.2.5 - Дендрограма. Дані про витрати

З дендрограми бачимо, що найкраще – взяти два кластери.

Тепер можемо кластеризувати дані про витрати методами k-середніх, Варда та дальнього сусіда та зобразити кластери у просторі головних компонент.

Набір даних, що містить інформацію про купівельну поведінку клієнтів складається із багатьох ознак (Wines, Fruits, Meat, Fish, Sweets, Gold). Чим більша кількість ознак, тим важче із ними працювати. Багато з них є корельованими, а отже, зайві.

Зменшення розмірності – це зменшення кількості випадкових величин, що розглядаються, шляхом отримання набору головних компонент.

Метод головних компонент (МГК, Principal component analysis(PCA)) – це метод зменшення розмірності даних, при якому втрата інформації мінімізована. Він з'єднує корельовані ознаки і створює таку ж кількість ознак, що не корелюються одна з одною, і стискає більшу кількість інформації у перші компоненти, що допомагає впоратися з мультиколінеарністю.

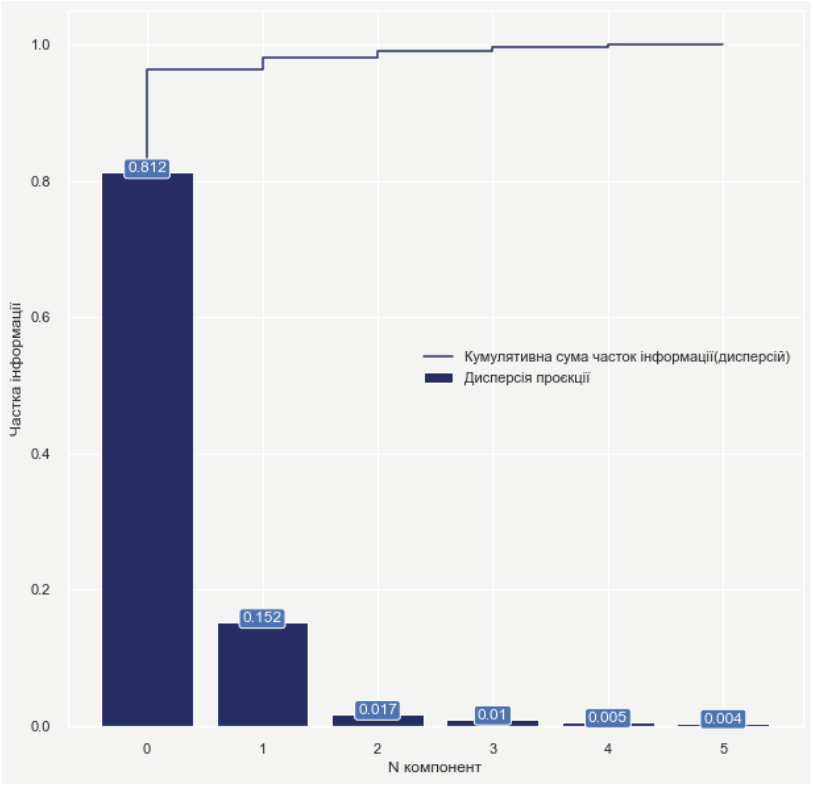

Рисунок 3.2.6 — МГК. Розподіл дисперсії між головними компонентами. Дані про витрати

Оберемо три перших головних компоненти, які містять у собі біля 98% інформації, що є дуже гарним показником і поглянемо, як наші дані виглядатимуть за зменшеної розмірності до трьох вимірів.

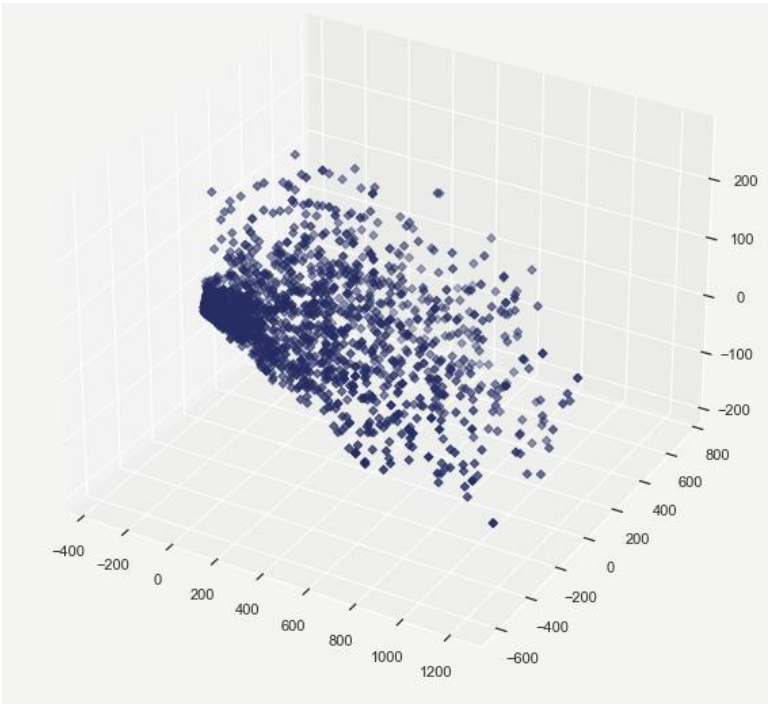

Рисунок 3.2.7 — Проєкція даних про витрати за зменшеної розмірності

Тепер поглянемо на кластеризовані дані щодо купівельної поведінки у просторі трьох головних компонент.

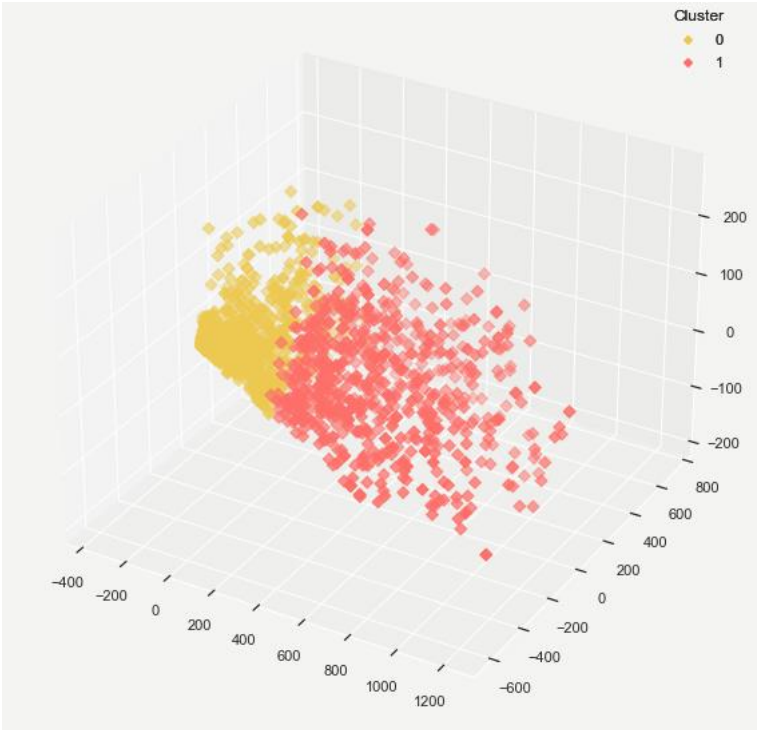

Рисунок 3.2.8 — Кластери сформовані методом k-середніх. Дані про витрати

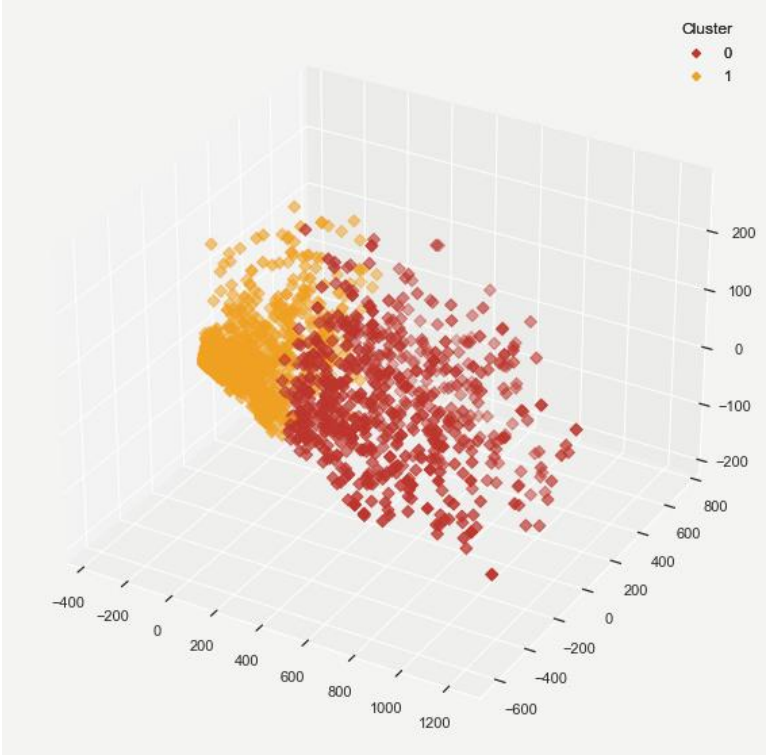

Рисунок 3.2.9 — Кластери сформовані методом Варда. Дані про витрати

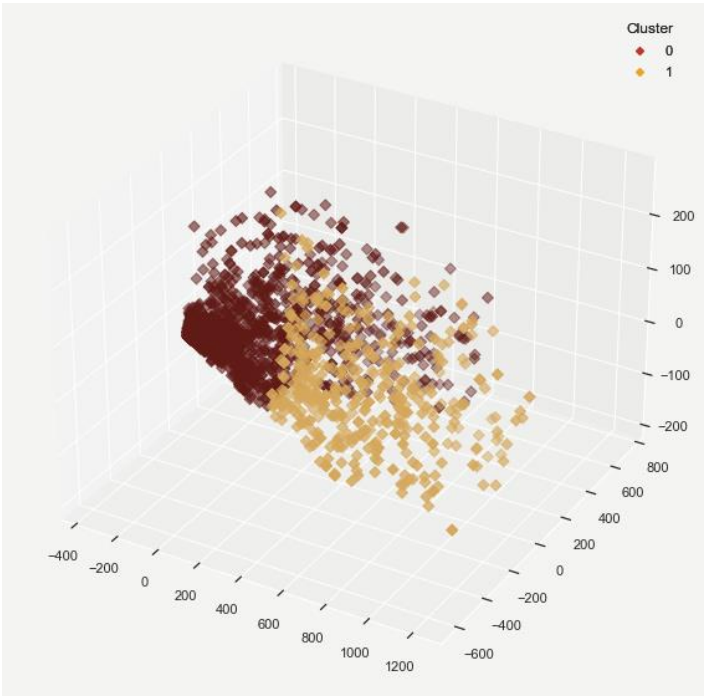

Рисунок 3.2.9 — Кластери сформовані методом дальнього сусіда. Дані про витрати

3.3 Кластеризація даних про кількість покупок

Тепер поглянемо на кластеризацію тих даних, що містять інформацію про покупки із різних каналів продажу (такі атрибути: NumDealsPurchases, NumWebPurchases, NumStorePurchases, NumWebVisitsMonth). Наші дії щодо процесу кластеризації цього набору даних аналогічні діям у попередньому підрозділі.

Обираємо оптимальну кількість кластерів:

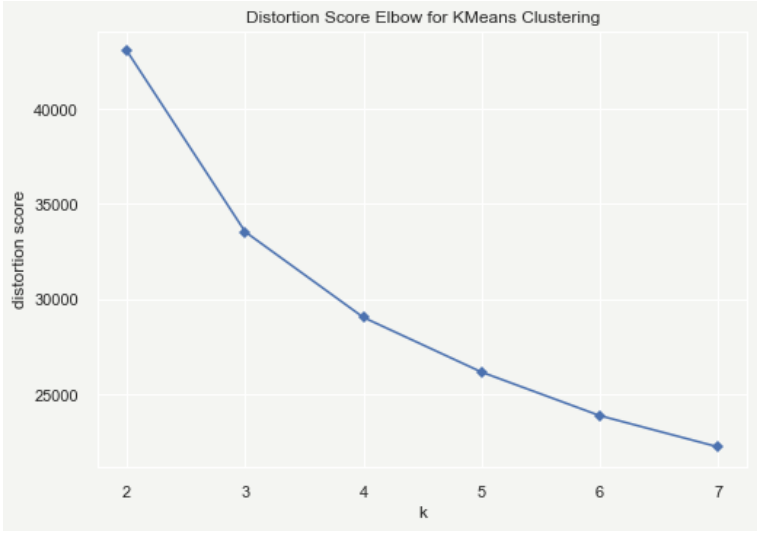

Рисунок 3.3.1 - Метод ліктя для визначення кількості кластерів. Дані про покупки

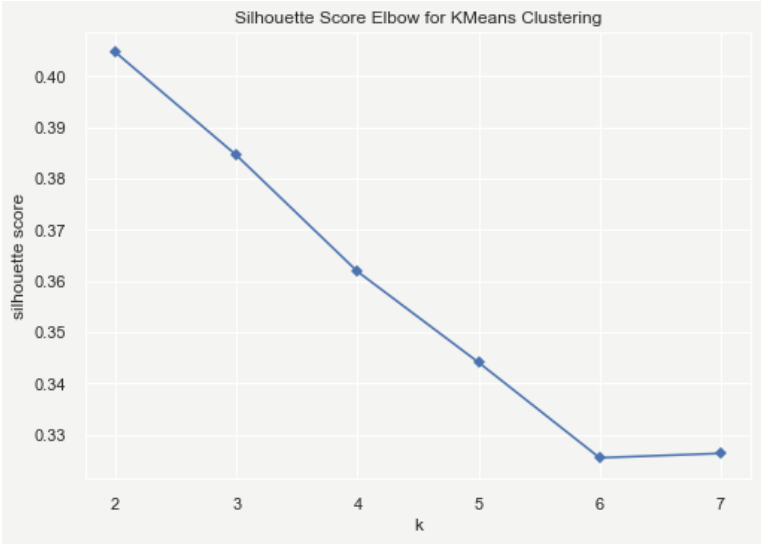

Рисунок 3.3.2 - Метод силуету для визначення кількості кластерів. Дані про покупки

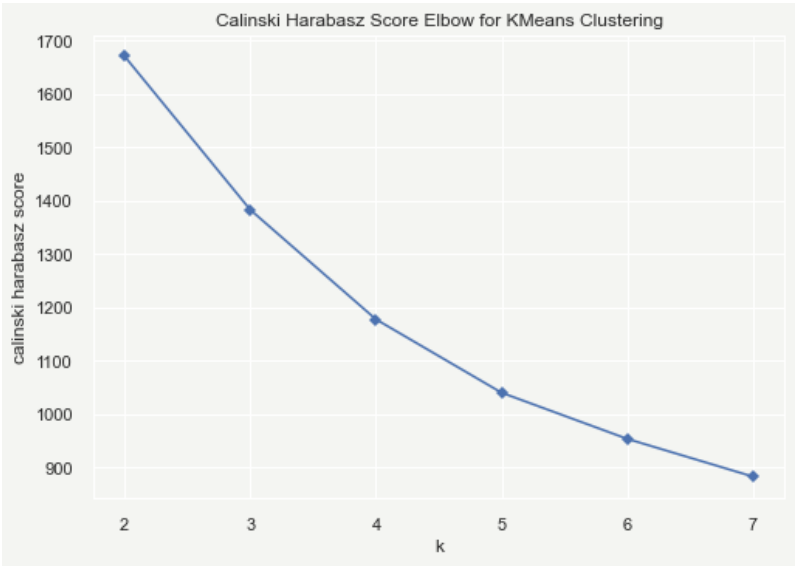

Рисунок 3.3.3 - Індекс Калінські-Харабаша для методу k-середніх. Дані про покупки

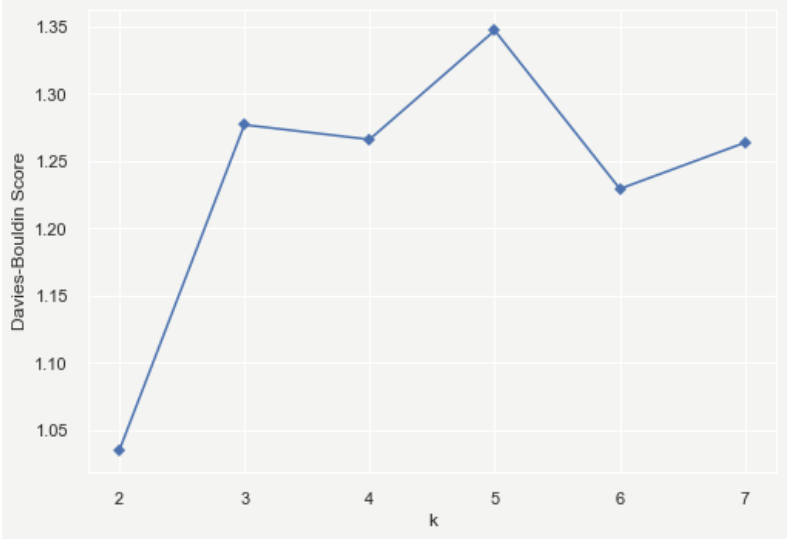

Рисунок 3.3.4 - Індекс Девіса-Болдіна для методу k-середніх. Дані про покупки

Метод ліктя пропонує обрати три або чотири кластери, але метод силуету та індекси Калінські-Харабаша і Девіса-Болдіна пропонують обрати два. Тому оберемо кількість k рівну двом.

Кластеризуємо наш набір даних, та зображуємо його у просторі головних компонент.

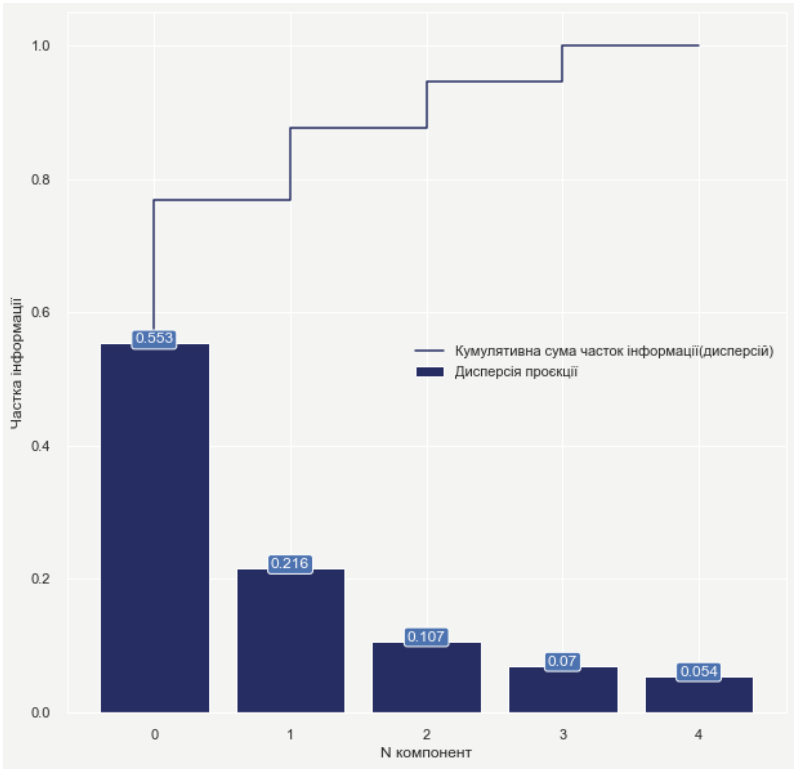

Рисунок 3.3.5 — Розподіл дисперсії між головними компонентами. Дані про покупки

Обираємо три перших головних компоненти, які містять у собі біля 87% інформації, що є гарним показником.

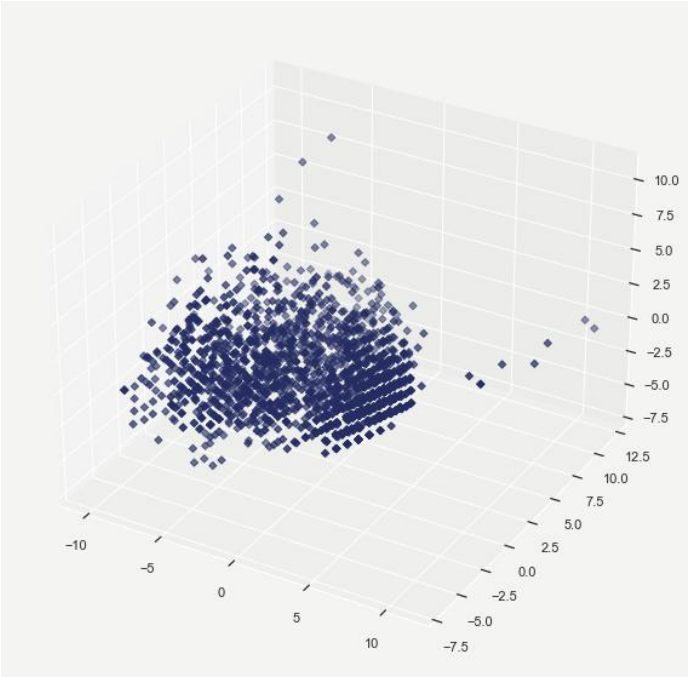

Рисунок 3.3.6 — Проєкція даних про покупки за зменшеної розмірності

Тепер поглянемо на кластеризовані дані про покупки із різних каналів продажу у просторі трьох головних компонент.

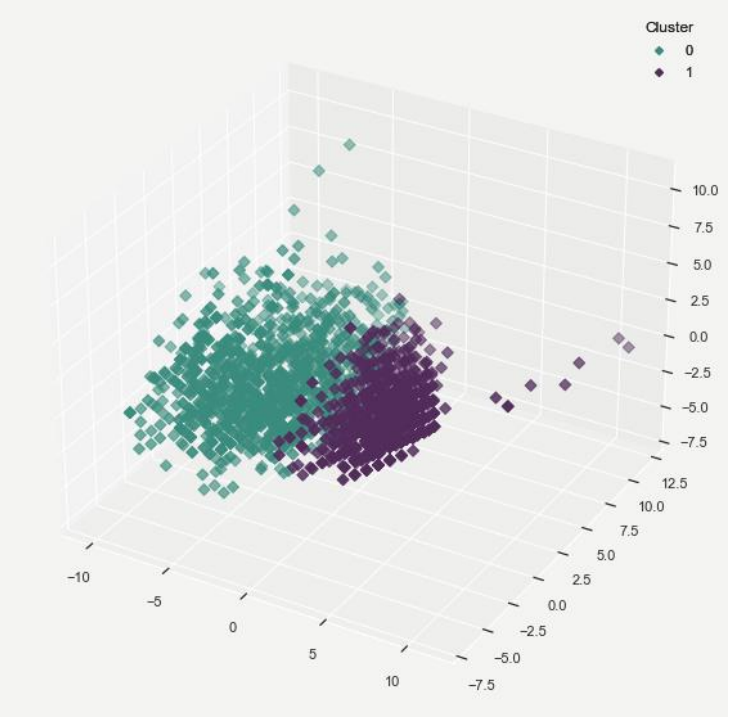

Рисунок 3.3.7 — Кластери сформовані методом k-середніх. Дані про покупки

Рисунок 3.3.8 — Кластери сформовані методом Варда. Дані про покупки

Рисунок 3.3.9 — Кластери сформовані методом дальнього сусіда. Дані про покупки

Бачимо, що метод дальнього сусіда погано розділив кластери, тому ми не використовуватимемо його для подальшого аналізу, тоді як метод Варда та метод k-середніх сформував кластери по-схожому, що свідчить про стійкість кластеризації.

3.4 Оцінювання моделей

Оскільки кластеризація - це метод навчання без учителя, складно оцінити якість кластеризації. Тому в цьому підрозділі вивчимо закономірності в утворених різними методами кластерах із даних про купівельну поведінку та з даних про покупки з різних каналів продажів.

Для цього розглядатимемо уже кластеризовані дані шляхом розвідувального аналізу даних та зробимо висновки.

Спершу поглянемо яка кількість об’єктів входить до кожного кластеру. Спочатку для даних про витрати на різні види продуктів, а потім на дані про покупки із різних каналів продажу.

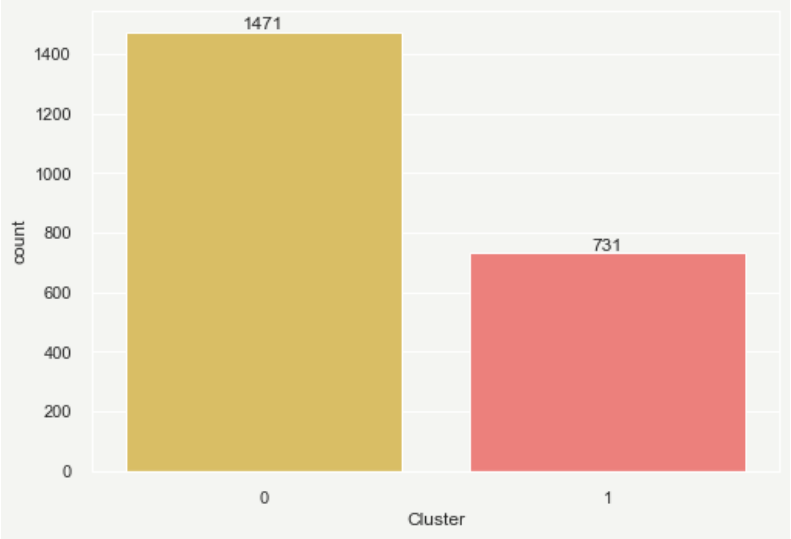

Рисунок 3.4.1 — Розподіл кластерів за методом k-середніх. Дані про витрати

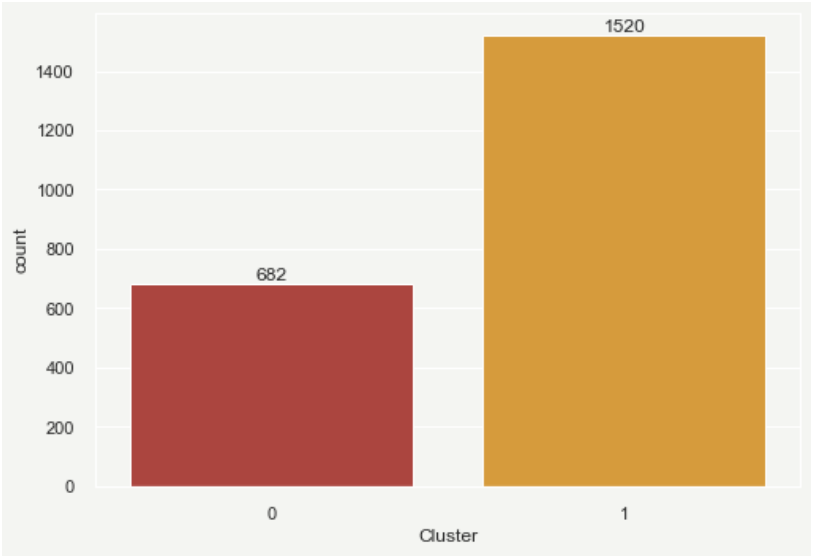

Рисунок 3.4.2 — Розподіл кластерів за методом Варда. Дані про витрати

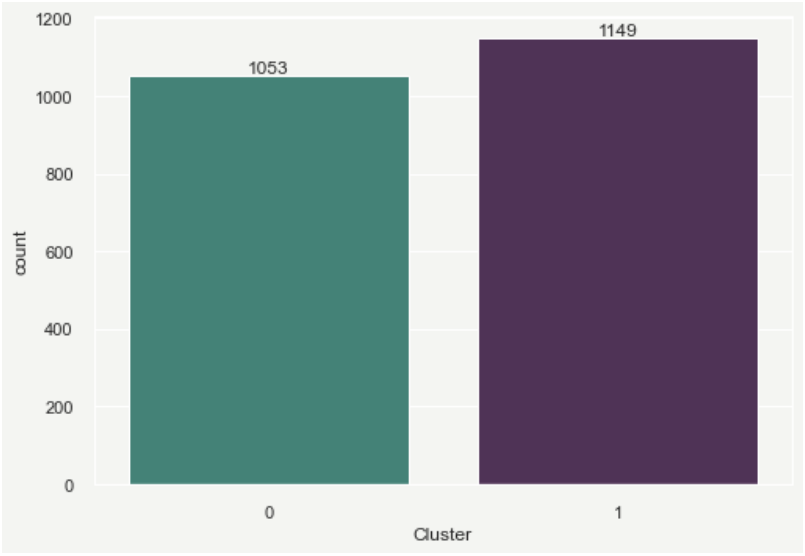

Рисунок 3.4.3 — Розподіл кластерів за методом k-середніх. Дані про покупки

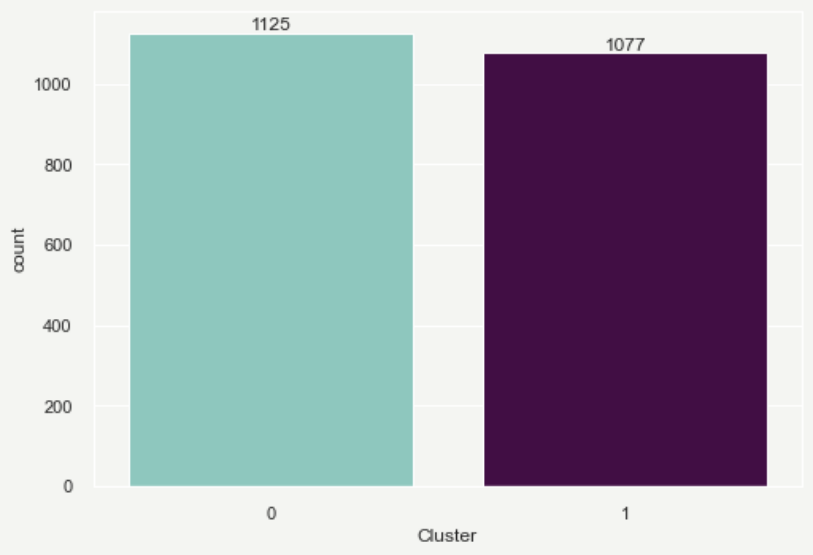

Рисунок 3.4.4 — Розподіл кластерів за методом Варда. Дані про покупки

Впевнилися, що методи Варда та k-середніх по-схожому сформували і дані про купівельну поведінку (витрати), і дані про покупки з різних каналів продажу (покупки). Отже, для подальшого аналізу оберемо кластери, що сформовані методом Варда для обох наборів даних. Окрім того, з графіків бачимо, що кластери, сформовані з даних про купівельну поведінку та кластери, сформовані з даних про кількість покупок із різних каналів продажу містять у собі різну кількість об’єктів. Пізніше ми оберемо за яким із цих наборів ми будемо описувати кластери клієнтів.

3.5 Профілювання клієнтів

Тепер, коли сформовані кластери клієнтів за їх купівельною поведінкою та за покупками з різних каналів продажу, варто подивитися, ким представлений кожен кластер. Для цього потрібно профілювати сформовані кластери, щоб з'ясувати, який тип клієнта нині є найбільш зацікавленим у витратах та покупках у магазині, а якому потрібно більше уваги з боку маркетингової команди.

Щоб це з'ясувати будемо дивитися на розподіл кластерів по різних характеристиках. На основі цього будуть побудовані висновки. Для профілювання, як зазначено раніше, ми обираємо кластеризацію здійснену

методом Варда для двох наборів даних, що містять відповідно інформацію про купівельну поведінку клієнтів (витрати) та про кількість покупок із різних каналів продажу (покупки).

Подивимось як розподілені об'єкти у кластерах сформованих методами k-середніх та Варда відносно Income та Spent (доходу та витрат). Після чого подивимось на розподіл кластерів за доходами та витратами на окремі продукти.

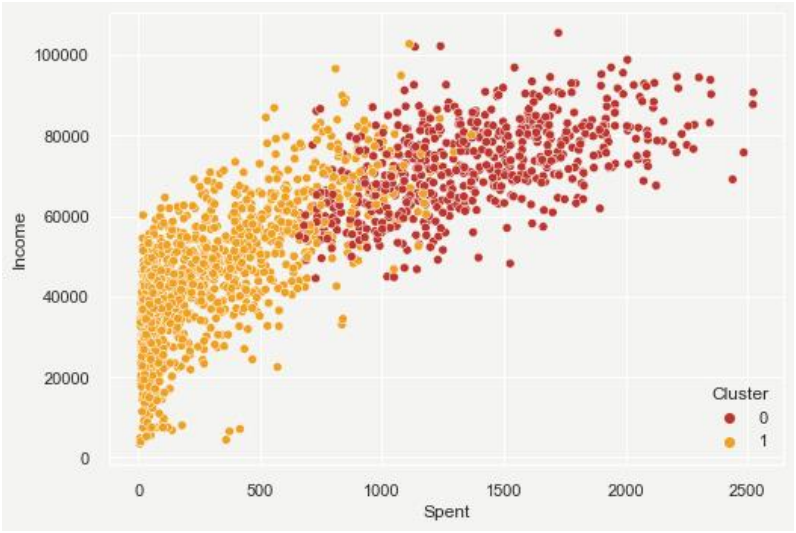

Рисунок 3.5.1 — Розподіл кластерів за методом Варда за доходами й річними витратами. Дані про витрати

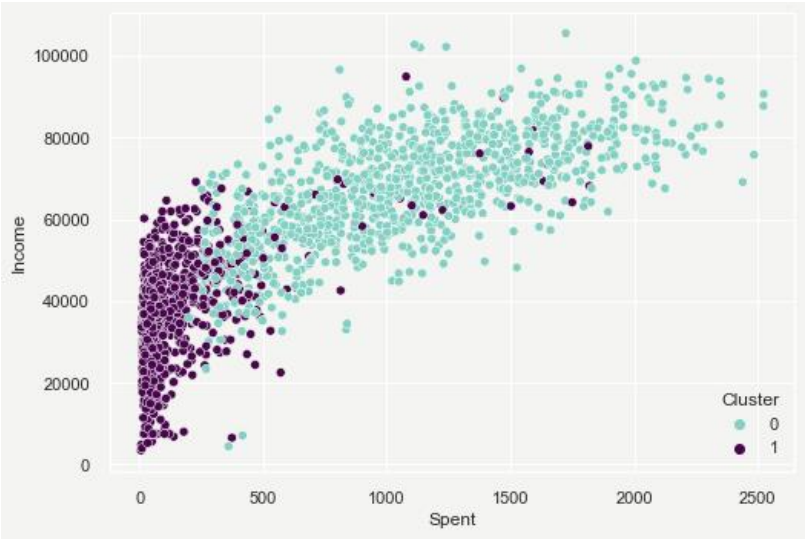

Рисунок 3.5.2 — Розподіл кластерів за методом Варда за доходами й річними витратами. Дані про покупки

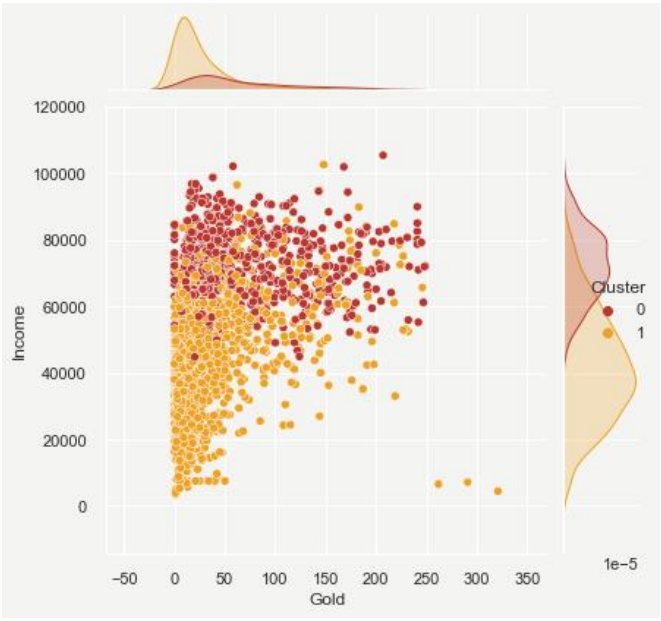

Рисунок 3.5.3 — Розподіл кластерів за доходами та витратами на золото. Дані про витрати

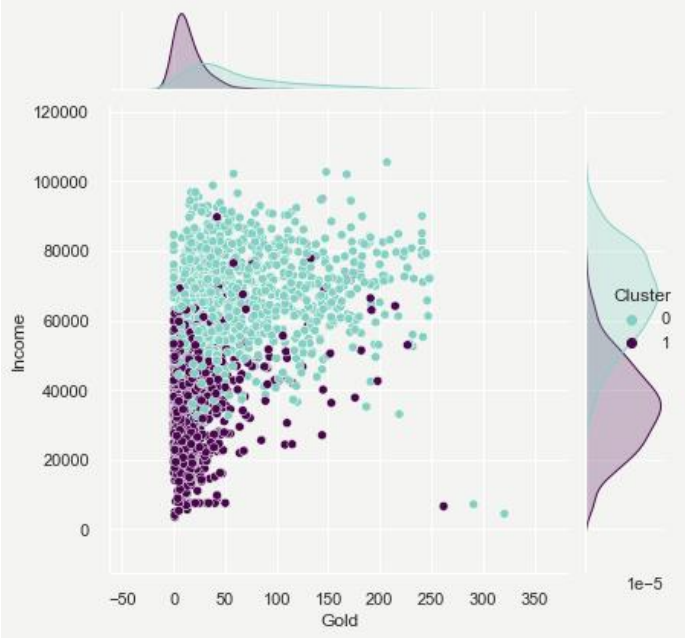

Рисунок 3.5.4 — Розподіл кластерів за доходами та витратами на золото. Дані про покупки

Бачимо, що кластери сформовані з набору даних про купівельну поведінку клієнтів (витрати) трохи відрізняються від сформованих кластерів за покупками з різних каналів продажу. «Кластер 1», сформований з даних про витрати, який представляє собою групу людей, що мають низький дохід все ж, витрачають великі суми на золото, а от у «Кластері 1», який сформований з даних про покупки, набагато менше таких людей. Також об’єкти першого кластеру з даних

про витрати є дуже розкиданими, що спостерігаємо на Рис. 3.5.3, і така тенденція зберігається для усіх інших видів продуктів, тому для профілювання, все ж, оберемо ті кластери, що сформувалися з набору даних про кількість покупок із різних каналів продажу.

Тоді описати сформовані кластери можна так:

- Кластер 0: Високий дохід та середні й високі витрати
- Кластер 1: Низький дохід та низькі витрати

Продовжимо профілювання клієнтів.

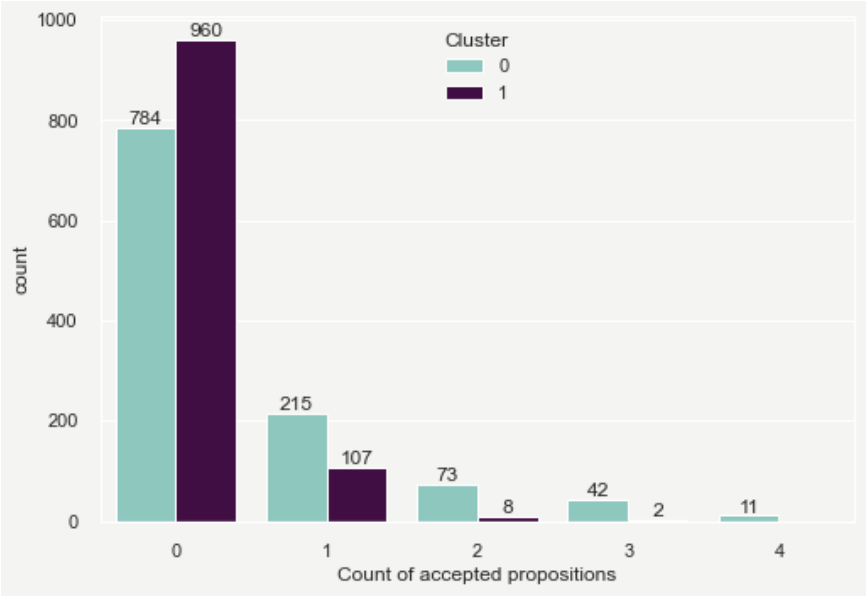

Рисунок 3.5.5 — Участь у маркетингових кампаніях. Дані про покупки

Великої кількості учасників у маркетингових кампаніях на Рис. 3.5.5 ми не спостерігаємо. Окрім того, ніхто не бере участі у всіх п’яти. Можливо, для збільшення продажів необхідно краще продумати самі кампанії.

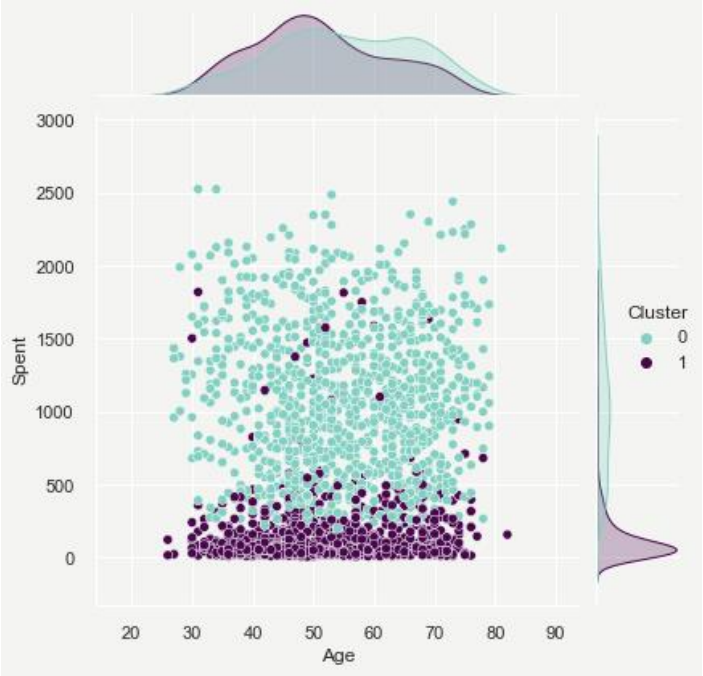

Рисунок 3.5.6 — Розподіл кластерів за віком та витратами. Дані про покупки

З рисунку вище можемо зробити висновок, що в обох кластерах присутні клієнти всіх вікових категорій.

Тепер дослідимо як розподілилися кластери щодо окремих каналів продажів.

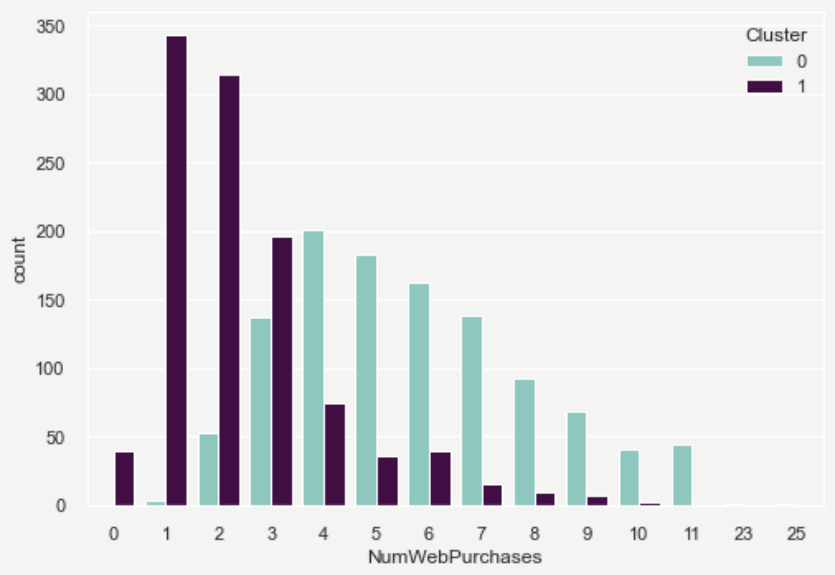

Рисунок 3.5.7 — Кількість об'єктів у кластерах за ознакою кількості онлайн покупок. Дані про покупки

Із Рис. 3.5.7 можемо описати кластери так : Кластер 1 здебільшого здійснює від однієї до шести онлайн покупок, в той час як Кластер 0 здійснює від двох до одинадцяти покупок онлайн.

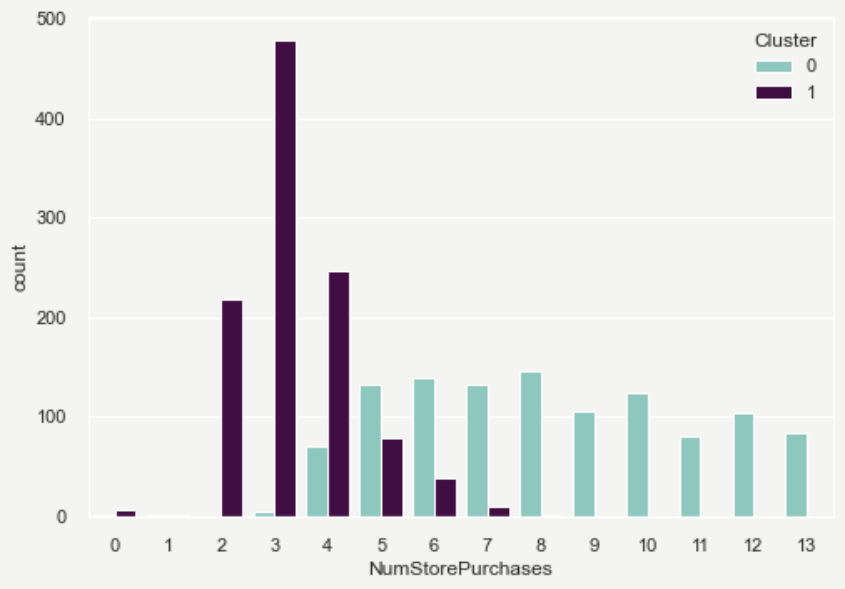

Рисунок 3.5.8 — Кількість об'єктів у кластерах за ознакою кількості покупок в магазині. Дані про покупки

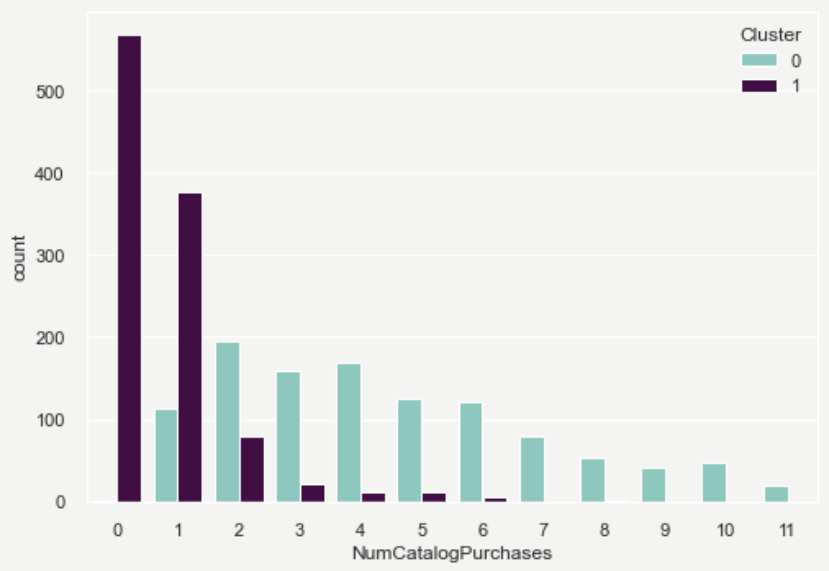

Рисунок 3.5.9 — Кількість об'єктів у кластерах за ознакою кількості покупок через каталог магазину. Дані про покупки

Така ж ситуація щодо кількості покупок в магазині та через каталог магазину. Із Рис. 3.5.8 та Рис. 3.5.9 можемо описати кластери так: Кластер 1 здійснює здебільшого від двох до шести покупок в магазині та одну покупку через каталог, зрідка дві, але більше половини об’єктів цього кластеру узагалі не

здійснюють покупок через каталог, а от об’єкти Кластеру 0 рівномірно розподілені за ознакою кількості покупок у магазині та через каталог (від чотирьох до тринадцяти покупок у магазині та від однієї до одинадцяти через каталог).

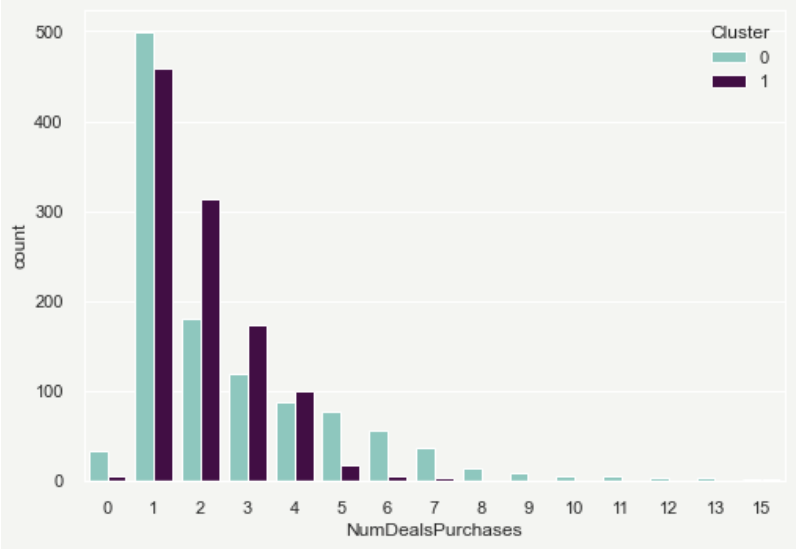

Рисунок 3.5.10 — Кількість об'єктів у кластерах за ознакою кількості акційних покупок. Дані про покупки

А от щодо акційних покупок, то кластери сформовані по-схожому. Відмінність у тому, що Кластер 0 робить у загальному більше акційних покупок, ніж Кластер 1.

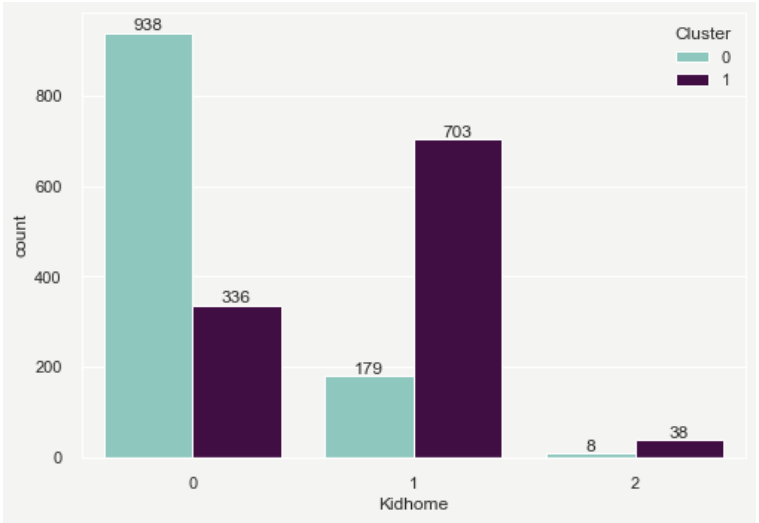

Рисунок 3.5.11 — Кількість об'єктів у кластерах за ознакою кількості малих дітей у сім'ї

Із рисунку вище: Кластер 0 здебільшого не має малих дітей у сім’ї, на відміну від Кластера 1.

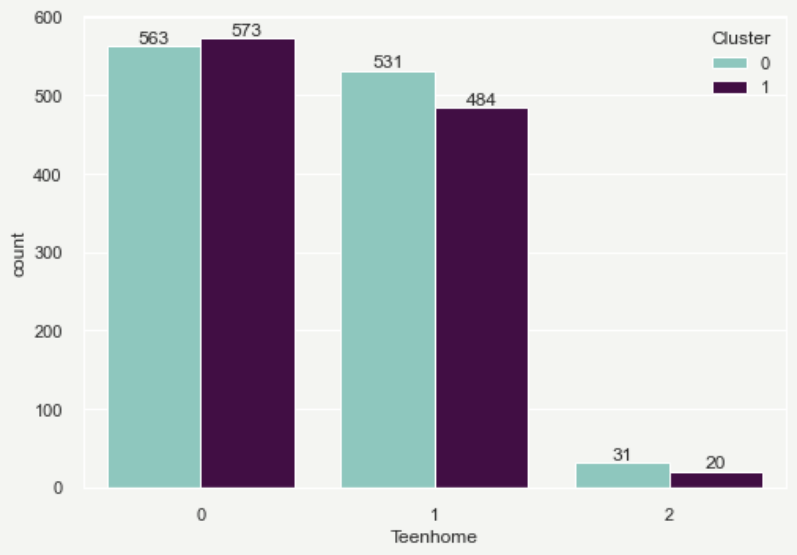

Рисунок 3.5.12 — Кількість об'єктів у кластерах за ознакою кількості підлітків у сім'ї

Бачимо, що у першого і другого кластерів приблизно половина має підлітка у сім’ї.

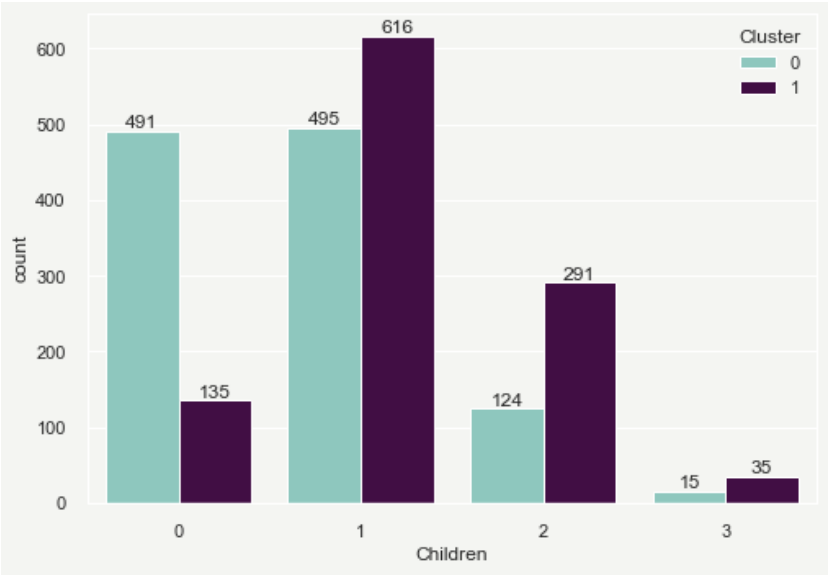

Рисунок 3.5.13 — Кількість об'єктів у кластерах за ознакою кількості дітей у сім'ї

Узагальнюючи, можемо сказати, що приблизно у половини об’єктів сумарна кількість дітей у сім’ї і у першому, і у другому кластері рівна одиниці. Проте, у Кластері 0 інша половина не має дітей узагалі, а десята частина мають

двох дітей, тоді як третина об’єктів Кластеру 1 мають двох дітей, а десята частина не має дітей узагалі.

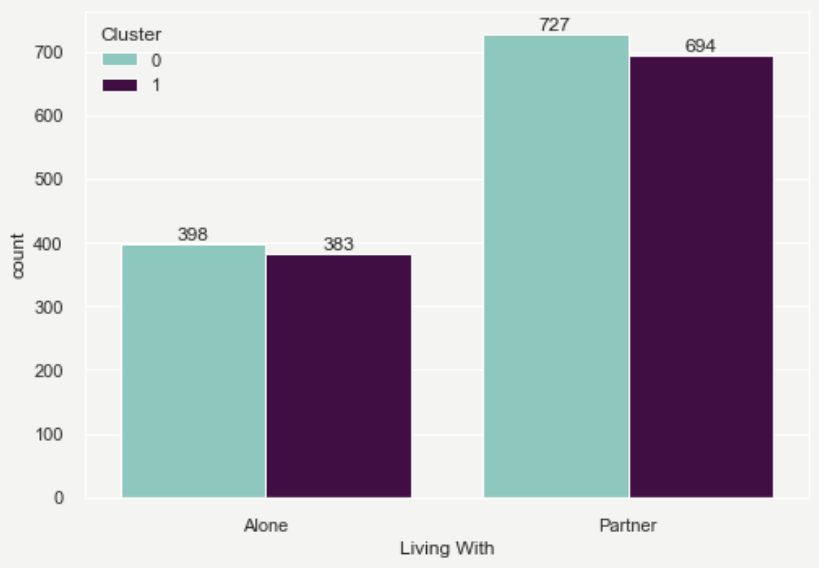

Рисунок 3.5.14 — Кількість об'єктів у кластерах за ознакою наявності партнера

За ознакою наявності партнера кластери сформовані однаково – лише приблизно третина об’єктів не мають партнера.

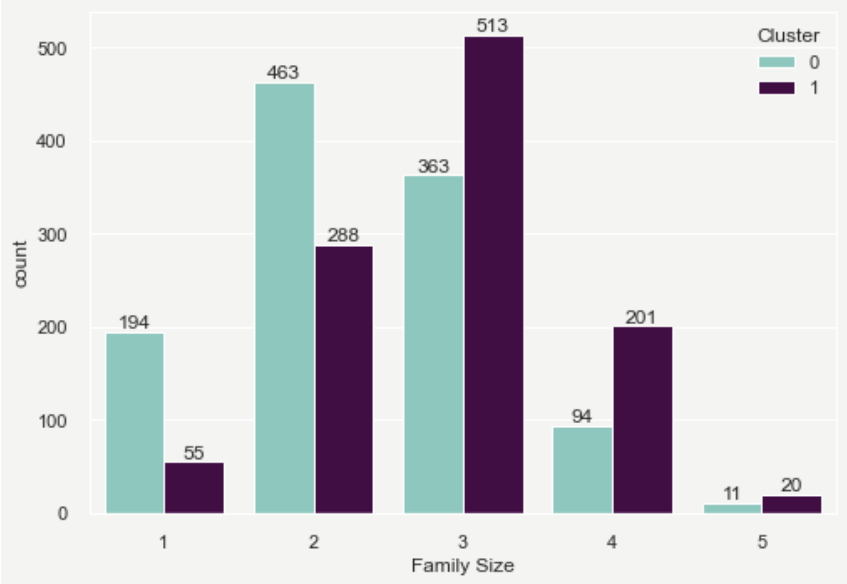

Рисунок 3.5.14 — Кількість об'єктів у кластерах за ознакою кількості членів у сім'ї

Із рисунку вище можемо зробити такі висновки: об’єкти Кластеру 0 здебільшого живуть у сім’ї, що складається із двох чи трьох осіб, третина ж живуть самі, а приблизно десята частина представників кластеру живуть у сім’ї

із чотирьох осіб; а об’єкти Кластеру 1 здебільшого проживають у сім’ї розміром від двох до трьох осіб.

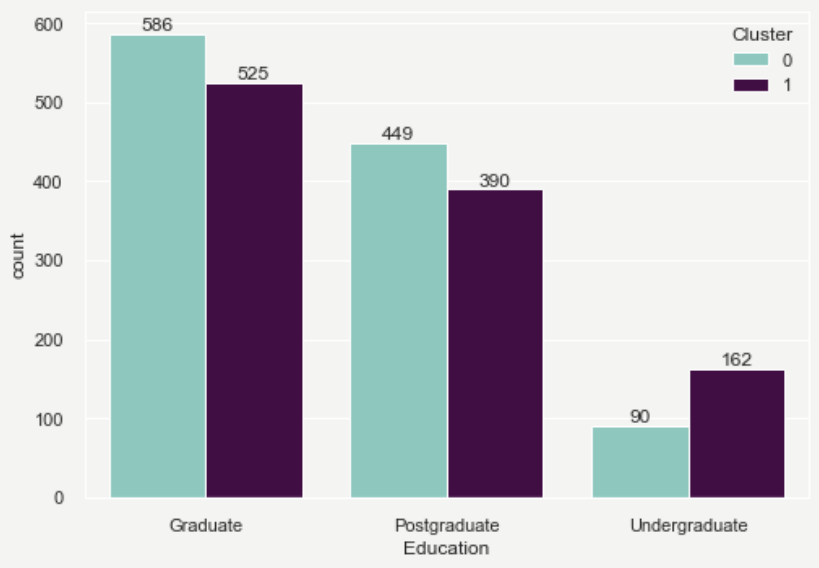

Рисунок 3.5.15 — Кількість об'єктів у кластерах за ознакою рівня освіти

За ознакою освіти кластери схожі. Переважна більшість мають вищу освіту.

Тепер, коли ми розглянули як розподілені об’єкти кластерів за окремими характеристиками, ми можемо створити узагальнюючий опис кластерів.

Кластер 0:

- Містить 1125 об'єктів
- Високий дохід, середній та високий рівень витрат
- Різних вікових груп
- Майже всі мають вищу освіту
- Більшість мають партнера
- Половина має одну дитину-підлітка
- Здебільшого у сім’ї від однієї до трьох осіб; трохи менше половини представників – батьки-одинаки
- Витрачають більші суми на усі види продуктів
- Здійснюють багато покупок через усі канали продажу (тобто зацікавлені у онлайн-покупках, покупках у магазині та покупках через каталог)

- Більше зацікавлені у акційних пропозиціях та купують більше акційних продуктів

Кластер 1:

- Містить 1077 об'єктів
- Низький дохід та низькі витрати
- Більшість мають партнера
- Різних вікових груп
- Майже всі мають вищу освіту
- Здебільшого мають одну чи дві дитини
- У сім'ї здебільшого від двох до чотирьох осіб
- Витрачають малі суми на усі види продуктів
- Здійснюють мало покупок через усі канали продажу
- Здійснюють небагато акційних покупок

ВИСНОВКИ

В ході виконання даної роботи були пройдені основні етапи обробки та кластерного аналізу даних. Для підготовки до кластеризації використовували розвідувальний аналіз даних, після якого була застосована сама кластеризація методами Варда, дальнього сусіда та k-середніх на даних, що характеризують купівельну поведінку об’єктів. Також була приділена увага методу зменшення розмірності даних – методу головних компонент, за використання якого ми змогли відобразити сформовані кластери у зменшеному просторі. Після кластеризації був проведений аналіз отриманих результатів, який здійснювався на основі отриманої візуалізації розподілів кластерів щодо окремих характеристик об’єктів.

Отримані в даній роботі результати можуть бути використані для планування кращих маркетингових стратегій для бізнесу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kaufman L. Finding Groups in Data: An Introduction to Cluster Analysis / L. Kaufman, P. J. Rousseeuw., 1990. – 342 с.
2. Cluster Analysis / B. S.Everitt, S. Landau, M. Leese, D. Stahl., 2011. – 330 с.
3. Calinski T. A dendrite method for cluster analysis / T. Calinski, J. Harabasz // Communication in Statistics / T. Calinski, J. Harabasz., 1974. – С. 1–27.
4. Clustering [Електронний ресурс] – Режим доступу до ресурсу: https://scikit-learn.org/stable/modules/clustering.html.
5. Aravind C. R. Exploring Clustering Algorithms: Explanation and Use Cases [Електронний ресурс] / C. R. Aravind. – 2021. – Режим доступу до ресурсу: https://neptune.ai/blog/clustering-algorithms.
6. Стеріна К. В. The features of cluster analysis for market segmentation / К. В. Стеріна // Науковий вісник Ужгородського національного університету / К. В. Стеріна., 2016. – (Міжнародні економічні відносини та світове господарство; вип. 7). – С. 115–117.

ДОДАТКИ

Додаток А

```
# імпортування бібліотек
import matplotlib
import matplotlib.pyplot as plt
from matplotlib import colors
from matplotlib.colors import ListedColormap

import matplotlib.pyplot as plt, numpy as np
import numpy as np
import pandas as pd
import datetime
from datetime import date
import seaborn as sns

from sklearn.preprocessing import StandardScaler, MinMaxScaler
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
from sklearn.cluster import AgglomerativeClustering
from sklearn.metrics import davies_bouldin_score
import scipy.cluster.hierarchy as shc

from kmodes.kmodes import KModes
from sklearn import metrics

from yellowbrick.cluster import KElbowVisualizer
from yellowbrick.cluster import SilhouetteVisualizer
from mpl_toolkits.mplot3d import Axes3D
from pyckmeans import MultiCKMeans

import warnings
import sys
if not sys.warnoptions:
    warnings.simplefilter("ignore")
np.random.seed(40)

# завантаження датасету
dataset = pd.read_csv("marketing.csv", sep=";")
print("Кількість рядків у датасеті:", len(dataset))
dataset.head(10)
dataset.info()

# Видаляємо пропущені значення
dataset = dataset.dropna()
print("Кількість рядків після вилучення пустих рядків із пропущеними значеннями
в колонці Income:", len(dataset))

dataset["Dt_Customer"] = pd.to_datetime(dataset["Dt_Customer"])
dates = dataset["Dt_Customer"]

# Створення колонки Customer_Days
days = []
td = np.datetime64('2022-05-17')
for i in dates:
    delta = td - i
    days.append(delta)
days_int = []
for i in days:
```

```
    d = i.days
    days_int.append(d)

dataset["Customer_Days"] = days_int
print("Категорії у колонці Marital_Status:\n",
dataset["Marital_Status"].value_counts(), "\n")
print("Категорії у колонці Education:\n", dataset["Education"].value_counts())
# Вік клієнта на сьогодні
dataset["Age"] = 2022-dataset["Year_Birth"]
# Загальні витрати
dataset["Spent"] = dataset["MntWines"]+ dataset["MntFruits"]+
dataset["MntMeatProducts"]+ dataset["MntFishProducts"]+
dataset["MntSweetProducts"]+ dataset["MntGoldProds"]
# Виділимо сімейний статус як Alone та Partner
dataset["Living_With"] = dataset["Marital_Status"].replace({"Married":"Partner",
"Together":"Partner", "Absurd":"Alone", "Widow":"Alone", "YOLO":"Alone",
"Divorced":"Alone", "Single":"Alone",})
# Кількість дітей у сім'ї клієнта
dataset["Children"] = dataset["Kidhome"]+ dataset["Teenhome"]
# Кількість членів у сім'ї
dataset["Family_Size"] = dataset["Living_With"].replace({"Alone": 1, "Partner":
2}) + dataset["Children"]
# Колонка, що вказує на те, чи має клієнт дітей
dataset["Parenthood"] = np.where(dataset.Children > 0, 'Parent', 'Not a Parent')
# Поділ рівнів освіти на 3 групи
dataset["Education"] = dataset["Education"].replace({"Basic": "Undergraduate",
"2n Cycle": "Undergraduate", "Graduation": "Graduate", "Master": "Postgraduate",
"PhD": "Postgraduate"})
# Для ясності перейменуємо деякі колонки
dataset = dataset.rename(columns={"MntWines": "Wines", "MntFruits": "Fruits",
"MntMeatProducts": "Meat", "MntFishProducts": "Fish", "MntSweetProducts":
"Sweets", "MntGoldProds": "Gold"})
# Видаляємо зайві колонки
to_drop = ["Marital_Status", "Dt_Customer", "Year_Birth", "ID"]
dataset = dataset.drop(to_drop, axis=1)
# статистичні показники датасету
dataset.describe()

# Для зображення деяких обраних колонок
# Вибираємо кольори
sns.set(rc={"axes.facecolor": "#f4f5f2", "figure.facecolor": "#f4f5f2"})

pallet1 = ["#edc951", "#ff6f69"]
cmap1 = matplotlib.colors.ListedColormap(pallet1)
pallet2 = ["#bd352d", "#f0a122"]
cmap2 = matplotlib.colors.ListedColormap(pallet2)
pallet3 = ["#611b17", "#d6a85c"]
cmap3 =  matplotlib.colors.ListedColormap(pallet3)

pallet4 = ["#3a8c7e", "#522d5c"]
cmap4 = matplotlib.colors.ListedColormap(pallet4)
pallet5 = ["#84d1c4", "#49054d"]
cmap5 = matplotlib.colors.ListedColormap(pallet5)
pallet6 = ["#04c2a1", "#472c4a"]
cmap6 = matplotlib.colors.ListedColormap(pallet6)

# Будуємо графіки з такими змінними
for_plotting = [ "Income", "Recency", "Age", "Spent", "Parenthood"]
plt.figure()
pp = sns.pairplot(dataset[for_plotting], hue="Parenthood", palette=(["#ff6f69",
"#96ceb4"]))
pp.fig.suptitle("Рисунок 1 – Попарні відношення деяких змінних: підмножина
даних")
```

```
plt.show()

# Видалення викидів за рахунок встановлення обмеження на ці дані
dataset = dataset[(dataset["Age"] < 90)]
dataset = dataset[((dataset["Income"] < 110000) & (dataset["Income"] > 2700))]

print("Кількість рядків у датасеті після видалення викидів:", len(dataset))

for_plotting = ["Income", "Recency", "Age", "Spent", "Parenthood"]
plt.figure()
pp = sns.pairplot(dataset[for_plotting], hue = "Parenthood",palette =
(["#ff6f69", "#96ceb4"]))
pp.fig.suptitle("Рисунок 2 — Попарні відношення деяких змінних після видалення
викидів")
plt.show()
dataset.describe()

# Створення набору даних з інформацією про купівельну поведінку клієнтів
selected_columns = dataset[["Wines","Fruits", "Meat", "Fish", "Sweets", "Gold"]]
d_purchasing = selected_columns.copy()

# Метод ліктя для знаходження кількості кластерів для методу k-середніх
elbow_m = KElbowVisualizer(KMeans(), k=7, timings = False, locate_elbow=False,
title = "Рисунок 3 - Метод ліктя для визначення кількості кластерів. Дані про
витрати")
elbow_m.fit(d_purchasing)
elbow_m.show()

# Метод силуету для визначення кількості кластерів
visualizer = KElbowVisualizer(KMeans(), k=(2,8), metric = 'silhouette', timings=
False, locate_elbow=False, title = 'Рисунок 4 - Метод силуету для визначення
кількості кластерів. Дані про витрати')
visualizer.fit(d_purchasing)
visualizer.show()

# Індекс Калінські-Харабаша для методу k-середніх
visualizer = KElbowVisualizer(KMeans(), k=(2,8), metric = 'calinski_harabasz',
timings= False, locate_elbow=False, title = 'Рисунок 5 - Індекс Калінські-
Харабаша для методу k-середніх. Дані про витрати')
visualizer.fit(d_purchasing)
visualizer.show()

# Індекс Девіса-Болдуіна для методу k-середніх
res = {}
for i in range(2, 8):
    kmeans = KMeans(n_clusters=i, random_state=55)
    labels = kmeans.fit_predict(d_purchasing)
    db_index = davies_bouldin_score(d_purchasing, labels)
    res.update({i: db_index})
plt.plot(list(res.keys()), list(res.values()), marker="D")
plt.xlabel("k")
plt.ylabel("Davies-Bouldin Score")
plt.title('Рисунок 6 - Індекс Девіса-Болдуіна для методу k-середніх. Дані про
витрати')
plt.show()

# Дендрограма для ієрархічної кластеризації
plt.figure(figsize=(10, 10))
plt.title("Рисунок 7 - Дендрограма. Дані про витрати")
dend = shc.dendrogram(shc.linkage(d_purchasing, method='ward'))

# Метод k-середніх
k_means = KMeans(n_clusters = 2, init = 'k-means++', random_state = 55)
```

```
# підганяємо модель і передбачаємо кластери
cl_kmeans = k_means.fit_predict(d_purchasing)
# Додаємо змінну Clusters_kmeans до вихідного фрейму даних
dataset["Clusters_kmeans"] = cl_kmeans
# Модель агломеративної кластеризації к = 2 методом Ворда
agc = AgglomerativeClustering(n_clusters=2, linkage = 'ward')
# підганяємо модель і передбачаємо кластери
cl_agc = agc.fit_predict(d_purchasing)
# Додаємо змінну Clusters_ward до вихідного фрейму даних.
dataset["Clusters_ward"]= cl_agc
# Модель агломеративної кластеризації к = 2 методом далекого сусіда
agc_compl = AgglomerativeClustering(n_clusters = 2, linkage = 'complete')
# підганяємо модель і передбачаємо кластери
cl_agc_compl = agc_compl.fit_predict(d_purchasing)
# Додаємо змінну Clusters_ward до вихідного фрейму даних.
dataset["Clusters_complete"]= cl_agc_compl

pca = PCA()
pca.fit(d_purchasing)
def add_value_label(x_list,y_list):
    for i in range(1, len(x_list)+1):
        a = round(y_list[i-1], 3)
        plt.annotate(a, (i-1,y_list[i-1]), c='w', ha="center",
bbox=dict(boxstyle='round', pad=0.2, fc='b', alpha=1))
cumsum_eigenv = np.cumsum(pca.explained_variance_ratio_)
plt.figure(figsize=(10, 10))
plt.bar(range(0, len(pca.explained_variance_ratio_)),
pca.explained_variance_ratio_,  label="Дисперсія проєкції", color = "#252d63")
add_value_label(range(0, len(pca.explained_variance_ratio_)),
pca.explained_variance_ratio_)
plt.step(range(0, len(cumsum_eigenv)), cumsum_eigenv, label="Кумулятивна сума
часток інформації(дисперсій)", color = "#252d63")
plt.xlabel("N компонент")
plt.ylabel("Частка інформації")
plt.legend(loc="center right")
plt.title("Рисунок 8 — Розподіл дисперсії між головними компонентами. Дані про
витрати")
plt.show()

pca = PCA(n_components=3)
pca.fit(d_purchasing)
PCA_purchasing = pd.DataFrame(pca.transform(d_purchasing),
columns=(["column1","column2", "column3"]))
PCA_purchasing.describe().T

# Проєкція даних за зменшеної розмірності
x = PCA_purchasing["column1"]
y = PCA_purchasing["column2"]
z = PCA_purchasing["column3"]

# Побудова
fig = plt.figure(figsize=(10,10))
p = fig.add_subplot(111, projection = "3d")
p.scatter(x, y, z, c="#252d63", marker="D" )
p.set_title("Рисунок 9 — Проєкція даних про витрати за зменшеної розмірності.
Дані про витрати")
plt.show()

# Побудова графіку із кластерами
x = PCA_purchasing["column1"]
y = PCA_purchasing["column2"]
z = PCA_purchasing["column3"]
```

```
fig = plt.figure(figsize=(10,10))
p = plt.subplot(111, projection = "3d", label = "bla")
scatter = p.scatter(x, y, z, s = 40, c = dataset["Clusters_kmeans"], marker =
'D', cmap = cmap1)
p.set_title("Рисунок 10 — Кластери сформовані методом k-середніх. Дані про
витрати")
legend1 = p.legend(*scatter.legend_elements(num=1), loc="upper right",
title="Cluster")
p.add_artist(legend1)
plt.show()

# Побудова графіку із кластерами сформованими методом Варда
fig = plt.figure(figsize = (10,10))
p = plt.subplot(111, projection = "3d", label = "bla")
scatter = p.scatter(x, y, z, s = 40, c = dataset["Clusters_ward"], marker ='D',
cmap = cmap2)
p.set_title("Рисунок 11 — Кластери сформовані методом Варда. Дані про витрати")
legend1 = p.legend(*scatter.legend_elements(num=1), loc="upper right",
title="Cluster")
p.add_artist(legend1)
plt.show()

# Побудова графіку із кластерами сформованими методом далекого сусіда
fig = plt.figure(figsize = (10,10))
p = plt.subplot(111, projection = "3d", label = "bla")
p.scatter(x, y, z, s = 40, c = dataset["Clusters_complete"], marker ='D', cmap =
cmap3)
p.set_title("Рисунок 12 — Кластери сформовані методом далекого сусіда. Дані про
витрати")
legend1 = p.legend(*scatter.legend_elements(num=1), loc="upper right",
title="Cluster")
p.add_artist(legend1)
plt.show()

# Створення набору даних, що містять інформацію про кількість покупок клієнтів з
різних каналів продажу
selected_col = dataset[["NumDealsPurchases", "NumWebPurchases",
"NumCatalogPurchases", "NumStorePurchases", "NumWebVisitsMonth"]]
d_place = selected_col.copy()

# Метод ліктя для знаходження кількості кластерів для методу k-середніх
elbow_m = KElbowVisualizer(KMeans(), timings = False, k=7, locate_elbow = False,
title = "Рисунок 13 - Метод ліктя для визначення кількості кластерів. Дані про
покупки")
elbow_m.fit(d_place)
elbow_m.show()

# Метод силуету для визначення кількості кластерів
visualizer = KElbowVisualizer(KMeans(), k=(2,8), metric = 'silhouette', timings=
False, locate_elbow = False, title = 'Рисунок 14 - Метод силуету для визначення
кількості кластерів. Дані про покупки')
visualizer.fit(d_place)
visualizer.show()

# Індекс Калінські-Харабаша для методу k-середніх
visualizer = KElbowVisualizer(KMeans(), k=(2,8), metric = 'calinski_harabasz',
timings= False, locate_elbow = False, title = 'Рисунок 15 - Індекс Калінські-
Харабаша для методу k-середніх. Дані про покупки')
visualizer.fit(d_place)
visualizer.show()

# Індекс Девіса-Болдуіна для методу k-середніх
res = {}
```

```
for i in range(2, 8):
    kmeans = KMeans(n_clusters=i, random_state=60)
    labels = kmeans.fit_predict(d_place)
    db_index = davies_bouldin_score(d_place, labels)
    res.update({i: db_index})
plt.plot(list(res.keys()), list(res.values()), marker="D")
plt.xlabel("k")
plt.ylabel("Davies-Bouldin Score")
plt.title('Рисунок 16 - Індекс Девіса-Болдуіна для методу k-середніх. Дані про покупки')
plt.show()

# Метод k-середніх
k_meansss = KMeans(n_clusters = 2, init = 'k-means++', random_state = 55)
# підганяємо модель і передбачаємо кластери
cl_kmeansss = k_meansss.fit_predict(d_place)
# Додаємо змінну Clusters_kmeans до вихідного фрейму даних
dataset["Clusters_kmeans_place"] = cl_kmeansss
# Модель агломеративної кластеризації к = 3 методом Ворда
agccc = AgglomerativeClustering(n_clusters=2, linkage = 'ward')
# підганяємо модель і передбачаємо кластери
cl_agccc = agccc.fit_predict(d_place)
# Додаємо змінну Clusters_ward до вихідного фрейму даних.
dataset["Clusters_ward_place"]= cl_agccc
# Модель агломеративної кластеризації к = 3 методом далекого сусіда
agc_complll = AgglomerativeClustering(n_clusters = 2, linkage = 'complete')
# підганяємо модель і передбачаємо кластери
cl_agc_complll = agc_complll.fit_predict(d_place)
# Додаємо змінну Clusters_ward до вихідного фрейму даних.
dataset["Clusters_complete_place"]= cl_agc_complll

pca_pl = PCA()
pca_pl.fit(d_place)
cumsum_eigenv = np.cumsum(pca_pl.explained_variance_ratio_)
plt.figure(figsize=(10, 10))
plt.bar(range(0, len(pca_pl.explained_variance_ratio_)),
pca_pl.explained_variance_ratio_,  label="Дисперсія проєкції", color =
"#252d63")
add_value_label(range(0, len(pca_pl.explained_variance_ratio_)),
pca_pl.explained_variance_ratio_)
plt.step(range(0, len(cumsum_eigenv)), cumsum_eigenv, label="Кумулятивна сума часток інформації(дисперсій)", color = "#252d63")
plt.xlabel("N компонент")
plt.ylabel("Частка інформації")
plt.legend(loc="center right")
plt.title("Рисунок 17 — Розподіл дисперсії між головними компонентами. Дані про покупки")
plt.show()

pca_pl = PCA(n_components=3)
pca_pl.fit(d_place)
PCA_place = pd.DataFrame(pca_pl.transform(d_place),
columns=(["column1","column2", "column3"]))
PCA_place.describe().T

# Проєкція даних за зменшеної розмірності
x = PCA_place["column1"]
y = PCA_place["column2"]
z = PCA_place["column3"]

# Побудова
fig = plt.figure(figsize=(10,10))
ppl = fig.add_subplot(111, projection = "3d")
```

```
ppl.scatter(x, y, z, c="#252d63", marker="D" )
ppl.set_title("Рисунок 18 — Проєкція даних про покупки за зменшеної
розмірності")
plt.show()

# Побудова графіку із кластерами
fig = plt.figure(figsize=(10,10))
p = plt.subplot(111, projection = "3d", label = "bla")
scatter = p.scatter(x, y, z, s = 40, c = dataset["Clusters_kmeans_place"],
marker = 'D', cmap = cmap4)
p.set_title("Рисунок 19 — Кластери сформовані методом k-середніх. Дані про
покупки")
legend1 = p.legend(*scatter.legend_elements(num=1), loc="upper right",
title="Cluster")
p.add_artist(legend1)
plt.show()

# Побудова графіку із кластерами сформованими методом Варда
fig = plt.figure(figsize = (10,10))
p = plt.subplot(111, projection = "3d", label = "bla")
scatter = p.scatter(x, y, z, s = 40, c = dataset["Clusters_ward_place"], marker
='D', cmap = cmap5)
p.set_title("Рисунок 20 — Кластери сформовані методом Варда. Дані про покупки")
legend1 = p.legend(*scatter.legend_elements(num=1), loc="upper right",
title="Cluster")
p.add_artist(legend1)
plt.show()

# Побудова графіку із кластерами сформованими методом далекого сусіда
fig = plt.figure(figsize = (10,10))
p = plt.subplot(111, projection = "3d", label = "bla")
p.scatter(x, y, z, s = 40, c = dataset["Clusters_complete_place"], marker ='D',
cmap = cmap6)
p.set_title("Рисунок 21 — Кластери сформовані методом далекого сусіда. Дані про
покупки")
legend1 = p.legend(*scatter.legend_elements(num=1), loc="upper right",
title="Cluster")
p.add_artist(legend1)
plt.show()

pl = sns.countplot(x = dataset["Clusters_kmeans"], palette = pallet1)
pl.bar_label(pl.containers[0])
pl.set(xlabel="Cluster")
pl.set_title("Рисунок 22 — Розподіл кластерів за методом k-середніх. Дані про
витрати")
plt.show()

pl = sns.countplot(x = dataset["Clusters_ward"], palette = pallet2)
pl.bar_label(pl.containers[0])
pl.set(xlabel="Cluster")
pl.set_title("Рисунок 23 — Розподіл кластерів за методом Варда. Дані про
витрати")
plt.show()

pl = sns.countplot(x = dataset["Clusters_complete"], palette = pallet3)
pl.bar_label(pl.containers[0])
pl.set(xlabel="Cluster")
pl.set_title("Рисунок 24 — Розподіл кластерів за методом далекого сусіда. Дані
про витрати")
plt.show()

pl = sns.countplot(x = dataset["Clusters_kmeans_place"], palette = pallet4)
pl.bar_label(pl.containers[0])
```

```
pl.set(xlabel="Cluster")
pl.set_title("Рисунок 25 — Розподіл кластерів за методом k-середніх. Дані про покупки")
plt.show()

pl = sns.countplot(x = dataset["Clusters_ward_place"], palette = pallet5)
pl.bar_label(pl.containers[0])
pl.set(xlabel="Cluster")
pl.set_title("Рисунок 26 — Розподіл кластерів за методом Варда. Дані про покупки")
plt.show()

pl = sns.countplot(x = dataset["Clusters_complete_place"], palette = pallet6)
pl.bar_label(pl.containers[0])
pl.set(xlabel="Cluster")
pl.set_title("Рисунок 27 — Розподіл кластерів за методом далекого сусіда. Дані про покупки")
plt.show()

pal = sns.scatterplot(data = dataset, x = dataset["Spent"], y =
dataset["Income"], hue = dataset["Clusters_kmeans"], palette = pallet1)
pal.set_title("Рисунок 28 — Розподіл кластерів за методом k-середніх за доходами й річними витратами. Дані про витрати")
plt.legend(loc="lower right", title="Cluster")
plt.show()

pal = sns.scatterplot(data = dataset, x = dataset["Spent"], y =
dataset["Income"], hue = dataset["Clusters_ward"], palette = pallet2)
pal.set_title("Рисунок 29 — Розподіл кластерів за методом Варда за доходами й річними витратами. Дані про витрати")
plt.legend(loc="lower right", title="Cluster")
plt.show()

pal = sns.scatterplot(data = dataset, x = dataset["Spent"], y =
dataset["Income"], hue = dataset["Clusters_kmeans_place"], palette = pallet4)
pal.set_title("Рисунок 30 — Розподіл кластерів за методом k-середніх за доходами й річними витратами. Дані про покупки")
plt.legend(loc="lower right", title="Cluster")
plt.show()

pal = sns.scatterplot(data = dataset, x = dataset["Spent"], y =
dataset["Income"], hue = dataset["Clusters_ward_place"], palette = pallet5)
pal.set_title("Рисунок 31 — Розподіл кластерів за методом Варда за доходами й річними витратами. Дані про покупки")
plt.legend(loc="lower right", title="Cluster")
plt.show()

# Створення змінної, що є сумою колонок, що відображають участь у маркетингових кампаніях
dataset["AcceptedAll"] = dataset["AcceptedCmp1"]+ dataset["AcceptedCmp2"]+
dataset["AcceptedCmp3"]+ dataset["AcceptedCmp4"]+ dataset["AcceptedCmp5"]

# Зображення розподілу по сумах
plt.figure()
pal = sns.countplot(x = dataset["AcceptedAll"], hue = dataset["Clusters_ward"],
palette = pallet2)
pal.set_title("Рисунок 32 — Участь у маркетингових кампаніях. Дані про витрати")
pal.set_xlabel("Count of accepted propositions")
pal.legend(title="Cluster")
pal.bar_label(pal.containers[0])
pal.bar_label(pal.containers[1])
plt.show()
```

```
# Зображення розподілу по сумах
plt.figure()
pal = sns.countplot(x = dataset["AcceptedAll"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
pal.set_title("Рисунок 33 — Участь у маркетингових кампаніях. Дані про покупки")
pal.set_xlabel("Count of accepted propositions")
pal.legend(title="Cluster")
pal.bar_label(pal.containers[0])
pal.bar_label(pal.containers[1])
plt.show()

plt.figure()
pal = sns.swarmplot(x = dataset["Clusters_ward"], y = dataset["Income"], color=
"#e8cbed", alpha=0.5 )
pal = sns.boxenplot(x = dataset["Clusters_ward"], y = dataset["Income"], palette
= pallet2)
pal.set(xlabel='Cluster')
pal.set_title("Рисунок 34 — Розподіл об'єктів у кластерах щодо річних доходів.
Дані про витрати")
plt.show()

plt.figure()
pal = sns.swarmplot(x = dataset["Clusters_ward_place"], y = dataset["Income"],
color= "#e8cbed", alpha=0.5 )
pal = sns.boxenplot(x = dataset["Clusters_ward_place"], y = dataset["Income"],
palette = pallet5)
pal.set(xlabel='Cluster')
pal.set_title("Рисунок 35 — Розподіл об'єктів у кластерах щодо річних доходів.
Дані про покупки")
plt.show()

plt.figure()
pal = sns.swarmplot(x = dataset["Clusters_ward"], y = dataset["Spent"], color=
"#e8cbed", alpha=0.5 )
pal = sns.boxenplot(x = dataset["Clusters_ward"], y = dataset["Spent"], palette
= pallet2)
pal.set(xlabel='Cluster')
pal.set_title("Рисунок 36 — Розподіл об'єктів у кластерах щодо річних витрат.
Дані про витрати")
plt.show()

plt.figure()
pal = sns.swarmplot(x = dataset["Clusters_ward_place"], y = dataset["Spent"],
color= "#e8cbed", alpha=0.5 )
pal = sns.boxenplot(x = dataset["Clusters_ward_place"], y = dataset["Spent"],
palette = pallet5)
pal.set(xlabel='Cluster')
pal.set_title("Рисунок 37 — Розподіл об'єктів у кластерах щодо річних витрат.
Дані про покупки")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Age"], y = dataset["Spent"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
plo.fig.suptitle("Рисунок 38 — Розподіл кластерів за віком та витратами. Дані
про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Age"], y = dataset["Spent"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
```

```
plo.fig.suptitle("Рисунок 39 — Розподіл кластерів за віком та витратами. Дані
про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Gold"], y = dataset["Income"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
plo.fig.suptitle("Рисунок 40 — Розподіл кластерів за доходами та витратами на
золото. Дані про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Gold"], y = dataset["Income"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
plo.fig.suptitle("Рисунок 41 — Розподіл кластерів за доходами та витратами на
золото. Дані про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Meat"], y = dataset["Income"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
plo.fig.suptitle("Рисунок 42 — Розподіл кластерів за доходами та витратами на
м'ясо. Дані про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Meat"], y = dataset["Income"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
plo.fig.suptitle("Рисунок 43 — Розподіл кластерів за доходами та витратами на
м'ясо. Дані про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Wines"], y = dataset["Income"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
plo.fig.suptitle("Рисунок 44 — Розподіл кластерів за дохлдами та витратами на
вино. Дані про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Wines"], y = dataset["Income"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
plo.fig.suptitle("Рисунок 45 — Розподіл кластерів за доходами та витратами на
вино. Дані про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Sweets"], y = dataset["Income"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
```

```
plo.fig.suptitle("Рисунок 46 — Розподіл кластерів за доходом та витратами на
солодощі. Дані про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Sweets"], y = dataset["Income"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
plo.fig.suptitle("Рисунок 47 — Розподіл кластерів за доходом та витратами на
солодощі. Дані про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Fruits"], y = dataset["Income"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
plo.fig.suptitle("Рисунок 48 — Розподіл кластерів за доходами та витратами на
фрукти. Дані про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Fruits"], y = dataset["Income"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
plo.fig.suptitle("Рисунок 49 — Розподіл кластерів за доходами та витратами на
фрукти. Дані про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Fish"], y = dataset["Income"], hue =
dataset["Clusters_ward"], kind = "scatter", palette = pallet2)
plo.fig.suptitle("Рисунок 50 — Розподіл кластерів за доходами та витратами на
рибу. Дані про витрати")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
plo = sns.jointplot(x = dataset["Fish"], y = dataset["Income"], hue =
dataset["Clusters_ward_place"], kind = "scatter", palette = pallet5)
plo.fig.suptitle("Рисунок 51 — Розподіл кластерів за доходами та витратами на
рибу. Дані про покупки")
plo.ax_joint.legend_._visible=False
plo.fig.legend(loc='center right', title="Cluster")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumWebPurchases"], hue =
dataset["Clusters_ward"], palette = pallet2)
plt.title("Рисунок 52 — Кількість об'єктів у кластерах за ознакою онлайн
покупок. Дані про витрати")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumWebPurchases"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 53 — Кількість об'єктів у кластерах за ознакою онлайн
```

```
покупок. Дані про покупки")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumStorePurchases"], hue =
dataset["Clusters_ward"], palette = pallet2)
plt.title("Рисунок 54 — Кількість об'єктів у кластерах за ознакою покупок в
магазині. Дані про витрати")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumStorePurchases"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 55 — Кількість об'єктів у кластерах за ознакою покупок в
магазині. Дані про покупки")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumCatalogPurchases"], hue =
dataset["Clusters_ward"], palette = pallet2)
plt.title("Рисунок 56 — Кількість об'єктів у кластерах за ознакою покупок через
каталог магазину. Дані про витрати")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumCatalogPurchases"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 57 — Кількість об'єктів у кластерах за ознакою покупок через
каталог магазину. Дані про покупки")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumDealsPurchases"], hue =
dataset["Clusters_ward"], palette = pallet2)
plt.title("Рисунок 58 — Кількість об'єктів у кластерах за ознакою кількості
акційних покупок. Дані про витрати")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["NumDealsPurchases"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 59 — Кількість об'єктів у кластерах за ознакою кількості
акційних покупок. Дані про покупки")
plt.legend(title = "Cluster", loc="upper right")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["Kidhome"], hue = dataset["Clusters_ward_place"],
palette = pallet5)
plt.title("Рисунок 60 — К-сть об'єктів у кластерах за ознакою кількості малих
дітей у сім'ї")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.show()

plt.figure()
```

```
c = sns.countplot(x = dataset["Teenhome"], hue = dataset["Clusters_ward_place"],
palette = pallet5)
plt.title("Рисунок 61 — Кількість об'єктів у кластерах за ознакою кількості
підлітків у сім'ї")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.show()

plt.figure()
c = sns.countplot(x = dataset["Children"], hue = dataset["Clusters_ward_place"],
palette = pallet5)
plt.title("Рисунок 62 — Кількість об'єктів у кластерах за ознакою кількості
дітей у сім'ї")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.show()

plt.figure()
c = sns.countplot(x = dataset["Family_Size"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 63 — Кількість об'єктів у кластерах за ознакою кількості
членів у сім'ї")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.xlabel("Family Size")
plt.show()

plt.figure()
c = sns.countplot(x = dataset["Parenthood"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 64 — Кількість об'єктів у кластерах за ознакою батьківства")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.show()

plt.figure()
c = sns.countplot(x = dataset["Education"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 65 — Кількість об'єктів у кластерах за ознакою рівня освіти")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.show()

plt.figure()
c = sns.countplot(x = dataset["Living_With"], hue =
dataset["Clusters_ward_place"], palette = pallet5)
plt.title("Рисунок 66 — Кількість об'єктів у кластерах за ознакою наявності
партнера")
plt.legend(title = "Cluster")
plt.bar_label(c.containers[0])
plt.bar_label(c.containers[1])
plt.xlabel("Living With")
plt.show()
```